

DISSERTAÇÃO DE MESTRADO

**MODELAGEM, NAVEGAÇÃO  
E LOCALIZAÇÃO DE ROBÔ SUBAQUÁTICO  
UTILIZANDO FILTROS NÃO PARAMÉTRICOS**

**Nathan Costa Alves Souza**

Brasília, novembro de 2014

**UNIVERSIDADE DE BRASÍLIA**

FACULDADE DE TECNOLOGIA

**UNIVERSIDADE DE BRASÍLIA  
FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**MODELAGEM, NAVEGAÇÃO E LOCALIZAÇÃO DE ROBÔ  
SUBAQUÁTICO UTILIZANDO FILTROS NÃO  
PARAMÉTRICOS**

**NATHAN COSTA ALVES SOUZA**

**ORIENTADORA: DIANNE MAGALHÃES VIANA  
COORIENTADORA: CARLA MARIA CHAGAS E  
CAVALCANTE KOIKE**

**DISSERTAÇÃO DE MESTRADO EM SISTEMAS  
MECATRÔNICOS**

**PUBLICAÇÃO: ENM.DM – 78/14  
BRASÍLIA/DF: NOVEMBRO - 2014**

## FICHA CATALOGRÁFICA

Ficha catalográfica elaborada pela Biblioteca Central da Universidade de Brasília. Acervo 1020780.

Souza, Nathan Costa Alves.  
S729m Modelagem, navegação e localização de robô subaquático utilizando filtros não paramétricos / Nathan Costa Alves Souza. -- 2014.  
xi, 117 f. : il. ; 30 cm.

Dissertação (mestrado) - Universidade de Brasília, Faculdade de Tecnologia, Departamento de Engenharia Mecânica, 2014.

Orientação: Dianne Magalhães Vianna.

Inclui bibliografia.

1. Robôs. 2. Robótica. 3. Detectores. I. Viana, Dianne Magalhães. II. Título.

CDU 621.865.8

### REFERÊNCIA BIBLIOGRÁFICA

SOUZA, N. C. A. (2014). MODELAGEM, NAVEGAÇÃO E LOCALIZAÇÃO DE ROBÔ SUBAQUÁTICO UTILIZANDO FILTROS NÃO PARAMÉTRICOS, Dissertação de Mestrado em Sistemas Mecatrônicos, Publicação ENM.DM-78/14, Departamento de Engenharia Mecânica, Faculdade de Tecnologia, Universidade de Brasília, Brasília, DF, 117p.

### CESSÃO DE DIREITOS

AUTOR: Nathan Costa Alves Souza

TÍTULO: MODELAGEM, NAVEGAÇÃO E LOCALIZAÇÃO DE ROBÔ SUBAQUÁTICO UTILIZANDO FILTROS NÃO PARAMÉTRICOS.

GRAU: Mestre ANO: 2014

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desse trabalho de conclusão de curso pode ser reproduzida sem autorização por escrito do autor.

---

Nathan Costa Alves Souza  
SMPW Quadra 12 Conj.1 Lote 6 Casa B  
Park Way  
CEP 71741-201 - Brasília - DF – Brasil

UNIVERSIDADE DE BRASÍLIA  
Faculdade de Tecnologia

DISSERTAÇÃO DE MESTRADO

**MODELAGEM, NAVEGAÇÃO  
E LOCALIZAÇÃO DE ROBÔ SUBAQUÁTICO  
UTILIZANDO FILTROS NÃO PARAMÉTRICOS**

**Nathan Costa Alves Souza**

*Relatório submetido ao Departamento de Engenharia  
Mecânica como requisito parcial para obtenção  
do grau de Mestre em Sistemas Mecatrônicos*

**Banca Examinadora**

Prof. Dianne Magalhães Viana, ENM/UnB

*Orientadora*

\_\_\_\_\_

Prof. Eugênio Libório Feitosa Fortaleza, ENM/UnB

*Examinador interno*

\_\_\_\_\_

Prof. Márcio Yamamoto, Poli/USP

*Examinador externo*

\_\_\_\_\_

## **Dedicatória**

*Dedico meu trabalho ao meu pai, a minha mãe e a minha irmã, sempre presentes na minha vida, em que diante de uma situação aparentemente impossível e com condições assustadoras, olham para mim e dizem "Se vira amigo!". Tão logo aprendi que eles já sabiam do meu futuro, e que dele eu voltaria sorridente e contando as aventuras que tive ao encarar inúmeros desafios. Foi assim que eles me deram o meu maior presente: Confiança.*

*Nathan Costa Alves Souza*

## **Agradecimentos**

*Primeiramente a Deus por ter me guiado pelo caminho não tão reto nesta aventura que chamo de vida.*

*À minha família, que sempre está ao meu lado me apoiando e acreditando no meu esforço.*

*Aos meus amigos, que me ajudaram em todos os momentos, seja nas correções gramaticais, seja na companhia nos momentos em que precisei de distração.*

*Às minhas orientadoras, pelo suporte no pouco tempo que lhes coube e pela paciência que tiveram com meu ritmo de trabalho.*

*A essa Universidade e seu corpo docente, que me mostraram os caminhos do conhecimento, com os quais sempre posso contar como guias no futuro.*

*Ao programa de recursos humanos da Petrobrás (PRH-PB 223), que apoiou com o financiamento do projeto e dos meios para fazer deste trabalho uma realidade.*

*E a todos que direta ou indiretamente fizeram parte desta etapa de vida, o meu muito obrigado.*

*Nathan Costa Alves Souza*

---

## RESUMO

Com a expansão da barreira de exploração nas atividades petrolíferas no mar, as empresas estão utilizando cada vez mais veículos subaquáticos em operações offshore. A substituição de mergulhadores de profundidades por UUVs (Unmanned Underwater Vehicles) é devido não somente à capacidade de atuar em profundidades maiores que 300 metros, mas também em operações de grande duração e menor risco nas atividades de instalação e manutenção de infraestrutura subaquática. No entanto, os UUVs necessitam superar as outras duas dificuldades técnicas apresentadas na exploração offshore: as condições prevalecentes no clima e a invisibilidade das operações no mar. Essas condições interferem na eficácia de navegação dos veículos aquáticos, principalmente em UUVs que são dependentes das leituras dos seus sensores para funcionarem como projetados. Quanto mais programação e processamentos dos múltiplos sensores, melhor fica para que o sistema (ou operador no caso de ROVs) consiga navegar bem no ambiente subaquático. Porém, cada sensor capta ruído, processando de forma limitada a informação. Isso em conjunto com a elevada dinâmica não-linear e invisibilidade no mar faz com que não seja possível garantir a navegação ou operação confiável dos veículos subaquáticos baseando-se somente na leitura dos sensores. A metodologia apresentada neste trabalho tem o objetivo de possibilitar a implementação de navegação autônoma, manipulando as informações captadas pelos sensores e a incerteza de atuação do modelo não-linear do veículo em um sistema de tratamento por uso de um filtro probabilístico. Juntamente com algoritmos de planejamento de trajetória em grades de ocupação, esta metodologia é ensaiada em um veículo subaquático autônomo com resultados que demonstram seu pleno funcionamento em um caso de localização global em ambiente subaquático não controlado.

---

## ABSTRACT

With the expansion of the boundaries of oil and gas exploration at sea, the industry is using more underwater vehicles in offshore operations than divers. Replacing professional divers by UUVs (Unmanned Underwater Vehicles) is due not only to its capacity to operate in depths greater than 300 meters, but also to lower the risk and time it can stay underwater in installation and maintenance of underwater infrastructure. However, UUVs need to overcome other two technical difficulties presented in offshore exploration: the adverse conditions in marine climate and the invisibility of operations at sea. These conditions affect the efficiency of navigation of marine vessels, especially of UUVs, which are dependent of the readings from its sensors to function as designed. The more information that are inferred from the readings of multiple sensor, the better it is for the system (or for the pilot in the case of ROVs) to plan the navigation in underwater environments. However, in each sensor reading, noise is captured during the process, which interferes the accuracy of the reading. The noise, together with the high nonlinear dynamics and the invisibility at sea makes it difficult to guarantee reliable navigation and operation of underwater vehicles based solely on sensor reading. The procedure presented in this paper aims to enable implementation of autonomous navigation by manipulating the information captured by sensors and by the nonlinear model actuation of the vehicle to infer position through use of a probabilistic filter. Along with path planning algorithms in occupancy grids, this methodology is tested on a autonomous underwater vehicle with results that demonstrate its full operation in a global location problem in a uncontrolled underwater environment.



# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
1.1	APRESENTAÇÃO	1
1.2	CONTEXTUALIZAÇÃO	1
1.3	OBJETIVOS	9
1.4	ESTRUTURA DO DOCUMENTO	10
<b>2</b>	<b>MODELAGEM</b>	<b>11</b>
2.1	INTRODUÇÃO	11
2.2	SISTEMA DE COORDENADAS	11
2.3	CINEMÁTICA	12
2.4	DINÂMICA DE CORPO RÍGIDO	13
2.5	HIDRODINÂMICA	17
2.6	DINÂMICA DO VEÍCULO SUBAQUÁTICO	26
<b>3</b>	<b>LOCALIZAÇÃO</b>	<b>28</b>
3.1	INTRODUÇÃO	28
3.2	TIPOS DE LOCALIZAÇÃO	30
3.3	ROBÓTICA PROBABILÍSTICA	31
3.4	BASE TEÓRICA - FILTRO DE BAYES	34
3.5	FILTRO HISTOGRAMA	36
3.6	FILTRO DE PARTÍCULAS	38
3.7	SUMÁRIO	45
<b>4</b>	<b>PLANEJAMENTO DE TRAJETÓRIA</b>	<b>46</b>
4.1	INTRODUÇÃO	46
4.2	FUNÇÃO DE EXPANSÃO	49
4.3	A-STAR	51
4.4	DYNAMIC PROGRAMMING	52
<b>5</b>	<b>DESENVOLVIMENTO</b>	<b>54</b>
5.1	INTRODUÇÃO	54
5.2	METODOLOGIA DO TRABALHO	54
5.3	MODELAGEM DE UM VEÍCULO SUBAQUÁTICO	60
5.4	MODELO DE PERCEPÇÃO	81
5.5	DIRETRIZES	82
5.6	SISTEMA IMPLEMENTADO	84
<b>6</b>	<b>RESULTADOS EXPERIMENTAIS</b>	<b>87</b>
6.1	INTRODUÇÃO	87

6.2	CENÁRIO .....	87
6.3	FLUXOGRAMA DO ENSAIO .....	89
6.4	EXPERIMENTO .....	90
<b>7</b>	<b>CONCLUSÕES.....</b>	<b>98</b>
7.1	SÍNTESE DO TRABALHO REALIZADO .....	98
7.2	PERSPECTIVAS DE TRABALHOS FUTUROS .....	99
	<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>101</b>
	<b>ANEXOS .....</b>	<b>104</b>
<b>I</b>	<b>DADOS TÉCNICOS DO UNBOT.....</b>	<b>105</b>
I.1	DADOS TÉCNICOS DOS EQUIPAMENTOS DO ROV .....	105
I.2	ENTRADAS DAS GEOMETRIAS DE MASSA ADICIONADA (CAP. 2) .....	109
I.3	MATRIZES DO MODELO ROV .....	115
<b>II</b>	<b>PLANTAS DO ROV "UNBOT" .....</b>	<b>117</b>

# LISTA DE FIGURAS

1.1	Equipamento <i>Rebreather</i> . .....	2
1.2	Mergulhadores com umbilical de suporte de vida. ....	2
1.3	Exosuit. ....	3
1.4	O OCROV VideoRAY Pro IV.....	4
1.5	O ROV Classe Trabalhador Magellan 725 da Oceaneering. ....	5
1.6	O ROV Classe Especial MV Nico de escavação submarina para instalação de turbinas eólicas ( <i>offshore</i> ). ....	5
1.7	O AUV SeaBED em uma expedição da Woods Hole Oceanographic Institute.....	6
1.8	O HROV Nereus da Woods Hole Oceanographic Institute. No meio a configuração AUV e à direita a configuração ROV.....	7
1.9	Diferentes condições de visibilidade no ambiente subaquático. À esquerda exemplo de visibilidade boa em uma piscina de treino de mergulho. À direita exemplo de visibilidade baixa encontrada no Lago Paranoá, Brasília. ....	8
1.10	Etapas para obter navegação autônoma em UUVs .....	9
1.11	Veículo exemplo do trabalho, OCROV "UnBot" .....	10
2.1	Veículo com um plano de simetria .....	19
2.2	Veículo com dois planos de simetria.....	20
2.3	Veículo com axisimetria ou simetria rotacional em um eixo.....	21
2.4	Veículo com axisimetria ou simetria rotacional em dois eixos .....	22
2.5	Veículo com axisimetria ou simetria rotacional nos três eixos .....	23
2.6	Veículo com três planos de simetria .....	23
3.1	Navegação comprometida pela imprecisão do sensor.....	29
3.2	Localização Global em um terreno unidimensional - com incerteza representada em distribuição uniforme.....	32
3.3	Distribuição alterada com medição de sensor.....	32
3.4	Distribuição alterada com mudança de posição do robô. ....	33
3.5	Distribuição da confiança ao identificar um segundo marcador. Pico maior concentrado na posição estimada do robô. ....	33
3.6	Aumento dos picos de confiança ao fazer uma nova medição do marcador. ....	34
3.7	Localização por malha - Filtro histograma com distribuição ampliada para 100 células no vetor posição no terreno.....	36
3.8	Tempo médio requerido do CPU para localização global em função da resolução da célula na matriz discreta, com curvas utilizando sensores de ultrassom e LIDAR ( <i>Light Detection and Ranging</i> ). ....	37
3.9	Ilustração da incerteza de localização do robô subaquático com partículas distribuídas em posições e orientações aleatórias pelo terreno bidimensional.....	38
3.10	Ilustração das partículas após movimento. ....	39

3.11	Comparação da medição prevista de uma partícula com o robô real.....	40
3.12	Ilustração das partículas com pesos variados .....	41
3.13	Concentração das partículas redistribuídas após locomoção e medição via sonar. ....	41
3.14	Nova redistribuição após movimento e sonar. ....	42
3.15	Localização global resolvida para posição e orientação.....	42
3.16	Ilustração da roda de reamostragem baseado no exemplo descrito. ....	43
3.17	Equações das funções <i>Atualização</i> e <i>Previsão</i> .....	44
4.1	Robô em terreno conhecido com múltiplas soluções de rota.....	46
4.2	Terreno exemplo para planejamento. ....	47
4.3	Modelagem dos movimentos do "UnBot". ....	47
4.4	Solução analítica do problema de planejamento no terreno exemplo. ....	48
4.5	Solução analítica do problema de planejamento com função de custo modificado. ....	48
4.6	Terreno exemplo para planejamento de rota. ....	49
4.7	Procedimento de cálculo da função de expansão. ....	50
4.8	Resultado final do cálculo da função de expansão e rota ao alvo encontrado. ....	50
4.9	Procedimento da cálculo do <i>valor f</i> pelo algoritmo <i>A-star</i> . ....	51
4.10	Matrizes expansão e heurística resultando na trajetória ótima via <i>A-star</i> . ....	52
4.11	Matriz gerada da função valor aplicada no terreno. ....	53
4.12	Série de ações em todas as posições no terreno resultante do algoritmo <i>Dynamic Programming</i> .....	53
5.1	Os três primitivos da robótica. ....	55
5.2	Fluxograma do Paradigma Hierárquico. ....	56
5.3	Fluxograma do Paradigma Reativo.....	56
5.4	Fluxograma do Paradigma Híbrido.....	57
5.5	Etapas para obter primitivos. ....	57
5.6	Etapas da metodologia implementada no trabalho. ....	59
5.7	Modelo Original do <i>OpenROV</i> (à esquerda) e Modelo modificado e exemplo deste trabalho (à direita). ....	61
5.8	Vista Frontal do ROV.....	62
5.9	Vista Isométrica do ROV .....	63
5.10	Vista de topo do ROV .....	63
5.11	Vista Traseira do ROV .....	63
5.12	Sistema de coordenadas e localização dos componentes .....	64
5.13	Dados de centro de massa e inércia CAD - Solidworks. ....	65
5.14	Massa adicionada do cilindro (à esquerda) e do cilindro elipsoidal (à direita) no AMass. ....	69
5.15	Geometria do ROV no AMass.....	71
5.16	Amortecimento hidrodinâmico em baixas e altas velocidades do veículo subaquático. ....	73
5.17	Escoamento CFD em afundamento realizado no SolidWorks Flow 2013 com domínio exposto	74
5.18	Gráficos do Amortecimento Hidrodinâmico em avanço e joga positivo e negativo. ....	75
5.19	Gráficos do Amortecimento Hidrodinâmico em deriva e arfagem positiva e negativa.....	76
5.20	Gráficos do Amortecimento Hidrodinâmico em afundamento e guinada positiva e negativa. .	77

5.21	Programa <i>Mission Planner</i> - Etapas para planejamento de trajetória .....	83
5.22	Programa <i>Mission Planner</i> - Resultados do <i>A-star</i> (à esquerda) e <i>Dynamic Programming</i> (à direita) .....	84
5.23	Sistema de navegação autônoma aplicado no ROV "UnBot" .....	85
6.1	Cenário do Ensaio. ....	87
6.2	Mapa montado pelo programa <i>Mission Planner</i> . ....	88
6.3	Foto do Cabo umbilical do ROV. ....	88
6.4	Fluxograma do algoritmo implementado no Ensaio de Localização. ....	89
6.5	Modelos de Movimento e de Percepção do Ensaio.....	91
6.6	Inicialização da crença do veículo no meio. ....	92
6.7	Primeira Célula de Análise no Ensaio de Localização.....	92
6.8	Atualização da crença após processamento das medições e movimentos até a célula 1.....	93
6.9	Segunda Célula de Análise no Ensaio de Localização.....	94
6.10	Terceira Célula de Análise no Ensaio de Localização. ....	94
6.11	Atualização da crença após processamento das medições e movimentos pelas células 2 e 3..	95
6.12	Quarta Célula de Análise no Ensaio de Localização. ....	95
6.13	Quinta Célula de Análise no Ensaio de Localização. ....	96
6.14	Sexta Célula de Análise no Ensaio de Localização.....	96
6.15	Atualização da crença após processamento das medições e movimentos pelas células 4, 5 e 6. ....	97
6.16	Trajetória e diretrizes resultantes do algoritmo <i>A-star</i> . ....	97
7.1	Fluxograma do algoritmo implementado no Ensaio de Localização. ....	99
I.1	Motor WayPoint - Propulsor.....	105
I.2	Esc Turnigy Brushless com reversível.....	105
I.3	Mini-Maestro .....	106
I.4	Placa Raspberry Pi .....	107
I.5	Rpi Camera .....	107
I.6	IMU Razor 9DOF .....	108
I.7	Entradas da geometria do veículo na Fig. 2.1, pg. 19 .....	109
I.8	Entradas da geometria do veículo na Fig. 2.2, pg. 20 .....	110
I.9	Entradas da geometria do veículo na Fig. 2.3, pg. 21 .....	111
I.10	Entradas da geometria do veículo na Fig. 2.4, pg. 22 .....	112
I.11	Entradas da geometria do veículo na Fig. 2.5, pg. 23 .....	113
I.12	Entradas da geometria do veículo na Fig. 2.6, pg. 23 .....	114

## LISTA DE TABELAS

5.1	Principais características do ROV "UnBot" .....	61
5.2	Materiais selecionados para modelagem dos componente do ROV .....	62
5.3	Dados de Corpo Rígido .....	65
5.4	Dados de Entrada para Cilindros Circulares .....	70
5.5	Dados de Entrada para Cilindros Elipsoidais .....	70
5.6	Dados dos termos da Massa Adicionada.....	78
5.7	Dados de força de restabelecimento .....	79
I.1	Especificações do Motor WayPoint 2213-0760 .....	105
I.2	Especificações da Esc .....	106
I.3	Especificações do Mini-Maestro.....	106
I.4	Especificações do Raspberry Pi .....	107
I.5	Especificações do Rpi Camera .....	108
I.6	Especificações do IMU.....	108

# LISTA DE SÍMBOLOS

## Matrizes

$C_A$	Matriz dos termos de Coriolis e centrípetos da massa adicionada
$C_{RB}$	Matriz dos termos de Coriolis e centrípetos do corpo rígido
$D_L$	Matriz dos termos lineares de amortecimento viscoso
$d_q(v_r)$	Matriz dos termos quadráticos de amortecimento viscoso
$D_{v_r}$	Matriz de Amortecimento viscoso relativo a velocidade referencial
$g_{\eta_2}$	Matriz das Forças e Momentos de restabelecimento
$I$	Matriz de Momento de inércia do veículo
$J_1$	Jacobiano da transformada entre os referenciais de coordenadas dos vetores lineares
$J_2$	Jacobiano da transformada entre os referenciais de coordenadas dos vetores angulares
$M_A$	Matriz da massa adicionada
$M_{massa}$	Matriz de massa do veículo
$M_{RB}$	Matriz do corpo rígido
$R_{x,\phi}$	Matriz de rotação segundo eixo $x$
$R_{y,\theta}$	Matriz de rotação segundo eixo $y$
$R_{z,\psi}$	Matriz de rotação segundo eixo $z$
$0_{3 \times 3}$	Matriz nula de dimensão 3x3
$S(r_{CG})$	Matriz Antissimétrica do vetor $r_{CG}$

## Vetores

$I_{ii}$	Momento de inércia em relação ao eixo $i$
$I_{ij}$	Momento de inércia em relação ao plano formado pelos eixos $i$ e $j$
$\eta_1$	Vetor de posição linear no referencial fixo na Terra
$\eta_2$	Vetor de posição angular no referencial fixo na Terra
$\dot{\eta}_1$	Vetor de velocidades lineares no referencial fixo na Terra
$\dot{\eta}_2$	Vetor de velocidades angulares no referencial fixo na Terra
$\tau_1$	Vetor de forças segundo os eixos lineares do referencial fixo ao veículo
$\tau_2$	Vetor de forças segundo os eixos angulares do referencial fixo ao veículo
$\tau_A$	Vetor de forças de atuação do veículo
$v_1$	Vetor de velocidades lineares no referencial fixo ao veículo
$v_2$	Vetor de velocidades angulares no referencial fixo ao veículo
$r_{CG}$	Vetor posição do CG do veículo
$r_{CF}$	Vetor posição do CF do veículo
$BG$	Vetor Distância entre o CG e CF

## Variáveis

$A_{ij}$	Área projetada no plano formado pelos eixo $i$ e $j$	[m <sup>2</sup> ]
$C_{di}$	Coefficiente de amortecimento viscoso seguindo o eixo $i$	[]
$W$	Peso do veículo	[N]
$B$	Empuxo do veículo	[N]
$BG_x$	Distância entre o CG e CF no eixo $x$	[m]
$BG_y$	Distância entre o CG e CF no eixo $y$	[m]
$BG_z$	Distância entre o CG e CF no eixo $z$	[m]
$x_{CG}$	Posição do CG no eixo $x$ do veículo	[m]
$y_{CG}$	Posição do CG no eixo $y$	[m]
$z_{CG}$	Posição do CG no eixo $z$	[m]
$x_{CF}$	Posição do CF no eixo $x$	[m]
$y_{CF}$	Posição do CF no eixo $y$	[m]
$z_{CF}$	Posição do CF no eixo $z$	[m]
$x$	Posição no eixo $x$ no referencial fixo na Terra	[m]
$y$	Posição no eixo $y$ no referencial fixo na Terra	[m]
$z$	Posição no eixo $z$ no referencial fixo na Terra	[m]
$\phi$	Orientação no eixo $x$ no referencial fixo na Terra	[rad]
$\theta$	Orientação no eixo $y$ no referencial fixo na Terra	[rad]
$\psi$	Orientação no eixo $z$ no referencial fixo na Terra	[rad]
$u$	Velocidade linear no eixo $x$	[m/s]
$v$	Velocidade linear no eixo $y$	[m/s]
$w$	Velocidade linear no eixo $z$	[m/s]
$p$	Velocidade angular no eixo $x$	[rad/s]
$q$	Velocidade angular no eixo $y$	[rad/s]
$r$	Velocidade angular no eixo $z$	[rad/s]
$X$	Força exercida no eixo $x$	[N]
$Y$	Força exercida no eixo $y$	[N]
$Z$	Força exercida no eixo $z$	[N]
$K$	Momento exercido no eixo $x$	[N]
$M$	Momento exercido no eixo $y$	[N]
$N$	Momento exercido no eixo $z$	[N]
$\rho$	Densidade	[kg/m <sup>3</sup> ]

## Constantes

$g$	Gravidade	[m/s <sup>2</sup> ]
-----	-----------	---------------------



## Abreviações

$s(\theta)$	Seno do ângulo theta
$c(\theta)$	Cosseno do ângulo theta
$t(\theta)$	Tangente do ângulo theta

## Siglas

ABNT	Associação Brasileira de Normas Técnicas
CG	Centro de Gravidade
CF	Centro de Flutuação ou Empuxo
NED	Sistema de coordenada "North-East-Down"
SNAME	"Society of Naval Architects & Marine Engineers"

# 1 INTRODUÇÃO

## 1.1 APRESENTAÇÃO

Este capítulo apresenta uma contextualização do uso de veículos subaquáticos, abordando as motivações para a produção desta dissertação, assim como o problema proposto, objetivo a ser alcançado, etapas do projeto e a organização do texto.

## 1.2 CONTEXTUALIZAÇÃO

Ao longo dos tempos os oceanos demonstraram importância na história da humanidade. Sua presença é inigualável, pois os oceanos cobrem 2/3 da superfície terrestre e a maioria das grandes cidades situam-se nos litorais pelas vantagens oferecidas no transporte, no comércio e na obtenção de alimentos.

Na história contemporânea, a exploração de óleo e gás nos oceanos aumentou mais ainda a necessidade de conhecer este terreno ainda pouco explorado. Por causa da grande extensão e profundidade, as tecnologias de exploração marítima/subaquática são ferramentas necessárias para colher informações e realizar operações onde os seres humanos não podem acessar fisicamente.

### 1.2.1 Mergulhadores de Profundidade

No início da exploração e produção de petróleo no Brasil era necessário o uso de mergulhadores para instalação e acionamento dos equipamentos submarinos ("árvores de natal molhadas", por exemplo), assim como manutenção de estruturas submersas. Esta classe de profissionais era o principal meio de atuação para operar e manter a produção *offshore*. A exploração em águas rasas utilizou mergulhadores de profundidade para montar e operar os equipamentos subaquáticos para extrair as jazidas dos reservatórios, porém há mais riscos neste ramo de trabalho ao explorar áreas mais profundas.

O mergulho de profundidade é perigoso porque a grande pressão exercida pela coluna de água comprime os gases nos tecidos humanos a uma mera fração de seu volume normal. Compensa-se o diferencial de pressão ao inalar o ar dos tanques pressurizados que, em consequência, aumenta a concentração dos gases no organismo. O nitrogênio é um dos gases que causa maior preocupação em mergulho de profundidade. Quando respirado em pressões mais altas, tem um destacado efeito anestésico chamado de narcose por nitrogênio. A narcose pode prejudicar a habilidade do mergulhador em trabalhar, tendo em vista que afeta a concentração, a habilidade motora e o humor, entre outros. O tempo que o mergulhador permanece na profundidade de trabalho é limitado pela concentração de nitrogênio no sangue. Quanto mais profundo, menos tempo ele possui sem sofrer narcose e na sua subida são necessárias paradas de descompressão para liberar o nitrogênio acumulado no corpo [1].

Atualmente, o mergulhadores profissionais utilizam equipamentos que diminuem os efeitos narcosivos

dos gases e aumentam seu tempo dentro da água. O *rebreather*, ilustrado na figura 1.1, é um exemplo comumente visto em mergulho profundo. Esse equipamento possui dois recipientes: um que contém oxigênio e outro que contém um "trimix", uma mistura tríplice de hélio, oxigênio e nitrogênio. O *rebreather* possui um sistema que processa o gás exalado pelo mergulhador, eliminando o dióxido de carbono e injeta uma mistura de gases na sua respiração. Assim, o equipamento administra a respiração para minimizar os efeitos narcóticos e estender o período na profundidade de trabalho.



Figura 1.1: Equipamento *Rebreather*.  
[Fonte: John Clark<sup>1</sup>.]

Esse sistema também pode ser administrado na superfície, com uma equipe de apoio injetando os gases necessários por meio de uma tubulação até o mergulhador, chamada de umbilical de suporte de vida 1.2.



Figura 1.2: Mergulhadores com umbilical de suporte de vida.  
[Fonte: Jayme Pastoric<sup>2</sup>.]

<sup>1</sup> Disponível em: <http://johnclarkeonline.com/2014/11/01/redundancy-a-life-saver-in-diving-and-aviation/>; Acesso em fev. 2015.

<sup>2</sup> Disponível em: [http://www.navy.mil/view\\_image.asp?id=105412](http://www.navy.mil/view_image.asp?id=105412); Acesso em fev. 2015.

Um outro equipamento utilizado é o *Exosuit*, ilustrado na fig. 1.3. Ele consiste em um macacão metálico com juntas flexíveis que mantém o ocupante humano à pressão atmosférica durante todo o mergulho. Como o ocupante não sofre influência direta da pressão e não há acúmulo de gases no seu organismo, com o Exosuit é possível retornar a superfície rapidamente, sem parada de decompressão. O equipamento pesa cerca de 227 kg, mas dentro da água o peso aparente é reduzido para poucos quilos. Seu limite de operação é até 300 metros abaixo do nível do mar [2].



Figura 1.3: Exosuit.  
[Fonte: Nuytco Research<sup>3</sup>.]

No entanto, a busca constante de novos reservatórios em águas cada vez mais distantes da costa e com maior profundidade ressaltou a necessidade de investir em novos meios de atuação no ambiente subaquático, que superam a limitação de operação dos mergulhadores.

### 1.2.2 ROVs e AUVs

Com a expansão da barreira de exploração nas atividades petrolíferas *offshore*, as empresas estão investindo em inovações tecnológicas para superar as dificuldades técnicas presentes, principalmente quanto às condições físicas e ambientais do meio a ser explorado [3]. Elas são:

1. As condições prevalentes no clima, no ambiente marinho e nas rochas abaixo do leito oceânico;
2. As grandes distâncias entre as plataformas e os poços no fundo do oceano, e entre as plataformas e a costa;
3. A invisibilidade das operações no mar;

A principal dificuldade citada que limita a continuidade do uso dos mergulhadores é a grande distância entre a plataforma e o ponto de extração no leito marinho. Atualmente, as atividades e operações como

<sup>3</sup> Disponível em: <http://nuytco.com/products/exosuit/>. Acesso em fev. 2015.

completação de poços, o monitoramento e reparos nos equipamentos e outras operações em águas profundas e ultra-profundas são somente possíveis por veículos subaquáticos não tripulados, ou mais conhecidos como UUVs (*Unmanned Underwater Vehicles*).

Nas últimas quatro décadas houve um aumento expressivo do uso de UUVs na indústria de óleo e gás. De acordo com o Plano Mestre de UUVs da Marinha Americana [4, 5], um UUV é definido como um veículo submersível com propulsão própria que tenha operação autônoma (seja programação pré-definida, seja em tempo real) ou que seja teleoperado. Nesse caso é necessário uma conexão física via cabo umbilical para transmitir dados à superfície.

A maioria dos UUVs são utilizados para inspeção submarina de estruturas e cabeamento de ligação aos dutos de bombeamento, ANMs - árvores de natal molhadas e estruturas submarinas de apoio. O uso de UUVs é dominante em operações em águas profundas e ultra-profundas, onde a utilização de mergulhadores é inviável em razão da grande pressão.

Tradicionalmente, pelo seu uso na indústria de óleo e gás, a categoria de UUVs pode ser desmembrada por dois principais grupos de robôs [6]: AUVs (*Autonomous Underwater Vehicles*) e ROVs (*Remote Operated Vehicles*), que diferem pelo seu modelo, projeto mecânico e operações capazes de executarem.

ROVs são veículos subaquáticos operados remotamente a partir da superfície, via transmissão de um sinal de vídeo captado por uma ou mais câmeras presentes no veículo. Como a comunicação via rádio não é possível de ser realizada à profundidade operacional dos veículos subaquáticos é necessário ter um caminho físico que possibilita a transmissão de dados e vídeo para controle do veículo. Todo ROV possui um componente essencial, que é um cabo umbilical que transmite sinais de vídeos e energia do veículo até a superfície.

Há três principais classes de ROVs presentes no mercado [6]:

- **Classe Observação:** Também conhecidos como OCROVs (*Observational Classe ROV*), são projetados para serem um sistema de captura de dados (seja vídeo ou coleta de amostras de água) em regiões subaquáticas a serem exploradas. Esses veículos são normalmente menores em tamanho, com propulsores leves para deslocamento, apenas podendo carregar ou não sensores para medições do ambiente ou do próprio veículo. A figura 1.4 ilustra um exemplo de um OCROV.



Figura 1.4: O OCROV VideoRAY Pro IV.

[Fonte: VideoRay<sup>4</sup>.]

<sup>4</sup> Disponível em: <http://www.videoray.com/>; Acesso em fev. 2015.

- **Classe Trabalhador:** São sistemas grandes, com múltiplos manipuladores para atividades pesadas de trabalho subaquático, como construção de estruturas marinhas, linhas de transmissão de óleo e gás e manutenções de equipamentos submersos. Um exemplo de ROV desta classe está ilustrado nas figura 1.5.



Figura 1.5: O ROV Classe Trabalhador Magellan 725 da Oceaneering.  
[Fonte: Oceaneering<sup>5</sup>.]

- **Classe Especial:** São ROVs com geometrias alternativas ao bloco comum, especialmente projetados para executar uma tarefa específica no terreno. Um exemplo é o ROV especial com objetivo de cavar o leito marinho para instalação de turbinas eólicas, como ilustrado na figura 1.6.



Figura 1.6: O ROV Classe Especial MV Nico de escavação submarina para instalação de turbinas eólicas (*offshore*).

[Fonte: Site offshore.no<sup>6</sup>.]

<sup>5</sup> Disponível em: <http://www.oceaneering.com/6211/lost-riser-located/>;. Acesso em fev. 2015.

<sup>6</sup> Disponível em: <http://www.offshore.no/international/news/article.aspx?id=15453>;. Acesso em fev. 2015.

AUVs são veículos autônomos que possuem vantagens que superam as dificuldades encontradas na conexão física do cabo umbilical presente nos ROVs. Tais veículos carregam sua própria fonte de energia (ex: baterias, células de hidrogênio ou sistemas de reaproveitamento de energia solar) que faz com que eles não precisem carregar um cabo que compromete sua mobilidade. Porém, as reservas de energia de um veículo autônomo necessitam ser monitoradas constantemente para permitir que este possa cumprir sua missão planejada e retornar à sua origem de partida.

Atualmente, a comunicação entre AUVs é limitada [6]. Como o veículo não possui conexão física, há necessidade de utilização de algum meio para comunicação, sendo os dois principais a comunicação acústica (via água) e a comunicação de rádio (quando o veículo está na superfície). Ambas não possuem alta capacidade de transmissão de dados e sofrem na qualidade de transmissão por perturbações e más condições do ambiente. Assim, só é possível transmitir dados por estes meios sobre localização e estado do veículo, normalmente em baixa frequência, para conseguir atingir um raio maior de alcance. A comunicação limitada faz com que a autonomia do sistema seja robusta, para lidar com as incertezas sem intervenção humana.

AUVs são comumente utilizados em levantamentos de dados em expedições científicas, coleta de amostras para oceanografia (como ilustrado na figura 1.7), arqueologia submarina e inspeções em regiões abaixo do gelo. Sua utilização em aplicações militares atualmente envolve o monitoramento de embarcações pela região, busca de minas ou o uso como batedor para verificar se as águas à frente estão seguras para que a embarcação militar cruze. Porém, os AUVs ainda não conseguem substituir o trabalho de coleta e manipulação dos ROVs, pois estes trabalhos exigem atuação complexa e necessitam de um operador experiente para lidar com as incertezas do ambiente e da situação, fato que ainda não foi configurado para sistemas autônomos lidarem.



Figura 1.7: O AUV SeaBED em uma expedição da Woods Hole Oceanographic Institute.

[Fonte: Site AUVAC<sup>7</sup>.]

Ultimamente, há um novo desmembramento de classificação desses equipamentos juntando as vantagens de AUVs com ROVs em uma categoria chamada HROVs (*Hybrid Remote Operated Vehicle*) a qual utiliza a versatilidade de trabalho do ROV com algumas automações que facilitam o trabalho do operador e de sua equipe de apoio. Um exemplo é o HROV Nereus, ilustrado na figura 1.8. Ele possui dois modos

<sup>7</sup> Disponível em: <http://auvac.org/configurations/view/74;>. Acesso em fev. 2015.

complementares: um modo AUV, que tem o objetivo de analisar grandes áreas na profundidade programada e retornar dados à superfície. Quando há um objeto de interesse, a equipe configura o veículo no modo ROV para atuação e manipulação no ambiente subaquático, conectado somente por um cabo umbilical de fibra ótica para transmissão de dados.

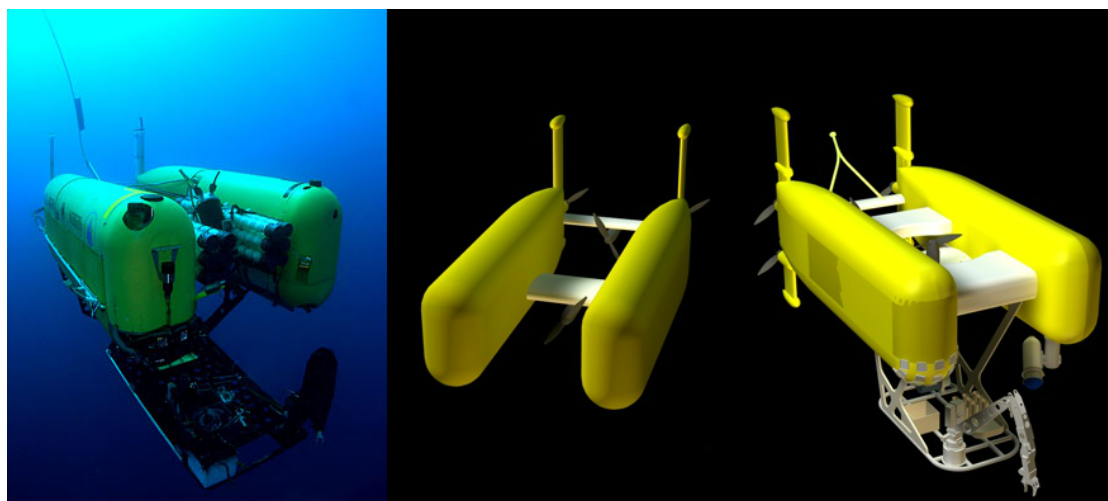


Figura 1.8: O HROV Nereus da Woods Hole Oceanographic Institute. No meio a configuração AUV e à direita a configuração ROV.

[Fonte: Woods Hole Oceanographic Institute <sup>8</sup>.]

### 1.2.3 Dificuldades na atuação de UUVs

Mesmo vencendo os limites de profundidade que impedem a atuação de mergulhadores, os UUVs necessitam superar as outras duas dificuldades técnicas apresentadas na exploração *offshore*: As condições prevalentes no clima e a invisibilidade das operações no mar.

O clima é um problema presente não só nos veículos subaquáticos, mas também em veículos e estruturas na superfície. A complexa dinâmica do ambiente marinho, caracterizada por ondas, correntezas e interação com seres marinhos, complica o processo de posicionamento no espaço. Processos de controle dinâmico e localização global são extremamente necessários para que todos os veículos presentes consigam interagir com as estruturas e atingir seu objetivo, seja para completar uma trajetória predefinida, seja para executar uma operação.

A invisibilidade das operações constitui-se na terceira particularidade das atividades *offshore*. A falta de luz solar em áreas profundas e a turbidez da água, em razão da presença de partículas em suspensão, complicam a navegação e a operação tanto de mergulhadores quanto de veículos subaquáticos, como pode ser observado na figura 1.9. Os UUVs, quando atuam nessas condições, necessitam utilizar iluminação artificial por meio de lâmpadas potentes que resistem às condições marinhas. Mesmo assim, os algoritmos de visão computacional, utilizados pelos UUVs para analisar as imagens nas operações de busca e mapeamento, necessitam de imagens bem iluminadas para funcionarem adequadamente. Há avanços na área de

<sup>8</sup> Disponível em: <http://www.whoi.edu/main/nereus>; Acesso em fev. 2015.



visão computacional em ambientes pouco iluminados [7], porém ainda não estão em níveis confiáveis para atuação autônoma.



Figura 1.9: Diferentes condições de visibilidade no ambiente subaquático. À esquerda exemplo de visibilidade boa em uma piscina de treino de mergulho. À direita exemplo de visibilidade baixa encontrada no Lago Paranoá, Brasília.

[Fonte: Elaborada pelo autor.]

Tanto AUVs quanto ROVs dependem das leituras dos seus sensores para funcionarem como projetados. Quanto mais programação e processamentos dos múltiplos sensores, melhor fica para que o sistema (ou operador no caso de ROVs) consiga navegar bem no ambiente subaquático. Porém, cada sensor capta ruído, processando de forma limitada a informação. Isso em conjunto com a elevada dinâmica não-linear e invisibilidade no mar faz com que não seja possível garantir a navegação ou operação confiável dos veículos subaquáticos baseando-se somente na leitura dos sensores.

Com isso, o foco desta dissertação é apresentar um procedimento de navegação autônoma de veículos subaquáticos que lida com as condições não lineares e incertas do ambiente e do veículo utilizando métodos probabilísticos de localização. Esse procedimento é descrito pelas etapas de Modelagem, Controle, Localização, Planejamento e Ensaios, como ilustrado na figura 1.10.

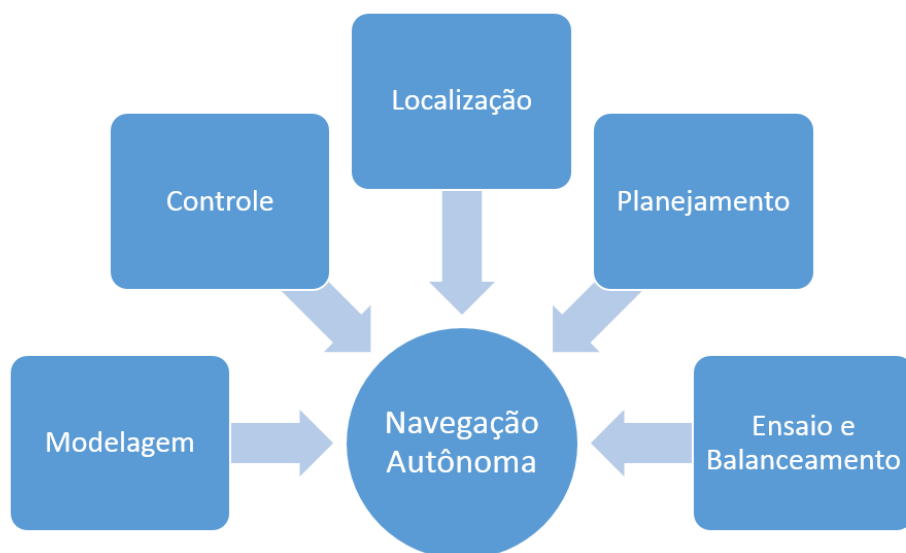


Figura 1.10: Etapas para obter navegação autônoma em UUVs  
 [Fonte: Elaborada pelo autor.]

### 1.3 OBJETIVOS

O objetivo geral deste trabalho é desenvolver um procedimento para obter a navegação autônoma em veículos subaquáticos, principalmente em ROVs. Ao mesmo tempo em que projetistas de AUVs também podem usufruir deste trabalho, ele possui o foco de oferecer esta funcionalidade aos operadores de ROV como auxílio em guiar o veículo até o local de trabalho. Visto que a tendência na indústria de UUVs é cada vez mais implementar funções de apoio aos operadores, a fim de que eles fiquem focados no objetivo e não na navegação, esta metodologia servirá como passo inicial na adaptação de um sistema teleoperado para um sistema automatizado.

Os objetivos específicos definidos neste trabalho são:

- Esquematizar a metodologia modular de navegação autônoma;
- Construir um veículo de teste;
- Determinar os parâmetros para construir o modelo matemático do veículo;
- Utilizar algoritmos de planejamento de trajetória visando encontrar a rota otimizada e adaptativa ao ambiente;
- Implementar filtros probabilísticos para obter localização 2D no meio subaquático.

## 1.4 ESTRUTURA DO DOCUMENTO

Esta dissertação foi dividida em 7 capítulos.

No capítulo 2 é feita a revisão bibliográfica sobre a metodologia atual de modelagem de veículos subaquáticos, apresentando as equações que regem a dinâmica do veículo no meio fluido e os parâmetros necessários para montagem de um modelo que prevê o seu comportamento.

Em seguida, o capítulo 3 apresenta métodos para lidar com as incertezas, via robótica probabilística. Também é apresentada a implementação dos filtros histograma e filtro de partículas que são utilizados para estimar a localização no meio, baseados em processamento de movimento e percepção.

No capítulo 4 é apresentado o problema de planejamento de trajetórias em robótica móvel, expondo quais parâmetros são necessários informar para que o robô possa determinar a melhor rota de navegação em um determinado ambiente.

O procedimento para obter a navegação é apresentado no capítulo 5. Nesse capítulo será exposta a metodologia proposta para obter navegação autônoma utilizando um ROV denominado "UnBot" como veículo teste, ilustrado na figura 1.11. O modelo físico dele é determinado por sistemas CAD e de simulação em CFD. Os algoritmos de localização e planejamento de trajetória são implementados na plataforma embarcada e são feitos ensaios com o veículo no intuito de demonstrar a funcionalidade dessa metodologia em ambiente real.

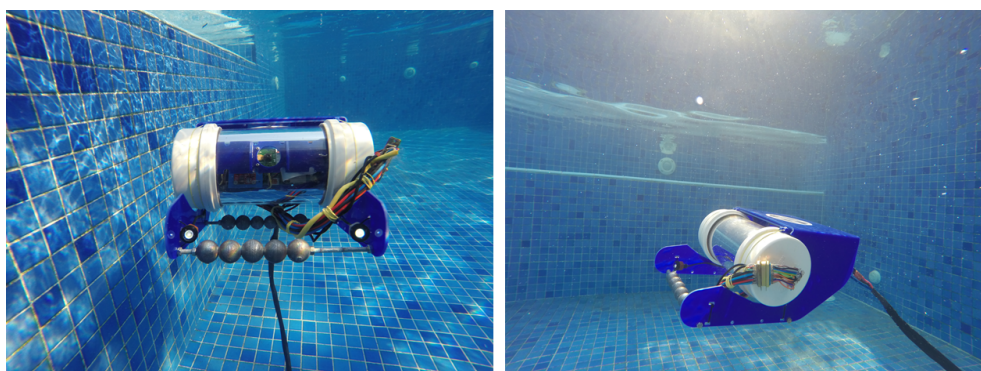


Figura 1.11: Veículo exemplo do trabalho, OCROV "UnBot"

[Fonte: Elaborada pelo autor.]

Os ensaios conduzidos com o veículo teste em ambiente subaquático são descritos no capítulo 6. A metodologia proposta no capítulo 5 é aplicada em um estudo de caso para localização auxiliada em um ambiente bidimensional utilizando filtro de partículas.

Finalmente, são apresentadas as conclusões e recomendações de trabalhos futuros no capítulo 7.

## 2 MODELAGEM

### 2.1 INTRODUÇÃO

Neste capítulo, apresentam-se as equações necessárias para obter modelo matemático de veículos subaquáticos e dos controladores. Apresenta-se as notações de posicionamento e orientação utilizadas, assim como os parâmetros físicos necessários a determinar para caracterizar o veículo nas matrizes do sistema de equações da dinâmica do veículo.

### 2.2 SISTEMA DE COORDENADAS

Na mecânica newtoniana, a posição de um corpo rígido é conhecida recorrendo a pontos de referência. Existem muitos exemplos de posição relativa dos quais se podem destacar os seguintes: a medição da longitude no globo terrestre possui como referência o meridiano de Greenwich (longitude =  $0^\circ$ ) definido por convenção; da mesma forma, a linha do equador serve de referência para a medição da latitude. Estes referenciais estão fixos à Terra. Assumindo que as suas acelerações de translação e de rotação são nulas, pode-se considerar os referenciais fixos à Terra como inerciais. Um referencial inercial é um referencial que não está sujeito a acelerações lineares nem angulares.

Para veículos marítimos, adota-se dois referenciais:

- NED (referencial n): O sistema de coordenada *North-East-Down* é definido com base no elipsoide referencial da Terra. É comumente definido como um plano tangencial da superfície da Terra que se move juntamente com o veículo, mas com eixos em direções diferentes aos do sistema do veículo. Para o sistema NED, o eixo x aponta para o verdadeiro Norte, o eixo y para o Leste e o eixo z para dentro da Terra. Para veículos marítimos operando em área local (com latitude e longitude aproximadamente constantes), se utiliza o sistema NED como ponto de referência inercial de localização junto a Terra.
- Corpo (referencial b): Esse é um sistema de coordenadas móvel que é fixo ao veículo. A posição e orientação do veículo é descrito relativo à referência inercial (referencial n), enquanto a velocidade linear e angular do veículo é expresso ao referencial b. A origem O do sistema é normalmente coincidente com o centro de gravidade (CG) quando esse está no centro do plano de simetria do veículo, ou em qualquer outro ponto quando conveniente para a modelagem. No caso de veículos marítimos, os eixos dos sistemas são orientados para ficarem paralelos aos eixos principais de inércia do veículo.

A SNAME (*Society of Naval Architects & Marine Engineers*) implementou nos anos 50 uma notação para representar as grandezas relacionadas a esses referenciais, de modo a criar um padrão das notações utilizadas na indústria náutica e simplificar as equações matemáticas. Tem-se:

$$\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \end{bmatrix}, \boldsymbol{\eta}_1 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \boldsymbol{\eta}_2 = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \quad (2.1)$$

$$\boldsymbol{v} = \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{bmatrix}, \boldsymbol{v}_1 = \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \boldsymbol{v}_2 = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.2)$$

$$\boldsymbol{\tau} = \begin{bmatrix} \boldsymbol{\tau}_1 \\ \boldsymbol{\tau}_2 \end{bmatrix}, \boldsymbol{\tau}_1 = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \boldsymbol{\tau}_2 = \begin{bmatrix} K \\ M \\ N \end{bmatrix} \quad (2.3)$$

nas quais  $\boldsymbol{\eta}_1$  e  $\boldsymbol{\eta}_2$  são vetores posição e orientação, respectivamente, do referencial  $b$  relativo ao referencial inercial;  $\boldsymbol{v}_1$  e  $\boldsymbol{v}_2$  são vetores velocidades linear e angular no referencial fixo ao veículo;  $\boldsymbol{\tau}_1$  e  $\boldsymbol{\tau}_2$  são vetores força e momento sobre o veículo no referencial fixo que lhe está associado.

## 2.3 CINEMÁTICA

Dada a orientação do referencial fixo ao corpo relativo ao referencial fixo à Terra é possível relacionar os vetores de ambos. A orientação do referencial  $\{x, y, z\}$  pode ser obtida através de três rotações elementares segundo cada eixo do referencial  $b$  fixo ao corpo  $\{x_B, y_B, z_B\}$ , isto é, uma rotação segundo  $x_B$  de um ângulo  $\phi$ , seguido de uma rotação de um ângulo  $\theta$  segundo  $y_B$  e de uma rotação de um ângulo  $\psi$  segundo  $z_B$ . Obtêm-se as respectivas matrizes de rotação, segundo cada eixo:

$$\boldsymbol{R}_{x,\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}, \boldsymbol{R}_{y,\theta} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, \boldsymbol{R}_{z,\psi} = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

Combinando-as, obtêm-se o jacobiano  $\boldsymbol{J}_1(\boldsymbol{\eta}_2)$  do referencial  $b$  fixo ao referencial NED:

$$\boldsymbol{J}_1(\boldsymbol{\eta}_2) = \begin{bmatrix} c(\psi)c(\theta) & -s(\psi)c(\phi) + c(\psi)s(\theta)s(\phi) & s(\psi)s(\phi) + c(\psi)c(\phi)s(\theta) \\ s(\psi)c(\theta) & c(\psi)c(\phi) + s(\phi)s(\theta)s(\psi) & -c(\psi)s(\phi) + s(\theta)s(\psi)c(\theta) \\ -s(\theta) & c(\theta)s(\phi) & c(\theta)c(\phi) \end{bmatrix} \quad (2.5)$$

com ela é possível relacionar a velocidade linear  $\boldsymbol{v}_1$  no referencial do veículo  $\{x_B, y_B, z_B\}$  com a velocidade linear no referencial fixo na Terra, resultando na seguinte expressão:

$$\dot{\boldsymbol{\eta}}_1 = \boldsymbol{J}_1(\boldsymbol{\eta}_2)\boldsymbol{v}_1 \quad (2.6)$$

A velocidade angular do veículo  $\boldsymbol{v}_2 = [p, q, r]^T$  e a velocidade no referencial NED  $\boldsymbol{\eta}_2$  são relacionadas pelo jacobiano  $\boldsymbol{J}_2(\boldsymbol{\eta}_2)$  pela expressão:

$$\dot{\eta}_2 = J_2(\eta_2)v_1 \quad (2.7)$$

Para determinar o  $J_2(\eta_2)$ , calcula-se seu inverso antes. Pelo método vetorial entre o referencial fixo do veículo com o referencial NED, encontra-se:

$$v_2 = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_{x,\phi}^T \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_{x,\phi}^T R_{y,\theta}^T \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} = J_2(\eta_2)^{-1}\dot{\eta}_2 \quad (2.8)$$

Resolvendo essa expressão, encontra-se:

$$J_2(\eta_2)^{-1} = \begin{bmatrix} 1 & 0 & -s(\theta) \\ 0 & c(\phi) & c(\theta)s(\phi) \\ 0 & -s(\phi) & c(\theta)c(\phi) \end{bmatrix} \Rightarrow J_2(\eta_2) = \begin{bmatrix} 1 & s(\phi)t(\theta) & c(\phi)t(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & \frac{s(\phi)}{c(\theta)} & \frac{c(\phi)}{c(\theta)} \end{bmatrix} \quad (2.9)$$

sendo  $s = \sin(\cdot)$ ,  $c = \cos(\cdot)$  e  $t = \tan(\cdot)$ . De forma mais compacta, a relação entre as velocidades lineares e angulares nos referenciais NED e fixo no veículo é:

$$\dot{\eta} = J(\eta_2)v \quad (2.10)$$

sendo,

$$J(\eta_2) = \begin{bmatrix} J_1(\eta_2) & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & J_2(\eta_2) \end{bmatrix} \quad (2.11)$$

A transformada é indefinida quando  $\theta = \pm 90^\circ$ . Para superar esta singularidade é necessário utilizar a notação via "quaternions", porém no caso do ROV ensaiado este problema não existe, pois o veículo não foi projetado nem requerido a trabalhar com  $\theta = \pm 90^\circ$ . O jacobiano é necessário para obter a transformação da coordenada local para o sistema inercial, servindo como parâmetro para controle de posição durante a navegação.

## 2.4 DINÂMICA DE CORPO RÍGIDO

Nesta seção, é importante definir previamente alguns conceitos. A segunda lei de Newton relaciona a aceleração de um corpo rígido com as forças que nele atuam. Isso se deve ao fato de um corpo em repouso se movimentar apenas se uma força lhe for aplicada. No caso contrário, um corpo em movimento atingirá o estado de repouso apenas se uma força no sentido contrário ao seu movimento lhe for aplicada. Interessa ainda referir que um corpo rígido é um corpo indeformável constituído por um número finito de partículas.

### 2.4.1 Momento Linear

Considere-se um corpo com densidade de massa  $\rho_B(r_{CG})$ , volume  $V$  e superfície  $S$ . Um ponto do corpo encontra-se na posição  $r = r_B + r_{CG}$  em que  $r_B$  é o vetor de posição da origem do referencial fixo ao corpo em relação ao referencial inercial absoluto e  $r_{CG}$  um vetor de posição no referencial do corpo.

A expressão da segunda lei de Newton para a conservação do momento linear é dada por:

$$\int_V \frac{d^2 r}{dt^2} \rho_B(r_{CG}) dV = \int_V \rho_B(r_{CG}) g dV + \int_S f(r_{CG}) dS \quad (2.12)$$

em que  $g$  é o vetor de aceleração da gravidade,  $f(r_{CG})$  é a força aplicada no ponto  $r_{CG}$ ,  $dV$  e  $dS$  são uma porção infinitesimal de volume e de superfície do corpo, respectivamente. A conservação do momento linear pode ser expressa qualitativamente como a soma das forças de campo com as forças de contato.

Considerando que o corpo possui massa constante ( $m = \int_V \rho_B(\vec{r}) dV, m \in \mathbb{R}$ ) é possível deduzir a expressão

$$M_{massa}(\dot{v}_B + \omega_B \times v_B + \dot{\omega}_B \times r_{CG} + \omega_B \times (\omega_B \times r_{CG})) = \sum f_{ext} \quad (2.13)$$

em que:

- $v_B \equiv \eta_1$  é a velocidade da origem do referencial fixo ao corpo em relação ao referencial inercial.
- $\dot{v}_B \equiv \dot{v}_1$  é a derivada de primeira ordem da velocidade no referencial fixo ao corpo.
- $\omega_B \equiv v_2$  é a velocidade angular segundo cada eixo do referencial fixo ao corpo;
- $r_{CG} = (x_{CG}, y_{CG}, z_{CG})^T$  é a posição do centro de gravidade do corpo;
- $\sum f_{ext} = (\sum X_{ext}, \sum Y_{ext}, \sum Z_{ext})^T$  é a soma das forças externas que atuam no corpo;
- $M_{massa} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & m \end{bmatrix} = mI_{3 \times 3}$  é a matriz de massa do veículo subaquático.

Decompondo as forças segundo os eixos cartesianos do referencial do corpo, obtêm-se:

$$\begin{aligned} \sum X_{ext} &= m(\dot{u} - vr + wq - x_{CG}(q^2 + r^2) + y_{CG}(pq - \dot{r}) + z_{CG}(pr + \dot{q})); \\ \sum Y_{ext} &= m(\dot{v} - wp + ur - y_{CG}(r^2 + p^2) + z_{CG}(qr - \dot{p}) + x_{CG}(qp + \dot{r})); \\ \sum Z_{ext} &= m(\dot{w} - uq + vp - z_{CG}(p^2 + q^2) + x_{CG}(rp - \dot{q}) + y_{CG}(rq + \dot{p})); \end{aligned} \quad (2.14)$$

## 2.4.2 Momento angular

Relativamente à conservação do momento angular, a segunda lei de Newton resulta na seguinte expressão:

$$\int_V \frac{d}{dt} \left( r \times \frac{dr}{dt} \right) \rho_B(r_{CG}) dV = \int_V r \times \rho_B(r_{CG}) g dV + \int_S r \times f(r_{CG}) dS \quad (2.15)$$

O desenvolvimento dessa expressão fornece o seguinte:

$$I_B \dot{\omega}_B + \omega_B \times (I_B \omega_B) + M_{massa} r_{CG} (\dot{v}_B + \omega_B \times v_B) = \sum m_{ext} \quad (2.16)$$

Em que  $\sum \mathbf{m}_{ext} = (\sum K_{ext}, \sum N_{ext}, \sum M_{ext})^T$  é a soma dos momentos exercidos no corpo e  $\mathbf{I}_B \in \mathbb{R}^{3 \times 3}$  o tensor de inércia dado por:

$$\mathbf{I}_B = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix}, I_B = I_B^T > 0 \quad (2.17)$$

sendo  $I_{xx}, I_{yy}, I_{zz}$  são os momentos de inércia sobre os eixos  $x_B, y_B$  e  $z_B$ , e  $I_{xy} = I_{yx}, I_{xz} = I_{zx}$  e  $I_{yz} = I_{zy}$  são os produtos de inércia definidos como:

$$\begin{aligned} I_{xx} &= \int_V (y^2 + z^2) \rho_B dV & I_{xy} &= \int_V xy \rho_B dV = \int_V yx \rho_B dV = I_{yx} \\ I_{yy} &= \int_V (x^2 + z^2) \rho_B dV & I_{xz} &= \int_V xz \rho_B dV = \int_V zx \rho_B dV = I_{zx} \\ I_{zz} &= \int_V (x^2 + y^2) \rho_B dV & I_{yz} &= \int_V yz \rho_B dV = \int_V zy \rho_B dV = I_{zy} \end{aligned} \quad (2.18)$$

Decompondo-se os momentos exercidos sobre o corpo segundo cada eixo de rotação no referencial do veículo, obtêm-se:

$$\begin{aligned} \sum K_{ext} &= I_{xx} \dot{p} + (I_{zz} - I_{yy})qr - (\dot{r} + pq)I_{xz} + (r^2 - q^2)I_{yz} + (pr - \dot{q})I_{xy} \\ &\quad + m[y_{CG}(\dot{w} - uq + vp) - z_{CG}(\dot{v} - wp + ur)]; \\ \sum M_{ext} &= I_{yy} \dot{q} + (I_{xx} - I_{zz})rp - (\dot{p} + qr)I_{xy} + (p^2 - r^2)I_{zx} + (qp - \dot{r})I_{yz} \\ &\quad + m[z_{CG}(\dot{u} - vr + wq) - x_{CG}(\dot{w} - uq + vp)]; \\ \sum N_{ext} &= I_{zz} \dot{r} + (I_{yy} - I_{xx})pq - (\dot{q} + rp)I_{yz} + (q^2 - p^2)I_{xy} + (rq - \dot{p})I_{zx} \\ &\quad + m[x_{CG}(\dot{v} - wp + ur) - y_{CG}(\dot{u} - vr + wq)]; \end{aligned} \quad (2.19)$$

### 2.4.3 Representação Matricial

É possível reescrever as expressões dadas por (2.14) e (2.19) sob forma matricial. Para tal, considere-se  $\tau_{ext}$  o vetor soma de todas as forças e momentos exercidos sobre o corpo, tal que:

$$\tau_{ext} = \begin{bmatrix} \sum f_{ext} \\ \sum m_{ext} \end{bmatrix} = \begin{bmatrix} \sum X_{ext} \\ \sum Y_{ext} \\ \sum Z_{ext} \\ \sum K_{ext} \\ \sum M_{ext} \\ \sum N_{ext} \end{bmatrix} \quad (2.20)$$

A equação de corpo rígido, que consiste nas forças inerciais e nas forças centrípetas e de Coriolis, pode ser expressa na forma matricial:

$$\tau_{ext} = \mathbf{M}_{RB} \dot{\mathbf{v}} + \mathbf{C}_{RB}(\mathbf{v})\mathbf{v} \quad (2.21)$$

em que  $\mathbf{M}_{RB} \in \mathbb{R}^{6 \times 6}$  é uma matriz constante conhecida como a matriz de inércia do corpo rígido, a matriz  $\mathbf{C}_{RB} \in \mathbb{R}^{6 \times 6}$  é uma matriz variável conhecida como matriz dos termos centrípetos e de Coriolis  $\mathbf{v} = [\mathbf{v}_1, \mathbf{v}_2]^T \in \mathbb{R}^6$  é o vetor velocidade linear e angular no referencial do veículo.



A matriz de inércia do corpo rígido  $M_{RB}$  pode ser determinada a partir das equações (2.13) e (2.16) como:

$$M_{RB}\dot{v} = \begin{bmatrix} M_{massa}\dot{v}_1 + m\dot{v}_2 \times r_{CG} \\ I\dot{v}_2 + mr_{CG} \times \dot{v}_1 \end{bmatrix} \quad (2.22)$$

Utilizando a equação anterior, a matriz positiva  $M_{RB} = M_{RB}^T > 0$  pode ser separada e definida como:

$$M_{RB} = \begin{bmatrix} M_{massa} & -mS(r_{CG}) \\ mS(r_{CG}) & I \end{bmatrix} \quad (2.23)$$

com

$$S(r_{CG}) = \begin{bmatrix} 0 & -z_{CG} & y_{CG} \\ z_{CG} & 0 & -x_{CG} \\ -y_{CG} & x_{CG} & 0 \end{bmatrix} \quad (2.24)$$

sendo  $S(r_{CG})$ , é a matriz antissimétrica do vetor  $r_{CG}$  (lembrando que  $\lambda \times a = S(\lambda)a$ ). Expandindo-se os termos individuais da matriz, tem-se:

$$M_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & mz_{CG} & -my_{CG} \\ 0 & m & 0 & -mz_{CG} & 0 & mx_{CG} \\ 0 & 0 & m & my_{CG} & -mx_{CG} & 0 \\ 0 & -mz_{CG} & my_{CG} & I_{xx} & -I_{xy} & -I_{xz} \\ mz_{CG} & 0 & -mx_{CG} & -I_{yx} & I_{yy} & -I_{yz} \\ -my_{CG} & mx_{CG} & 0 & -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix} \quad (2.25)$$

Similarmente, os termos centrípetos e de Coriolis, que descrevem o movimento angular do veículo em (2.13) e (2.16), podem ser reescritos como:

$$C_{RB}(v)v = \begin{bmatrix} m[v_2 \times v_1 + v_2 \times (v_2 \times r_{CG})] \\ v_2 \times (Iv_2) + mr_{CG} \times (v_2 \times v_1) \end{bmatrix} \quad (2.26)$$

Utilizando-se os resultados  $a \times b \times c = S(a)S(b)c$  e  $S(v_1)v_1 = 0$ , a equação (2.26) pode ser manipulada para resultar:

$$C_{RB}(v)v = \begin{bmatrix} -mS(v_1)v_2 - mS(v_2)S(r_{CG})v_2 \\ -mS(v_1)v_1 + mS(r_{CG})S(v_2)v_1 - S(Iv_2)v_2 \end{bmatrix} \quad (2.27)$$

Separando-se  $v_1$  e  $v_2$ , tem-se:

$$C_{RB}(v) = \begin{bmatrix} \mathbf{0}_{3 \times 3} & -mS(v_1) - mS(v_2)S(r_{CG}) \\ -mS(v_1) + mS(r_{CG})S(v_2) & -S(Iv_2) \end{bmatrix} \quad (2.28)$$

Expandindo-se os termos de (2.28), tem-se:

$$C_{RB} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & C_{12}(v) \\ -C_{12}(v)^T & C_{22}(v) \end{bmatrix} \quad (2.29)$$

com as matrizes  $C_{12}$  e  $C_{22}$  sendo:

$$C_{12}(v) = \begin{bmatrix} m(y_{CG}q + z_{CG}r) & -m(x_{CG}q - w) & -m(x_{CG}r + v) \\ -m(y_{CG}p + w) & m(z_{CG}r + x_{CG}p) & -m(y_{CG}r - u) \\ -m(z_{CG}p - v) & -m(z_{CG}q + u) & m(x_{CG}p + y_{CG}q) \end{bmatrix} \quad (2.30)$$

$$C_{22}(v) = \begin{bmatrix} 0 & -I_{yz}q - I_{xz}p + I_{zz}r & I_{yz}r + I_{xy}p - I_{yy}q \\ I_{yz}q + I_{xz}p - I_{zz}r & 0 & -I_{xz}r - I_{xy}q + I_{xx}p \\ -I_{yz}r - I_{xy}p + I_{yy}q & I_{xz}r + I_{xy}q - I_{xx}p & 0 \end{bmatrix} \quad (2.31)$$

## 2.5 HIDRODINÂMICA

Nesta seção, apresentam-se as forças geradas pelo movimento de um corpo num fluido, determinando-se as expressões que lhes estão associadas. Algumas considerações serão realizadas, de modo a simplificar as expressões.

### 2.5.1 Massa adicionada

Um corpo em movimento, inserido num fluido estacionário, gera um movimento das partículas que o envolvem. A aceleração destas partículas induz forças e momentos contrários ao movimento do corpo. Estes são conhecidos como forças e momentos de massa adicionada. Esse movimento do corpo induz uma variação na energia cinética  $T$  associada ao fluido, que pode ser representada como [8]:

$$T = \frac{\rho}{2} \int_V (u_1^2 + u_2^2 + u_3^2) dV = \frac{\rho}{2} \int_V u_i u_i dV \quad (2.32)$$

sendo  $u_i (i = 1, 2, 3)$  representa os componentes cartesianos da velocidade do fluido e  $V$  é o volume do fluido. Para a análise nesta dissertação, assume-se que o fluido é incompressível com densidade igual a  $\rho$ .

Se o corpo estiver com um movimento linear constante com velocidade translacional  $U$  em relação ao fluido em repouso, a quantidade de energia cinética  $T$  pelo tempo será constante, sendo expressa como:

$$T = \rho \frac{I}{2} U^2 \text{ onde } I = \int_V \frac{u_i}{U} \frac{u_i}{U} dV \quad (2.33)$$

Quando há uma variação de velocidade do corpo imerso, a energia cinética também varia. Para isso, é necessário que haja um fornecimento de trabalho para o fluido, que é fornecido pelo aumento de arrasto hidrodinâmico  $F$ , na mesma taxa de variação do trabalho. Este arrasto é expresso por [8]:

$$F = -\frac{1}{U} \frac{dT}{dt} = -\rho I \frac{dU}{dt} \quad (2.34)$$

Essa força tem a mesma forma e sinal requerido para acelerar uma massa  $m$  do próprio corpo, análogo à quantidade de momento  $m \frac{dU}{dt}$ . Assim, é conveniente visualizar este valor como uma massa adicionada  $M$ , que representa uma quantidade de massa de fluido que tem que ser acelerada para mudar a energia cinética.

Tal como sucede com a dinâmica de um corpo na seção 2.3, é realizada uma separação dos termos dependentes das acelerações e das velocidades do corpo, por conveniência. As forças e momentos que atuam no corpo devido ao efeito da massa adicionada são dados por:

$$\tau_A(\dot{v}, v) = -M_A \dot{v} - C_A(v)v \quad (2.35)$$

em que  $\tau_A$  é o vetor de forças e momentos de massa adicionada totais. Denomina-se matriz  $M_A \in \mathbb{R}^{6 \times 6}$  como matriz de massa adicionada e é dada por:

$$M_A = - \begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix} \quad (2.36)$$

sendo  $M_A > 0$ .

A matriz dos termos de Coriolis e centrípetos,  $C_A(v)$ , é dada por:

$$C_A(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & -a_3 & a_2 \\ 0 & 0 & 0 & a_3 & 0 & -a_1 \\ 0 & 0 & 0 & -a_2 & a_1 & 0 \\ 0 & -a_3 & a_2 & 0 & -b_3 & b_2 \\ a_3 & 0 & -a_1 & b_3 & 0 & -b_1 \\ -a_2 & a_1 & 0 & -b_2 & b_1 & 0 \end{bmatrix} \quad (2.37)$$

sendo:

$$\begin{aligned} a_1 &= X_{\dot{u}}u + X_{\dot{v}}v + X_{\dot{w}}w + X_{\dot{p}}p + X_{\dot{q}}q + X_{\dot{r}}r \\ a_2 &= Y_{\dot{u}}u + Y_{\dot{v}}v + Y_{\dot{w}}w + Y_{\dot{p}}p + Y_{\dot{q}}q + Y_{\dot{r}}r \\ a_3 &= Z_{\dot{u}}u + Z_{\dot{v}}v + Z_{\dot{w}}w + Z_{\dot{p}}p + Z_{\dot{q}}q + Z_{\dot{r}}r \\ b_1 &= K_{\dot{u}}u + K_{\dot{v}}v + K_{\dot{w}}w + K_{\dot{p}}p + K_{\dot{q}}q + K_{\dot{r}}r \\ b_2 &= M_{\dot{u}}u + M_{\dot{v}}v + M_{\dot{w}}w + M_{\dot{p}}p + M_{\dot{q}}q + M_{\dot{r}}r \\ b_3 &= N_{\dot{u}}u + N_{\dot{v}}v + N_{\dot{w}}w + N_{\dot{p}}p + N_{\dot{q}}q + N_{\dot{r}}r \end{aligned} \quad (2.38)$$

Então, caso tenha uma aceleração  $\dot{u}$  na direção X, uma força hidrodinâmica  $X_A$  em sentido contrário ao movimento surgirá com a seguinte relação [9]:

$$X_A = X_{\dot{u}}\dot{u} \quad (2.39)$$

sendo  $X_{\dot{u}} = \frac{\delta X}{\delta \dot{u}}$ . Isso é análogo para todas as acelerações de translação e rotação para cada eixo. Para um GDL, a equação do movimento pode ser escrita como:

$$(m - X_{\dot{u}})\dot{u} = \tau_x + X_u u \quad (2.40)$$

sendo  $m$  a massa,  $X_u$  é o amortecimento linear e  $\tau_x$  é o empuxo propulsor (todas na direção x). Como pode se ver, o termo hidrodinâmico  $X_{\dot{u}}$  pode ser considerado uma "massa acrescentada" na expressão diferencial.

Em geral, o movimento de um veículo subaquático movendo em 6 GDL em alta velocidade será não-linear e acoplado. Porém, em muitas aplicações de ROV, o veículo só se locomoverá em baixa velocidade, como na aplicação de busca e navegação de falhas em estruturas submarinas.

Na fase de projeto do veículo subaquático, a escolha de como será a geometria final da carenagem externa poderá simplificar o formato da matriz de massa adicionada. Utilizando simetria dos eixos principais, pode-se eliminar algumas variáveis, tornando o cálculo mais fácil de ser resolvido [10]. A seguir, apresenta-se uma série de geometria de veículos e suas matrizes de massa adicionada, com intuito de ilustrar a quantidade de variáveis independentes necessárias a determinar, assim como a simplificação dos acoplamentos devido ao aumento de simetria do veículo em relação a cada eixo rotacional. O formato dessas matrizes serve como base para projetos de veículos subaquáticos, demonstrando visualmente como a geometria influencia o acoplamento nos eixos quando o veículo está em movimento acelerado. Os dados utilizados para calcular as matrizes de massa adicionada estão no Anexo I, seção I.2.

Por exemplo, caso o veículo tenha simetria lateral entre o bombordo e boreste, a matriz de massa adicionada será simétrica com os seguintes componentes, ilustrado na fig. 2.1:

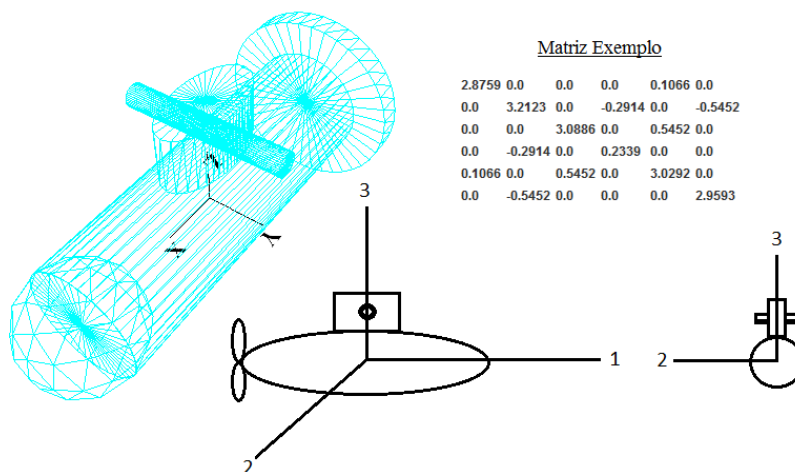


Figura 2.1: Veículo com um plano de simetria  
[Fonte: Elaborada pelo autor.]

$$M_A = - \begin{bmatrix} X_{\dot{u}} & 0 & 0 & 0 & X_{\dot{q}} & 0 \\ & Y_{\dot{v}} & 0 & Y_{\dot{p}} & 0 & Y_{\dot{r}} \\ & & Z_{\dot{w}} & 0 & Z_{\dot{q}} & 0 \\ & & & K_{\dot{p}} & 0 & 0 \\ & & & & M_{\dot{q}} & 0 \\ & & & & & N_{\dot{r}} \end{bmatrix} \quad (2.41)$$

Essa matriz é simétrica, então os elementos na parte triangular superior se repetem na parte inferior. Neste tensor de massa adicionada, pode-se verificar que com um plano de simetria, há somente necessidade de determinar 10 elementos independentes para obter a matriz.

Caso a simetria seja de dois planos, a configuração ficará como ilustrado na fig. 2.2:

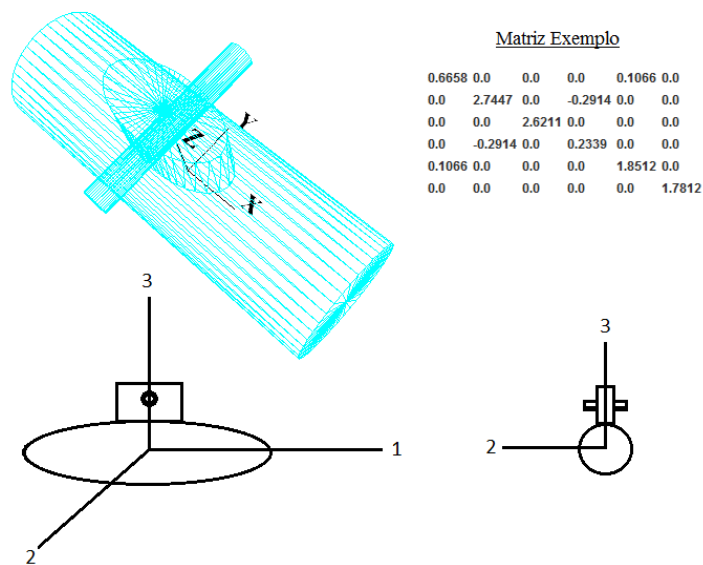


Figura 2.2: Veículo com dois planos de simetria  
[Fonte: Elaborada pelo autor.]

$$M_A = - \begin{bmatrix} X_{\dot{u}} & 0 & 0 & 0 & X_{\dot{q}} & 0 \\ & Y_{\dot{v}} & 0 & Y_{\dot{p}} & 0 & 0 \\ & & Z_{\dot{w}} & 0 & 0 & 0 \\ & & & K_{\dot{p}} & 0 & 0 \\ & & & & M_{\dot{q}} & 0 \\ & & & & & N_{\dot{r}} \end{bmatrix} \quad (2.42)$$

Nota-se que o número de coeficientes independentes reduziu-se devido ao aumento de simetria.

Quando há simetria axial (rotacional) em relação a um eixo, como ilustrado na figura 2.3, a matriz terá o seguinte formato:

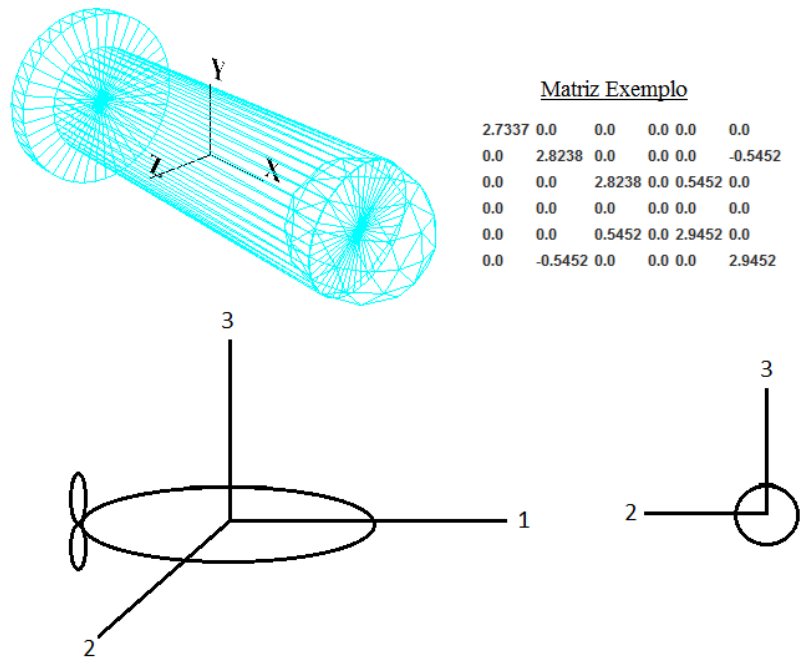


Figura 2.3: Veículo com axisimetria ou simetria rotacional em um eixo  
[Fonte: Elaborada pelo autor.]

$$M_A = - \begin{bmatrix} X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ & Y_{\dot{v}} & 0 & 0 & 0 & Y_{\dot{r}} \\ & & Z_{\dot{w}} & 0 & Z_{\dot{q}} & 0 \\ & & & 0 & 0 & 0 \\ & & & & M_{\dot{q}} & 0 \\ & & & & & N_{\dot{r}} \end{bmatrix} \quad (2.43)$$

sendo que  $Z_{\dot{q}} = -Y_{\dot{r}}$ ,  $Y_{\dot{v}} = Z_{\dot{w}}$  e  $M_{\dot{q}} = N_{\dot{r}}$ . Então, nessa configuração, a matriz só terá 4 coeficientes distintos. Um segundo caso de simetria axial, porém com dois eixos, será a matriz da figura 2.4:

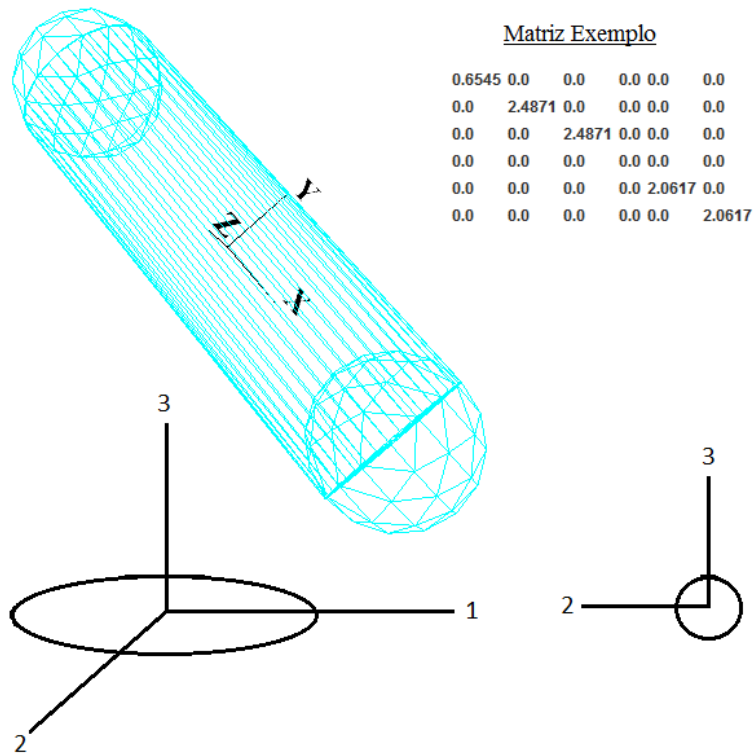
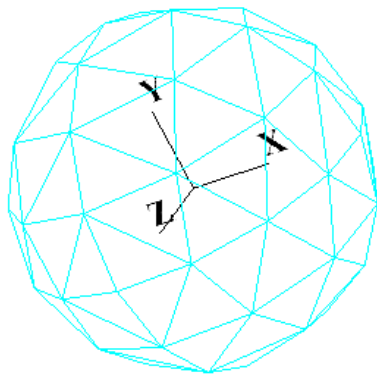


Figura 2.4: Veículo com axisimetria ou simetria rotacional em dois eixos  
 [Fonte: Elaborada pelo autor.]

$$M_A = - \begin{bmatrix} X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ & Y_{\dot{v}} & 0 & 0 & 0 & 0 \\ & & Z_{\dot{w}} & 0 & 0 & 0 \\ & & & 0 & 0 & 0 \\ & & & & M_{\dot{q}} & 0 \\ & & & & & N_{\dot{r}} \end{bmatrix} \quad (2.44)$$

Assim, só tem-se 3 termos independentes, com  $Y_{\dot{v}} = Z_{\dot{w}}$  e  $M_{\dot{q}} = N_{\dot{r}}$ . E no caso extremo de simetria axial em todos os eixos de rotação, tem-se a necessidade de apenas um termo independente, ilustrado na fig. 2.5:



Matriz Exemplo

2.0944	0.0	0.0	0.0	0.0	0.0
0.0	2.0944	0.0	0.0	0.0	0.0
0.0	0.0	2.0944	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0

Figura 2.5: Veículo com axisimetria ou simetria rotacional nos três eixos

[Fonte: Elaborada pelo autor.]

$$M_A = - \begin{bmatrix} X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ & Y_{\dot{v}} & 0 & 0 & 0 & 0 \\ & & Z_{\dot{w}} & 0 & 0 & 0 \\ & & & 0 & 0 & 0 \\ & & & & 0 & 0 \\ & & & & & 0 \end{bmatrix} \quad (2.45)$$

com  $X_{\dot{u}} = Y_{\dot{v}} = Z_{\dot{w}}$ .

Se o veículo tiver 3 planos de simetria, como ilustrado na figura 2.6, pode-se desprezar os termos avulsos da diagonal principal da matriz  $M_A$ , fazendo-se com que as matrizes  $M_A$  e  $C_A(v)$  fiquem:

Matriz Exemplo

3.1416	0.0	0.0	0.0	0.0	0.0
0.0	1.7671	0.0	0.0	0.0	0.0
0.0	0.0	12.5664	0.0	0.0	0.0
0.0	0.0	0.0	16.7552	0.0	0.0
0.0	0.0	0.0	0.0	0.8836	0.0
0.0	0.0	0.0	0.0	0.0	8.3776

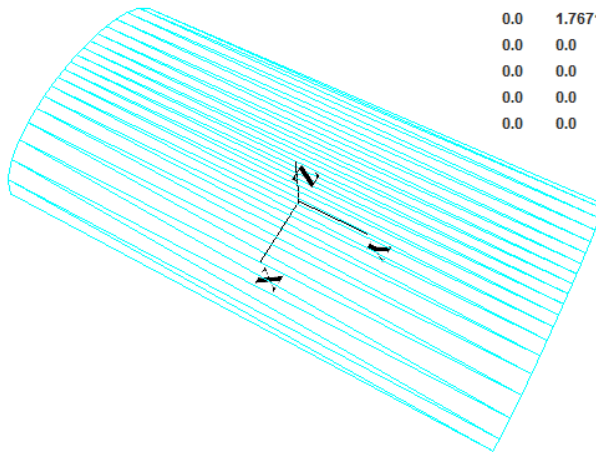


Figura 2.6: Veículo com três planos de simetria

[Fonte: Elaborada pelo autor.]

Essa estrutura é atrativa, pois elimina os termos fora da diagonal principal, os quais são difíceis de determinar analiticamente e experimentalmente. Na prática, essa aproximação é considerada muito boa



para várias aplicações na indústria de ROVs, devido ao fato de que os termos fora da diagonal são consideravelmente menores do que os termos da diagonal principal [11].

$$M_A = M_A^T = -diag\{X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}\} \quad (2.46)$$

$$C_A(v) = -C_A^T(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v \\ 0 & 0 & 0 & Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & -Y_{\dot{v}}v & X_{\dot{u}}u & 0 \\ 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ Z_{\dot{w}}w & 0 & -X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ -Y_{\dot{v}}v & X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{bmatrix} \quad (2.47)$$

## 2.5.2 Amortecimento

Amortecimento hidrodinâmico em veículos subaquáticos é principalmente provocado por [12]:

- Amortecimento Potencial;
- Arrasto de Atrito;
- Amortecimento por Ondas;
- Amortecimento Viscoso.

No caso de veículos totalmente submersos em águas calmas, o amortecimento mais significativo é o amortecimento viscoso. Esse amortecimento deve-se essencialmente à contribuição linear e quadrática da fricção na camada exterior do corpo, à diferencial de pressão em torno do corpo criada pelo movimento relativo do fluido em relação ao corpo, e ao arrastamento devido aos vórtices criados pelo movimento relativo ao fluido na superfície do corpo.

As forças e momentos de arrasto exercidos sobre o corpo são dependentes das velocidades lineares e angulares relativas:

$$v_r = v - v_f = [u_r, v_r, w_r, p_r, q_r, r_r] \quad (2.48)$$

sendo  $v_f$  o vetor das velocidades do fluido definido anteriormente.

Em termos práticos, a totalidade das forças e momentos de arrasto,  $\tau_V$  pode ser aproximada pelos dois primeiros termos de uma série de Taylor, embora possuam termos de ordem mais elevada:

$$\tau_V = -D_L v_r - d_Q(v_r) \quad (2.49)$$

em que  $D_L \in \mathbb{R}^{6 \times 6}$  é a matriz dos coeficientes dos termos lineares, positiva definida, tal que:

$$D_L = \begin{bmatrix} X_u & X_v & X_w & X_p & X_q & X_r \\ Y_u & Y_v & Y_w & Y_p & Y_q & Y_r \\ Z_u & Z_v & Z_w & Z_p & Z_q & Z_r \\ K_u & K_v & K_w & K_p & K_q & K_r \\ M_u & M_v & M_w & M_p & M_q & M_r \\ N_u & N_v & N_w & N_p & N_q & N_r \end{bmatrix} \quad (2.50)$$

Os termos quadráticos são dados por  $d_Q(v_r) \in \mathbb{R}^{6 \times 1}$ :

$$d_Q(v_r) = - \begin{bmatrix} v_r X_Q |v_r| \\ v_r Y_Q |v_r| \\ v_r Z_Q |v_r| \\ v_r K_Q |v_r| \\ v_r M_Q |v_r| \\ v_r N_Q |v_r| \end{bmatrix} \quad (2.51)$$

sendo  $X_Q, Y_Q, Z_Q, K_Q, M_Q, N_Q \in \mathbb{R}^{6 \times 6}$  são matrizes de termos constantes. Em muitas aplicações, os termos fora da diagonal principal destas matrizes podem ser desprezados, assim como para  $D_L$ .

A soma de todas as forças e momentos devido ao amortecimento viscoso resulta em:

$$\tau_V(v_r) = -D(v_r)v_r \quad (2.52)$$

$$D(v_r) = - \begin{bmatrix} X_u + X_{u|u}|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v + Y_{v|v}|v| & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_w + Z_{w|w}|w| & 0 & 0 & 0 \\ 0 & 0 & 0 & K_p + K_{p|p}|p| & 0 & 0 \\ 0 & 0 & 0 & 0 & M_q + M_{q|q}|q| & 0 \\ 0 & 0 & 0 & 0 & 0 & N_r + N_{r|r}|r| \end{bmatrix} \quad (2.53)$$

Os coeficientes dos termos quadráticos da matriz  $D(v_r)$  são dependentes das áreas projetadas do corpo nos planos formados pelos eixos do referencial que lhe está fixado. Definindo  $A_{xy}, A_{xz}$  e  $A_{yz}$  como sendo as áreas projetadas nos planos  $\{x_B, y_B\}, \{x_B, z_B\}$  e  $\{y_B, z_B\}$  respectivamente, obtêm-se os elementos da diagonal principal:

$$\begin{aligned} X_{u|u} &= -\frac{1}{2}\rho_f C_{du} \int_{A_{yz}} dA \\ Y_{v|v} &= -\frac{1}{2}\rho_f C_{dv} \int_{A_{xz}} dA \\ Z_{w|w} &= -\frac{1}{2}\rho_f C_{dw} \int_{A_{xy}} dA \\ K_{p|p} &= -\frac{1}{2}\rho_f C_{dp} \int_{A_{yz}} |r_{CG}|^3 dA \\ M_{q|q} &= -\frac{1}{2}\rho_f C_{dq} \int_{A_{xz}} |r_{CG}|^3 dA \\ N_{r|r} &= -\frac{1}{2}\rho_f C_{dr} \int_{A_{xy}} |r_{CG}|^3 dA \end{aligned} \quad (2.54)$$

em que  $\rho_f$  é a densidade do fluido em que o corpo está inserido,  $C_{di}$ , com  $i = u, v, w, p, q, r$  é o coeficiente de amortecimento viscoso segundo o eixo  $i$ , que depende da geometria do corpo. Este último pode ser obtido por expressões empíricas ou experimentalmente.

### 2.5.3 Forças e Momento de restauração

As forças e momentos de restauração devem-se à ação da gravidade e, no caso de um corpo submerso num fluido, também da ação do empuxo. Estas forças são expressas pelo seguinte vetor:

$$\mathbf{g}(\boldsymbol{\eta}_2) = \begin{bmatrix} (W - B)\sin(\theta) \\ (B - W)\cos(\theta)\sin(\phi) \\ (B - W)\cos(\theta)\sin(\theta) \\ (y_{CF}B - y_{CG}W)\cos(\theta)\cos(\phi) + (z_{CG}W - z_{CF}B)\cos(\theta)\sin(\phi) \\ (z_{CG}W - z_{CF}B)\sin(\theta) + (x_{CG}W - x_{CF}B)\cos(\theta)\cos(\phi) \\ (x_{CG}W - x_{CF}B)\cos(\theta)\sin(\phi) + (y_{CF}B - y_{CG}W)\sin(\theta) \end{bmatrix} \quad (2.55)$$

No caso especial de um veículo com empuxo neutro, no qual  $W = B$ , e com a distância entre o centro de gravidade CG e o centro de flutuação CF sendo:

$$\mathbf{BG} = [B\bar{G}_x, B\bar{G}_y, B\bar{G}_z]^T = [x_{CG} - x_{CF}, y_{CG} - y_{CF}, z_{CG} - z_{CF}]^T \quad (2.56)$$

obtém-se:

$$\mathbf{g}(\boldsymbol{\eta}_2) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -B\bar{G}_yW\cos(\theta)\cos(\phi) + B\bar{G}_zW\cos(\theta)\sin(\phi) \\ B\bar{G}_zW\sin(\theta) + B\bar{G}_xW\cos(\theta)\cos(\phi) \\ -B\bar{G}_xW\cos(\theta)\sin(\phi) - B\bar{G}_yW\sin(\theta) \end{bmatrix} \quad (2.57)$$

Quando os centros de flutuação e gravidade estão verticalmente alinhados no eixo z, ou seja,  $x_{CG} = x_{CF}$  e  $y_{CG} = y_{CF}$ , o vetor  $\mathbf{g}(\boldsymbol{\eta}_2)$  configura-se no seguinte formato:

$$\mathbf{g}(\boldsymbol{\eta}_2) = [0, 0, 0, B\bar{G}_zW\cos(\theta)\sin(\phi), B\bar{G}_zW\sin(\theta), 0]^T \quad (2.58)$$

## 2.6 DINÂMICA DO VEÍCULO SUBAQUÁTICO

A dinâmica não-linear do veículo subaquático em um meio fluido sem perturbações é expressa pela seguinte equação:

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{C}(\mathbf{v})\mathbf{v} + \mathbf{D}(\mathbf{v})\mathbf{v} + \mathbf{g}(\boldsymbol{\eta}_2) = \boldsymbol{\tau}_A \quad (2.59)$$

sendo:

- $M = M_{RB} + M_A \in \mathbb{R}^{6 \times 6}$  é a soma do tensor inercial de massa, eq. (2.25), com a matriz massa adicionada, eq. (2.36);
- $C(v) = C_{RB}(v) + C_A(v) \in \mathbb{R}^{6 \times 6}$  é a soma do tensor de Coriolis e Centrípeto, eq. (2.29) com o tensor de força e momentos do fluido, eq. (2.37);
- $D(v) \in \mathbb{R}^{6 \times 6}$  é a matriz de amortecimento hidrodinâmico do fluido, eq. (2.53), que incorpora o meio;
- $g(\eta_2) \in \mathbb{R}^{6 \times 1}$  é o vetor de forças de restabelecimento, eq. (2.55), que rege a dinâmica de estabilidade do veículo à perturbações nas coordenadas  $\eta_2 = [\phi, \theta, \psi]$ .

A entrada  $\tau_A$  é o vetor de forças e momentos que relaciona os propulsores à dinâmica do veículo no ambiente subaquático. Este vetor é dependente da transformada de variáveis que o sistema de controle rege, podendo ser modelado em função da rotação da hélice, em função da tensão de entrada nos propulsores ou em função somente das forças e momentos medidos perante o sinal de comando de atuação.

# 3 LOCALIZAÇÃO

## 3.1 INTRODUÇÃO

O termo "localização", em robótica móvel, se refere às ações executadas pelo robô que permitem determinar sua posição no ambiente. É uma área de pesquisa essencial para várias tarefas, como exploração, mapeamento, vigilância e navegação [13].

Uma importante definição em relação aos problemas de localização é a descrição mensurável do ambiente, conhecidos como mapas. Os mapas são descritos em relação a uma referência padronizada, na maioria das vezes a própria Terra. A localização é o processo de estabelecer a relação entre o sistema de coordenadas globais e o sistema de coordenadas locais do robô, baseado nas leituras de sensores e operação dos atuadores. Por meio de operações de transformação de coordenadas, o robô é capaz de determinar sua localização e a de objetos no ambiente a partir do próprio sistema de referência [14]. Como visto no capítulo de Modelagem, ter conhecimento da posição e orientação do robô, ou seja, o vetor  $\eta$  da equação (2.1) para 6 GDLs ou, no caso 2D o vetor  $x_t = (xy\theta)^T$  das coordenadas do referencial local relativo ao referencial global é suficiente para realizar transformação dos sistemas, uma vez que o sistema de referência no qual a localização do robô é expressa seja conhecida.

Para se localizar, são necessárias informações sobre a região a ser navegada, ou seja, é requerido o uso de um mapa que forneça a representação do ambiente (espaço 2D ou 3D). Pode-se atar dois métodos usuais para a realização do mapeamento: por grade de ocupação (*occupancy grid*) e pelo método topológico. O mapeamento por grade de ocupação é uma representação matricial do espaço. Este é dividido em múltiplas células, nas quais o robô tenta se localizar na célula mais próxima da sua posição real, com base em uma estimativa probabilística do seu estado. O mapa topológico necessita de informações prévias sobre a localização de pontos de referência do ambiente e como estes se conectam relativamente uns aos outros. O mapeamento topológico também pode ser integrado com mapas sensoriais construídos a partir dos dados capturados do ambiente por grupos de sensores, ou mesmo com sensores independentes. A integração e uso de diferentes tipos de mapas conferem maior riqueza de informação e inferem na localização e navegação. O uso de um mapa perde o seu sentido se o robô não conseguir localizar sua posição no ambiente.

O maior problema é que a postura do robô (*pose*), ou seja, sua posição e orientação, não pode ser medida diretamente. Ela é determinada a partir de medições de sensores, e tais sensores possuem erro inerente. Dificilmente um sensor é capaz de determinar com precisão a posição e orientação do robô. Além dos sensores serem limitados para estimar a postura, um ambiente dinâmico, como o subaquático, poderá interferir na atuação e percepção. A robótica está atuando em áreas onde a entrada dos sensores está ficando cada vez mais importante, e nas quais o software de controle necessita ser robusto o suficiente para lidar com uma gama de situações, tantas que fica difícil de antecipar todas.

A figura 3.1 mostra um exemplo de um problema de localização 2D no meio aquático.

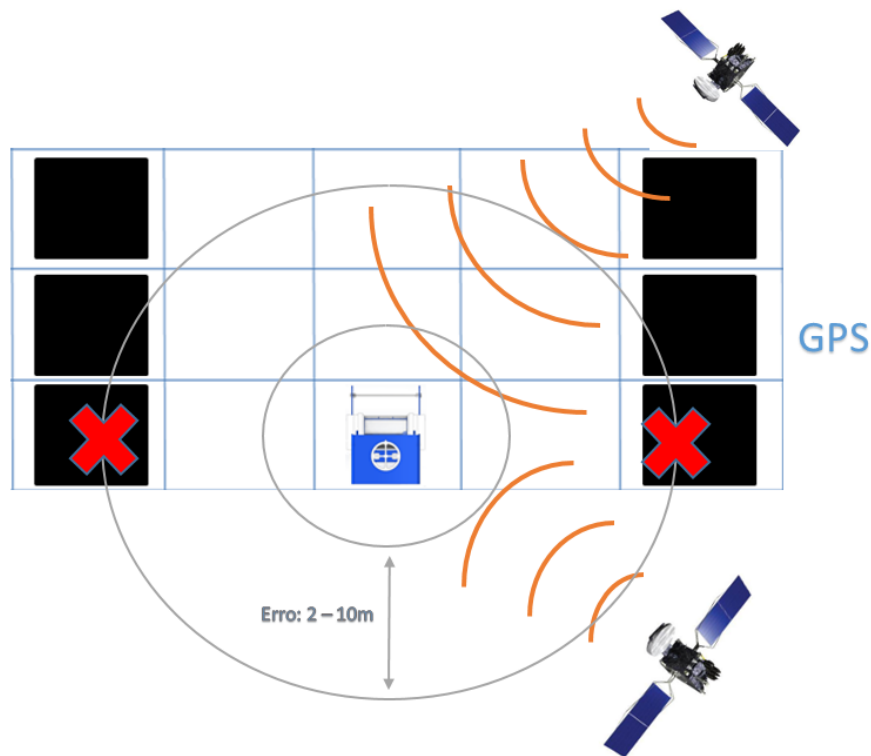


Figura 3.1: Navegação comprometida pela imprecisão do sensor.  
 [Fonte: Elaborada pelo autor.]

O robô aquático que se locomove na superfície (ilustrado neste exemplo com a vista superior do ROV "UnBot") passa pelo canal sem colidir com as barreiras laterais. Para realizar a trajetória de forma autônoma sem colisões, o robô necessita saber onde ele está além de perceber o ambiente. Um meio do robô obter essa informação é através do uso de um sensor GPS (*Global Positioning System*), que recebe os sinais emitidos por satélites e calcula sua posição, fazendo isso durante todo o percurso. O problema desta aplicação é que o sensor GPS não é preciso o suficiente para a navegação, podendo ter um erro de 2 a 10 metros [14] da distância medida e portanto não é preciso o suficiente para evitar a colisão com as barreiras laterais. Além disso, o GPS identifica uma posição absoluta, mas não consegue identificar uma posição relativa a respeito de outros objetos, tendo chance de colidir com as barreiras laterais. Então, para navegar com segurança dentro deste canal é necessário melhorar a leitura de posição, sendo necessário colher informações de outros sensores, utilizar um sensor GPS mais preciso ou combinar sensores para estimar com maior confiança sua localização, além de basear em mapas afim de orientar o sentido da navegação.

Robótica probabilística é uma área de pesquisa que lida com incertezas na percepção e atuação do sistema robótico usando modelos probabilísticos [14]. Sua principal ideia é representar a incerteza de forma expressa, utilizando cálculo da teoria de probabilidade, e algoritmos probabilísticos, nos dados lidos do sensor, para representar a distribuição em um espaço de possíveis hipóteses. Assim, a ambiguidade e a confiança podem ser modelados matematicamente, possibilitando que o sistema trabalhe com a incerteza presente. Isso facilita a aplicação de sistemas em ambientes de difícil modelagem, mesmo que o sistema utilize sensores ruidosos e imprecisos.

## 3.2 TIPOS DE LOCALIZAÇÃO

Os tipos de localização são divididos em dimensões que diferem entre si em relação ao conhecimento prévio que o sistema possui sobre o impacto do ambiente no processo de localização e sobre o peso do próprio algoritmo de localização na navegação do robô.

A primeira dimensão diz respeito ao conhecimento prévio que o robô possui sobre sua localização, há três tipos de problemas em localização definidos [14]:

- **Rastreamento:** Assume-se que o robô possui conhecimento prévio de sua postura, ou seja, sua posição e orientação inicial. A incerteza é modelada como uma distribuição unimodal, sendo sua variância controlada pelo ruído do sensor e do movimento, e a média como a melhor estimativa de posição atual do robô.
- **Localização Global:** Neste caso, o robô não possui conhecimento prévio do seu estado. A metodologia na localização global não pode ser modelada somente como uma boa estimativa, sendo necessário que a distribuição se estenda por todo o terreno para lidar com todas as possíveis soluções, baseadas nas ações executadas pelo robô durante o processo de localização. Por este motivo, distribuições unimodais são inapropriados para este tipo de localização.
- **Problema do Robô Sequestrado:** É uma variação do problema da localização global. Durante a localização, o robô é removido da sua posição atual e colocado em um outro local no terreno. Isso configura uma situação em que o robô acredita estar em um local, mas na realidade, não está. Assim, é necessário que o robô se localize novamente. A importância de tal problema é apontar que os algoritmos atuais de localização não são infalíveis. A necessidade de recuperação de um erro no processo de localização do robô é essencial em sistemas autônomos.

A segunda dimensão se refere ao impacto do meio na localização do robô. Eles são divididos em:

- **Ambientes estáticos:** Neste tipo de ambiente, a única variável é o estado do robô. Somente o robô tem sua posição ou orientação alteradas, enquanto todos os objetos mantêm-se na mesma posição. Ambientes estáticos são preferíveis para atuação de sistemas autônomos, pois são mais fáceis de modelar.
- **Ambientes dinâmicos:** Para estes ambientes, a localização ou configuração dos objetos pode sofrer alterações durante a navegação. A dinâmica do meio interfere nas medidas do sensor, sendo necessário que o robô mude sua percepção a medida que o ambiente se alterar. Alguns ambientes dinâmicos possuem modelos de suas alterações que podem ser utilizados no processo de localização. Porém, a maioria dos ambientes são modelados como processos estocásticos e o robô necessita utilizar uma medida como constante ou esperada, e tratar as outras mudanças no meio não previstas como ruído.

A terceira dimensão diz respeito ao fato de o algoritmo de localização também controlar a navegação do robô. São dois casos:

- **Localização Passiva:** Nesta abordagem, o módulo de localização observa a locomoção do robô e o controle do robô é realizado por outro módulo, de tal modo que o movimento do robô não é executado com o objetivo de auxiliar sua localização.
- **Localização Ativa:** Neste caso, o algoritmo de localização define a movimentação do robô, com o objetivo de minimizar o erro de localização. A localização ativa normalmente obtêm resultados melhores que a localização passiva, porém é melhor implementada quando combinada a alguma rotina de tarefa que o robô precisa executar.

Além dos tipos acima, outras particularidades podem impactar o algoritmo de localização, por exemplo a coleta da informação do sensor e a incerteza na execução dos comando pelo robô.

### 3.3 ROBÓTICA PROBABILÍSTICA

O foco da robótica probabilística é estimar informações sobre o robô a partir de dados coletados pelos sensores. As quantidades estimadas não podem ser observadas diretamente, mas deduzidas medindo outro fenômeno físico relacionado à variável desejada. Os sensores adquirem a informação parcial e são geralmente alterados por ruído, seja na construção do sensor, seja no ambiente que prejudica a medição. Estimar a postura do robô via métodos probabilísticos permite lidar com ruídos e tratar incerteza usando distribuições de probabilidade para estimar a informação necessária ao robô. Essa abordagem apresenta bons resultados em localização, pois a modelagem dos ambientes físicos atuais não consegue prever com precisão todos as possíveis saídas e fontes de erros ao estimar a postura do robô. Ambientes subaquáticos são ainda mais difíceis de modelar, pois é complicado prever o comportamento do fluido ao sofrer perturbação de objetos ou da dinâmica de correntezas e ondas [12, 15, 16]. Portanto, os algoritmos de localização que utilizam filtros probabilísticos fornecem um método eficaz para estimar a informação desejada.

A seguir, a localização via filtro probabilístico é aplicada em um estudo de caso em robótica subaquática <sup>1</sup>. A figura 3.2 mostra um ambiente no qual está presente um robô subaquático que não possui conhecimento da sua posição atual. Esta incerteza é modelada como uma distribuição onde o eixo horizontal corresponde a todas as possíveis localizações neste meio (modelado em uma dimensão neste exemplo) e o eixo vertical corresponde à probabilidade para cada posição neste ambiente. A estimativa momentânea da posição do robô (que é denominada neste trabalho como *crença*) é representada por uma distribuição probabilística pelo espaço de todas as possíveis localizações. Na figura 3.2, uma distribuição uniforme é utilizada e corresponde ao maior grau de incerteza da posição do robô. Nesse exemplo, o espaço de posições é discreto e existem 10 possíveis posições: a probabilidade do robô ocupar uma delas é a mesma, ou seja, 10% como informa o gráfico de barras na figura.

<sup>1</sup> Adaptada do exemplo de robótica móvel de. Thrun [14]



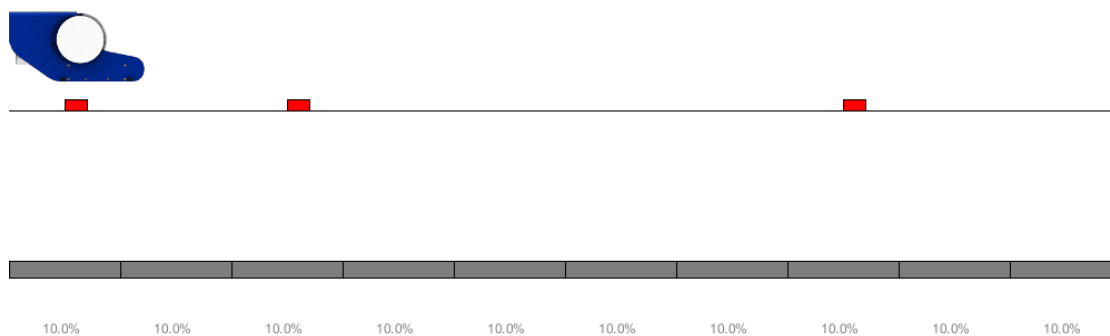


Figura 3.2: Localização Global em um terreno unidimensional - com incerteza representada em distribuição uniforme.

[Fonte: Elaborada pelo autor.]

Para localização, o ambiente precisa de características distintas (pontos de referência), que neste exemplo são três marcadores vermelhos idênticos no chão. Estes marcadores são comumente chamados de *landmarks*. O robô possui um sensor (câmera) orientado para o chão que identifica se há *landmark* embaixo dele ou não. O robô possui conhecimento prévio da localização de todos os *landmarks*, sendo essencial para que ele se localize.

Suponha que o robô faça a leitura do sensor e descubra que ele está localizado bem acima de um *landmark*. Essa informação na função resulta em uma nova crença, ilustrada na figura 3.3, que mostra uma maior probabilidade nas localizações próximas dos *landmarks* e uma menor probabilidade onde não existem *landmarks*. Percebe-se que há três picos na distribuição, cada um correspondendo a um *landmark* no mapa. Isso ilustra o fato de que a metodologia probabilística pode lidar com múltiplas hipóteses, onde há mais que uma possível solução para o problema. Também pode-se notar que as localizações onde não existem *landmarks* no mapa possuem uma probabilidade não-nula, ou seja, há uma pequena chance que o robô esteja nessas posições. Isso é decorrente da incerteza inerente dos sensores.

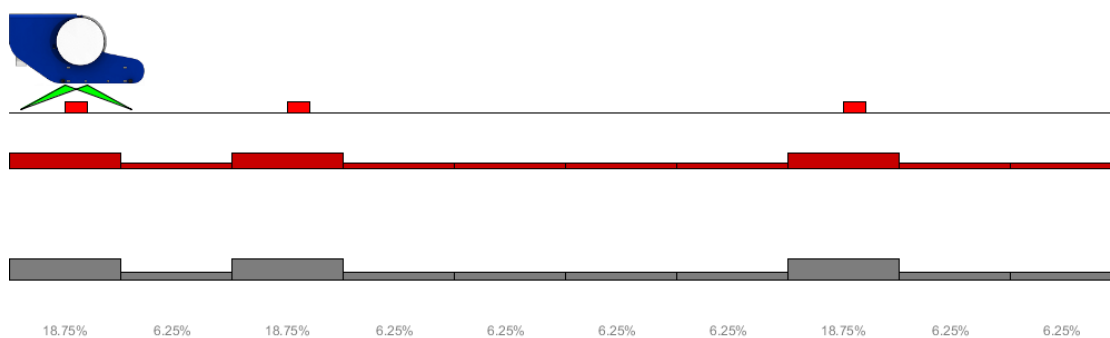


Figura 3.3: Distribuição alterada com medição de sensor.

[Fonte: Elaborada pelo autor.]

A figura 3.4 ilustra o efeito do movimento para frente do robô na crença sobre sua posição. A distribuição de probabilidade deslocou-se à frente, porém o valor da probabilidade dos picos diminuiu e a variância

aumentou como consequência do erro presente na atuação do robô.

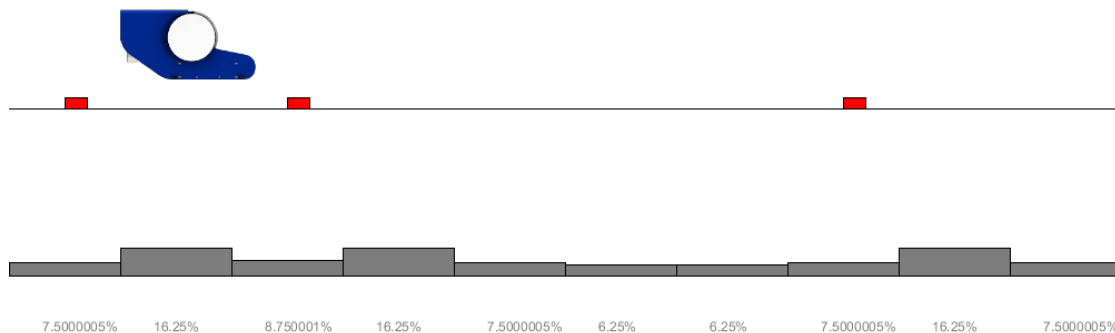


Figura 3.4: Distribuição alterada com mudança de posição do robô.  
 [Fonte: Elaborada pelo autor.]

Finalmente, a figura 3.5 apresenta a distribuição de probabilidade da crença após o robô perceber outro marcador vermelho. Essa medida pelo sensor aumenta o valor da probabilidade do pico na localização do *landmark* percebido, ilustrando que há uma alta probabilidade que o robô esteja nessa posição, oferecendo uma possível solução para o problema da localização global.

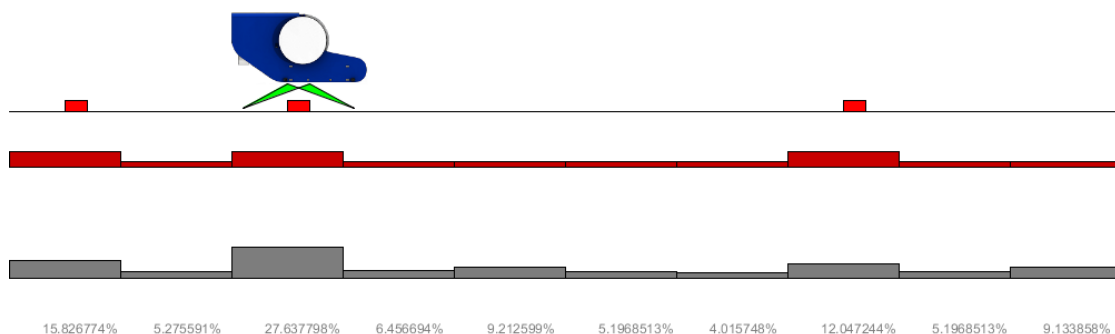


Figura 3.5: Distribuição da confiança ao identificar um segundo marcador. Pico maior concentrado na posição estimada do robô.  
 [Fonte: Elaborada pelo autor.]

No instante mostrado na figura 3.5, só uma leitura do sensor é considerada. Caso houvesse mais leituras da câmera, como ilustrado na figura 3.6, a leitura iria reforçar o pico com maior probabilidade, diminuindo a incerteza da primeira medição. Múltiplas medições do mesmo estado oferecem mais informações sobre o meio, aumentando assim a crença de localização do robô.

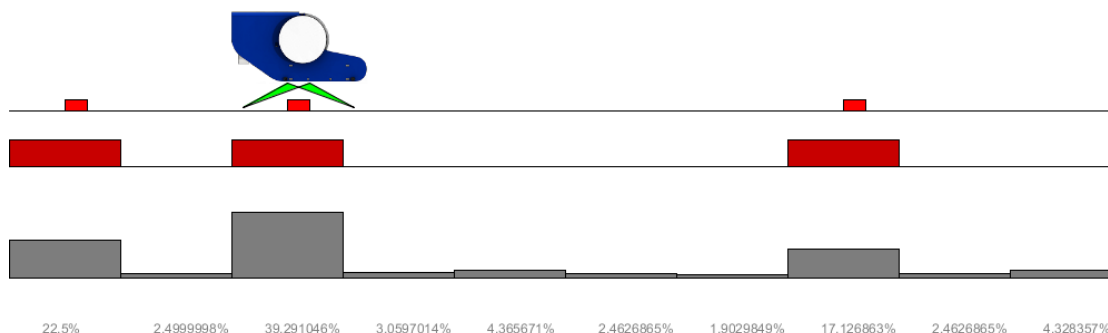


Figura 3.6: Aumento dos picos de confiança ao fazer uma nova medição do marcador.  
[Fonte: Elaborada pelo autor.]

O exemplo acima demonstra o processo de localização baseado em filtros probabilísticos. A incerteza continua presente durante a navegação, porém é analisada em conjunto com as informações captadas pelo sensor. Assim, é possível conseguir encontrar a posição do robô, mesmo possuindo modelos incompletos ou incertezas tanto na percepção quanto na atuação do robô. Os filtros probabilísticos utilizados na robótica são baseados no filtro de Bayes, que é apresentado a seguir.

### 3.4 BASE TEÓRICA - FILTRO DE BAYES

A distribuição da crença atribui um valor de probabilidade para cada hipótese, independente do real estado do robô. As distribuições são probabilidades posteriores, calculadas com base em informações disponíveis ao sistema robótico. A crença ( $bel(x_t)$ , de *belief*) é função do estado em um determinado momento,  $x_t$ , expressa como [14]:

$$bel(x_t) = p(x_t | z_{1:t}, u_{1:t}). \quad (3.1)$$

O lado direito da equação (3.1) representa a distribuição de probabilidade sobre o estado em um tempo  $t$ , baseado em todas as medidas passadas dos sensores e todas os movimentos passados, onde  $x_t$ ,  $z_{1:t}$  e  $u_{1:t}$  representam, respectivamente, as posições, medidas dos sensores e movimentos ao decorrer do tempo  $t$ . A distribuição de crença é calculada após a incorporação da medição dos sensores. Para auxiliar na sua determinação, é interessante definir uma variável  $\overline{bel}(x_t)$  que representa a crença prévia sem a informação adquirida no instante  $t$ , sendo expressa como:

$$\overline{bel}(x_t) = p(x_t | z_{1:t-1}, u_{1:t}). \quad (3.2)$$

A determinação da distribuição de probabilidade que representa a crença prévia  $\overline{bel}(x_t)$ , a partir das ações de movimento do robô  $u_{1:t}$ , é conhecida como *previsão*. O movimento impreciso do robô, graças a incerteza gerada pela dinâmica dos atuadores, faz com que a crença prévia tenha uma variância maior, ou seja, o sistema torna-se mais incerto sobre o seu estado atual.

A incorporação da certeza inferida pelo sensor na crença prévia é conhecida como *atualização*, e com ela determina-se a crença atual  $bel(x_t)$ . As medições feitas pelos sensores produzem informações sobre o ambiente ao robô, sendo que ao adicionar a nova informação à distribuição, possibilita-se que a nova crença calculada ofereça mais certeza sobre o estado do robô.

A *previsão* é calculada pelo teorema da Probabilidade Total, pois o movimento é um evento independente da confiança adquirida. Conhecendo a distribuição de confiança anterior,  $bel(x_{t-1})$ , a *previsão* da confiança é calculada pela convolução da distribuição do modelo movimento, que expressa a probabilidade do robô executar a ação desejada com a confiança anterior. A equação da *previsão* é:

$$\overline{bel}(x_t) = \int p(x_t|u_t, x_{t-1})bel(x_{t-1})dx, \quad (3.3)$$

onde  $p(x_t|u_t, x_{t-1})$  é a probabilidade do robô situar na posição  $x_t$  dado o movimento  $u_t$  e a posição prévia  $x_{t-1}$ .

Diferentemente, a *atualização* é baseada no Teorema de Bayes, um método fundamental em dedução probabilística. Quando aplicado na atualização é expresso como:

$$bel(x_t|z_t) = \frac{p(z_t|x_t)\overline{bel}(x_t)}{p(z_t)} \quad (3.4)$$

A distribuição de probabilidade de medida do sensor,  $p(z_t|x_t)$  é interpretada como a probabilidade da medida em função da posição no terreno. Esta distribuição depende do funcionamento do sensor e como sua medição se relaciona com a posição estimada do robô. No exemplo das figuras 3.2 a 3.6, o sensor é capaz de identificar se há marcador ou não, e nesse caso a sua precisão é relativamente alta. Assume-se que a probabilidade do sensor em identificar que há marcador onde realmente tem marcador é de 0.8, enquanto a probabilidade do sensor afirmar que há marcador em uma posição onde não há marcador é de 0.2. Isso pode ser visualizado na figura 3.3, na primeira distribuição vermelha, que é a função *atualização*. O elemento restante na equação 3.4 é  $p(z_t)$ , que é a probabilidade da medida do sensor sem qualquer informação de posição no mapa. Essa probabilidade é considerada constante em qualquer posição, pois o sensor é considerado independente de leitura passada, não *biased*. O problema se origina em como calcular o  $p(z_t)$ , já que não temos conhecimento do valor desta probabilidade. Usando a regra de Bayes novamente, chegamos à equação da crença expressa como:

$$p(z_t) = \int p(z_t|(x_t)_i)\overline{bel}((x_t)_i) \quad (3.5)$$

Com as duas operações definidas, pode-se descrever o filtro de Bayes do seguinte algoritmo:

### Algoritmo 3.1: Filtro de Bayes

```
Filtro_Bayes ( $bel(x_{t-1}, u_t, z_t)$ ):  
    para todos os valores positivos de  $x_t$  faça :  
         $\overline{bel}(x_t) = \int p(x_t|u_t, x_{t-1})bel(x_{t-1})dx$   
         $bel(x_t|z_t) = \frac{p(z_t|x_t)\overline{bel}(x_t)}{p(z_t)}$   
    fim  
    retorna  $bel(x_t)$ 
```

## 3.5 FILTRO HISTOGRAMA

O exemplo apresentado nas figuras 3.2 a 3.6 ilustra as fases do funcionamento da localização por malha, utilizando filtro histograma em um terreno unidimensional. Este filtro é modelado para atuar em um mundo discreto, onde o modelo movimento do robô é calculado para cada célula individual no eixo de movimento. Ao diminuir o tamanho das células, tal método aproxima-se do algoritmo de localização de Markov, um algoritmo de localização global utilizando filtro de Bayes, porém como o algoritmo analisa todo o terreno a cada iteração, não é recomendável para implementação em grandes ambientes. A figura 3.7 ilustra como a distribuição discretizada fica mais perto de uma distribuição contínua quanto maior a divisão do mundo em células menores. Ressalta-se que o pico é alinhado com a localização da parte traseira do robô subaquático.

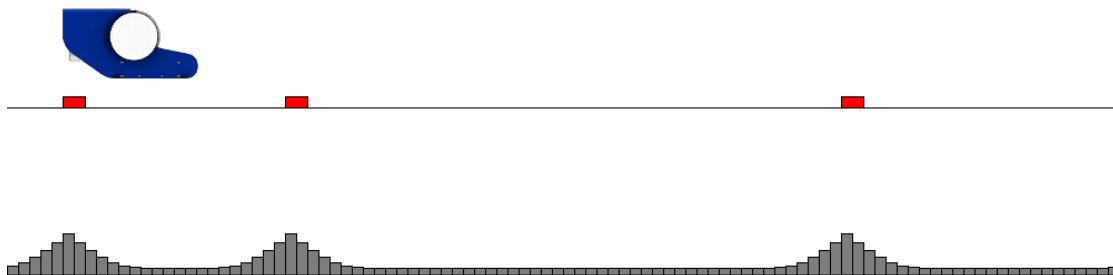


Figura 3.7: Localização por malha - Filtro histograma com distribuição ampliada para 100 células no vetor posição no terreno.

[Fonte: Elaborada pelo autor.]

O filtro histograma é um variante do filtro de Bayes. O algoritmo do filtro histograma é expressa como:

### Algoritmo 3.2: Filtro Histograma

```
Filtro_Histograma ( $bel(x_{t-1}, u_t, z_t, m)$ ):  
  para todos os valores positivos de  $x_t$  faça:  
     $\overline{bel}(x_t) = \text{modelo\_movimento}(bel(x_{t-1}, u_t))$   
     $bel(x_t|z_t) = \text{modelo\_sensor}(\overline{bel}(x_t), z_t, m)$   
  fim  
  retorna  $bel(x_t)$ 
```

As distribuições de incerteza no movimento e ruído de sensor são modeladas dentro das funções *modelo movimento* e *modelo-percepção*. As operações nessas funções são as mesmas do filtro de Bayes, mas aplicadas a um ambiente discreto e descritas pelo mapa dado.

A figura 3.8 demonstra como o tempo de cálculo do algoritmo de localização aumenta em relação ao tamanho da célula. Este fato limita a utilização do filtro histograma para utilização de localização global em missões de mapeamento ou análise em grandes ambientes. Métodos alternativos pode ser aplicados para melhorar o desempenho de cálculo, mas recomenda-se utilizar outros algoritmos de localização global em meio contínuo nas missões onde discretizar o espaço do ambiente é mais complexo ou de grandes dimensões [14].

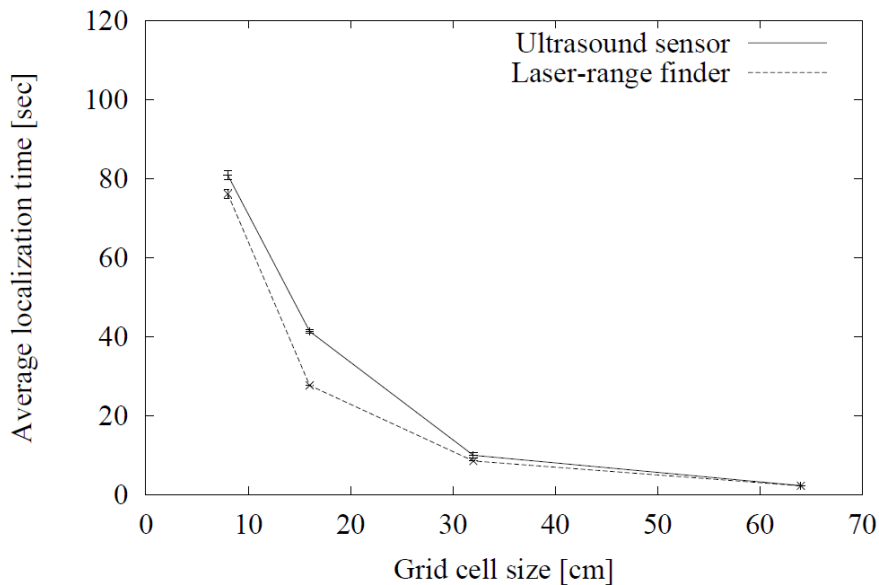


Figura 3.8: Tempo médio requerido do CPU para localização global em função da resolução da célula na matriz discreta, com curvas utilizando sensores de ultrassom e LIDAR (*Light Detection and Ranging*).

[Fonte: Livro *Probabilistic Robotics*<sup>2</sup>.]

<sup>2</sup> (Figura retirada do livro *Probabilistic Robotics* [14], pg. 193.)

### 3.6 FILTRO DE PARTÍCULAS

Filtro de partículas, também conhecido como Algoritmo de Localização de Monte Carlo (MCL), é um algoritmo popular que representa a função confiança como partículas. Similar ao filtro histograma, o algoritmo MCL é aplicável tanto em localização localizada quanto global, e também é multimodal, ou seja, pode trabalhar com múltiplas hipóteses. O MCL é utilizado em uma variedade de aplicações que necessitam localização, sendo uma escolha preferencial por ser um algoritmo robusto, simples de ser implementado e reduz o custo computacional do cálculo.

A figura 3.9 ilustra um exemplo de localização global utilizando filtro de partículas. Neste exemplo, o robô subaquático foi posto em uma posição desconhecida, sendo necessário ser localizado para posteriormente planejar sua rota até o alvo. O algoritmo MCL cria partículas (neste exemplo 1000 unidades espalhadas pelo mapa sem aglomeração) que são exibidas na crença atual no ambiente. Cada partícula representa um possível valor de postura do robô no ambiente, ou seja, para o exemplo apresentado em 2D, cada um é um vetor direção que possui uma possível posição e orientação  $particula_i = [x, y, \theta]$ . Como pode-se observar, a dispersão das partículas se estende por todo o terreno, ilustrando a incerteza da sua posição e orientação.

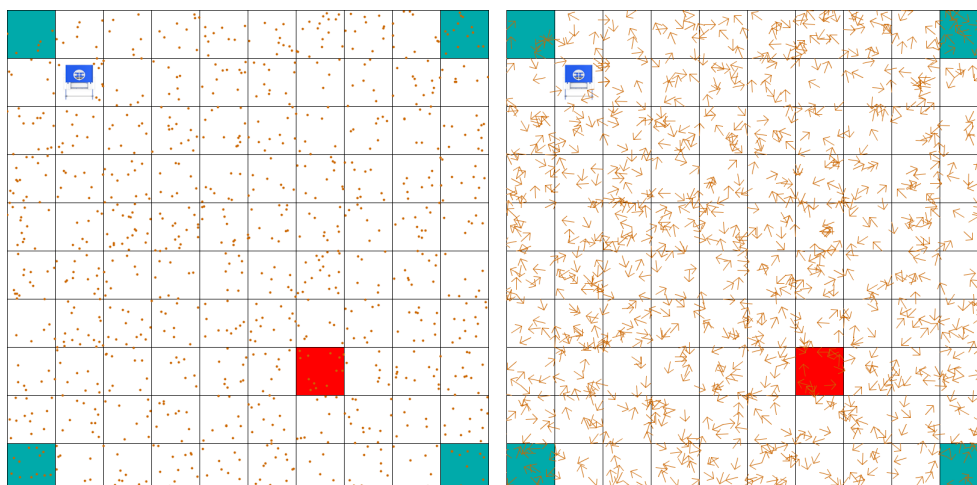


Figura 3.9: Ilustração da incerteza de localização do robô subaquático com partículas distribuídas em posições e orientações aleatórias pelo terreno bidimensional.

[Fonte: Elaborada pelo autor.]

Como informado na seção 3.3, é necessária alguma característica distinta no ambiente para que o robô possa se localizar. Neste exemplo, o terreno possui quatro pontos de referência, posicionados nos cantos do mapa. O robô tem conhecimento prévio da posição destes pontos no mapa, e para auxiliar sua localização, ele possui um sonar tipo *long baseline* [6], que consegue medir a distância entre sua posição atual e a posição dos pontos de referência. Como todo sensor, as medidas do sonar são afetados por ruído. Este ruído é modelado e implementado no modelo de percepção, que juntamente com o modelo movimento, são incorporados ao algoritmo MCL.

Então, o robô se locomove em busca de encontrar sua localização e toma medidas com seu sensor. O movimento faz com que todas as partículas se movam, seguindo a distribuição resultante do modelo de

movimento. Este fato não muda muito a distribuição das partículas, pois a ação de movimentar possui imprecisão inerente da atuação. Assim as partículas mantêm-se dispersas após o movimento, ilustrado na figura 3.10, sendo necessário agregar mais informações para diminuir a incerteza, que é obtida ao se fazer uma medição de distância dos pontos de referência via sonar.

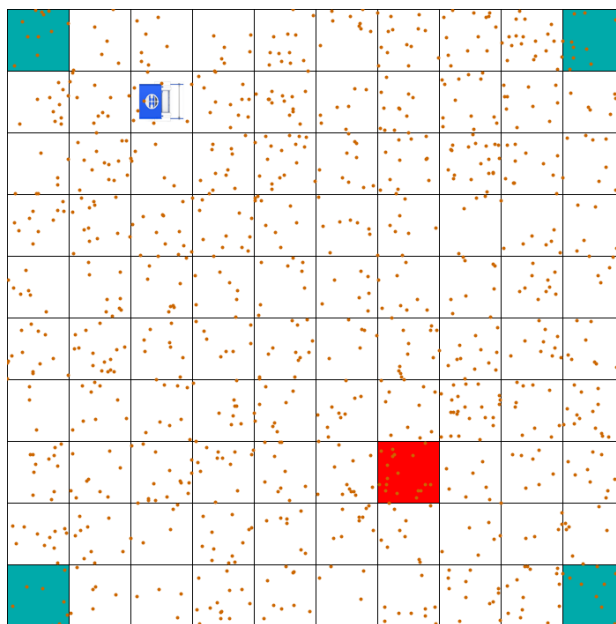


Figura 3.10: Ilustração das partículas após movimento.

[Fonte: Elaborada pelo autor.]

O processo que ocorre com todas as partículas é a comparação entre a medida real do sensor, incluindo o ruído, e a medida prevista da partícula. Para cada partícula, é feita uma análise da diferença da medida real e da medida prevista, ilustrada na figura 3.11.



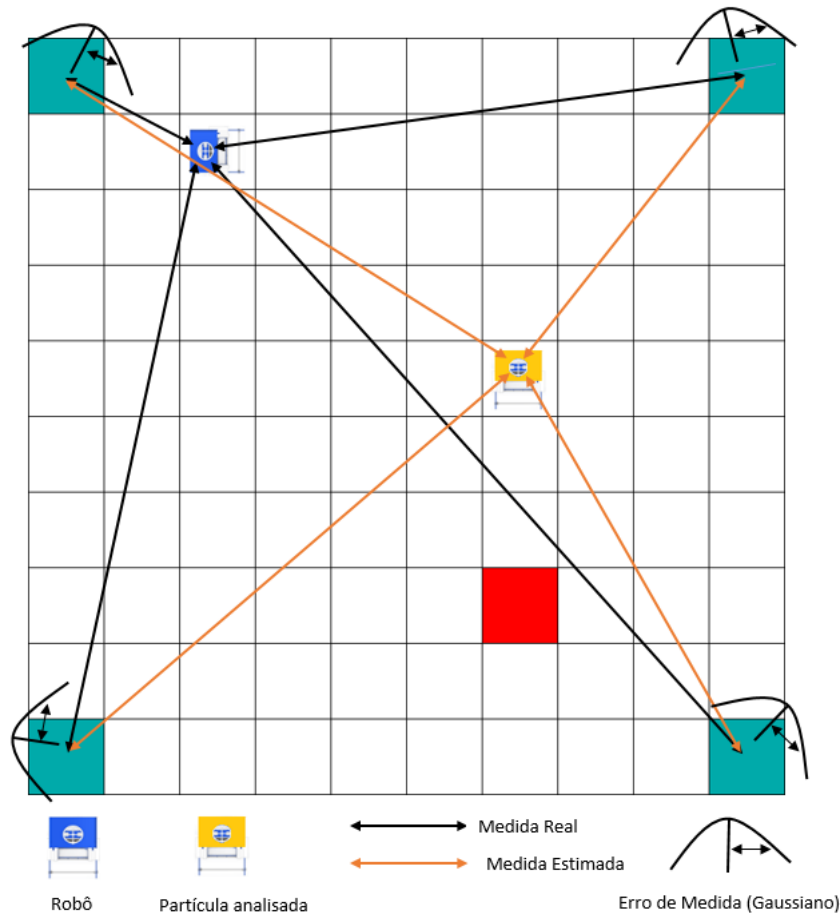


Figura 3.11: Comparação da medição prevista de uma partícula com o robô real.  
 [Fonte: Elaborada pelo autor.]

Baseado nessa diferença, é atribuído um peso de importância  $w_i$  para cada partícula. Quanto maior o  $w_i$ , mais importante é a partícula, configurando com isso que o estado desta partícula é mais plausível para solucionar o problema de localização. A figura 3.11 ilustra algumas partículas que possuem maior peso com posição perto do robô e as partículas mais longes com peso menor. A partir da variável peso  $w_i$ , o algoritmo de filtro de partículas efetua uma seleção de sobrevivência das partículas. Essa seleção é proporcional ao peso de importância atribuído a cada partícula, como no caso da figura 3.12, as partículas com peso maior têm mais chances de sobreviver do que as partículas com peso menor. Após um processo chamado *reamostragem*, um novo grupo de partículas é criado para substituir a distribuição antiga, reagrupando-as em posições baseadas proporcionalmente ao peso de importância das partículas antigas. A distribuição nova posiciona as partículas em regiões com maior probabilidade de localização do robô, mas ainda existem partículas em outras regiões, representando a incerteza da medida afetada por ruído.

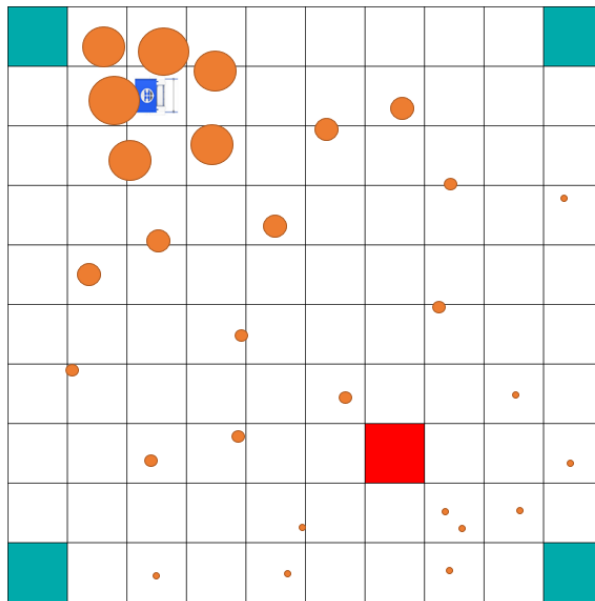


Figura 3.12: Ilustração das partículas com pesos variados  
 [Fonte: Elaborada pelo autor.]

Após a *reamostragem*, a nova distribuição das 1000 partículas concentra-se na região próxima ao robô, oferecendo uma estimativa melhor de onde se localiza no terreno, como mostra a figura 3.13.

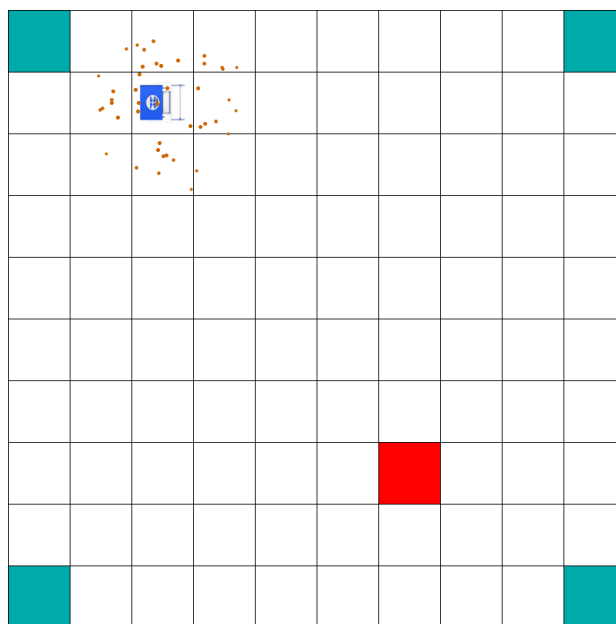


Figura 3.13: Concentração das partículas redistribuídas após locomoção e medição via sonar.  
 [Fonte: Elaborada pelo autor.]

Para reduzir a incerteza de posição de 9 células, aplica-se novamente o filtro de partículas após movimento e medição do sonar, redistribuindo as partículas agora para três células, como se pode observar na figura 3.14. É possível perceber que a distribuição planar das partículas em x (eixo horizontal) possui variância pequena, quase concentrada em uma única célula, enquanto em y (eixo vertical) a variância é

maior. Este fato demonstra como as partículas são aglomeradas baseadas nas informações adquiridas e expressas na distribuição prévia. O robô movimentou-se na direção  $x$ , diminuindo a incerteza neste vetor da sua posição, porém não adicionou informação no eixo  $y$ , exceto em medidas ruidosas do sonar, que expressa a incerteza nessa variável posição em  $y$ .

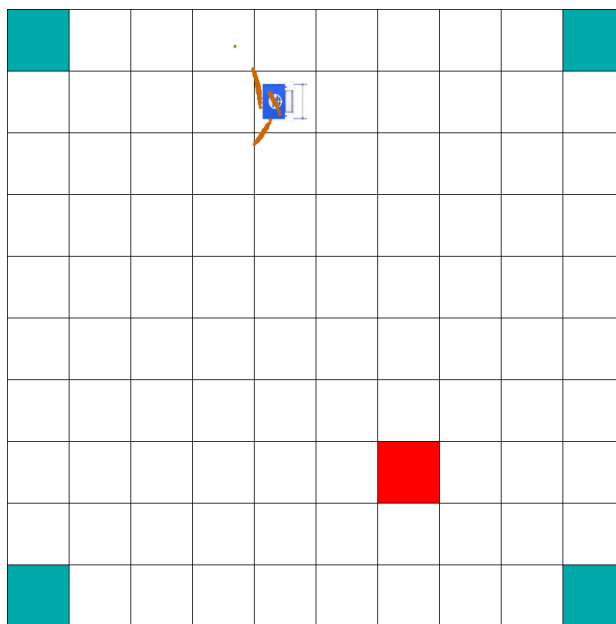


Figura 3.14: Nova redistribuição após movimento e sonar.  
[Fonte: Elaborada pelo autor.]

Baseado na distribuição prévia, o robô faz uma curva e desloca-se no eixo  $y$  para diminuir a incerteza nessa direção. Com poucos movimentos e medições, foi possível redistribuir as partículas com as variâncias de  $x$  e  $y$  quase totalmente confinados em uma única célula, e possuindo a média das orientações das partículas na direção e sentido correto, como ilustra a figura 3.5.

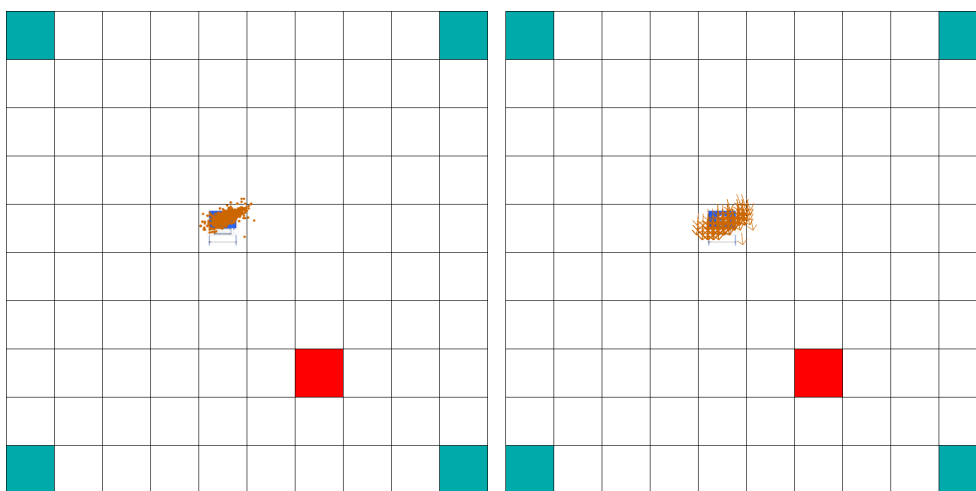


Figura 3.15: Localização global resolvida para posição e orientação.  
[Fonte: Elaborada pelo autor.]

O algoritmo do filtro de partículas é o seguinte:

Algoritmo 3.3: Filtro de Partícula

```

Filtro_Partículas ( $particulas_{t-1}, u_t, z_t, m$ ):
     $\overline{particulas}_t = []$ 
    para todos  $particulas_{t-1}$  faça:
         $p(i) = \text{modelo\_movimento}(particulas_{t-1}(i), u_t)$ 
         $w(i) = \text{modelo\_sensor}(\overline{particulas}_t(i), z_t, m)$ 
         $\overline{particulas}_t(i) = (p(i), w(i))$ 
    fim
     $particula_t = \text{reamostragem}(\overline{particulas}_t)$ 
    retorna  $particula_t$ 

```

A função *reamostragem* utilizada neste texto é baseada no algoritmo de reamostragem de roda [17]. Este algoritmo representa todos os pesos das partículas dentro de uma roda de pesos, ilustrada na figura 3.16, seccionando os pesos na roda proporcionalmente ao seu valor. Partículas com pesos maiores ocupam um espaço maior, enquanto partículas com pesos menores são mais estreitas na roda.

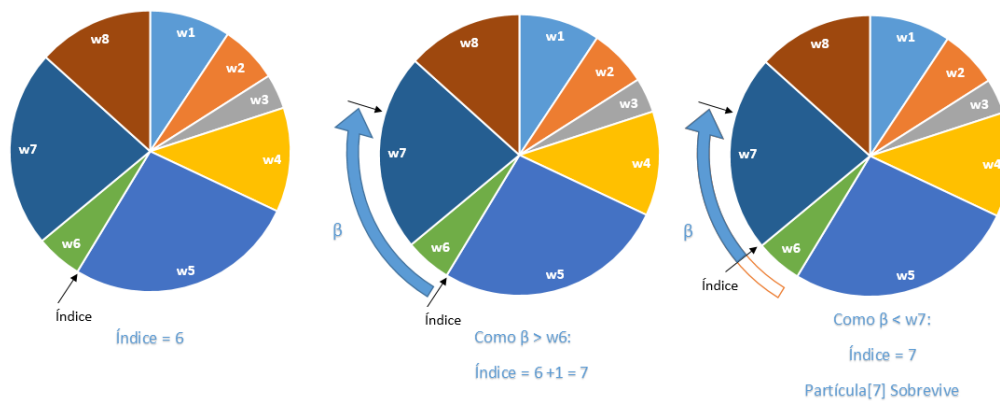


Figura 3.16: Ilustração da roda de reamostragem baseado no exemplo descrito.

[Fonte: Elaborada pelo autor.]

Inicialmente estima-se um elemento do vetor *particulas* por amostragem do índice a partir de uma distribuição uniforme  $U[1...N]$ . Imagina-se que o primeiro índice sorteado, refere-se a uma partícula com peso  $w_6$ . Neste momento, inicializa-se uma variável  $\beta = 0$  na qual se atribui um valor sorteado de uma distribuição uniforme de 0 até  $2w_{max}$ , na qual  $w_{max}$  é o maior peso de todas as partículas, que é  $w_5$  no exemplo. Suponha-se que o valor de  $\beta$  foi tal que ele se encontra dentro da região de  $w_7$ . Entra-se em um *loop* para verificar se enquanto  $\beta > w_6$  (que é o caso deste exemplo) subtrai-se de beta o peso deste índice, e incrementa o índice por uma unidade. No exemplo, essa situação novamente é avaliada ao decorrer da iteração e verifica-se que  $\beta < w_7$  é válida. No caso positivo, faz-se sobreviver a partícula deste índice na próxima amostragem, senão executa-se novamente a rotina. Este algoritmo valoriza as partículas com maior peso, em proporção maior que as outras, porém pode haver situações onde o  $\beta$  selecionado na distribuição uniforme é tão pequeno, que ocorre a sobrevivência de uma partícula de menor peso. Este fato

proporciona a continuação da incerteza na próxima amostragem, que é importante porque mantém uma certa confiança de que a atribuição de pesos pode ter sido corrompida por ruído na medição, permitindo uma forma de reavaliar em novas medições. A figura 3.16 ilustra como seria uma visualização do processo de reamostragem e o algoritmo abaixo é a função implementada em conjunto com o filtro de partículas.

Algoritmo 3.4: Função Reamostragem

```

Função_Reamostragem(particulas):
    particulasnovas = []
    indice = U[1...N]
     $\beta = 0$ 
    Para i = 1...N:
         $\beta = \beta + U[0...2w_{max}]$ 
        Enquanto  $w[\textit{indice}] < \beta$ :
             $\beta = \beta - w[\textit{indice}]$ 
            indice = indice + 1
        fim Enquanto
    particulasnovas[i] = particulas[indice]
    fim Para
    retorna particulasnovas

```

Baseado no filtro de Bayes, o filtro de partículas utiliza o mesmo formato para as funções *previsão*, que é a atualização do estado após movimento, e a *atualização de sensor*, como ilustrado na figura 3.16. A função *atualização de sensor* é aquela que faz a reamostragem das partículas existentes, baseada em novas informações captadas pelos sensores, que atribuem pesos para as partículas. Com isso, a nova distribuição terá incorporado a confiança resultante da informação captada. A função *previsão* faz uma amostragem completa das partículas, modificando os estados de cada uma delas baseada nas probabilidades inerentes do movimento do robô. As equações na figura 3.17 mostram que o filtro partículas é coerente com as teorias do Teorema de Bayes e Probabilidade Total, porém em visão modularizada de partículas no espaço contínuo.

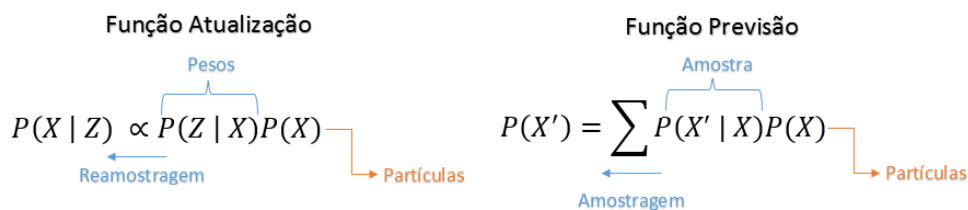


Figura 3.17: Equações das funções *Atualização* e *Previsão*.

[Fonte: Elaborada pelo autor.]

### 3.7 SUMÁRIO

Foi apresentado neste capítulo como o algoritmo do filtro de partículas conseguiu, após algumas iterações, obter a localização global. Este método apresenta vantagens quando comparado ao método histograma, anteriormente apresentado, e ao método mais usado em localização atual, o filtro de Kalman [14]. O filtro de partículas possui as seguintes características:

- **Multi-modal:** O filtro partícula consegue lidar com múltiplas hipóteses decorrentes das ambiguidades presentes em ambientes com simetria, algo que não é possível com o filtro de Kalman.
- **Contínuo:** Este filtro pode ser aplicado em um estado de espaços contínuos, comparado ao filtro histograma que é discreto. Assim, é possível analisar um mapa topológico sem grade de ocupação, o qual apresenta um terreno contínuo onde só se conhece a posição dos pontos de referência.
- **Eficiência:** A eficiência do filtro de partícula é variável, pois é possível modificar o número de partículas para obter o melhor tempo de localização, baseado nas medidas e na plataforma que serão implementadas. Quanto maior o número de partículas, mais provável que o problema de localização seja resolvido, mas também maior é o tempo de cálculo. É possível otimizar este valor, baseando-se em um limite predeterminado de ações, para obter a localização.

Assim, o filtro de partículas é uma escolha possível para implementação de um módulo robusto de localização global para robôs subaquáticos em ambientes dinâmicos.

# 4 PLANEJAMENTO DE TRAJETÓRIA

## 4.1 INTRODUÇÃO

Após a utilização da metodologia de localização global, o robô necessita saber como chegar no alvo previamente definido para completar seu objetivo. O processo de encontrar um caminho de uma posição inicial até o alvo é chamado de planejamento de trajetória. Ao planejar, o robô precisa saber onde poderá navegar para evitar colisões com obstáculos presentes no meio, sendo necessário um mapa do terreno. Da mesma forma, é necessário avaliar o melhor caminho nos casos em que pode haver mais de uma rota para atingir o objetivo, como mostra a figura 4.1.

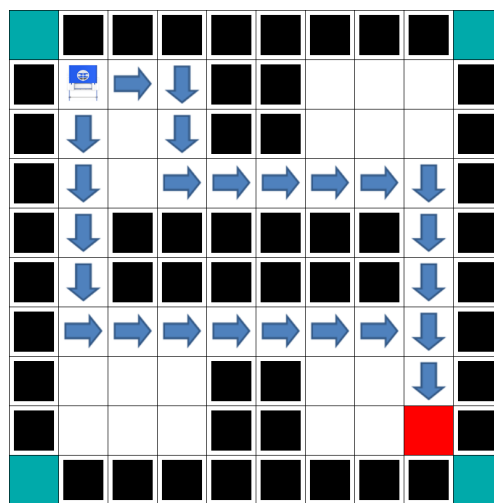


Figura 4.1: Robô em terreno conhecido com múltiplas soluções de rota.

[Fonte: Elaborada pelo autor.]

Para o planejamento do melhor caminho é necessária uma função de custo. O custo é um parâmetro mensurável que quantifica o gasto para completar a ação de locomoção. Há vários tipos de custos que podem ser configurados para planejamento em robótica móvel, sendo os mais populares o tempo de locomoção ou energia gasta. A função de custo não precisa ser necessariamente de uma única variável, podendo unir vários tipos de custos com diferentes ponderações para modelar o quanto o sistema robótico gasta para executar o movimento. Assim, o planejamento de trajetória tem o objetivo de achar não só um caminho para o objetivo, mas o caminho que minimiza a função de custo. Dependendo da função, poderá existir mais do que uma solução ótima, sendo a critério da modelagem do software de planejamento escolher qual caminho seguir.

A figura 4.2 ilustra um exemplo de planejamento de trajetória. O terreno é modelado como um mundo discretizado, dividido em uma matriz 2D composta de células quadradas de área fixa. O robô possui conhecimento da sua posição inicial e da sua orientação, além de saber a posição do alvo e a localização dos obstáculos presentes no meio.

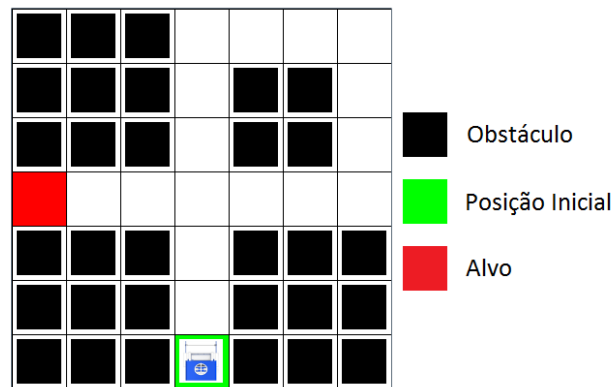


Figura 4.2: Terreno exemplo para planejamento.  
 [Fonte: Elaborada pelo autor.]

O robô possui um número limitado de ações que pode realizar durante a locomoção. No caso do veículo "UnBot", os seguintes movimentos no plano são possíveis:

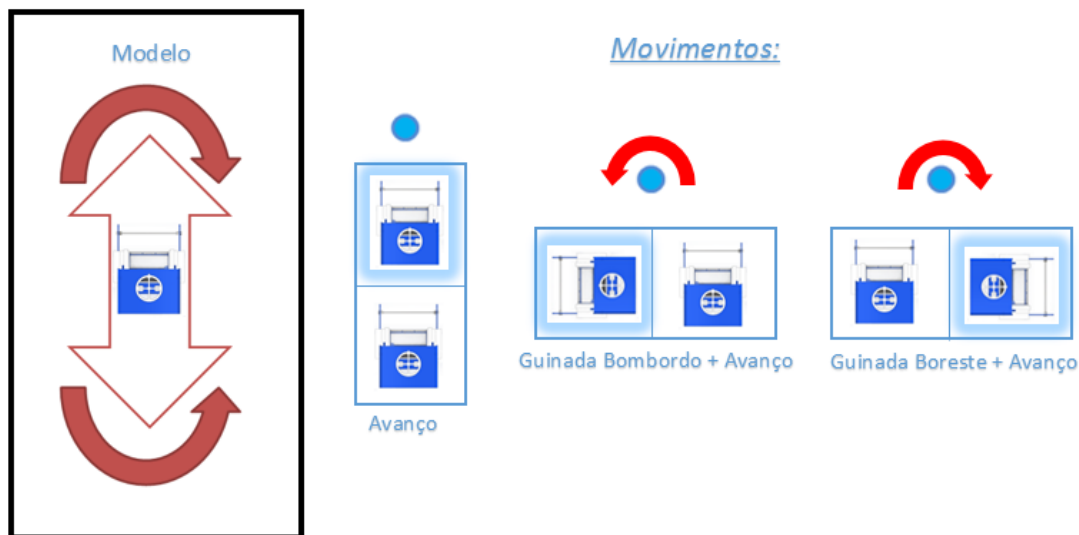


Figura 4.3: Modelagem dos movimentos do "UnBot".  
 [Fonte: Elaborada pelo autor.]

Observe que cada um dos três movimentos sempre avança uma célula no mapa. Cada avanço terá um gasto, que poderá ser modelado baseado em energia, tempo ou em outro parâmetro que deseja-se otimizar. Esse gasto é chamado de custo, o qual se relaciona com o tipo de movimento por uma função de custo. Neste exemplo, a função de custo será modelada como custo unitário, ou seja, cada movimento apresentado terá o mesmo gasto e é modelado em algoritmo como:

Algoritmo 4.1: Função Custo

```
Para todos os movimentos:
    custo = 1 * ação
fim Para
```



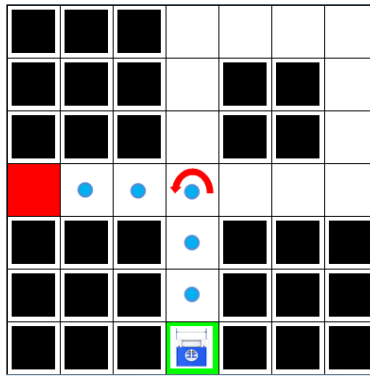


Figura 4.4: Solução analítica do problema de planejamento no terreno exemplo.  
[Fonte: Elaborada pelo autor.]

Tendo conhecimento de todos os parâmetros previamente listados, o caminho de menor custo para atingir o alvo no exemplo da figura 4.2 será 6, sendo cinco de avanço somado a uma guinada bombordo + avanço.

Caso a função de custo fosse distinta entre os movimentos, a trajetória poderia ser alterada. Utilizando o mesmo mapa, posição inicial e alvo, altera-se o custo do movimento guinada bombordo + avanço para um valor de 20, mostrando que o robô tem preferência de fazer os outros movimentos (em um caso real, poderia se configurar na ação de uma correnteza ou falha de força em um dos motores).

Algoritmo 4.2: Função Custo Modificado

```

Caso movimento == "guinada bombordo + avanço":
    custo = 20 * ação
Senão
    custo = 1 * ação

```

Assim, a trajetória ótima seria a seguinte:

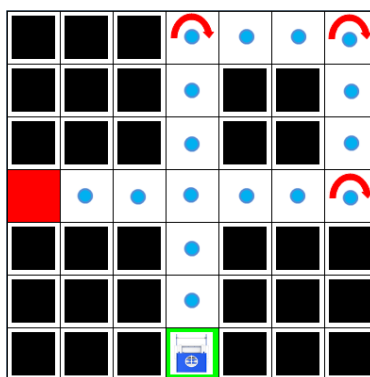


Figura 4.5: Solução analítica do problema de planejamento com função de custo modificado.  
[Fonte: Elaborada pelo autor.]

No final a melhor trajetória teria o custo 18, que é menor que a trajetória com guinada bombordo + avanço que teria no total 25.

Este exemplo ilustra como seria o planejamento de trajetória com escolha de rotas distintas, de acordo com a função de custo. Porém, em mapas de exploração maiores, essas trajetórias são mais complicadas de serem obtidas, além do fato de poderem existir rotas com custos iguais, sendo necessário adotar algum critério de análise para escolher qual seguir. Com esta necessidade exposta, demonstra-se a importância de um algoritmo de busca de melhor trajetória dado o mapa, a posição inicial e posição final desejado.

## 4.2 FUNÇÃO DE EXPANSÃO

Uma metodologia comum para resolver o problema de busca da trajetória de menor custo é utilizar uma função de expansão. A figura 4.6 ilustra um terreno onde o robô tem conhecimento da sua posição, a localização do alvo e o mapa. Este terreno, sendo discretizado, é modelado como uma matriz de início [0,0] até o comprimento e largura do terreno, neste caso [6,6].

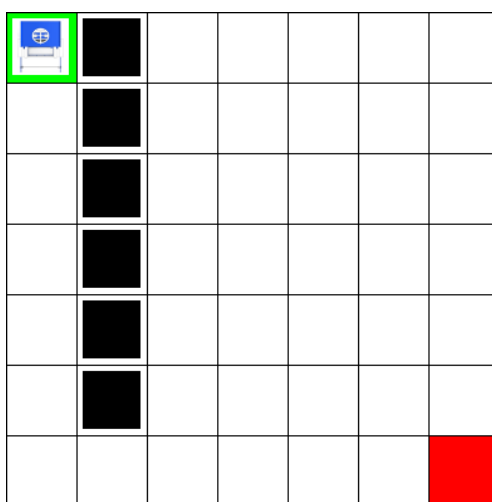


Figura 4.6: Terreno exemplo para planejamento de rota.

[Fonte: Elaborada pelo autor.]

A ideia básica da função de expansão é de manter uma lista das células que deseja investigar dentro da matriz [17]. Inicialmente, cria-se uma lista denominada *Aberto*, onde guarda-se a informação da posição em  $x$  e  $y$  da célula investigada e a *valor g* (que corresponde ao número em vermelho nos vetores ilustrados na 4.7). O *valor g* é uma variável que representa a quantidade de vezes que utilizou-se a função expansão para chegar até esta célula. No exemplo ilustrado na 4.7, a primeira célula analisada, que corresponde à posição inicial do veículo, possui o *valor g* nulo, porque não houve nenhuma chamada anterior da função de expansão.

Em cada iteração, a célula investigada é analisada para saber se atingiu o alvo. Caso não seja, aloca-se a célula investigada para a lista *fechada*, na qual são guardadas todas as células previamente analisadas, e chama-se a função expansão para analisar as células adjacentes, expandindo primeiramente as que possuem o menor *valor g*. Assim, encontra-se somente a célula [0,1] na lista *aberto*, pois a célula [1,0] está ocupada por um obstáculo. A célula é analisada para saber se é o alvo e caso não seja, chama-se a função expansão. Isso ocorre cada vez que o algoritmo é executado: acrescenta-se células na lista *aberto*, em seguida analisa-

se as células e atribui-se o *valor g* e depois aloca-se para lista fechada até atingir o alvo.

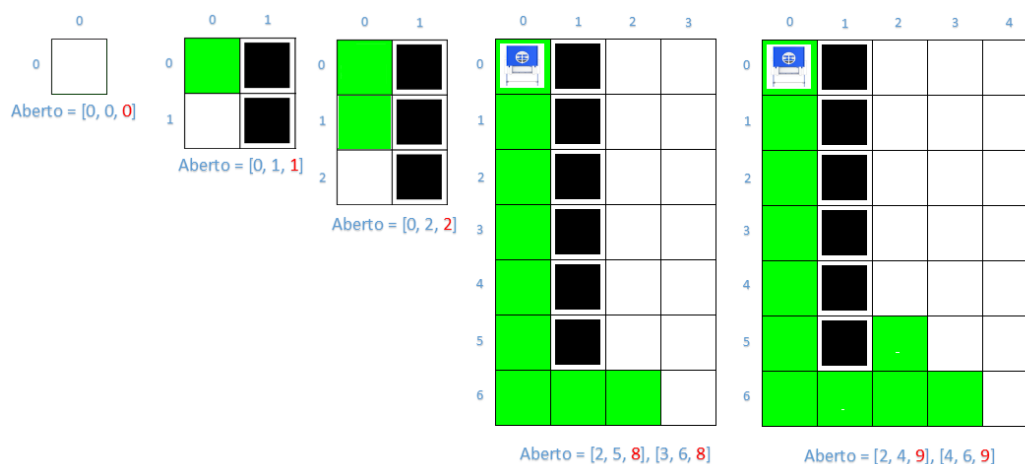


Figura 4.7: Procedimento de cálculo da função de expansão.

[Fonte: Elaborada pelo autor.]

Após todas as células do mapa serem submetidas a iteração, é montada a trajetória da posição do alvo (ilustrado na 4.8), analisando quais células adjacentes possuem o menor valor *g* e segue-se até atingir a posição inicial.

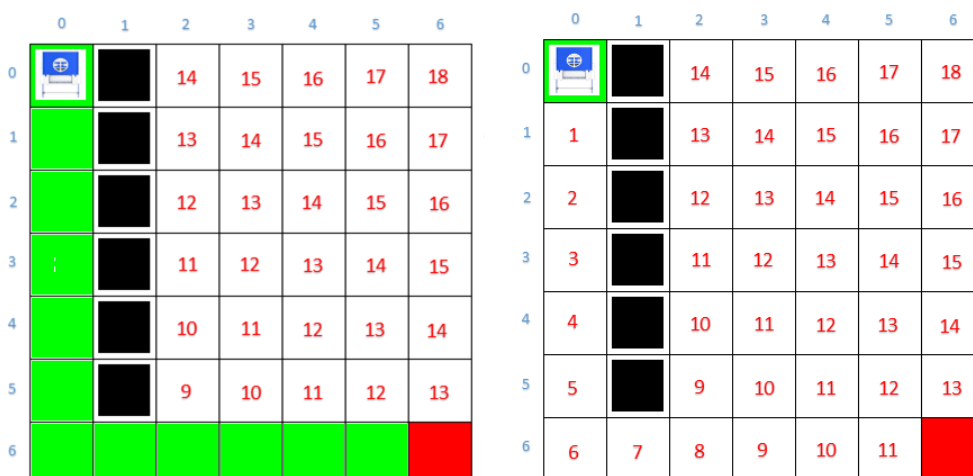


Figura 4.8: Resultado final do cálculo da função de expansão e rota ao alvo encontrado.

[Fonte: Elaborada pelo autor.]

A função expansão possui a capacidade de encontrar a melhor rota, porém ela não é eficiente, uma vez que é necessário expandir para todas as células do mapa. Porém, existe uma alternativa que otimiza a função expansão, implementada em um algoritmo chamado *A-star*.

### 4.3 A-STAR

Criado por Peter Hart, Nils Nilsson e Bertram Raphael [18], *A-star* (ou  $A^*$ ) é um dos algoritmos mais utilizados para busca de trajetórias em meios discretos. *A-star* utiliza uma função heurística juntamente com a função expansão para otimizar o processo de busca de melhor rota sem precisar analisar todo o terreno. A função heurística determina um *valor h* para cada célula do terreno, que corresponde ao número de passos do alvo até a trajetória. Observa-se que a função heurística determina um valor até nas células que possuem obstáculo.

Para verificar o funcionamento do algoritmo *A-star*, é necessário determinar a matriz heurística e executar a função de expansão. Em cada célula, é computado um *valor f*, que é a soma do *valor g* da célula analisada juntamente com o *valor h* da função heurística dessa posição, como ilustrado na 4.9. Assim, o algoritmo *A-star* busca na iteração sempre expandir para a célula adjacente com menor *valor f*, ao invés do menor *valor g* anteriormente feito quando executava somente a função de expansão. O diferencial deste algoritmo é que a lista *aberto* pode possuir múltiplas células com mesmo *valor g*, porém o *valor f* é diferente, determinando que a célula que minimiza essa função seja a próxima a ser analisada.

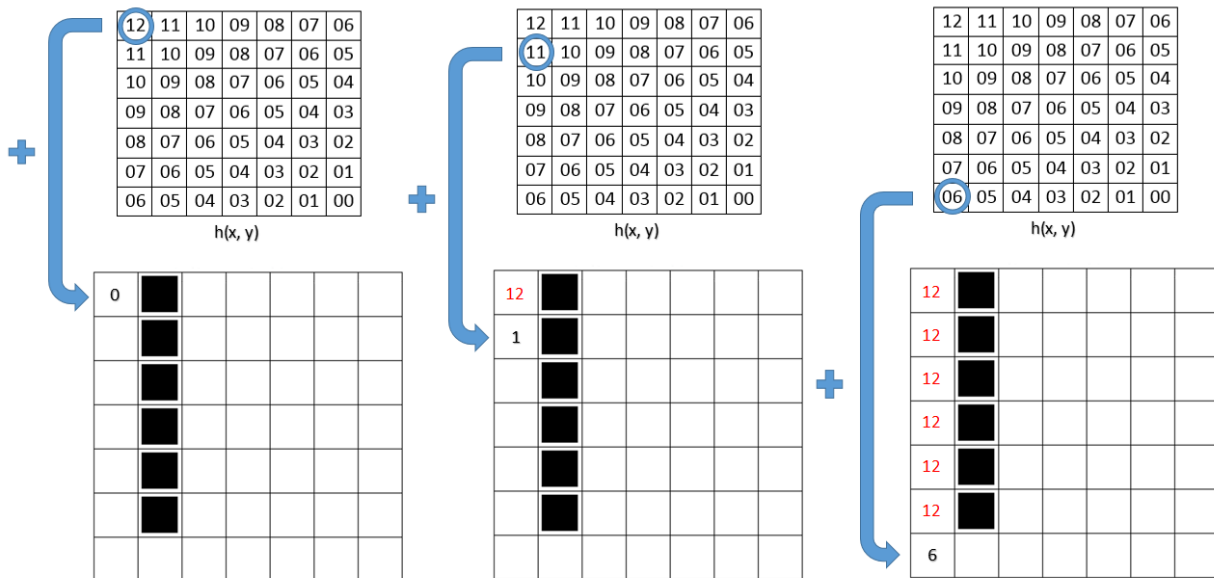


Figura 4.9: Procedimento da cálculo do *valor f* pelo algoritmo *A-star*.

[Fonte: Elaborada pelo autor.]

Assim, o algoritmo *A-star* direciona a busca no caminho ótimo, economizando o poder computacional que seria utilizado para calcular o restante das células do terreno. A figura 4.10 ilustra o resultado das matrizes das funções de expansão e heurístico na matriz final do *A-star*.

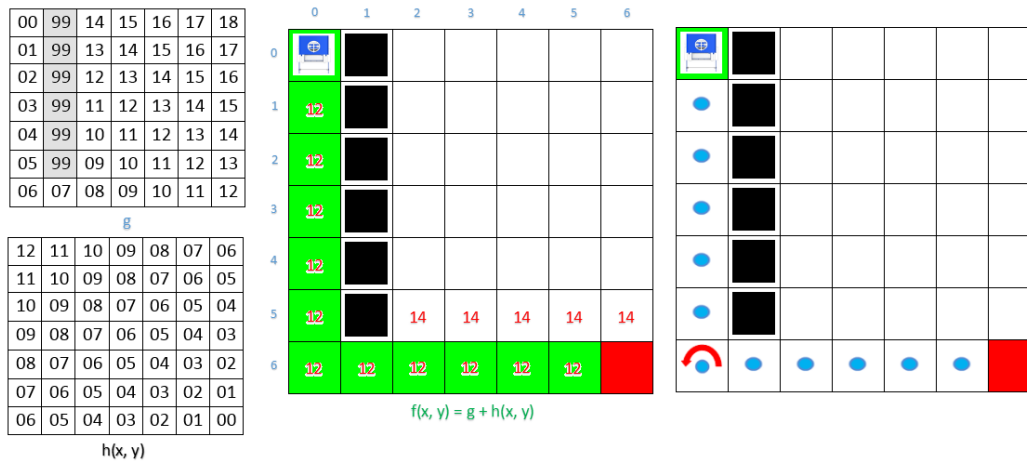


Figura 4.10: Matrizes expansão e heurística resultando na trajetória ótima via *A-star*.

[Fonte: Elaborada pelo autor.]

A divisão do terreno em uma grade de ocupação de células quadradas é intuitiva, porém não produz rotas suaves e realísticas na aplicação de sistemas no contínuo. Um processo proposto por [19] propõe a alteração do projeto da malha quadrada com células para uma malha composto por triângulos. Este método tem a vantagem de possibilitar a modelagem mais próxima ao real dos obstáculos e especificar onde realmente possui área livre, já que qualquer polígono poderá ser aproximado por triângulos.

#### 4.4 DYNAMIC PROGRAMMING

Neste momento, o robô possui um meio de encontrar a sequência de ações necessárias para atingir seu objetivo, porém o ambiente poderá causar um distúrbio na locomoção e tirar o robô da sua rota. Assim, é necessário que o sistema robótico se localize após o distúrbio e defina uma nova trajetória. O algoritmo *A-star* poderá se implementado novamente para encontrar a trajetória, mas em algumas situações é conveniente um método de busca mais robusto para encontrar a série de ações necessárias para atingir o alvo independente de onde o robô estiver. Este método é implementado no algoritmo *Dynamic Programming*.

O algoritmo *Dynamic Programming* oferece um plano prévio, independentemente da localização do robô. Os ambientes subaquáticos são estocásticos, ou seja, eles se alteram no decorrer do tempo por ações de natureza ou intervenção humana. Distúrbios no meio fluido originados por translações de navios, correnteza ou ondas podem interferir no movimento do robô durante sua missão, de tal forma que desviem seu rumo durante a navegação. Dependendo da capacidade computacional do robô e da velocidade de deslocamento, é interessante prever as medidas necessárias a serem feitas caso aconteça um distúrbio, para obter uma trajetória alternativa que talvez não fosse a ótima anteriormente planejada.

Inicialmente é executada a *função valor*, que é uma função que atribui a cada célula a distância da trajetória mais curta até o alvo. Ela é parecida com a função heurística implementada no *A-star*, porém desta vez ele analisa somente as células que estão livres de obstáculos.

Função Valor  $v(x, y)$   
 $v(x, y) = \min v(x', y') + \text{custo}$

12		10	09	08	07	06
11		09	08	07	06	05
10		08	07	06	05	04
09		07	06	05	04	03
08		06	05	04	03	02
07		05	04	03	02	01
06	05	04	03	02	01	00

Figura 4.11: Matriz gerada da função valor aplicada no terreno.

[Fonte: Elaborada pelo autor.]

Terminando este procedimento inicia-se uma rotina para comparar que ação atribuir para cada valor presente nas células. Como o robô atua somente no plano, utiliza-se o seguinte esquema para criar o guia de ações:

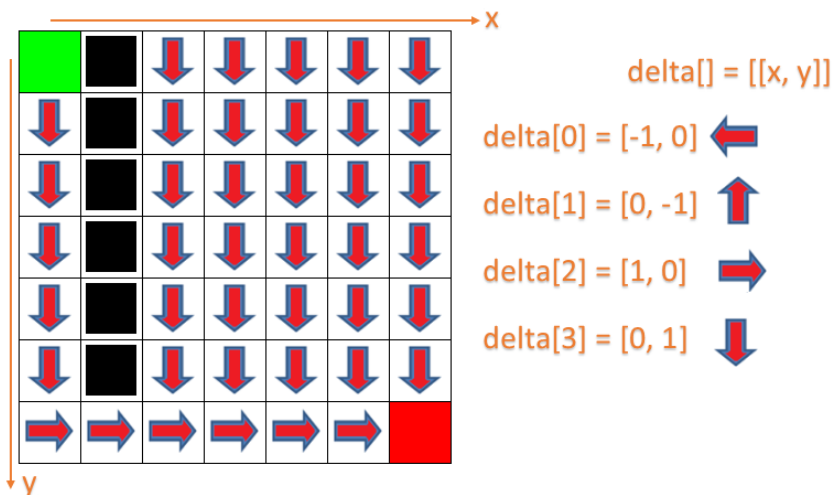


Figura 4.12: Série de ações em todas as posições no terreno resultante do algoritmo *Dynamic Programming*.

[Fonte: Elaborada pelo autor.]

Com a lista *delta* de ações criadas, executa-se a iteração para saber qual das ações utilizar, analisando o valor das células adjacentes, dessa forma atribuindo o delta correspondente à célula que satisfaz o valor somado da célula analisada. A figura 4.12 ilustra o resultado do algoritmo *Dynamic Programming*, no qual atribui-se uma ação a cada célula do terreno. Observa-se que o resultado mostrado possui a política *A-star* presente, porém é possível que o algoritmo forneça outra alternativa caso haja ambiguidade de trajetória, sendo a critério do programador inserir uma função de escolha de trajetórias, ou como feito neste trabalho, escolher a última solução ótima encontrada.

# 5 DESENVOLVIMENTO

## 5.1 INTRODUÇÃO

Este capítulo apresenta a metodologia utilizada para determinar os parâmetros físicos de veículos subaquáticos baseados no que foi apresentado nos capítulos anteriores. São estabelecidas as matrizes regentes da dinâmica do corpo imerso em um fluido e definido o modelo físico que permita a análise de seu comportamento frente ao movimento próprio efetuado pelos propulsores ou pelas perturbações no meio.

## 5.2 METODOLOGIA DO TRABALHO

Em termos gerais, o problema proposto neste trabalho consiste em estabelecer uma metodologia para proceder à navegação autônoma em veículos subaquáticos. A navegação autônoma consiste em o veículo trafegar no meio aquoso numa rota otimizada, sem intervenção humana, adaptando seu movimento a perturbações externas. A princípio, o robô necessita:

- Conhecer o mapa e seu objetivo;
- Se localizar no terreno, obtendo sua posição inicial estimada;
- Executar um planejamento de rota para preparar diretrizes de movimentos;
- Locomover-se e controlar seu estado por meio de sensores até atingir o seu objetivo.

Como ocorre em qualquer problema lógico, é necessário determinar qual será a estratégia de resolução, definindo os pressupostos e segmentando um problema complexo em partes menores. A escolha do paradigma é o primeiro passo para montagem do sistema inteligente, definindo a lógica do fluxo de informações que dominam a operação do veículo.

### 5.2.1 Paradigmas da Robótica

Paradigma é uma filosofia ou conjunto de pressupostos que caracterizam uma metodologia de solução para uma classe de problemas. Combina-se o modo de observar o mundo e o conjunto de ferramentas implícitas para resolver problemas. Aplicar o paradigma certo é essencial para simplificar o processo de resolução de problemas em sistemas inteligentes. Atualmente, existem três tipos de paradigmas para organizar inteligência: hierárquico, reativo e híbrido [20]. Todos os paradigmas são descritos em duas formas:

1. Sobre o relacionamento entre três primitivos aceitos pela robótica: Percepção (*SENSE*), Planejamento (*PLAN*) e Atuação (*ACT*), ilustrados na fig. 5.1.

- *SENSE* é a função que coleta informações do ambiente por meio de sensores que produzem uma saída útil para outras funções.
  - *PLAN* é a função que trata as informações adquiridas (seja por *SENSE* ou informações prévias do ambiente e objetivos) e produz uma série de tarefas para o robô executar.
  - *ACT* corresponde às funções de comando dos atuadores.
2. Sobre o modo como as medições dos sensores são processadas e distribuídas pelo sistema. A dificuldade surge na modelagem do modo como os sensores vão receber as informações e quais variáveis serão retiradas. Já as outras variáveis presentes na medição serão consideradas como ruído.

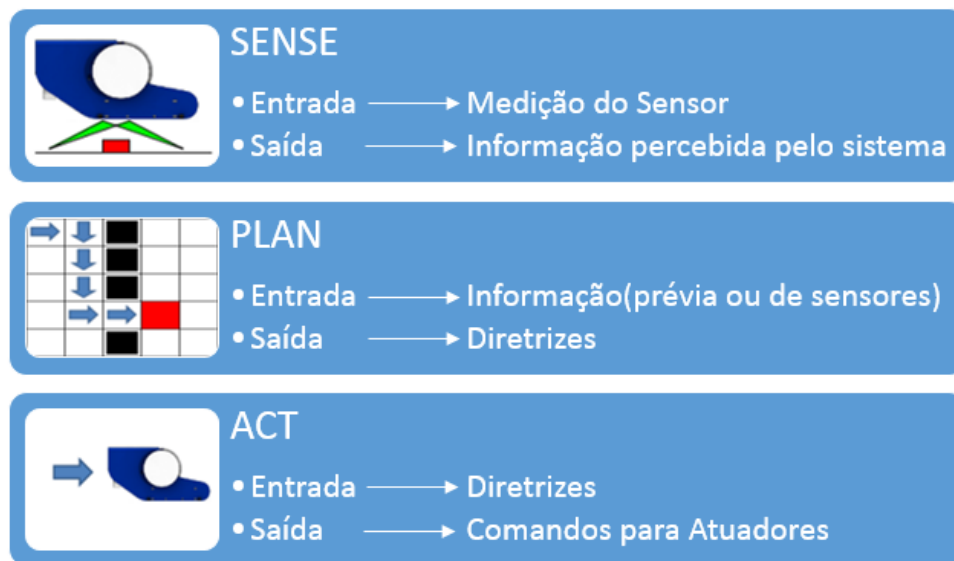


Figura 5.1: Os três primitivos da robótica.  
[Fonte: Elaborada pelo autor.]

Atualmente há três paradigmas [20] que apresentam organizações diferentes das funções *SENSE*, *PLAN* e *ACT* para esquematizar a inteligência na robótica:

1. **Paradigma Hierárquico:** Inicialmente o robô obtém a percepção do ambiente (*SENSE*), a utiliza como entrada para efetuar planejamento da próxima ação (*PLAN*) e executa o comando (*ACT*), como esquematizado na fig. 5.2. A cada passo esse ciclo é repetido sempre possuindo conhecimento do que será feito a cada momento. Uma característica marcante neste paradigma é que toda informação colhida por sensores é introduzida em um modelo global do ambiente, no qual o algoritmo de planejamento se baseia para montar suas diretrizes. A desvantagem desse método é que o sistema necessita remontar o modelo global a cada medição nova para analisar e preparar as diretrizes. Este processo irá resultar em respostas lentas a estímulos ou perturbações do meio no sistema.



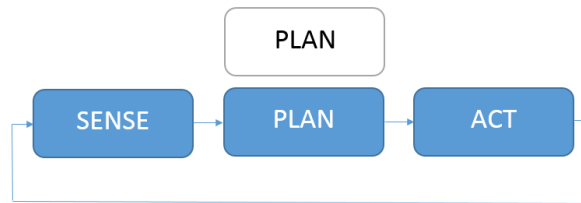


Figura 5.2: Fluxograma do Paradigma Hierárquico.  
[Fonte: Elaborada pelo autor.]

2. **Paradigma Reativo:** Foi uma alternativa proposta para resolver os problemas de processamento presentes no Paradigma Hierárquico. Com base em evidências biológicas e psicologia cognitiva, conclui-se que os seres vivos respondem rapidamente às perturbações, utilizando a percepção como entrada direta à atuação, não fazendo planejamento do reflexo ao distúrbio. O método reativo implementa esta filosofia, retirando o módulo de planejamento do ciclo de inteligência e interligando o *SENSE* com o *ACT*, como ilustrado na fig. 5.3. Isso possibilita que o robô não necessite de todas as informações dos sensores para planejar uma ação, fazendo com que cada sensor ou conjunto de sensores sejam ligados diretamente a um atuador, possibilitando que ações múltiplas executem e ao mesmo tempo gerem um resultado idêntico ao que foi anteriormente planejado no paradigma hierárquico. Mesmo com respostas rápidas e inteligentes aos estímulos externos, robôs com paradigmas reativos possuem carência de executarem tarefas globais por não haver um módulo de planejamento para direcionar as atuações locais.



Figura 5.3: Fluxograma do Paradigma Reativo.  
[Fonte: Elaborada pelo autor.]

3. **Paradigma Híbrido:** Neste método, inicialmente o sistema determina como dividir o objetivo global em várias tarefas locais. Conhecido como planejamento da missão, o supramencionado método determina quais estados são recomendados para cumprir as tarefas. A medição dos sensores é roteada para cada estado que necessita daquele sensor, mas também é disponível ao planejador para construção de um modelo global orientado em tarefas. O esquema desse paradigma está ilustrado na fig. 5.4. O planejador pode ser configurado para supervisionar a execução reativa em cada estado e para diagnosticar o progresso das ações em relação ao objetivo local e global. Cada função tem sua própria taxa de atualização. O *PLAN*, como necessita de mais processamento, pode ser atualizado em um período maior do que a leitura de sensores e atuação.

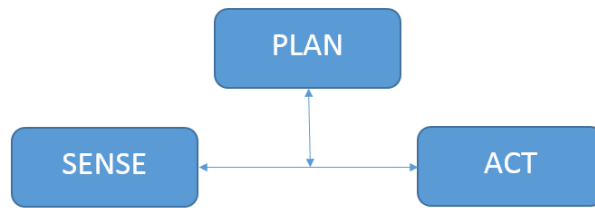


Figura 5.4: Fluxograma do Paradigma Híbrido.  
[Fonte: Elaborada pelo autor.]

O Paradigma Híbrido apresenta uma melhor escolha no método de resolução do problema de navegação autônoma. A complexidade dinâmica do meio exige que o veículo possua resposta rápida para contrapor possíveis distúrbios, porém é necessário que o veículo siga o objetivo local determinado pelas diretrizes do primitivo de planejamento. A localidade, o tipo de sensores presentes e o momento em que podem ser utilizados são também indicadores para esse paradigma segmentado, pois dependendo da visibilidade do ambiente é interessante uma troca de sensores, o que irá alterar a função de percepção. Há lugares em que uma câmera informa mais do que o sonar e há outros onde acontece o contrário. Novamente as diretrizes globais e o acompanhamento das ações locais são os itens que irão alterar a estratégia reativa de ações do robô em relação ao que ele percebe do meio.

### 5.2.2 Esquemático de Solução

A figura 5.5 ilustra a segmentação de análises necessárias para obter os três primitivos *SENSE*, *PLAN* e *ACT* em um veículo subaquático.

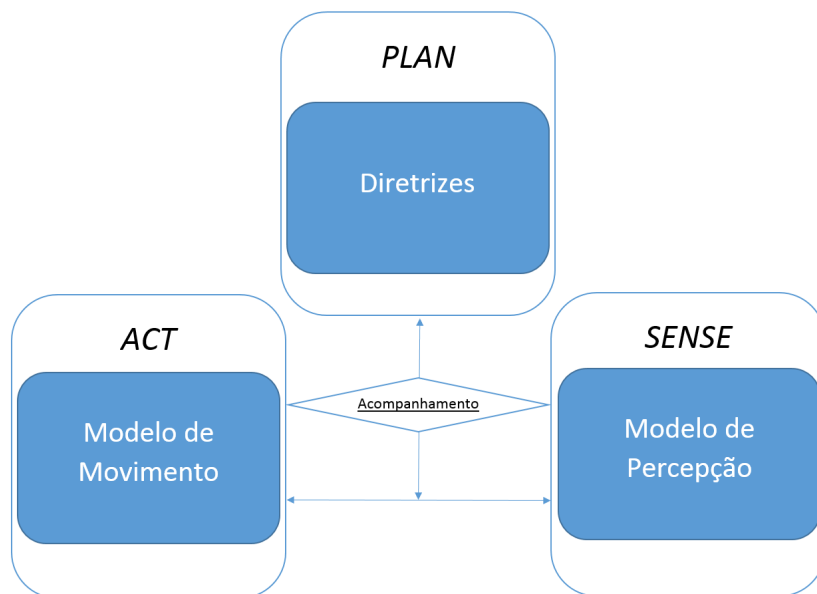


Figura 5.5: Etapas para obter primitivos.  
[Fonte: Elaborada pelo autor.]

O primitivo *ACT* é determinado com base no sistema de controle e atuação do veículo subaquático. É necessário conhecer o comportamento do veículo (que é obtido pela análise dinâmica do modelo físico) frente às forças e momentos provenientes do meio ou do próprio veículo. Após sua montagem é necessário programar um controlador que possibilite o comando dos atuadores para obter os resultados ditados pelas diretrizes do módulo planejamento ou para responder às reações informadas pelo módulo de percepção.

O primitivo *SENSE* é obtido por meio da escolha dos sensores que capturam o parâmetro físico medido. O dado captado é tratado de forma que o módulo libere a informação, que é alimentada para os outros módulos, resultando na atualização da crença do veículo. Esse dado é acompanhado por ruído, inerente à construção do sensor, que será também incorporado à atualização da crença. A escolha do sensor na tomada de informações e taxa de atualização são determinações das diretrizes do módulo planejamento, podendo variar em função do objetivo local e análise do meio.

O primitivo *PLAN* dependerá das entradas iniciais do operador, as quais serão a posição desejada, funções de custo de movimento, quais e como os sensores irão medir, entre outras possíveis configurações dadas antes de iniciar a execução. Utilizando essas entradas que estão descritas no item "Objetivos", o sistema embarcado efetua um planejamento de trajetória para encontrar as diretrizes de operação em cada célula na matriz de ocupação, que serão suavizadas para definir posições que o controlador deverá atingir. Caso a posição inicial tenha sido definida, o módulo de localização será executado para obter a posição com o mínimo de crença determinada pelo operador, com o intuito de completar as informações necessárias para o planejamento de trajetória e para o acompanhamento do modelo de movimento.

Baseado nessa estrutura, o texto a seguir irá descrever os métodos utilizados para obter o modelo de movimento, o modelo de percepção e as diretrizes aplicados ao ROV. Esses métodos poderão ser estendidos para determinação dos parâmetros de outros veículos, porém fazendo-se as adaptações necessárias a fim de obter, corretamente, as constantes ou variáveis independentes.

### **5.2.3 Etapas da Metodologia**

A figura 5.6 ilustra todas as etapas da metodologia implementada neste trabalho para obter a navegação autônoma. Com a organização dos primitivos, descritos anteriormente, é necessário determinar os três modelos principais: o modelo de movimento, o modelo de percepção e as diretrizes.

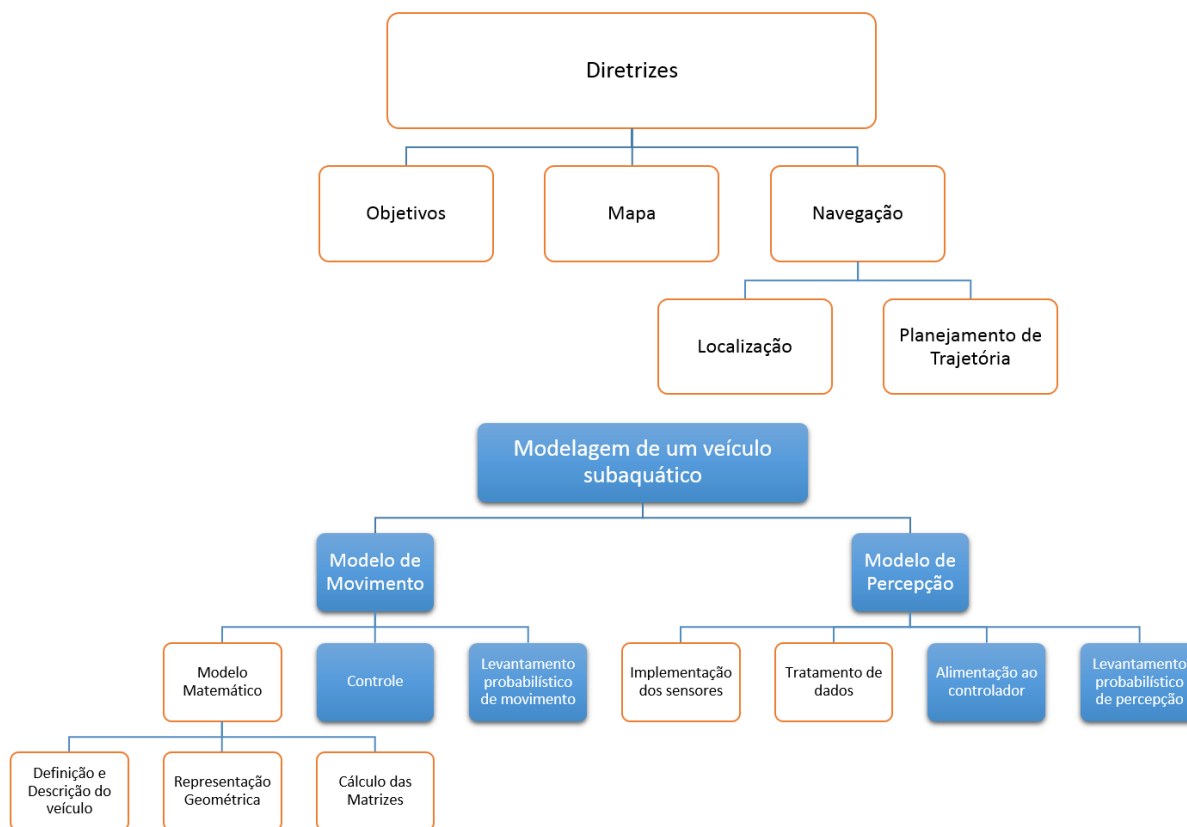


Figura 5.6: Etapas da metodologia implementada no trabalho.

[Fonte: Elaborada pelo autor.]

Na definição do grupo das diretrizes é necessário explicitar os objetivos da missão, o mapa do ambiente e o método de navegação. No grupo de navegação as ações de localização e planejamento de trajetória serão os principais agentes para orientar o veículo na navegação autônoma.

Os modelos de movimento e de percepção são subitens do processo de modelagem de um veículo subaquático. O modelo de movimento é subdividido em três áreas: o modelo matemático, baseado na teoria apresentada no capítulo 2; a área de controle e o levantamento probabilístico de movimento. O modelo de percepção tem a seguinte divisão: a implementação dos sensores embarcados; o tratamento de dados deles obtidos para extrair as informações do meio e do veículo; a alimentação ao controlador e o levantamento probabilístico de percepção, para determinar a confiabilidade dos dados e modelar os algoritmos probabilísticos.

Na figura 5.6 os itens em laranja foram complementados neste trabalho e serão descritos no decorrer deste capítulo e no capítulo 6. Os outros blocos, que estão em azul, são propostas de uma futura pesquisa e serão comentados no decorrer do texto.

## 5.3 MODELAGEM DE UM VEÍCULO SUBAQUÁTICO

### 5.3.1 Modelo de movimento

A compreensão da dinâmica de qualquer corpo físico dentro do meio é fundamental para preparar os controladores necessários para efetuar a locomoção. Principalmente no meio subaquático, as interações de um veículo com o meio fluido são demonstradas de uma maneira não intuitiva, tal fato resulta na necessidade de construir um modelo baseado nas leis da mecânica dos fluidos, com a finalidade de determinar as ações que serão executadas pelo sistema de comando do veículo, e assim obter o movimento desejado. Somente após compreender seu comportamento é possível implementar um controlador que consiga transformar variáveis de um planejamento prévio em ações de movimento para obter velocidades ou posições dentro do meio, podendo com isso navegar de forma precisa.

Assim, definido o equipamento que será usado para exemplificar o procedimento proposto para a navegação autônoma, determina-se o modelo matemático que representará o comportamento dinâmico do ROV no meio fluido. Deste modo, a partir do sistema de equações obtidas no capítulo 2 - eq. (2.59) - serão demonstradas as etapas necessárias para obtenção do modelo de movimento.

#### 5.3.1.1 Definição e descrição do ROV

No âmbito do presente trabalho, com a finalidade de exemplificar e validar o procedimento proposto, foi escolhido e montado um OCROV<sup>1</sup>, denominado "UnBot" (*Underwater roBot*). Sua geometria é baseada no veículo *OpenROV* [21], um projeto aberto de ROVs de observação, com popularidade crescente na comunidade acadêmica e de exploração oceanográfica. Há mais de 400 exemplares utilizados em várias universidades do mundo, sendo que todos os veículos montados possuem um número serial que é cadastrado no repositório de controle no sítio na internet da empresa [21].

Os fatores que determinaram a escolha deste veículo foram o seu baixo custo, facilidade de aquisição, montagem modularizada e com projeto aberto, e principalmente o fato de que o ROV não é provido de um sistema de navegação autônoma.

Dois modelos foram construídos para estudo: os veículos com números seriais 359, com geometria original do *OpenROV* versão 2.3, e 360, um modelo adaptado, com geometria modificada e motores alterados, ilustrados na fig. 5.7. Este último, cujas adaptações serão explicadas a seguir, será o equipamento utilizado como objeto de aplicação deste trabalho.

<sup>1</sup> *Observational Class ROV*, descrito na seção 1.2 no capítulo 1

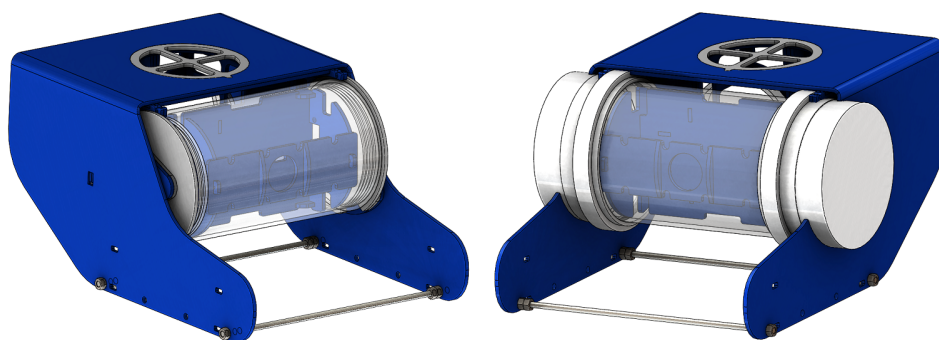


Figura 5.7: Modelo Original do *OpenROV* (à esquerda) e Modelo modificado e exemplo deste trabalho (à direita).

[Fonte: Elaborada pelo autor.]

As principais modificações realizadas foram a troca dos propulsores para um motor *brushless* mais eficiente, com menor KVA (menor rotação e maior torque, recomendadas para propulsão aquática), e um par de hélices com menor passo para propulsão de avanço e que anula o torque induzido (hélice com passo normal e outro invertido). Assim, perturbações no eixo de rotação teoricamente serão nulas em estado permanente com o veículo balanceado. Além disso, foi redimensionada a cápsula eletrônica prevendo futuramente a implementação de visão estéreo. Cabe observar que neste trabalho a visão é obtida por uma única câmera. Os dados técnicos dos componentes embarcados no ROV podem ser consultados no Anexo I.

As principais características do veículo estão listadas na tabela 5.1 abaixo:

Tabela 5.1: Principais características do ROV "UnBot"

Modelo	OpenROV versão 2.3 modificado
Tamanho	300 x 240 x 160mm (comprimento,largura,altura)
Peso	2.6 Kg
Profundidade máx.	20m
Propulsores	3 (2 horizontais e 1 vertical)
Sensores	Câmera <i>Rpi camera</i> e IMU <i>Razor 9DOF</i>
Alimentação	12 V

### 5.3.1.2 Modelagem geométrica

Para iniciar a modelagem é necessário possuir o modelo CAD do veículo em questão. Aplica-se os materiais corretos de construção em todos os componentes possíveis de serem modelados dentro do desenho, a fim de obter um modelo tridimensional do seu corpo rígido para apuração de resistência, inércia e objeto de simulação<sup>2</sup>. Recomenda-se ao projetista sempre aplicar o material mais próximo da peça que

<sup>2</sup> O ROV foi modelado dentro do pacote CAD SolidWorks 2013a, aplicando à carenagem e à cápsula de eletrônica o material acrílico de médio/alto impacto, às tampas do tubo o PVC Rígido, às barras de apoio de carga auxiliar e às porcas de fixação o

está modelando, para manter os dados alimentados no software CAD mais próximos ao que está sendo construído.

A modelagem 3D do ROV, feita com uso do programa CAD SolidWorks 2013a, consiste na sua representação gráfica e na especificação dos materiais de todos os componentes. A correção das informações é importante para a avaliação da resistência, verificação da inércia do sistema e simulação computacional. A especificação dos materiais está indicada na tabela 5.2.

Tabela 5.2: Materiais selecionados para modelagem dos componente do ROV

<b>Componente do ROV</b>	<b>Material (da lista do SolidWorks 2013a)</b>
Carrenagem	Acrílico de Médio/Alto Impacto
Cápsula Eletrônica	Acrílico de Médio/Alto Impacto
Tampas da Cápsula	PVC Rígido
Barras de apoio de carga auxiliar	Aço Inox Ferrítico
Porcas de fixação	Aço Inox Ferrítico
Carga Auxiliar	Chumbo
Propulsores	Alumínio 1060

As figuras 5.8 a 5.11 ilustram o modelo CAD e o veículo construído. Nota-se que o cabo de conexão lateral, assim como as lâmpadas frontais e conexão ao cabo umbilical não foram modelados. Estes componentes foram omitidos a fim de conservar a simetria do modelo. Alterações serão realizadas posteriormente com o intuito de retirar do veículo real tais itens da superfície externa, a fim de reduzir o arrasto induzido desses elementos no modelo real.

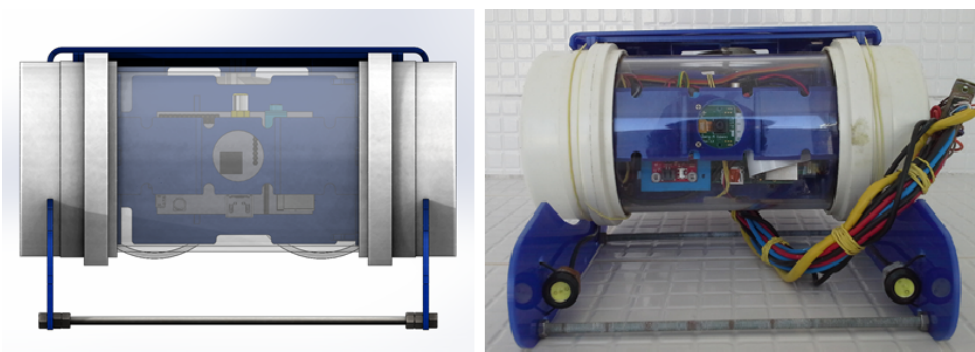


Figura 5.8: Vista Frontal do ROV

[Fonte: Elaborada pelo autor.]

aço inox ferrítico, à carga auxiliar o material chumbo e aos propulsores o alumínio 1060.

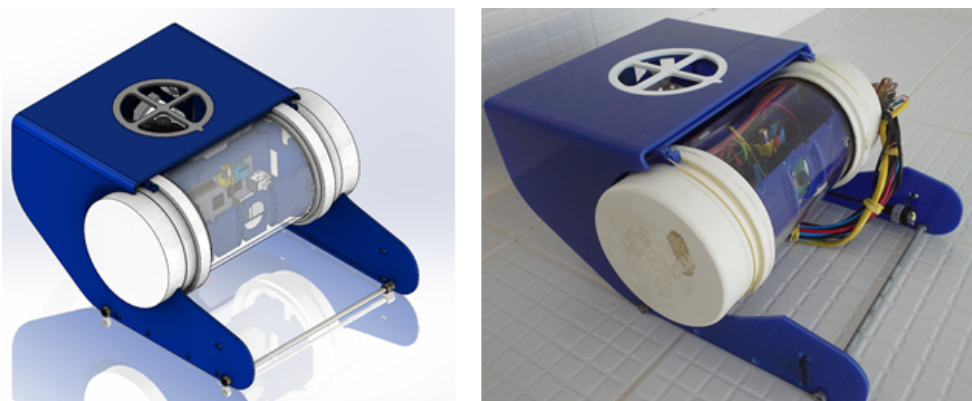


Figura 5.9: Vista Isométrica do ROV  
[Fonte: Elaborada pelo autor.]

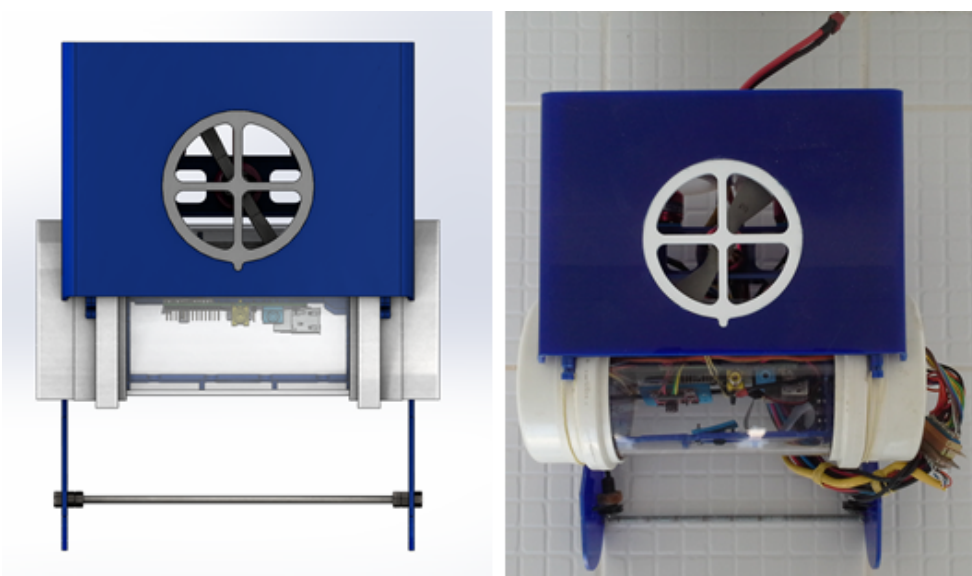


Figura 5.10: Vista de topo do ROV  
[Fonte: Elaborada pelo autor.]

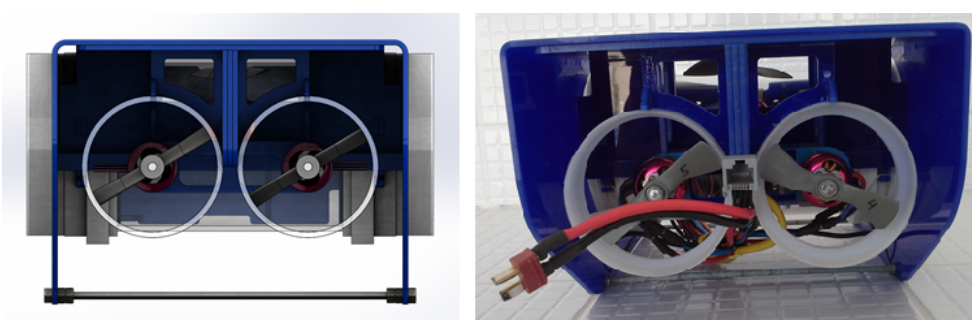


Figura 5.11: Vista Traseira do ROV  
[Fonte: Elaborada pelo autor.]

As principais características, tais quais, o sistema de coordenadas adotadas, o centro de gravidade, o



centro de empuxo, a localização da carga auxiliar (contrapesos) e dos propulsores estão ilustradas na figura 5.12. Como se pode notar, o veículo é simétrico no plano formado pelos eixos  $x, z$ , visando simplificar os efeitos de acoplamento hidrodinâmicos comentados na seção 2.5. A denominação do sistema de coordenadas utiliza a nomenclatura vigente para embarcações e veículos subaquáticos em português [22] [23].

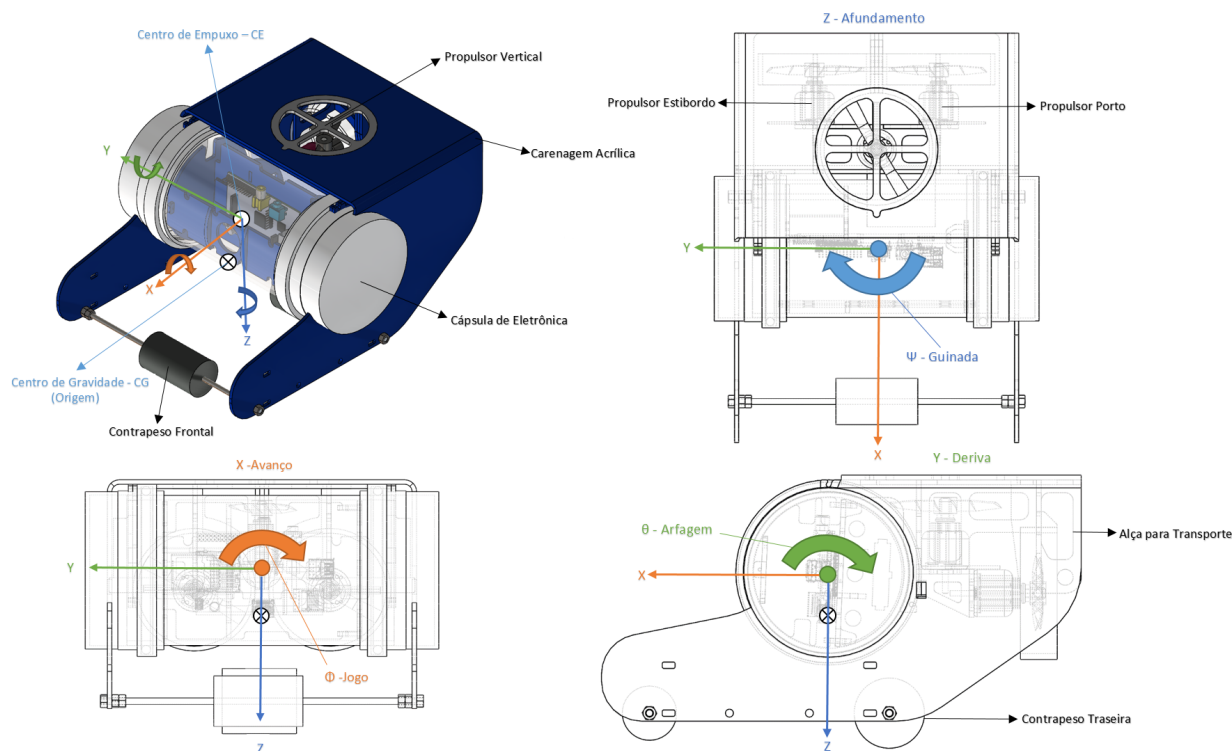


Figura 5.12: Sistema de coordenadas e localização dos componentes

[Fonte: Elaborada pelo autor.]

### 5.3.1.3 Matrizes de Massa e Inércia

Ao finalizar o desenho e as especificações dos materiais são obtidas as massas dos componentes com auxílio do programa CAD Solidworks 2013a e também os momentos de inércia para todos os eixos translacionais e rotacionais. Esses parâmetros são as variáveis  $[m, I_{xx}, I_{yy}, I_{zz}, I_{xy}, I_{xz}, I_{yz}]$  das matrizes de massa de corpo rígido  $M_{RB}$  e a matriz de Coriolis e de forças centrípetas de corpo rígido  $C_{RB}$ . A referência para extrair as propriedades foi fixada no centro de gravidade (CG) do ROV, pois como visto anteriormente, na seção 2.4.3 (Representação Matricial), ela simplifica o formato do tensor de massa de corpo rígido do veículo subaquático. Os dados retirados do modelo CAD referentes aos momentos de inércia e massa estão indicados na figura 5.13 e na tabela 5.3.

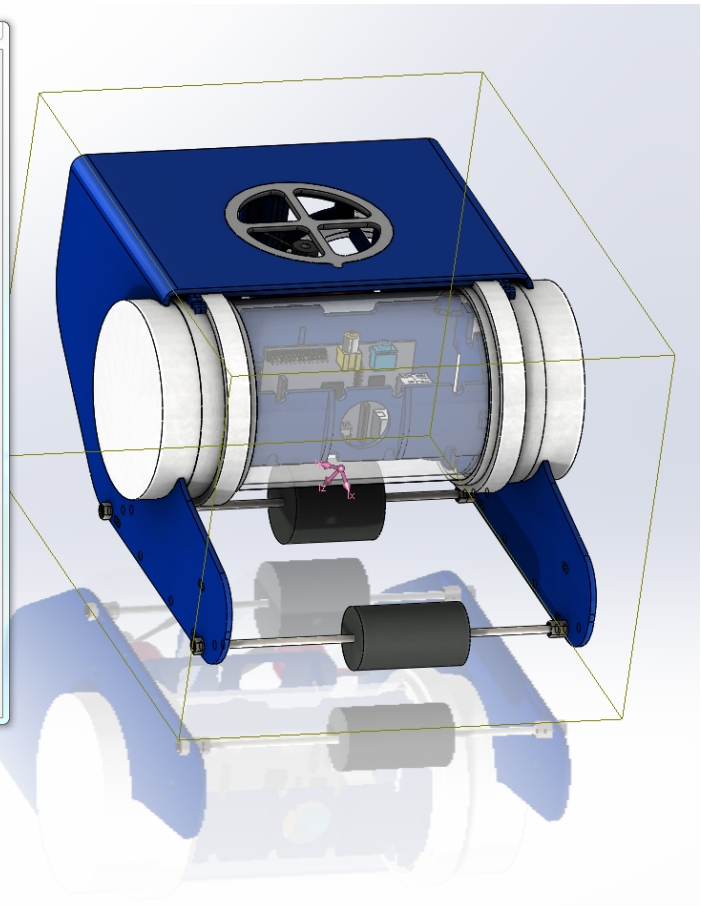
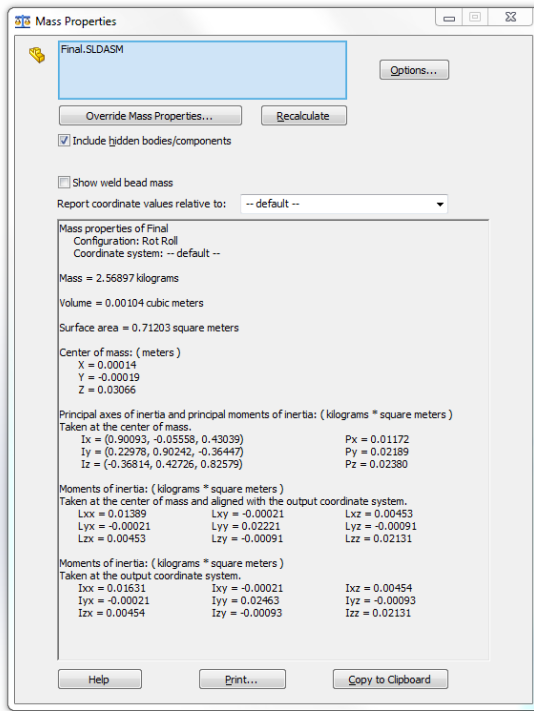


Figura 5.13: Dados de centro de massa e inércia CAD - Solidworks.  
[Fonte: Elaborada pelo autor.]

Tabela 5.3: Dados de Corpo Rígido

Variáveis	Valores	Unidade de Medida
Massa vazio $m_{vazio}$	1.9	$Kg$
Massa com peso $m$	2.6	$Kg$
$I_{xx}$	0.01631	$Kgm^2$
$I_{yy}$	0.02463	$Kgm^2$
$I_{zz}$	0.02131	$Kgm^2$
$I_{xy}$	-0.00021	$Kgm^2$
$I_{xz}$	0.00454	$Kgm^2$
$I_{yz}$	-0.00093	$Kgm^2$

Com esses parâmetros determinados e utilizando as equações eq. (2.25) e (2.29), procede-se à montagem das matrizes  $M_{RB}$  e  $C_{RB}$ :

$$M_{RB} = \begin{bmatrix} 2.6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2.6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01631 & 0.00021 & -0.00454 \\ 0 & 0 & 0 & 0.00021 & 0.02463 & 0.00093 \\ 0 & 0 & 0 & -0.00454 & 0.00093 & 0.02131 \end{bmatrix} \quad (5.1)$$

$$C_{12}(v) = \begin{bmatrix} 0 & 2.6w & -2.6v \\ -2.6w & 0 & 2.6u \\ 2.6v & -2.6u & 0 \end{bmatrix} \quad (5.2)$$

$$C_{22}(v) = \begin{bmatrix} 0 & 0.00093q - 0.00454p + 0.02131r & -0.00093r - 0.00021p - 0.02463q \\ -0.00093q + 0.00454p - 0.02131r & 0 & -0.00454r + 0.00021q + 0.01631p \\ 0.00093r + 0.00021p + 0.02463q & 0.00454r - 0.00021q - 0.01631p & 0 \end{bmatrix} \quad (5.3)$$

Sendo que as unidades dos parâmetros são para massa  $[kg]$  e para os momentos de inércia  $[kg \times m^2]$ . As posições do  $C_G = (0, 0, 0)$  e do centro de empuxo em  $C_F = (0, 0, -0.03)$ , que são dados em  $[m]$ . O  $C_F$  é alinhado com o centro da cápsula eletrônica.

Esta metodologia de determinação das variáveis de corpo rígido proporciona resultados mais próximos do caso real se comparada às aproximações de figuras geométricas com distribuições uniformes de massa que algumas bibliografias [24] [25] [26] adotam. Assim, esse método recomendado por [9] e [27] se mostra mais adequado para projetos de concepção de veículos subaquáticos.

#### 5.3.1.4 Matrizes de Massa adicionada e Matriz de Forças e Momentos Hidrodinâmicos

As definições das matrizes hidrodinâmicas de massa adicionada e de forças e momentos, por serem mais complexas, são obtidas por aproximações numéricas ou fórmulas semi-empíricas. Ambos os métodos se baseiam em análises hidrodinâmicas de formas geométricas conhecidas.

Neste trabalho, para determinação da matriz de massa adicionada, foi utilizado o programa AMass<sup>3</sup>, que usa objetos geométricos 3D definidos na sua biblioteca, orientados no espaço em relação a um referencial comum. Os resultados deste programa foram comparados com os resultados obtidos por Chin [9] com os programas WAMIT e MultiSurf, os dois citados por Fossen [11] como os softwares padrões usados na indústria para cálculo numérico das características hidrodinâmicas do veículo [15]. O programa AMass possui a vantagem de ser fácil de ser aplicado aos veículos representados a partir de geometrias simples e por terem sido validados com formas padrões que possuem solução analítica. Em seu trabalho, Chin [9] afirma que a comparação entre os resultados dos dois programas na geometria definida por seu veículo apresentou uma variação pequena.

<sup>3</sup> Utilizado pelos alunos da matéria 2.20 *Marine Hydrodynamics* do *Massachusetts Institute of Technology* (MIT) [28], lecionada no outono de 2013. Esse software está disponível na referência, e há uma cópia da versão 1.4 no Anexo CD juntamente com manual de instalação do Java3D para Windows 7 e um manual de utilização fornecida pelo curso do MIT.

Pressupõe-se que a geometria do ROV, objeto deste trabalho, possa ser substituída por geometrias existentes no programa AMass como, por exemplo, cilindros circulares (representando o tubo de eletrônica, tampas laterais e propulsores) e cilindros elipsoidais com pequena espessura representando as placas planas. Essas geometrias possuem soluções analíticas tridimensionais presentes no relatório de Brennen [8], no qual são apresentadas tabelas de soluções para escoamentos 2D e 3D (livre e com fronteiras).

Primeiramente, demonstra-se a validade de cálculo para o cilindro circular. De acordo com o relatório de Brennen (pg. 5) [8], para um escoamento potencial ideal, um cilindro de raio  $R$  está se movendo com velocidade variando pelo tempo  $U(t)$  na direção positiva de  $x$ . Utilizando coordenadas polares  $(r, \theta)$ , pode-se obter a relação entre o movimento translacional com raio e ângulo como sendo  $x = r \cos(\theta)$ .

As velocidades resultantes do fluido em coordenadas polares são dadas pela velocidade potencial  $\phi$  como sendo:

$$u_r = \frac{\delta \phi}{\delta r}; u_\theta = \frac{1}{r} \frac{\delta \phi}{\delta \theta} \quad (5.4)$$

Para o caso do cilindro:

$$\phi_{cilindro} = \frac{-UR^2}{r} \cos(\theta) \quad (5.5)$$

Essa solução é encontrada com as seguintes considerações:

- Equação de Laplace,  $\nabla^2 \phi = 0$  para as coordenadas cilíndricas;
- A condição de contorno da velocidade relativa normal à superfície do corpo é zero,  $(u_r)_{r=R} = U \cos(\theta)$ .

Pode-se afirmar para esses escoamentos que  $u_i$  é proporcional à velocidade do escoamento no infinito  $U$ , e, assim, a integral  $I$  pode ser calculada como:

$$Cilindro : I = \int_R^\infty \int_0^{2\pi} \left[ \left( \frac{1}{U} \frac{\delta \phi}{\delta r} \right)^2 + \left( \frac{1}{Ur} \frac{\delta \phi}{\delta \theta} \right)^2 \right] r \delta \phi \delta r = \pi R^2 \text{ por unidade de medida.} \quad (5.6)$$

Esse resultado é para um cilindro infinito (ou que pode ser aproximado, tendo uma razão de aspecto alto). Para o caso do ROV, tendo comprimento delimitado, é necessário multiplicar o comprimento do cilindro por  $\pi R^2$ .

Encontrado  $I$ , a massa adicionada do cilindro é expressa como:

$$M = \rho I = \rho \pi R^2, \text{ por unidade de comprimento} \quad (5.7)$$

Para um cilindro de raio  $1m$  e comprimento  $8m$ , a massa adicionada pelo método analítico será de 25.1327 para avanço (similarmente para deriva). Para o topo e fundo do cilindro é necessário determinar a

massa adicionada de uma placa circular. No relatório do Brennen [8] é exposto que a expressão da energia cinética da placa é:

$$T = \frac{4}{3}\rho\pi r^3 U^2 \text{ com } k = \frac{2}{\pi} \quad (5.8)$$

A constante  $k$  expressa uma relação entre o aumento aparente de massa do corpo em movimento com a massa de fluido deslocado. Assim, a constante  $k$  relaciona a massa adicionada à energia cinética como sendo:

$$k = \frac{\text{aumento aparente em massa}}{\text{massa do fluido deslocado}} = \frac{2T}{M_a U^2} \quad (5.9)$$

Isolando a massa adicionada obtém-se:

$$M_a = \frac{4}{3}\pi\rho r^3 \quad (5.10)$$

Resolvendo os valores de  $r = 1m$ , encontra-se o valor de 4.188, aproximadamente.

No caso de um cilindro elipsoidal o cálculo é o mesmo, porém há uma variação de raio ao se mudar o eixo de análise. Então para um cilindro elipsoidal com os raios  $a = 1m$  e  $b = 0.5m$ , com comprimento  $8m$ , os valores seriam de 6.2832 para o eixo de raio  $b$  e 25.1327 para o eixo de raio  $a$ . Para o topo e fundo, a expressão de energia cinética é [8]:

$$T = \frac{2\pi}{3E}\rho k'' a^2 b U^2 \text{ com } k'' = 0.826 \text{ para } \frac{a}{b} = 2 \quad (5.11)$$

Novamente, utiliza-se a relação entre o aumento aparente de massa do corpo em movimento com a massa de fluido deslocado  $k$ :

$$k = \frac{1}{E} = \frac{\text{aumento aparente em massa}}{\text{massa do fluido deslocado}} = \frac{2T}{M_a U^2} \quad (5.12)$$

$E$  é a integral elíptica da placa elipsoidal. Sua fórmula está presente no relatório de Brennen [8], porém não é necessária sua determinação, pois ela será anulada pelo desenvolvimento da equação anterior, proporcionando a equação da massa adicionada:

$$M_a = \frac{4}{3}\rho\pi a^2 b k'' \quad (5.13)$$

Com  $a = 1$ ,  $b = 0.5$  e  $k'' = 0.826$ , obtém-se o valor de 1.729 para a massa adicionada com escoamento na direção  $y$  da placa elipsoidal plana.

Utilizando o software AMass obtém-se as matrizes de massa adicionada para o cilindro circular e para o cilindro elipsoidal, com parâmetros anteriormente descritos. Os valores encontrados são iguais aos analiticamente determinados, tendo divergência somente no cilindro elipsoidal na placa plana, mas o erro é menor que 3%. Assim, é demonstrado que o programa pode ser usado para determinar as matrizes de massa

adicionada para cilindros circulares e cilindros elipsoidais. A figura 5.14 ilustra as geometrias comparadas e o resultado da matriz de massa adicionada do programa AMass.

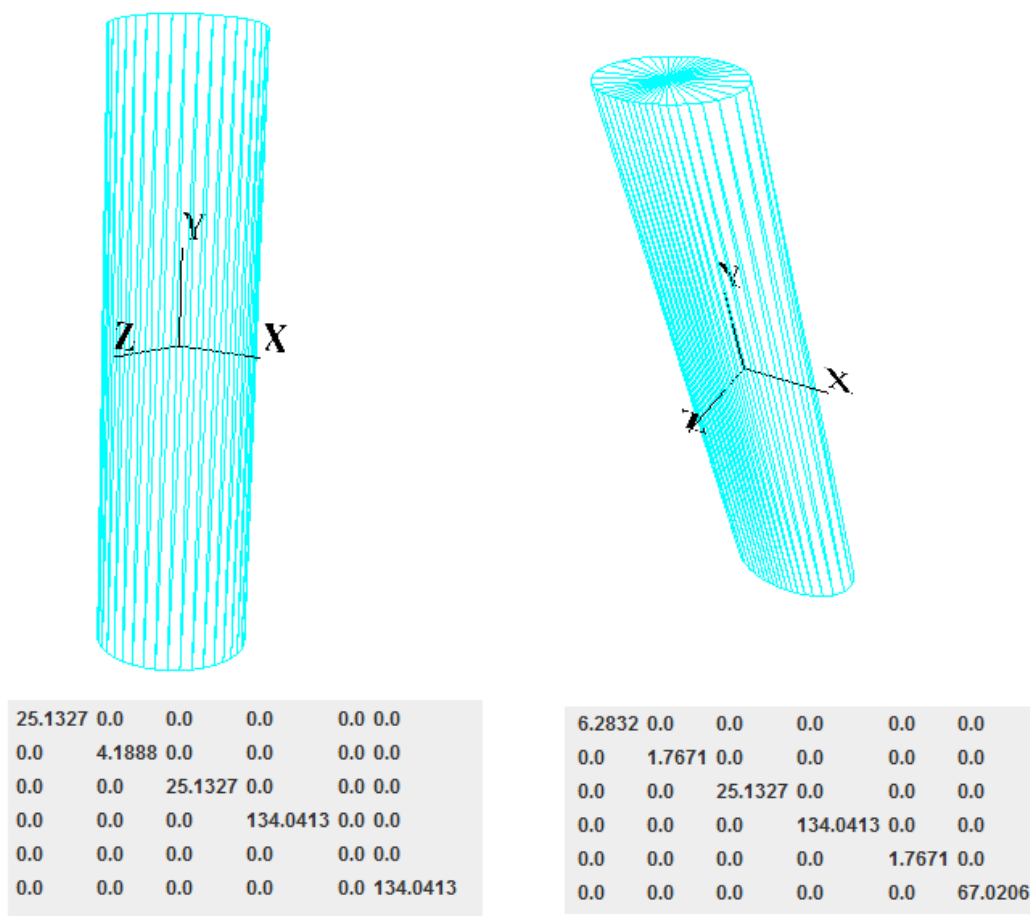


Figura 5.14: Massa adicionada do cilindro (à esquerda) e do cilindro elipsoidal (à direita) no AMass.  
[Fonte: Elaborada pelo autor.]

A determinação da matriz de massa adicionada pode então ser realizada com o uso do programa AMass. As partes que podem ser representadas por cilindros de seção circulares são: a cápsula eletrônica, as tampas laterais do tubo e os propulsores. Já as placas planas podem ser aproximadas por cilindros de seção elipsoidal.

É necessário definir uma origem para orientar o posicionamento de cada geometria. O recomendado é o centro de gravidade (CG) do ROV, o qual é utilizado como referência para determinação das matrizes da equação diferencial [11], mas como ainda não foi determinado o peso adicional para anular o efeito de empuxo do tubo de eletrônica, utiliza-se como referência o centro de empuxo (CE), que se localiza no centro do tubo de eletrônica do ROV (devido a sua simetria). Posteriormente essa referência será corrigida para o centro de gravidade (CG) a fim de manter as simplificações da matriz  $M_{RB}$ .

Nas tabelas 5.4 e 5.5, os dados de entrada são apresentados para representar o ROV no programa AMass. Os valores do comprimento, raio e espessura são retirados das plantas do ROV que se encontram no Anexo I.2.

Tabela 5.4: Dados de Entrada para Cilindros Circulares

Entradas	Tubo Eletrônico	CapE	CapD	AnelD
Raio	0.050	0.053	0.053	0.058
Top center(x,y,z)	(0, -0.119, 0)	(0, -0.119, 0)	(0, 0.066, 0)	(0, -0.084, 0)
Bottom center(x,y,z)	(0, 0.119, 0)	(0, -0.066, 0)	(0, 0.119, 0)	(0, -0.071, 0)

Entradas	AnelE	Motor1	Motor2	Motor3
Raio	0.058	0.0315	0.0315	0.0315
Top center(x,y,z)	(0, 0.084, 0)	(-0.143, -0.046, 0)	(-0.143, 0.046, 0)	(-0.0715, 0, -0.06)
Bottom center(x,y,z)	(0, 0.071, 0)	(-0.120, -0.046, 0)	(-0.120, 0.046, 0)	(-0.0715, 0, -0.037)

Tabela 5.5: Dados de Entrada para Cilindros Elipsoidais

Entradas	Placa Topo	Lateral 1D	Lateral 1E
Bottom center (x,y,z)	(-0.075, -0.096, -0.06)	(-0.075, -0.096, -0.06)	(-0.075, 0.096, -0.06)
End Point of 1st axis (x,y,z)	(0, n, -0.06)*	(0, -0.096, n)*	(0, 0.096, n)*
Input top second axis length	0.0015	0.0015	0.0015
Base center (x,y,z)	(-0.075, 0.096, -0.06)	(-0.075, -0.096, 0.09)	(-0.075, 0.096, 0.09)
Input base 1st axis length	0.075	0.075	0.075
Input base 2nd axis length	0.0015	0.0015	0.0015

Entradas	Lateral 2D	Lateral 2E	Tampa Traseira
Bottom center (x,y,z)	(-0.150, -0.096, 0.048)	(-0.150, 0.096, 0.048)	(-0.058, -0.096, 0)
End Point of 1st axis (x,y,z)	(n, -0.096, 0.09)*	(n, 0.096, 0.09)*	(-0.058, n, -0.06)*
Input top second axis length	0.0015	0.0015	0.0015
Base center (x,y,z)	(0.150, -0.096, 0.048)	(0.150, 0.096, 0.048)	(-0.058, 0.096, 0)
Input base 1st axis length	0.042	0.042	0.06
Input base 2nd axis length	0.0015	0.0015	0.0015

\* Nota: Nos cilindros Elipsoidais, onde aparece na tabela a coordenada "n", significa que essa entrada está bloqueada do programa, não necessitando seu valor para determinação do parâmetro pedido.

Com esses valores obtém-se a geometria mostrada na figura 5.15:

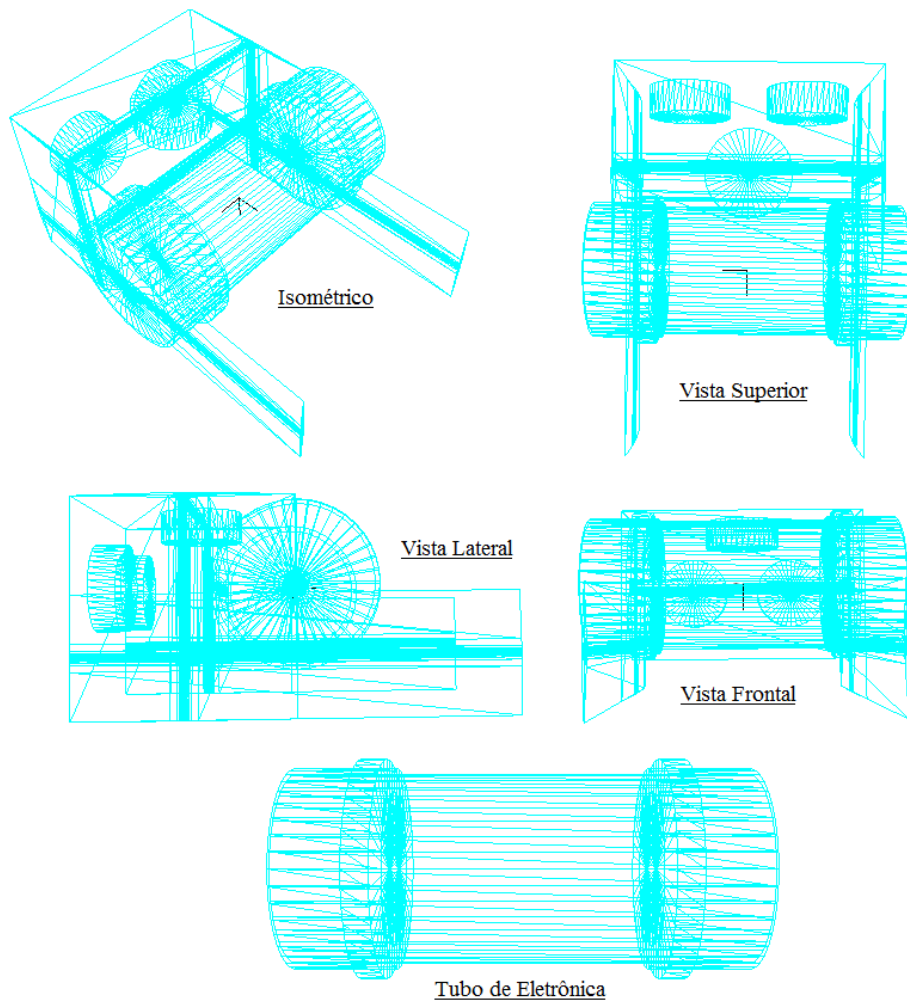


Figura 5.15: Geometria do ROV no AMass.  
 [Fonte: Elaborada pelo autor.]

O tensor massa adicionada em referência ao centro de empuxo (CE) encontrado é:

$$M_A(CE) = \begin{bmatrix} 0.0057 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.0126 & 0 & -2E^{-4} & 0 & -4E^{-4} \\ 0 & 0 & 0.0072 & 0 & 3E^{-4} & 0 \\ 0 & -2E^{-4} & 0 & 1E^{-4} & 0 & 0 \\ 0 & 0 & 3E^{-4} & 0 & 0 & 0 \\ 0 & -4E^{-4} & 0 & 0 & 0 & 1E^{-4} \end{bmatrix} \quad (5.14)$$

Esse formato condiz com a configuração de simetria dos planos X-Z (bombordo e boreste) e X-Y (topo e inferior). A área das placas laterais é muito pequena comparada ao restante do ROV, o que aproxima um resultado de simetria no plano X-Y.

Agora, é necessário mudar a origem para o centro de gravidade (CG) do ROV, com o objetivo de manter as simplificações feitas nas matrizes de corpo rígido [9]. Altera-se no software AMass a origem para as



coordenadas (0,0,-0.03) - posição obtida pelo software CAD SolidWorks 2013a - determinando-se o tensor final de massa adicionada:

$$M_A = \begin{bmatrix} 0.0057 & 0 & 0 & 0 & -0.0002 & 0 \\ 0 & 0.0126 & 0 & 0.0002 & 0 & -0.0004 \\ 0 & 0 & 0.0072 & 0 & 0.0003 & 0 \\ 0 & 0.0002 & 0 & 0.0001 & 0 & 0 \\ -0.0002 & 0 & 0.0003 & 0 & 0 & 0 \\ 0 & -0.0004 & 0 & 0 & 0 & 0.0001 \end{bmatrix} \quad (5.15)$$

Ao determinar os elementos da matriz adicionada é possível calcular o tensor de forças e momentos hidrodinâmicos. Utilizando as equações 2.37 e 2.38, obtém-se a seguinte matriz:

$$C_A = \begin{bmatrix} 0 & 0 & 0 & 0 & -0.072w - 0.0003q & 0.0002p - 0.0004r + 0.0126v \\ 0 & 0 & 0 & 0.072w + 0.0003q & 0 & -0.0057u + 0.0002q \\ 0 & 0 & 0 & -0.0002p + 0.0004r - 0.0126v & 0.0057u - 0.0002q & 0 \\ 0 & -0.072w - 0.0003q & 0.0002p - 0.0004r + 0.0126v & 0 & -0.0001r + 0.0004v & 0.0003w - 0.0002u \\ 0.072w + 0.0003q & 0 & -0.0057u + 0.0002q & 0.0001r - 0.0004v & 0 & -0.0001p - 0.0002v \\ -0.0002p + 0.0004r - 0.0126v & 0.0057u - 0.0002q & 0 & -0.0003w + 0.0002u & 0.001p + 0.0002v & 0 \end{bmatrix} \quad (5.16)$$

Como comentado anteriormente, o AMass 3D é uma alternativa livre para o WAMIT, que é o programa mais usado comercialmente para caracterização hidrodinâmica. Quando comparados, o AMass apresenta um resultado próximo ao do WAMIT, que por sua vez é capaz de representar a dinâmica real do veículo [9].

### 5.3.1.5 Matrizes de Amortecimento Hidrodinâmico

O passo seguinte será a determinação da matriz de amortecimento hidrodinâmico. A obtenção desta matriz depende da resolução da equação de arrasto hidrodinâmico no corpo para cada eixo de translação e rotação. Em determinações analíticas, o projetista aproxima a forma do veículo a uma forma geométrica que possui solução conhecida para todos os eixos. No entanto, na maioria das vezes, não modela o comportamento real do veículo, introduzindo erros ao modelo cada vez que a velocidade aumenta.

Um método proposto por [9] [27] recomenda a utilização de CFD *Computacional Fluid Dynamic* para encontrar as variáveis desejadas. Esse método necessita de um planejamento maior na malha e determinação do domínio para convergência do fluxo no veículo, porém apresenta resultados mais concisos em veículos com geometrias complexas, como acontece com a maioria dos ROVs atuais em operação.

Como mencionado na seção 2.5.2 (Amortecimento), o amortecimento hidrodinâmico em veículos subaquáticos consiste em ambos os termos lineares e quadráticos [9]. Pode ser obtido da soma das contribuições do amortecimento potencial devido ao arrasto linear de atrito (das superfícies) e do amortecimento viscoso devido ao arrasto de forma. Este último contribui para os termos não lineares.

Como o ROV foi projetado para ser estável em arfagem e rolagem, a magnitude desses ângulos deve ser muito pequena. Por ter um pequeno ângulo de ataque em relação ao escoamento, pode-se considerar que não haverá uma separação da camada limite.

Assim, os termos não lineares das forças e momentos da matriz de amortecimento, devido ao efeito viscoso do escoamento, aproximam-se de zero quando o ângulo de ataque é pequeno [9].

Por outro lado, as forças oriundas do escoamento potencial têm uma relação linear com o ângulo de arfagem (o ângulo de ataque específico na direção de deriva). Então, é considerado que os coeficientes lineares são suficientes para representar as forças e momentos originados da parte invíscida do escoamento, especialmente para baixas velocidades (menores que 2 m/s) e com pequena variação do ângulo de ataque durante o movimento. Para veículos aquáticos este termo linear sempre estará presente devido ao arrasto por atrito linear. Porém é importante notar que é necessário inserir o amortecimento linear e dinâmico no modelo do ROV, pois somente o termo quadrático irá proporcionar um movimento oscilatório em baixa velocidade. O termo linear é necessário para que o amortecimento convirja exponencialmente ao repouso [11]. Essa aproximação é válida para o ROV analisado, pois o empuxo propulsor máximo em avanço será menor que o arrasto de 2m/s, limitando as manobras feitas dentro da faixa de manutenção do estado (parâmetro linear > parâmetro quadrático), como ilustrado na figura 5.16 (adaptado de T.I.Fossen [11]).

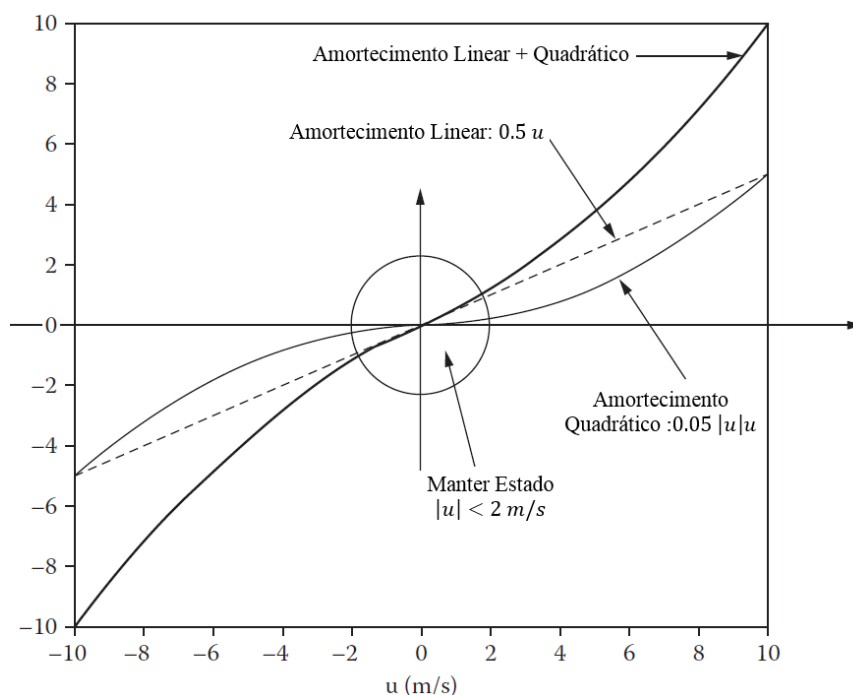


Figura 5.16: Amortecimento hidrodinâmico em baixas e altas velocidades do veículo subaquático.

[Fonte: Elaborada pelo autor.]

Sob o ponto de vista de análise hidrodinâmica, o ROV é um sistema complexo com vários componentes que interagem com o meio fluido. Dr. Chin [9] sugere utilizar uma metodologia semi-preditiva para determinar os coeficientes de amortecimento hidrodinâmico, utilizando o método de escoamento em tanque aberto e um programa de CFD *Computational Fluid Dynamics*, denominado Solidworks Flow 2013, o qual foi utilizado neste projeto.

Para obter a matriz de amortecimento hidrodinâmico é necessário estabelecer certas condições sobre o escoamento.

- O ROV irá locomover-se entre 0 a 0.5 m/s para translação (rotação em yaw em 0.5 rad/s), limitando-se à faixa de amortecimento linear, ilustrada no círculo da figura 5.16. Assim, o termo linear será o mais importante na modelagem, mas para deixar o modelo com dados sobre o decaimento oscilatório, os termos quadráticos serão também determinados para cálculo;
- O fluido utilizado será a água doce, à 25 °C modelado como incompressível e em escoamento permanente, sendo necessário fazer um novo cálculo ou adaptação caso o veículo navegue em água salgada;
- O comprimento do cubo que delimita o domínio da análise foi definido em 5 vezes o comprimento do ROV (como ilustrado na figura 5.17, por considerar-se desprezíveis efeitos fluidos com o veículo além desse limite);
- Efeitos de transferência de calor e dilatação térmica foram desconsiderados na simulação CFD.

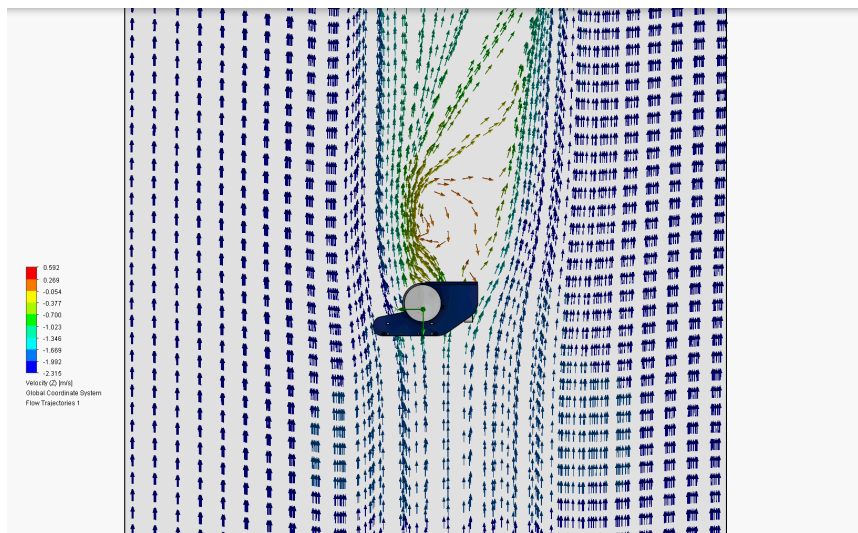


Figura 5.17: Escoamento CFD em afundamento realizado no SolidWorks Flow 2013 com domínio exposto  
[Fonte: Elaborada pelo autor.]

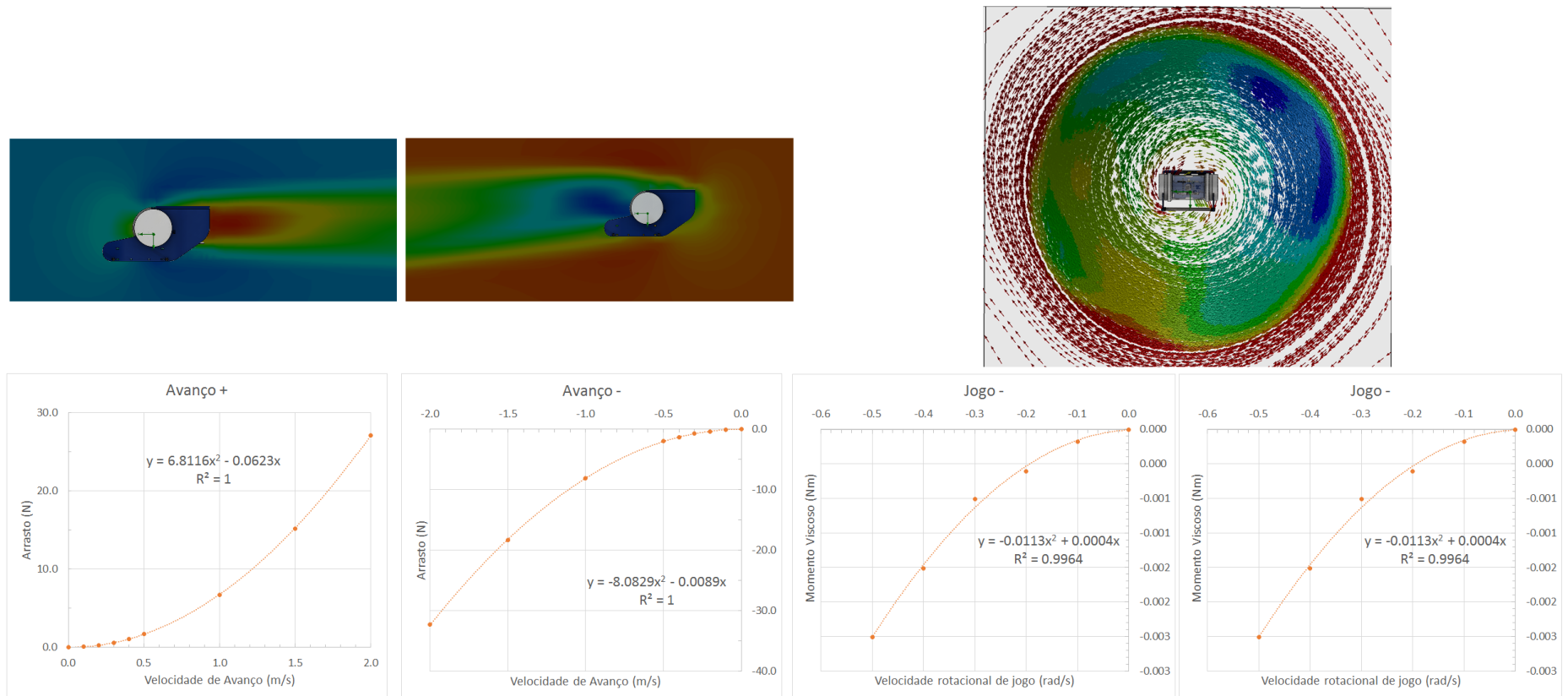


Figura 5.18: Gráficos do Amortecimento Hidrodinâmico em avanço e jogo positivo e negativo.

[Fonte: Elaborada pelo autor.]

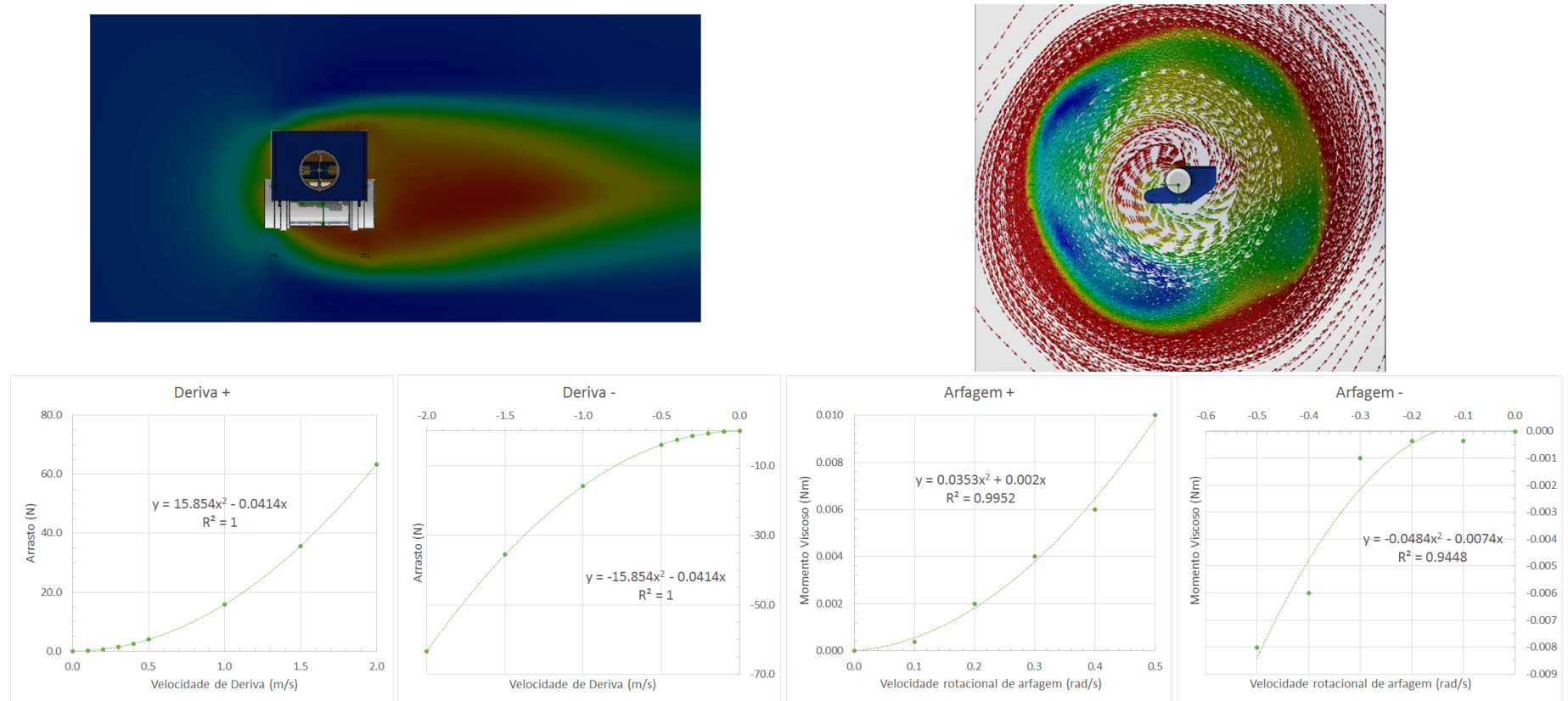


Figura 5.19: Gráficos do Amortecimento Hidrodinâmico em deriva e arfagem positiva e negativa.

[Fonte: Elaborada pelo autor.]

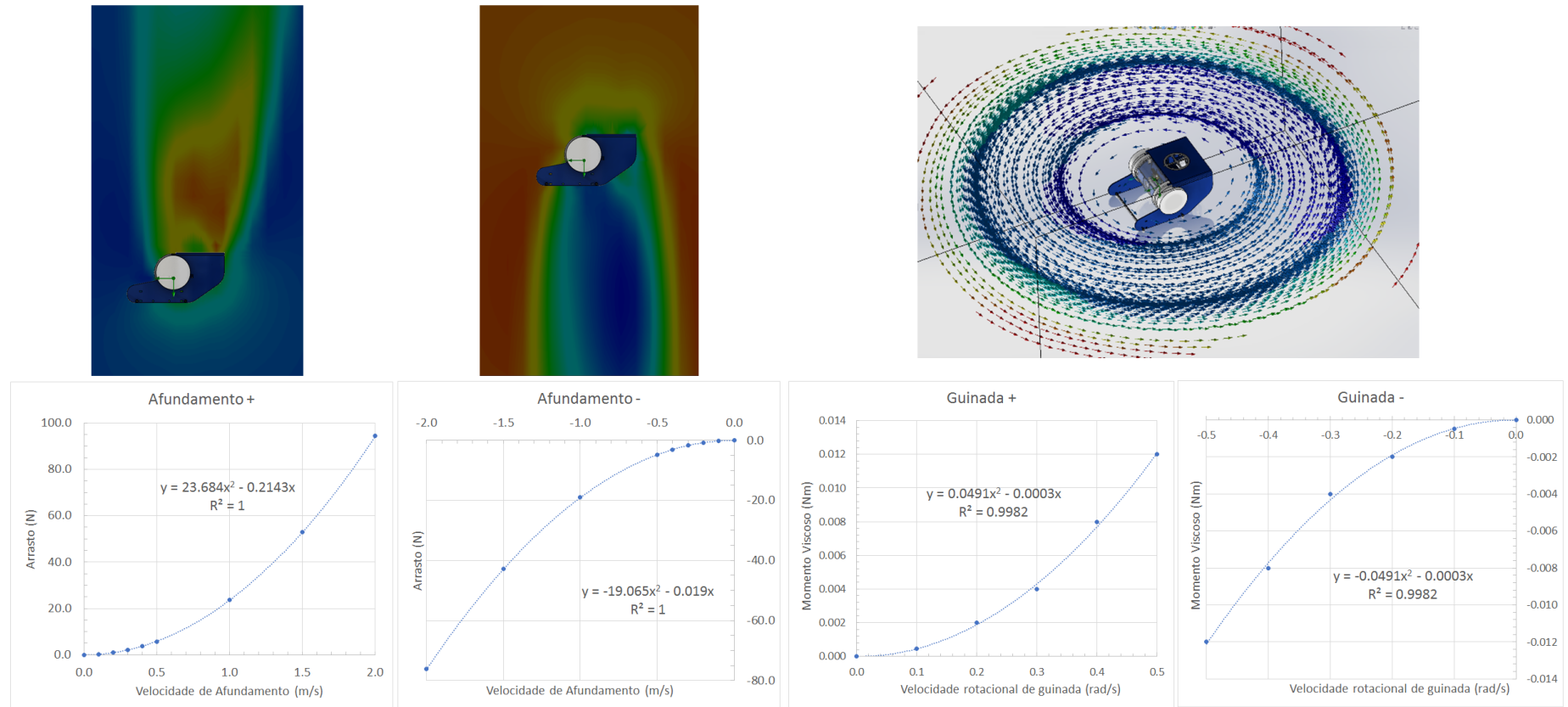


Figura 5.20: Gráficos do Amortecimento Hidrodinâmico em afundamento e guinada positiva e negativa.

[Fonte: Elaborada pelo autor.]

As figuras 5.18, 5.19 e 5.20 ilustram o resultado da simulação de escoamento feita em CFD para todos os eixos translacionais e rotacionais do veículo. Para cada velocidade foi determinada a força de arrasto hidrodinâmico exercida no veículo presente nos gráficos apresentados nas figuras. Foi calculada a regressão polinomial de segunda ordem para cada movimento e encontrado o seu fator de determinação com o intuito de expor a importância da variável velocidade na função força ou momento de arrasto. É possível verificar a simetria axial e rotacional do veículo pelos dados encontrados, sendo simétricos os valores para velocidades negativas. Somente para avanço, afundamento e arfagem que os valores são diferentes. Esse fato requer a construção de uma segunda matriz de amortecimento hidrodinâmico para situações que o veículo irá locomover-se em velocidades negativas nestes eixos. Os valores para as variáveis lineares e quadráticas são apresentados abaixo na tabela 5.6, caracterizando as duas tabelas de amortecimento hidrodinâmico do ROV "UnBot".

Tabela 5.6: Dados dos termos da Massa Adicionada

$X_u$	-0.0623	$X_{u u }$	6.8116	$X_u$	-0.0089	$X_{u u }$	-8.0829
$Y_v$	-0.0414	$Y_{v v }$	15.854	$Y_v$	-0.0414	$Y_{v v }$	-15.854
$Z_w$	-0.2143	$Z_{w w }$	23.684	$Z_w$	-0.019	$Z_{w w }$	-19.065
$K_p$	-0.0019	$K_{p p }$	0.0191	$K_p$	-0.0019	$K_{p p }$	-0.0191
$M_q$	0.002	$M_{q q }$	0.0353	$M_q$	-0.074	$M_{q q }$	-0.0484
$N_r$	-0.0003	$N_{r r }$	0.0491	$N_r$	-0.0003	$N_{r r }$	-0.0491

### 5.3.1.6 Matriz de Força de restabelecimento

No projeto do ROV foi escolhida a configuração do veículo no estado de empuxo neutro (empuxo compensa o peso) ou levemente positivo (empuxo > peso), pois de acordo com [6] é interessante em operações de ROVs deixar o veículo levemente positivo para garantir seu retorno à superfície caso algum problema de comunicação ou suprimento energético ocorra. Tal diferencial de forças para OCROVs pode chegar até 0,5 kgf, dependendo do tamanho do veículo e do controle para o operador. Também é recomendável essa configuração para navegação próxima ao terreno, pois ao acionar a propulsão do motor para controle de profundidade, o fluxo será direcionado para cima, evitando qualquer contato que a água tenha em dispersar possíveis sedimentos acumulados no fundo do terreno, comprometendo a visibilidade para observação e navegação.

Para AUVs, no entanto, é interessante a configuração neutra, pois a prioridade para tais veículos normalmente se aplica ao consumo de energia, sendo aconselhável otimizar a utilização dos propulsores para somente alterar trajetórias definidas, tendo o equilíbrio mantido por configuração mecânica.

Alguns projetos de veículos subaquáticos modernos utilizam sistemas variáveis de estados de empuxo [6], onde há atuadores internos para preencher espaços dentro de tubos acoplados na estrutura, a fim de mudar a quantidade de massa e de empuxo no próprio veículo, assim, construindo uma nova configuração adaptativa para cada ambiente sem precisar voltar a superfície para retirar ou adicionar pesos.

Para o trabalho em questão, cujo objetivo é a navegação autônoma, foi escolhida a configuração neutra

priorizando a economia de energia quando se configura neste modo. Porém também é possível ser exigida uma condição levemente positiva a fim de que o veículo sempre retorne à superfície caso haja pane elétrico. Isso altera a matriz de força de restabelecimento na equação (5.17), necessitando de um controlador do propulsor de afundamento para manter a posição no eixo  $z$  constante.

O vetor de forças de restabelecimento  $g(\eta_2)$  pode ser calculado usando a equação (2.58), por possuir empuxo neutro, ou seja  $W = B$ , e por ser simétrico no projeto, coincidindo as coordenadas X e Y do centro de empuxo e centro de gravidade  $x_{CM} - x_{CF} = 0$  e  $y_{CM} - y_{CF} = 0$ , como anteriormente apresentado na figura 5.12.

Os valores das variáveis necessárias para cálculo estão na tabela 5.7, sendo determinados pelo sistema de análise de massa do SolidWorks, e posteriormente apurados com medida em balança e análise qualitativa na água. Assim, a configuração final para empuxo neutro é:

Tabela 5.7: Dados de força de restabelecimento

Variáveis	Valores	Unidade de Medida
Peso $W$	26	$N$
Empuxo $B$	26	$N$
$z_{CM} - z_{CF}$	-0.03	$m$

$$g(\eta_2) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ (z_{CM}W - z_{CF}B)\cos(\theta)\sin(\phi) \\ (z_{CM}W - z_{CF}B)\sin(\theta) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ (0.78)\cos(\theta)\sin(\phi) \\ (0.78)\sin(\theta) \\ 0 \end{bmatrix} \quad (5.17)$$

Como explicitado na equação (5.17), as forças de restabelecimento só atuam nos eixos rotacionais de jogo  $p$  e arfagem/rolagem  $r$ . Nestes eixos, a estabilidade é retomada após perturbação pelo diferencial entre o centro de gravidade e o centro de empuxo. Dessa forma, não é necessária a atuação para restabelecer o estado de origem. Nos outros eixos é indispensável a atuação dos propulsores para restabelecer equilíbrio, o que configura um corpo de 4 GDLs.

No caso do ROV, objeto deste trabalho, as coordenadas  $x$  e  $y$  são relacionadas com o ângulo de guinada  $y$ , diminuindo para 3 o número de graus de liberdade a ser considerado.

### 5.3.1.7 Resumo dos Dados de Modelagem

Pode-se assumir as seguintes hipóteses para simplificar a equação geral de dinâmica do ROV na modelagem [9] [11]:

1. O referencial da Terra (NED) é inercial.



2. O fluido de análise é água doce, à 25 °C com os parâmetros informados no arquivo de configuração do SolidWorks 2013a.
3. O ROV é um corpo rígido e está completamente submerso na água.
4. Objetos externos na superfície do ROV (cabo de conexão, iluminação externa) não são modelados, a fim de manter simetria.
5. Despreza-se efeitos de interação fluida entre o veículo e os de superfícies de objetos presentes no meio, limitando-se à análise dentro das fronteiras do domínio.
6. Perturbações causadas por ondas são desprezíveis, pois o veículo está completamente submerso.
7. Na determinação da massa adicionada, o fluido ao redor é tratado como incompressível e a geometria do veículo é simplificada para aproximações numéricas de geometrias cilíndricas e de cilindros elipsoidais.
8. O ROV se locomove em baixas velocidades, aproximando o efeito de amortecimento hidrodinâmico em uma regressão quadrática [8].
9. Para a determinação do amortecimento viscoso, a água é tratada como fluido incompressível e calculada pela aproximação numérica da equação de Navier-Stokes descrita em [29].
10. Os efeitos do cabo umbilical do veículo não são considerados.
11. A diferença de pressão estática da superfície superior do veículo e da parte inferior é desprezível.

É importante possuir conhecimento de quais simplificações foram feitas na montagem do modelo para compreender possíveis causas de divergência entre o comportamento do modelo e o veículo real.

Todos os parâmetros e matrizes determinados para o modelo não linear do ROV estão expostos no Anexo I.2.

### **5.3.2 Controle**

A razão pela qual é importante definir adequadamente os parâmetros do modelo é avaliar seu comportamento frente a uma perturbação e compreender como irá se comportar dinamicamente. Somente com esse conhecimento é possível construir um controlador capaz de comandar os propulsores para movimentar o veículo da forma desejada. É necessário um controlador capaz de representar o comportamento dinâmico não-linear do modelo frente às perturbações externas. A robustez do controlador é desejada, porém é modelada em função das incertezas presentes tanto no modelo quanto nos sensores utilizados para realimentar a malha fechada. Por esta razão, a utilização de filtros probabilísticos poderá auxiliar o tratamento da leitura dos sensores na alimentação do controlador.

Há uma variedade de projetos de controladores presentes na bibliografia veículos subaquáticos, tais como: Controladores PID, controladores adaptativos [30], controladores em cascata [9], controladores de lógica fuzzy [31], controladores de rede neurais [32] entre outros. Nesta dissertação não foi elaborado o

projeto de controlador do veículo "UnBot". O foco foi determinar o modelo físico e preparar os dados em um bloco característico para permitir testes de comportamento por controladores diferentes. Há trabalhos como o [31] em que é preferível linearizar o modelo para integrar um controlador linear robusto, porém há outros como [9] em que se prefere utilizar técnicas de controle não-linear para comandar a ação dos atuadores no veículo a fim de proporcionar a resposta desejada. A escolha de qual linha seguir dependerá do projetista do controlador, e o autor desta dissertação deixa esse tema a ser explorado como trabalho futuro.

### 5.3.3 Levantamentos probabilísticos de movimento

Após obter um controlador que consiga comandar os propulsores de forma desejada é necessário apurar seu funcionamento fora da simulação em ensaios. O funcionamento previsto nele durante as simulações pode não ser o mesmo por simplificações feitas no modelo, suposições previstas durante o projeto do controlador ou efeitos externos não modelados. Assim, é necessário obter um levantamento probabilístico da crença de movimento do veículo para estimar a probabilidade de o veículo realmente encontrar-se na posição que se espera. O erro estacionário é depurado a cada iteração na malha fechada do esquemático de controle, porém, em aplicações reais, esse erro não será nulo nem constante pela dinâmica não linear do modelo e nem pelo ruído inerte dos sensores.

Recomenda-se à alternativa do método clássico - de diminuir o erro ao máximo - o método de estimar o erro em uma distribuição probabilística. Por essa alternativa, a imprecisão dos controladores não impediria sua possível aplicação, pois o erro será integrado como parâmetro de incerteza na função *previsão* do modelo de movimento. O foco será determinar um controlador que seja robusto, mas não necessite ser preciso.

O processo de levantamento probabilístico de movimento produz melhores resultados em ensaios aplicados ao veículo real durante os ajustes do controlador obtido.

## 5.4 MODELO DE PERCEPÇÃO

O modelo de percepção é caracterizado pela transmissão das informações após a medição de parâmetros físicos por sensores e pelo tratamento para isolar e inferir o dado desejado. Para tal, é necessário possuir conhecimento de como os sensores funcionam, seus limites de medição e confiabilidade frente a dinâmica do meio ou do próprio veículo. Esse modelo depende dos sensores embarcados no veículo e pode ser alterado para comportar outros sensores que inferirem dados, dependendo da aplicação. O ROV "UnBot" possui basicamente dois sensores embarcados: uma câmera e um IMU (*Inertial Measurement Unit*). Ambos são interligados e processados por uma placa de processamento embarcado de arquitetura ARM denominado *Raspberry Pi* [33].

A câmera embarcada é uma *Raspberry Pi Camera Board*, também conhecida como *Rpi Camera*, com as especificações descritas no Anexo I.1, que é ligada diretamente no conector CSI da GPU do Raspberry Pi, fato que diminui a carga computacional na CPU, permitindo captura de imagens de melhor qualidade

em uma taxa de resolução maior que câmeras USB ou câmeras analógicas com conversor A-D (Analogico-Digital). A captura de imagem digital em alta definição auxilia na navegação teleoperada por permitir uma imagem com menos distorção e cores reais, e na navegação autônoma por ser diretamente digital, fato que diminui o ruído acrescentado na conversão A-D e auxilia o tratamento de imagens por algoritmos de visão computacional. Outro interesse na escolha dessa câmera é seu pequeno tamanho e menor consumo de energia comparado ao projeto original do *OpenROV* que utiliza uma câmera USB. Esta apresentou um consumo elétrico maior, que gera mais calor dentro do tubo eletrônico, fato que poderia prejudicar não só o processamento da placa embarcada, mas também a vedação do tubo. Por esses motivos, houve a escolha do *Rpi Camera* como sensor principal no veículo "UnBot".

O sensor IMU consiste em um conjunto de sensores inerciais combinados em uma placa que transmite informações sobre a mudança de estado do objeto fixo à ela. O IMU utilizado no "UnBot" é o sensor IMU Razor 9DOF (descrição técnica presente no Anexo I) que une um acelerômetro triaxial, com giroscópio triaxial e um magnetômetro triaxial. Eles são combinados em um algoritmo de fusão embarcado na placa e fornecem os seguintes dados com medição direta: aceleração nos três eixos, velocidades rotacionais imediatas e a orientação em arfagem, rolagem e jogo. Velocidades lineares e posições podem ser inferidas por integração dos dados em um sistema separado, porém esse método possui um erro polinomial variante a cada integração, proporcionando cálculos imprecisos para aplicações reais. Atualmente, ele tem a funcionalidade de informar a orientação do veículo no meio, a fim de corrigir erros na estimativa provenientes dos filtros probabilísticos.

Similarmente ao modelo de movimento, há necessidade de conhecer a crença das informações transmitidas por estes sensores, por meio da determinação de uma distribuição probabilística de percepção. Após a calibração é necessário que cada sensor seja ensaiado no veículo para averiguar seu dado e modelar a incerteza que ele fornece à função *atualização*.

## 5.5 DIRETRIZES

O módulo das Diretrizes, baseado na primitiva PLAN, é composto por três elementos: Objetivos, Planejamento de Trajetória e Localização.

### 5.5.1 Objetivo

O objetivo do veículo na navegação autônoma se resume ao posicionamento e orientação final da locomoção. Este dado é informado pelo operador, que poderá definir como a trajetória será formada pelo algoritmo de planejamento ao determinar as diretrizes de movimento. As informações iniciais referentes ao objetivo são carregadas antes de qualquer ação no meio, para que todo o módulo de planejamento possa funcionar sem conflito. Modificações durante o movimento poderão ser feitas para que o módulo altere a rota de navegação para o novo objetivo informado.

Neste trabalho, as informações necessárias são:

- Posição inicial \*

- Posição final
- Mapa
- Função de custo

Estes dados compõem o mínimo a ser informado para que o veículo possa navegar em um ambiente modelado em grade de ocupação. A única informação que poderá ser inferida é a posição inicial por meio de algoritmos de localização do terreno, que serão explicados na seção 5.6. Informações adicionais poderão modificar a trajetória, baseadas nos parâmetros de custo para a convergência dos dados para a rota otimizada.

### 5.5.2 Planejamento de Trajetória

Como apresentado no 4, há dois principais algoritmos de planejamento de trajetória, diferenciando-se principalmente no custo computacional. O *A-star* produz a rota otimizada de trajetória sem distúrbios, possuindo menor custo em cálculo para mapas de grandes dimensões. Porém, ele não consegue prever como retornar à rota otimizada em situações nas quais o veículo sofreu desvio de rota, casos em que o algoritmo *Dynamic Programming* consegue lidar. Assim, a escolha do algoritmo de planejamento é em função da análise prévia do operador. Caso ele determine que a dinâmica do meio poderá exercer uma grande influência, de forma que o controlador não consiga interpor sem mudança de posição, o algoritmo *Dynamic Programming* é o mais recomendado.

Para montagem e visualização dos algoritmos em funcionamento foi criado um programa escrito na linguagem *Processing* (baseado no Java) de definição dos objetivos para planejamento de trajetória. Este programa apresenta uma interface de visualização e montagem do mapa que o veículo utilizará para navegar.

Primeiramente, determina-se a posição inicial do veículo (representado pela figura do "UnBot"), a posição do alvo (célula em vermelho) e as células bloqueadas com obstáculos para navegação (células com quadrado preto), conforme mostrado na figura 5.21.

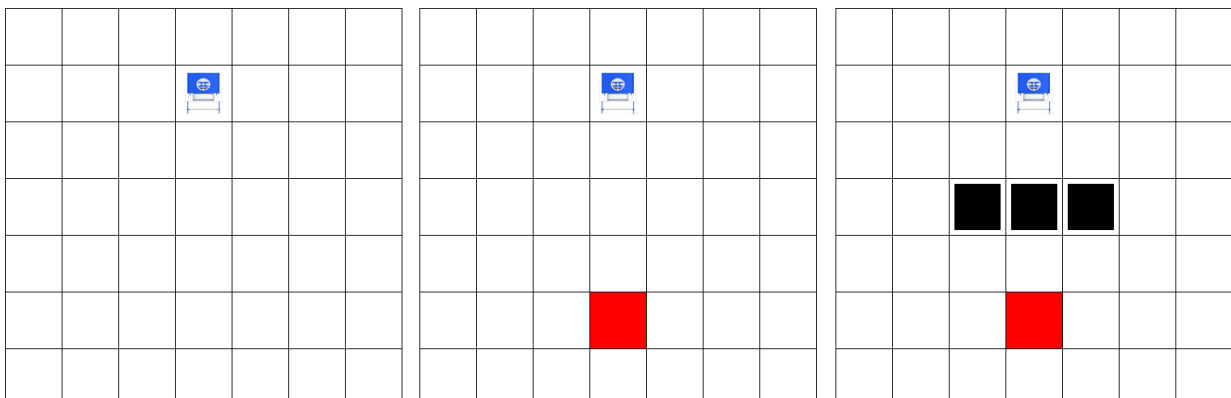


Figura 5.21: Programa *Mission Planner* - Etapas para planejamento de trajetória

[Fonte: Elaborada pelo autor.]

Após definir os parâmetros necessários para planejamento de trajetória, salva-se o arquivo de configurações e executam-se os algoritmos de determinação de diretrizes: o *A-star* ou *Dynamic Programming*. Ambos os algoritmos estão implementados em *Python*, que é a linguagem utilizada para confeccionar os programas do conjunto *Anav* no Raspberry Pi. O resultado, após o cálculo, é exposto na fig. 5.22:

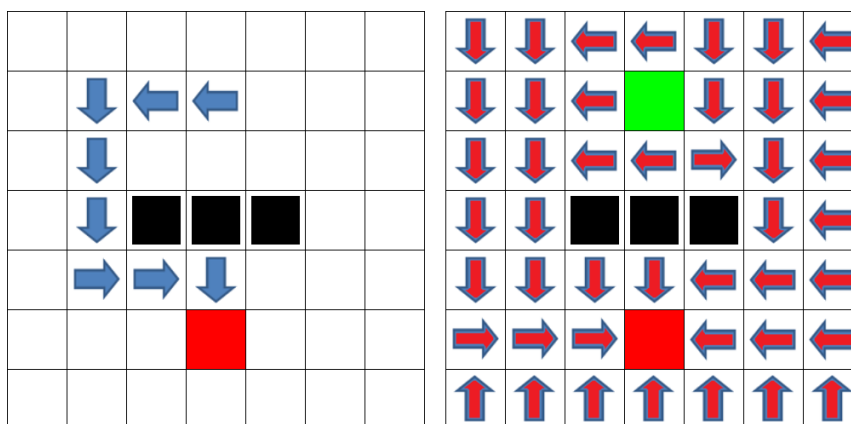


Figura 5.22: Programa *Mission Planner* - Resultados do *A-star* (à esquerda) e *Dynamic Programming* (à direita)

[Fonte: Elaborada pelo autor.]

Essas diretrizes estão presentes em dois arquivos .csv para apuração pelo visualizador *Mission Planner* e como entrada para o modelo de movimento. Cada célula possui um caracter que define a ação desejada nas coordenadas cartesianas na grade de ocupação. Com essas variáveis o controlador poderá determinar quais serão os comandos necessários para completar a diretriz proposta e, assim, obter a posição desejada.

### 5.5.3 Localização no Terreno

A localização é somente chamada quando há necessidade de obter a posição atual do veículo, seja para obter uma estimativa da posição inicial para planejamento de trajetória, seja para acompanhar o veículo durante o percurso. O filtro escolhido foi o filtro de partículas, apresentado no capítulo 3, pelas razões de ser multimodal e possibilitar controle de custo computacional, uma característica que afeta imensamente o desempenho em sistemas embarcados.

O módulo de localização no terreno é integrado no programa *Anav*, presente no próprio servidor embarcado no veículo.

## 5.6 SISTEMA IMPLEMENTADO

A figura 5.23 ilustra o esquemático de funcionamento do sistema de navegação autônoma no veículo "UnBot", baseado no paradigma hierárquico e na metodologia proposta.

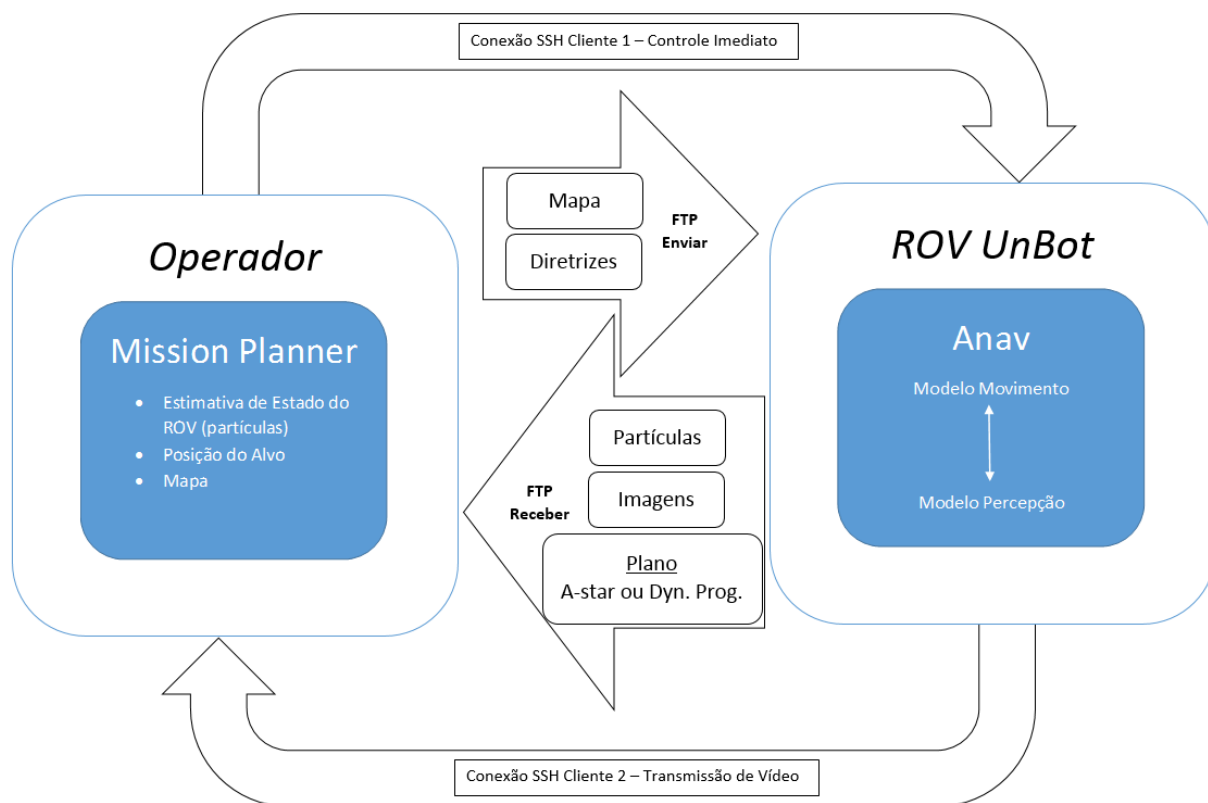


Figura 5.23: Sistema de navegação autônoma aplicado no ROV "UnBot".

[Fonte: Elaborada pelo autor.]

O esquemático ilustra todos os componentes, conexões e dados que fluem dentro do sistema, desde o operador até o ROV "UnBot".

No bloco do operador, o programa *Mission Planner* é o meio de confecção do planejamento de trajetória e visualização do estado atual do veículo no terreno. Como descrito anteriormente, o *Mission Planner* faz o mapa e as diretrizes que o robô deverá executar, transmitindo-as por FTP (*File Transfer Protocol*) com os comandos e credenciais registrados no próprio programa por arquivo *.bat* (no Windows).

No bloco do ROV "UnBot", o programa *Anav* recebe as informações passadas, faz seu planejamento e executa seus primitivos *ACT* e *SENSE* (baseados no paradigma híbrido) em loop até atingir objetivo local ou interrupção pelo usuário. Dados são gerados e enviados via FTP para alimentar o programa *Mission Planner*, baseado no primitivo *PLAN* para acompanhamento do veículo durante a navegação. Esses dados podem ser partículas, leitura de sensores ou planos de rota, seja o algoritmo *Dynamic Programming* para terrenos incertos e estocásticos, seja o *A-star* para terrenos com poucas perturbações. As partículas são atualizadas em um arquivo pelo filtro de partículas implementado na placa embarcada do ROV para estimar a localização. Neste trabalho a localização é passiva enquanto as variâncias dos eixos  $x$  e  $y$  não ultrapassarem o limite estabelecido pelo operador, pois, caso isso ocorra, o algoritmo de localização passa a ser ativo e define as diretrizes locais para reencontrar o veículo no terreno.

Há também duas comunicações externas via SSH entre os blocos operador e ROV. A primeira conexão é um SSH de Controle imediato, que permite que o operador retome o controle do veículo do programa

*Anav* interrompendo a navegação autônoma. É considerada uma ferramenta útil em casos nos quais o programa está apresentando um resultado divergente do esperado ou em casos em que o próprio operador quer navegar o veículo, não sendo possível essa função em AUVs.

A segunda conexão SSH é para a transmissão de vídeo em tempo real para acompanhamento da navegação. A placa Raspberry Pi, em conjunto com a *Rpi Camera*, possibilita geração de vídeo digital em alta definição, que é retransmitido pelo mesmo cabo de rede utilizado pelos programas *Anav*, *Mission Planner* e SSH Cliente 1. O vídeo é capturado por um programa denominado *NetCat* no computador do usuário, que sequencializa as imagens recebidas e guarda em um *cache* para reprodução em um programa que possibilita execução de stream, como VLC ou MPlayer. Pode-se diminuir o delay do vídeo alterando-se a memória de cache, permitindo transmissão de vídeo digital em alta definição, em tempo real, juntamente com a execução dos outros programas de comando. Essa conexão SSH é vantajosa comparada aos sistemas existentes no mercado, que utilizam captura de vídeo analógico com conversão A-D e transmissão em cabo separado do cabo de comando, exemplo do VideoRAY PRO IV [34]. Essa conexão SSH permite a transmissão de vídeo digital em alta definição sem adicionar ruído, ação esta que a transmissão analógica não possibilita e prejudica a visualização do terreno subaquático pelo operador. A transmissão de vídeo é somente limitada pela distância máxima de comunicação do cabo de rede, e mantém sua qualidade dentro deste limite (em torno de 100m para CAT6), oposto ao cabo coaxial que aumenta a resistência quanto maior o comprimento e, com isso, diminui a qualidade da imagem. Outra vantagem é a transmissão simultânea no mesmo cabo, que possibilita diminuir a espessura do cabo umbilical e, assim, diminuir sua interferência no veículo.

Com essas conexões e programas implementados em cada bloco foi possível obter um sistema de navegação autônoma para o ROV "UnBot". Este esquema poderá ser aplicado em outros veículos subaquáticos alterando-se as configurações dos modelos movimento e percepção, baseadas no veículo em uso e nos sensores que possui.

## 6 RESULTADOS EXPERIMENTAIS

### 6.1 INTRODUÇÃO

A seguinte seção descreve um ensaio realizado com o veículo subaquático "UnBot" com objetivo de depurar o funcionamento dos programas Mission Planner e Anav no operador e no ROV, respectivamente, e verificar se o veículo possibilita a identificação de *landmarks* no meio para resolver o problema de localização global.

### 6.2 CENÁRIO

O terreno de ensaio proposto é uma piscina, de profundidade 1.30m, comprimento 7.50m, e largura 4.5m. Dentro da piscina, existem três pratos circulares brancos de diâmetro 10cm alinhados e com posições conhecidas, conforme ilustra a figura 6.1. Esses pratos serão os *landmarks* para auxiliar a localização no terreno.

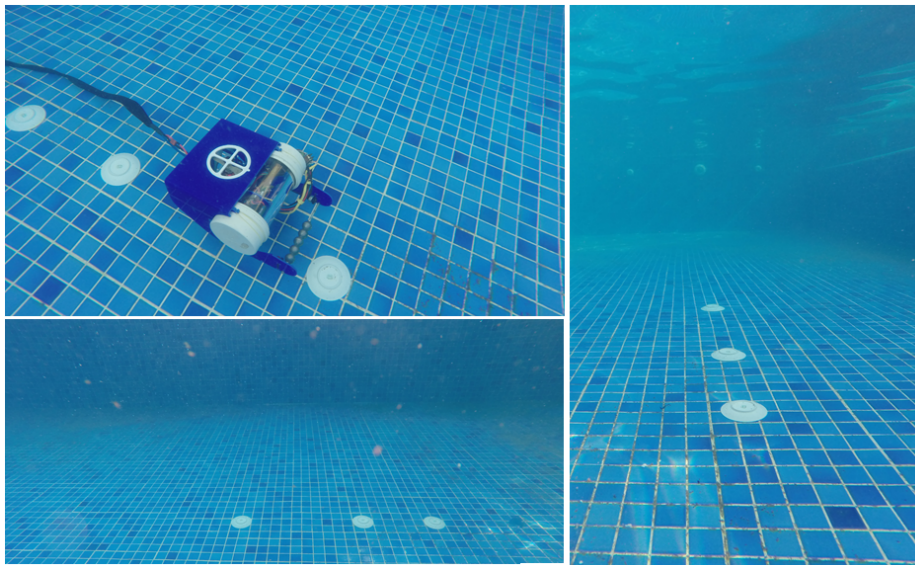


Figura 6.1: Cenário do Ensaio.

[Fonte: Elaborada pelo autor.]

Inicialmente, faz-se uso do programa *Mission Planner* presente no computador do operador para criar o mapa do terreno a fim de preparar as diretrizes do ROV. Utilizando as medidas previamente informadas, o mapa foi modelado como uma grade de ocupação de 25 células x 15 células, sendo cada célula um quadrado de 30 cm de aresta. Esse mapa é apenas bidimensional, pelo fato de o veículo não possuir um sensor de profundidade que impede um controle preciso no eixo z. O mapa está ilustrado na figura 6.2:



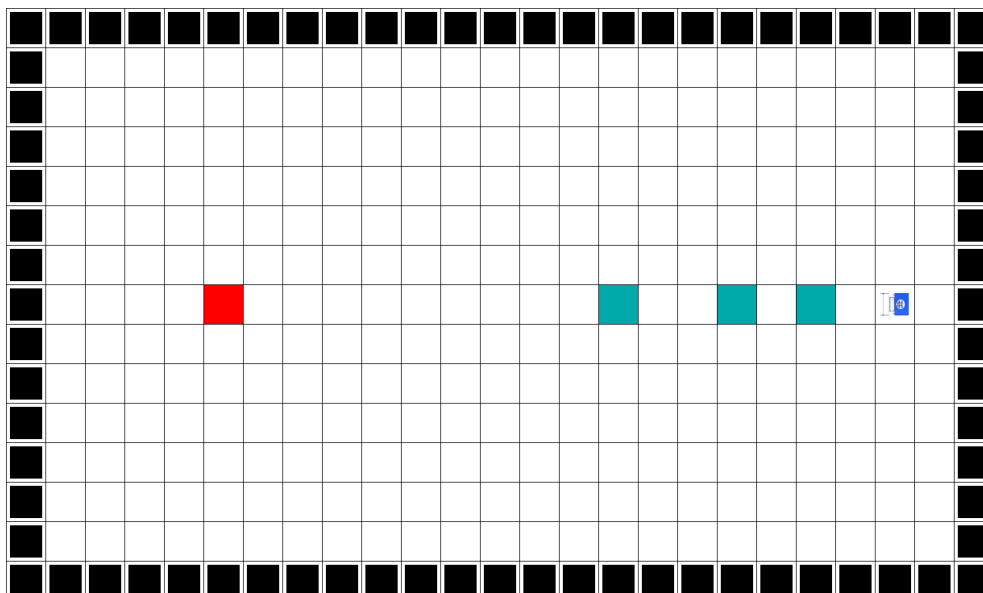


Figura 6.2: Mapa montado pelo programa *Mission Planner*.

[Fonte: Elaborada pelo autor.]

O ROV será inserido na piscina e iniciará sua rotina de localização procurando informações para se orientar no meio. Nesse caso, serão os *landmarks* que ajudarão a determinar sua posição. O sensor utilizado será o *Rpi Camera*, que captura imagens do chão enquanto navega e processa de forma a identificar se há um *landmark* abaixo dele ou não.

Como neste trabalho não foi desenvolvido um controlador para comandar o movimento do veículo, um operador foi posicionado para orientar a navegação do ROV durante a trajetória. O operador utilizará as pastilhas do chão da piscina - via transmissão de vídeo pelo cliente SSH 2 - em conjunto com a leitura do IMU para posicionar o veículo na posição e orientação estimada pelo modelo de movimento. Para existir esta comunicação e alimentação de energia foi conectado um cabo umbilical com a superfície, ilustrado na figura 6.3.

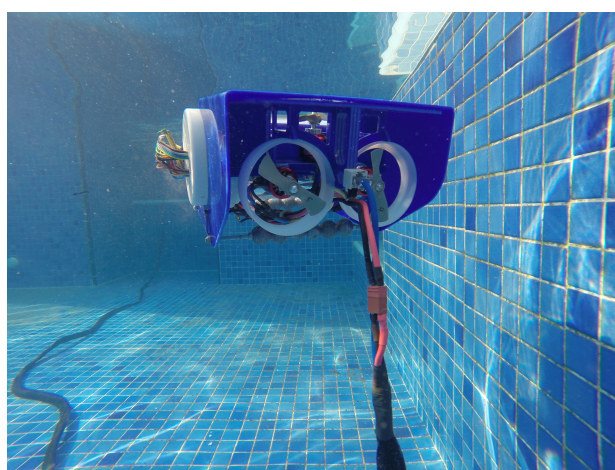


Figura 6.3: Foto do Cabo umbilical do ROV.

[Fonte: Elaborada pelo autor.]

### 6.3 FLUXOGRAMA DO ENSAIO

O seguinte fluxograma de ações foi desenvolvido para resolver o problema de localização do veículo no meio:

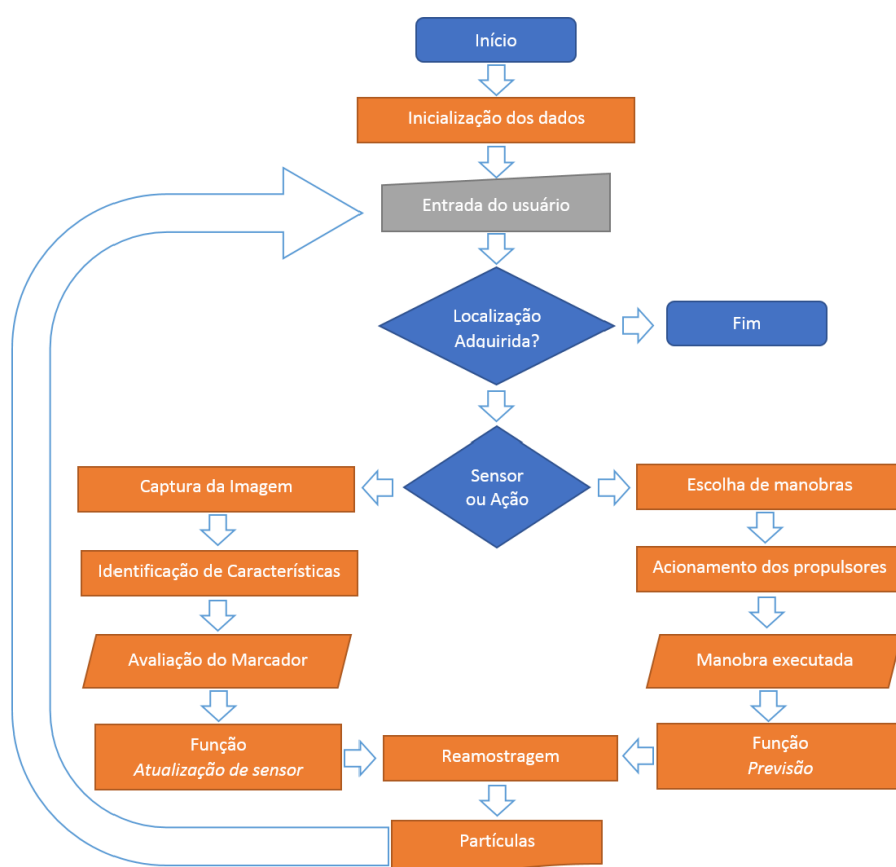


Figura 6.4: Fluxograma do algoritmo implementado no Ensaio de Localização.

[Fonte: Elaborada pelo autor.]

O primeiro passo do fluxograma é a inicialização dos dados do sistema de navegação autônoma. Como este ensaio é apenas para apurar a localização do veículo no terreno, somente o mapa deverá ser carregado anteriormente no programa *Anav*.

Em seguida, o operador entra com um dado no processo. O sistema verifica se esse dado corresponde à confirmação da localização pelo operador. Em caso afirmativo, encerra-se o programa por ter completado seu objetivo. Em caso negativo, o programa analisa se o comando é uma leitura de sensor ou uma ação requisitada.

Caso seja um pedido de leitura do sensor, a câmera é posicionada para visualizar o chão da piscina e captura uma imagem abaixo do veículo. Essa imagem entra no programa de visão computacional para identificação de características. No ensaio foi utilizado o *SimpleCV*, uma *framework* livre que utiliza bibliotecas de visão computacional aplicadas em funções na linguagem de programação *Python* [35]. Sua escolha se deu pela fácil integração entre o código existente no *Anav* e por possuir biblioteca compatível com a placa *Raspberry Pi*. Após captura da imagem, aplica-se a função *colorDistance* para obter o maior

contraste à cor predominante do terreno (nesse caso a cor azul), que foi previamente informada como parâmetro no mapa. Em sequência faz-se uma segregação na imagem de pixels similares, agrupando-os em conjuntos chamados *blobs*, que representam possíveis objetos na imagem. Ao binarizar a imagem, é possível definir corretamente quais são os inúmeros *blobs* presentes na imagem.

O código, então, analisa o maior *blob* e verifica sua quantidade de pixels agrupados. Caso a quantidade seja maior que a mínima calibrada, há uma grande probabilidade de ser um *landmark*, caso contrário há uma pequena probabilidade de não ser o *landmark*. Essa quantidade mínima de pixels foi definida com base em uma calibração prévia de imagens do chão, com e sem *landmarks*, em várias profundidades. Como a cor branca possui um grande contraste com o azul dos azulejos, o prato é facilmente identificado. Porém, o rejunte entre os azulejos também é branco e, frequentemente, somado ao *blob* do prato ao ser processado. Contudo, baseado na calibração e em como os azulejos estão posicionados, a quantidade de pixels nunca ultrapassará o mínimo. Após determinação desse mínimo, para as imagens que possuem prato branco, o tamanho do *blob* será muito maior e, assim, será possível distinguir quando há imagem com ou sem prato.

Após avaliação do *landmark* o resultado é repassado à alimentação da função *atualização de sensor*, que atribui os pesos às partículas no filtro e transmite para a função reamostragem, a fim de criar a nova distribuição de crença do ROV. As partículas geradas ficam visíveis para o usuário, que avalia se a crença é suficiente ou não para confirmar a posição e orientação do veículo no terreno.

Caso o pedido do operador seja uma ação, o operador irá informar a manobra e duração da atuação para atingir o local destinado. O programa *Anav* interpreta esses dados informados pelo operador e gera os sinais de comando a cada propulsor com a finalidade de executar a manobra desejada. Durante a locomoção o operador pode ajustar a posição obtida, a fim de corrigir erros inerentes ao cabo umbilical ou erros na calibração dos pesos auxiliares para corrigir o empuxo. Assim que o usuário alcançar a sua posição, é acionada a função *previsão* para estimar o movimento das partículas, baseada na manobra informada e usando a função reamostragem para criar a nova distribuição.

Neste processo, a incerteza está presente no processamento das imagens para identificação do *landmark* e no processo de controle de movimento pelo operador. Para esse estudo não foi feita a determinação da distribuição de probabilidade do modelo de movimento e sensor, fato este que poderia auxiliar na predição da crença posterior. Recomenda-se para um futuro ensaio a determinação destas probabilidades para serem implementadas no veículo, a fim de obter um sistema autônomo.

## 6.4 EXPERIMENTO

Nesta seção será descrito o processo de localização global do ensaio proposto. Em cada etapa, o operador manobra o veículo até a posição desejada e processa o chão, com a finalidade de colher informações sobre sua localização no mapa. Cada medição feita pela câmera aumenta a crença no modelo, auxiliando a estimar onde o ROV "UnBot" se encontra. Porém, cada manobra feita pelo veículo induz a perda de informação, por não garantir com 100% de probabilidade a posição final na qual deseja-se que o veículo esteja. Esta incerteza juntamente com a incerteza da câmera e IMU foram modeladas como ruídos introduzidos em forma gaussiana. Os valores utilizados estão expostos na figura 6.5.

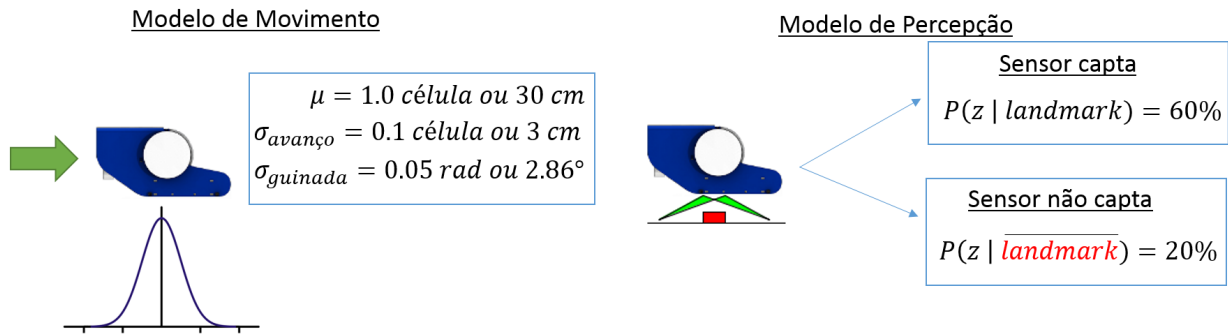


Figura 6.5: Modelos de Movimento e de Percepção do Ensaio.

[Fonte: Elaborada pelo autor.]

Como se pode observar, o valor do desvio padrão em guinada é menor que o de avanço, pela razão de o sensor IMU informar com mais precisão este parâmetro. Como não há sensor dedicado para acompanhar o avanço, a variância neste eixo é maior. Estes valores foram estimados anteriormente em testes na piscina nas condições do ensaio, porém para a aplicação completamente autônoma é necessário o levantamento probabilístico de movimento e de percepção (comentados no Cap. 5) em várias condições de luminosidade e de dinâmica do meio, a fim de obter uma melhor compreensão das capacidades do veículo e dos seus sensores. Somente assim o filtro apresentará dados que abrangem a realidade do veículo independente de onde ele poderá atuar.

Seguindo o fluxograma do ensaio, ilustrado na fig. 6.3, o operador inicializa todas as variáveis e dados do modelo de movimento e de percepção, como também as variáveis para computar a crença no filtro de partículas. A figura 6.6 ilustra a visão inicial do operador sobre a localização do veículo. As células pretas representam o contorno da piscina e são modeladas como obstáculos impenetráveis nesse ensaio. As células azul-claro representam a posição dos *landmarks*, que são conhecidos previamente pelo sistema. A célula vermelha é o alvo que o veículo precisa alcançar, e os pontos laranjas são as partículas que representam a crença da localização do sistema no mapa. A ilustração do ROV "UnBot" é a posição real do veículo, que foi introduzido como parâmetro de comparação entre a sua posição atual e a posição estimada pelo conjunto de partículas. Esse parâmetro serve somente como meio de apuração do resultado e foi inserido por medição de um observador externo.

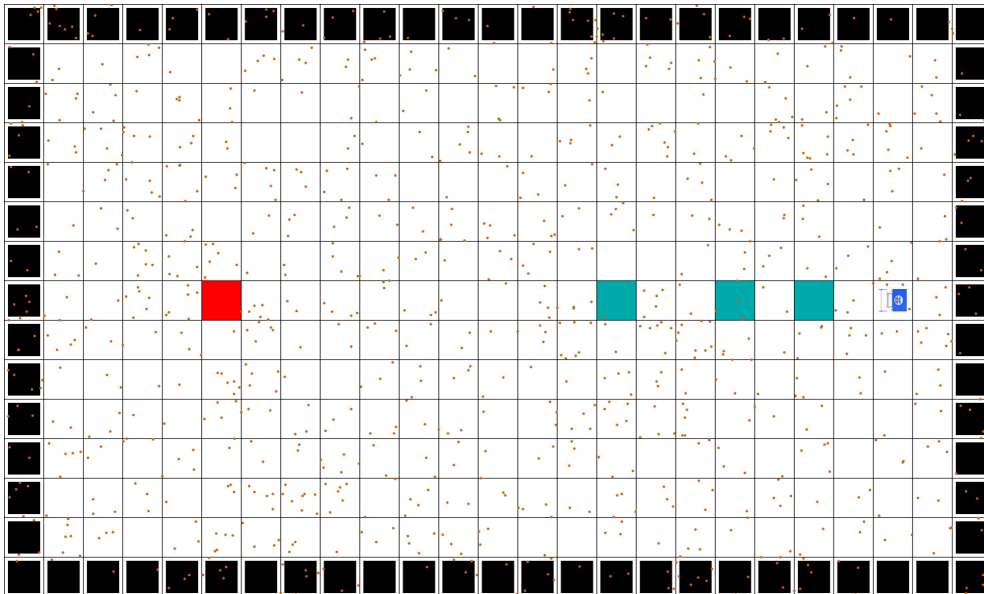


Figura 6.6: Inicialização da crença do veículo no meio.  
 [Fonte: Elaborada pelo autor.]

Inicialmente, o ROV é manobrado para se situar acima do primeiro prato. Esse posicionamento inicial somente auxilia o adiantamento do processo de localização, pois é necessário algum *landmark* para que a crença se aglomere em uma região conhecida. Com o ROV situado acima do primeiro prato, o operador executa um comando de sensor. Uma foto é capturada e é processada pelas funções de visão computacional utilizando a biblioteca *SimpleCV*. Na imagem, o sistema identifica um *blob* de tamanho acima do limite mínimo e responde como positiva a existência de um *landmark* abaixo do veículo. A imagem do veículo em perspectiva no terreno, juntamente com a imagem capturada e processada são ilustradas na figura 6.7.

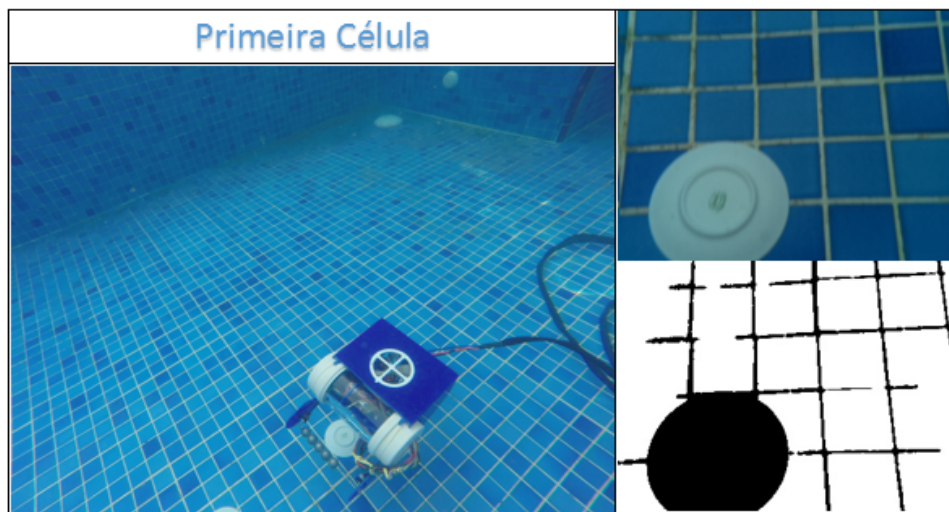


Figura 6.7: Primeira Célula de Análise no Ensaio de Localização.  
 [Fonte: Elaborada pelo autor.]

Neste momento, o operador consegue visualizar no programa *Mission Planner* a crença atual da localização do veículo no mapa, ilustrado pela figura 6.8. A dispersão e incerteza inicial - ilustradas na figura 6.6 - diminuíram em razão das informações recebidas pelos sensores. As novas partículas ficaram concentradas em pequenos grupos em diferentes regiões no mapa. Isso se deve ao desvio padrão e à pouca influência na atribuição dos pesos durante o processo de reamostragem. Porém, pode-se notar que quase a totalidade das partículas estão localizadas na metade direita do mapa, onde realmente se encontra o veículo.

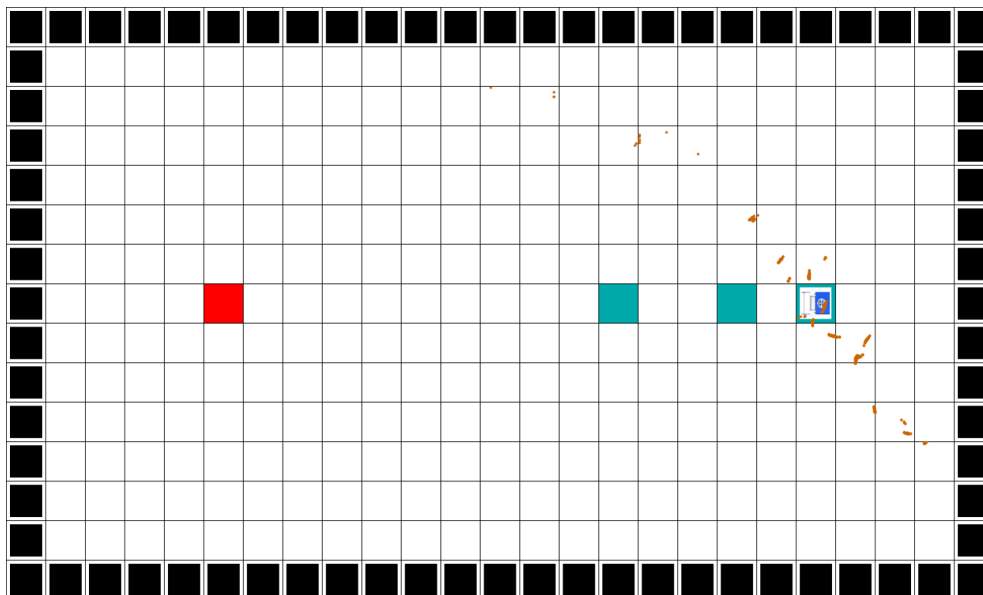


Figura 6.8: Atualização da crença após processamento das medições e movimentos até a célula 1.

[Fonte: Elaborada pelo autor.]

Como a localização do veículo ainda não atingiu um nível de crença requerido, o operador executa o comando de ação para locomover o veículo em busca de mais informações do meio. Repete-se o procedimento de manobra-medição nas próximas duas células do mapa, ilustradas nas figuras 6.9 e 6.10.

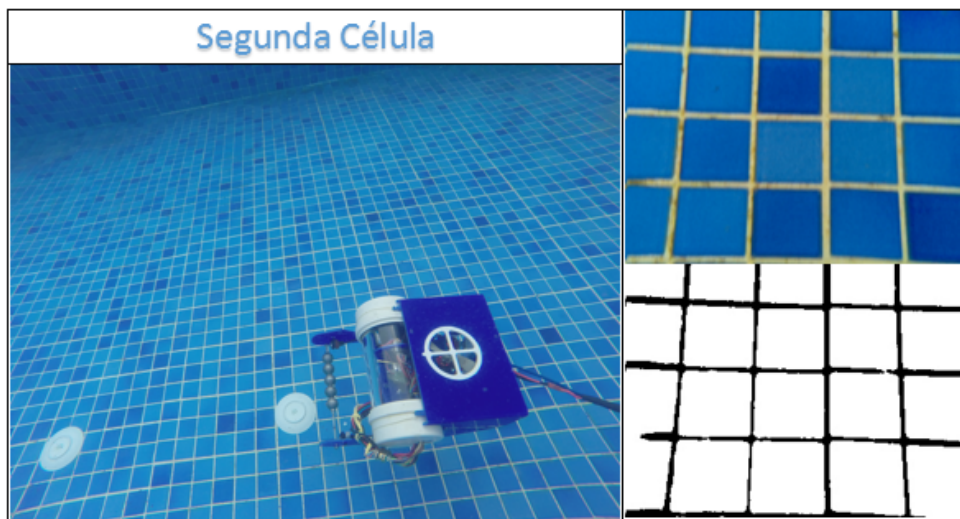


Figura 6.9: Segunda Célula de Análise no Ensaio de Localização.  
[Fonte: Elaborada pelo autor.]

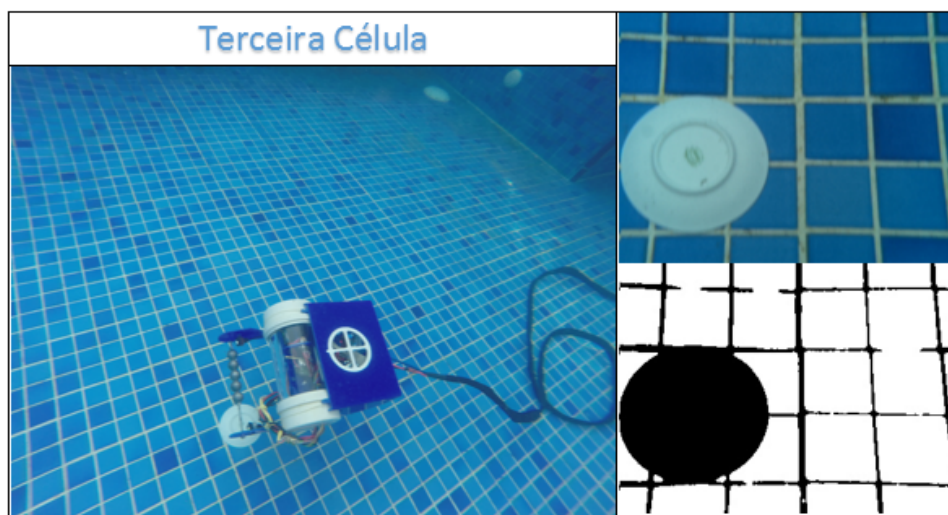


Figura 6.10: Terceira Célula de Análise no Ensaio de Localização.  
[Fonte: Elaborada pelo autor.]

Com essas duas novas medições, o programa *Mission Planner* é atualizado e é possível visualizar o aumento na crença do veículo no mapa, ilustrado na figura 6.11.

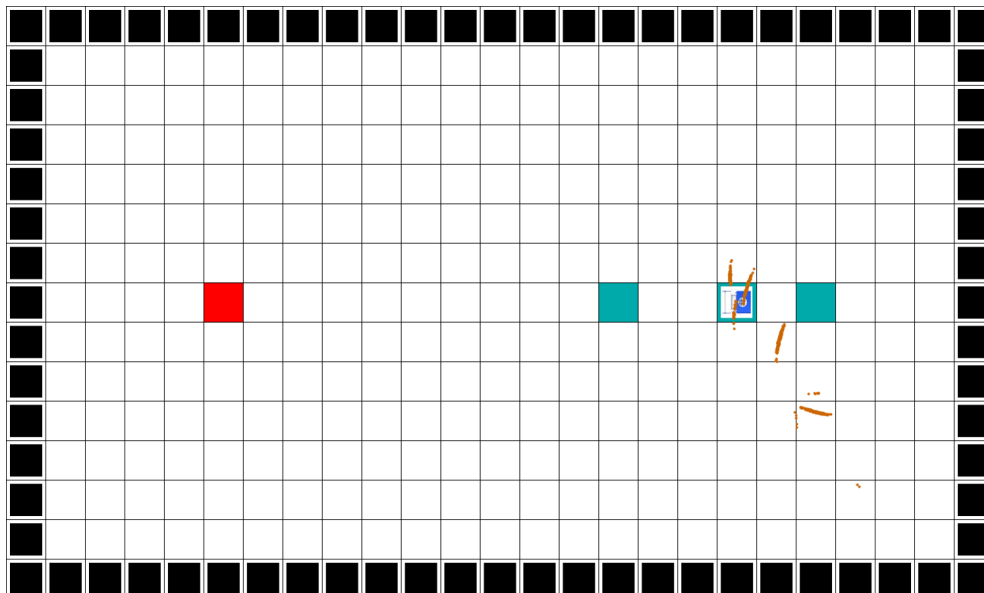


Figura 6.11: Atualização da crença após processamento das medições e movimentos pelas células 2 e 3.  
 [Fonte: Elaborada pelo autor.]

Em seguida, o veículo executa uma manobra de avanço e captura mais uma foto, que após processamento, o resultado é somado à crença prévia. Este processo repete-se nas próximas três células e é ilustrado nas figuras 6.12, 6.13 e 6.14.

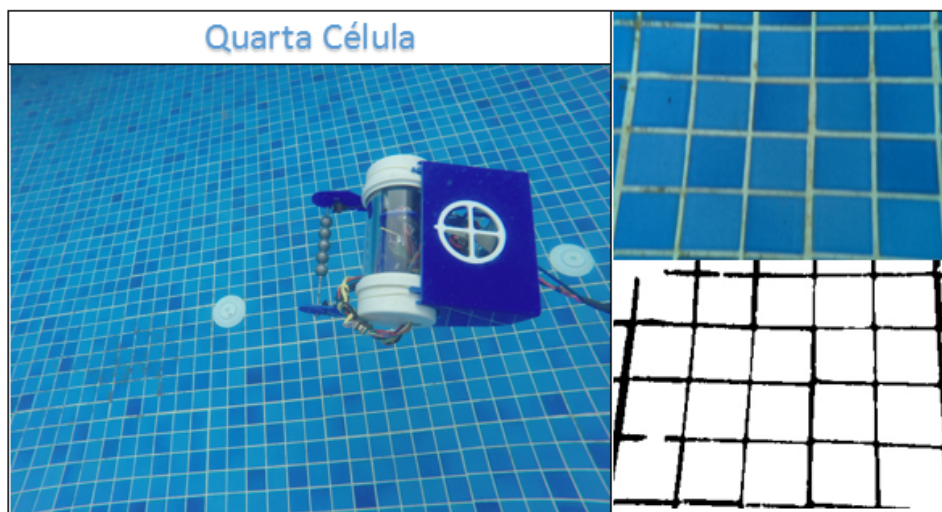


Figura 6.12: Quarta Célula de Análise no Ensaio de Localização.  
 [Fonte: Elaborada pelo autor.]

Na quinta célula, ilustrada pela figura 6.13, é interessante notar que na imagem capturada pelo veículo há uma parte do prato na parte superior da imagem, porém pequena. Na etapa de processamento da imagem, a área do *blob* identificado foi computada como sendo maior que as células que não possuíam marcador. Porém esse valor não passou do limite mínimo e, portanto, não foi classificado como célula que possui *landmark*. Esta observação é importante, pois uma informação errônea poderia ser inferida por erro



na configuração do modelo de percepção. Assim, a montagem dos modelos é que rege como a informação irá ser moldada para entrar no cálculo da distribuição de crença do sistema.

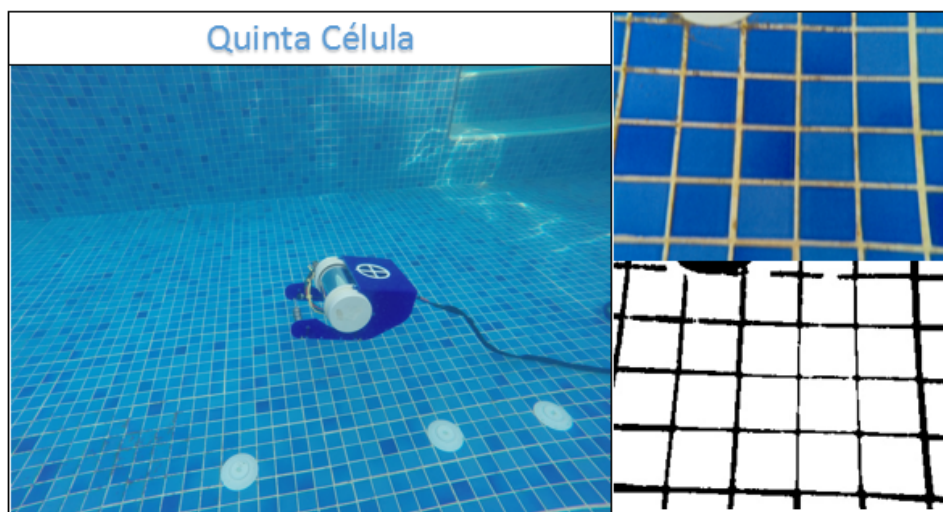


Figura 6.13: Quinta Célula de Análise no Ensaio de Localização.

[Fonte: Elaborada pelo autor.]

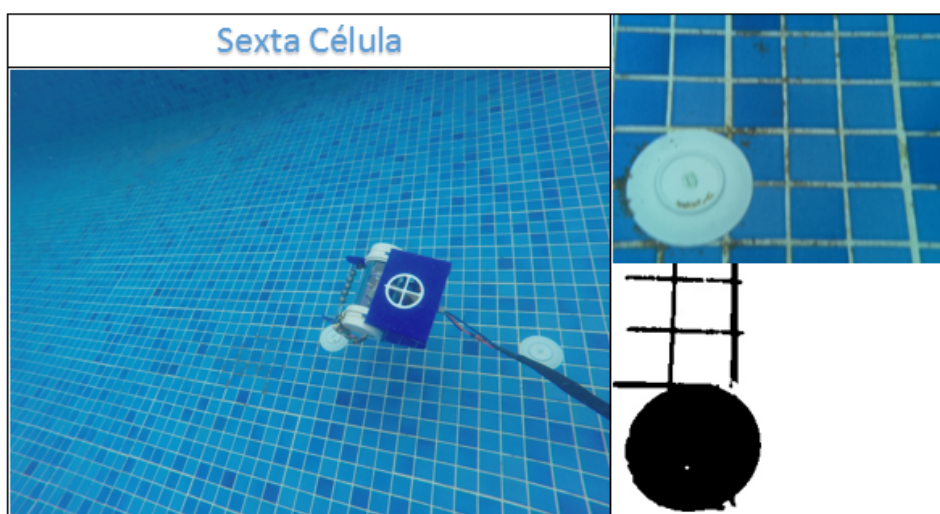


Figura 6.14: Sexta Célula de Análise no Ensaio de Localização.

[Fonte: Elaborada pelo autor.]

Finalmente, quando o veículo atinge o último prato, ilustrado pela figura 6.14, a distribuição atribui um alto grau de crença na posição acima do terceiro *landmark*, como pode ser visto na fig. 6.15. Como o nível de crença está acima do mínimo exigido pelo operador, a localização é considerada adquirida e encerra-se a rotina de busca.

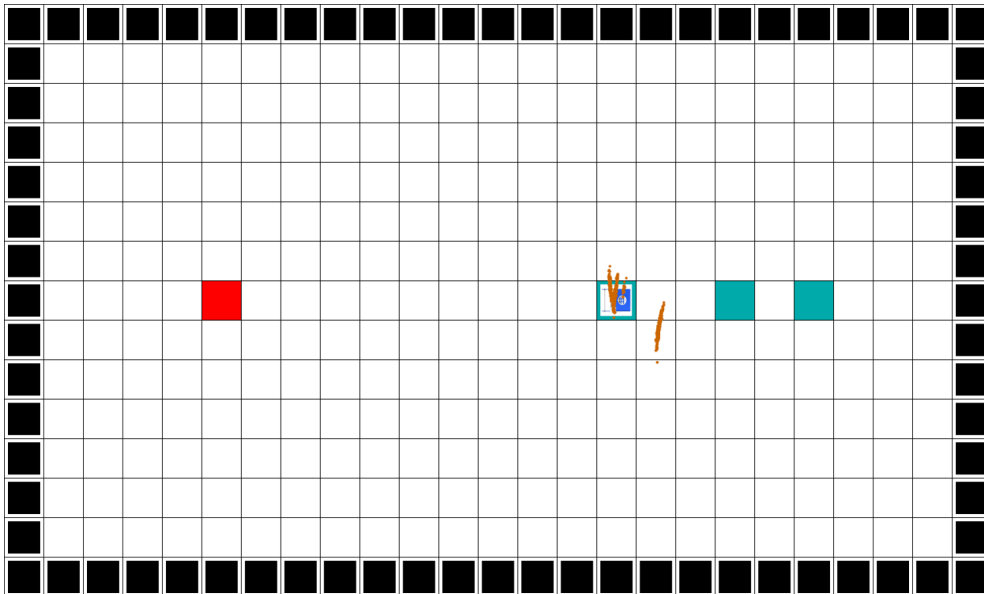


Figura 6.15: Atualização da crença após processamento das medições e movimentos pelas células 4, 5 e 6.  
 [Fonte: Elaborada pelo autor.]

Possuindo a sua localização no meio, o veículo utiliza o algoritmo *A-star* para traçar a trajetória a ser executada até o alvo. Como o meio de ensaio é controlado e com pouca dinâmica, o algoritmo *Dynamic Programming* não foi utilizado.

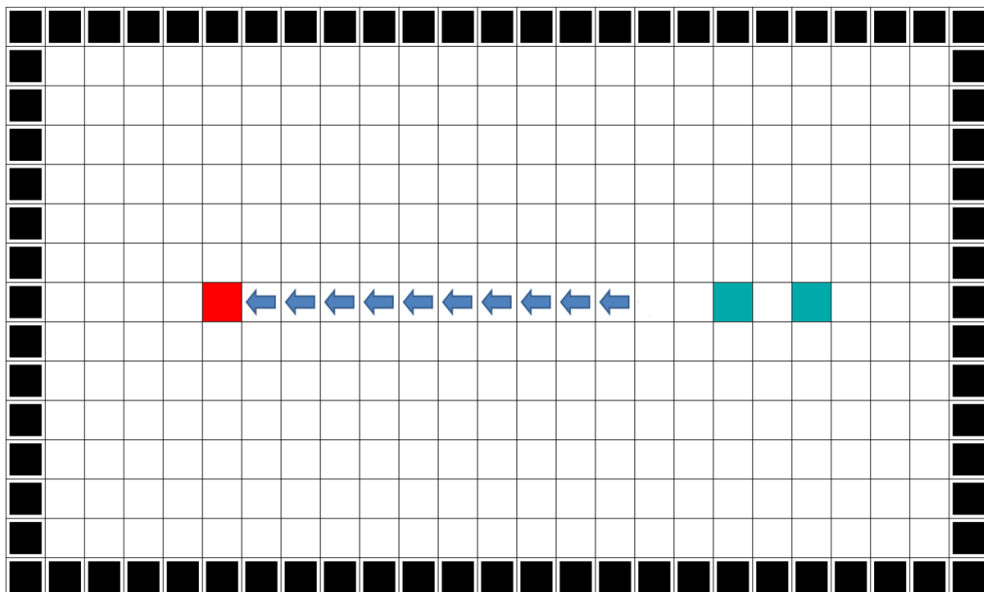


Figura 6.16: Trajetória e diretrizes resultantes do algoritmo *A-star*.  
 [Fonte: Elaborada pelo autor.]

É importante deixar claro que mesmo com o auxílio de atuação comandada pelo operador, o sistema computou de forma autônoma a crença. Ao fazer um controlador e calibrar o sistema com o intuito de conhecer suas distribuições probabilísticas de movimento e percepção, este mesmo ensaio poderá ser efetuado de modo autônomo no terreno bidimensional.

# 7 CONCLUSÕES

## 7.1 SÍNTESE DO TRABALHO REALIZADO

Nos capítulos apresentados foi exposta uma metodologia de obter navegação autônoma em veículos subaquáticos, com um caso exemplo aplicado em um ROV. Para essa metodologia são necessárias as seguintes etapas: Modelagem, Controle, Localização, Planejamento e Ensaios. Feitas essas etapas foi possível obter todos os parâmetros necessários para a montagem do sistema de navegação autônoma apresentado no capítulo 5.

Para demonstrar a metodologia em um caso aplicado foi construído um ROV denominado "UnBot", e dele foram retirados parâmetros físicos para sua modelagem dinâmica de 6 GDLs em ambientes subaquáticos. A modelagem baseou-se na revisão bibliográfica apresentada [11] por um método de estimativa utilizando softwares CAD, CFD e massa adicionada recomendadas para veículos com geometrias complexas que não podem ser aproximados aos métodos semi-empíricos [27]. Com os parâmetros determinados foi possível montar as matrizes dinâmicas da equação de movimento que servirá para o projeto do controlador do veículo.

Juntamente foi apresentada a robótica probabilística e a vantagem dos seus métodos em tratar erros e incertezas como variáveis nas equações. O filtro histograma e filtro de partículas foram expostos como soluções práticas em aplicações de localização de robôs em terrenos conhecidos, comparando as vantagens e desvantagens de cada método para aplicações em veículos subaquáticos. O foco neste trabalho foi resolver o problema de localização global, porém esses algoritmos podem ser aplicados em outras funcionalidades do veículo, caso seja avaliada a necessidade.

Também foram apresentadas soluções de planejamento para terrenos de grade de ocupação, utilizando algoritmos existentes na bibliografia. A necessidade de compreender o funcionamento desses algoritmos foi ressaltada para forçar o leitor a buscar sempre o método de menor custo computacional a fim de obter resultados satisfatórios e rápidos quando aplicados em sistemas embarcados.

Finalmente, uniu-se os dados determinados para realização de um ensaio de localização bidimensional em um ambiente subaquático. O veículo, mesmo limitado pela falta de um controlador para comando autônomo, conseguiu fornecer informações sobre os *landmarks* presentes no terreno e informar ao operador uma posição estimada no meio. O ensaio proposto nesta dissertação serve como base para melhorias e implementações em partes do sistema de navegação autônoma em um veículo subaquático. As maiores dificuldades encontradas foram: a falta de um controlador, baseado no modelo físico montado no capítulo 5, para automatizar o modelo de movimento, e a falta de distribuições de confiabilidade dos sensores e do movimento do veículo, com o intuito de saber quanta incerteza possui o sistema a fim de estimar a crença posterior.

Com esses itens solucionados foi possível executar um segundo ensaio, com a finalidade de apurar a localização global de forma autônoma, com o operador somente supervisionando seu movimento. O fluxograma do algoritmo no ensaio seria alterado para a forma ilustrada na figura 7.1. Em resumo, a entrada

do usuário seria substituída por uma função de avaliação de crença e diretrizes calculadas no planejamento para executar a medição de sensor ou ação desejada ao controlador.

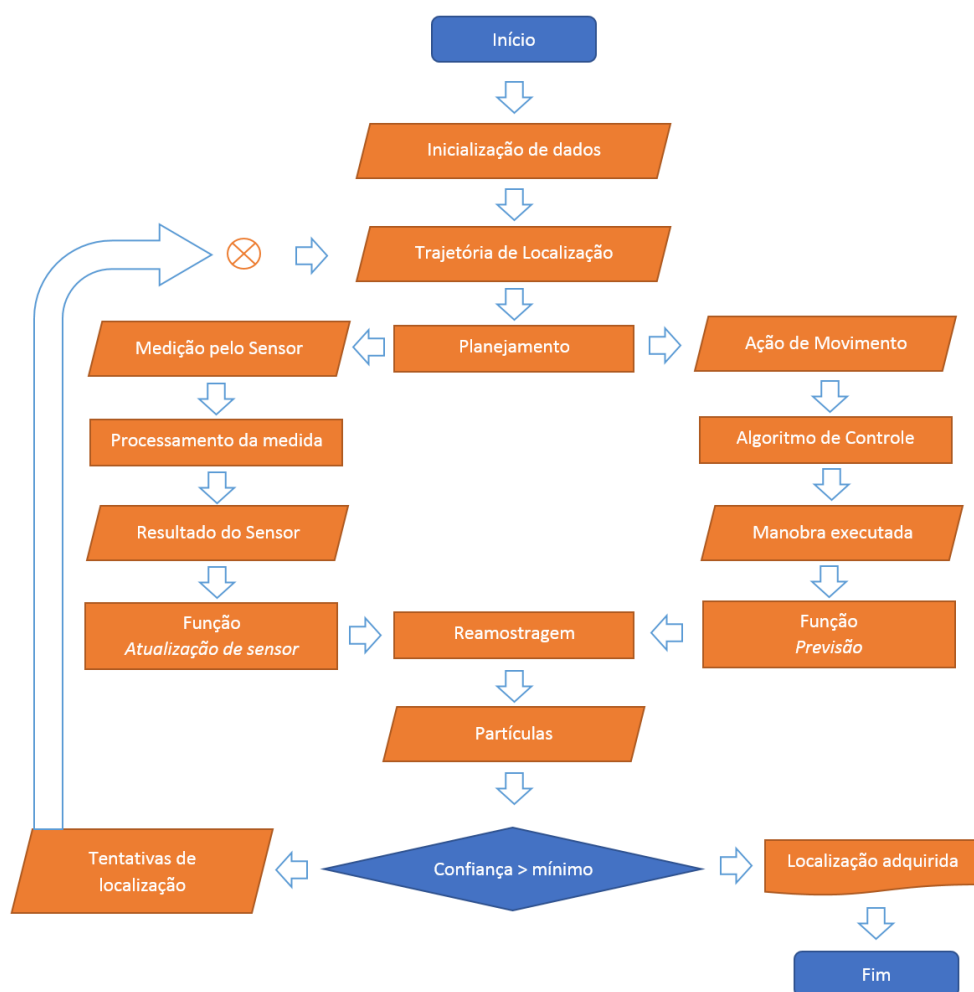


Figura 7.1: Fluxograma do algoritmo implementado no Ensaio de Localização.

[Fonte: Elaborada pelo autor.]

Com bons resultados em um ensaio de localização de modo autônomo foi possível integrar esse fluxograma ilustrado anteriormente no sistema de navegação autônoma por completo.

## 7.2 PERSPECTIVAS DE TRABALHOS FUTUROS

A obtenção de navegação autônoma em UUVs necessita de vários processos que não foram desenvolvidos neste trabalho. Assim, há recomendações de possíveis áreas de melhorias e desenvolvimento para ocorrerem aplicando a metodologia apresentada.

- **Modelagem dos Propulsores:** O sistema de propulsão é o último bloco que falta na determinação completa da dinâmica do veículo. A modelagem poderá ocorrer por métodos semi-empíricos

propostos por [24], porém para aplicação prática é recomendável a caracterização dos propulsores ao estimar as curvas de potência x velocidade e relacionar a força propulsora diretamente ao sinal enviado do sistema, ao invés de inferir a força da geometria da hélice.

- **Projeto do Controlador:** Como comentado no capítulo 5 há vários métodos para determinar um controlador baseado em um modelo não-linear. Com os dados expostos neste trabalho é possível comparar os vários métodos em uma simulação modularizada por blocos, a fim de estimar quais desses métodos proporcionam melhor resultado. Caso possua um veículo teste, é recomendado que esses controladores sejam ensaiados em ambientes subaquáticos para averiguar seu funcionamento e obter sua distribuição de crença.
- **Aumentar as capacidades de atuação e percepção:** Todo ROV é uma plataforma móvel para testar atuadores e sensores em ambientes subaquáticos. A modelagem e calibração desses sensores podem integrar o sistema de navegação autônoma utilizando a robótica probabilística como conexão dos modelos ao sistema. Com ensaios é possível obter estimativas de quais sensores oferecem melhores resultados na localização, comparados à outros em ambientes conhecidos e desconhecidos. Aplicações de algoritmos SLAM (*Simultaneous Localization and Mapping*) no veículo tornam-se possíveis ao conhecer melhor como seus instrumentos funcionam e como tratar a informação proveniente deles.

# REFERÊNCIAS BIBLIOGRÁFICAS

- [1] OLIVER, P. Nauí nitrox diver em português. National Association of Underwater Instructors (NAUI), 2005.
- [2] HILTS, P. J. Em busca de tesouros submersos. *Scientific American Brasil*, 2015.
- [3] MORAIS, J. d. *Petróleo em águas profundas - Uma história tecnológica da PETROBRAS na exploração e produção offshore*. [S.l.]: IPEA, 2013.
- [4] NAVY, U. S. The navy unmanned undersea vehicle (uuv) master plan. Department of the Navy - United States of America, 2004.
- [5] COE, C. Guidance for developing maritime unmanned systems (mus) capability. Combined Joint Operations from the Sea Centre of Excellence, 2012.
- [6] CHRIST, R. *The ROV Manual: A User Guide for Observation Class Remotely Operated Vehicles*. Massachusetts, EUA: Elsevier, 2007. ISBN 0750681489.
- [7] HORGAN, J.; TOAL, D. *Computer Vision Applications in the Navigation of Unmanned Underwater Vehicles, Underwater Vehicles*. [S.l.]: InTech, 2009.
- [8] BRENNEN, C. A review of added mass and fluid inertial forces. Naval Civil Engineering Laboratory, 1982.
- [9] CHIN, C. S. *Computer-Aided Control Systems Design - Practical Applications using MATLAB and Simulink*. Florida, EUA: CRC Press, 2013. ISBN 1466568518.
- [10] 2.20 Marine Hydrodynamics Lecture 13 - Fall 2012. Acessado em 28/05/2014. Disponível em: <<http://web.mit.edu/2.20/www/lectures/Lecture-2012/lecture13-2012.pdf>>.
- [11] FOSSEN, T. I. *Marine Control Systems - Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*. Trondheim, Noruega: Marine Cybernetics, 2002. ISBN 8292356002.
- [12] NEWMAN, J. *Marine Hydrodynamics*. Massachusetts, EUA: MIT Press, Cambridge, 1977. ISBN 0262140268.
- [13] BEKEY, G. A. *Autonomous robots: from biological inspiration to implementation and control*. Massachusetts, EUA: Intelligent Robotics and Autonomous Agents series, MIT Press, 2005. ISBN 0262025787.
- [14] THRUN, S.; BURGARD, W.; FOX, D. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. Massachusetts, EUA: The MIT Press, 2005. ISBN 0262201623.
- [15] KOROTKIN, A. *Added Mass of Ship Structures*. St. Petersburg, Rússia: Springer Science, 2009. ISBN 1402094310.

- [16] OGILVIE, T. Recent progress towards the understanding and prediction of ship motions. 5th Symposium Naval Hydrodynamics, pg. 3 - 79, 1964.
- [17] CS373 - Artificial Intelligence for Robotics. Acessado em 31/10/2014. Disponível em: <<https://udacity.com/>>.
- [18] HART, P.; NILSSON, N.; RAPHAEL, B. A formal basis for the heuristic determination of minimum cost paths. IEEE Transactions of Systems and Cybernetics, Vol SSC-4, No. 2, 1968.
- [19] WORLD Representation and Path Planning Using the A-star Algorithm. Acessado em 31/10/2014. Disponível em: <<http://computerrobotvision.org/2010/>>.
- [20] MURPHY, R. R. *Introduction to AI Robotics*. Massachusetts, EUA: A Bradford Book, MIT Press, 2000. ISBN 0262133830.
- [21] OPENROV project. Acessado em 27/08/2014. Disponível em: <<http://openrov.com/>>.
- [22] THOMAS, J. E. *Fundamentos de engenharia de petróleo*. Rio de Janeiro: Editora Interciência, Petrobrás, 2004. ISBN 8571930996.
- [23] FONSECA, M. Arte naval, volumes i e ii. Serviço de Documentação da Marinha, 7a ed., 2005.
- [24] FERREIRA, B. Modelação e controlo de veículo submarino com quatro graus de liberdade. Faculdade de Engenharia da Universidade de Porto, Portugal, 2009.
- [25] WANG, W. Autonomous control of a differential thrust micro rovs. University of Waterloo, Ontario, Canada, 2006.
- [26] VALDOVINOS, L. Modelling, design and robust control of a remotely operated underwater vehicle. International Journal of Advanced Robotic Systems, 2013.
- [27] LAU, M. Modeling and testing of hydrodynamic damping model for a complex-shaped remotely-operated vehicle for control. Journal of Marine Science and Application, 2012.
- [28] AMASS Software 1.4. Acessado em 25/05/2014. Disponível em: <<http://web.mit.edu/2.20/www/software.html>>.
- [29] SOBACHKIN, A. Flow basis of cad embedded cfd whitepaper. Dassault Systems, 2013.
- [30] VALAVANIS, K. Control architecture for autonomous underwater vehicles. IEEE Controle Systems, vol. 17, pg.48 - 64, 1997.
- [31] LUQUE, J. Controle robusto multivariável para um veículo submersível autônomo. Escola Politécnica da Universidade de São Paulo - Departamento de Engenharia Mecânica, 2007.
- [32] SANTOS, J. Utilização de redes neurais artificiais no controle de sistemas mecânicos incertos. Universidade Federal do Rio Grande do Norte (UFRN), Natal, 2012.
- [33] RICHARDSON, M. *Getting Started with Raspberry Pi*. Califórnia, EUA: MAKE Magazine - O'Reilly Media Inc, 2012. ISBN 1449344216.

[34] VIDEORAY. Videoray pro 4 operator's manual. VideoRay LLC, 2014.

[35] DEMAAGD, K. et al. *Practical Computer Vision with SimpleCV*. California, EUA: O'Reilly Media, 2012. ISBN 1449320368.





# I. DADOS TÉCNICOS DO UNBOT

## I.1 DADOS TÉCNICOS DOS EQUIPAMENTOS DO ROV



Figura I.1: Motor WayPoint - Propulsor

Tabela I.1: Especificações do Motor WayPoint 2213-0760

<b>Tipo de Motor</b>	Outrunner de 12 polos trifásico
<b>RPM/V</b>	760Kv
<b>Corrente Máx.</b>	11A / Burst 30s
<b>Potência Operacional (no ar)</b>	110W
<b>Corrente Operaciona (no ar)</b>	4 9A
<b>Células Lipo indicado (no ar)</b>	2 3S
<b>Peso</b>	57g
<b>Dimensões</b>	28.8mm x 29mm
<b>Diâmetro do Eixo</b>	5mm

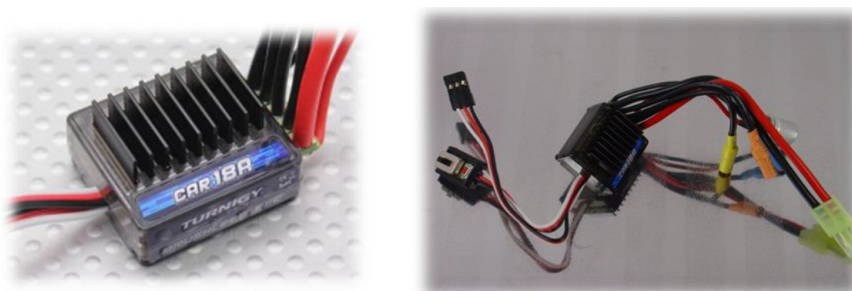


Figura I.2: Esc Turnigy Brushless com reversível

Tabela I.2: Especificações da Esc

<b>Modelo</b>	CAR 18A
<b>Corrente Máx - Contínua</b>	18A
<b>Corrente Máx - Instantânea(Burst)</b>	50A
<b>Resistência</b>	0.001 $\Omega$
<b>Tensão de Trabalho</b>	4V - 16V
<b>Células Lipo indicados</b>	2 3S
<b>BEC(battery Eliminator Circuit)</b>	Incluso para operações até 3S. Saída de 6V - 1A
<b>Dimensões</b>	31.5mm x 24mm x 15mm
<b>Proteção de Superaquecimento</b>	Corte aos 110°C
<b>Comando</b>	PWM a 8KHz



Figura I.3: Mini-Maestro

Tabela I.3: Especificações do Mini-Maestro

<b>Modelo</b>	6 Canais
<b>Comunicação</b>	Serial com protocolo próprio(300 - 200000bps)
<b>Alimentação</b>	5 - 16V
<b>Corrente</b>	30mA (para placa). Alimentação externa para os motores
<b>Dimensão</b>	22mm x 30.5mm

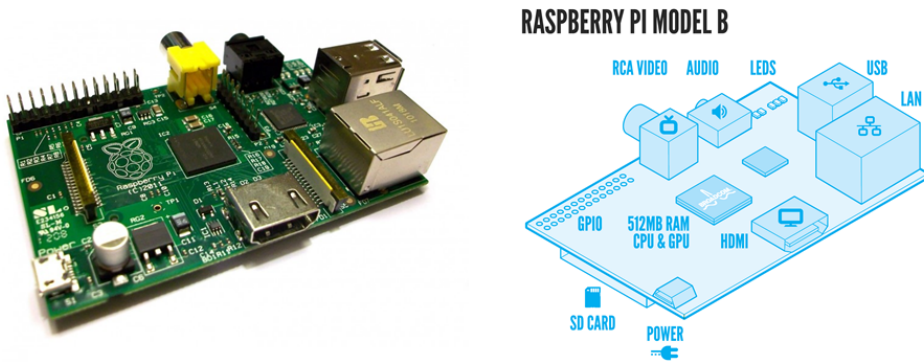


Figura I.4: Placa Raspberry Pi

Tabela I.4: Especificações do Raspberry Pi

<b>Modelo</b>	B (rev. 2)
<b>SoC</b>	Broadcom BCM2835
<b>CPU</b>	ARM1176JZF-S(ARMv6k) 700MHz
<b>RAM</b>	512MB
<b>GPU</b>	Broadcom VideCore IV @ 250MHz
<b>Comunicação</b>	10/100 Mbit/s Ethernet com 2 portas USB e Comunicação Serial via GPIO
<b>Alimentação</b>	5V via MicroUSB ou entrada GPIO
<b>Consumo</b>	700 mA (3.5W)
<b>Dimensões</b>	85.60mm x 56mm
<b>Peso</b>	45g



Figura I.5: Rpi Camera

Tabela I.5: Especificações do Rpi Camera

<b>Resolução@fps de video</b>	1080p@30fps, 720p@60fps e 480p@90fps
<b>Resolução do sensor</b>	5MP
<b>Resolução Máxima de imagem de foto</b>	2592 x 1944
<b>Conector</b>	MIPI Interface Serial de 15 pinos(CSI)
<b>Dimensões</b>	20mm x 25mm
<b>Espessura da placa</b>	9mm

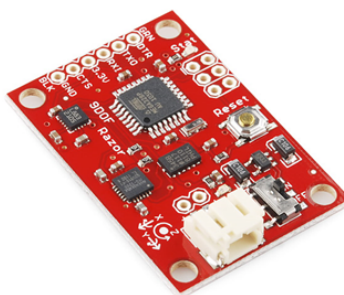


Figura I.6: IMU Razor 9DOF

Tabela I.6: Especificações do IMU

<b>Graus de Liberdade</b>	9
<b>Gyrocópio</b>	ITG-3200 de três eixos
<b>Acelerômetro</b>	ADXL345 de três eixos
<b>Magnetômetro</b>	HMC5883L de três eixos
<b>Placa embarcada de processamento</b>	Atmega 328p a 8Mhz
<b>Saída de dados</b>	Serial (default configurado a 57600bps)
<b>Alimentação</b>	3.3V
<b>Dimensão</b>	28 x 41 mm

## I.2 ENTRADAS DAS GEOMETRIAS DE MASSA ADICIONADA (CAP. 2)

**Esfera**

Input radius 0.5  
Input X displacement 1.5  
Input Y displacement 0  
Input Z displacement 0  
Enter

**Clindro 1**

Top  
Input top radius 0.5  
Input X top center 1.5  
Input Y top center 0  
Input Z top center 0  
Base  
Input base radius 0.5  
Input X base center -1.5  
Input Y base center 0  
Input Z base center 0  
Enter

**Clindro 2**

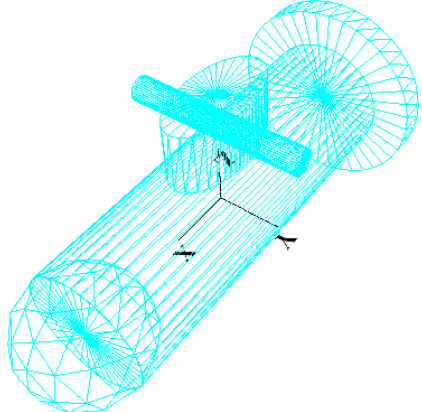
Top  
Input top radius 0.8  
Input X top center -1.5  
Input Y top center 0  
Input Z top center 0  
Base  
Input base radius 0.8  
Input X base center -1.7  
Input Y base center 0  
Input Z base center 0  
Enter

**Clindro 3**

Top  
Input top radius 0.1  
Input X top center 0  
Input Y top center 0.7  
Input Z top center 0.75  
Base  
Input base radius 0.1  
Input X base center 0  
Input Y base center -0.7  
Input Z base center 0.75  
Enter

**Clindro Elipsoidal**

Top  
Input X top center 0  
Input Y top center 0  
Input Z top center 1  
Input first axes point  
 Input X coordinate 0.5  
 Input Y coordinate 0  
 Input Z coordinate  
Input top second axis length 0.25  
Base  
Input X base center 0  
Input Y base center 0  
Input Z base center 0.5  
Input base first axis length 0.5  
Input base second axis length 0.25  
Enter



**Matriz Exemplo**

2.8759	0.0	0.0	0.0	0.1066	0.0
0.0	3.2123	0.0	-0.2914	0.0	-0.5452
0.0	0.0	3.0886	0.0	0.5452	0.0
0.0	-0.2914	0.0	0.2339	0.0	0.0
0.1066	0.0	0.5452	0.0	3.0292	0.0
0.0	-0.5452	0.0	0.0	0.0	2.9593

Figura I.7: Entradas da geometria do veículo na Fig. 2.1, pg. 19  
[Fonte: Elaborada pelo autor.]

# Clindro 1

Input para...

Top

Input top radius 0.5

Input X top center 1.5

Input Y top center 0

Input Z top center 0

Base

Input base radius 0.5

Input X base center -1.5

Input Y base center 0

Input Z base center 0

Enter

# Clindro 2

Input para...

Top

Input top radius 0.1

Input X top center 0

Input Y top center 0.7

Input Z top center 0.75

Base

Input base radius 0.1

Input X base center 0

Input Y base center -0.7

Input Z base center 0.75

Enter

# Clindro Elipsoidal

Input parameters

Top

Input X top center 0

Input Y top center 0

Input Z top center 1

Input first axes point

Input X coordinate 0.5

Input Y coordinate 0

Input Z coordinate

Input top second axis length 0.25

Base

Input X base center 0

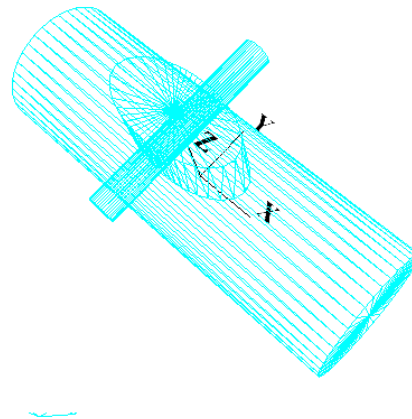
Input Y base center 0

Input Z base center 0.5

Input base first axis length 0.5

Input base second axis length 0.25

Enter



## Matriz Exemplo

0.6658	0.0	0.0	0.0	0.1066	0.0
0.0	2.7447	0.0	-0.2914	0.0	0.0
0.0	0.0	2.6211	0.0	0.0	0.0
0.0	-0.2914	0.0	0.2339	0.0	0.0
0.1066	0.0	0.0	0.0	1.8512	0.0
0.0	0.0	0.0	0.0	0.0	1.7812

Figura I.8: Entradas da geometria do veículo na Fig. 2.2, pg. 20

[Fonte: Elaborada pelo autor.]

# Esfera

Input radius

Input X displacement

Input Y displacement

Input Z displacement

Enter

# Clindro 1

Top

Input top radius

Input X top center

Input Y top center

Input Z top center

Base

Input base radius

Input X base center

Input Y base center

Input Z base center

Enter

# Clindro 2

Top

Input top radius

Input X top center

Input Y top center

Input Z top center

Base

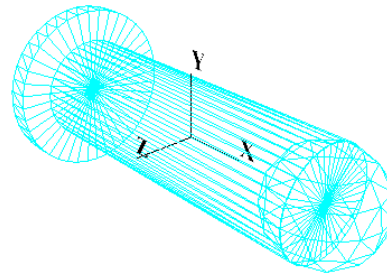
Input base radius

Input X base center

Input Y base center

Input Z base center

Enter



### Matriz Exemplo

2.7337	0.0	0.0	0.0	0.0	0.0
0.0	2.8238	0.0	0.0	0.0	-0.5452
0.0	0.0	2.8238	0.0	0.5452	0.0
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.5452	0.0	2.9452	0.0
0.0	-0.5452	0.0	0.0	0.0	2.9452

Figura I.9: Entradas da geometria do veículo na Fig. 2.3, pg. 21

[Fonte: Elaborada pelo autor.]



# Esfera 1

Input radius

Input X displacement

Input Y displacement

Input Z displacement

# Esfera 2

Input radius

Input X displacement

Input Y displacement

Input Z displacement

# Clindro

Top

Input top radius

Input X top center

Input Y top center

Input Z top center

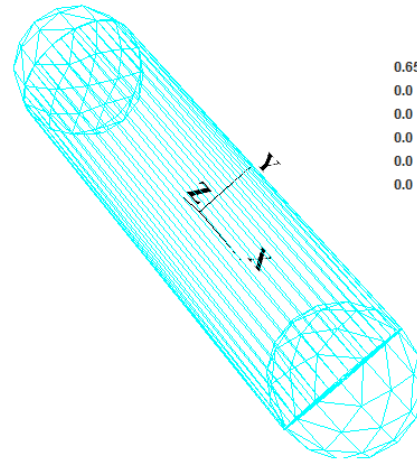
Base

Input base radius

Input X base center

Input Y base center

Input Z base center

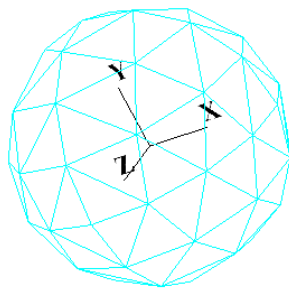


### Matriz Exemplo

0.6545	0.0	0.0	0.0	0.0	0.0
0.0	2.4871	0.0	0.0	0.0	0.0
0.0	0.0	2.4871	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	2.0617	0.0
0.0	0.0	0.0	0.0	0.0	2.0617

Figura I.10: Entradas da geometria do veículo na Fig. 2.4, pg. 22

[Fonte: Elaborada pelo autor.]



Matriz Exemplo

2.0944	0.0	0.0	0.0	0.0	0.0
0.0	2.0944	0.0	0.0	0.0	0.0
0.0	0.0	2.0944	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0

## Esfera

Input parame...

Input radius

Input X displacement

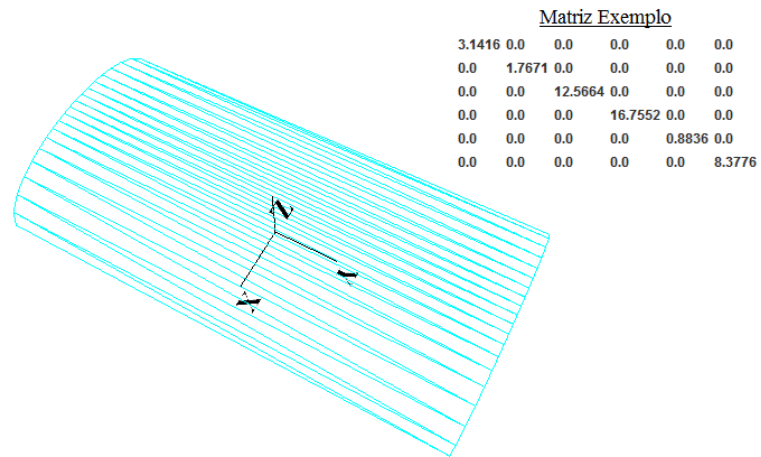
Input Y displacement

Input Z displacement

Figura I.11: Entradas da geometria do veículo na Fig. 2.5, pg. 23

[Fonte: Elaborada pelo autor.]

# Clindro Elipsoidal



Input parameters

Top

Input X top center 0

Input Y top center -2

Input Z top center 0

Input first axes point

Input X coordinate 1

Input Y coordinate

Input Z coordinate 0

Input top second axis length 0.5

Base

Input X base center 0

Input Y base center 2

Input Z base center 0

Input base first axis length 1

Input base second axis length 0.5

Enter

Figura I.12: Entradas da geometria do veículo na Fig. 2.6, pg. 23

[Fonte: Elaborada pelo autor.]

### I.3 MATRIZES DO MODELO ROV

$$M_{RB} = \begin{bmatrix} 2.6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2.6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01631 & 0.00021 & -0.00454 \\ 0 & 0 & 0 & 0.00021 & 0.02463 & 0.00093 \\ 0 & 0 & 0 & -0.00454 & 0.00093 & 0.02131 \end{bmatrix} \quad (\text{I.1})$$

$$C_{12}(v) = \begin{bmatrix} 0 & 2.6w & -2.6v \\ -2.6w & 0 & 2.6u \\ 2.6v & -2.6u & 0 \end{bmatrix} \quad (\text{I.2})$$

$$C_{22}(v) = \begin{bmatrix} 0 & 0.00093q - 0.00454p + 0.02131r & -0.00093r - 0.00021p - 0.02463q \\ -0.00093q + 0.00454p - 0.02131r & 0 & -0.00454r + 0.00021q + 0.01631p \\ 0.00093r + 0.00021p + 0.02463q & 0.00454r - 0.00021q - 0.01631p & 0 \end{bmatrix} \quad (\text{I.3})$$

$$M_A = \begin{bmatrix} 0.0057 & 0 & 0 & 0 & -0.0002 & 0 \\ 0 & 0.0126 & 0 & 0.0002 & 0 & -0.0004 \\ 0 & 0 & 0.0072 & 0 & 0.0003 & 0 \\ 0 & 0.0002 & 0 & 0.0001 & 0 & 0 \\ -0.0002 & 0 & 0.0003 & 0 & 0 & 0 \\ 0 & -0.0004 & 0 & 0 & 0 & 0.0001 \end{bmatrix} \quad (I.4)$$

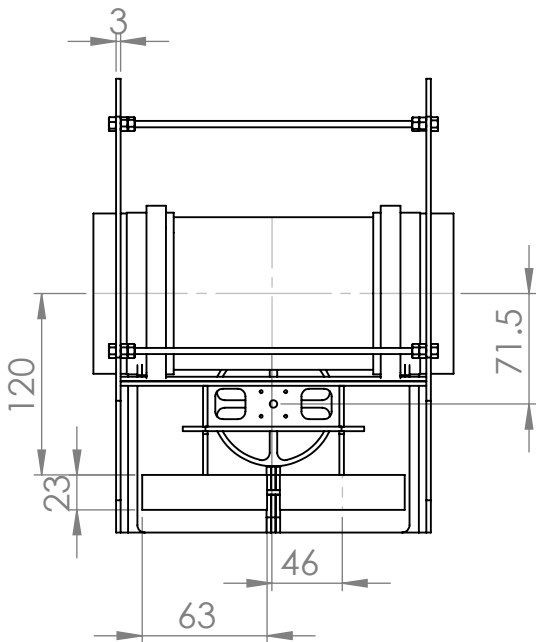
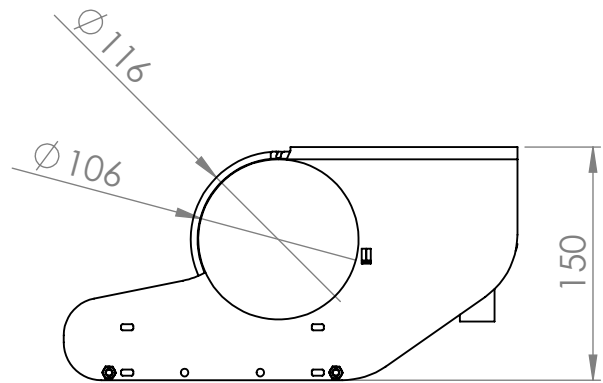
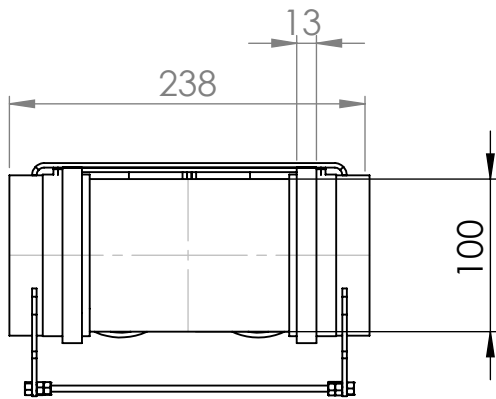
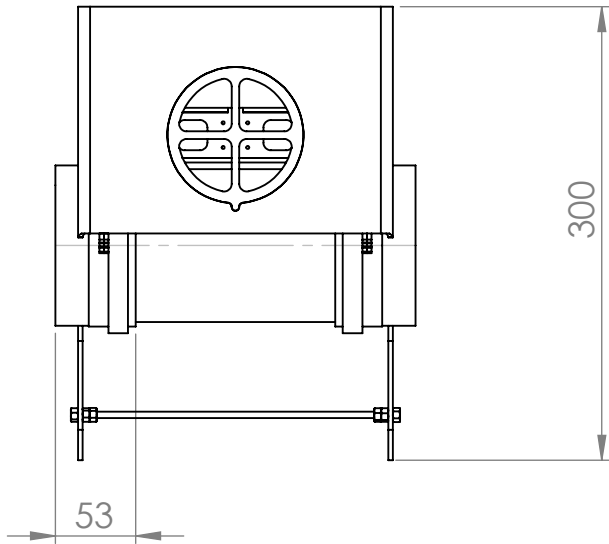
$$C_A = \begin{bmatrix} 0 & 0 & 0 & 0 & -0.072w - 0.0003q & 0.0002p - 0.0004r + 0.0126v \\ 0 & 0 & 0 & 0.072w + 0.0003q & 0 & -0.0057u + 0.0002q \\ 0 & 0 & 0 & -0.0002p + 0.0004r - 0.0126v & 0.0057u - 0.0002q & 0 \\ 0 & -0.072w - 0.0003q & 0.0002p - 0.0004r + 0.0126v & 0 & -0.0001r + 0.0004v & 0.0003w - 0.0002u \\ 0.072w + 0.0003q & 0 & -0.0057u + 0.0002q & 0.0001r - 0.0004v & 0 & -0.0001p - 0.0002v \\ -0.0002p + 0.0004r - 0.0126v & 0.0057u - 0.0002q & 0 & -0.0003w + 0.0002u & 0.001p + 0.0002v & 0 \end{bmatrix} \quad (I.5)$$

$$D(\mathbf{v}|\mathbf{v} \geq \mathbf{0}) = -diag[-0.0623 + 6.8116|u|, -0.0414 + 15.854|v|, -0.2143 + 23.684|w|, -0.0019 + 0.0191|p|, 0.002 + 0.0353|q|, -0.0003 + 0.0491|r|] \quad (I.6)$$

$$D(\mathbf{v}|\mathbf{v} < \mathbf{0}) = -diag[-0.0089 - 8.0829|u|, -0.0414 - 15.854|v|, -0.019 - 19.065|w|, -0.0019 - 0.0191|p|, -0.074 - 0.0484|q|, -0.0003 - 0.0491|r|] \quad (I.7)$$

$$G_f = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.78\cos(\theta)\sin(\eta) \\ 0.78\sin(\theta) \\ 0 \end{bmatrix} \quad (I.8)$$

## **II. PLANTAS DO ROV "UNBOT"**



COMENTÁRIOS:

DATE: 19/11/2014

UnB

NAME: Nathan Costa Alves Souza

TITLE:

Vistas Principais

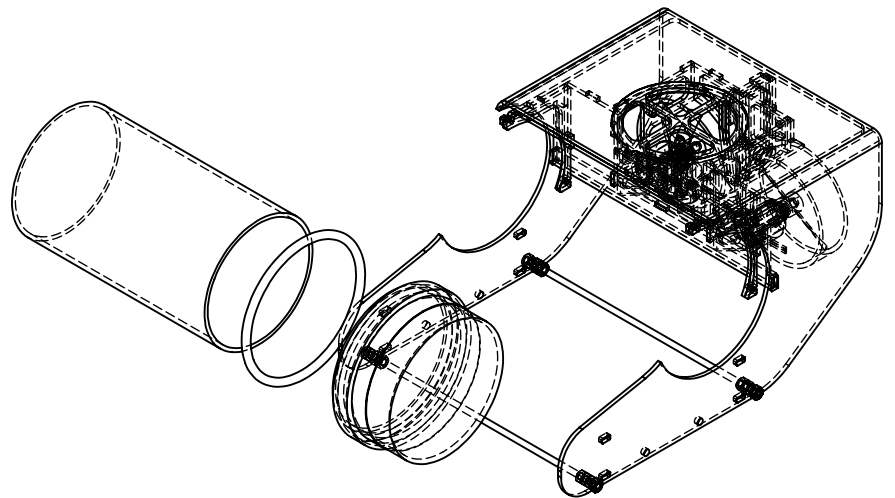
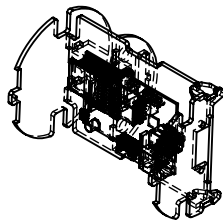
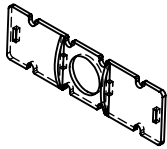
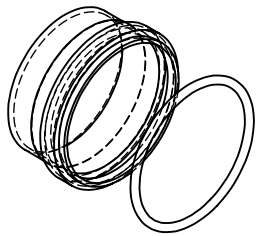
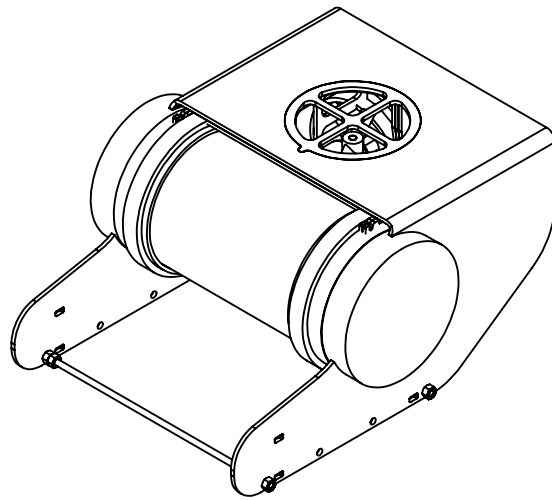
DWG NO.

ROV "UnBot"

A4

ESCALA: 1:5

PLANTA 1



COMENTÁRIOS:

DATE: 19/11/2014

UnB

NAME: Nathan Costa Alves Souza

TITLE: Montagem Cápsula  
Eletrônica

DWG NO. ROV "UnBot"

A4

ESCALA: 1:5

PLANTA 2