



*Dissertação de Mestrado*

# Framework de monitoramento persistente multiagente em ambientes marítimos

de Glauber Rodrigues Leite

orientado por

Prof. Dr. Heitor Judiss Savino

Prof. Dr. Ícaro Bezerra Queiroz de Araújo

Universidade Federal de Alagoas  
Instituto de Computação  
Maceió, Alagoas  
28 de Fevereiro de 2022

UNIVERSIDADE FEDERAL DE ALAGOAS  
Instituto de Computação

**FRAMEWORK DE MONITORAMENTO PERSISTENTE  
MULTIAGENTE EM AMBIENTES MARÍTIMOS**

Dissertação de Mestrado submetida ao Instituto de Computação da Universidade Federal de Alagoas como requisito parcial para a obtenção do grau de Mestre em Informática.

Glauber Rodrigues Leite

*Orientador: Prof. Dr. Heitor Judiss Savino*

*Coorientador: Prof. Dr. Ícaro Bezerra Queiroz de Araújo*

**Banca Avaliadora:**

Thiago Damasceno Cordeiro Prof. Dr., IC-UFAL

Allan de Medeiros Martins Prof. Dr., UFRN

Maceió, Alagoas  
28 de Fevereiro de 2022

**Catálogo na Fonte**  
**Universidade Federal de Alagoas**  
**Biblioteca Central**  
**Divisão de Tratamento Técnico**

Bibliotecário: Marcelino de Carvalho Freitas Neto – CRB-4 - 1767

L533f Leite, Glauber Rodrigues.  
*Framework* de monitoramento persistente multiagente em ambientes marítimos / Glauber Rodrigues Leite. – 2022.  
41 f. : il.

Orientador: Heitor Judiss Savino.  
Coorientador: Ícaro Bezerra Queiroz de Araújo.  
Dissertação (mestrado em informática) - Universidade Federal de Alagoas. Instituto de Computação. Maceió, 2022.

Bibliografia: f. 38-41.

1. Monitoramento ambiental. 2. Robôs cooperativos. 3. Derramamento de óleo. 4. Busca e rastreamento. 5. Sensoriamento remoto. I. Título.

CDU: 528.8



UNIVERSIDADE FEDERAL DE ALAGOAS/UFAL  
**Programa de Pós-Graduação em Informática – PPGI**  
**Instituto de Computação/UFAL**  
Campus A. C. Simões BR 104-Norte Km 14 BL 12 Tabuleiro do Martins  
Maceió/AL - Brasil CEP: 57.072-970 | Telefone: (082) 3214-1401



## Folha de Aprovação

GLAUBER RODRIGUES LEITE

### FRAMEWORK DE MONITORAMENTO PERSISTENTE MULTIAGENTE EM AMBIENTES MARÍTIMOS

Dissertação submetida ao corpo docente do Programa de Pós-Graduação em Informática da Universidade Federal de Alagoas e aprovada em 28 de fevereiro de 2022.

#### Banca Examinadora:



Documento assinado digitalmente  
Heitor Judiss Savino  
Data: 08/03/2022 08:58:16-0300  
Verifique em <https://verificador.iti.br>

---

**Prof. Dr. HEITOR JUDISS SAVINO**  
UFAL – Instituto de Computação  
**Orientador**



Documento assinado digitalmente  
Icaro Bezerra Queiroz de Araujo  
Data: 07/03/2022 09:42:15-0300  
Verifique em <https://verificador.iti.br>

---

**Prof. Dr. ICARO BEZERRA QUEIROZ DE ARAUJO**  
UFAL – Instituto de Computação  
**Coorientador**



Documento assinado digitalmente  
Thiago Damasceno Cordeiro  
Data: 07/03/2022 22:51:26-0300  
Verifique em <https://verificador.iti.br>

---

**Prof. Dr. THIAGO DAMASCENO CORDEIRO**  
UFAL – Instituto de Computação  
**Examinador Interno**



Documento assinado digitalmente  
ALLAN DE MEDEIROS MARTINS  
Data: 07/03/2022 10:35:45-0300  
Verifique em <https://verificador.iti.br>

---

**Prof. Dr. ALLAN DE MEDEIROS MARTINS**  
UFRN – Universidade Federal do Rio Grande do Norte  
**Examinador Externo**

# Dedicatória

À minha família

# Agradecimentos

Agradeço à minha família, meus pais Geovane e Luciene que vibram a cada conquista que a educação proporcionou em minha vida, para além disso não há palavras que possam expressar a gratidão que tento demonstrar por ser filho deles. Igualmente à minha noiva Elian, que consegue me ajudar a suportar as aventuras da vida acadêmica, pois ela também as vive na sua jornada, me oferecendo a visão balanceada que preciso para conseguir lidar com outras coisas da vida que também são importantes. Não posso deixar de agradecer a meus irmãos Gabriel e Geovane Filho, pois compartilharam alegrias do passado e sonhos do futuro, sendo meus grandes amigos desde muito antes da vida acadêmica. Um agradecimento especial a meu tio Antônio e minha avó Marilene, *in memoriam*, por terem me dado a oportunidade de ter crescido sob o olhar deles, oferecendo lembranças que guardarei comigo pelo resto da vida.

Agradeço ao meu orientador Prof. Heitor Savino, que mesmo deixando o cargo formal de professor, continuou me ensinando a ver além, não só através de palavras, mas de atos. Também gostaria de destacar um agradecimento ao meu coorientador Prof. Ícaro Bezerra, por me ensinar com sua forma de resolver as coisas e fornecer um ambiente de trabalho completamente compatível com meu estilo, sem perder a oportunidade de me corrigir no que for necessário. Deixo meu profundo agradecimento ao restante do corpo docente do Instituto de Computação, em especial aos professores do seguimento de Engenharia, Prof. Davi Bibiano, Prof. Thiago Cordeiro e Prof. Tiago Vieira, pois foram pessoas que sempre se mostraram acessíveis, além de confiarem em mim para trabalhos que, por vezes, nem eu acreditaria que seria capaz.

Gostaria de repetir um agradecimento que fiz no meu trabalho de conclusão de curso de graduação para pessoas que não deixaram de estar comigo desde aquele tempo, começando como parceiros de aulas e construindo uma amizade, participando de aventuras, projetos e metas comigo, são eles Andressa Martins, Arthur Vangasse, Bruno Georgevich, Eduardo Miranda, Gregory Albertt, Luís Felipe e Maria Júlia.

Muitas pessoas estão nas entrelinhas deste trabalho, até o produto final que resultou neste texto, contribuindo indiretamente para o desenvolvimento das técnicas aqui abordadas. Professores e autores que trouxeram clareza ao desafio que foi pretendido encarar. Escritores, artistas e músicos que, sem saber, me ajudaram e me inspiraram a fazer o que foi feito.

O presente trabalho é parte de um projeto apoiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 88887.469533/2019-00 e 88887.478167/2020-00 – Programa CAPES - Entre Mares.

25 de Janeiro de 2022, Maceió - AL

*Now let me go (now let me go)  
Away across the sea (now let me go)  
The waves can't be as high  
As they pretend to be.*

*Silence And Distance, Angra*

# Resumo

Monitorar o oceano de forma eficiente é essencial para planejar estratégias que permitem a manutenção da saúde do litoral. Isso fica evidente em cenários de desastre ambiental, como o derramamento de óleo reportado em 2019 que atingiu uma grande extensão da costa nordestina brasileira, trazendo consequências tanto ambientais quanto socioeconômicas para os locais afetados. O evento mostrou a necessidade de expansão da rede de monitoramento nacional, composta principalmente por boias marítimas, que são componentes estáticos ou passivos, nessa tarefa. Com base nesta problemática, este trabalho propõe um *framework* para sistema de monitoramento persistente com sensoriamento ativo, a partir de veículos autônomos colaborativos envolvidos em uma missão marítima. Esse sistema é capaz de gerenciar simulações de um processo de dispersão e sincronizar agentes em uma missão, trabalhando em uma política de patrulhamento.

***Palavras-chave:*** Monitoramento ambiental; Robôs cooperativos; Resposta a derramamento de óleo; Busca e rastreamento; Sensoriamento remoto.

# Abstract

Efficiently monitoring the ocean is essential to plan strategies that enable the health of the coast to be maintained. This is evident in environmental disaster scenarios, such as the oil spill reported in 2019 over a large stretch of the northeastern Brazilian coast, bringing environmental and socioeconomic consequences to the affected locations. The event showed the need to expand the national monitoring network, composed mainly of marine buoys, which are static or passive components in this task. Based on this problem, this work proposes a framework for a persistent monitoring system with active sensing from autonomous collaborative vehicles involved in a maritime mission. This system manages simulations of a dispersion process and synchronizes agents on a mission, working on a patrol policy.

***Keywords:*** *Environmental Monitoring; Cooperative Robots; Oil spill response; Search and tracking; Remote sensing.*

# Lista de Figuras

1.1	Exemplos de danos pela presença do óleo na (a) fauna marinha e (b) ecossistemas importantes, como os manguezais. Fonte: BBC/Victor Uchôa . . .	2
1.2	Mancha de óleo detectada em Maragogi, Alagoas. Fonte: REUTERS/Diego Nigro . . . . .	3
2.1	Variáveis usadas para interpolação bilinear da velocidade do vento na direção $x$ . . . . .	11
2.2	Aplicação de interpolação bilinear para variável de velocidade de correntes marítimas em parte da costa de Alagoas. . . . .	12
2.3	Evolução da simulação da equação de advecção e difusão com quatro fontes de derramamento contínuo de óleo durante 1 dia. . . . .	13
2.4	Evolução da simulação usando modelo do GNOME com quatro fontes de derramamento contínuo de óleo durante 1 dia. . . . .	15
2.5	Tratamento das partículas usando (a) histograma 2D e (b) <i>Kernel Density Estimation</i> . . . . .	16
2.6	Visualização de segmento com levantamento de ISL e alguns dos atributos considerados. . . . .	17
3.1	Taxonomia para problemas de <i>target management</i> utilizando múltiplos robôs, segundo Robin and Lacroix (2016) . . . . .	19
3.2	Perfil de caminho gerado pela política reativa (a) sem considerar e (b) considerando menor deslocamento da frente do robô. . . . .	20
3.3	Região exemplo para aplicação da política reativa adaptada, em forma (a) poligonal e (b) discretizada em mapa de grade. . . . .	21
3.4	Aplicação da estratégia de patrulhamento reativa adaptada com 7 agentes. As imagens (a) e (d) mostram também o caminho percorrido pelos agentes, enquanto as imagens (b) e (c), apenas as posições atuais. . . . .	22
4.1	Arquitetura para sistema de monitoramento persistente. . . . .	23
4.2	Exemplos de regiões de interesse para simulação e missão. . . . .	24
4.3	Exemplos de mapas de (a) sensibilidade ambiental e de (b) concentração aplicados em região de interesse da missão. . . . .	26

4.4	Esquema de funcionamento local de veículo envolvido na missão. . . . .	27
5.1	Regiões de (a) simulação e (b) missão no cenário 1, com foco em área de Maragogi-AL. . . . .	29
5.2	Estado inicial da área de missão no (a) mapa de partículas e (b) mapa de concentração. . . . .	30
5.3	Regiões de (a) simulação e (b) missão no cenário 2, com foco em área de Natal-RN. . . . .	31
5.4	Estado inicial da área de missão no (a) mapa de partículas e (b) mapa de concentração. . . . .	32
5.5	Estado final da área de missão no (a) estudo de caso I e no (b) estudo de caso II. . . . .	34
5.6	Estado final da área de missão no (a) estudo de caso III e no (b) estudo de caso IV. . . . .	35
5.7	Estado final da área de missão no estudo de caso V. . . . .	36

# Lista de Tabelas

5.1	Resumo da configuração dos estudos de caso . . . . .	33
5.2	Parâmetros usados pela política reativa nos estudos de caso I e II . . . . .	33
5.3	Parâmetros usados pela política reativa nos estudos de caso III e IV . . . . .	34
5.4	Parâmetros usados pela política reativa no estudo de caso V . . . . .	35

# Lista de Símbolos

- $\mathbb{R}$  Conjunto dos números reais.
- $\mathbf{x}$  Vetor  $x$  (escrito em minúsculo).
- $\mathbf{X}$  Matriz  $X$  (escrito em maiúsculo).
- $\mathbf{X}^\top$  Transposta da matriz  $X$ .
- $\mathbf{X}^{-1}$  Inversa da matriz  $x$ .
- $\mathbf{diag}(x_1, x_2, \dots, x_n)$  Matriz diagonal  $n$  elementos em sua diagonal principal.
- $c(\mathbf{X}, t)$  Concentração de óleo na posição  $\mathbf{X}$  e tempo  $t$ .
- $k_i$  Coeficiente de difusão.
- $T_s$  Passo temporal de simulação.
- $\delta x$  Passo espacial de simulação na coordenada  $x$ .
- $\delta y$  Passo espacial de simulação na coordenada  $y$ .
- $\partial\Omega$  Borda do contorno definido por  $\Omega$
- $\frac{\partial f}{\partial x}$  Derivada parcial de primeira ordem de  $f$  em relação a  $x$ .
- $\frac{\partial^2 f}{\partial x^2}$  Derivada parcial de segunda ordem de  $f$  em relação a  $x$ .
- $\nabla$  Vetor gradiente composto pelas derivadas parciais.
- $\hat{c}_{\mathbf{kde}}(\mathbf{x})$  Estimativa de concentração usando KDE.
- $K(\cdot)$  Função de densidade de probabilidade usada como *Kernel* do KDE.
- $\kappa$  Ganho da política reativa.
- $c_j$  Concentração do poluente na  $j$ -ésima célula na política reativa.
- $s_j$  Sensibilidade ambiental na  $j$ -ésima célula na política reativa.

$\delta_{ij}$  Distância do  $i$ -ésimo agente à  $j$ -ésima célula na política reativa.

$\max\{x, y\}$  Função máximo entre  $x$  e  $y$ .

$\min_{i \neq j} f_i$  Menor valor de  $f_i$ , excluindo  $f_j$ .

# Lista de Abreviaturas

- API** *Application Programming Interface*
- GOOS** *Global Ocean Observing System*
- GFS** *Global Forecasting System*
- GNOME** *General NOAA Operational Modeling Environment*
- HTTP** *Hypertext Transfer Protocol*
- HYCOM** *HYbrid Coordinate Ocean Model*
- ITOPF** *International Tanker Owners Pollution Federation*
- ISL** *Índice de Sensibilidade do Litoral*
- KDE** *Kernel Density Estimation*
- KML** *Keyhole Markup Language*
- NetCDF** *Network Common Data Form*
- NOAA** *National Oceanic and Atmospheric Administration*
- PIRATA** *Prediction and Research Moored Array in the Tropical Atlantic*
- PNBOIA** *Programa Nacional de Bóias Marítimas*
- PIP** *Point-In-Polygon*
- REST** *Representational State Transfer*
- UAV** *Unmanned Aerial Vehicle*

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Acidente no Brasil . . . . .	3
1.2	Monitoramento por veículos autônomos . . . . .	4
1.3	Justificativa . . . . .	5
1.4	Objetivos . . . . .	6
1.4.1	Objetivo Geral . . . . .	6
1.4.2	Objetivos Específicos . . . . .	6
1.5	Organização do Trabalho . . . . .	7
<b>2</b>	<b>Dinâmica do óleo</b>	<b>8</b>
2.1	Equações de advecção e difusão . . . . .	8
2.1.1	Método das diferenças finitas . . . . .	9
2.1.2	Interpolação bilinear . . . . .	11
2.1.3	Simulação de óleo . . . . .	12
2.2	GNOME . . . . .	13
2.2.1	KDE . . . . .	14
2.3	ISL . . . . .	16
<b>3</b>	<b>Monitoramento Persistente</b>	<b>18</b>
3.1	Política reativa . . . . .	18
3.1.1	Cenário multiagente . . . . .	20
3.2	Política reativa adaptada . . . . .	20
<b>4</b>	<b>Arquitetura do framework</b>	<b>23</b>
4.1	Simulação . . . . .	24
4.2	Missão . . . . .	26
<b>5</b>	<b>Experimentos</b>	<b>28</b>
5.1	Funcionamento do sistema supervisor . . . . .	28
5.1.1	Cenário 1 . . . . .	29
5.1.2	Cenário 2 . . . . .	30
5.2	Supervisor e política implementada . . . . .	31

5.2.1	Estudos de caso I e II . . . . .	32
5.2.2	Estudos de casos III e IV . . . . .	33
5.2.3	Estudo de caso V . . . . .	35
	<b>Conclusão</b>	<b>37</b>
	<b>Bibliografia</b>	<b>38</b>

# Capítulo 1

## Introdução

Um acidente de vazamento de óleo é um evento que acontece quando produtos relacionados a óleo, em sua forma bruta ou refinada, são despejados em larga escala no ambiente em forma de poluente, na maior parte das vezes em cenários marítimos. Segundo De la Huz et al. (2011), conhecer a composição química do poluente, seu comportamento, a área afetada e metodologias de respostas aplicáveis, são características que têm papel importante na extensão do dano resultante causado pelo desastre.

A origem de um acidente de vazamento de óleo pode variar. Ela pode ser proveniente de causas naturais, como erosão de rochas sedimentadas no fundo do mar e transtornos climáticos. Óleo vindo debaixo do solo oceânico pode escapar para a superfície através de falhas associadas ao processo de diapirismo entre as rochas e o sal (Anderson et al., 1983). Quando despejado a um ritmo devagar, vazamentos naturais permitem que o ecossistema se adapte, mas é fato que todo vazamento de óleo causa impacto no ambiente marítimo (NOAA, 2019).

No entanto, a maioria dos impactos acontecem por ação humana, uma vez que o óleo é um componente vital e altamente demandado pela sociedade. De acordo com a *U.S. Information Administration*<sup>1</sup>, o consumo global de petróleo é de mais de 100 milhões de barris, demandando maior produção, exploração e transporte do recurso. Cada um desses processos tem chance de causar acidentes de vazamento.

As causas mais usuais de acidentes são relacionadas a situações de colisão, encalhe (impacto do navio no solo marítimo), falha de equipamentos durante o transporte, perfuração *offshore* e outras atividades relacionadas a petróleo. Em geral, esses desastres são acompanhados de uma série de consequências negativas para o meio ambiente, economia e a sociedade. Sistemas importantes, como manguezais, pântanos e recifes de coral são altamente afetados pelo poluente pois, uma vez depositado, ele se adere a estas superfícies e raramente se move, arriscando perda permanente dessas estruturas (Guzmán et al., 1994; Duke, 2016). A existência do poluente é prejudicial em qualquer cenário, não obstante,

---

<sup>1</sup>Disponível em [https://www.eia.gov/outlooks/steo/report/global\\_oil.php](https://www.eia.gov/outlooks/steo/report/global_oil.php)



(a)



(b)

Figura 1.1: Exemplos de danos pela presença do óleo na (a) fauna marinha e (b) ecossistemas importantes, como os manguezais. Fonte: BBC/Victor Uchôa

ela apresenta maiores riscos nos sistemas costeiros do que em alto-mar.

A presença do poluente causa perturbação em populações de plantas e põe em risco a fauna marinha, desde comunidades microbianas do fundo do mar a peixes, tartarugas e aves marinhas (Bik et al., 2012; Putman et al., 2015; Beyer et al., 2016), como mostrado na Figura 1.1.

Com relação aos impactos socioeconômicos, serviços relacionados ao mar, como turismo e a indústria alimentícia são gravemente afetados, considerando o nível de relevância dessas atividades na economia local e na vida dos habitantes costeiros (García Negro et al., 2009). Além disso, quando chega a rios, o óleo pode contaminar nascentes, ameaçando a saúde humana e o meio ambiente.

Diversas estratégias podem ser empregadas como contramedida a tais desastres, como controlar a fonte de derramamento, utilização de equipamentos mecânicos e dispersantes químicos especializados para conter, tratar ou remover o poluente. Entretanto, para a aplicação eficiente das demais, a primeira estratégia de resposta a ser empregada é a supervisão e monitoramento de óleo derramado, o que permite observar e prever o movimento do poluente (Walker, 2017).

A integração de tecnologias de sensoriamento remoto com análises de previsão climática, condições oceânicas e modelagem dinâmica de processos de dispersão de óleo é um fator crucial no processo de tomada de decisões (Ornitz and Champ, 2002). Além de auxiliar contramedidas estratégicas, a aquisição de dados do derramamento pode ser importante para adquirir evidência legal para aplicação da lei (Fingas and Brown, 2018). Portanto, o desenvolvimento de tecnologias eficientes para o monitoramento do mar é fundamental para vários setores da sociedade.



Figura 1.2: Mancha de óleo detectada em Maragogi, Alagoas. Fonte: REUTERS/Diego Nigro

## 1.1 Acidente no Brasil

Em 2019, um grande derramamento de óleo aconteceu na costa brasileira, afetando ao menos 900 localidades <sup>2</sup>. Dentre as regiões afetadas, zonas de proteção ambiental, como a Costa dos Corais, também estão entre os locais atingidos, como mostrado na Figura 1.2, As primeiras manchas foram noticiadas entre meados de agosto e começo de setembro na região nordeste, aparecendo outras manchas no decorrer dos meses, com um pico de 536 ocorrências em novembro. O poluente reapareceu em praias por vários vezes, por conta das mudanças na dinâmica das marés relacionadas à habilidade de mover sedimentos presos em rochas, o que demandava esforços de limpeza urgentes<sup>3</sup>.

Quase um ano depois da detecção da primeira mancha, novos fragmentos de óleo provenientes do mesmo derramamento foram encontrados nas costas de Alagoas e Pernambuco<sup>4</sup>. Uma das razões que dificultaram a detecção desse derramamento, acessar a origem e sua escala total, é a densidade do óleo bruto, que frequentemente faz com que o poluente flutue abaixo da superfície da água<sup>5</sup>. Diversas vezes, o óleo só era detectado na superfície quando chegava à margem.

<sup>2</sup><https://g1.globo.com/natureza/desastre-ambiental-petroleo-praias/noticia/2019/12/07/mais-de-900-localidades-foram-atingidas-por-manchas-de-oleo-no-litoral-brasileiro-diz-ibama.ghtml>. Acessado em 20 de Março de 2020.

<sup>3</sup><https://www.redebrasilatual.com.br/ambiente/2020/03/quase-sete-meses-depois-oleo-ainda-mancha>. Acessado em 20 de março de 2020.

<sup>4</sup><https://g1.globo.com/al/alagoas/noticia/2020/06/25/analise-confirma-que-oleo-que-reapareceu-em>. Acessado em 26 de junho de 2020 Accessed on June 26th, 2020.

<sup>5</sup><https://www.reuters.com/article/us-brazil-environment-oil-explainer/explainer-the-mysterious-oil-washing-up-on-brazilian-shores-idUSKBN1X421J>. Acessado em 20 de março de 2020.

Outro incidente, de proporções relativamente menores, ocorreu em 2020, quando 333 litros de óleo residual foi encontrado ao redor de um cargueiro encalhado a 100 quilômetros da costa brasileira. Apesar dos tanques carregando cerca de 4000 toneladas estarem intactos, a possibilidade de vazamento estava sendo considerada<sup>6</sup>. Esses relatos são importantes para mostrar que eventos de derramamento de óleo podem ser súbitos, desastrosos e recorrentes, demandando estratégias de respostas rápidas e eficientes

A aquisição de dados para monitoramento oceanográfico no Brasil é operado pelo sistema nacional chamado GOOS-Brasil, parte da aliança regional OCEATLAN GOOS. Ela provê dados oceanográficos para pesquisas e outros propósitos práticos de grande importância, como previsão climática e avaliação da saúde costeira e oceânica. Para monitorar um conjunto de variáveis de processo no oceano atlântico tropical, uma rede de observação *in-situ* feita por 21 bóias marítimas fixas é mantida pelo programa PIRATA (do inglês, *Prediction and Research Moored Array in the Tropical Atlantic*), através de uma cooperação multinacional entre Brasil, França e Estados Unidos. Nessa mesma linha, o Programa Nacional de Bóias Marítimas (PNBOIA) implementa um sistema com 22 bóias fixas e 297 bóias à deriva.

Apesar do sistema de monitoramento existente, a detecção da mancha de óleo foi problemática. Na maior parte das vezes as manchas de óleo só eram detectadas quando já atingiam as praias, dificultando as tarefas de planejamento, demandando trabalhos de limpeza urgentes, sem a elaboração de estratégias de respostas prévias otimizadas ou eficientes. Uma expansão dessa rede de aquisição de dados oceânicos se torna essencial para garantir a saúde climática e marítima nacional.

## 1.2 Monitoramento por veículos autônomos

Estratégias de sensoriamento remoto vem sendo empregadas ao redor do globo, como radares de alta frequência, imagens de satélite e radares de abertura sintética (SAR). Observações no local são importantes, pois sensores ópticos como câmeras operando na faixa de luz visível, infravermelho e ultravioleta podem ser instalados, da mesma forma que sensores mais específicos, como fluorosensores a laser. Isso permite que um melhor desempenho na detecção do poluente, de acordo com as características e disposição da pluma química no ambiente. Interferências em sensores ultravioleta são diferentes das apresentadas em sensores infravermelho, de forma que uma combinação das duas tecnologias melhora os resultados da detecção de óleo (Goodman, 1994). Fluorosensores a laser podem ser usados tanto de dia quanto de noite, sendo capazes de distinguir a assinatura de emissão de fluorescência de diferentes tipos de óleo (Brown et al., 2004).

Para observação *in-situ*, podem ser usados veículos aéreos comportando sensores para

---

<sup>6</sup><https://g1.globo.com/ma/maranhao/noticia/2020/02/28/ibama-encontra-333-litros-de-oleo-notificacao.html>. Acessado em 20 de março de 2020.

varredura de grandes áreas. Entretanto, a aplicação de veículos aéreos e de superfície autônomos em tarefas de monitoramento e rastreamento, trabalhando de forma coordenada, tem atraído grande interesse, pois possibilitam acompanhar em tempo real situações de desastre ambiental por óleo ou outro poluente de forma eficiente e adaptativa. Dessa forma, tem-se as vantagens da detecção do óleo *in-situ*, com a possibilidade de detectar a mancha ainda em alto mar, onde apresenta menor risco e permite antecipar estratégias de contenção, coleta ou dispersão.

A aplicação de robôs móveis executando atividades como busca distribuída em mapa, localização de fonte de vazamento, seguimento de rastros de pluma química, entre outras é revisada em Ishida et al. (2012). Usando um robô de superfície e um subaquático trabalhando colaborativamente, Vasilijevic et al. (2017) apresentou um sistema de amostragem oceânica com abordagem *human-on-the-loop* para coordenação de missões de monitoramento ambiental. Seguindo com um objetivo similar, Wang et al. (2019) propôs uma estratégia de controle cooperativo para dois veículos autônomos com o objetivo de rastrear a propagação das bordas de uma pluma de poluente enquanto a mesma evoluía dinamicamente, desenvolvendo um observador através das equações diferenciais do processo.

Considerando os estudos acima descritos e a aplicabilidade dessas abordagens em um cenário como o do acidente ocorrido na costa brasileira em 2019, este trabalho apresenta uma estrutura computacional genérica em forma de *framework*. Técnicas de navegação em veículos autônomos existentes podem ser testadas ou novas técnicas podem ser desenvolvidas com a ajuda desse arcabouço para simular ou supervisionar atividades como busca, patrulhamento, observação ou rastreamento de óleo.

### 1.3 Justificativa

A Federação Internacional de Poluição por Petroleiros (do inglês *International Tanker Owners Pollution Federation*, ITOPF) relata mais de 1800 derramamentos, despejando cerca de 5,86 milhões de toneladas como resultado de desastres com petroleiros desde 1970 em nível global. Apesar do número de grandes acidentes de derramamento de óleo ter reduzido nos últimos anos, a ocorrência de apenas um grande evento implica em resultados catastróficos para a sociedade e o meio ambiente.

Com relação ao desastre que atingiu o litoral nordeste do Brasil em 2019, pôde ser observada ausência de respostas concretas e imediatas sobre o derramamento em questão, de forma que houve demora para detecção e conseqüente planejamento de ações de resposta. A ineficiência no monitoramento e previsão da evolução do desastre mostraram a necessidade de investimento em expansão da rede de supervisão marítima, como a utilização de tecnologias de monitoramento persistente.

A construção de um *framework* para sistema de monitoramento persistente, supor-

tando veículos autônomos trabalhando de forma colaborativa, provê uma estrutura que permitirá supervisionar áreas com maior probabilidade de concentração do poluente, priorizar visitas dos agentes em áreas mais sensíveis e tratar o recebimento de informações adquiridas *in situ* pelos veículos envolvidos na missão. Assim, o sistema desempenhará um papel ativo como estratégia de resposta a desastres ambientais marítimos, inclusive permitindo melhor planejamento de outras ações nesse contexto.

## 1.4 Objetivos

Os objetivos a seguir estabelecem as intenções e delimitações do trabalho, considerando que, por se tratar da proposta de uma arquitetura modular, algumas partes foram desenvolvidas como prova de funcionamento, mas a mudança, por exemplo, de técnica de patrulhamento ou simulador externo é possível.

### 1.4.1 Objetivo Geral

Desenvolver um *framework* para implantação de sistemas de monitoramento marítimo usando veículos autônomos em cenários de derramamento de óleo. O *framework* será responsável por gerenciar a simulação da dinâmica do processo de dispersão da substância no mar, fornecer dados do estado dos componentes de simulação e missão para veículos e supervisores, e prover interface para lidar com *feedback* dos agentes envolvidos na missão.

### 1.4.2 Objetivos Específicos

- Proporcionar um serviço persistente de simulação de processos de dispersão de óleo através de *web service*;
- Estabelecer campos potenciais de sensibilidade ambiental a partir do Índice de Sensibilidade do Litoral (ISL);
- Adquirir, de forma recorrente, previsões climáticas para a simulação;
- Implementar interface de configuração de simulação e missão;
- Disponibilizar um mecanismo de requisições *HTTP* (em formato *RESTful API*) para acessar e modificar variáveis de simulação e missão;
- Adaptar política reativa de patrulhamento para o contexto de ambientes variantes no tempo, com áreas de sensibilidade e região de interesse não-quadrada.

## 1.5 Organização do Trabalho

O primeiro capítulo deste trabalho serviu para introduzir o tema a ser dissertado, explicitando histórico de acidentes, estratégias de resposta, trabalhos relacionadas, relevância e objetivos. O Capítulo 2 discute como modelar e simular a dispersão do poluente, no caso óleo bruto, usando abordagens que podem ser utilizadas no *framework* desenvolvido, além de determinar quais variáveis são importantes para possibilitar simular esse processo. O Capítulo 3 abrange técnicas de monitoramento persistente, descrevendo a área, uma técnica de aplicação e uma adaptação dessa técnica para o problema deste trabalho. O Capítulo 4 aborda a arquitetura do *framework*, onde os temas discutidos anteriormente são encapsulados e as tecnologias envolvidas. O Capítulo 5 apresenta a definição de experimentos realizados para testar a ferramenta e discute os resultados obtidos. O Capítulo 6 fecha este trabalho trazendo uma conclusão para o que foi debatido e fornecendo uma visão de trabalhos futuros.

# Capítulo 2

## Dinâmica do óleo

O comportamento de óleo sob água pode ser modelado de acordo com as características que se deseja observar, de forma que esse comportamento pode ser dividido em processos. Esses processos representam as transformações físicas de massa, concentração e propriedades químicas que o poluente vai sofrer no decorrer do tempo e do espaço em que se localiza. De acordo com NOAA (2019), alguns exemplos de processos que afetam o derramamento de óleo são evaporação, emulsificação, sedimentação, dispersão, dissolução, entre outros.

Nas áreas costeiras, onde o transporte de poluentes acontece em águas relativamente rasas, a interação apresentada em fenômenos de advecção, difusão, evaporação, entre outros, é fortemente relacionada às variáveis meteorológicas, principalmente o vento e correntes marítimas superficiais (Fingas, 2011; Proctor et al., 1994). Portanto, um sistema de monitoramento eficiente deve incorporar simulações de probabilidade da concentração de óleo, adquirir constantemente previsões climáticas e prestar interface para algoritmos capazes de trabalhar com ambientes que mudam com o tempo.

Neste trabalho, o processo de dispersão foi priorizado por ser parte importante do grupo de processos que focam em descrever a movimentação do óleo no ambiente (Fingas and Brown, 2018). Duas abordagens para a modelagem dessa dispersão foram consideradas, uma a partir das equações diferenciais parciais dos fenômenos de advecção e difusão, e outra abordagem lagrangiana, onde partículas de óleo interagem com a simulação computacional.

### 2.1 Equações de advecção e difusão

Considerando um contorno  $\Omega$ , onde o fluido modelado deverá se movimentar, os fenômenos de advecção e difusão podem ser derivados da equação de continuidade de Navier-Stokes aplicada à concentração do óleo  $c$  em uma posição  $\mathbf{X}$  e tempo  $t$  (Li et al., 2014). A equação 2.1 mostra essa relação, considerando ainda a velocidade  $v$  que o fluido está submetido e um coeficiente de difusão  $k$ .

$$\begin{aligned} \frac{\partial c(\mathbf{X}, t)}{\partial t} + \mathbf{v}(\mathbf{X}, t)^\top \nabla c(\mathbf{X}, t) &= k \nabla^2 c(\mathbf{X}, t) \\ \mathbf{X} \in \Omega \subset \mathbb{R}^n \end{aligned} \quad (2.1)$$

A equação 2.1 pode ser dividida em três componentes, a primeira é a variação da concentração no tempo, que poderá assumir valores diferentes de zero até o processo de dispersão chegar em regime permanente, quando essa parcela é anulada. A componente de advecção,  $\mathbf{v}(\mathbf{X}, t)^\top \nabla c(\mathbf{X}, t)$ , é responsável pelo movimento da substância por conta da dinâmica do fluido. Já a componente de difusão,  $k \nabla^2 c(\mathbf{X}, t)$ , provê um comportamento suave que expande a concentração no espaço.

Considerando que simulações de óleo podem trabalhar em 2 ou 3 dimensões, Wang et al. (2019) adaptou a componente de difusão para forma quadrática, com coeficientes de difusão para cada dimensão espacial. O resultado dessa adaptação pode ser observado na equação 2.2.

$$\begin{aligned} \frac{\partial c(\mathbf{X}, t)}{\partial t} + \mathbf{v}(\mathbf{X}, t)^\top \nabla c(\mathbf{X}, t) &= \nabla^\top D \nabla c(\mathbf{X}, t) \\ D &= \text{diag}(k_1, \dots, k_n) \\ \mathbf{X} \in \Omega \subset \mathbb{R}^n, \quad n &= \{2, 3\} \end{aligned} \quad (2.2)$$

Aplicando a equação 2.2 no cenário bidimensional ( $n = 2$ ), podem ser explicitadas as componentes cartesianas de posição  $x$  e  $y$ , tal como para velocidade ( $\mathbf{v}_x$  e  $\mathbf{v}_y$ ). Essa delimitação fornece uma descrição simplificada da equação diferencial parcial mostrada na equação 2.3, lembrando que a concentração  $c$  é uma variável escalar.

$$\begin{aligned} \frac{\partial c(\mathbf{X}, t)}{\partial t} + \mathbf{v}_x(\mathbf{X}, t) \frac{\partial c(\mathbf{X}, t)}{\partial x} + \mathbf{v}_y(\mathbf{X}, t) \frac{\partial c(\mathbf{X}, t)}{\partial y} &= k_1 \frac{\partial^2 c(\mathbf{X}, t)}{\partial x^2} + k_2 \frac{\partial^2 c(\mathbf{X}, t)}{\partial y^2} \\ \mathbf{X} \in \Omega \subset \mathbb{R}^2 \end{aligned} \quad (2.3)$$

### 2.1.1 Método das diferenças finitas

Para desenvolver uma simulação computacional da equação 2.3, é necessário realizar um tratamento numérico em forma de discretização. O método de diferenças finitas tem sido usado na simulação de fluidos para resolver equações de Navier-Stokes (Griebel et al., 1998), como uma alternativa discreta ao método de elementos finitos. Kajishima and Taira (2017) explorou essa abordagem especificamente na implementação de equações de advecção-difusão em uma região retangular.

Em duas dimensões, o método das diferenças finitas pode aplicar a discretização a

partir o conceito de diferenças centrais. Dessa forma, dados os parâmetros de passo de amostragem temporal  $T_s$  e espacial  $\delta x$  e  $\delta y$ , as derivadas podem ser reescritas da seguinte forma:

$$\frac{\partial c(\mathbf{X}, t)}{\partial t} \rightarrow \frac{c(\mathbf{X}, t + T_s) - c(\mathbf{X}, t)}{T_s} \quad (2.4a)$$

$$\frac{\partial c(\mathbf{X}, t)}{\partial x} \rightarrow \frac{c(x + \delta x, y, t) - c(x - \delta x, y, t)}{2\delta x} \quad (2.4b)$$

$$\frac{\partial c(\mathbf{X}, t)}{\partial y} \rightarrow \frac{c(x, y + \delta y, t) - c(x, y - \delta y, t)}{2\delta y} \quad (2.4c)$$

$$\frac{\partial^2 c(\mathbf{X}, t)}{\partial x^2} \rightarrow \frac{c(x + \delta x, y, t) + 2c(x, y, t) - c(x - \delta x, y, t)}{\delta x^2} \quad (2.4d)$$

$$\frac{\partial^2 c(\mathbf{X}, t)}{\partial y^2} \rightarrow \frac{c(x, y + \delta y, t) + 2c(x, y, t) - c(x, y - \delta y, t)}{\delta y^2} \quad (2.4e)$$

Com isso, o passo de simulação pode ser definido aplicando as equações 2.4 de diferenças centrais à equação 2.3. São definidas condições iniciais de concentração para  $t_0$  e condições de contorno de Dirichlet para delimitar a dispersão do óleo às regiões com água. Essas considerações podem ser resumidas no conjunto de equações 2.5.

$$\begin{aligned} \text{adv} = & \mathbf{v}_x(x, y, t) \left( \frac{c(x + \delta x, y, t) - c(x - \delta x, y, t)}{2\delta x} \right) + \\ & + \mathbf{v}_y(x, y, t) \left( \frac{c(x, y + \delta y, t) - c(x, y - \delta y, t)}{2\delta y} \right) \end{aligned} \quad (2.5a)$$

$$\begin{aligned} \text{diff} = & k_1 \left( \frac{c(x + \delta x, y, t) + 2c(x, y, t) - c(x - \delta x, y, t)}{\delta x^2} \right) + \\ & + k_2 \left( \frac{c(x, y + \delta y, t) + 2c(x, y, t) - c(x, y - \delta y, t)}{\delta y^2} \right) \end{aligned} \quad (2.5b)$$

$$c(x, y, t + T_s) = c(x, y, t) + T_s(\text{diff} - \text{adv}) \quad (2.5c)$$

$$c(\mathbf{X}, t)|_{\mathbf{X} \in \partial\Omega} = 0, \quad t \geq t_0 \quad (2.5d)$$

$$c(\mathbf{X}, t_0) = c_0(\mathbf{X}) \quad (2.5e)$$

Fontes de derramamento de óleo podem ser definidas em posições determinadas  $\mathbf{X}_s$ , de forma que a concentração nesses locais tenha valor máximo  $c_{max}$ . A simulação pode restringir esse vazamento a um momento inicial, um momento específico ou mesmo um vazamento contínuo.

## 2.1.2 Interpolação bilinear

Correntes marítimas e ventos são fatores climáticos que influenciam na movimentação do óleo, sendo importantes na determinação da velocidade a qual o poluente vai estar submetido no fenômeno de advecção, presente na equação 2.5a. De fato, próximo da costa, com ventos abaixo de 10 km/h, a velocidade que a pluma de óleo se move está relacionada a 100% da velocidade de corrente marítima e 3% da velocidade do vento, havendo, portanto, predominância da influência da corrente. Em alto mar, com ventos atingindo velocidades próximas de 20 km/h, o vento passa a ser fator predominante no movimento da pluma (Fingas and Brown, 2018).

O problema é que leituras de ventos e correntes fornecidas por repositórios de monitoramento de variáveis climáticas, como GFS (*Global Forecast System*) e HYCOM, oferecem resolução baixa, no máximo 0,08 graus ou 8,8 quilômetros, para os propósitos deste trabalho. Para que a simulação seja aplicável em cenário de busca de um veículo de monitoramento, pretende-se realizar simulação com células de discretização de cerca de 500 metros a 1 quilômetro.

Nesse cenário bidimensional, a interpolação bilinear (Press et al., 2007) possibilita estimar valores intermediários desconhecidos para os campos vetoriais, já desacoplados na direção  $x$  e  $y$ , da velocidade do vento ( $\mathbf{v}_{\text{air},x}$  e  $\mathbf{v}_{\text{air},y}$ ) e das correntes marítimas ( $\mathbf{v}_{\text{cur},x}$  e  $\mathbf{v}_{\text{cur},y}$ ). A figura 2.1 mostra um exemplo com as variáveis necessárias para realizar a estimativa de  $\mathbf{v}_{\text{air},x}$  em uma localização arbitrária.

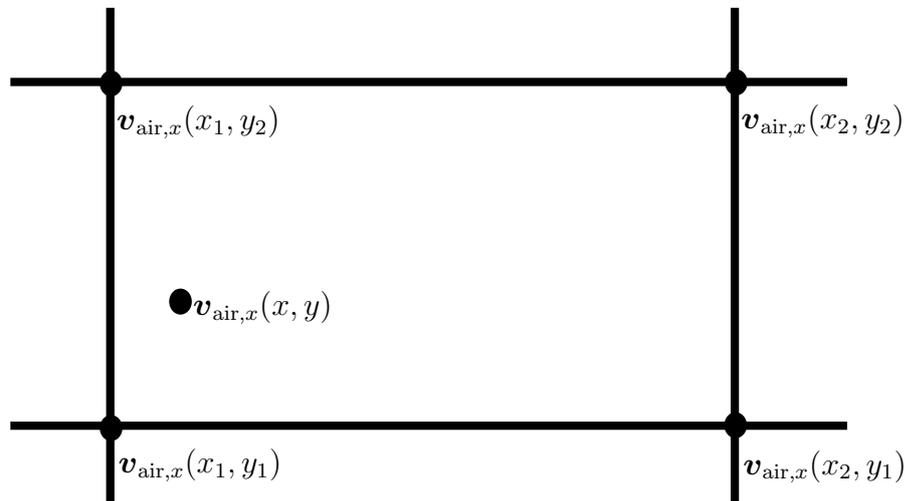


Figura 2.1: Variáveis usadas para interpolação bilinear da velocidade do vento na direção  $x$ .

Seguindo o exemplo da figura 2.1, o valor de  $\mathbf{v}_{\text{air},x}(x, y)$  é calculado a partir da equação 2.6, usando dados das leituras mais próximas de velocidade do vento na direção  $x$ . O procedimento é análogo para  $\mathbf{v}_{\text{air},y}$ ,  $\mathbf{v}_{\text{cur},x}$  e  $\mathbf{v}_{\text{cur},y}$ .

$$\mathbf{v}_{\text{air},x}(x, y) = \frac{1}{(x_2 - x_1)(y_2 - y_1)} \begin{bmatrix} x_2 - x & x - x_1 \end{bmatrix} \begin{bmatrix} \mathbf{v}_{\text{air},x}(x_1, y_1) & \mathbf{v}_{\text{air},x}(x_1, y_2) \\ \mathbf{v}_{\text{air},x}(x_2, y_1) & \mathbf{v}_{\text{air},x}(x_2, y_2) \end{bmatrix} \begin{bmatrix} y_2 - y \\ y - y_1 \end{bmatrix} \quad (2.6)$$

A figura 2.2 mostra o resultado do cálculo de interpolação bilinear a partir de leituras de correntes marítimas provenientes do repositório HYCOM, com resolução de 0,08 graus, aplicadas à um mapa discreto com células de 500 metros de altura e largura. O campo vetorial azul mostra vetores reais amostrados do repositório de dados ambientais, com leituras datadas de 03 de Novembro de 2020, às 12 horas, enquanto o campo vetorial vermelho apresenta as estimativas computadas pela técnica de interpolação apresentada.

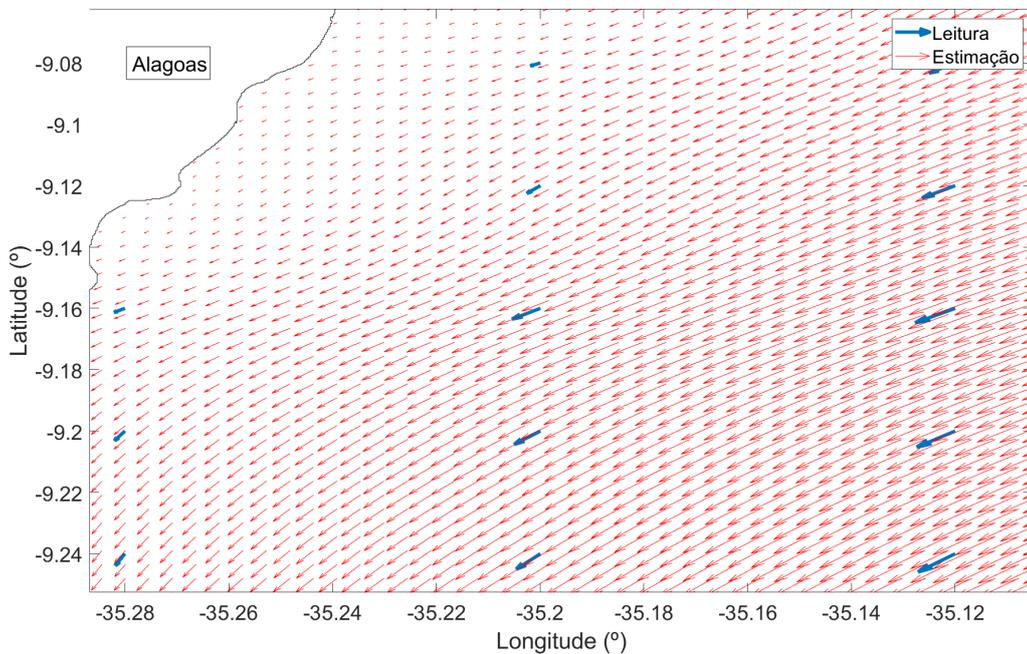


Figura 2.2: Aplicação de interpolação bilinear para variável de velocidade de correntes marítimas em parte da costa de Alagoas.

### 2.1.3 Simulação de óleo

Utilizando esses vetores de velocidade, uma simulação de vazamento de óleo usando as equações de advecção-difusão pelo método das diferenças finitas pode ser gerada. Como cenário de prova de conceito, foi considerado um derramamento persistente de óleo proveniente de quatro fontes geograficamente espaçadas, ocorrendo em um período de 24 horas. Foram ainda configurados passo temporal de simulação  $T_s = 0,01$  segundos, passos espaciais  $\delta x = \delta y = 500$  metros e coeficientes de difusão  $k_1 = k_2 = 2 \text{ m}^2/\text{s}$ . A figura 2.3 mostra a evolução da simulação.

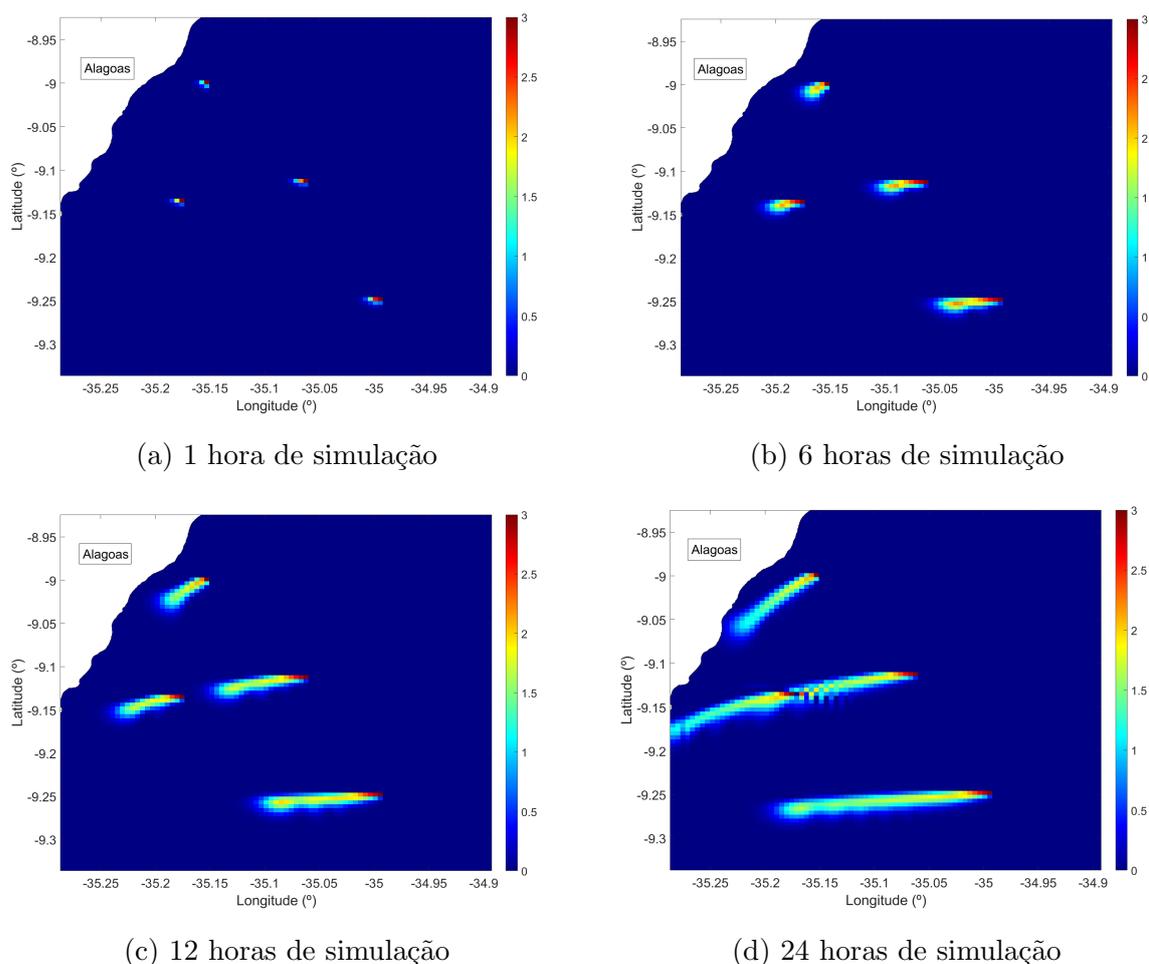


Figura 2.3: Evolução da simulação da equação de advecção e difusão com quatro fontes de derramamento contínuo de óleo durante 1 dia.

Duas fontes foram posicionadas próximo da costa, cerca de 5 km, uma em 35,153W 8,999S e outra em 35,176W 9,135S, mostrando que um derramamento com essas propriedades sob determinadas condições climáticas é capaz de chegar ao litoral no período de um dia. As demais fontes estão localizadas em regiões mais afastadas, uma em 35,062W 9,112S e outra em 34,994W 9,248S, mas dentro da zona contígua do território brasileiro (Brasil, 1993), que se estende a cerca de 38 quilômetros da costa.

## 2.2 GNOME

O GNOME (do inglês, *General NOAA Operational Modeling Environment*) é uma ferramenta de modelagem desenvolvida pelo NOAA (do inglês, *National Oceanic and Atmospheric Administration*), uma instituição governamental estadunidense que trata de assuntos ligados à administração e saúde dos oceanos. O foco do GNOME é modelar diversos aspectos de um derramamento de óleo e estimar sua movimentação. Para isso, a ferramenta é capaz de considerar características como a composição do poluente,

incertezas no processo e influências climáticas.

Para realizar as previsões, o GNOME usa uma abordagem de transporte Lagrangiano, que define a movimentação do derramamento a partir da interação de partículas de óleo. Além da aplicação *standalone* para configuração e visualização de simulações, é fornecido o pyGNOME<sup>1</sup> um conjunto de utilitários e *bindings* para a linguagem Python, que funciona de maneira independente das outras plataformas dos GNOME. A partir do pyGNOME, as funcionalidades dos modelos de previsão podem ser incorporados em ferramentas customizadas.

Por considerar variações em partículas, ao invés de células de discretização, a abordagem oferecida pelo GNOME provê computação do movimento do óleo de forma mais rápida na maioria das situações. Isso fica evidente em grandes mapas, quando o número de células da discretização é bem maior que o número de partículas. Além disso, o GNOME, por ser uma ferramenta já consolidada na área de modelagem e previsão de óleo, é capaz de calcular a influência de outros fenômenos mais facilmente. A evaporação, por exemplo, pode ser simulada como um parâmetro individual de uma partícula, que define seu tempo de vida na simulação.

A figura 2.4 mostra o progresso de uma simulação com GNOME em uma região próxima à da simulação com a técnica de diferenças finitas, entretanto com leituras meteorológicas de outro dia. Também foram consideradas quatro fontes de derramamento contínuo com duração de um dia, em posições similares à simulação anterior. Dessa vez, condições meteorológicas levaram a pluma simulada diretamente para a costa, mostrando uma das características diferenciais do GNOME, a aglutinação de partículas em áreas de terra, permitindo supor áreas que possivelmente seriam afetadas.

Como parâmetros para a simulação, foram considerados um mapa da região, variáveis climáticas de vento e correntes, além de características específicas da substância que compõe o poluente. Cada fonte de derramamento é um objeto do tipo *Spill*, onde são configurados parâmetros como o número de partículas que serão expelidas a cada passo de simulação, distribuição estatística que as partículas devem seguir e propriedades químicas do poluente, que podem ser extraídas de uma biblioteca opcional denominada *Oil Library*<sup>2</sup> com um banco de dados de propriedades de óleo proveniente de vários eventos de derramamentos ocorridos no decorrer da história.

### 2.2.1 KDE

Como o mapa da missão para os algoritmos a serem trabalhados são discretos em uma grade de células, é necessário passar o quantitativo das partículas computadas nos passos da simulação para dados de concentração em células. Considerando que cada partícula tem registro de localização (longitude e latitude), é possível computar uma estimativa

---

<sup>1</sup>Disponível em <https://github.com/NOAA-ORR-ERD/PyGnome>

<sup>2</sup>Disponível em <https://github.com/NOAA-ORR-ERD/OilLibrary>

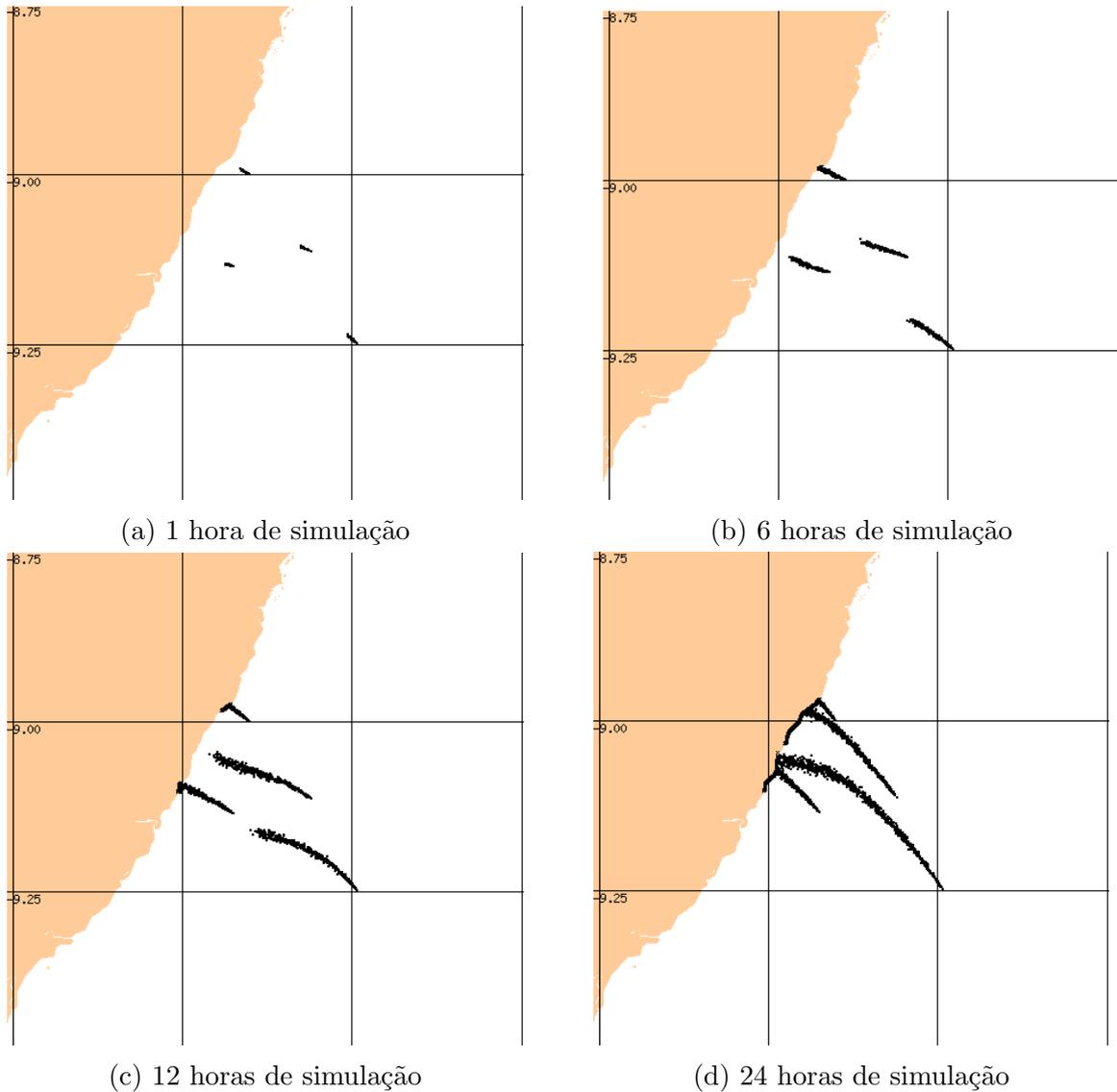


Figura 2.4: Evolução da simulação usando modelo do GNOME com quatro fontes de derramamento contínuo de óleo durante 1 dia.

bidimensional de concentração usando KDE (do inglês, *Kernel Density Estimation*) na região de interesse da missão. Enquanto histogramas simples aglutinam amostras em uma célula, potencialmente deixando espaços vazios em 2D e maiores dimensões, o KDE aplica um *kernel* centralizada em cada amostra, representando melhor uma distribuição probabilística real na região.

A primeira abordagem de KDE focava em análise univariada Parzen (1962), sendo posteriormente expandida para multivariada. A equação 2.7, adaptada de Scott (2015), representa a função de distribuição de uma amostra usando KDE. No caso,  $\hat{c}_{\text{kde}}(\mathbf{x})$  é a estimativa de concentração no plano.

$$\hat{c}_{\text{kde}}(\mathbf{x}) = \frac{1}{n|\mathbf{H}|} \sum_{i=1}^n K(\mathbf{H}^{-1}(\mathbf{x} - \mathbf{x}_i)) \quad (2.7)$$

Assume-se que a variável independente  $\mathbf{x} = [x \ y]^T$  está identicamente distribuída

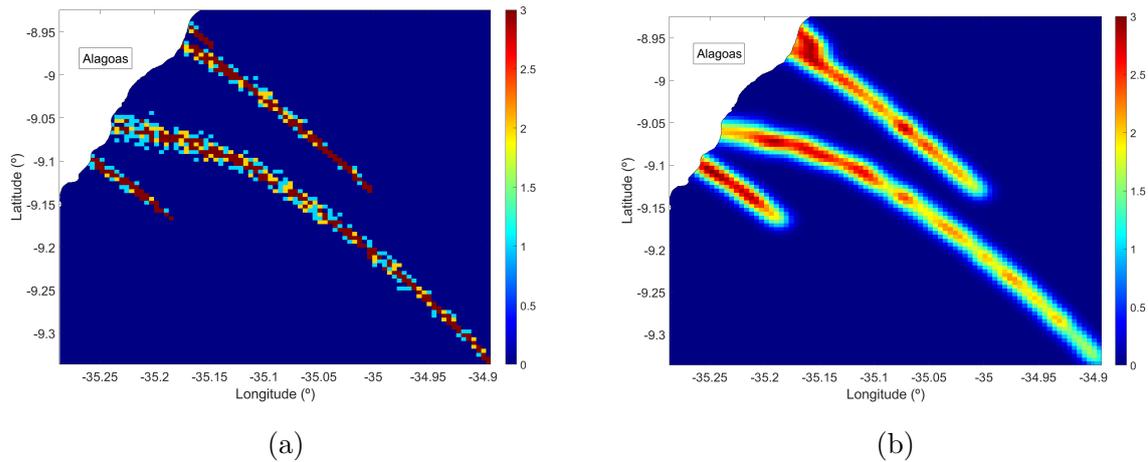


Figura 2.5: Tratamento das partículas usando (a) histograma 2D e (b) *Kernel Density Estimation*.

em  $n$  amostras  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$  com função de densidade de probabilidade  $\hat{c}_{\text{kde}}(\mathbf{x})$ . Uma largura de banda  $H$  é determinada por uma matriz positiva definida, no caso, uma matriz diagonal com valores iguais pode ser usada. Uma função de kernel  $K : \mathbb{R}^2 \rightarrow \mathbb{R}$ , como a gaussiana 2D, é aplicada.

A figura 2.5 mostra uma comparação entre histograma bidimensional e KDE no resultado final da simulação computada pelo GNOME (figura 2.4). Pode-se observar que a distribuição do fluido no caso do KDE tem uma forma mais suave e realística, considerando que o processo de dispersão deve apresentar comportamento gaussiano. As partículas que estão na costa, em terra, foram deixadas de lado nesse procedimento, pois o objetivo do trabalho é monitorar, com os veículos autônomos, poluentes ainda flutuando na água.

## 2.3 ISL

Apesar de todas regiões apresentarem danos quando sob contato com o poluente, algumas regiões são consideradas mais sensíveis que outras. O Ministério do Meio Ambiente disponibiliza dados acerca da vulnerabilidade de várias áreas costeiras através dos índices de sensibilidade do litoral (ISL), que podem assumir valores de 0 a 10. Um ISL que atinge 10 representa uma localidade com altíssimo risco se exposto ao poluente. Esse índice levanta características geomorfológicas do local, considerando atributos como ecossistemas, formações rochosas, tipo de solo, unidades de conservação, dificuldade de acesso, facilidade de absorção e dificuldade de desprendimento do poluente.

A figura 2.6 mostra uma parte das localidades com levantamento de ISL em Alagoas (em vermelho), além de uma tabela com algumas das características consideradas para estimação desse índice nessa região. É interessante observar que, na porção de dados apresentados, localidades de pesca e manguezais atingem o ISL máximo.

Conhecer o ISL de uma região é importante, não só para definir estratégias de locais onde missões de monitoramento vão acontecer, sendo também um recurso útil para ser incorporado em técnicas de navegação implementadas nos agentes. Essa é uma característica a ser considerada ao adaptar técnicas provenientes de outras áreas.

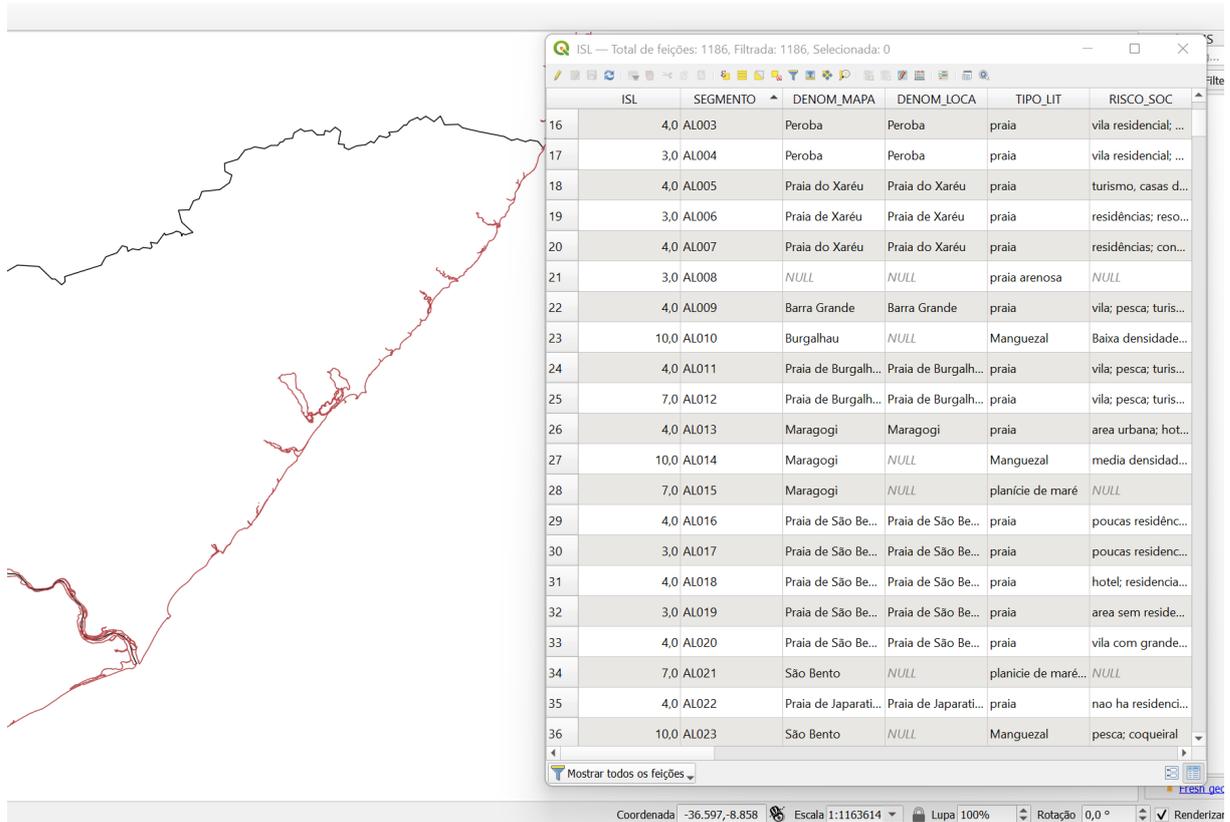


Figura 2.6: Visualização de segmento com levantamento de ISL e alguns dos atributos considerados.

O próximo capítulo irá detalhar a problemática desse trabalho com relação à política de patrulhamento, que será executada como algoritmo de navegação dos veículos. Será a partir dela que a verificação e monitoramento serão desempenhadas nas áreas escolhidas como estudo de caso.

# Capítulo 3

## Monitoramento Persistente

Uma taxonomia para problemas de *target management* por múltiplos robôs trabalhando de forma colaborativa foi proposta por Robin and Lacroix (2016), resumida na Figura 3.1. A partir dela, pode-se descrever que o foco das missões a serem empregadas no sistema proposto se encaixam na área de patrulhamento (*patrolling*). Isso porque o problema se concentra, a princípio, na detecção da pluma química no oceano, usando agentes móveis em uma busca baseada na probabilidade de presença do poluente, de forma que pode ser necessário visitar algumas localidades de acordo com a dinâmica costeira. Mais especificamente, quando o problema de patrulhamento envolve objetivos que não reagem à presença dos agentes, esse tipo de missão pode ser classificada como monitoramento persistente, ou supervisionamento persistente.

O problema de patrulhamento com sistemas multiagentes é considerado um problema NP-Difícil, entrando na classe dos problemas que não podem ser resolvidos em tempo polinomial por uma máquina determinística. Uma ampla revisão na literatura em supervisionamento persistente pode ser encontrada no trabalho de Nigam (2014), focando na utilização de um conjunto de UAVs (do inglês, *Unmanned Aerial Vehicle*) implementando planejamento, decomposição espacial, padrões de busca, resolução distribuída de problemas, entre outros. Podem ser encontrados também trabalhos que aplicam métodos de elementos finitos (Elwin et al., 2020) e algoritmos evolucionários (Xiong et al., 2020) em situações de amostragem ambiental e oceanográfica.

### 3.1 Política reativa

Dentre as técnicas levantadas, se destaca a estratégia reativa de Nigam and Kroo (2008), que propõe a utilização de uma política reativa de patrulhamento projetada para um espaço de geometria quadrada. Cada célula do mapa de grade tem uma idade  $A_j$ , que aumenta com o tempo quando não há visita do agente. Usando essa idade e a distância do agente para a célula, uma pontuação é calculada por uma *value function* para essa célula. A equação 3.1 resume essa função.

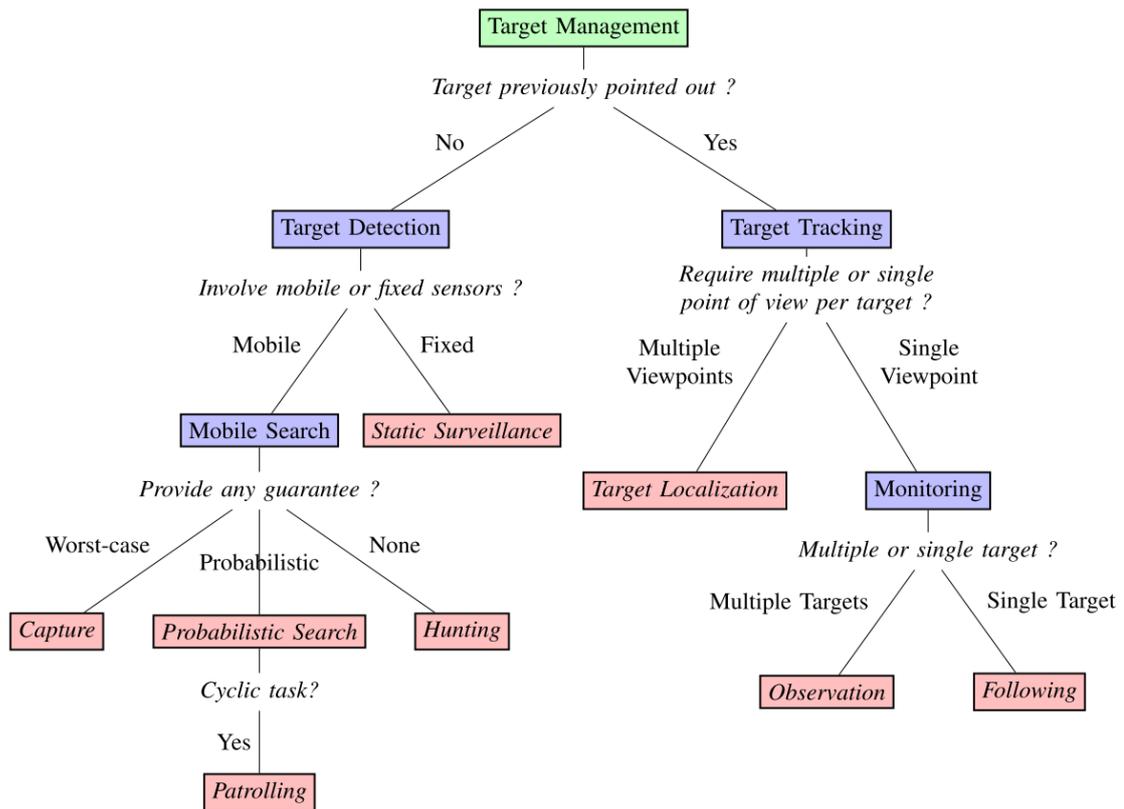


Figura 3.1: Taxonomia para problemas de *target management* utilizando múltiplos robôs, segundo Robin and Lacroix (2016)

$$v_j = \max\{(A_j + \omega_0 \delta_j, 0)\}, \quad (3.1)$$

Um peso  $\omega_0 < 0$  é associado à distância entre o robô e a  $j$ -ésima célula. É escolhido a célula com maior  $v_j$ , se tornando a célula objetivo, a qual o agente se move em direção a ela. Na implementação clássica é comum acontecer de mais de uma célula apresentar os maiores valores da política reativa (problema multimodal). Nessa situação, a direção e sentido que a frente do agente está apontando no momento pode ser usada como desempate, ou seja, é escolhida a célula que exige menor deslocamento da frente do robô.

As figuras 3.2a e 3.2b mostram o perfil de caminho gerado pela política reativa em um mapa de grade 10x10, com o agente iniciando na célula (0,0). Como o envelhecimento das células, denotado pela variável  $A$ , cresce na mesma taxa em todo mapa, o agente só decide voltar para o início depois de varrer toda região. Esse momento em que o agente decide voltar para a posição inicial é apresentada pelo caminho de cor vermelha em ambas figuras.

Para calcular a célula objetivo, a política reativa só precisa aplicar as *value functions* e registrar as maiores pontuações. Esse trabalho pode ser paralelizado e não requer muitas operações computacionais. Entretanto, o algoritmo supõe que o veículo é único na região. Uma maneira de envolver mais de um agente no problema é por decomposição espacial

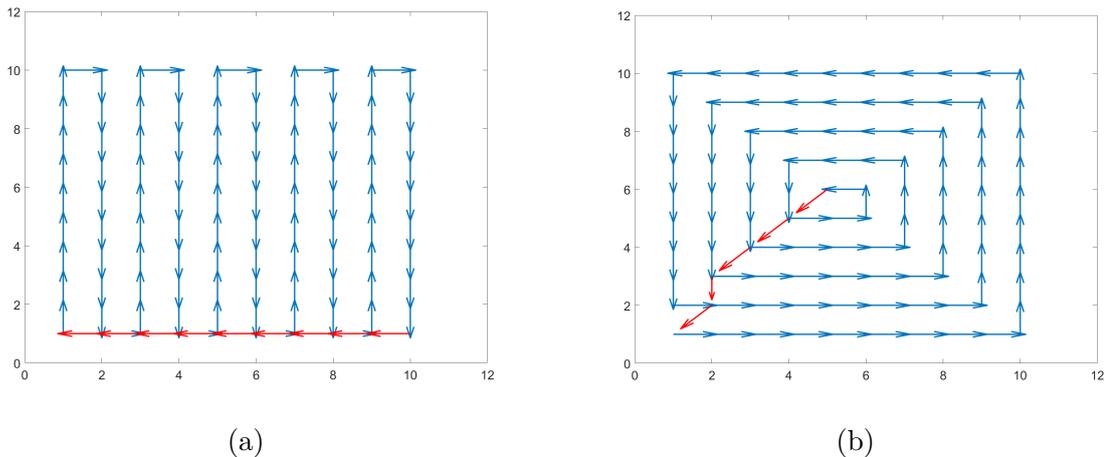


Figura 3.2: Perfil de caminho gerado pela política reativa (a) sem considerar e (b) considerando menor deslocamento da frente do robô.

da região, de forma que os agentes não iriam compartilhar espaço entre si.

### 3.1.1 Cenário multiagente

Ainda em Nigam and Kroo (2008), mas sendo explorada de maneira mais detalhada em Nigam et al. (2012), foi proposta uma política distribuída, ainda projetada para espaço de geometria quadrada, permitindo aplicar um cenário multiagente no problema de patrulhamento, com veículos compartilhando a mesma região. Através dessa política, também é designada uma célula objetivo no mapa de grade, que representa a região de interesse da missão. Dessa forma, o robô se move diretamente para a célula objetivo, que é computada de acordo com pesos oferecidos na equação da política reativa e associados a componentes como tempo sem visitação, distância da célula para o robô e distância da célula a um robô vizinho.

A implementação dessa política reativa distribuída utiliza a seguinte formulação:

$$v_{ij} = \max\{(A_j + \omega_{di}\delta_{ji} + \omega_{ni} \min_{k \neq i} \delta_{kj}, 0)\}, \quad (3.2)$$

onde cada  $i$ -ésimo agente calcula o valor de cada  $j$ -ésima célula no mapa,  $A_j$  é a idade da célula,  $\omega_{di}$  e  $\omega_{ni}$  são pesos respectivos a distância  $\delta_{ji}$  do robô à célula e a distância de um robô vizinho mais próximo à célula  $\delta_{kj}$ .

## 3.2 Política reativa adaptada

Para a aplicação no cenário que este trabalho se propõe a atuar, a abordagem proposta por Nigam et al. (2012) deve ser modificada para atuar em regiões poligonais arbitrárias e levar em consideração parâmetros que importam no monitoramento marítimo, como probabilidade de concentração de óleo e regiões de maior sensibilidade.

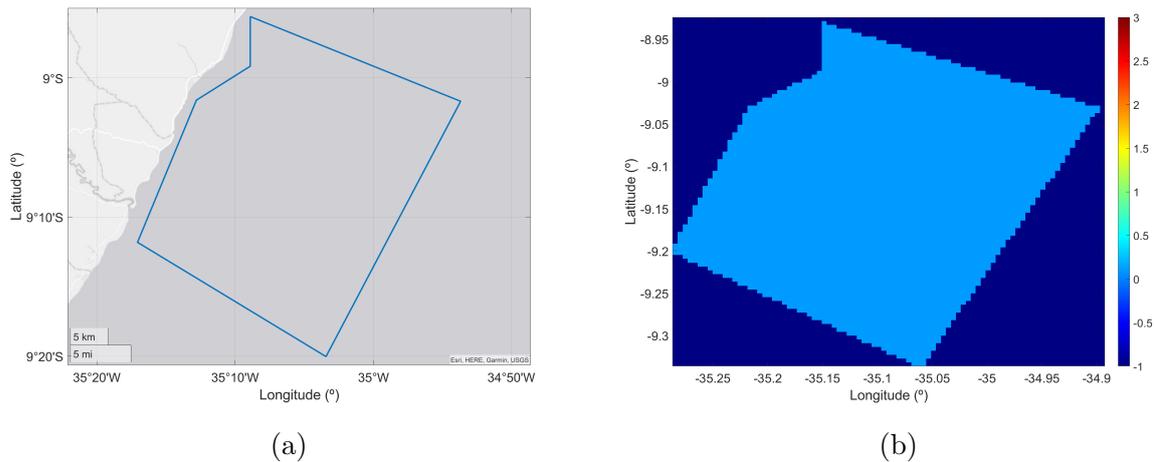


Figura 3.3: Região exemplo para aplicação da política reativa adaptada, em forma (a) poligonal e (b) discretizada em mapa de grade.

Os agentes envolvidos na missão utilizam a informação de concentração estimada pelo KDE da região de interesse e da posição dos outros veículos. Tendo esses recursos, cada robô executa localmente um algoritmo que planeja a movimentação para a próxima célula. Desconsidera-se o tempo de visitação da célula, enquanto componentes são adicionados para estimação de concentração do poluente  $c_j$ , vindo do KDE, e sensibilidade ambiental  $s_j$ , que pode ser proximidade da costa ou através do Índice de Sensibilidade do Litoral (ISL), resultando na formulação a seguir:

$$v_{ij} = \max\{\kappa_i + \omega_{ci}c_j + \omega_{si}s_j + \omega_{di}\delta_{ij} + \omega_{ni} \min_{k \neq i} \delta_{kj}, 0\}, \quad (3.3)$$

de forma que  $\kappa$  é um valor de ganho que auxilia na movimentação do agente, e  $\omega_{ci}$  e  $\omega_{si}$  são pesos para concentração e sensibilidade, respectivamente. Assim, torna-se possível designar agentes com configurações diferentes para cada peso, refletindo em prioridades diferentes na missão, como células que contenham maior probabilidade de presença de óleo, mesmo que longe da costa, ou focar em regiões mais sensíveis.

Para resolver o problema de movimentação em áreas poligonais, ao invés de restringir a regiões quadradas, pode-se utilizar o algoritmo de busca em grafo  $A^*$  logo após a determinação da célula objetivo do agente. Com isso, além de planejar um caminho ótimo e apresentar baixo custo computacional, podem-se designar áreas de voo proibido e evitar células com presença de outros agentes.

Para aplicar essa adaptação, foi definida a região poligonal da figura 3.3, com área de aproximadamente  $1061 \text{ km}^2$ , localizada na área de proteção ambiental Costa dos Corais. Após um dia de simulação de derramamento de óleo na borda leste da região, foram usados 7 veículos divididos em dois grupos com prioridades diferentes, onde 4 veículos focaram em regiões com maior concentração e 3 veículos focaram em regiões mais sensíveis. Nesta aplicação, a sensibilidade foi definida pela proximidade da costa, de forma que células com

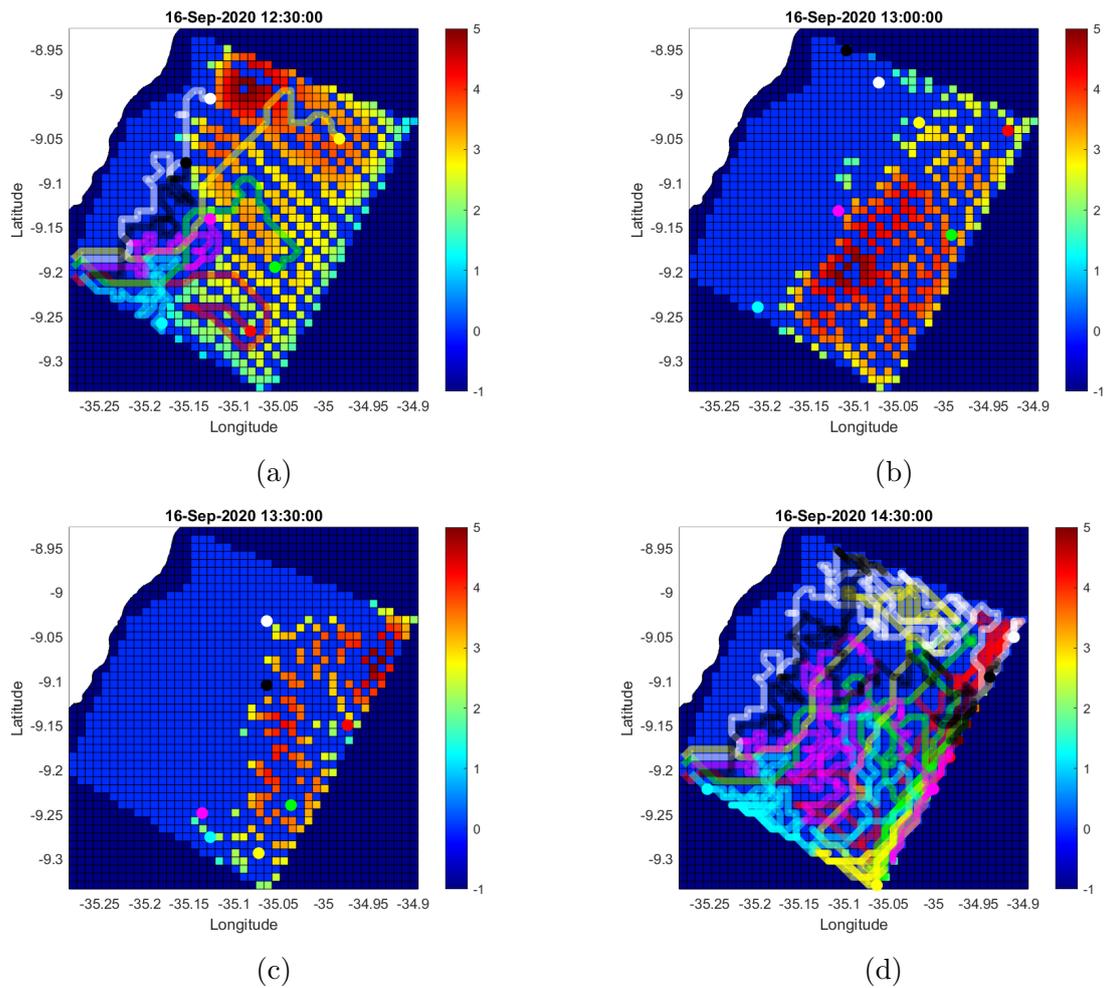


Figura 3.4: Aplicação da estratégia de patrulhamento reativa adaptada com 7 agentes. As imagens (a) e (d) mostram também o caminho percorrido pelos agentes, enquanto as imagens (b) e (c), apenas as posições atuais.

partículas próximas da costa têm  $s_j$  maior. A figura 3.4 mostra os estados da simulação em momentos diferentes.

Como a ideia dessa simulação foi apresentar o funcionamento do algoritmo de patrulhamento, ao passar por uma célula o agente consome as partículas presentes na simulação e não adiciona novas. Entretanto, é interessante observar que a concentração de óleo nas células vai mudando no tempo, mesmo depois do início da missão, por conta da execução em paralelo da simulação com GNOME.

# Capítulo 4

## Arquitetura do framework

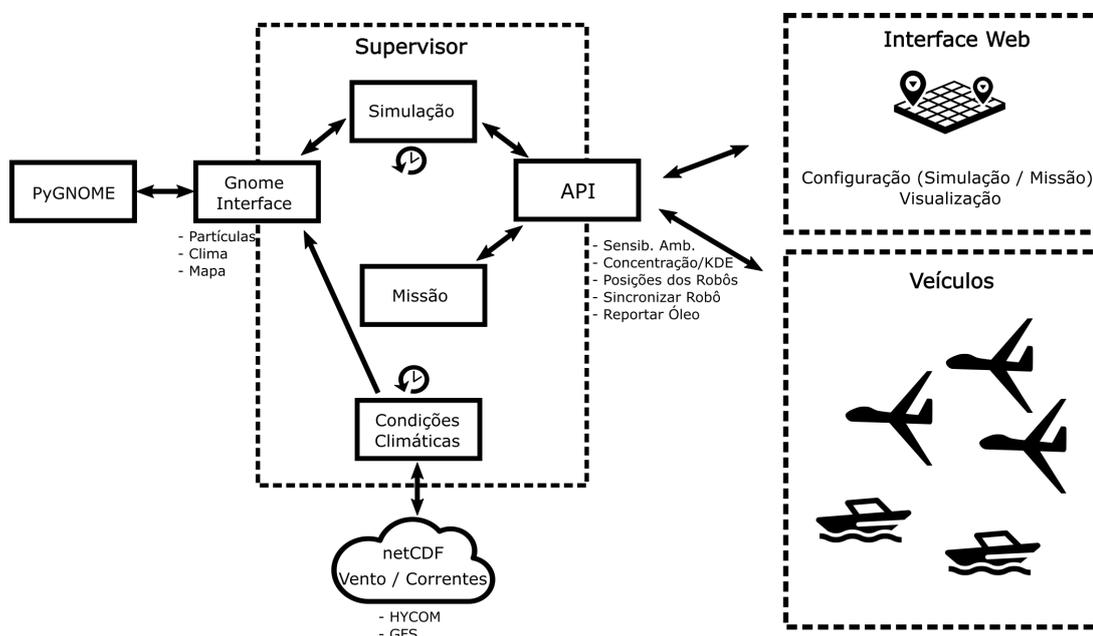
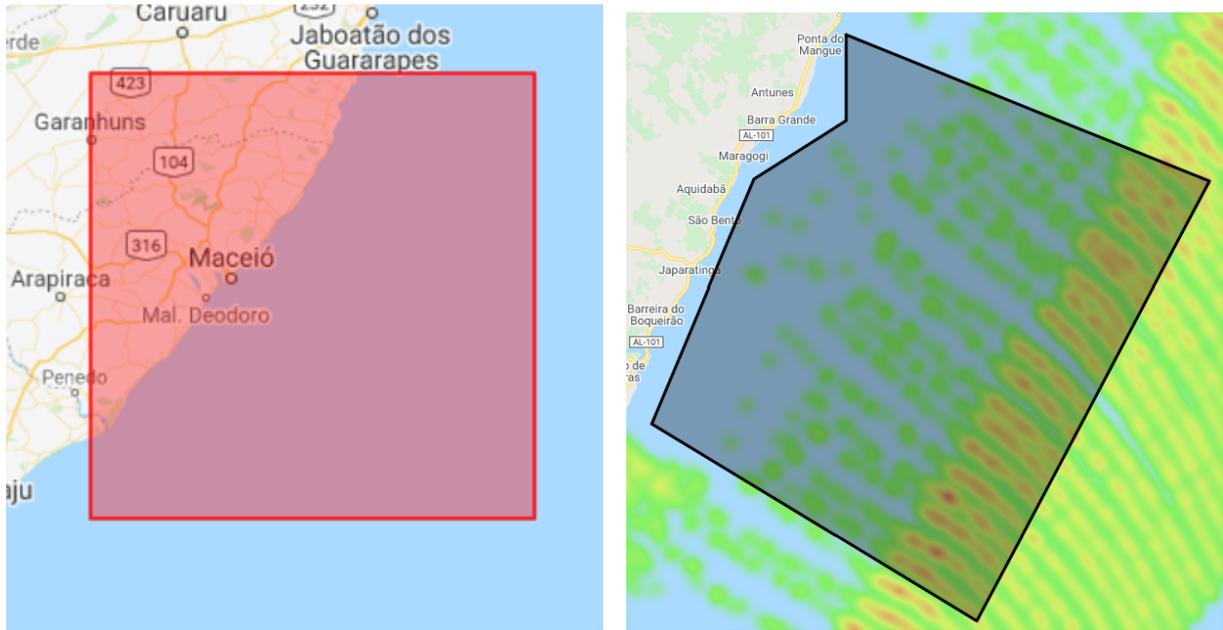


Figura 4.1: Arquitetura para sistema de monitoramento persistente.

Neste trabalho foi desenvolvida uma arquitetura com estrutura modular (Figura 4.1), de forma que os componentes desse sistema possam ser substituídos por outras implementações, de acordo com o cenário que o problema de monitoramento está lidando. Assim, o funcionamento de cada módulo, por exemplo, supervisor, interface *web* e veículos, desempenha suas tarefas de forma assíncrona e independente dos demais, sincronizando informações quando necessário através de uma interface.

O sistema consiste de um módulo supervisor rodando em uma estação central, onde são instanciadas simulação e missão, sendo fornecida uma API (do inglês, *Application Programming Interface*) responsável por se comunicar com veículos envolvidos na missão e uma interface *web* a partir de requisições HTTP no padrão REST (Masse, 2011). Essa API foi desenvolvida como um serviço usando *Flask*, que roda em *Python*. A interface *web* possibilita configurar parâmetros de simulação e missão, além de visualizar posiciona-



(a) Região de simulação, representando a costa de Alagoas.

(b) Região de missão, com partículas, na Costa dos Corais.

Figura 4.2: Exemplos de regiões de interesse para simulação e missão.

mento dos agentes, partículas da simulação, entre outras variáveis. Os veículos recebem informações pela API e executam uma política local de movimentação, seguindo uma tarefa de patrulhamento.

De forma geral, dois componentes são centrais na execução do sistema supervisor, a Simulação e a Missão. Enquanto a simulação lida com uma região maior da costa, realizando simulações de processos de dispersão de óleo de registros observados ou hipotéticos, a missão se concentra em uma subárea, fornecendo uma interface para a aplicação de veículos em tarefa de monitoramento.

A Figura 4.2 mostra exemplos de regiões de interesse em ambos os casos, aplicando no cenário de monitoramento de uma região de preservação em Alagoas, a Costa dos Corais. Essa abordagem permite que partículas nas proximidades da região de missão possam influenciar ou entrar nessa região, de acordo com a dinâmica do processo de transporte ou variáveis climáticas.

## 4.1 Simulação

O componente Simulação gerencia variáveis mais abrangentes associadas ao processo de transporte do poluente e a sensibilidade ambiental, trabalhando em uma grande região da costa. A simulação é implementada em forma de *thread*, com a definição de um tempo  $t_s$  que determina a periodicidade que o supervisor deve executar rotinas internas, seja para tratar as partículas da simulação ou realizar chamadas para rodar um passo do

simulador externo, no caso, o GNOME. Além disso, a Simulação responde às requisições de API relacionadas às partículas, região de sensibilidade ambiental e relatos de provável presença de óleo.

Para incorporar o GNOME na solução desenvolvida neste trabalho, foi usado o pyGNOME, que é um conjunto de utilitários e *bindings* para a linguagem Python e funciona de maneira independente das outras plataformas do GNOME. Na arquitetura, foi desenvolvido o módulo *GNOME Interface*, servindo de comunicação entre as funcionalidades nativas do pyGNOME e o restante do sistema supervisor. Essa interface abre espaço para que outras implementações e ferramentas computacionais de simulação possam ser usadas pelo sistema, sendo necessário construir apenas uma nova interface, com funcionamento similar ao padrão de projeto *Adapter* (Gamma et al., 2015).

A natureza dinâmica dos oceanos afeta diretamente o transporte de óleo. Assim, relatos de derramamentos ou manchas de óleo podem se tornar menos confiáveis com o tempo, ou mesmo durante a viagem dos agentes até a localização. Para tratar isso, essas novas observações são inseridas na simulação em forma de novas partículas, se movimentando e interagindo com as demais e o ambiente.

Variáveis de previsões meteorológicas, como intensidade e direção de vento e correntes marítimas de superfície, podem ser adquiridas automaticamente a partir de repositórios na internet. Alguns dos provedores dessas informações são o consórcio HYCOM (do inglês, *HYbrid Coordinate Ocean Model*) e o GFS (do inglês, *Global Forecast System*), que disponibilizam leituras antigas e previsões de correntes e ventos, respectivamente, em arquivos no formato NetCDF que são suportados no GNOME. A periodicidade  $t_w$  de aquisição desses arquivos é determinada em uma *thread* específica denominada *Condições Climáticas*.

A sensibilidade ambiental é uma característica importante para os veículos que seguirão em missão na costa. O ISL é um dado que pode ser usado para computar esse parâmetro e pode ser encontrado em todo litoral nordeste. Com isso, podem ser construídos campos potenciais em forma de gaussiana 2D para cada leitura de ISL, onde o valor do índice representa a amplitude do pico localizado na latitude e longitude onde há registro de cálculo do indicador.

Como essa tarefa pode ser realizada para toda a costa e não requer repetição, a computação dos campos potenciais é realizada no início da simulação. Ao desempenhar uma missão, o veículo pode adquirir um mapa de grade do campo potencial de sensibilidade ambiental computado na simulação, aplicado na região de interesse da missão, como mostrado na Figura 4.3a.

Uma vez que derramamentos de óleo são mais facilmente combatíveis no alto mar, antes de poluir o litoral, uma opção alternativa para o ISL como indicador de sensibilidade ambiental pode ser utilizar a distância até a costa. Nesse caso, a sensibilidade de uma determinada região no mapa de simulação pode ser calculada a partir da distância para

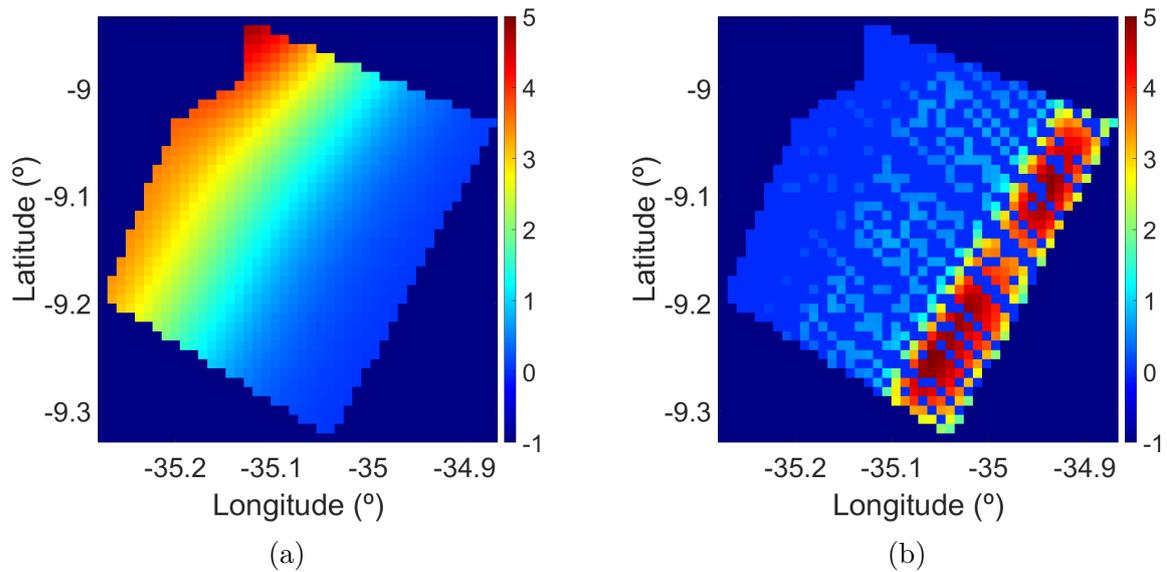


Figura 4.3: Exemplos de mapas de (a) sensibilidade ambiental e de (b) concentração aplicados em região de interesse da missão.

o ponto mais próximo da costa.

Outra funcionalidade do sistema é a de reportar prováveis situações de derramamento de óleo, seja por um relato de derramamento ocorrido, seja pela passagem de um navio petroleiro ou presença de estações de extração de petróleo. Esse recurso é dos quais permitem inserir novas partículas na simulação, interagindo com as partículas já existentes do GNOME. Essa funcionalidade também é útil para simular uma fonte constante de derramamento, sendo chamada periodicamente, de acordo com a vazão estimada dessa fonte.

## 4.2 Missão

A Missão é a entidade que designa e administra uma região de interesse dentro da área de simulação, onde os veículos devem realizar suas tarefas de monitoramento. Essa região de interesse pode ser representada por um polígono arbitrário e georreferenciado, esse polígono é definido *a priori* através de um arquivo escrito em KML (do inglês, *Keyhole Markup Language*), linguagem de marcação para dados geográficos. KML é uma linguagem comum em sistemas de informação geográfica, como ArcGIS e QGIS, e demais ferramentas que fazem uso de dados georreferenciados, como o *Google Earth* e o *Google Maps*.

Os agentes, por sua vez, se movimentam em um mapa de grade que é uma discretização do polígono. A partir da resolução definida para as células, é construído um mapa de grade resolvendo um problema de PIP (do inglês, *point-in-polygon*), um problema clássico da geometria computacional, onde é calculado se os pontos representados pela célula estão

dentro ou fora do polígono (Hormann and Agathos, 2001). Células do mapa que ficam de fora da região de interesse são instanciadas com concentração  $c_j = -1$ , fornecendo um fator consequente de repulsão das mesmas para os agentes.

É papel da missão oferecer aos veículos envolvidos a configuração de tarefas para o *deploy*, o estado da região de interesse da missão, o posicionamento dos agentes vizinhos e sincronizar quando esses se movimentam e adquirem novas informações *in situ*. Considera-se que essas ações possam ser realizadas a qualquer momento, de forma assíncrona, através de requisições ao sistema supervisor.

Como o mapa da missão é uma grade de células, é necessário passar o quantitativo das partículas trabalhadas nos passos da simulação para dados de concentração em células. Considerando que cada partícula tem registro de localização (longitude e latitude), é possível computar uma estimativa bidimensional de concentração usando KDE na região de interesse da missão. A Figura 4.3b traz uma visualização do KDE de concentração sobre a região e as partículas do exemplo de aplicação de missão na Costa dos Corais (Figura 4.2b).

Através da API, os veículos da missão podem enviar uma requisição de sincronização das informações obtidas. Dentro dessa requisição, o agente comunica seu posicionamento atual e, se detectado óleo no local, a quantidade de partículas relativas a essa observação. Caso não seja observada presença do poluente na célula em que o agente se encontra, as partículas da simulação referenciadas na mesma são subtraídas do processo. Essas informações adquiridas atualizam variáveis importantes tanto da missão, por exemplo, posicionamento do robô, como da simulação, como no caso das partículas envolvidas, no sistema supervisor.

A figura 4.4 sumariza o funcionamento do robô e as chamadas de requisição usadas para sua comunicação com o sistema supervisor. As chamadas */mission/kde*, */mission/env\_sensibility* e */mission/robots\_pos*, requisitam ao supervisor, respectivamente, o mapa de concentração, o mapa de sensibilidade ambiental e a posição dos robôs em missão. Já */robot\_fb* envia à central a nova posição do agente, de acordo com o resultado do algoritmo de navegação, e registro de presença de óleo, que é convertido em partículas inseridas na simulação.

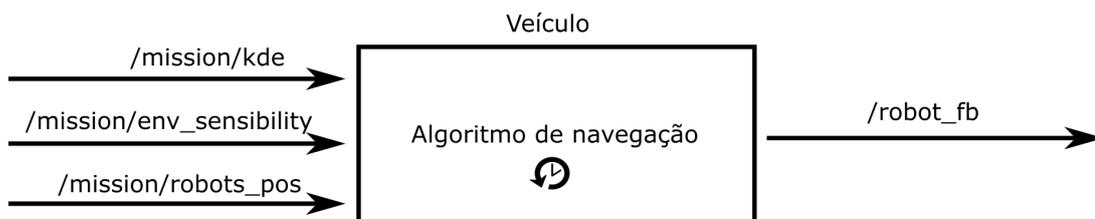


Figura 4.4: Esquema de funcionamento local de veículo envolvido na missão.

# Capítulo 5

## Experimentos

Os testes realizados neste trabalho se dividem em duas categorias. Uma categoria avalia o funcionamento do sistema computacional como um todo, isso é, o fluxo de definição, configuração e acompanhamento das missões, em cenários diferentes. Outra categoria de experimento é com relação à integração do sistema supervisor com a política implementada, usando diferentes parâmetros nos cenários definidos. Vídeos dos experimentos estão disponibilizados na *web*, através do serviço de *streaming* de vídeos *YouTube*<sup>1</sup>.

### 5.1 Funcionamento do sistema supervisor

A meta desta fase de experimentos é testar a generalidade do sistema supervisor em aspectos como suporte a diferentes localidades, com diferentes condições climáticas e parâmetros de sensibilidade ambiental diferentes. Considera-se que o sistema supervisor fica executando continuamente em uma estação central ou um servidor. A implementação desse sistema, que foi desenvolvido em *Flask*, se encontra disponível publicamente em repositório do GitHub<sup>2</sup>.

Nesta etapa foram definidos dois cenários geograficamente distintos, um nos arredores de Maragogi - AL e outro em Natal - RN. Em cada um dos cenários, foi computada, em momento distintos, uma simulação prévia de derramamento contínuo de óleo proveniente do limite da zona territorial contígua. Em um intervalo de um dia, uma grande quantidade de partículas foi adicionada gradualmente à simulação. O objetivo dessa simulação foi estabelecer uma mesma condição inicial de derramamento hipotético de óleo, proveniente de uma fonte desconhecida em alto-mar, para os estudos de casos que usarem o cenário. Partindo desse suposto relato de aparecimento de óleo, com partículas computadas em uma simulação que evolui a cada 3 minutos, é designada uma missão para que agentes viagem por uma região arbitrária e observem *in loco* a presença, ou não, do poluente.

---

<sup>1</sup>*Playlist* com vídeos de todos os experimentos dos estudos de caso em <https://youtube.com/playlist?list=PLB179mvE3N69smaAserAX1URSEZj00iIR>

<sup>2</sup>Disponível em <https://github.com/EASY-SPARC/patrol-for-oil-supervisor>

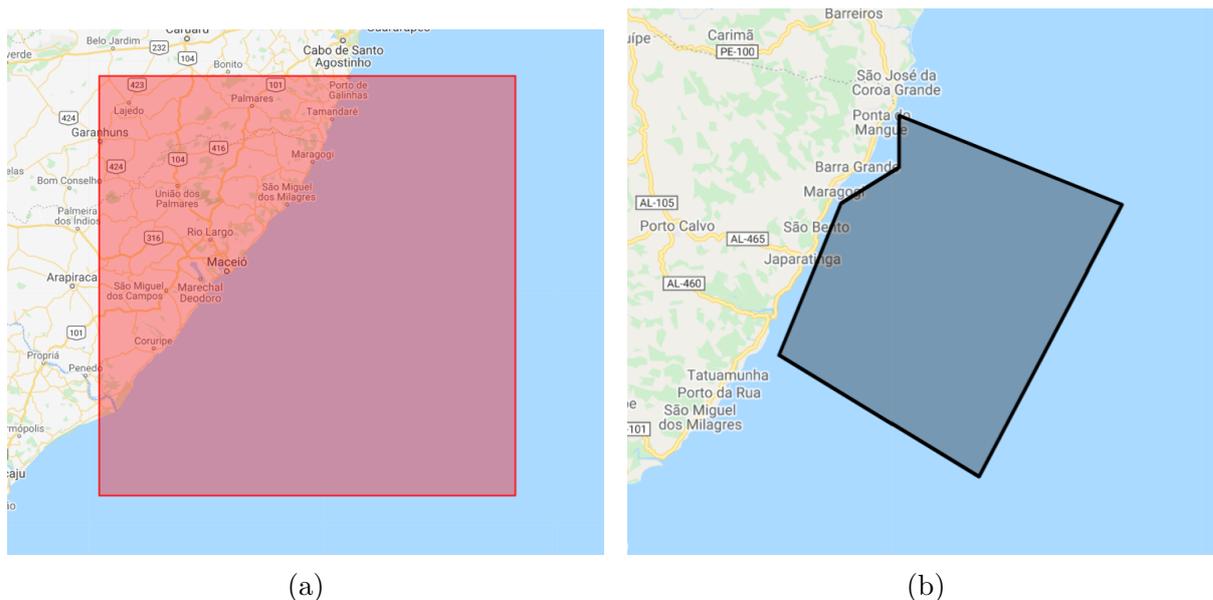


Figura 5.1: Regiões de (a) simulação e (b) missão no cenário 1, com foco em área de Maragogi-AL.

### 5.1.1 Cenário 1

O primeiro cenário explora uma parte da área de proteção ambiental Costa dos Corais, com foco em Maragogi, Alagoas. A região de simulação foi configurada via interface *web*, com limites de coordenadas  $-8,5^{\circ}$  ao norte,  $-11^{\circ}$  ao sul,  $-34^{\circ}$  a leste e  $-36,5^{\circ}$  a oeste. Dentro dessa região, foi definido um polígono irregular arbitrário e não-convexo. Esse polígono, com área de cerca  $1000 \text{ km}^2$ , tem o papel de representar a área de voo para os veículos envolvidos na missão. A figura 5.1 apresenta ambas regiões.

Para este cenário, o mapa de sensibilidade ambiental foi configurado para ser gerado a partir dos campos potenciais provenientes do ISL, com distribuição gaussiana 2D e desvio padrão  $\sigma = 0.1$ . Para computação dos mapas de sensibilidade e concentração, foi considerada uma discretização em células de  $1 \text{ km}^2$ . A concentração baseada em KDE foi designada com *kernel* gaussiano de duas dimensões com largura de banda de 0,2.

Enquanto o mapa de grade de sensibilidade é o mesmo que o apresentado na figura 4.3a, por se tratar do ISL, os mapas de visualização das partículas e de concentração são apresentados na figura 5.2. Pode ser observada uma grande quantidade de partículas, não só dentro da área da missão, como também fora. Eventualmente, essas partículas podem entrar na região de interesse da missão, da mesma forma que partículas de óleo hipotético que estão dentro podem sair do escopo da missão.

Ter acesso ao mapa de partículas da simulação antes de definir a missão foi um requisito levantado, uma vez que ter acesso ao estado inicial esperado de um processo de dispersão pode auxiliar em fatores de decisão importantes, como o número de agentes a serem usados e os papéis de cada um na missão. Não obstante, esperasse que no decorrer do funcionamento do sistema, uma nova missão possa ser configurada enquanto a simulação

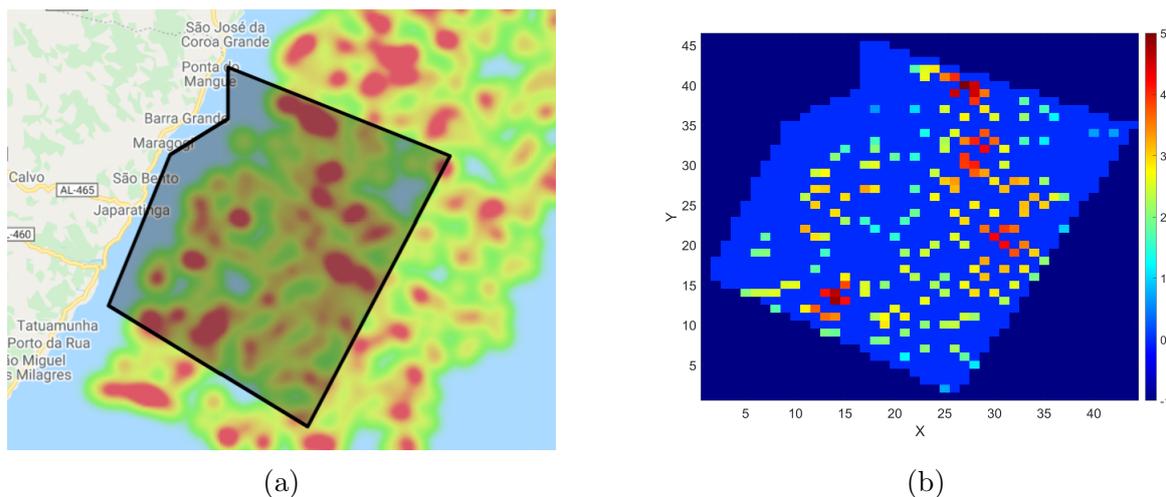


Figura 5.2: Estado inicial da área de missão no (a) mapa de partículas e (b) mapa de concentração.

de partículas segue executando em plano de fundo.

### 5.1.2 Cenário 2

O segundo cenário trabalha nos arredores de Natal, capital do Rio Grande do Norte, uma área mais ao norte da anterior, exposta a outras condições climáticas e variáveis ambientais. A região de simulação foi configurada via interface *web*, com limites de coordenadas  $-4,5^{\circ}$  ao norte,  $-7^{\circ}$  ao sul,  $-33^{\circ}$  a leste e  $-36^{\circ}$  a oeste. Um novo polígono foi construído com área de aproximadamente  $1000 \text{ km}^2$ , desta vez com topologia diferente do cenário anterior, por apresentar um furo em seu interior.

O furo no interior do polígono da missão está representando, arbitrariamente, uma região onde a passagem de agentes não é permitida. Partículas dentro dessa região são desconsideradas da missão, tais como as partículas no exterior do polígono, mas seguem fazendo parte da área de simulação do GNOME. Eventualmente, essas partículas podem entrar em uma zona de locomoção permitida, se tornando relevantes para os algoritmos de navegação dos agentes.

Este cenário se assemelha com o outro quanto à discretização em células dos mapas de sensibilidade e concentração em elementos de  $1 \text{ km}^2$ . Além disso, a concentração baseada em KDE também usou *kernel* gaussiano de duas dimensões com largura de banda de 0,2. Não obstante, o mapa de sensibilidade ambiental foi gerado de forma diferente, usando a distância da célula ao ponto mais próximo da costa. Os mapas de concentração e sensibilidade ambiental iniciais deste cenário podem ser conferidos na figura 5.4

Pode ser observado que esse cenário apresentou menos partículas que o anterior, isso foi decorrente das condições de vento e corrente da região no dia em que aconteceu a simulação, fazendo o óleo se espalhar para outros locais, com apenas alguns chegando à costa. No cenário 1, os fatores climáticos apresentavam vetores que apontavam direta-

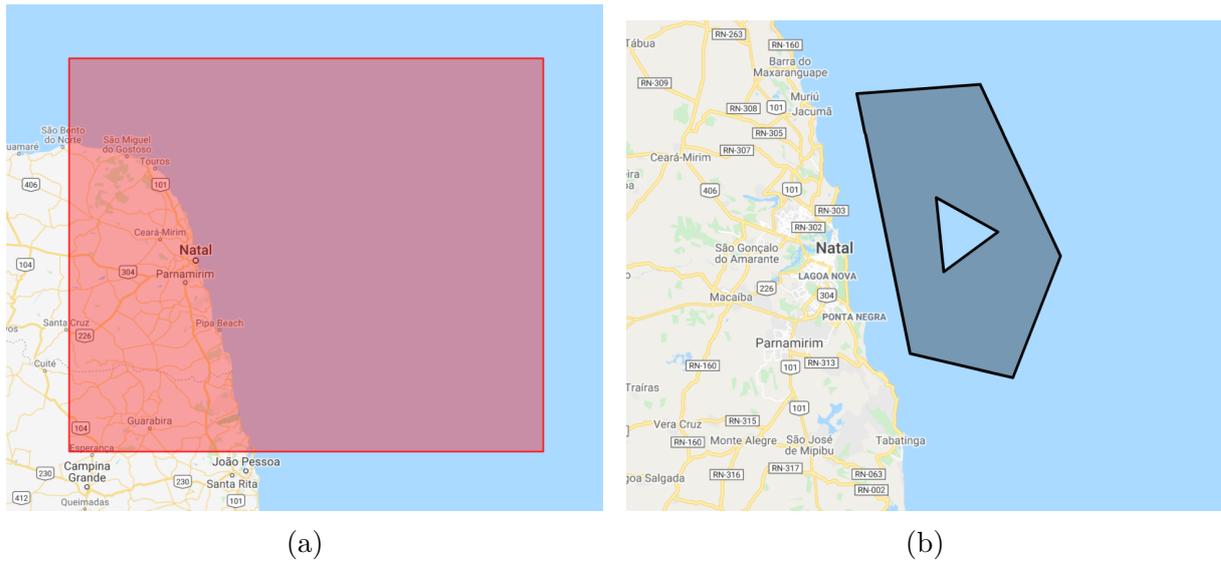


Figura 5.3: Regiões de (a) simulação e (b) missão no cenário 2, com foco em área de Natal-RN.

mente para o litoral.

## 5.2 Supervisor e política implementada

Durante a configuração da missão, são escolhidos a quantidade de agentes envolvidos e os parâmetros de configuração desses agentes, para então dar início ao *deploy* dos veículos na tarefa de monitorar a região. Entretanto, essas etapas apenas preparam o sistema supervisor para fornecer e receber dados desses veículos, ao invés de controlá-los diretamente. O *framework* foi projetado de forma que o agente tenha independência de tecnologia ou linguagem de programação no desempenho de sua atividade, favorecendo a computação local do algoritmo de navegação.

É possível adaptar o *framework* para aplicação de técnicas de navegação centralizada, entretanto essa abordagem não foi favorecida neste trabalho por conta da natureza da comunicação no ambiente em que os veículos devem realizar a missão. Em alto mar, não se deve esperar sistemas com alta disponibilidade de comunicação, então a utilização de técnicas descentralizadas se destacam.

O ponto central do *framework* neste trabalho é fornecer um arcabouço para soluções de monitoramento persistente que, de acordo com a taxonomia apresentada anteriormente para problemas de *target management*, estão dentro dos problemas de detecção de objetivo (*target detection*). Assim, considera-se que o objeto de interesse, ou seja, a pluma do poluente, não está no campo de visão do supervisor, mas existe informação probabilística de onde pode estar.

Por conta disso, nos experimentos deste trabalho, os veículos ao passarem por uma célula, limpam o mapa de partículas dessa localidade. No caso de detecção do poluente,

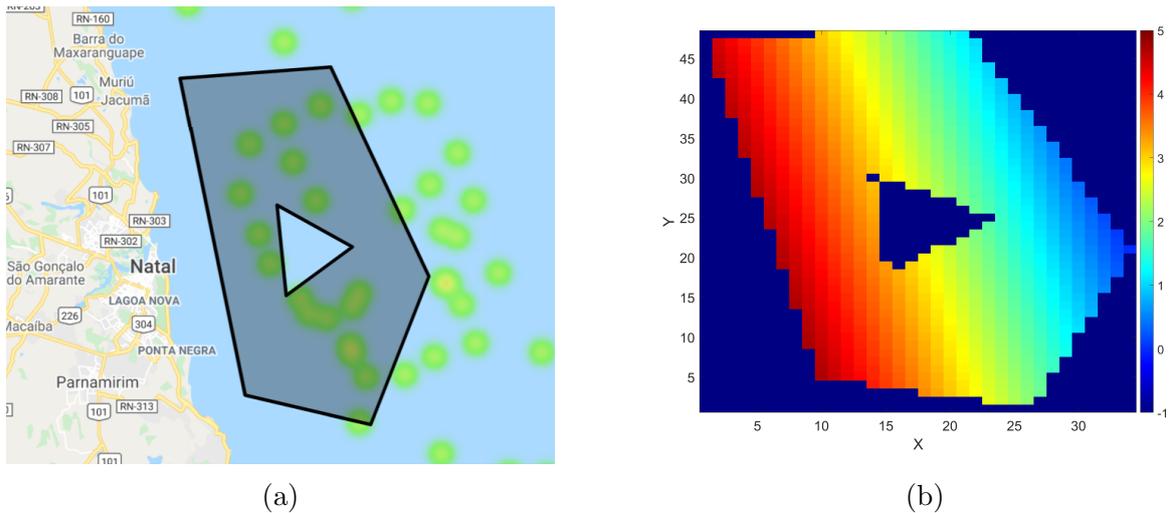


Figura 5.4: Estado inicial da área de missão no (a) mapa de partículas e (b) mapa de concentração.

suas partículas podem ser inseridas na simulação, mas esse aspecto não é considerado como escopo destes experimentos por requerer explorar técnicas de rastreamento de objetivo (*target tracking*), que estão no outro espectro da taxinomia adotada.

Como as informações necessárias para rodar o algoritmo de navegação são fornecidas pelo supervisor através de uma API, qualquer aplicação com capacidade de realizar requisições no padrão REST pode participar da missão como um agente. Sendo assim, foi implementado um algoritmo em MATLAB para simular veículos independentes na missão, usando recursos da *Parallel Computing Toolbox* com o intuito de isolar o funcionamento desses agentes. Os códigos usados nessa implementação podem ser conferidos em repositório do GitHub<sup>3</sup>.

Esta etapa de experimentos foi dividida em 6 estudos de caso, explorando características diferentes do sistema e dos agentes envolvidos. Os casos I e II aplicam nos cenários 1 e 2, respectivamente, agentes rodando a política reativa adaptada com todos compartilhando os mesmos parâmetros. Os casos III e IV aplicam nos cenários 1 e 2, respectivamente, a política reativa com agentes desempenhando papéis diferentes, alguns priorizando a concentração enquanto outros priorizam zonas de sensibilidade. O caso V apresenta uma situação diferente dos demais, rodando apenas no cenário 1, envolve vários agentes com alguns apresentando maior lentidão de ciclo que os outros na viagem e na comunicação. A Tabela 5.1 resume a definição desses experimentos.

### 5.2.1 Estudos de caso I e II

Neste experimento foram usados 7 veículos, todos com a mesma configuração de parâmetros na política reativa. Os pesos utilizados estão apresentados na Tabela 5.2.

<sup>3</sup>Disponível em <https://github.com/EASY-SPARC/patrol-for-oil-client>

Caso	Cenário	Nº de agentes	Heterogeneidade	Tempo para ciclo
I	1	7	Não	Igual
II	2	7	Não	Igual
III	1	7	Sim	Igual
IV	2	7	Sim	Igual
V	1	10	Sim	Diferentes

Tabela 5.1: Resumo da configuração dos estudos de caso

Robô	$\kappa$	$\omega_{ci}$	$\omega_{si}$	$\omega_{di}$	$\omega_{ni}$
0	1	2,0	0,1	-0,5	0,5
1	1	2,0	0,1	-0,5	0,5
2	1	2,0	0,1	-0,5	0,5
3	1	2,0	0,1	-0,5	0,5
4	1	2,0	0,1	-0,5	0,5
5	1	2,0	0,1	-0,5	0,5
6	1	2,0	0,1	-0,5	0,5

Tabela 5.2: Parâmetros usados pela política reativa nos estudos de caso I e II

A escolha dos pesos foi tomada considerando prioridade em limpar a região, sem grande importância para zonas mais sensíveis.

No decorrer da simulação, principalmente nos momentos iniciais, pôde ser observado que, apesar da presença do algoritmo  $A^*$  computação da política de navegação, alguns veículos permaneciam compartilhando espaço em mesma célula. Isso aconteceu porque os veículos iniciavam no mesmo local e estavam sincronizados. Quando computavam o próximo passo, a informação dos vizinhos estava atrasada. Quando acontecia alguma mudança mais significativa no mapa de partículas, por conta da simulação rodando em plano de fundo, acontecia de veículos escolherem células diferentes. Essas mudanças aconteciam ao mesmo tempo em que ocorria o cálculo de movimentação de alguns veículos. Nessa situação, os veículos seguiam seus caminhos sem compartilhar o espaço outra vez pelo restante da simulação, por conta do  $A^*$ .

A figura 5.5 mostra o estado final das missões avaliadas, quando maior parte das células com partículas foram visitadas. O fato de todos veículos apresentarem a mesma configuração favoreceu que a maioria optasse por tomar decisões de movimentação parecidas. Esse fenômeno pode ser observado no vídeo do experimento e, principalmente, no estado final do segundo estudo de caso, mostrado na figura 5.5.

## 5.2.2 Estudos de casos III e IV

Neste experimento foram usados 7 veículos, dessa vez com grupos apresentando configurações diferentes de parâmetros na política reativa. Os pesos utilizados estão apresentados na Tabela 5.3. O projeto desse experimento lembra a configuração desenvolvida

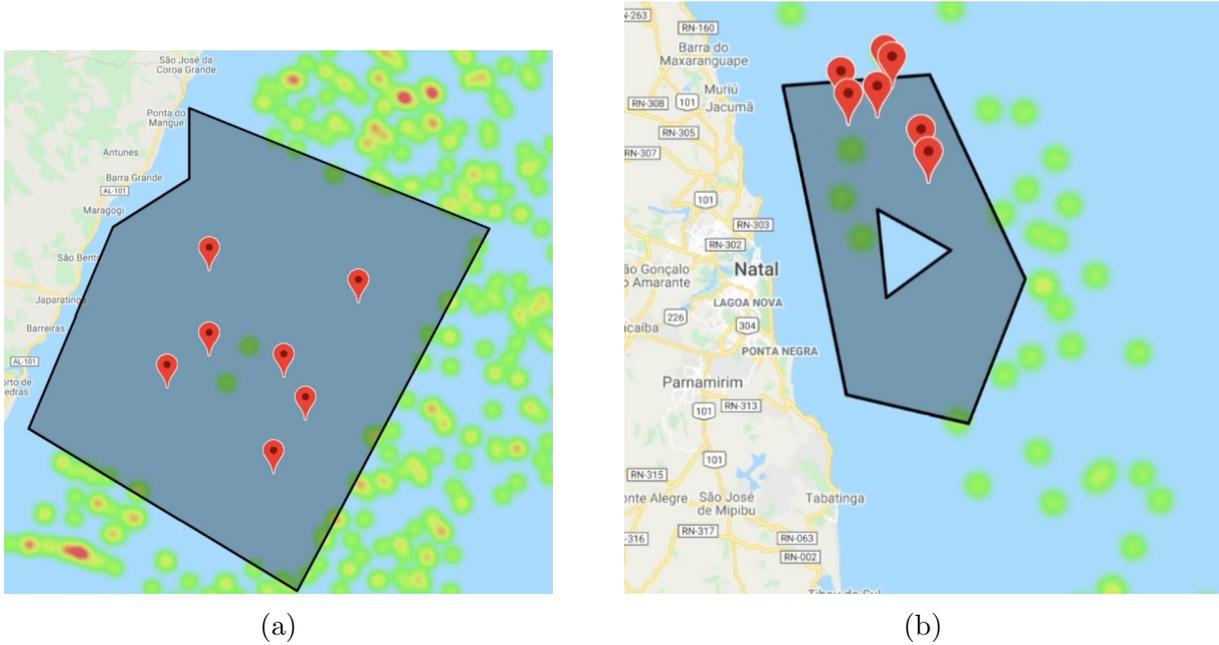


Figura 5.5: Estado final da área de missão no (a) estudo de caso I e no (b) estudo de caso II.

Robô	$\kappa$	$\omega_{ci}$	$\omega_{si}$	$\omega_{di}$	$\omega_{ni}$
0	2	2,0	0,1	-0,3	0,5
1	2	2,0	0,1	-0,3	0,5
2	2	2,0	0,1	-0,3	0,5
3	2	2,0	0,1	-0,3	0,5
4	2	0,3	2,0	-0,5	0,1
5	2	0,3	2,0	-0,5	0,1
6	2	0,3	2,0	-0,5	0,1

Tabela 5.3: Parâmetros usados pela política reativa nos estudos de caso III e IV

para a aplicação-teste da política adaptada de patrulhamento presente na figura 3.4.

A figura 5.6 apresentou o resultado da aplicação da política de patrulhamento, em conjunto com o *framework* desenvolvido nos cenários de Maragogi-AL e Natal-RN. Mais uma vez a característica de aglutinação de veículos com mesma configuração se deu de forma mais evidente no segundo cenário, visto na figura 5.6b.

Apesar de ainda persistir o problema de veículos ocupando o mesmo espaço no início das missões, o equilíbrio entre agentes priorizando aspectos diferentes do mapa provou ser uma estratégia melhor que a abordagem com parâmetros idênticos para todos veículos, mostrada neste trabalho. Foi observado que era necessário menos tempo para o mapa chegar em uma condição mais estável, com robôs se mexendo apenas para observar partículas recém-inseridas. De forma correlata, pôde ser percebido um mapa final de missão mais limpo, quanto à concentração de óleo provável.

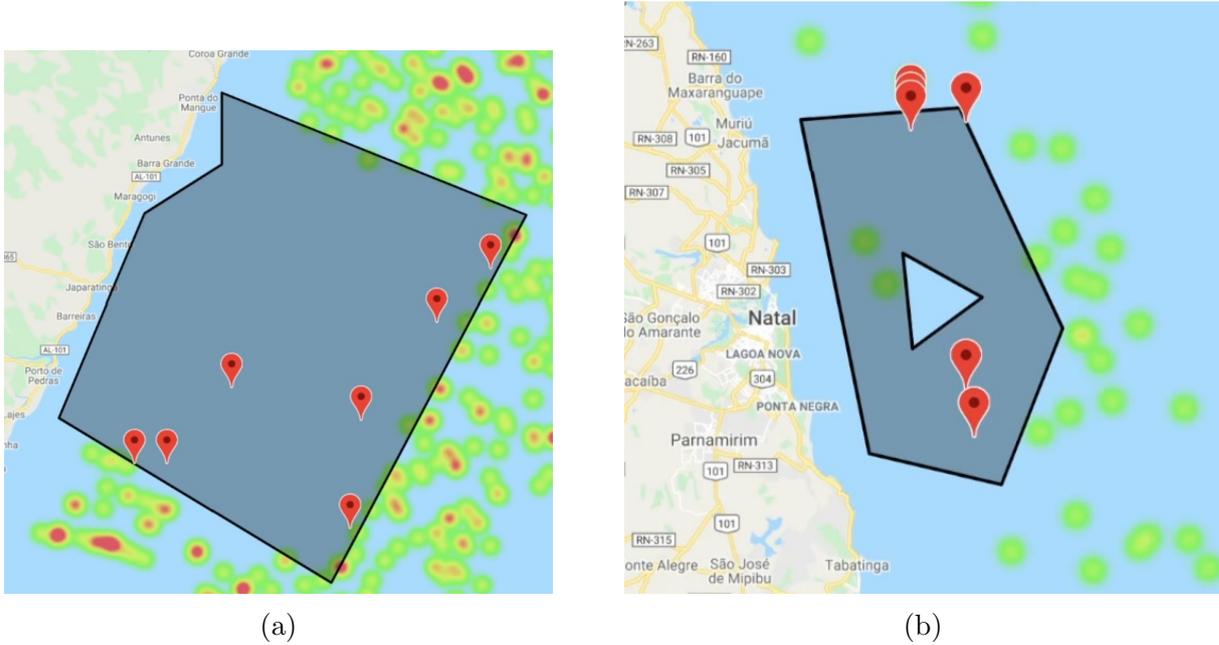


Figura 5.6: Estado final da área de missão no (a) estudo de caso III e no (b) estudo de caso IV.

Robô	$\kappa$	$\omega_{ci}$	$\omega_{si}$	$\omega_{di}$	$\omega_{ni}$	Tempo de ciclo
0	2,0	2,0	0,1	-0,1	0,5	3
1	2,0	3,0	0,5	-0,3	0,1	5
2	2,0	0,1	2,0	-0,8	0,5	2
3	2,0	0,4	2,3	-0,1	0,4	4
4	2,0	1,3	0,1	-0,2	0,1	3
5	2,0	0,1	1,0	-0,6	0,5	5
6	2,0	0,2	0,8	-0,5	0,1	6
7	2,0	0,5	1,7	-0,5	0,2	3
8	2,0	0,1	2,0	-0,3	0,5	3
9	2,0	2,0	2,0	-0,5	0,1	4

Tabela 5.4: Parâmetros usados pela política reativa no estudo de caso V

### 5.2.3 Estudo de caso V

Neste experimento foram usados 10 veículos, com ciclos operando em momentos distintos, com parâmetros escolhidos arbitrariamente, mas seguindo o princípio de focar em alta concentração de partículas ou sensibilidade ambiental, com exceção do último agente, que tenta considerar ambas situações igualmente. Por padrão, nos outros estudos de caso, foi usado mesmo tempo de ciclo de 3 segundos para todos agentes. Os pesos utilizados nesta etapa estão apresentados na Tabela 5.3, que, se diferenciando das demais tabelas, apresenta o tempo de ciclo (em segundos).

Essa configuração se assemelha mais de uma missão real do que os estudos de caso anteriores, uma vez que dificilmente agentes espalhados na missão apresentariam funcionamento sincronizado. Não obstante, essa abordagem favoreceu um maior espalhamento



# Conclusão

Este trabalho propôs um *framework* para desenvolvimento de sistemas de monitoramento oceânico usando veículos cooperativos. Esse sistema é separado em três módulos principais, uma estação central, uma interface *web* de configuração e visualização, e os veículos envolvidos na missão. Alguns serviços, como a simulação de partículas, se mantêm em execução permanente na central, outros são disponibilizados a partir de uma API, como as informações de posicionamento dos agentes, fazendo com que o sistema gerencie requisições assíncronas de agentes independentes.

Uma política reativa distribuída foi implementada e adaptada para o cenário estudado. Essa política permitiu aos veículos computar localmente suas estratégias de movimentação. Para isso, são adquiridas informações da região de interesse da missão, como localização outros agentes, índice de sensibilidade ambiental e estimação de concentração, com base nas simulações que executam permanentemente na estação central. Com isso, foi possível elaborar configurações de parâmetros únicos para grupos de agentes com prioridades diferentes, seja para priorizar movimentação para áreas de maior possibilidade de concentração de óleo, ou áreas com maior sensibilidade ao poluente,

O *framework* apresentado é capaz de trabalhar com outras estratégias de monitoramento e simuladores de transporte para demais substâncias, construindo aplicações diferentes. Entretanto, percebeu-se que configurar uma missão não é uma tarefa simples, pois uma região pode ser muito grande para um certo número de agentes, demandando muito tempo para a varredura, ou mesmo a configuração dos agentes pode torná-los muito conservadores para movimentação até células distantes.

Ao sincronizar observações *in situ* dos agentes, podem ser adquiridos registros reais da presença de óleo. Essa funcionalidade de reportar presença do poluente está implementada no sistema, entretanto, essa leitura é inserida na simulação com outras partículas do poluente hipotéticas, sem discernimento entre ambas. Outro trabalho futuro é desenvolver simulações paralelas para partículas de óleo hipotético e óleo observado, ou uma abordagem de simulação híbrida que englobe ambas as situações mantendo a diferenciação. Isso será capaz de proporcionar a utilização dessa informação nas estratégias de movimentação dos agentes, ou aplicação de outras classes de tarefas, como rastreamento ativo do poluente.

# Bibliografia

- RK Anderson, RS Scalan, PL Parker, and EW Behrens. Seep oil and gas in gulf of mexico slope sediment. *Science*, 222(4624):619–621, 1983.
- Jonny Beyer, Hilde C. Trannum, Torgeir Bakke, Peter V. Hodson, and Tracy K. Collier. Environmental effects of the Deepwater Horizon oil spill: A review. *Marine Pollution Bulletin*, 110(1):28–51, 2016. ISSN 0025326X. doi: 10.1016/j.marpolbul.2016.06.027. URL <http://dx.doi.org/10.1016/j.marpolbul.2016.06.027>.
- Holly M. Bik, Kenneth M. Halanych, Jyotsna Sharma, and W. Kelley Thomas. Dramatic shifts in benthic microbial eukaryote communities following the deepwater horizon oil spill. *PLoS ONE*, 7(6), 2012. ISSN 19326203. doi: 10.1371/journal.pone.0038550.
- Brasil. Lei nº 8.617, de 4 de janeiro de 1993. dispõe sobre o mar territorial, a zona contígua, a zona econômica exclusiva e a plataforma continental brasileiros, e dá outras providências. *Diário Oficial [da] República Federativa do Brasil*, 1993. ISSN 1677-7042. URL [http://www.planalto.gov.br/ccivil\\_03/leis/18617.htm](http://www.planalto.gov.br/ccivil_03/leis/18617.htm).
- CE Brown, MF Fingas, and R Marois. Oil spill remote sensing: laser fluorosensor demonstration flights off the east coast of canada. In *ARCTIC AND MARINE OILSPILL PROGRAM TECHNICAL SEMINAR*, volume 1, pages 317–334. Environment Canada; 1999, 2004.
- R. De la Huz, M. Lastra, and J. López. Oil spills. In J.O. Nriagu, editor, *Encyclopedia of Environmental Health*, pages 251 – 255. Elsevier, Burlington, 2011. ISBN 978-0-444-52272-6. doi: <https://doi.org/10.1016/B978-0-444-52272-6.00568-7>. URL <http://www.sciencedirect.com/science/article/pii/B9780444522726005687>.
- Norman C. Duke. Oil spill impacts on mangroves: Recommendations for operational planning and action based on a global review. *Marine Pollution Bulletin*, 109(2):700–715, 2016. ISSN 18793363. doi: 10.1016/j.marpolbul.2016.06.082. URL <http://dx.doi.org/10.1016/j.marpolbul.2016.06.082>.
- Matthew L. Elwin, Randy A. Freeman, and Kevin M. Lynch. Distributed Environmental Monitoring with Finite Element Robots. *IEEE Transactions on Robotics*, 36(2):380–398, 2020. ISSN 19410468. doi: 10.1109/TRO.2019.2936747.

- Merv Fingas and Carl E Brown. A review of oil spill remote sensing. *Sensors*, 18(1):91, 2018.
- Mervin Fingas. *Oil Spill Science and Technology – Prevention, Response, and Cleanup*. Gulf Professional Publishing, 2011.
- E. Gamma, R. Helm, R.E. Johnson, and J. Vlissides. *Design Patterns: Elements of Reusable Object-oriented Software*. Pearson Education, 2015. ISBN 9789332555402. URL <https://books.google.com.br/books?id=vqp2AQAACAAJ>.
- M. C. García Negro, S. Villasante, A. Carballo Penela, and G. Rodríguez Rodríguez. Estimating the economic impact of the Prestige oil spill on the Death Coast (NW Spain) fisheries. *Marine Policy*, 33(1):8–23, 2009. ISSN 0308597X. doi: 10.1016/j.marpol.2008.03.011.
- Ron Goodman. Overview and future trends in oil spill remote sensing. *Spill Science & Technology Bulletin*, 1(1):11–21, 1994.
- Michale Griebel, Thomas Dornseifer, and Tilman Neunhoeffler. *Numerical Simulation in Fluid Dynamics*. Society for industrial and Applied Mathematics, 1998.
- Héctor M Guzmán, Kathryn A Burns, and Jeremy BC Jackson. Injury, regeneration and growth of caribbean reef corals after a major oil spill in panama. *Marine Ecology Progress Series*, pages 231–241, 1994.
- Kai Hormann and Alexander Agathos. The point in polygon problem for arbitrary polygons. *Computational Geometry*, 20(3):131–144, 2001. ISSN 0925-7721. doi: [https://doi.org/10.1016/S0925-7721\(01\)00012-8](https://doi.org/10.1016/S0925-7721(01)00012-8). URL <https://www.sciencedirect.com/science/article/pii/S0925772101000128>.
- H. Ishida, Y. Wada, and H. Matsukura. Chemical sensing in robotic applications: A review. *IEEE Sensors Journal*, 12(11):3163–3173, 2012.
- Takeo Kajishima and Kunihiko Taira. Finite-difference discretization of the advection-diffusion equation. pages 23–72. Springer International Publishing, 2017. ISBN 9783319453040. doi: 10.1007/978-3-319-45304-0\_2. URL [http://link.springer.com/10.1007/978-3-319-45304-0\\_2](http://link.springer.com/10.1007/978-3-319-45304-0_2).
- Shuai Li, Yi Guo, and Brian Bingham. Multi-robot cooperative control for monitoring and tracking dynamic plumes. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 67–73. IEEE, 2014.
- Mark Masse. *REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces*. ”O’Reilly Media, Inc.”, 2011.

- Nikhil Nigam. The multiple unmanned air vehicle persistent surveillance problem: A review. *Machines*, 2(1):13–72, January 2014. doi: 10.3390/machines2010013. URL <https://doi.org/10.3390/machines2010013>.
- Nikhil Nigam and Ilan Kroo. Control and design of multiple Unmanned Air Vehicles for a persistent surveillance task. In *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, MAO*. American Institute of Aeronautics and Astronautics Inc., 2008. ISBN 9781563479472. doi: 10.2514/6.2008-5913.
- Nikhil Nigam, Stefan Bieniawski, Ilan Kroo, and John Vian. Control of Multiple UAVs for Persistent Surveillance: Algorithm and Flight Test Results. *IEEE Transactions on Control Systems Technology*, 20(5):1236–1251, sep 2012. ISSN 1063-6536. doi: 10.1109/TCST.2011.2167331. URL <http://ieeexplore.ieee.org/document/6045299/>.
- NOAA. What is an oil seep? <https://oceanservice.noaa.gov/facts/oilseep.html>, 2019. Online; accessed 14 April 2020.
- NOAA. Weathering processes affecting spills. <https://response.restoration.noaa.gov/oil-and-chemical-spills/oil-spills/weathering-processes-affecting-spills>, 2019. Online; Acessado 18 de Julho 2020.
- Barbara Ornitz and Michael Champ. *Oil spills first principles: prevention and best response*. Elsevier, 2002. ISBN 9780080537962.
- Emanuel Parzen. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065 – 1076, 1962. doi: 10.1214/aoms/1177704472. URL <https://doi.org/10.1214/aoms/1177704472>.
- W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, 2007. ISBN 9780521880688. URL <https://books.google.com.br/books?id=1aA0dzK3FegC>.
- Roger Proctor, Roger A Flather, and Alan J Elliott. Modelling tides and surface drift in the arabian gulf—application to the gulf oil spill. *Continental Shelf Research*, 14(5): 531–545, 1994.
- Nathan F. Putman, F. Alberto Abreu-Grobois, Iñaky Iturbe-Darkistade, Emily M. Putman, Paul M. Richards, and Philippe Verley. Deepwater Horizon oil spill impacts on sea turtles could span the Atlantic. *Biology Letters*, 11(12), 2015. ISSN 1744957X. doi: 10.1098/rsbl.2015.0596.

- Cyril Robin and Simon Lacroix. Multi-robot target detection and tracking: taxonomy and survey. *Autonomous Robots*, 40(4):729–760, apr 2016. ISSN 0929-5593. doi: 10.1007/s10514-015-9491-7. URL <http://link.springer.com/10.1007/s10514-015-9491-7>.
- David W Scott. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015.
- A. Vasilijevic, D. Nad, F. Mandic, N. Miškovic, and Z. Vukic. Coordinated navigation of surface and underwater marine robotic vehicles for ocean sampling and environmental monitoring. *IEEE/ASME Transactions on Mechatronics*, 22(3):1174–1184, 2017.
- Ann Walker. *Oil Spills and Risk Perceptions*, pages 1–70. Gulf Professional Publishing, 01 2017. ISBN 9780128094136. doi: 10.1016/B978-0-12-809413-6.00001-1.
- Jun-Wei Wang, Yi Guo, Muhammad Fahad, and Brian Bingham. Dynamic plume tracking by cooperative robots. *IEEE/ASME Transactions on Mechatronics*, PP:1–1, 01 2019. doi: 10.1109/TMECH.2019.2892292.
- Chengke Xiong, Di Lu, Zheng Zeng, Lian Lian, and Caoyang Yu. Path Planning of Multiple Unmanned Marine Vehicles for Adaptive Ocean Sampling Using Elite Group-Based Evolutionary Algorithms. *Journal of Intelligent & Robotic Systems*, 2020. ISSN 0921-0296. doi: 10.1007/s10846-020-01155-7.