

**EXPLORAÇÃO AUTÔNOMA DE AMBIENTES
BASEADA EM GANHO DE INFORMAÇÃO**

JHIELSON MONTINO PIMENTEL

**EXPLORAÇÃO AUTÔNOMA DE AMBIENTES
BASEADA EM GANHO DE INFORMAÇÃO**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

ORIENTADOR: DOUGLAS GUIMARÃES MACHARET
COORIENTADOR: MARIO FERNANDO MONTENEGRO CAMPOS

Belo Horizonte
Outubro de 2016

Ficha catalográfica elaborada pela Biblioteca do IEx - UFMG

Pimentel, Jhielson Montino.

P644e Exploração autônoma de ambientes baseada em ganho de Informação. / Jhielson Montino Pimentel. – Belo Horizonte, 2016.
xxxiv, 93 f.: il.; 29 cm.

Dissertação (mestrado) - Universidade Federal de Minas Gerais – Departamento de Ciência da Computação.

Orientador: Douglas Guimarães Macharet.

Coorientador: Mário Fernando Montenegro Campos

1. Computação - Teses. 2. Exploração autônoma. 3. Predição (Lógica). 4. Entropia (Teoria da informação). 5. Ganho de informação. I. Orientador. II. Coorientador. III. Título.

CDU 519.6*82.9 (043)



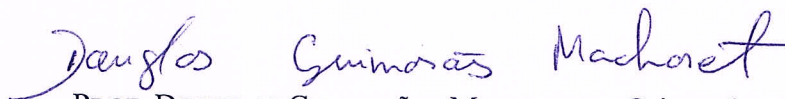
UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

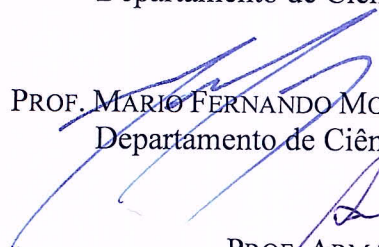
Exploração autônoma de ambientes baseada em ganho de informação

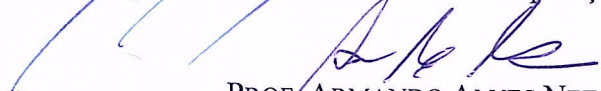
JHIELSON MONTINO PIMENTEL

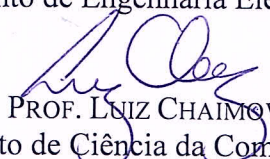
Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:




PROF. DOUGLAS GUIMARÃES MACHARET - Orientador
Departamento de Ciência da Computação - UFMG


PROF. MARIO FERNANDO MONTENEGRO CAMPOS - Coorientador
Departamento de Ciência da Computação - UFMG


PROF. ARMANDO ALVES NETO
Departamento de Engenharia Eletrônica - UFMG


PROF. LUIZ CHAIMOWICZ
Departamento de Ciência da Computação - UFMG


PROF. MARIO SÉRGIO FERREIRA ALVIM JÚNIOR
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 20 de outubro de 2016.

*Dedico este trabalho a minha família e aos colegas,
que tanto me ajudaram a superar os desafios na elaboração desse trabalho.*

Agradecimentos

Primeiramente, agradeço a Deus por ter sempre me iluminado nos desafios da vida, por ter me dado forças para prosseguir no mestrado, mesmo após a perda do meu pai, e por ter sempre me dado coragem para seguir em frente em um futuro tão incerto.

Agradeço aos meus pais, Carlos Q. Pimentel e Ana Célia dos S. M. Pimentel, e aos meus irmãos, Carlison M. Pimentel e Ana Carla M. Pimentel, por terem sempre sido minha base e meu suporte em todos os momentos da minha vida. Eu estando perto ou longe, estavam sempre ao meu lado com palavras de apoio, pensamentos positivos e no que eu mais precisasse.

Agradeço ao meu orientador, professor Douglas G. Macharet, e ao meu co-orientador, professor Mario F. M. Campos, pela paciência e pelas orientações durante todo o mestrado permitindo que essa etapa acadêmica fosse concluída com sucesso. Todas as reuniões e conversas me ajudaram não só no desenvolvimento do trabalho como também no aperfeiçoamento do meu perfil de pesquisador.

Agradeço aos colegas do VeRLab e do DCC, pelo apoio nos diversos momentos difíceis do mestrado, desenvolvimento de TPs e atividades do laboratório, além dos momentos de descontração dentro e fora da universidade. Em especial, a família Fleming que sempre se reunia nos momentos de trabalho e lazer.

Agradeço à CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pela concessão da bolsa durante todo o período de realização deste mestrado.

A todos que de alguma forma participaram desse processo, muito obrigado.

“Se queres prever o futuro, estuda o passado.”
(Confúcio)

Resumo

A exploração de ambientes desconhecidos usando robôs móveis e autônomos é essencial em diferentes aplicações reais, como na busca e resgate de vítimas após catástrofes. Isso devido à capacidade de atuação desses robôs em regiões desconhecidas mesmo quando desprovidos da intervenção humana. O principal objetivo dessa tarefa consiste em cobrir eficientemente o ambiente e construir um mapa completo e preciso. Contudo, diferentes aplicações podem demandar diferentes estratégias de exploração. A estratégia mais simples é a abordagem gulosa que busca visitar sempre as regiões mais próximas sem considerar uma redução significativa da incerteza do mapa. Nesse trabalho, foram propostas duas principais contribuições para a tarefa de exploração autônoma. Primeiro, foi elaborada uma nova abordagem para predição das regiões além das fronteiras através da análise das estruturas locais no mapa. Assim, pode ser definido o ganho estimado usando a entropia de Shannon para cada fronteira candidata a destino. Em seguida, a segunda contribuição foi desenvolvida com o objetivo de criar um planejamento unificado permitindo a identificação dos melhores destinos em conjunto com a criação das próprias rotas de navegação. A metodologia foi avaliada através de vários experimentos em um ambiente simulado, mostrando que a abordagem proposta é mais rápida que o método clássico de exploração conhecido como Exploração pelas Fronteiras Mais Próximas (NFE).

Palavras-chave: Exploração Autônoma, Predição, Ganho de Informação, Planejamento Unificado.

Abstract

The exploration of unknown environments using autonomous mobile robots is essential for different applications, for example, search and rescue missions after natural disasters. This is due to the performance capability of these robots in unknown environments even when the human intervention is deprived. The main objective of this task is to efficiently transverse the environment and build a complete and accurate map. However, different applications may demand different exploration strategies. The simplest strategy is a greedy approach which visits the closest frontier without considering whether it will yield a significant reduction in map uncertainty. In this dissertation, we proposed two main contributions. First, we elaborated a novel method to predict information beyond the candidate frontiers by analysing the local structures in a building map. In this way, it turns out possible to estimate the information gain of each frontier candidate to be the next destination using Shanon entropy. Afterwards, the second contribution was developed with the intention to create a unified planning which allows the robot to identify the best destinations and, at the same time, its own paths. The methodology was evaluated through several experiments in a simulated environment, showing that our exploration approach is better suited for rapid exploration than the classic Near-Frontier Exploration (NFE).

Keywords: Autonomous Exploration, Prediction, Information Gain, Unified Planning.

Lista de Figuras

1.1	Algumas aplicações da robótica. Robôs manipuladores: (a) linha de montagem. Robôs móveis terrestres: (b) exploração de planetas e (c) limpeza de casas. Robôs móveis aquáticos: (d) limpeza dos cascos dos navios [Souto et al., 2015]. Robôs móveis aéreos: (e) monitoramento de áreas remotas.	2
1.2	Exemplo de um mapa híbrido em construção. As imagens apresentam etapas na formação do mapa métrico e as inferiores apresentam etapas na formação do mapa topológico. As regiões em branco indicam as áreas livres para navegação; em preto, as áreas com obstáculos; e em cinza, as regiões ainda não exploradas. As cores no mapa topológico indicam cômodos diferentes do ambiente detectados pelo robô.	4
1.3	Exemplo de mapa construído usando um sensor de medição de temperatura por infravermelho para identificação de possíveis vítimas após catástrofe [Nagatani et al., 2012].	5
1.4	Na tarefa de exploração com restrição de tempo, um problema fundamental é a determinação das próximas configurações de destino com o objetivo de maximizar a cobertura do ambiente.	6
2.1	Exemplo de Malha de Ocupação criada ao término de uma exploração. As áreas em cinza representam regiões ainda inexploradas, com alta incerteza, e as regiões em branco e preto correspondem as regiões conhecidas e classificadas como livres e ocupadas, respectivamente.	11
2.2	Imagens do mapa gerado. a) Mapa geo-referenciado com o caminho percorrido pelo robô. b) Imagem obtida por satélite em comparação com o mapa gerado [Emter & Petereit, 2014].	12
2.3	Construção de uma representação 3D do ambiente. a) Mapa extraído das imagens aéreas com o trajeto do robô terrestre antes da correção e b) depois da correção da localização do agente. c) Modelo final 3D de uma das áreas mapeadas [Früh & Zakhor, 2003].	13

2.4	Correção da estimativa da localização do agente com base nos referências identificados localmente no ambiente. A partir dessas novas referências, a representação estimada do ambiente se aproxima à real [Durrant-Whyte & Bailey, 2006].	14
2.5	Trajetórias baseadas em estruturas concêntricas [Sim & Dudek, 2003; Sim et al., 2004].	16
2.6	Exemplo ilustrativo para identificação de fronteiras num mapa (Malha de Ocupação). Esquerda: o próprio mapa. Centro: grupos com todas as células livres em contato direto com células desconhecidas. Direita: as fronteiras formadas após ignorar pequenos grupos definidos pelas células previamente selecionadas [Yamauchi, 1997].	17
2.7	Etapas de uma exploração usando o algoritmo guloso baseado no custo de deslocamento do robô para cada ponto candidato a destino [Yamauchi, 1997].	18
2.8	Etapas para identificação do próximo destino durante uma exploração. (a) Observação do ambiente e identificação das fronteiras. (b) Gerar pontos candidatos a destino. (c) Analisar a área que representa o ganho de informação estimado para cada ponto candidato. (d) Definir o ponto de destino com base na função de utilidade [González-Baños & Latombe, 2002].	20
2.9	Processo para identificação da região com maior incerteza para exploração. (a) Comparação dos dois mapas gerados com base no alcance do sensor. (b) Identificação das regiões de fronteira. (c) Escolha do melhor destino com base na área formada além das fronteiras [Visser & Slamet, 2008].	21
2.10	Parte de um mapa de ocupação com uma janela deslizante (representada em vermelho) calculando a quantidade de informação (função de utilidade) de uma célula (representada em verde) com base nas informações dos seus vizinhos. O valor ‘-1’ representa áreas desconhecidas do ambiente e qualquer valor entre ‘0’ (áreas livres) e ‘1’ (áreas ocupadas) indicam o conhecimento do robô em relação a uma porção do ambiente [Colares & Chaimowicz, 2014].	22
2.11	Detecção das regiões de interesse em um mapa de ocupação contínuo (<i>Cave</i>). No mapa da esquerda, as cores em vermelho escuro representam as áreas em que o robô possui a maior certeza da existência de obstáculos e as cores em azul escuro representam as áreas com a maior certeza de serem livres para navegação. A variação entre essas cores indicam o nível de certeza sobre o mapa formado pelo robô. Em relação ao mapa da direita, a cor vermelha indica as regiões de maior interesse para navegação e a azul representa o oposto [Jadidi et al., 2015].	22

2.12	Mapa contruído com base nas inferências realizadas usando o modelo do sensor laser. Em vermelho estão representadas as áreas ainda desconhecidas ao robô. Em cor azul, as áreas livres. E, com a variação de cor entre azul para vermelho, as regiões com as inferências [Yang et al., 2013].	23
2.13	Etapas de uma exploração utilizando conhecimento prévio para melhor prever as áreas além das fronteiras [Ström et al., 2015]. (a) Mapa com a localização do robô. (b) Identificação da melhor predição com base nos mapas locais já existentes. (c) Construção de caminhos usando o Diagrama de Voronoi. (d) Identificação de um Loop Closure, ideal para corrigir o problema da incerteza na localização do agente. (e) Navegação pelo caminho gerado. (f) Confirmação da predição feita pelo método proposto.	24
2.14	Otimização da exploração com base num esboço do ambiente fornecido previamente ao robô [Oßwald et al., 2016]. (a) Esboço do ambiente a ser explorado com caminho otimizado. (b) Início da exploração e identificação de um obstáculo no caminho gerado. (c) Diagrama atualizado do espaço a ser explorado. (d) Novo caminho otimizado gerado.	25
2.15	Exemplo de propagação de onda do algoritmo <i>Wavefront Propagation</i> usando um único robô [Colares & Chaimowicz, 2015].	28
2.16	Exemplo de propagação de onda do algoritmo <i>Wavefront Propagation</i> usando multi-robôs [Kasperski et al., 2013].	28
2.17	Campos Potenciais usado para identificar possíveis Loop-Closure durante a tarefa de exploração [Jorge et al., 2015].	29
2.18	Exemplo da expansão de uma eRRT* [Vallve & Andrade-Cetto, 2015]. Nesse experimento, o robô iniciou a exploração no canto inferior esquerdo do mapa. A evolução da exploração pode ser analisada através dos pequenos círculos em cinza que demonstram a localização do agente da etapa inicial até a atual. Nesse instante a árvore foi criada para identificação dos melhores caminhos. A variação das cores nas rotas está relacionada com o nível de interesse das regiões para serem exploradas. As cores “quentes” (vermelho e laranja) indicam um peso maior e as “frias” (verde e azul) o oposto.	30
2.19	Apresentação do Diagrama de Voronoi formado com base no esboço do mapa obtido previamente junto com o próprio mapa em construção. No algoritmo proposto, cada fronteira ilustrada em verde é associada ao nó mais próximo do grafo, permitindo identificar os melhores destinos para exploração [Oßwald et al., 2016].	31

3.1	Distribuição da entropia para uma variável aleatória com espaço amostral binário. O eixo das abscissas indica a probabilidade da célula estar ocupada ou livre e o eixo das ordenadas o valor relativo a entropia dessa célula. . . .	35
3.2	Exemplo de uma árvore RRT expandida [Lavelle, 1998].	37
3.3	Expansão de uma árvore RRT em um processo de exploração. (a) Criação de uma amostra aleatória (x_{rand}) dentro do espaço de configurações livre do mapa e identificação do nodo da árvore (x_{near}) que se localiza mais próximo à amostra. (b) Adição de um novo nodo (x_{new}) à árvore com base na amostra aleatória e no nodo identificado como o mais próximo. (c) Finalização de uma etapa do ciclo de expansão da árvore com a anexação do novo nodo. (d) Caminho gerado após diversos ciclos de expansão partindo da localização do robô (x_{init}) e alcançando uma região ainda desconhecida do mapa. . . .	38
3.4	Representação de dois possíveis exemplos no uso de um sensor laser com alcance limitado. (a) demonstra o caso quando o laser detecta uma superfície dentro de seu alcance. A posição do robô no instante t é indicado por R_t e dos pontos detectados pelo sensor pelo conjunto $O_t = \{o_{t,0}, o_{t,1}, \dots, o_{t,u}\}$. Em (b) não foram detectados obstáculos e, assim, toda a área de cobertura do sensor é interpretada como livre.	40
3.5	Demonstração do ganho de informação baseado na predição das fronteiras. A imagem (a) corresponde ao mapa original e as imagens (b), (c) e (d) representam possíveis predições para cada fronteira com as rotas partindo sempre do robô para cada destino. As rotas para as fronteiras 2 e 3 irão trazer um ganho de informação limitado enquanto que a rota para a fronteira 1 irá trazer um ganho de informação maior, sendo ainda expansível. . . .	43
3.6	Representação dos dois possíveis modelos de propagação de paredes. A primeira coluna representa o mapa com a fronteira detectada e a segunda o mapa com a predição baseada na propagação das paredes. Na imagem (a), as paredes que limitam as fronteiras são identificadas e propagadas linearmente no mapa conforme imagem (b), enquanto que as paredes detectadas em (c) se propagam sendo imediatamente bloqueadas por uma região livre. Consequentemente, a direção da propagação é reconfigurada para as regiões desconhecidas conforme demonstrado em (d).	44

3.7	Dois passos necessários para projetar uma zona de predição com base nas dimensões da fronteira selecionada. (a) Ilustração de parte do mapa contendo a fronteira selecionada. (b) Primeiro, são determinadas as células mais extremas que irão formar uma estrutura geométrica agrupando todas as células dessa fronteira. (c) Em seguida, a estrutura é expandida uniformemente através de um valor fixo representado em verde. (d) Ilustração da Zona de Predição projetada sobre a fronteira selecionada.	45
3.8	Exemplo de predições realizadas após a detecção de quatro fronteiras no mapa em (c). Para as fronteiras em (a), (b) e (d), foram prolongadas as paredes uma de forma reta e outra em quina. Na fronteira em (e), as duas paredes foram prolongadas de forma reta.	46
3.9	Seleção por roleta. Os possíveis destinos são inseridos na roleta com seus respectivos pesos (Setor 1 = 40%, Setor 2 = 30%, Setor 3 = 19%, Setor 4 = 11%). O elemento selecionado será usado como viés na formação da árvore RRT. Nessa roleta não foi inserido um setor para um destino aleatório. . .	49
3.10	Função gerada para definir o tamanho do setor aleatório da roleta com base no número de fronteiras detectadas no mapa.	50
3.11	Diagrama representando o ciclo que compõe as principais etapas da metodologia proposta.	52
4.1	Mapas de referência para cada ambiente (20mx20m). As três posições iniciais para cada mapa estão representadas pelas letras A, B e C.	57
4.2	Identificação das fronteiras 1, 2 e 3 e criação das zonas de predição na etapa inicial de exploração para o ambiente Escritório.	59
4.3	Os ganhos de informação baseado nas predições. Em (a) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas predições realizadas. (b) Predição da fronteira 1. (c) Predição da fronteira 2. (c) Predição da fronteira 3.	59
4.4	Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os três destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto. O caminho final geral parte do robô para a fronteira de número 1.	60
4.5	Identificação das fronteiras 1, 2, 3 e 4 e criação das zonas de predição cinco minutos após iniciar a exploração.	60

4.6	Os ganhos de informação baseado nas predições. Em (c) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas predições realizadas. (a) Predição da fronteira 1. (c) Predição da fronteira 2. (d) Predição da fronteira 3. (e) Predição da fronteira 4.	61
4.7	Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os quatro destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto. O caminho final geral parte do robô para a fronteira de número 1.	62
4.8	Exemplos de mapas formados após 15 minutos de execução dos experimentos para cada estratégia.	63
4.9	Identificação das fronteiras 1, 2 e 3 e criação das zonas de predição trinta minutos após iniciar a exploração.	64
4.10	Os ganhos de informação baseado nas predições. Em (a) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas predições realizadas. (b) Predição da fronteira 1. (c) Predição da fronteira 2. (c) Predição da fronteira 3.	64
4.11	Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os três destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto. O caminho final geral parte do robô para a fronteira de número 3.	65
4.12	Identificação das fronteiras 1 e 2 e criação das zonas de predição na etapa inicial de exploração para o ambiente Cave.	66
4.13	Os ganhos de informação baseado nas predições. Em (a) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas predições realizadas. (b) Predição da fronteira 1. (c) Predição da fronteira 2. (c) Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os dois destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto.	66
4.14	Identificação das fronteiras 1, 2, 3 e 4 e criação das zonas de predição cinco minutos após iniciar a exploração.	67

4.15	Imagens formadas com as predições, os valores dos ganhos de informação e o caminho gerado para exploração. Em (a) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas predições realizadas. (b) Predição da fronteira 1. (c) Predição da fronteira 2. (c) Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os quatro destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto.	68
4.16	Exemplos de mapas formados após 15 minutos de experimentos para cada estratégia.	69
4.17	Identificação das fronteiras 1 e 2 e criação das zonas de predição na etapa final de exploração.	70
4.18	Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os dois destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto. O caminho final geral parte do robô para a fronteira de número 2.	70
4.19	Gráfico com a média do percentual de completude do mapa pelo tempo para a posição A do mapa Escritório.	72
4.20	Deslocamento total do agente durante a tarefa de exploração partindo da posição A do mapa Escritório.	73
4.21	Gráfico com a média do percentual de completude do mapa pelo tempo para a posição B do mapa Escritório.	73
4.22	Deslocamento total do agente durante a tarefa de exploração partindo da posição B do mapa Escritório.	74
4.23	Gráfico com a média do percentual de completude do mapa pelo tempo para a posição C do mapa Escritório.	74
4.24	Deslocamento total do agente durante a tarefa de exploração partindo da posição C do mapa Escritório.	75
4.25	Gráfico com a média do percentual de completude do mapa pelo tempo para a posição A do mapa Caverna.	76
4.26	Deslocamento total do agente durante a tarefa de exploração partindo da posição A do mapa Caverna.	76
4.27	Gráfico com a média do percentual de completude do mapa pelo tempo para a posição B do mapa Caverna.	77
4.28	Deslocamento total do agente durante a tarefa de exploração partindo da posição B do mapa Caverna.	77

4.29	Gráfico com a média do percentual de completude do mapa pelo tempo para a posição C do mapa Caverna.	78
4.30	Deslocamento total do agente durante a tarefa de exploração partindo da posição C do mapa Caverna.	78
4.31	Exemplos de fronteiras em que o ganho estimado de informação foi reduzido devido ao prolongamento das paredes. As imagens do lado esquerdo representam partes do mapa com as zonas de predição projetadas e as imagens do lado direito exemplos de fronteiras em que existiu uma limitação da expansão das áreas livres pelo prolongamento das paredes.	79
4.32	Exemplos de mapas após a exploração do mesmo ambiente com diferente taxas de erro.	80
4.33	Comparação entre experimentos de uma abordagem com e sem erro no sensor de odometria.	81
4.34	Mapa formado com qualidade baixa na etapa final de uma exploração. . .	82

Lista de Tabelas

2.1 Tabela comparativa com alguns dos principais trabalhos que abordam a exploração autônoma de ambientes na literatura. (Símbolo '-': não mencionado explicitamente.)	32
--	----

Lista de Algoritmos

1	RRT($x_{init}, K, \Delta t$) [Lavalle, 1998]	39
2	CalcularGanhosDeInformacao(F, M)	48
3	SelecionarAmostra(F, GI)	51
4	ExploracaoAutonoma(n)	53

Lista de Acrônimos

EKF	<i>Extended Kalman Filter</i> (Filtro de Kalman Estendido)
GPS	<i>Global Positioning System</i> (Sistema de Posicionamento Global)
IG	<i>Information Gain</i> (Ganho de Informação)
IMU	<i>Inertial Measurement Unit</i> (Unidade de Medida Inercial)
KF	<i>Kalman Filter</i> (Filtro de Kalman)
MCDM	<i>Multiple Criteria Decision Making</i> (Tomada de Decisão Multicritério)
MI	<i>Mutual Information</i> (Informação Mútua)
NFE	<i>Near-Frontier Exploration</i> (Exploração pela Fronteira Mais Próxima)
RBPF	<i>Rao-Blackwelized Particle Filter</i> (Filtro de Partículas Rao-Blackwelized)
ROS	<i>Robot Operating System</i> (Sistema Operacional de Robôs)
RRT	<i>Rapidly-Exploring Random Tree</i> (Árvore Aleatória de Exploração Rápida)
SLAM	<i>Simultaneous Localization and Mapping</i> (Mapeamento e Localização Simultâneos)

Lista de Símbolos

\mathcal{E}	Ambiente a ser explorado
ϕ_i	i -ésima Zona de Predição
\mathcal{T}	Árvore RRT
F_t	Conjunto de fronteiras detectadas no instante t
$I(.)$	Ganho de Informação ou Informação Mútua
$H(.)$	Entropia de uma variável aleatória
$p(.)$	Probabilidade de um evento
$P(.)$	Distribuição de probabilidade de uma variável aleatória
M	Mapa no modelo Malha de Ocupação
m_{ij}	Célula de um mapa no modelo Malha de Ocupação
R_t	Posição do robô no \mathcal{E} no instante t
$O_{t,i}$	Posição de parte de um obstáculo no \mathcal{E} no instante t
O_t	Conjunto de posições referente aos obstáculos no \mathcal{E} no instante t
x_{init}	Localização do robô no mapa e raiz da árvore RRT
x_{rand}	Amostra aleatória gerada na formação da RRT
x_{near}	Nó da RRT mais próximo da amostra gerada
x_{new}	Novo nó anexado a RRT
X	Variável aleatória
Y	Variável aleatória
t	Instante de tempo
Z_i	i -ésima observação realizada pelo robô

Sumário

Agradecimentos	ix
Resumo	xiii
Abstract	xv
Lista de Figuras	xvii
Lista de Tabelas	xxv
Lista de Algoritmos	xxvii
Lista de Acrônimos	xxix
Lista de Símbolos	xxxi
1 Introdução	1
1.1 Contextualização	1
1.2 Motivação	3
1.3 Problema	5
1.4 Objetivo	7
1.5 Contribuições	8
1.6 Organização do Texto	8
2 Revisão Bibliográfica	9
2.1 Mapeamento	9
2.1.1 Mapeamento com o Uso de Sensores de Localização	11
2.1.2 Mapeamento sem o Uso de Sensores de Localização	13
2.2 Estratégias para Exploração Autônoma de Ambientes	15
2.2.1 Estratégias sem Avaliação dos Melhores Destinos	15

2.2.2	Estratégias com Avaliação dos Melhores Destinos	16
2.3	Navegação	27
2.3.1	Wavefront Propagation	27
2.3.2	Campos Potenciais Artificiais	28
2.3.3	Rapidly-exploring Random Trees	29
2.3.4	Outros Algoritmos	31
2.4	Contextualização do Trabalho	31
3	Metodologia	33
3.1	Fundamentos e Formalização	33
3.1.1	Entropia	33
3.1.2	Rapidly-exploring Random Tree	36
3.1.3	Formalização do Problema	39
3.2	Exploração Autônoma	42
3.2.1	Predição Baseada nas Fronteiras	43
3.2.2	Seleção e Planejamento de Caminho	48
3.2.3	Sistema	51
4	Experimentos	55
4.1	Procedimento dos Experimentos	56
4.2	Exemplo Ilustrativo	58
4.2.1	Escritório	58
4.2.2	Caverna	65
4.3	Comparativo Geral	70
4.4	Comparativo da Metodologia	71
4.4.1	Escritório	71
4.4.2	Caverna	74
4.5	Comparativo: Maiores Fronteiras vs. Maiores Ganhos Estimados de Informação	79
4.6	Exemplo de Exploração com Erro na Construção do Mapa	80
5	Conclusão e Trabalhos Futuros	83
5.1	Conclusão	83
5.2	Trabalhos Futuros	84
	Referências Bibliográficas	87

Capítulo 1

Introdução

O uso de robôs móveis e autônomos tem sido recorrente em inúmeros trabalhos relacionados à tarefa de exploração de ambientes. Isso devido à capacidade de atuação desses robôs em regiões desconhecidas mesmo quando desprovidos da intervenção humana. Ou seja, utilizando apenas as informações extraídas localmente pelo próprio agente. Durante a exploração, é importante a construção de uma representação do espaço físico para que as futuras tarefas atribuídas ao robô possam ser executadas adequadamente. No entanto, nem sempre o tempo alocado para exploração é amplo. Assim, o agente deve navegar de forma inteligente, garantindo uma representação o mais completa possível nos minutos iniciais.

Este trabalho apresenta uma nova abordagem sobre exploração autônoma de ambientes desconhecidos capaz de cobrir as regiões inexploradas de forma mais rápida gerando mapas com uma maior abrangência.

1.1 Contextualização

Nas últimas décadas, a Robótica vem evoluindo e alcançando diversos setores da sociedade. Um dos primeiros grandes avanços foi no início da década de 1960 quando George Devol projetou o primeiro modelo de um robô manipulador [Devol, 1961]. Esse modelo foi rapidamente atribuído ao setor da indústria acelerando as linhas de produção (Figura 1.1a). No entanto, os robôs manipuladores eram fixos a uma base, podendo executar apenas atividades repetitivas dentro do próprio campo de alcance [Siegwart & Nourbakhsh, 2004]. Com o surgimento dos robôs móveis, essa limitação foi superada e diversas aplicações surgiram.

A capacidade de locomoção dos robôs móveis permitiu a inserção dos agentes em novos cenários e na realização de tarefas com diferentes níveis de complexidade,

desde a simples limpeza de casas (Figura 1.1c) [Elara et al., 2014] até a exploração de outros planetas (Figura 1.1b) [Husain et al., 2013]. No entanto, para execução dessas tarefas o robô precisa de uma certa autonomia, sendo que os primeiros robôs móveis comercializados eram bem restritos nesse parâmetro. Muitos ainda dependiam completamente do homem para o funcionamento adequado das atividades.



Figura 1.1: Algumas aplicações da robótica. Robôs manipuladores: (a) linha de montagem. Robôs móveis terrestres: (b) exploração de planetas e (c) limpeza de casas. Robôs móveis aquáticos: (d) limpeza dos cascos dos navios [Souto et al., 2015]. Robôs móveis aéreos: (e) monitoramento de áreas remotas.

É oportuno lembrar que, a princípio, um robô móvel controlado pelo homem perdia toda sua funcionalidade quando os canais de comunicação entre o próprio agente e o operador eram instáveis ou inexistentes. Com a perda da comunicação, o robô era incapaz de executar as operações com sucesso. Buscando solucionar esse problema, foi observada a necessidade dos agentes planejarem e executarem as tarefas de forma autônoma. Com o avanço dos sensores e atuadores, diversas estratégias surgiram e novos sistemas foram desenvolvidos possibilitando aos robôs uma interferência humana cada vez menor. A limpeza de cascos de navios (Figura 1.1d) [Souto et al., 2015] e a vigilância aérea (Figura 1.1e) [Lee et al., 2015] são aplicações reais que já fazem uso desses robôs na execução de suas tarefas.

Os robôs móveis e autônomos pertencem a um ramo da Robótica em que é fundamental o uso dos sensores para perceber o ambiente e determinar as ações que serão executadas. Normalmente, tais ações correspondem à navegação do agente a uma

configuração de destino ou manipulação de objetos dentro do espaço de configurações livre. Essa categoria da Robótica é bastante ampla e ainda apresenta grandes desafios na comunidade científica.

Entre as aplicações que utilizam robôs móveis e autônomos, o processo de exploração é definido como indispensável. Isso devido a constante inserção desses robôs em ambientes desconhecidos para a execução das tarefas desejadas. A exploração garante ao robô a obtenção de uma representação do ambiente que servirá como importante ferramenta para as atividades seguintes. Dessa forma, a partir do mapa gerado, o robô poderá planejar ações de forma mais precisa, tornando possível a execução até mesmo de tarefas mais complexas, como o transporte de objetos em ambientes não estruturados [Melo et al., 2015]. Nesse sentido, uma boa exploração e, conseqüentemente, um bom mapa deve garantir melhores resultados para as tarefas atribuídas ao agente.

1.2 Motivação

A exploração de ambientes desconhecidos utilizando robôs móveis e autônomos é uma tarefa importante, como mencionado anteriormente. Para a execução dessa tarefa, o agente deve ser capaz de realizar as seguintes subtarefas: localização, mapeamento, planejamento de trajetórias e navegação.

As etapas de localização e mapeamento são fundamentais para o funcionamento adequado do processo de exploração. Com o conhecimento da localização do agente a construção do mapa se torna mais simples, no entanto, tal informação normalmente não é fornecida. Dessa forma, a localização precisa ser identificada com base no mapa em construção. Contudo, o mapa também depende da localização do agente para ser construído. Esse paradigma conhecido como SLAM (Localização e Mapeamento Simultâneos) tem sido bastante estudado na comunidade da Robótica Móvel e diversos algoritmos já foram desenvolvidos para solucionar esse problema. Assim, é possível atualmente determinar a localização do agente e mapear o ambiente de forma simultânea mesmo em áreas remotas sem acesso a informação externa.

O mapeamento citado acima consiste na composição de um modelo que descreve um ambiente físico utilizando um ou mais agentes móveis. O modelo ou mapa pode ser projetado usando uma representação métrica, topológica ou híbrida. Como pode ser visto na Figura 1.2, a principal diferença entre as representações é que a métrica agrupa as informações espaciais capturadas do ambiente e a topológica registra a conectividade entre as diferentes regiões. Os ambientes físicos mapeados são classificados como internos (*indoor*), e.g. casas e escritórios, ou externos (*outdoor*), e.g. campos e cidades. De

modo geral, os ambientes internos são áreas planas e de menor complexidade, podendo ser mapeado em 2D (duas dimensões), enquanto que os ambientes externos possuem maior complexidade necessitando uma representação em 3D (três dimensões).

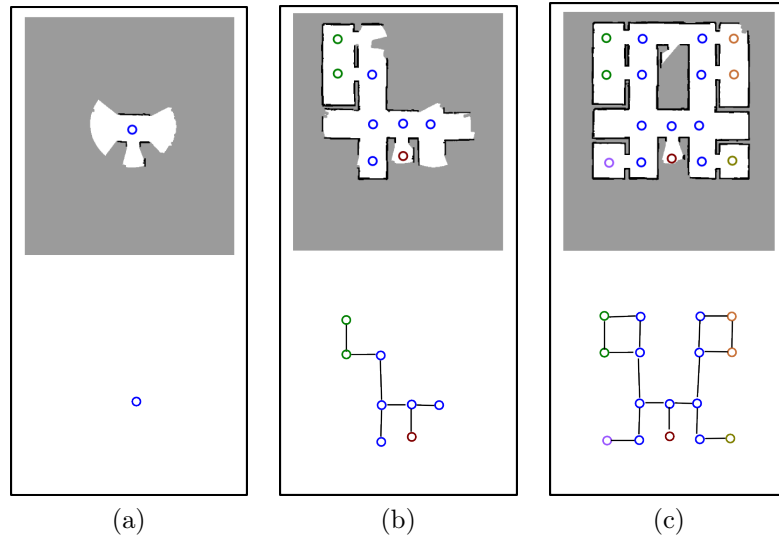


Figura 1.2: Exemplo de um mapa híbrido em construção. As imagens apresentam etapas na formação do mapa métrico e as inferiores apresentam etapas na formação do mapa topológico. As regiões em branco indicam as áreas livres para navegação; em preto, as áreas com obstáculos; e em cinza, as regiões ainda não exploradas. As cores no mapa topológico indicam cômodos diferentes do ambiente detectados pelo robô.

O mapa gerado é um componente necessário na execução de tarefas básicas, como no planejamento de caminhos otimizados e no desvio de obstáculos. No entanto, tarefas mais complexas também precisam de um mapa, como na inspeção de fábricas e no patrulhamento de instalações. A depender de como o mapa será utilizado, a estratégia de exploração poderá ser diferente. Por exemplo, algumas aplicações necessitam de um mapa rico em detalhes enquanto que outras apenas precisam de uma representação superficial do ambiente.

De modo geral, segundo Huang & Gupta [2008], as estratégias para exploração autônoma são formadas por um laço que consiste basicamente em três etapas a cada iteração: I) **planejar** a próxima configuração consistente do agente de forma a ampliar a região explorada do mapa em construção; II) **executar** o planejamento, por exemplo, mover o robô da configuração atual até a planejada e extrair novas informações do ambiente através dos sensores; e III) **atualizar** o modelo do ambiente e a pose (posição e orientação) do próprio agente.

O planejamento citado acima consiste na etapa mais importante quando se deseja definir uma estratégia para exploração. Por exemplo, no resgate de vítimas em um

edifício após um terremoto (Figura 1.3), é importante que o agente explore o prédio de forma rápida e eficiente. Mas para que isso seja possível, o agente deve identificar as regiões de maior interesse autonomamente. Desse modo, considerando que o tempo de exploração pode ser limitado, o robô deve planejar uma sequência de destinos ótimos visando construir um mapa que seja o mais abrangente possível sem se preocupar com detalhes que carregam pouca informação. Esse mapa pode ser usado como guia no resgate das vítimas por uma equipe de profissionais treinados para o resgate.

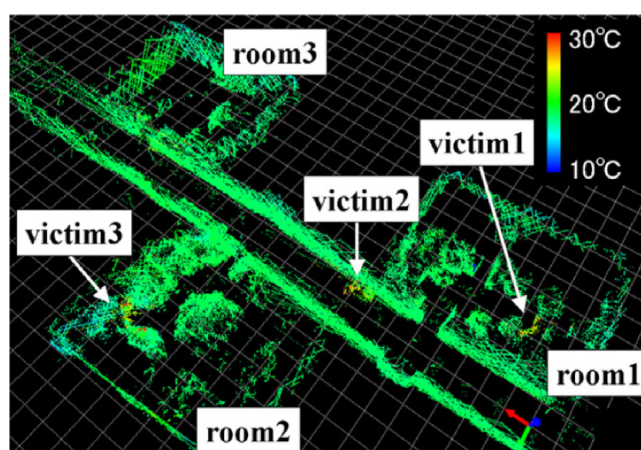


Figura 1.3: Exemplo de mapa construído usando um sensor de medição de temperatura por infravermelho para identificação de possíveis vítimas após catástrofe [Nagatani et al., 2012].

Se não houvesse uma restrição no tempo, o robô realizaria a tarefa com mais detalhes cobrindo todo o ambiente a ser explorado. Desse modo, análises mais complexas poderiam ser efetuadas e mapas ricos em informação poderiam ser obtidos. Contudo, uma exploração rápida e parcial deve garantir um número maior de vítimas resgatadas vivas em uma situação de catástrofe. Isso porque diversos fatores reduzem as chances de sobrevivência das vítimas com o passar do tempo, como gases tóxicos, incêndio e desmoronamento.

1.3 Problema

A exploração autônoma de ambientes consiste em um grande desafio devido a diversidade de cenários em que a tarefa pode ser aplicada. Fatores como a irregularidade do terreno, elementos dinâmicos e espaços desestruturados podem gerar barreiras na execução de uma exploração. Com o objetivo de otimizar o tempo, o robô ainda precisa tomar as melhores decisões possíveis de forma rápida na seleção das rotas para navegação.

De forma bem simples, o problema enfrentado nesse trabalho consiste em encontrar uma estratégia que permita ao agente melhorar a seleção dos destinos para a navegação do robô de forma que o mapa seja expandido rapidamente nos minutos iniciais da exploração (Figura 1.4). Desse modo, o robô ao ser inserido num ambiente completamente desconhecido deve autonomamente perceber esse ambiente, navegar e gerar uma representação o mais abrangente possível.

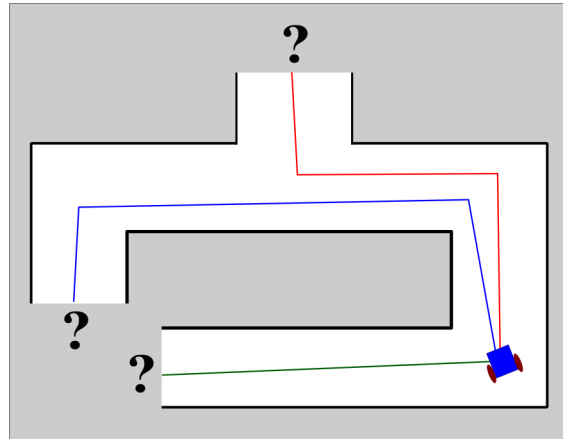


Figura 1.4: Na tarefa de exploração com restrição de tempo, um problema fundamental é a determinação das próximas configurações de destino com o objetivo de maximizar a cobertura do ambiente.

Em outras palavras, uma definição, ainda que superficial, sobre o problema é apresentada a seguir:

Dado um robô móvel com um sensor de alcance limitado, um mapa inicial do ambiente com a identificação das áreas livres e a própria localização do robô, selecionar e planejar um caminho até as regiões inexploradas de forma a maximizar o ganho de conhecimento sobre o ambiente rapidamente.

Muitas estratégias abordam a exploração autônoma selecionando as configurações de destino através de critérios que avaliam as regiões do ambiente que representam a transição entre as áreas livres para navegação e as desconhecidas ao robô. Tais regiões de transição são definidas como fronteiras. A concepção de que a fronteira mais próxima ou a de maior dimensão irá trazer sempre maior ganho de informação é falha e isto já foi demonstrado em alguns trabalhos na comunidade [Juliá et al., 2012]. No entanto, tais critérios ainda são importantes no processo global de avaliação e escolha dos destinos.

Outras estratégias mais elaboradas buscam avaliar a redução da entropia do mapa durante a exploração [Jadidi et al., 2015; Colares & Chaimowicz, 2016]. Assim, as regiões com o maior nível de incerteza devem atrair o robô durante a navegação. Os

valores de incerteza do mapa são reduzidos quando os sensores do robô cobrem as regiões inexploradas do ambiente. Com base em cálculos probabilísticos, todo o ambiente é marcado com valores que representam o grau de conhecimento do robô sobre o espaço. Se o agente consegue identificar e classificar o espaço como uma região livre ou ocupada para navegação essa região será definida com uma baixa incerteza. A incerteza sobre uma região é alta quando o robô ainda não possui informação suficiente para inferir uma possível classificação desse espaço. Essa abordagem é bastante interessante, no entanto, poucos trabalhos publicados na comunidade buscam inferir a representação das regiões além das fronteiras. É comum realizarem apenas uma estimativa simplificada sobre a incerteza nas próprias fronteiras para seleção dos melhores destinos a serem explorados.

Uma estimativa das regiões além das fronteiras é muito importante na tomada de decisão do robô. Apesar de ser um conhecimento local, essa informação pode trazer um grande avanço em diversas etapas da exploração. Atualmente, alguns trabalhos já buscam realizar previsões nessas áreas [Ström et al., 2015; Oßwald et al., 2016]. No entanto, a maioria das estratégias são muito simples ou aplicadas a ambientes extremamente específicos. Portanto, é necessário o desenvolvimento de estratégias capazes de identificar com maior precisão o ganho de informação estimado além das fronteiras e, em conjunto com os outros critérios, determinar caminhos para explorar as regiões de maior interesse.

1.4 Objetivo

O objetivo principal dessa dissertação consiste no desenvolvimento de uma nova abordagem para otimizar o problema da exploração autônoma de ambientes usando um robô móvel. Nessa abordagem, é esperado uma constante maximização do conhecimento do ambiente durante o processo de exploração. Ou seja, quando o agente precisa expandir a representação ou modelo do ambiente rapidamente. Será utilizada uma técnica baseada em previsão para estimar o ganho de informação das regiões ainda inexploradas e, assim, junto com outros critérios, definir os caminhos livres de colisão para serem navegados. Nesse trabalho, os ambientes de teste serão estáticos, planos e com obstáculos. Assim, o trabalho terá um maior foco nas estratégias aplicadas a identificação e seleção das regiões de maior interesse para exploração e na geração de rotas para alcançar tais regiões.

1.5 Contribuições

Nessa seção são apresentadas em duas etapas as contribuições obtidas de acordo com os principais desafios enfrentados nesse trabalho. Os conceitos e algoritmos apontados na contribuição serão apresentados de forma mais detalhada no Capítulo 3.

- **Predição das Áreas Desconhecidas:** Desenvolvimento de uma estratégia com o objetivo de estimar o ganho de informação das áreas além das fronteiras. A representação dessas regiões permite não apenas melhorar na estimativa do grau de incerteza dessas áreas como também permite identificar regiões que não são de interesse e que devem ser visitadas posteriormente. O robô deixando de se deslocar para tais regiões de baixo interesse pode garantir um bom ganho no tempo de exploração sem alterar muito a completude do mapa.
- **Planejamento Unificado:** É apresentada uma adaptação do algoritmo *Rapidly-exploring Random Tree* (RRT) [Lavalle, 1998] bastante importante para a seleção das melhores fronteiras na tarefa de exploração. O algoritmo difere do tradicional na geração das amostras usadas para a expansão da árvore. Nesse trabalho, a seleção das amostras é baseada numa técnica conhecida como Seleção por Roleta [Goldberg, 1989], muito comum nos trabalhos que envolvem Algoritmos Genéticos. Assim, as rotas para as melhores fronteiras são geradas em conjunto com a seleção das próprias fronteiras.

Durante o desenvolvimento do projeto, parte da metodologia proposta (Predição das Áreas Desconhecidas) foi publicado em um artigo em uma conferência internacional.

- Jhielson M. Pimentel, Douglas G. Macharet e Mario F. M. Campos. *Information-theoretic Frontier Selection for Environment Exploration. 4th Brazilian Symposium on Robotics and 13th Latin American Robotics Symposium (SBR-LARS)*, 2016. [Qualis-CC B4]

1.6 Organização do Texto

O Capítulo 2 apresenta uma revisão dos principais trabalhos relacionados. No Capítulo 3, o problema e suas restrições são inicialmente formalizados, em seguida, é apresentada a etapa de predição junto com a estratégia de planejamento unificado para detecção das melhores rotas de exploração. O Capítulo 4 apresenta os experimentos desenvolvidos e resultados obtidos para validar a metodologia proposta. E, por fim, o Capítulo 5 discute as conclusões e apresenta possíveis direções para os trabalhos futuros.

Capítulo 2

Revisão Bibliográfica

A construção de mapas é uma tarefa extremamente desafiadora. Os ambientes reais são complexos, desestruturados e, em alguns casos, possuem elementos dinâmicos. Além disso, é comum o uso de sensores mais baratos, com uma precisão reduzida, o que intensifica a dificuldade de mapear. Os principais trabalhos nessa área de pesquisa visam o desenvolvimento de algoritmos capazes de construir representações completas e precisas de ambientes desconhecidos. Contudo, nem sempre um modelo completo é necessário. Em alguns casos, é importante que o agente saiba identificar e explorar apenas as principais áreas de interesse do ambiente.

A seguir, serão apresentadas e discutidas algumas técnicas sobre mapeamento de ambientes e estratégias aplicadas a exploração autônoma encontradas em trabalhos na literatura.

2.1 Mapeamento

O mapeamento na Robótica consiste na tarefa responsável por obter modelos de ambientes físicos. Para a formação desse modelo, diversos tipos de sensores, como câmera, laser, sonar, bússola e GPS, podem ser usados para equipar o robô e permitir uma percepção apropriada do ambiente. Durante o mapeamento, o robô precisa realizar diversas observações extraindo informações importantes para a formação do mapa. A depender do sensor, algumas observações podem ser bastante limitadas no espaço e outras podem possuir alta taxa de ruído. Dessa forma, os sensores possuem uma importante influência na escolha das técnicas relacionadas ao mapeamento.

A tarefa de mapear utilizando um agente móvel e autônomo pode ser realizada com ou sem o conhecimento direto da localização do robô. No melhor cenário, uma localização sem erros possibilitaria a construção de um mapa a partir dos dados brutos

coletados por um sensor laser. O mapa seria facilmente expandido a partir da detecção dos novos obstáculos e o conhecimento do posicionamento desses elementos em relação a localização do agente. Contudo, a obtenção direta da localização do agente com alta precisão pode ser inviável para muitos ambientes, como cavernas, edifícios e até mesmo algumas áreas abertas. Sem essa informação, a construção de um modelo espacial se torna mais complexa. No entanto, ainda é possível mapear o ambiente de forma apropriada utilizando técnicas que permitem o refinamento da localização do agente. Por exemplo, um receptor GPS (*Global Positioning System*) não garante a precisão necessária para um mapeamento direto sem erros, mas com a fusão dos dados coletados com os de outros sensores é possível corrigir o erro de localização e, assim, gerar um mapa apropriado.

Em casos que o robô não possui sensores que capturam diretamente a própria localização, o mapeamento torna-se ainda mais complexo. Desse modo, o robô precisa gerar um mapa para se localizar e se localizar com base no mapa gerado. Esse paradigma é bastante conhecido e tem sido amplamente estudado na comunidade da Robótica Móvel nos últimos anos. Atualmente, já existem diversas soluções para esse problema. As duas principais vertentes serão brevemente apresentadas na Seção 2.1.2.

Além dos sensores, é importante definir como se deseja mapear o ambiente, qual modelo deve ser usado para representá-lo. Assim, o mapa pode ser definido em três categorias: métrico, topológico e híbrido. A categoria do mapa métrico possui uma representação do espaço bastante popular conhecida como Malha de Ocupação (*Occupancy Grid*) (Figura 2.1) [Moravec & Elfes, 1985]. A escolha dessa representação é importante pois permite o armazenamento das informações probabilísticas obtidas em observações passadas pelo robô. Desse modo, cada célula do mapa representa uma região do ambiente real e armazena um valor $p(x)$ que denota a probabilidade agregada da região estar ocupada, por exemplo ser parte de um obstáculo que inviabiliza a navegação do robô. Assim, quando uma célula possui um valor próximo a zero ou a um, significa que o nível de incerteza é pequeno, pois já é de conhecimento do robô que essa região representa uma área livre ou ocupada, respectivamente, para navegação. Outras representação do espaço podem ser encontradas em trabalhos na literatura, como a representação por Segmentos [Newman et al., 2003] e por Pontos [Lu & Milios, 1997].

Além do mapa métrico, o modelo topológico é utilizado como alternativa diferente para representar o ambiente. O mapa é representado por um grafo, onde objetos, localizações ou características de interesse são identificados por nodos e as arestas representam as relações espacial entre esses elementos. O mapa topológico usado junto com o mapa métrico, ou seja, o modelo híbrido (Figura 1.2), permite realizar análises

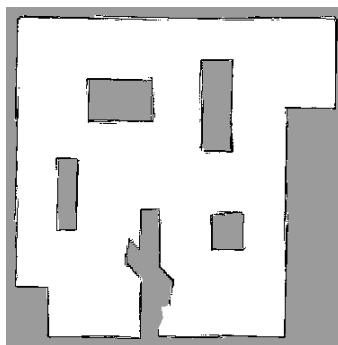


Figura 2.1: Exemplo de Malha de Ocupação criada ao término de uma exploração. As áreas em cinza representam regiões ainda inexploradas, com alta incerteza, e as regiões em branco e preto correspondem as regiões conhecidas e classificadas como livres e ocupadas, respectivamente.

mais amplas. Por exemplo, pode ser usado para armazenar características visuais importantes do ambiente que servirão para identificar a localização do próprio agente [Korrapati & Mezouar, 2014] e também pode ser usado em ambientes com grandes dimensões para unir mapas métricos locais de diferentes regiões [Simhon & Dudek, 1998].

2.1.1 Mapeamento com o Uso de Sensores de Localização

O uso de sensores que informam a localização absoluta do agente é bastante comum no mapeamento de ambientes externos (*outdoor*), como em áreas urbanas e campos. Entre os sensores disponíveis no mercado, o receptor GPS é o dispositivo mais popular, possuindo o melhor custo-benefício.

Uma das vantagens de se utilizar o GPS consiste na obtenção da localização absoluta do agente sem o acúmulo de erro. Ou seja, o robô pode navegar por todo o ambiente e a localização do agente terá a mesma distribuição de erro, caso a qualidade do sinal se mantenha. A principal desvantagem está relacionada com a instabilidade do sinal. Essa limitação pode causar uma redução na precisão da localização do agente durante o mapeamento.

Para evitar que esse fator influencie na qualidade do mapa em construção, foram desenvolvidas técnicas que realizam a fusão dos dados coletados pelo GPS com os dados de outros sensores, como o IMU (*Inertial Measurement Unit*), o odômetro e, até mesmo, a bússola. Desse modo, é possível reduzir o erro na estimativa da localização do agente durante a navegação pelo ambiente [Zhang et al., 2005; Li et al., 2006].

A técnica com o maior destaque foi desenvolvida por Kalman [1960] e foi adaptada para a fusão de sensores anos após o seu desenvolvimento. Essa técnica ficou

conhecida como o Filtro de Kalman (KF) e, até os dias atuais, é bastante utilizada. Em um dos trabalhos sobre fusão de sensores aplicado na navegação de veículos terrestres e autônomos, as informações obtidas pelo IMU e GPS foram combinadas para calcular a navegação estimada do veículo (*Dead Reckoning*) [Stephen & Lachapelle, 2001; Stephen, 2000]. Os resultados dos experimentos ainda demonstravam um aumento rápido no erro da localização do agente sempre que ocorria a perda completa do sinal do GPS. No entanto, com o bom funcionamento dos sensores, a localização se tornava bem mais precisa após a fusão das informações. Atualmente existem trabalhos que utilizam essa técnica ou variações dela para melhor estimar a localização dos agentes [Oh & Chung, 2012].

O trabalho proposto por Emter & Petereit [2014] utiliza uma variação conhecida como o Filtro Estendido de Kalman (EKF) para fusão de multi-sensores com o objetivo de realizar o mapeamento de ambientes externos. Os sensores usados para localização foram IMU, GPS, Giroscópio de fibra ótica e Odômetro. Como pode ser visto na Figura 2.2, o mapa foi gerado com sucesso na execução do experimento.

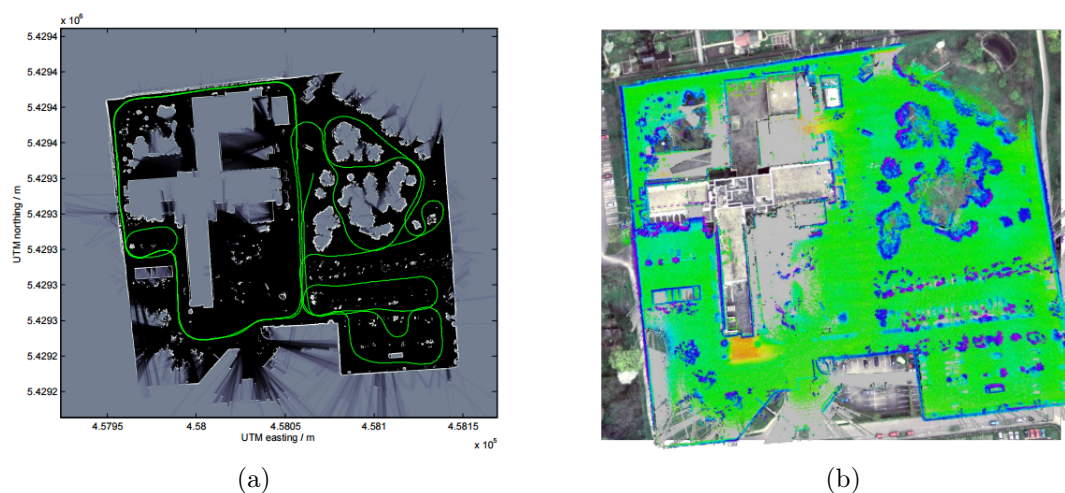


Figura 2.2: Imagens do mapa gerado. a) Mapa geo-referenciado com o caminho percorrido pelo robô. b) Imagem obtida por satélite em comparação com o mapa gerado [Emter & Petereit, 2014].

Uma abordagem diferente para o refinamento da localização foi apresentada por Vishal et al. [2015]. Nesse trabalho, é utilizada uma câmera como sensor para corrigir a baixa precisão de um receptor GPS através de técnicas de reconhecimento de padrões (*SIFT/Bag-of-Words*) bastante conhecidas na Visão Computacional. Essas técnicas buscam por imagens, na base de dados disponível, com características similares às capturadas durante uma simples navegação. Como as imagens da base de dados possuem a marcação com a localização global da câmera no momento em que foi capturada,

torna-se possível realizar o refinamento da localização do agente obtida através de um recepto GPS. O problema para essa abordagem é a necessidade de uma base de dados com diversas imagens (com a localização) dos ambientes que se deseja realizar o mapeamento.

2.1.2 Mapeamento sem o Uso de Sensores de Localização

Visto que sensores de baixo custo podem fornecer a localização de um agente em ambientes externos, não é comum encontrar trabalhos que realizam o mapeamento nesses ambientes sem o uso desses sensores. No entanto, alguns trabalhos atuais como o de Schmid et al. [2013] abordam esse problema utilizando técnicas da Visão Computacional junto com os dados coletados por sensores como o laser e o IMU.

Uma outra abordagem utiliza um mapa aéreo a priori para auxiliar na localização do agente e, assim, poder realizar um mapeamento final mais detalhado. Por exemplo, na abordagem apresentada por Früh & Zakhor [2003] são utilizadas as imagens capturadas por um veículo aéreo, as imagens capturadas por um robô terrestre, além dos dados coletados pelos sensores laser do robô. Assim, um mapa 2D é construído com base nas imagens capturadas pelo veículo aéreo e a localização do agente é estimada utilizando os dados coletados pelo laser. Aplicando a técnica conhecida como Monte Carlo, a localização do agente é, por fim, refinada e o modelo 3D do ambiente é construído com sucesso (Figura 2.3). Contudo, para o funcionamento adequado dessa abordagem ainda é necessária a obtenção de uma mapa aéreo a priori para a construção do mapa final.

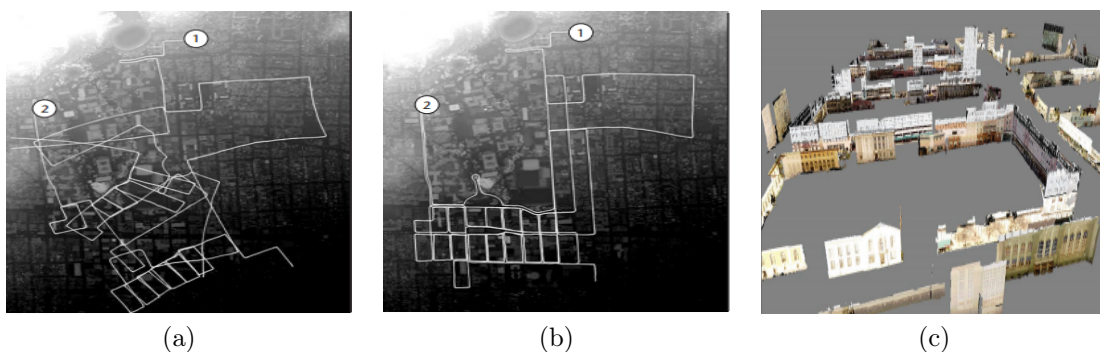


Figura 2.3: Construção de uma representação 3D do ambiente. a) Mapa extraído das imagens aéreas com o trajeto do robô terrestre antes da correção e b) depois da correção da localização do agente. c) Modelo final 3D de uma das áreas mapeadas [Früh & Zakhor, 2003].

Em relação aos ambientes internos, os sensores de localização, como o GPS, podem não funcionar adequadamente. Sem a localização global do agente, o robô precisa realizar uma localização local. Caso o agente não possua qualquer tipo de representação do ambiente, é necessário construir um mapa usando os dados coletados pelos sensores e simultaneamente usar esse mapa para se localizar. Esse problema é conhecido como SLAM (*Simultaneous Localization and Mapping*) e foi apresentado oficialmente pela primeira vez por Durrant-Whyte et al. [1995]. Atualmente, o SLAM é considerado como um problema já resolvido num nível conceitual e teórico [Durrant-Whyte & Bailey, 2006]. No entanto, alguns ambientes reais trazem novos desafios e, consequentemente, novas abordagens estão sendo constantemente exploradas e divulgadas na comunidade científica [Li et al., 2015; Wen et al., 2014].

Os diferentes algoritmos que solucionam o problema do SLAM são abordagens probabilísticas. Através dos dados coletados pelos sensores, os marcos ou referências são identificados nos ambientes. Esses marcos, quando reobservados, são usados como guias para a identificação da localização do agente, tornando possível a construção do mapa. O algoritmo base para as diferentes abordagens está detalhado em [Durrant-Whyte & Bailey, 2006]. A Figura 2.4 apresenta um simples diagrama demonstrando o processo de reconhecimento dos marcos (m) e o ajuste na localização do agente (X) a partir do deslocamento do robô (U) e das observações realizadas (Z) a cada instante (k).

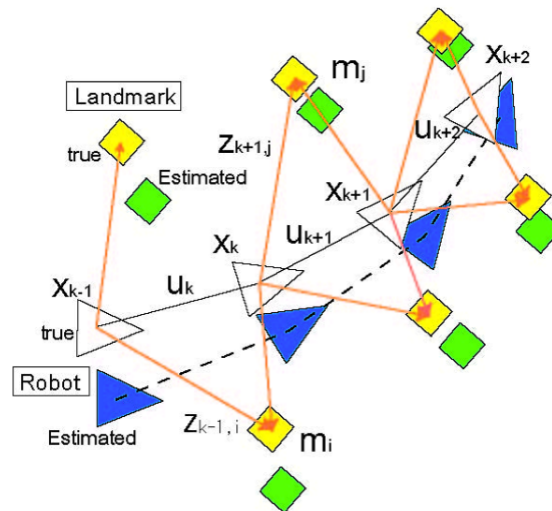


Figura 2.4: Correção da estimativa da localização do agente com base nos referências identificados localmente no ambiente. A partir dessas novas referências, a representação estimada do ambiente se aproxima à real [Durrant-Whyte & Bailey, 2006].

As duas vertentes mais populares na solução do problema SLAM são baseadas nos algoritmos conhecidos como o Filtro de Kalman Estendido (EKF-SLAM) [Dissanayake et al., 2002] e o Filtro de Partículas Rao-Blackwellized (RBPF) [Doucet et al., 2000]. O EKF corresponde a uma derivação do Filtro de Kalman adaptado para sistemas não-lineares. Essa abordagem realiza uma estimativa na localização do agente baseada nas informações obtidas através da observação do ambiente e no deslocamento do agente. A depender do modelo de transição e do nível de ruído dos sensores, a qualidade do mapa pode ser fortemente prejudicada. O RBPF gera diversas partículas com diferentes hipóteses de trajetórias do robô. Cada partícula representa uma trajetória do agente formando um mapa diferente. Semelhante ao Filtro de Partículas simples, as partículas com os menores erros recebem um maior peso, sendo mantidas durante a execução do algoritmo. O maior problema para essa abordagem é o grande consumo de memória quando utilizado um número elevado de partículas. Para cada uma dessas vertentes existem novos algoritmos que otimizam e solucionam problemas que existem nessas técnicas. Por exemplo, Paz et al. [2007] apresentou uma adaptação do EKF permitindo melhorar a performance através do uso da técnica conhecida como Divisão e Conquista (D&C SLAM).

2.2 Estratégias para Exploração Autônoma de Ambientes

A tarefa de exploração é bastante discutida na comunidade da Robótica Móvel, visto que a maioria dos trabalhos que utilizam robôs móveis e autônomos necessitam de um mapa para executar as tarefas de forma apropriada. E para realizar uma construção eficiente de um mapa, uma boa estratégia de exploração precisa ser aplicada. Essas estratégias estão relacionadas com a seleção de pontos de destino para a navegação do robô. Desse modo, uma boa trajetória num processo de exploração deve garantir um mapeamento eficiente.

2.2.1 Estratégias sem Avaliação dos Melhores Destinos

Um dos primeiros algoritmos para a exploração autônoma de ambientes gerava caminhos completamente aleatórios durante a navegação do agente. Essa estratégia é considerada a forma mais simples de exploração, pois não realiza qualquer tipo de avaliação de ganho. Alguns trabalhos mais recentes [Freda & Oriolo, 2005; Schmidt et al., 2006] ainda utilizam essa técnica com algumas adaptações ou em conjunto com outras

estratégias. Outra forma simples de exploração consiste na navegação por rotas fixas. Nos trabalhos apresentados por [Sim & Dudek, 2003; Sim et al., 2004], o robô define sucessivas trajetórias concêntricas (Figura 2.5) pelo mapa, partindo sempre da própria posição até alcançar as regiões inexploradas.

Essas duas estratégias possuem um grande problema devido a constante navegação por caminhos já explorados. De modo geral, esses dois algoritmos não são muito eficazes, pois não garantem mapas completos, e nem eficientes, pois demandam muito tempo para mapear.

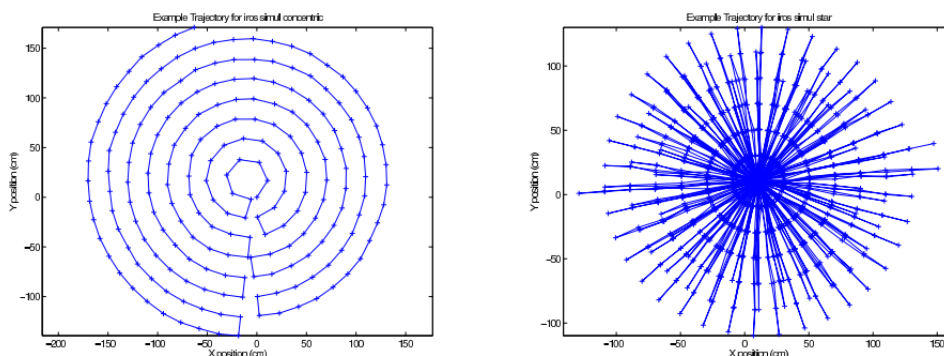


Figura 2.5: Trajetórias baseadas em estruturas concêntricas [Sim & Dudek, 2003; Sim et al., 2004].

2.2.2 Estratégias com Avaliação dos Melhores Destinos

Com o decorrer dos anos, foi observada a necessidade de selecionar e avaliar melhor os candidatos a destino com o objetivo de tornar o processo de exploração mais eficiente. Assim, uma sequência de destinos ótimos permitiria que o agente explorasse o máximo possível das áreas desconhecidas sem navegar desnecessariamente por regiões já exploradas. Como resultado, novas estratégias surgiram buscando melhores abordagens para a exploração autônoma de ambientes.

Para selecionar o melhor destino, é importante identificar todos os pontos candidatos a destino no mapa em construção. Assim, Yamauchi [1997] apresentou uma técnica conhecida como Exploração Baseada em Fronteiras. Ficou definido como fronteira o limite entre as regiões livres e desconhecidas no mapa (Figura 2.6). Após a atualização do mapa e identificação dessas regiões, o algoritmo propõe a configuração do novo ponto de destino do robô para uma das fronteiras formadas. Assim, surgem as opções de caminhos para exploração partindo sempre do robô a cada uma dessas fronteiras. Diversos trabalhos utilizam essa abordagem [Freda & Oriolo, 2005; Jebari

et al., 2011; Fang & Zhang, 2013]. Um estudo proposto por [Holz et al., 2010] avalia a eficiência de algumas das estratégias baseadas em fronteiras.

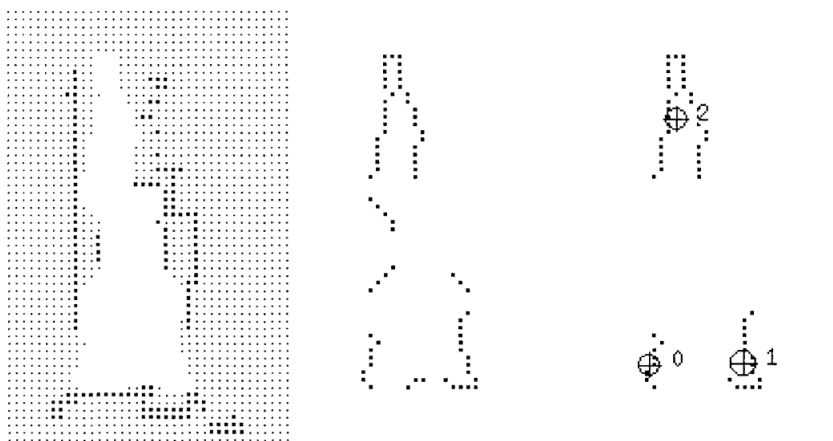


Figura 2.6: Exemplo ilustrativo para identificação de fronteiras num mapa (Malha de Ocupação). Esquerda: o próprio mapa. Centro: grupos com todas as células livres em contato direto com células desconhecidas. Direita: as fronteiras formadas após ignorar pequenos grupos definidos pelas células previamente selecionadas [Yamauchi, 1997].

Uma das primeiras estratégias desenvolvidas para a escolha dos melhores destinos usou uma abordagem gulosa. O algoritmo avalia cada ponto candidato através de uma função que analisa o custo de deslocamento partindo sempre da localização do próprio agente no mapa. Consequentemente, as regiões próximas ao agente são classificadas como de maior prioridade.

O trabalho pioneiro foi apresentado por Yamauchi [1997]. No artigo, Yamauchi demonstra as etapas de exploração de um dos experimentos através da Figura 2.7. Nesse experimento, o ambiente adotado para exploração consiste num corredor com um escritório adjacente ($7m \times 6m$) contendo móveis como armários, cadeiras e mesas. Nessa representação do ambiente, as regiões que permitem a livre navegação do robô são indicadas por áreas brancas; as ocupadas ou inavegáveis, por pontos pretos; e as ainda desconhecidas, por pontos cinzas. Os pontos selecionados como candidatos a destino estão representados cada um por um círculo com uma marcação cruzada e a pose do robô está representada por um círculo com a marcação indicando a orientação do próprio agente. Como pode ser observado, os candidatos a destino mais próximos são definidos sempre como as melhores soluções, exceto quando não são alcançáveis. Essa técnica ficou conhecida como Nearest Frontier Exploration (NFE), exploração pela fronteira mais próxima [Yamauchi, 1997]. Por exemplo, nas Figura 2.7e e 2.7f, dois candidatos a destino são ignorados mesmo estando mais próximos ao robô.

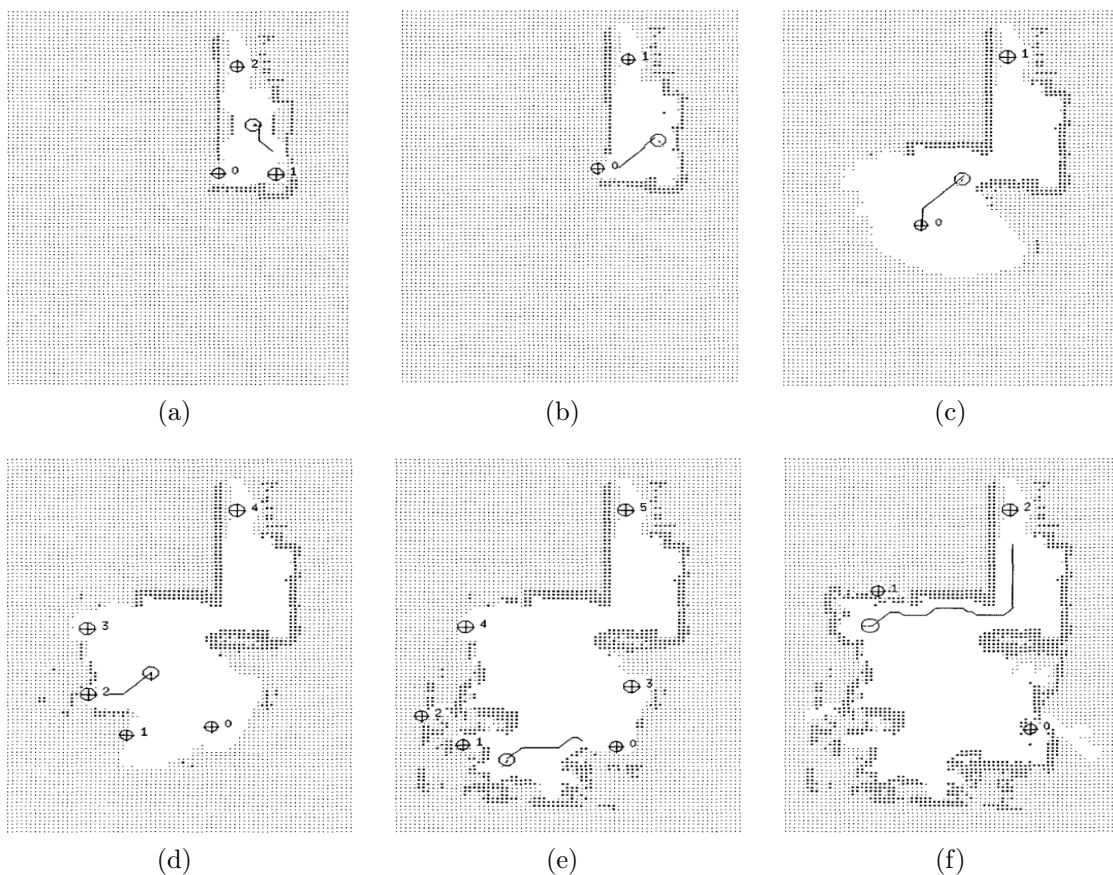


Figura 2.7: Etapas de uma exploração usando o algoritmo guloso baseado no custo de deslocamento do robô para cada ponto candidato a destino [Yamauchi, 1997].

Apesar da simplicidade da abordagem, os resultados obtidos referentes ao custo total de deslocamento e o tempo de exploração são próximos aos obtidos por algoritmos mais sofisticados [Holz et al., 2010]. E, assim, tem sido bastante comum o uso desse algoritmo como *baseline* nos trabalhos publicados na comunidade.

Abordagens mais recentes adaptaram a função de avaliação, transformando-a em uma função de utilidade. Essa nova função avalia a contribuição de cada ponto candidato a destino no processo de exploração. A utilidade é calculada baseada em diferentes critérios, os mais comuns são: o ganho de informação, o custo de deslocamento e o fator que mede a precisão na localização do agente [Amigoni & Caglioti, 2010].

Nesse contexto, serão apresentados alguns trabalhos com estratégias mais elaboradas para seleção dos melhores destinos durante um processo de exploração. No geral, essas técnicas visam a redução do tempo total de exploração e da distância percorrida por todo o processo.

2.2.2.1 Estratégias Baseadas na Visibilidade do Robô

No trabalho apresentado por González-Baños & Latombe [2002], o ganho de informação é projetado com base na cobertura obtida pelo alcance máximo do sensor laser ao inserir o robô num ponto do mapa definido como candidato a melhor destino para exploração. As áreas desconhecidas que forem cobertas pelo sensor irão representar uma simplificação do ganho estimado para aquele destino. Junto com esse ganho, o custo de deslocamento é usado para a avaliação e escolha dos melhores destinos.

O algoritmo aplicado para a escolha dos melhores destinos está apresentado pela Figura 2.8. Inicialmente, o agente alcança a posição representada pelo número quatro onde é realizada uma observação do ambiente e as fronteiras são identificadas. Em seguida, as amostras que correspondem aos pontos candidatos a destino são geradas próximas às fronteiras através de uma distribuição uniforme no espaço. Como pode ser visto na Figura 2.8c, a área de visibilidade indicada em vermelho irá representar o ganho estimado de informação para uma das amostras candidatas a destino (q). E, junto com o custo de deslocamento, irão definir a utilidade da fronteira para o robô. Nesse trabalho, os dois valores são combinados através de uma função exponencial. Por fim, o algoritmo seleciona o próximo ponto de destino de maior interesse para exploração (Figura 2.8d).

Um outro trabalho bastante similar ao anterior foi apresentado por Stachniss & Burgard [2003]. Este trabalho utiliza os mesmos critérios na função de utilidade. No entanto, combina os valores que representam os dois critérios através de uma função linear. O uso da visibilidade para estimar o ganho de informação é uma técnica não muito interessante pois traz pouca informação real ou nenhuma sobre as regiões além das fronteiras.

Buscando melhor identificar as regiões com os maiores níveis de incerteza, Visser & Slamet [2008] apresentaram uma alternativa que se baseia nas regiões de segurança formadas pelo sensor de alcance laser. Essas regiões são definidas com base na alta confiabilidade do sensor quando o alcance máximo é limitado a um valor pequeno. Por exemplo, todas as células do mapa cobertas pelo sensor com o alcance de até metade do seu valor máximo possuem incerteza nula e, assim, são classificadas como seguras. De modo geral, o algoritmo constrói dois mapas: um baseado nas informações obtidas pelo sensor laser com o alcance limitado e outro com o alcance máximo. Na Figura 2.9a, o mapa com as informações seguras foi sobreposto ao mapa formado pelo laser com alcance máximo. Para diferenciar os dois mapas, as regiões livres do mapa “seguro” foram definidas como cinza, as do mapa com alcance máximo em branco e as áreas desconhecidas de ambos os mapas foram marcadas em azul. Assim, as fronteiras são

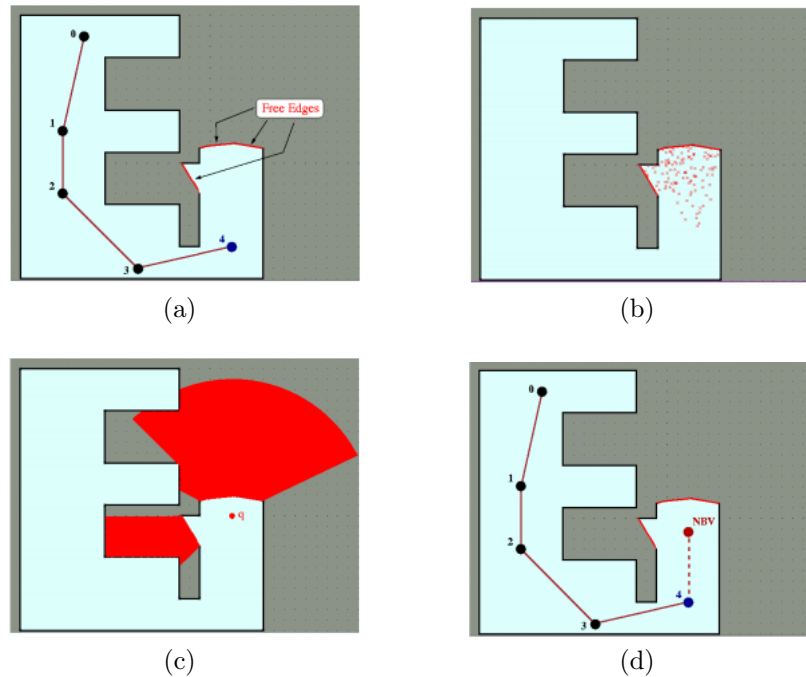


Figura 2.8: Etapas para identificação do próximo destino durante uma exploração. (a) Observação do ambiente e identificação das fronteiras. (b) Gerar pontos candidatos a destino. (c) Analisar a área que representa o ganho de informação estimado para cada ponto candidato. (d) Definir o ponto de destino com base na função de utilidade [González-Baños & Latombe, 2002].

definidas como o limite entre as regiões livres dos dois mapas (Figura 2.9b). Como pode ser visto, essa técnica não constroi as fronteiras como foram definidas anteriormente. As regiões livres além das fronteiras (Figura 2.9c) representam áreas captadas pelo sensor com um certo nível de incerteza. Assim, a função de utilidade é projetada com base no ganho de informação dessas áreas, no custo do agente em se deslocar para as fronteiras e na qualidade do sinal que permite a comunicação do robô com as estações fixas espalhadas pelo mapa. Apesar de ser uma estratégia interessante, este algoritmo não busca inferir sobre as regiões que estão além do alcance máximo do sensor.

2.2.2.2 Estratégias Baseadas no Mapa

Uma abordagem que também busca avaliar as fronteiras pelo ganho de informação estimado foi apresentado por Colares & Chaimowicz [2014]. No entanto, essa técnica é bastante diferente das outras pois não utiliza o sensor laser para estimar o ganho de informação das regiões além das fronteiras. De modo geral, essa estratégia executa uma análise do mapa a partir de uma janela (semelhante a uma máscara usada em filtros de imagem) que desliza sobre as células do próprio mapa obtendo informações

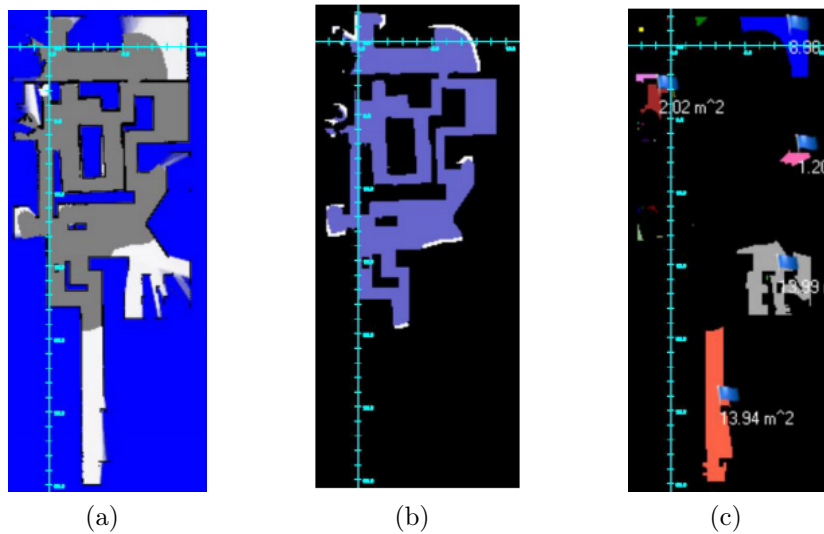


Figura 2.9: Processo para identificação da região com maior incerteza para exploração. (a) Comparação dos dois mapas gerados com base no alcance do sensor. (b) Identificação das regiões de fronteira. (c) Escolha do melhor destino com base na área formada além das fronteiras [Visser & Slamet, 2008].

sobre possíveis fronteiras e regiões de incerteza (Figura 2.10). De forma simplificada, o ganho de informação estimado para uma fronteira é definido com base na incerteza das células vizinhas da própria fronteira. Tais informações são atribuídas a uma função de utilidade que destaca as regiões de interesse para a exploração. Essa técnica também busca melhorar a qualidade do mapa através de novas visitas às áreas já exploradas que ainda possuam um certo nível de incerteza. Assim, o trabalho permite não apenas gerar um mapa completo, mas também alcançar boa precisão no mapa gerado.

2.2.2.3 Estratégias Baseadas no Mapa e no Modelo do Sensor

Em um trabalho recente apresentado por Jadidi et al. [2015], um mapa contínuo é gerado através de um processo gaussiano para auxiliar na escolha das melhores fronteiras a serem exploradas. Para isso, o mapa deve ser transformado do espaço discreto (Malha de Ocupação) para o contínuo (Mapa de Ocupação Contínuo) (Figura 2.11, lado esquerdo). Em seguida, um mapa chamado de Informação Mútua (MI) deve ser construído através do cálculo da diferença entre as entropias atual e a posteriori para cada ponto do mapa recém transformado. Para o cálculo da entropia a posteriori, um modelo inverso do sensor é usado junto com uma inferência Bayesiana. Por fim, as fronteiras são identificadas através de um cálculo de gradiente no mapa de ocupação contínuo (Figura 2.11, lado direito) e a melhor opção de destino é selecionada usando uma função de utilidade no mapa MI. Como esse mapa representa o ganho estimado

-1	-1	-1	-1	-1	0,5	0,1	0
-1	-1	-1	0,5	0,5	0,41	0	0
-1	-1	0,5	0,35	0,23	0	0	0
-1	-1	0,5	0,25	Fu	0	0	0
-1	-1	0,4	0,2	0,1	0	0	0
-1	-1	-1	1	1	0,1	0	0
-1	-1	-1	-1	1	0,9	0	0

Figura 2.10: Parte de um mapa de ocupação com uma janela deslizante (representada em vermelho) calculando a quantidade de informação (função de utilidade) de uma célula (representada em verde) com base nas informações dos seus vizinhos. O valor ‘-1’ representa áreas desconhecidas do ambiente e qualquer valor entre ‘0’ (áreas livres) e ‘1’ (áreas ocupadas) indicam o conhecimento do robô em relação a uma porção do ambiente [Colares & Chaimowicz, 2014].

de informação, a função de utilidade seleciona a região com o maior ganho em conjunto com o fator de deslocamento do robô à região.

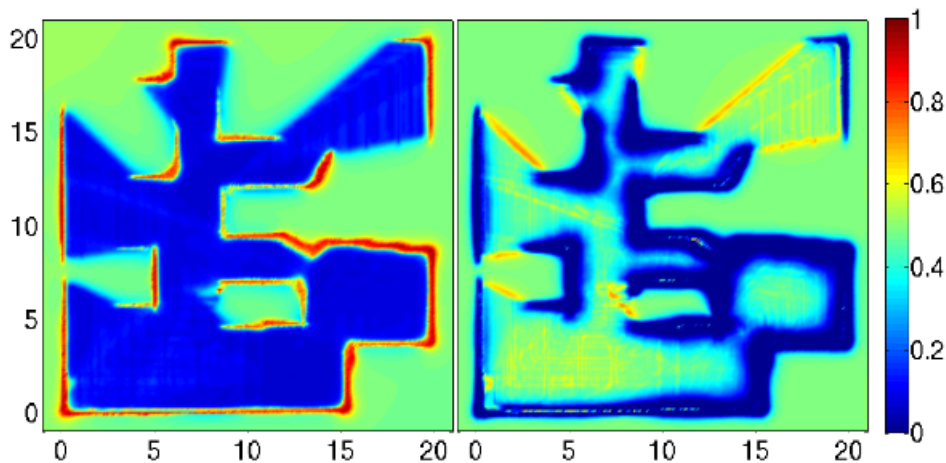


Figura 2.11: Detecção das regiões de interesse em um mapa de ocupação contínuo (*Cave*). No mapa da esquerda, as cores em vermelho escuro representam as áreas em que o robô possui a maior certeza da existência de obstáculos e as cores em azul escuro representam as áreas com a maior certeza de serem livres para navegação. A variação entre essas cores indicam o nível de certeza sobre o mapa formado pelo robô. Em relação ao mapa da direita, a cor vermelha indica as regiões de maior interesse para navegação e a azul representa o oposto [Jadidi et al., 2015].

Semelhante ao trabalho anterior, Yang et al. [2013] gera os melhores caminhos para exploração com base nas informações extraídas do mapa contínuo e nas inferências realizadas pelo modelo do sensor. Assim, o ganho de informação é estimado para cada opção de destino selecionado e a melhor rota para exploração pode ser navegada. Apesar dessas duas últimas técnicas apresentarem um bom uso da Teoria da Informação na exploração de ambientes desconhecidos, elas ainda não definem a configuração das células desconhecidas que estão além das fronteiras.

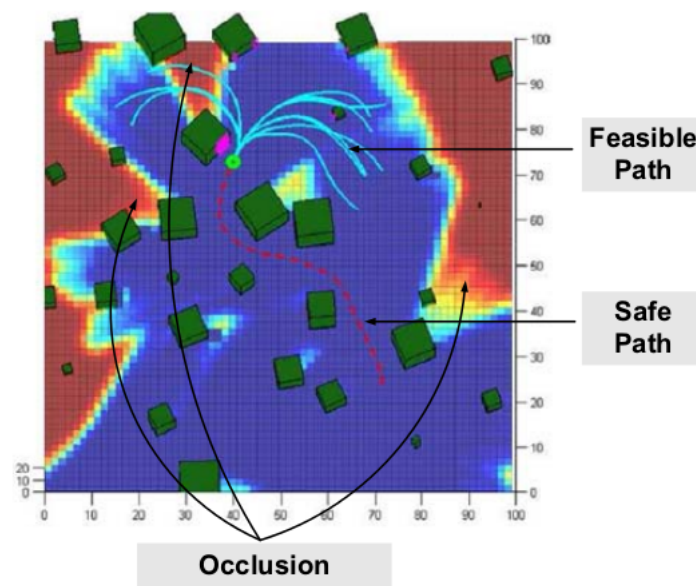


Figura 2.12: Mapa contruído com base nas inferências realizadas usando o modelo do sensor laser. Em vermelho estão representadas as áreas ainda desconhecidas ao robô. Em cor azul, as áreas livres. E, com a variação de cor entre azul para vermelho, as regiões com as inferências [Yang et al., 2013].

2.2.2.4 Estratégias Baseadas em Predição

Estratégias que utilizam a predição para obter a configuração das células além das fronteiras têm sido apresentadas como melhores alternativas comparadas as técnicas que analisam o ganho de informação com base apenas na visibilidade do robô ou unicamente pelas informações extraídas do mapa. O objetivo dessas estratégias consiste em guiar o robô em direção às fronteiras que potencialmente irão maximizar a redução da incerteza do mapa com base no conhecimento adquirido pela predição.

Por exemplo, Ström et al. [2015] usa o método que tenta combinar as áreas além das fronteiras com mapas locais já conhecidos previamente. Essa base de dados com mapas locais é fornecida ao robô no início da exploração e é gradualmente incrementada

durante novas explorações. O processo consiste em identificar um mapa local na base de dados que possua características semelhantes às regiões próximas às fronteiras. Desse modo, o mapa que obtiver a melhor combinação irá indicar a melhor predição da região além da fronteira. Na Figura 2.13a, a localização do robô é indicada por um círculo azul no mapa em construção. Em seguida, um mapa local da base de dados, apresentado em vermelho, é identificado como melhor predição. Assim, uma técnica conhecida como diagrama de Voronoi é aplicada para gerar um grafo que irá permitir realizar buscas por novas rotas de exploração. Como pode ser visto na Figura 2.13d, uma rota é identificada formando um *Loop Closure*. Para esse trabalho, essa rota tem uma maior prioridade pois ajuda no ajuste da precisão da localização do robô. Este método traz bons resultados para ambientes com estruturas repetitivas mas pode ser prejudicial a exploração em ambientes não tão bem estruturados.

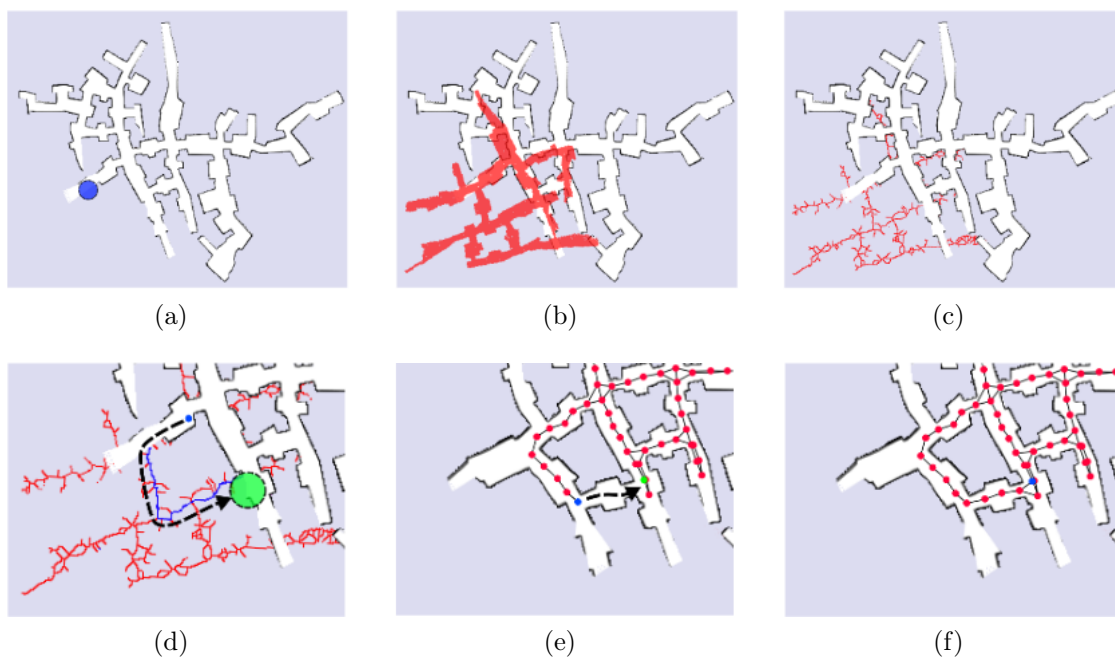


Figura 2.13: Etapas de uma exploração utilizando conhecimento prévio para melhor prever as áreas além das fronteiras [Ström et al., 2015]. (a) Mapa com a localização do robô. (b) Identificação da melhor predição com base nos mapas locais já existentes. (c) Construção de caminhos usando o Diagrama de Voronoi. (d) Identificação de um Loop Closure, ideal para corrigir o problema da incerteza na localização do agente. (e) Navegação pelo caminho gerado. (f) Confirmação da predição feita pelo método proposto.

Outra abordagem que utiliza de um conhecimento prévio para realizar a predição e, assim, selecionar as melhores fronteiras para exploração foi apresentada por Oswald et al. [2016]. Nesse trabalho, a exploração se baseia num esboço da estrutura

do ambiente conhecido antes mesmo de iniciar o processo de exploração. O objetivo dessa estratégia consiste em tornar a exploração mais eficiente dado que o esboço do ambiente já é conhecido. Contudo, o esboço apresentado ao robô não necessariamente irá corresponder a estrutura atual do ambiente. Dessa forma, rotas devem ser facilmente redefinidas quando caminhos forem identificados bloqueados ou, até mesmo, quando não existirem. Na Figura 2.14.a, um esboço do mapa é entregue ao robô e uma rota otimizada para exploração do ambiente é definida. Durante a exploração, o robô identifica um bloqueio e analisa novamente o esboço do mapa fornecido. Nesse mapa inicialmente entregue, a técnica conhecida como diagrama de Voronoi é aplicada. O segmento com o bloqueio é removido (Figura 2.14.c) e, por fim, uma nova rota otimizada é gerada. Essa estratégia é bastante interessante quando o ambiente é conhecido de forma superficial ou não atualizado. Para a exploração de ambientes totalmente desconhecidos essa técnica é inviável.

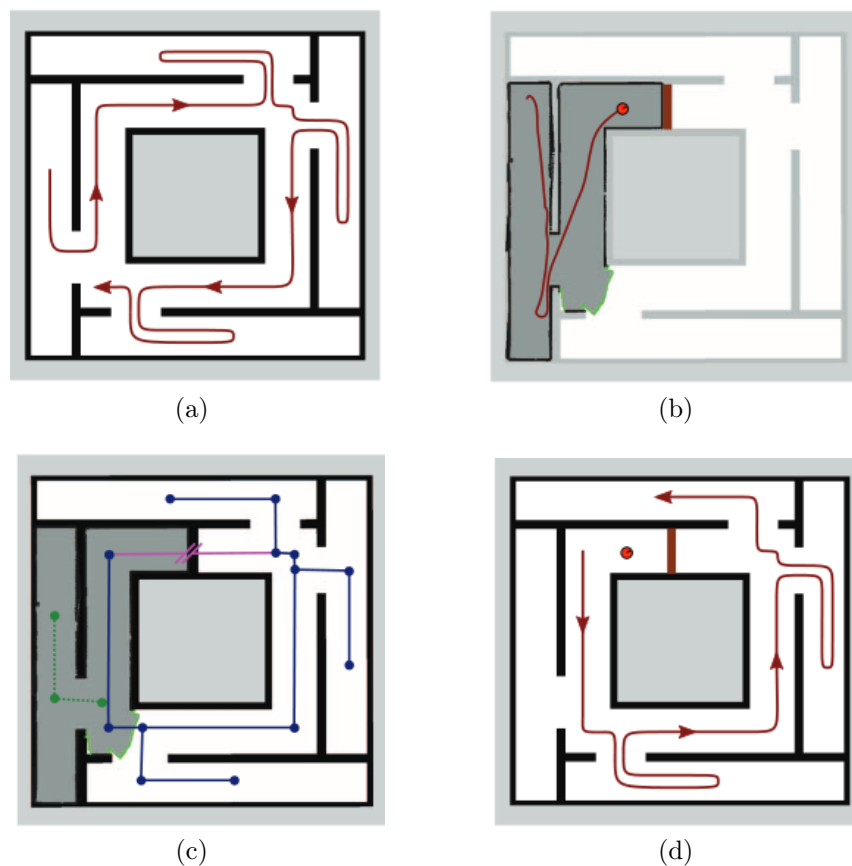


Figura 2.14: Otimização da exploração com base num esboço do ambiente fornecido previamente ao robô [Oskwald et al., 2016]. (a) Esboço do ambiente a ser explorado com caminho otimizado. (b) Início da exploração e identificação de um obstáculo no caminho gerado. (c) Diagrama atualizado do espaço a ser explorado. (d) Novo caminho otimizado gerado.

Por fim, outras abordagens também utilizam conhecimento prévio dos ambientes para melhorar a eficiência da exploração, no entanto, não possuem uma função de utilidade que calcule o ganho de informação dos pontos candidatos a destino. Por exemplo, Stachniss & Burgard [2003] utiliza o conhecimento de que os ambientes possuem corredores e, desse modo, quando os corredores são encontrados, a função de utilidade aumenta o peso dessas regiões para que o robô explore todos os corredores inicialmente e, em seguida, se desloque para as salas adjacentes. Wurm et al. [2008] usa o conhecimento do ambiente com uma finalidade diferente. Dessa vez, a estratégia é aplicada para coordenação de multi-robôs na exploração do ambiente.

2.2.2.5 Estratégias aplicadas a Multi-Robôs

O uso do ganho de informação e de alguns outros critérios já citados têm sido também aplicados na exploração com multi-robôs. Colares & Chaimowicz [2016] apresentam uma estratégia que utiliza uma função de utilidade linear composta tanto pelo ganho de informação como pelo custo de deslocamento. No entanto, essa função de utilidade possui um fator extra que penaliza as regiões já percorridas por outros robôs. Assim, a tomada de decisão também contribui para a coordenação dos multi-robôs durante uma exploração. O ganho de informação é calculado através de uma técnica chamada de potencial de informação. Essa técnica analisa o potencial de informação de cada célula da fronteira para definir o ganho de informação estimado da fronteira. O potencial de informação de uma célula é definido pela interação da célula com seus vizinhos. Assim, uma célula deixa de ser analisada isoladamente e passar a agregar informação das células vizinhas. No entanto, semelhante aos outros trabalhos, não é possível inferir informação sobre a configuração das células além das fronteiras.

O trabalho apresentado por Cipolleschi et al. [2013] utilizou uma técnica conhecida como *Multi Criteria Decision Maker* (MCDM). Essa técnica avalia e escolhe as fronteiras que possivelmente irão retornar um maior ganho de informação com base em diversos critérios de avaliação. Nesse trabalho, também foram utilizados multi-robôs para a exploração de ambientes. Essa estratégia usa uma estimativa com base numa função de utilidade $u(p, r)$ aplicada a cada agente r para cada fronteira p . Para definir essa função de utilidade diversos critérios são usados, como: o tamanho da fronteira, distância euclidiana do agente a fronteira, a energia gasta pelo agente para alcançar a fronteira, a relevância da fronteira (baseado no tamanho da sala e no conhecimento a priori da possível localização das vítimas) e o número de portas da sala em que o agente se encontra. Os valores para os critérios são normalizados e, em seguida, os pesos são atribuídos a cada um. Por fim, uma equação integraliza esses valores definindo o valor

real da função de utilidade entre um agente e uma fronteira. A fronteira com a maior função de utilidade torna-se, assim, a mais interessante pro agente. Nesse trabalho também é aplicado um algoritmo *Fuzzy* para definir o número de agentes necessários para explorar cada fronteira. Para ambientes internos e extensos, acredita-se que um número maior de agentes nesse mesmo ambiente pode otimizar a exploração. Apesar do uso de diversos critérios na formação da função de utilidade o trabalho ainda não busca inferir o que está além das fronteiras.

2.3 Navegação

A navegação é outra etapa fundamental na tarefa de exploração. Após a seleção de cada ponto de destino no mapa, o robô precisa definir rotas partindo sempre da própria posição até alcançar os destinos selecionados. Essas rotas devem ser livres de colisão e com o menor custo de deslocamento possível, visto que o tempo percorrido é um fator crítico para tarefas que buscam maior eficiência. Atualmente, existem diversos algoritmos para navegação aplicados à tarefa de exploração de ambientes desconhecidos. Essa seção irá trazer alguns trabalhos recentes que usam alguns dos algoritmos tradicionais.

2.3.1 Wavefront Propagation

Um dos algoritmos mais simples para navegação usado na exploração de ambientes tem sido o *Wavefront Propagation*. Esse algoritmo gera rotas ótimas e livres de colisão. No entanto, o algoritmo tem alto custo quando aplicado em ambientes amplos. De forma simples, o algoritmo usa um modelo de propagação no formato de onda partindo da célula do mapa que representa a posição do robô até alcançar o destino selecionado. Assim, quando a célula de destino é atingida, a rota ótima é gerada através de um caminho inverso.

O trabalho apresentado por Colares & Chaimowicz [2015] utiliza uma adaptação do método *Wavefront Propagation*. A técnica é aplicada a uma função de utilidade para melhor definir o custo de deslocamento para cada fronteira candidata a ser explorada (Figura 2.15).

Outro trabalho que também faz uso desse algoritmo na exploração de ambientes foi apresentado por Kasperski et al. [2013]. Um pouco diferente dos outros trabalhos, o algoritmo é usado para auxiliar na coordenação de multi-robôs durante a tarefa de exploração. Para cada robô, uma onda é propagada (Figura 2.16) e, assim, com base nas regiões delimitadas pela propagação das ondas, as diferentes fronteiras são classificadas como possíveis destinos a serem explorados por cada robô.



Figura 2.15: Exemplo de propagação de onda do algoritmo *Wavefront Propagation* usando um único robô [Colares & Chaimowicz, 2015].

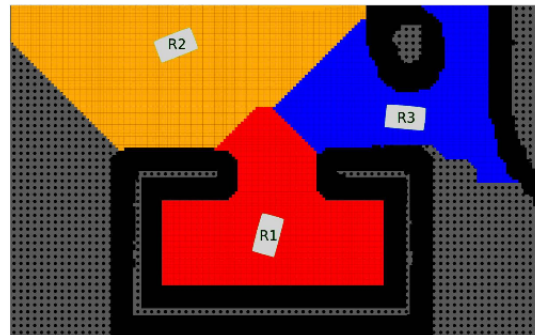


Figura 2.16: Exemplo de propagação de onda do algoritmo *Wavefront Propagation* usando multi-robôs [Kasperski et al., 2013].

2.3.2 Campos Potenciais Artificiais

O algoritmo de navegação conhecido como Campos Potenciais Artificiais [Khatib, 1986] é também bastante usado na exploração de ambientes desconhecidos. Esse algoritmo guia o robô pelo ambiente com base nas forças de repulsão, formadas entre o robô e os obstáculos, e nas de atração, formadas entre o robô e os pontos de destino selecionados. Por não precisar de um planejamento de rotas, muitos trabalhos utilizam essa técnica. Assim, o deslocamento do robô no ambiente ocorre com base apenas nas forças que atuam sob o próprio agente a cada instante. Atualmente, diversas adaptações com melhorias no algoritmo já foram propostas, como modificações para evitar o problema do local mínimo [Doria et al., 2013], para exploração de ambientes 3D [Rasche et al., 2011] e, até mesmo, para o uso em ambientes aquáticos [Gupta et al., 2012].

No trabalho apresentado por Jorge et al. [2015], o algoritmo Campos Potenciais Artificiais é aplicado em conjunto com uma técnica usada para identificação de

Loop-Closure (Figura 2.17). Essa técnica permite que o robô retorne para as regiões já exploradas e, assim, consiga corrigir a incerteza na localização do próprio agente durante o processo de exploração. Por fim, as regiões de fronteira irão atrair o robô para as áreas desconhecidas enquanto que os obstáculos irão repelir o agente, evitando possíveis colisões.

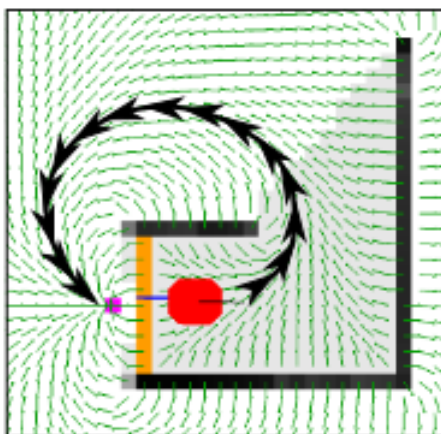


Figura 2.17: Campos Potenciais usado para identificar possíveis Loop-Closure durante a tarefa de exploração [Jorge et al., 2015].

Em um trabalho mais recente, Droeschel et al. [2016] realiza diversos planejamentos para a navegação do robô com o objetivo de mapear ambientes 3D. Entre as camadas de planejamento, o algoritmo Campos Potenciais Artificiais é aplicado para auxiliar na detecção dos obstáculos e livrar o robô de possíveis colisões.

2.3.3 Rapidly-exploring Random Trees

A *Rapidly-exploring Random Trees* (RRT) [Lavalle, 1998] é um dos algoritmos baseados em amostragem mais conhecido para criação de rotas para navegação. De forma simples, o algoritmo original gera uma árvore a partir da conexão de inúmeras amostras aleatórias criadas no espaço de configurações livres do mapa. Essas amostras se tornam novos nós apenas quando não existe a possibilidade de colisão durante a navegação entre os nós mais próximos da árvore e as próprias amostras. Assim, as arestas formadas correspondem aos caminhos livres de colisão que podem ser navegados pelo robô. A expansão da árvore é interrompida apenas quando um determinado ponto de interesse, ou fronteira, é alcançado. Nesse instante, uma rota é gerada partindo da última amostra conectada à árvore em direção à raiz, que deve corresponder ao nó que representa a localização do próprio agente no mapa.

Atualmente, existem diversas adaptações desse algoritmo. Por exemplo, o RRT* consiste na adaptação que permite a formação de rotas com menor custo de deslocamento. No trabalho apresentado por Vallve & Andrade-Cetto [2015], a exploração do ambiente é feita com base na navegação do robô por rotas construídas usando adaptações do RRT* [Karaman & Frazzoli, 2011]. Essa adaptações usam critérios como distância (dRRT*) e o ganho de informação por metro (eRRT*) na avaliação das novas amostras. Na Figura 2.18, é apresentado um mapa no modelo Malha de Ocupação com uma eRRT* expandida por todas as áreas livres. A variação de cor identifica a intensidade do fator ganho por metro. Assim, é representada pela cor vermelho os maiores valores e em verde os menores. Desse modo, é claramente vista a redução do interesse do robô pelas diversas rotas geradas. As áreas com poucas fronteiras e distantes representam um interesse menor, o lado esquerdo do mapa, e as áreas com diversas fronteiras e próximas ao robô representam um interesse maior em explorar, lado direito do mapa. Os pontos em cinza indicam a navegação do robô durante todo o processo de exploração. O robô teve seu ponto de partida no lado esquerdo inferior do mapa.

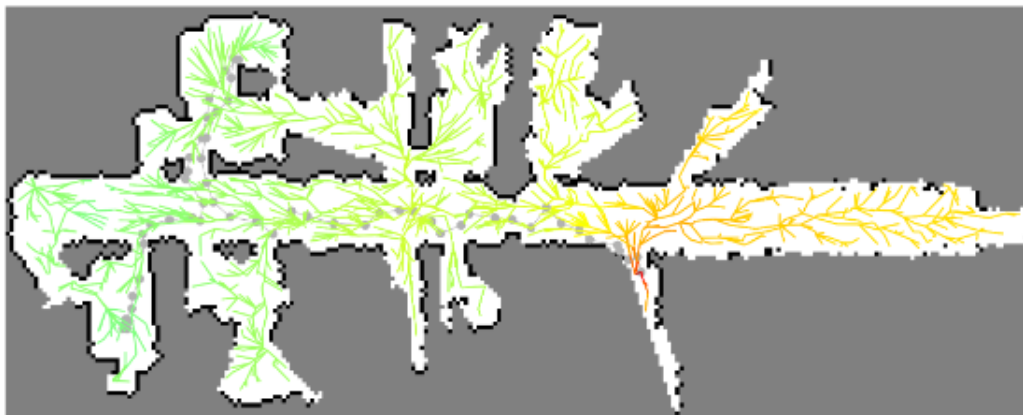


Figura 2.18: Exemplo da expansão de uma eRRT* [Vallve & Andrade-Cetto, 2015]. Nesse experimento, o robô iniciou a exploração no canto inferior esquerdo do mapa. A evolução da exploração pode ser analisada através dos pequenos círculos em cinza que demonstram a localização do agente da etapa inicial até a atual. Nesse instante a árvore foi criada para identificação dos melhores caminhos. A variação das cores nas rotas está relacionada com o nível de interesse das regiões para serem exploradas. As cores “quentes” (vermelho e laranja) indicam um peso maior e as “frias” (verde e azul) o oposto.

2.3.4 Outros Algoritmos

Outros trabalhos utilizam estratégias de navegações usando algoritmos tradicionais de busca, como o Dijkstra [Oßwald et al., 2016], Busca em Largura [Yamauchi, 1997] e A* [Cipolleschi et al., 2013]. No entanto, esses algoritmos necessitam de um grafo para a geração das rotas otimizadas. Para a criação do grafo, muitos trabalhos aplicados na área de exploração de ambientes utilizam o método conhecido como Diagrama de Voronoi [Oßwald et al., 2016] (Figura 2.19). Esse método permite gerar vértices e arestas no mapa sem colisão com obstáculos. Assim, com o grafo formado e a identificação dos vértices que representam a localização do robô e do destino selecionado, um dos algoritmos de busca pode ser executado para gerar a melhor rota. A depender do ambiente, os algoritmos para a criação do grafo podem ser computacionalmente pesados. Isso pode tornar a exploração de um ambiente amplo e complexo inviável para algumas aplicações.

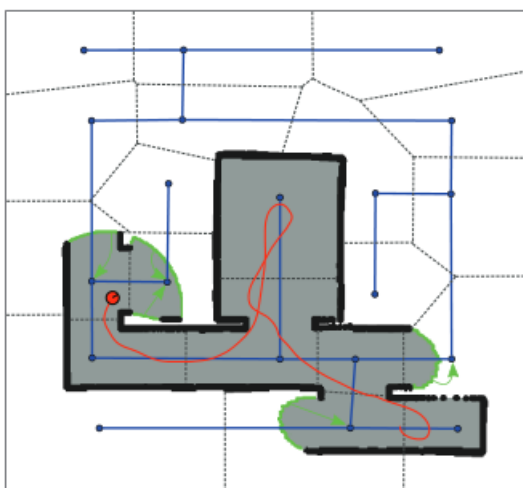


Figura 2.19: Apresentação do Diagrama de Voronoi formado com base no esboço do mapa obtido previamente junto com o próprio mapa em construção. No algoritmo proposto, cada fronteira ilustrada em verde é associada ao nó mais próximo do grafo, permitindo identificar os melhores destinos para exploração [Oßwald et al., 2016].

2.4 Contextualização do Trabalho

Conforme apresentado nas seções anteriores, a exploração autônoma de ambientes desconhecidos é uma tarefa bastante complexa com diversos desafios, como na execução da localização e mapeamento simultâneos e na tomada de decisão dos melhores caminhos a serem explorados. Diversas estratégias e algoritmos veem sendo propostos para

tornar mais eficiente uma exploração completa do ambiente. No entanto, neste trabalho, a maior contribuição está relacionada a uma abordagem inovadora que permite aprimorar a eficiência da exploração através da predição das áreas além das fronteiras em aplicações onde o tempo disponível para exploração pode ser reduzido.

A Tabela 2.1 apresenta um resumo dos principais trabalhos encontrados na literatura relacionados ao problema, destacando quais características específicas abordam, e ao final contextualizando o problema abordado neste trabalho

Tabela 2.1: Tabela comparativa com alguns dos principais trabalhos que abordam a exploração autônoma de ambientes na literatura. (Símbolo '-': não mencionado explicitamente.)

Trabalhos	Exploração	Navegação	Ambientes		Veículos	
			<i>Outdoor</i>	<i>Indoor</i>	<i>Single</i>	<i>Multi</i>
Yamauchi, B. 1997	Guloso	Busca em Largura		x	x	
Freada, et. al., 2005	Aleatório	SRT (Adaptação da RRT)	-	-	x	
González-Baños et. al., 2002	Visibilidade	-		x	x	
Visser et. al., 2008	Visibilidade	Busca em Largura		x		x
Colares et. al., 2016	Mapa	WaveFront Propagation		x		x
Jadidi et. al., 2015	Mapa Sensor Laser	-	x	x	x	
Cipolleschi et. al., 2013	MCDM	A*		x		x
Jorge et. al., 2015	Mapa	Campos Potenciais Artificiais	-	-	x	
Yang et.al., 2013	Mapa Sensor Laser	RRT	-	-	x	
Vallve et. al., 2015	Visibilidade	RRT* (Adaptação da RRT)		x	x	
Oswald et. al., 2016	Predição	Dijkstra		x	x	
Ström et. al., 2015	Predição	Busca em Largura		x	x	
Este trabalho	Predição	RRT		x	x	

Capítulo 3

Metodologia

3.1 Fundamentos e Formalização

Um conceito fundamental aplicado na elaboração da estratégia com foco na tomada de decisão do robô foi o Ganho de Informação. Esse recurso está ligado diretamente com a área da Teoria da Informação (um ramo da matemática que busca quantificar o conceito de informação) e será melhor detalhado na subseção seguinte. Em seguida, o algoritmo tradicional para navegação conhecido como RRT será explicado passo-a-passo. Esse algoritmo foi adaptado para permitir a seleção dos destinos e o planejamento de caminhos durante um processo de exploração. Por fim, o problema abordado nesse trabalho é apresentado formalmente.

3.1.1 Entropia

A Entropia, apresentada por Shannon [1948], é um dos principais conceitos abordados na área da Teoria da Informação e pode ser descrita como a medida da incerteza de uma variável aleatória. Dessa forma, dado uma variável aleatória discreta X com espaço amostral \mathcal{X} e com distribuição $P(X)$, a entropia será a função que mede a incerteza média sobre X [Cover & Thomas, 2006]. Essa função é dada por

$$H(X) = - \sum_{x \in \mathcal{X}} p(x) \log p(x). \quad (3.1)$$

onde $p(x)$ indica a probabilidade de cada evento com base na distribuição de propabilidade da variável aleatória discreta X . Dessa forma, quanto maior for $H(X)$ maior será a incerteza e, conseqüentemente, a variável aleatória terá um potencial de ganho de informação maior.

Quando uma variável aleatória possuir espaço amostral binário, a função da entropia poderá ser reformulada da seguinte forma. Sendo

$$X = \begin{cases} 1, & \text{com probabilidade } p \\ 0, & \text{com probabilidade } 1 - p \end{cases}, \quad (3.2)$$

então

$$H(X) = -p \log p - (1 - p) \log(1 - p). \quad (3.3)$$

Nesse contexto, relacionando a entropia com a tarefa de exploração, um mapa M no modelo conhecido como Malha de Ocupação é composto por células que podem ser definidas individualmente como livres (0) ou ocupadas (1). Além disso, a probabilidade de uma célula qualquer (m_{ij}) estar ocupada deve ser complementar a probabilidade da mesma célula estar livre, ou seja

$$p(m_{ij} = 1) = 1 - p(m_{ij} = 0). \quad (3.4)$$

Desse modo, a equação da entropia de uma célula pode ser definida como

$$H(m_{ij}) = -(p(m_{ij} = 1) \log p(m_{ij} = 1) + (1 - p(m_{ij} = 1)) \log(1 - p(m_{ij} = 1))). \quad (3.5)$$

Se todas as células do mapa forem inicializadas com a probabilidade de estarem ocupadas igual a probabilidade de estarem livres,

$$p(m_{ij} = 1) = p(m_{ij} = 0) = 0.5, \quad (3.6)$$

então a entropia para cada célula será máxima, o que representa a completa incerteza sobre as regiões do ambiente. Essa característica é bastante comum no processo inicial de exploração quando o robô ainda não contém dados necessários para inferir a classificação das células. Assim, a partir do momento em que as primeiras observações vão sendo feitas pelo robô, os valores de algumas células do mapa devem se aproximar a ‘0’ ou ‘1’. Consequentemente, a entropia dessas células será reduzida conforme pode ser visto na Figura 3.1.

Além disso, a equação da entropia do mapa pode ser descrita como o somatório da incerteza de todas as células dada as observações já realizadas pelo robô até o atual instante t com o conhecimento da localização do robô. Assim, a equação da entropia do mapa (M) dado todas as observações já realizadas pelo robô ($z_{1:t}$) pode

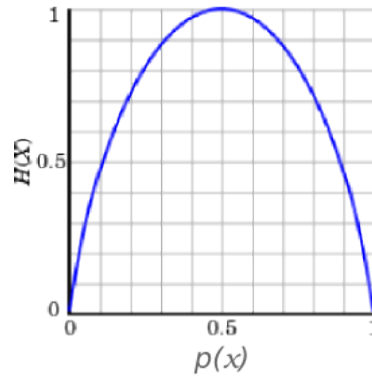


Figura 3.1: Distribuição da entropia para uma variável aleatória com espaço amostral binário. O eixo das abscissas indica a probabilidade da célula estar ocupada ou livre e o eixo das ordenadas o valor relativo a entropia dessa célula.

ser representada por

$$H(M|z_{1:t}) = - \sum_{m_{ij} \in M} p(m_{ij}) \log p(m_{ij}) + (1 - p(m_{ij})) \log(1 - p(m_{ij})). \quad (3.7)$$

Outro conceito bastante comum aplicado na exploração autônoma de ambientes é o Ganho de Informação, conhecido também como Informação Mútua. Na Teoria da Informação, esse fator representa a redução da incerteza da variável aleatória X dado o conhecimento de uma outra variável aleatória Y [Cover & Thomas, 2006]. Esse novo termo pode ser representado por

$$I(X; Y) = H(X) - H(X|Y). \quad (3.8)$$

Considerando que o objetivo de uma exploração autônoma é selecionar os melhores destinos que permitam maximizar a redução da incerteza do mapa, é interessante a inserção do Ganho de Informação como um dos critérios de avaliação para seleção desses destinos. Assim, reformulando a equação anterior para o problema de exploração de ambientes, o Ganho de Informação para um dos destinos pode ser calculado através da análise de todas as observações já realizadas ($z_{1:t}$) e pela observação ao alcançar o destino selecionado (Z_{t+1}),

$$I(M; Z_{t+1}|z_{1:t}) = H(M|z_{1:t}) - H(M|Z_{t+1}, z_{1:t}). \quad (3.9)$$

Ou seja, o ganho de informação que será obtido caso o robô alcance o destino em análise e realize a observação Z_{t+1} . Semelhante à Equação 3.8, o ganho de informação para um destino corresponde a diferença da entropia atual do mapa, dado todas as observações já

realizadas, com a entropia a *posteriori* do mapa, dado todas as observações já realizadas e mais a que deverá ser feita ao alcançar o destino desejado.

Com base na equação anterior, a entropia do mapa pode ser facilmente calculada através da Equação 3.9, visto que o mapa possui a incerteza de todas as suas células. No entanto, o cálculo da entropia do mapa a *posteriori* é computacionalmente pesado e inviável de ser calculada num tempo de decisão curto. Isso devido à necessidade de se calcular cada possível observação ($\hat{z} \in Z$) que o robô poderia realizar ao alcançar tal destino. A incerteza do mapa a posteriori pode ser descrita por

$$H(M|Z_{t+1}, z_{1:t}) = \int_{\hat{z} \in Z} p(\hat{z}|z_{1:t})H(M|Z_{t+1} = \hat{z}, z_{1:t})d\hat{z}. \quad (3.10)$$

Para que o ganho de informação possa ser calculado em tempo viável na escolha dos melhores destinos, diversas técnicas veem sendo elaboradas e publicadas na comunidade. O Capítulo 2 dessa dissertação, apresenta alguns trabalhos com tais adaptações.

3.1.2 Rapidly-exploring Random Tree

A *Rapidly-exploring Random Tree* (RRT) corresponde a um dos algoritmos baseados em amostragem mais utilizados para o planejamento de caminhos. A versão original, proposta por Lavelle [1998], foi desenvolvida com o objetivo de gerar rotas sem colisão de forma mais eficiente. Este algoritmo é considerado probabilisticamente completo, o que garante encontrar um caminho, caso exista, com o aumento do número das amostras.

De forma geral, o algoritmo gera uma árvore partindo da posição atual do robô em direção a todas as regiões livres do mapa (Figura 3.2). Essa árvore é construída através de diversas expansões realizadas com base em amostras geradas no espaço de configurações livres. Cada vértice representa um estado do robô e suas conexões indicam caminhos que podem ser navegados sem interferência de obstáculos. A expansão é executada repetidas vezes até que uma nova configuração que corresponda ao destino selecionado, ou uma região inexplorada qualquer no nosso caso, seja anexada à árvore. Assim, um caminho livre de obstáculos é imediatamente formado entre o robô e o destino.

Um exemplo de execução é demonstrado na Figura 3.3. Mas para que o algoritmo seja iniciado o robô precisa de algumas informações, como a própria localização (x_{init}) no mapa, o número máximo de iterações K que se deseja executar e o tamanho do passo Δt . O estado inicial do robô irá indicar o posicionamento da raiz da árvore que

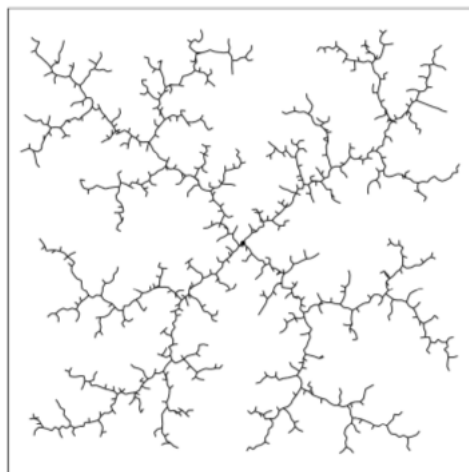


Figura 3.2: Exemplo de uma árvore RRT expandida [Lavalle, 1998].

será gerada no mapa. O número de iterações corresponde a um limite de execução do algoritmo caso uma solução não seja encontrada. E o tamanho do passo corresponde a distância entre os vértices da árvore, ou seja, o tamanho das arestas. Após obter tais informações, as amostras começam a ser geradas, uma para cada iteração do algoritmo. As três primeiras imagens ilustram as principais etapas no processo de expansão de uma árvore enquanto que a última imagem exibe o caminho formado com o término da execução do algoritmo.

Como pode ser visto, uma amostra aleatória (x_{rand}) é gerada dentro do espaço de configurações livre do mapa (Figura 3.3a). Em seguida, é identificada na árvore o vértice mais próximo (x_{near}) da amostra gerada. Com a identificação desse vértice, uma tentativa de expansão é realizada partindo de x_{near} em direção a x_{rand} . Essa expansão gera uma nova configuração (x_{new}) a uma distância fixa, baseada na execução da entrada de controle u por Δt segundos, de x_{near} (Figura 3.3b). Após a criação de x_{new} , é verificada a existência de obstáculos que possam interferir na navegação entre as duas configurações. Se não existir, a configuração x_{new} é conectada à árvore \mathcal{T} formando a aresta (x_{near}, x_{new}) (Figura 3.3c). Diversas iterações são executadas até que uma nova configuração contida em uma região ainda inexplorada pelo robô seja anexada à árvore. Imediatamente, um caminho é gerado entre o novo vértice e a raiz. Por fim, esse caminho é usado na exploração do ambiente (Figura 3.3d).

O Algoritmo 1 apresenta o pseudocódigo do método descrito acima apresentado em Lavalle [1998]. Este algoritmo contempla os passos citados anteriormente, como a geração de uma configuração aleatória (*RANDOM_STATE*), a identificação do vértice mais próximo (*NEAREST_NEIGHBOR*), a análise do segmento entre o vértice mais próximo e a nova configuração (*SELECT_INPUT*), a criação de um

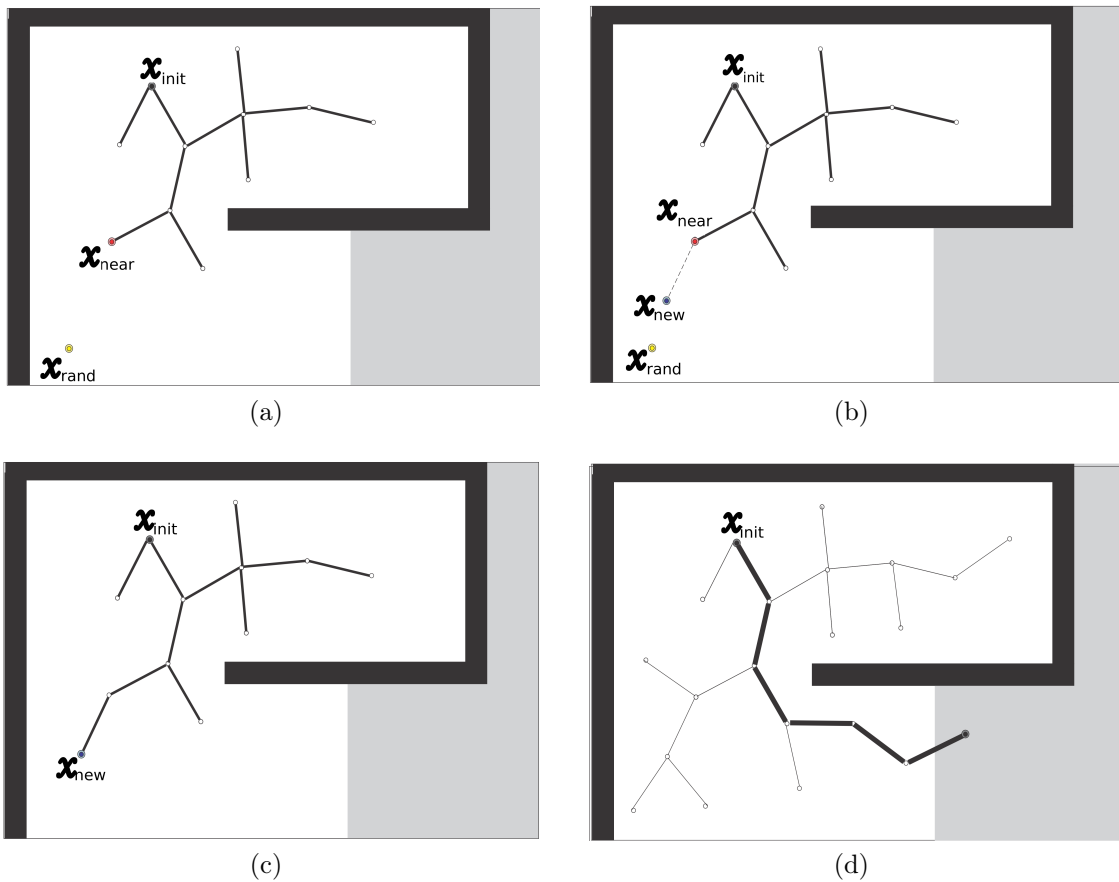


Figura 3.3: Expansão de uma árvore RRT em um processo de exploração. (a) Criação de uma amostra aleatória (x_{rand}) dentro do espaço de configurações livre do mapa e identificação do nodo da árvore (x_{near}) que se localiza mais próximo à amostra. (b) Adição de um novo nodo (x_{new}) à árvore com base na amostra aleatória e no nodo identificado como o mais próximo. (c) Finalização de uma etapa do ciclo de expansão da árvore com a anexação do novo nodo. (d) Caminho gerado após diversos ciclos de expansão partindo da localização do robô (x_{init}) e alcançando uma região ainda desconhecida do mapa.

novo vértice com base no tamanho fixo das arestas (NEW_STATE) e, por fim, a inserção do novo vértice e aresta na árvore em formação ($\mathcal{T}.add_vertex(x_{new})$ e $\mathcal{T}.add_edge(x_{near}, x_{new}, u)$).

O RRT é bastante conhecido pelo seu bom desempenho mesmo em ambientes complexos. Isso devido ao fato do algoritmo não precisar realizar pré-processamentos no mapa e por ser um método baseado em amostragem. Outras técnicas poderiam até gerar caminhos menos custosos ou ótimos. No entanto, a maioria das técnicas precisam inicialmente extrair informações do mapa com o objetivo de gerar uma estrutura de dados semelhante a um grafo para depois aplicar algum algoritmo que soluciona o

Algoritmo 1 RRT($x_{init}, K, \Delta t$) [Lavalle, 1998]

```

1:  $\mathcal{T}.\text{init}(x_{init});$ 
2: for  $k = 1$  to  $K$  do
3:    $x_{rand} \leftarrow \text{RANDOM\_STATE}();$ 
4:    $x_{near} \leftarrow \text{NEAREST\_NEIGHBOR}(x_{rand}, x_{init});$ 
5:    $u \leftarrow \text{SELECT\_INPUT}(x_{rand}, x_{near});$ 
6:    $x_{new} \leftarrow \text{NEW\_STATE}(x_{near}, u, \Delta t);$ 
7:    $\mathcal{T}.\text{add\_vertex}(x_{new});$ 
8:    $\mathcal{T}.\text{add\_edge}(x_{near}, x_{new}, u);$ 
9: end for
10: return  $\mathcal{T}$ 

```

problema do caminho mais curto, como o Dijkstra ou A*. Em mapas mais complexos, essas técnicas costumam ser custosas computacionalmente e, assim, inviáveis para uma atividade *online*. Atualmente, o algoritmo RRT possui diversas adaptações que permitem ainda melhorar a performance da navegação. Por exemplo, Wang et al. [2012] considera a criação de múltiplas árvores na geração do caminho, Denny et al. [2013] gera arestas de tamanho adaptativo para melhor explorar o ambiente, e Islam et al. [2012] gera caminhos mais otimizados.

3.1.3 Formalização do Problema

O problema da exploração autônoma tem sido substancialmente estudado no campo da Robótica Móvel nos últimos anos. Os diversos trabalhos demonstram que a análise desse problema pode ser exercida de forma singular para diferentes tipos de cenários. Conforme mencionado nas seções anteriores, este trabalho aborda o problema em aplicações cujo o tempo pode ser um fator crítico. Nesse caso, a tarefa pode ser caracterizada como um problema de tomada de decisão, no qual o agente deve selecionar as ações que irão maximizar o mapeamento do ambiente.

Para execução dessa tarefa, qualquer robô capaz de se deslocar arbitrariamente pelo ambiente e com campo de visibilidade restrito (alcance máximo do sensor laser fixo a uma determinada distância) pode ser utilizado. As possíveis ações do robô pertencem a um conjunto \mathcal{X} (e.g. mover em uma determinada direção fixa a uma certa velocidade, mudar a direção do robô através do movimento em seu próprio eixo, reduzir a velocidade até parar completamente o robô, etc.) e, em geral, são tomadas para deslocar o próprio agente a um destino desejado. O tempo é determinado por passos discretos $t \in \{1, 2, 3, \dots, n\}$, começando pelo momento em que o robô inicia a exploração ($t = 1$) e finalizando ao término da exploração ($t = n$). É assumido que não existem objetos se movendo pelo ambiente ξ exceto pelo próprio robô, cuja

a localização do veículo em um determinado instante de tempo é dado por R_t . Dessa forma, o robô é capaz de alcançar qualquer configuração livre do mapa, desde que seja alcançável, e realizar as observações necessárias, desde que estejam dentro do espaço de alcance do sensor.

Qualquer elemento detectado dentro da área de cobertura do sensor será interpretado como parte da superfície de um objeto ou de uma parede. Consequentemente, se tornará um obstáculo no mapa que deverá ser evitado pelo robô durante a etapa de navegação. A cada instante de tempo t , a localização física das superfícies escaneadas pelo sensor laser são representadas pelo conjunto $O_t = \{o_{t,0}, o_{t,1}, \dots, o_{t,u}\}$ de pontos em ξ . Mais precisamente, um ponto $o_{t,i}$ pertence a O_t se o sensor do robô determinar que uma localização particular contém um obstáculo que deverá ser evitado. É assumido que o modelo do sensor interpreta como um espaço livre todos os pontos entre a posição do robô R_t e os contidos em O_t (Figura 3.4a). Se não existirem obstáculos detectados pelo sensor (i.e., $O_t = \{\}$), toda a área coberta será interpretada como livre para navegação (Figura 3.4b). Pontos que estão além de O_t são ignorados no instante t e poderão ser avaliados como livres ou ocupados nos passos futuros de tempo.

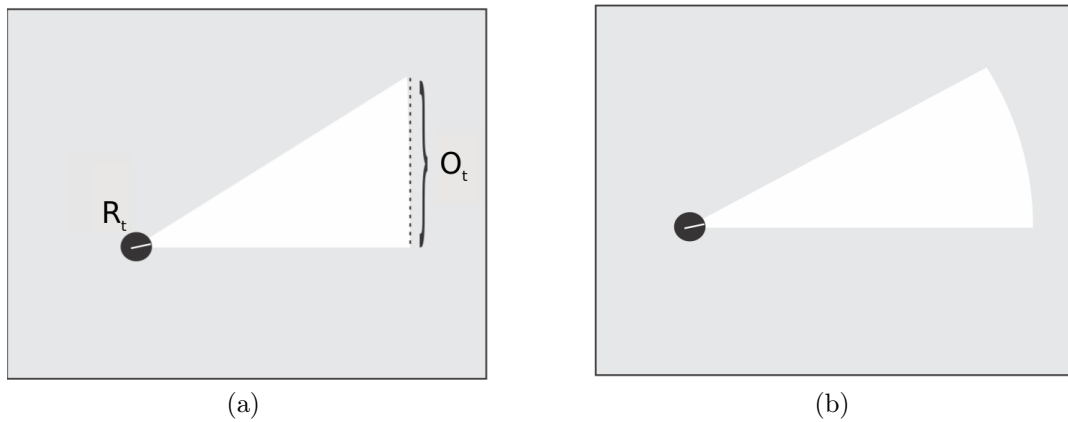


Figura 3.4: Representação de dois possíveis exemplos no uso de um sensor laser com alcance limitado. (a) demonstra o caso quando o laser detecta uma superfície dentro de seu alcance. A posição do robô no instante t é indicado por R_t e dos pontos detectados pelo sensor pelo conjunto $O_t = \{o_{t,0}, o_{t,1}, \dots, o_{t,u}\}$. Em (b) não foram detectados obstáculos e, assim, toda a área de cobertura do sensor é interpretada como livre.

Durante todo o processo de exploração, o robô deve observar o ambiente em sua volta e atualizar o mapa (M) com as informações relevantes. Este processo pode ser considerado como uma tradução do ambiente contínuo ($\mathcal{E} \in \mathbb{R}^2$) para o espaço discreto ($M \in \mathbb{Z}^2$). Nesse trabalho, o modelo Malha de Ocupação foi empregado na construção do mapa, onde cada ponto em ξ é representado por uma das células $m_{ij} \in M$.

Um mapa no modelo Malha de Ocupação consiste em uma malha espacial que reflete como o ambiente está ocupado. Baseado em cada observação (z_i) estabelecida pelo robô desde o momento inicial, as células são preenchidas com base nos três possíveis estados: ocupado (representado pela cor preta), livre (representado pela cor branca) ou desconhecido (representado pela cor cinza). Todas as células são inicializadas como desconhecidas no tempo $t = 1$ da exploração e seus valores são atualizados pelo algoritmo responsável por gerar o mapa e redefinir a localização do agente (SLAM). Foi utilizado o Filtro de Partícula Rao-Blackwellized para criar os mapas a partir dos dados obtidos pelo sensor laser.

Com o objetivo de aprimorar a tarefa de exploração, o robô precisa tomar as melhores decisões ao se deslocar pelo ambiente. Desse modo, todos os candidatos a destino, para cada momento de tomada de decisão do robô, devem ser avaliados e comparados. Nesse trabalho, os candidatos a destino estão sempre relacionados com as fronteiras [Yamauchi, 1997]. O conjunto de fronteiras detectadas em um determinado instante de tempo t é representada por $\mathcal{F}_t \in M$. Assim, algumas características das fronteiras podem ser manipuladas para selecionar a sequência dos melhores destinos para navegação, por exemplo, informações como o comprimento, a distância em relação ao robô e o ganho estimado de informação da região além da fronteira.

Nesse trabalho, o comprimento da fronteira é definido com base no número de células que a compõem e a distância é calculada heurísticamente através da distância Euclidiana entre a localização do robô e um ponto referencial que representa a própria fronteira. O ganho estimado de informação consiste em uma abordagem mais complexa, sendo obtido com base nas ações x_1, \dots, x_n realizadas pelo robô e nas observações O_1, \dots, O_n já coletadas. A combinação desses dois atributos a cada instante de tempo t pode ser redefinida pela variável aleatória $z_t = (O_t, x_t)$. No entanto, é necessário a realização de uma predição z_{t+1} para obtenção de um estado futuro do mapa e, assim, calcular o ganho estimado de informação ($I(M; Z_{t+1}|z_{1:t})$) da fronteira. O método desenvolvido que permite a extração dessa característica para cada fronteira será explicado com detalhes na seção seguinte. Por fim, é assumido nesse trabalho que o robô sabe perfeitamente sua localização durante toda a tarefa.

Nesse contexto, é apresentada uma formulação geral do problema da exploração autônoma de ambientes desconhecidos priorizando o ganho de informação.

Considerando um robô móvel com um sensor laser de alcance limitado, uma Malha de Ocupação M representando o conhecimento do robô sobre o ambiente ξ , uma faixa discreta $t = \{1, 2, \dots, n\}$ representando os diversos instantes de tempo da exploração, uma posição inicial $R_1 \in \xi$ e assumindo

que no tempo t cada fronteira $f \in \mathcal{F}_t$ pode ser identificada pelo robô, encontre uma sequência de destinos $\mathcal{D} = d_1, d_2, \dots, d_n$ in ξ de modo que cada destino d_t

- pertence a uma área livre para navegação próxima a uma $f \in \mathcal{F}_t$;
- guia o robô para áreas inexploradas que devem maximizar a redução da incerteza de M ao ser alcançado.

3.2 Exploração Autônoma

Assumindo que a pose do robô é conhecida durante todo o processo de exploração, esse trabalho determina quais ações devem reduzir rapidamente a incerteza do mapa. Se o ganho de informação atribuído a uma fronteira for maior que o das outras em um determinado instante, então é esperado que o robô navegue diretamente para essa fronteira. No entanto, outros critérios também devem ser analisados, como o custo de deslocamento. Em alguns casos, uma fronteira próxima ao agente com um ganho estimado inferior pode ainda ser mais interessante no aspecto geral da exploração.

O ganho estimado de informação, também conhecido por informação mútua, pode ser definido como a mudança esperada na entropia conforme Equação 3.8. Como já foi explicado, é computacionalmente inviável calcular a entropia *a posteriori* em tempo de execução da exploração sem realizar uma simplificação ou predição. A Figura 3.5 demonstra possíveis decisões e predições realizadas para cada fronteira candidata a melhor destino no cenário apresentado pela Figura 3.5a. Espera-se que as rotas para as fronteiras 2 (Figura 3.5c) e 3 (Figura 3.5d) irão trazer um ganho de informação limitado enquanto que a rota para a fronteira 1 (Figura 3.5b) irá trazer um ganho de informação maior, sendo ainda expansível.

Apesar de serem análises locais, a predição das áreas além das fronteiras permite realizar diversas observações importantes para a tarefa de exploração autônoma de ambientes, como:

- Identificar a fronteira que irá trazer o maior ganho de informação ao mapa;
- Identificar a fronteira que irá trazer a melhor relação custo-benefício (e.g. gasto energético, distância percorrida);
- Identificar as fronteiras que devem ser ignoradas sem perda significativa de informação sobre o ambiente;
- Identificar regiões do mapa de maior interesse para exploração.

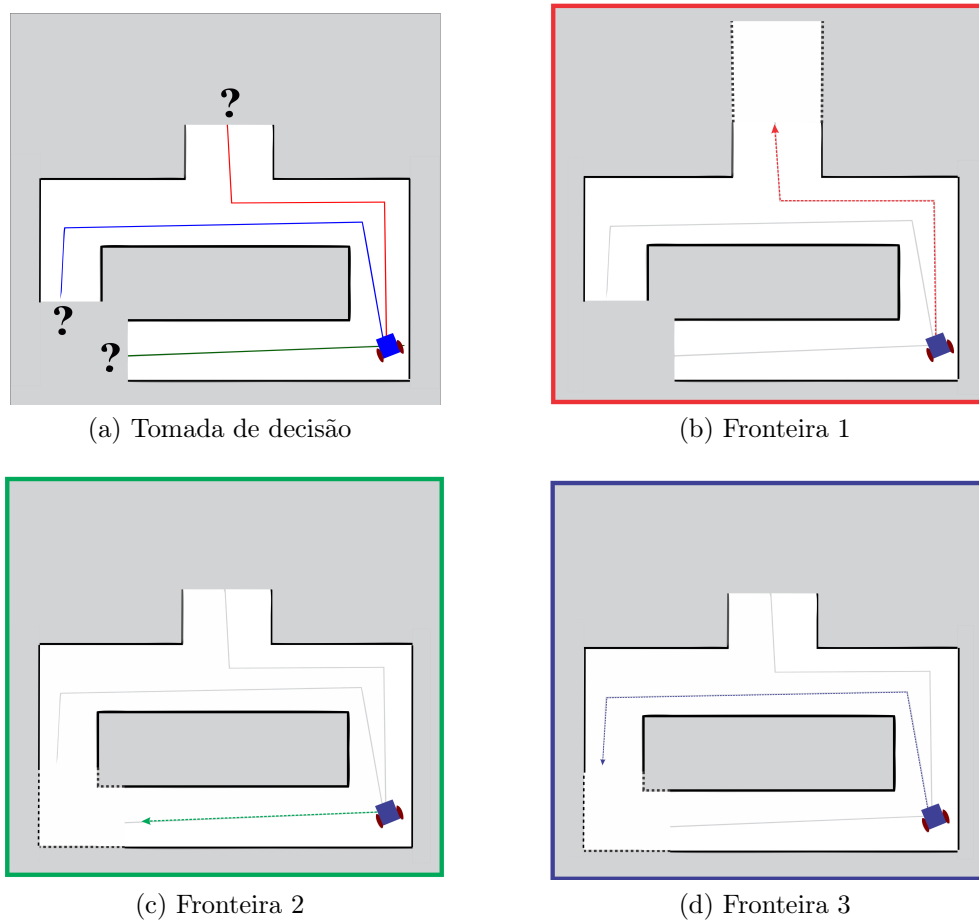


Figura 3.5: Demonstração do ganho de informação baseado na predição das fronteiras. A imagem (a) corresponde ao mapa original e as imagens (b), (c) e (d) representam possíveis predições para cada fronteira com as rotas partindo sempre do robô para cada destino. As rotas para as fronteiras 2 e 3 irão trazer um ganho de informação limitado enquanto que a rota para a fronteira 1 irá trazer um ganho de informação maior, sendo ainda expansível.

3.2.1 Predição Baseada nas Fronteiras

Esse trabalho apresenta uma abordagem simples e inovadora para a predição de áreas inexploradas em ambientes caracterizados como sendo internos. Tais ambientes são comumente representados através do uso de segmentos de reta que se conectam formando salas e corredores. Com base nessa característica, as paredes que envolvem as fronteiras podem ser propagadas, tornando possível estimar a configuração de algumas regiões ainda inexploradas.

Existem essencialmente duas formas distintas para a propagação das paredes que envolvem cada fronteira. A primeira corresponde a uma simples propagação contínua em linha reta (Figura 3.6a e 3.6b), enquanto que a segunda executa uma mudança de

direção sempre que a propagação da parede alcança uma área livre do mapa (Figura 3.6c e 3.6d). Aplicando essa estratégia em todas as fronteiras, é possível prever a configuração das regiões inexploradas e, assim, calcular o valor da entropia *a posteriori* necessário para estimar o ganho de informação caso o robô alcance tais regiões.

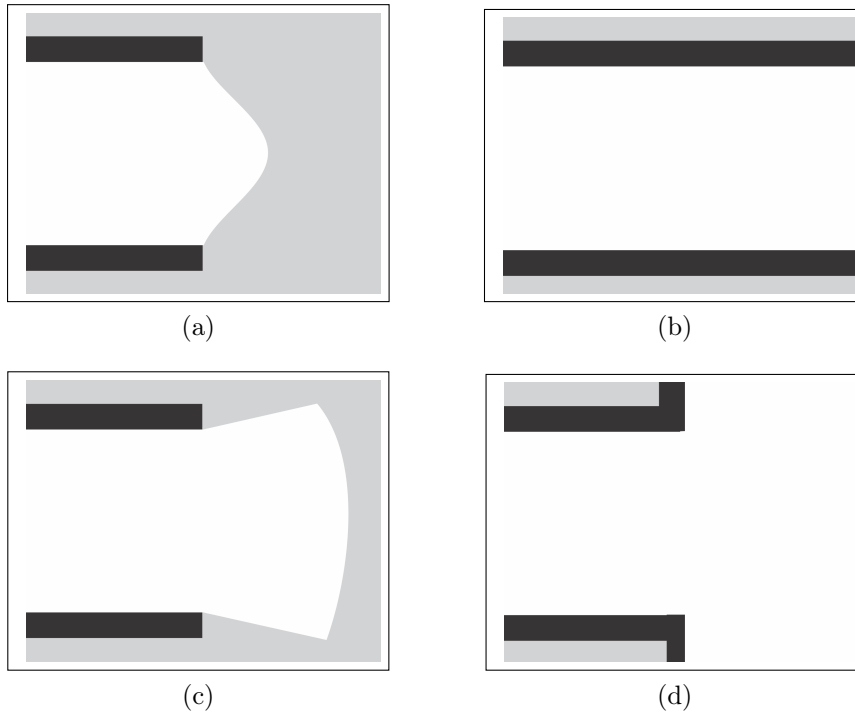


Figura 3.6: Representação dos dois possíveis modelos de propagação de paredes. A primeira coluna representa o mapa com a fronteira detectada e a segunda o mapa com a predição baseada na propagação das paredes. Na imagem (a), as paredes que limitam as fronteiras são identificadas e propagadas linearmente no mapa conforme imagem (b), enquanto que as paredes detectadas em (c) se propagam sendo imediatamente bloqueadas por uma região livre. Consequentemente, a direção da propagação é reconfigurada para as regiões desconhecidas conforme demonstrado em (d).

Contudo, as regiões além das fronteiras podem ser formadas pela combinação de paredes retas e quinas. E, assim, quanto maior for a propagação das predições menor será a probabilidade de se comportarem igual ao estado real do ambiente. Desse modo, as predições não devem se estender por grandes áreas visto que podem se deteriorar. Com o intuito de corrigir esse problema, foram propostas limitações $\phi_i \subseteq \xi$ para cada fronteira $f_i \in \mathcal{F}$, chamadas de Zonas de Predição, permitindo restringir o avanço de todas as previsões. O tamanho da predição ficou definido com base nas dimensões da própria fronteira. Consequentemente, as fronteiras devem gerar zonas de predição proporcionais ao seu próprio tamanho.

Nesse trabalho, os passos para a criação das janelas que representam as zonas de predição foram definidos conforme demonstrado na Figura 3.7. No primeiro momento, as células que pertencem a uma fronteira são identificadas (Figura 3.7a). Em seguida, as duas células mais extremas (semelhante a técnica conhecida como *minimum-perimeter bounding box*) são selecionadas determinando parcialmente as dimensões de ϕ_i como pode ser visto na Figura 3.7b. Dentro da estrutura geométrica formada (em vermelho) irá conter sempre todas as células da fronteira selecionada. Em seguida, a zona de predição pode ser projetada através da expansão dessa estrutura conforme demonstrado na Figura 3.7c. A adição desse valor fixo (em verde) contribui para que a zona de predição sempre incorpore também as paredes que limitam a fronteira. A Figura 3.7d apresenta a zona de predição projetada para a fronteira selecionada.

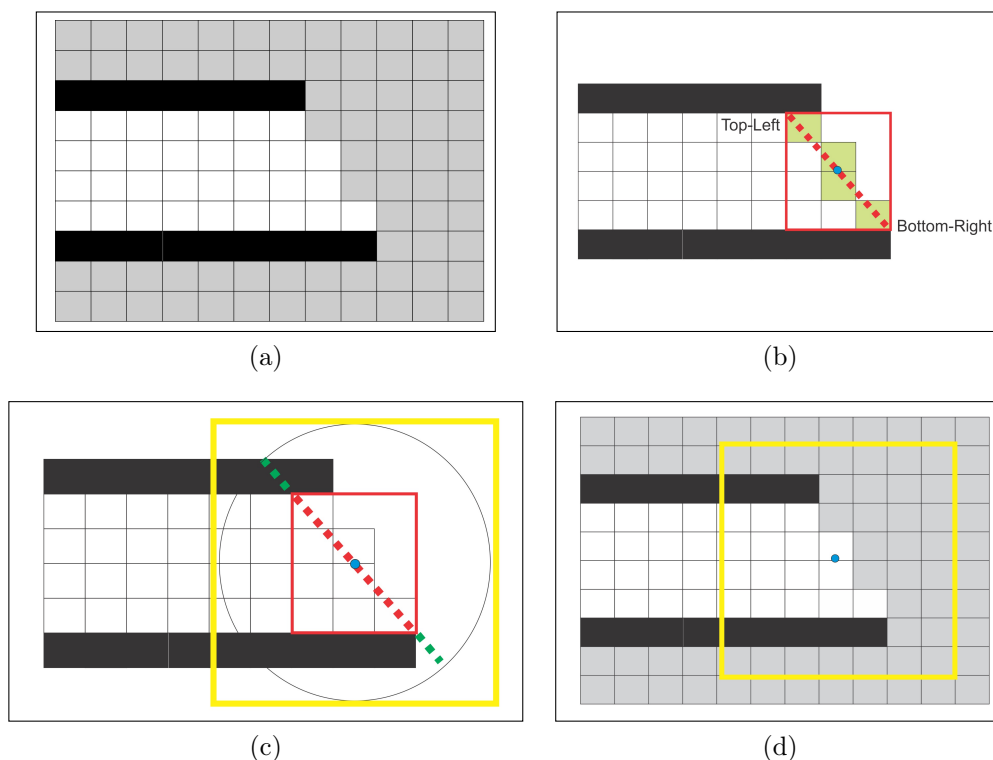


Figura 3.7: Dois passos necessários para projetar uma zona de predição com base nas dimensões da fronteira selecionada. (a) Ilustração de parte do mapa contendo a fronteira selecionada. (b) Primeiro, são determinadas as células mais extremas que irão formar uma estrutura geométrica agrupando todas as células dessa fronteira. (c) Em seguida, a estrutura é expandida uniformemente através de um valor fixo representado em verde. (d) Ilustração da Zona de Predição projetada sobre a fronteira selecionada.

Em alguns casos, ainda é possível que sejam projetadas zonas grandes que gerem predições com baixa acurácia. Para corrigir esse problema, as predições se propagam com uma redução linear de sua certeza. Dessa forma, fronteiras muito grandes irão

interromper a propagação ou reduzir a certeza dentro da própria zona de predição e zonas de predição pequenas não irão sofrer grandes mudanças na predição.

Um exemplo de uma etapa de predição com base na metodologia proposta está apresentada na Figura 3.8. O mapa gerado, durante o processo de exploração, possui quatro fronteiras para serem analisadas (Figura 3.8c). Como pode ser observado, as zonas de predição para cada fronteira foram projetadas cobrindo as próprias fronteiras e as paredes que as limitam. Assim, cada predição foi realizada respeitando as zonas de predição. As zonas de predição apresentadas nas Figuras 3.8a e 3.8d são maiores que as demais e, assim, é possível observar um pouco melhor a redução linear da certeza durante a propagação das células livres no mapa.

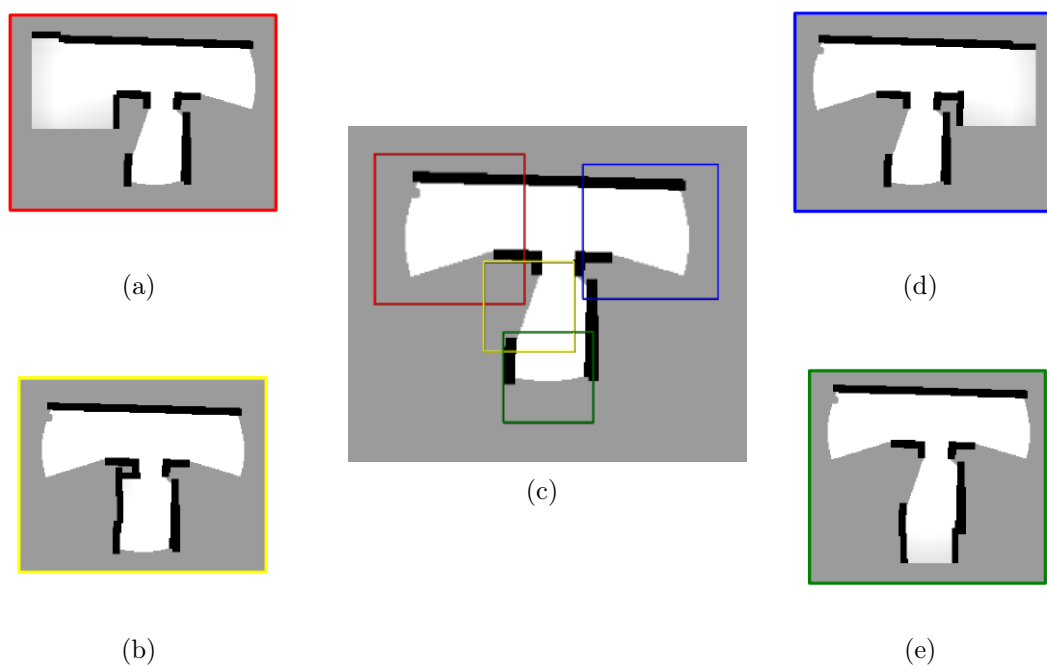


Figura 3.8: Exemplo de predições realizadas após a detecção de quatro fronteiras no mapa em (c). Para as fronteiras em (a), (b) e (d), foram prolongadas as paredes uma de forma reta e outra em quina. Na fronteira em (e), as duas paredes foram prolongadas de forma reta.

Com as zonas de predição projetadas, a etapa seguinte consiste na identificação das paredes que limitam cada fronteira. Uma parede pode ser representada por um segmento de reta no mapa. Assim, através do tradicional método para detecção de linhas conhecido como Transformada de Hough [Hough, 1962], cada parede do ambiente pode ser identificada. Em seguida, é verificado se existe interseções entre os segmentos

de reta com as zonas de predição já projetadas no mapa. Caso existam, cada segmento de reta irá representar uma das paredes que limitam o espaço de uma fronteira. Dessa forma, são identificadas todas as paredes que pertencem a cada fronteira do mapa.

Após o procedimento acima, a estratégia de predição pode ser finalmente executada conforme exemplo apresentado na Figura 3.8. Através da equação de reta, cada parede da fronteira é propagada, cobrindo as células desconhecidas, até alcançar um dos limites da zona de predição. Se uma célula que representa um obstáculo for atingida, a propagação em linha reta é interrompida. Se uma célula que representa um espaço livre também for atingida, a propagação será imediatamente redirecionada para o lado que possui uma área ainda desconhecida do mapa. Para finalizar o processo de predição, as células livres dentro da zona de predição são propagadas através do modelo baseado em ondas. Assim, as células desconhecidas em contato com a fronteira irão se tornar livres. E, em seguida, as novas células livres irão se tornar a nova fronteira. Esse processo irá se repetir até que as células da nova fronteira alcancem os limites da zona de predição.

O Algoritmo 2 formaliza os passos necessários para estimar o ganho de informação para cada fronteira usando a estratégia de predição. Semelhante ao que foi descrito acima, as células ocupadas são identificadas e transformadas em segmentos de reta. Em seguida, a entropia atual do mapa é obtida e armazenada. Depois, as zonas de predição são delimitadas e as predições são efetuadas para cada fronteira. Com as predições definidas, os valores da entropia a posteriori são calculados e o ganho de informação é estimado para cada fronteira.

Algoritmo 2 *CalcularGanhosDeInformacao(F, M)*

```

1: // Inicializar vetor: Ganho de Informação para cada fronteira
2:  $GI \leftarrow \emptyset$ 
3: // Transformar os obstaculos em retas (Transformada de Hough)
4:  $S \leftarrow identificarSegmentosDeReta(M)$ 
5: // Calcular a entropia atual do mapa
6:  $e \leftarrow calcularEntropia(M)$  ▷ Eq. (3.7)
7: // Para cada fronteira
8: for  $i = 0$  to  $size(F)$  do
9:   // Criar a zona de predicao
10:   $h_i \leftarrow tamanhoFronteira(F[i], M)$ 
11:   $l_i \leftarrow criarZonaPredicao(F[i], h_i, M)$ 
12:  // Detectar as paredes conectadas a fronteira
13:   $W \leftarrow identificarParedes(F[i], S, M)$ 
14:  // Iniciar a propagação da predicao
15:   $\hat{M} \leftarrow expandirParedes(l_i, W, M)$ 
16:   $\hat{M} \leftarrow expandirCelulasLivres(l_i, F[i], \hat{M})$ 
17:  // Calcular a entropia a posteriori
18:   $\hat{e} \leftarrow calcularEntropia(\hat{M})$  ▷ Eq. (3.7)
19:  // Calcular o Ganho de Informação da fronteira
20:   $g_i \leftarrow ganhoInformacao(e, \hat{e})$  ▷ Eq. (3.9)
21:   $GI[i] \leftarrow g_i$ 
22: end for
23: return  $GI$ 

```

3.2.2 Seleção e Planejamento de Caminho

A abordagem proposta utiliza um método bastante diferente dos tradicionais para navegação do robô durante um processo de exploração. Nesse trabalho, é realizada uma fusão de duas importantes etapas de planejamento em um único algoritmo: a seleção dos melhores destinos e a determinação dos caminhos até os objetivos. Assim, essa nova técnica é capaz de selecionar os destinos e, ao mesmo tempo, criar rotas livres de colisão até as regiões de interesse.

O algoritmo consiste numa adaptação da técnica conhecida como RRT. A Seção 3.1.2 dessa dissertação apresenta com detalhes o funcionamento do algoritmo RRT original. É importante que o algoritmo seja revisado primeiro para uma melhor compreensão da abordagem proposta nesse trabalho. A principal diferença entre o algoritmo

RRT tradicional e o aqui proposto consiste no método usado para a seleção das amostras (antigo x_{rand}). Esse método tem a função de criar elementos que se tornam vieses na criação das novas configurações que possivelmente serão anexadas à árvore (\mathcal{T}). Ao invés de gerar aleatoriamente de maneira uniforme os pontos no espaço do mapa, esse trabalho introduz uma técnica com referência ao sistema de roletas bastante usado em Algoritmos Genéticos. Em resumo, essa técnica seleciona de forma estocástica os novos pontos que serão usados como viés na expansão da árvore.

Para cada fronteira detectada no mapa, um setor da roleta é reservado conforme demonstrado na Figura 3.10. A configuração dessa roleta foi definida com base nas estimativas do ganho de informação das fronteiras ilustradas na Figura 3.8. O tamanho do setor está ligado diretamente ao ganho de informação calculado anteriormente através do método de predição. Desse modo, fronteiras com um maior ganho estimado de informação devem ocupar setores maiores na roleta e, assim, irão ter uma chance maior de serem selecionadas. A seleção é feita aleatoriamente como se girasse a roleta e esperasse parar. Computacionalmente, um número é gerado aleatoriamente entre os valores 0 e 100. Em seguida, o setor representado pela faixa que contém o número gerado é, então, selecionado.

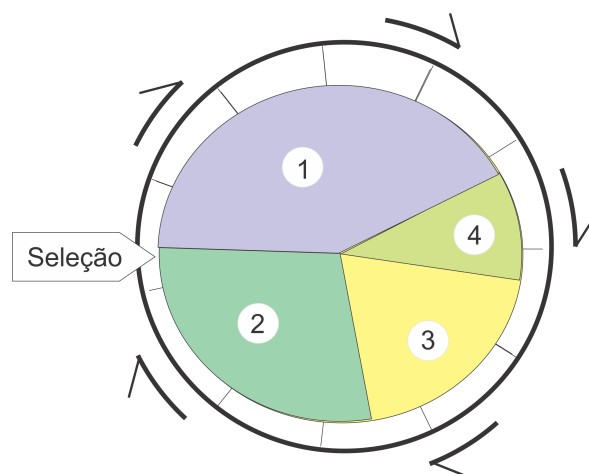


Figura 3.9: Seleção por roleta. Os possíveis destinos são inseridos na roleta com seus respectivos pesos (Setor 1 = 40%, Setor 2 = 30%, Setor 3 = 19%, Setor 4 = 11%). O elemento selecionado será usado como viés na formação da árvore RRT. Nessa roleta não foi inserido um setor para um destino aleatório.

Nesse trabalho, um novo setor da roleta deve ser reservado para um destino aleatório e o restante compartilhado entre as fronteiras detectadas no mapa. Essa característica é necessária para evitar que exista uma convergência da árvore. Por exemplo, quando todas as amostras geradas, direcionadas às fronteiras candidatas a serem exploradas, não podem ser anexadas à árvore devido às possíveis colisões. Assim,

o setor aleatório irá permitir a expansão da árvore circunavegando os obstáculos em busca da solução desejada. O tamanho do setor aleatório é configurado com base no número de fronteiras detectadas. Caso seja identificado apenas uma fronteira, o setor aleatório será definido com 25% do espaço da roleta. Com o aumento do número de fronteiras, o tamanho desse setor é gradualmente reduzido de acordo com a equação

$$y = -5.274764 + (99.99974 + 5.274764)/(1 + (x/0.2154589)^{0.5906926}). \quad (3.11)$$

Essa equação foi projetada com base nos valores de entrada (fronteiras) e saída (tamanho do setor aleatório na roleta) desejados. Esses valores foram definidos de forma empírica, com base na observação dos experimentos. Quando o número de fronteiras for igual a dez, o tamanho do setor aleatório será igual a 5% da roleta. A partir de dez fronteiras, esse setor deve ser reduzido assintoticamente para 1% da roleta.

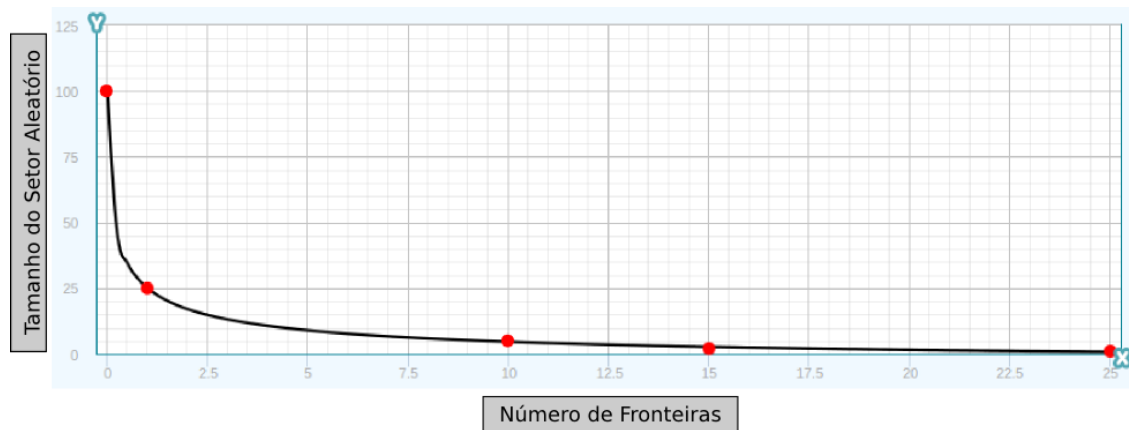


Figura 3.10: Função gerada para definir o tamanho do setor aleatório da roleta com base no número de fronteiras detectadas no mapa.

O algoritmo desenvolvido para seleção das amostras é apresentado a seguir (Algoritmo 3). De forma resumida, o método recebe por parâmetro os possíveis destinos (F) com os respectivos pesos (GI) para seleção de uma amostra que será usada como viés na expansão de uma árvore \mathcal{T} . A probabilidade de selecionar um destino específico está diretamente relacionado ao próprio peso fornecido. Para o algoritmo RRT, é importante adicionar também um peso para um destino aleatório dentro do espaço de configurações livre do mapa. Assim, é possível que os obstáculos sejam circunavegados e as regiões de interesse alcançadas, evitando o problema conhecido como Mínimo Local.

A seleção das amostras usadas como viés na expansão da árvore permite gerar caminhos para regiões ainda não exploradas com características importantes. Apesar do critério para configuração dos pesos da roleta ter sido apenas o ganho estimado de

Algoritmo 3 SelecionarAmostra(F, GI)

```

1: // Definir o tamanho do setor aleatório da roleta
2:  $a \leftarrow \text{setorAleatorio}(\text{size}(F))$ 
3: // Definir o tamanho do setor compartilhado da roleta
4:  $c \leftarrow 100 - a$ 
5: // Normalizar os valores (setor compartilhado)
6:  $W \leftarrow \text{normalizarValores}(GI, c)$ 
7:  $v \leftarrow \text{selecionarValorAleatoriamente}(0, 100)$ 
8: // Identificar a amostra que será usada como bias na RRT
9:  $b \leftarrow \text{identificarSetor}(W, v, a)$ 
10: return  $b$ 

```

informação, a seleção das amostras que não correspondem à fronteira de maior ganho permite eventuais expansões da árvore para regiões distintas do mapa. Dessa forma, fronteiras próximas ao robô também possuem uma chance de serem selecionadas mesmo não sendo o destino com o maior ganho estimado de informação. Conseqüentemente, o algoritmo proposto nesse trabalho realiza indiretamente uma seleção pela razão entre o ganho de informação com o custo de deslocamento. Ou seja, fronteiras com um maior ganho e com um menor custo tem chances maiores de serem selecionadas quando comparadas às fronteiras de menor ganho e distantes do robô. O tamanho da fronteira também consiste em outro critério indiretamente analisado. Como o tamanho da zona de predição está relacionado diretamente com o tamanho da fronteira, o ganho de informação também deve ser proporcional ao critério do tamanho da fronteira. Dessa forma, fronteiras maiores terão maiores chances de serem alcançadas na expansão da árvore.

3.2.3 Sistema

Nessa seção será descrito o sistema proposto por completo, contendo todas as técnicas discutidas nesse Capítulo. Será usado um diagrama para exibir as principais etapas da exploração além de um algoritmo para descrever a implementação de forma superficial.

O diagrama apresentado na Figura 3.11 demonstra o funcionamento desse sistema. O robô ao ser inserido em um ambiente desconhecido realiza uma varredura completa, colhendo informações em sua volta. Com esses dados, é formado o primeiro mapa com a identificação da própria localização do agente. Em seguida, as fronteiras são detectadas através da análise das células e seus vizinhos na malha de ocupação. Caso existam fronteiras para serem exploradas, o sistema calcula a entropia atual do mapa, realiza as predições e calcula a entropia estimada para cada fronteira detectada. Com os ganhos estimados de informação dessas regiões, a RRT adaptada é expandida

até alcançar uma região de interesse para ser explorada. Com o caminho gerado, o robô navega evitando a colisão com os obstáculos. Quando o robô atinge o destino desejado, após a atualização do modelo, é verificada novamente a detecção de fronteiras. Caso não existam, significa que o ambiente foi coberto por completo podendo finalizar a tarefa de exploração.

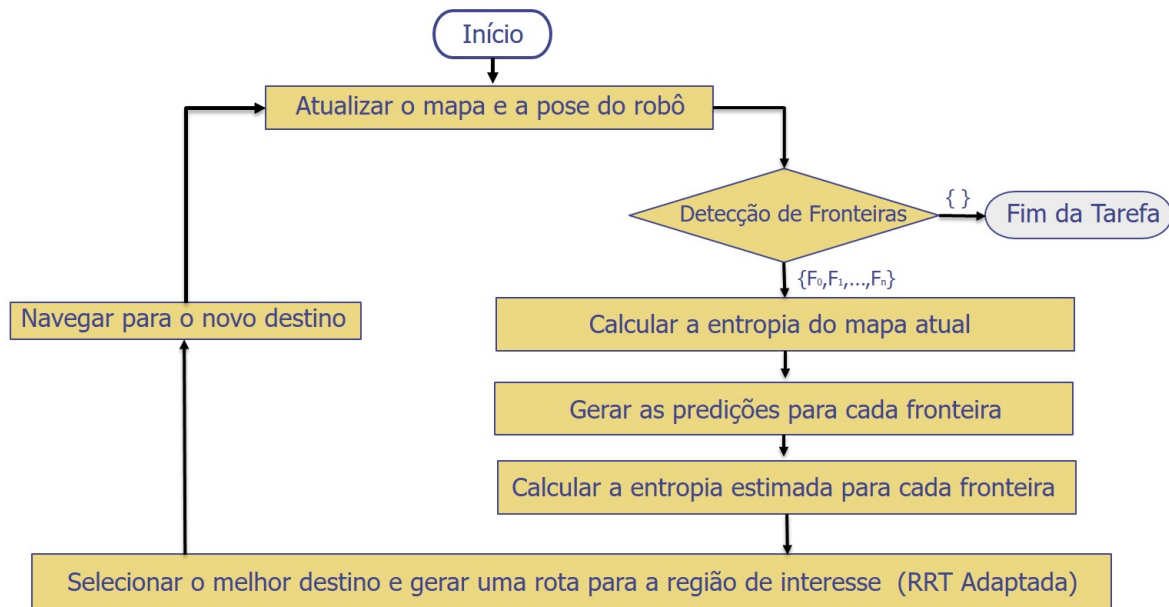


Figura 3.11: Diagrama representando o ciclo que compõe as principais etapas da metodologia proposta.

O pseudo-código sobre a metodologia proposta aplicada à tarefa de exploração é apresentado a seguir (Algoritmo 4). Esse algoritmo descreve os passos necessários para executar uma exploração de forma mais eficiente e, assim, reduzir a incerteza do mapa rapidamente. Para cada iteração do algoritmo, o robô verifica se o destino atual já foi alcançado. Se sim, o robô busca identificar o próximo destino; se não, o robô continua navegando até alcançar a localização desejada. Para calcular o próximo alvo da navegação, é necessário selecionar cada fronteira do mapa e avaliá-las através do ganho estimado de informação obtidos pela predição das áreas além das fronteiras. Em seguida, um caminho para uma nova região de interesse é formado através da geração de amostras com base no método proposto na seção anterior. Esse caminho, ou sequência de pontos no mapa (*waypoints*), é passado para a função *navegar()* desse algoritmo. A função garante a navegação segura do robô entre os pontos até alcançar o destino final do caminho gerado. Por fim, se o destino final for alcançado e nenhuma fronteira for identificada no mapa, a exploração é finalizada, retornando um mapa completo do ambiente em análise.

Algoritmo 4 ExploracaoAutonoma(n)

```
1: // Inicializar vetor: caminho para navegação
2:  $C \leftarrow \emptyset$ 
3: // Para cada ciclo de tempo
4: for timestep  $t$  do
5:    $g \leftarrow \text{checarDestino}()$ 
6:    $M \leftarrow \text{atualizarMapa}()$ 
7:    $R \leftarrow \text{checarPoseRobo}()$ 
8:   if destinoAlcancado( $R, g$ ) then
9:      $F \leftarrow \text{identificarFronteiras}(M)$ 
10:    if size( $F$ ) == 0 then
11:      finalizarExploracao()
12:    end if
13:     $GI \leftarrow \text{calcularGanhosDeInformacao}(F, M)$  ▷ Alg. (2)
14:     $C \leftarrow \text{gerarCaminhoRRT}(GI, R, M, n)$  ▷ Alg. (1)
15:  else
16:    navegar( $C$ )
17:  end if
18: end for
```

Capítulo 4

Experimentos

Esse capítulo apresenta uma avaliação da metodologia proposta através de experimentos em ambientes simulados com características distintas. O objetivo dessa seção consiste em expor as vantagens das duas principais contribuições desse trabalho: a estratégia de predição para calcular o ganho estimado de informação das regiões além das fronteiras e o sistema unificado para seleção e geração de rotas direcionadas aos melhores destinos para exploração. Assim, espera-se que a metodologia proposta consiga cobrir o ambiente de forma mais rápida.

Foram realizados experimentos em dois ambientes simulados com características semelhantes a um escritório e uma caverna. Esses ambientes foram selecionados por apresentarem estruturas comuns a maioria dos ambientes internos. As simulações foram realizadas utilizando o simulador Stage [Gerkey et al., 2003]. O código desenvolvido para a execução da exploração autônoma foi escrito na linguagem de programação C++ usando o *framework* ROS (*Robot Operating System*) e executado em um computador rodando 64-bit Ubuntu 14.04 com um processador Intel i7 e 8GB de RAM.

O pacote do ROS conhecido como gmapping SLAM foi usado para prover a localização do robô e gerar os mapas durante a navegação do agente pelo ambiente. Como o SLAM é considerado um problema resolvido e não corresponde ao foco desse trabalho, os parâmetros de erro dos sensores foram reduzidos para permitir a geração de mapas com alta acurácia. Caso exista uma interferência nos sensores e, assim, uma redução na qualidade do mapa, a técnica adotada para predição das regiões desconhecidas pode ser afetada. Como consequência, as regiões de maior interesse podem não ser identificadas corretamente prejudicando a cobertura rápida do ambiente. Esse efeito será demonstrado na Seção 4.6.

O modelo do agente utilizado na simulação possui características de um robô diferencial não holonômico que se desloca apenas em linhas retas mudando a orientação

através de rotações no próprio eixo. O robô possui um sensor laser de alcance limitado a 5 metros e uma abertura de aproximadamente 60° . Esses valores correspondem ao padrão estabelecido no pacote do *Turtlebot* obtido pela plataforma *ROS*. A navegação do agente é feita através de um sistema de controle dinâmico (PID) que permite o deslocamento do robô através dos caminhos gerados pela RRT. Como a RRT gera caminhos livres de obstáculos e o robô não realiza curvas no seu deslocamento, a navegação por todo o ambiente se torna segura. Ao término da navegação de cada caminho, o robô deve se direcionar para a região inexplorada devido a restrição da abertura do sensor laser. Assim, o mapa pode ser atualizado com as novas informações sobre as regiões inexploradas.

4.1 Procedimento dos Experimentos

Para avaliar a metodologia proposta através da simulação, foram executados os experimentos em seis cenários diferentes. A metade representando um ambiente com características de escritório e a outra metade com características de caverna. Para cada cenário, o robô foi inicializado em três posições pré-definidas. Dessa forma, permitiu uma análise mais rigorosa das estratégias abordadas nesse trabalho, mostrando os possíveis efeitos ao iniciar a exploração em diferentes cenários. Como as abordagens não são determinísticas, ou seja, não geram explorações iguais para experimentos iniciados na mesma posição e no mesmo ambiente, serão executados quinze experimentos para para cada estratégia em cada cenário.

A Figura 4.1 apresenta os mapas de referência para cada ambiente de teste. As posições iniciais estão representadas com as marcações usando as letras A, B e C. A imagem da esquerda representa um ambiente semelhante a um escritório, com longos corredores e salas, e a imagem da direita representa uma caverna contendo grandes áreas abertas com colunas e pedras.

Para cada cenário foram avaliadas as técnicas que abordam o problema de exploração autônoma de ambientes desconhecidos. Essas técnicas estão relacionadas diretamente com o processo de tomada de decisão do agente. Assim, a metodologia proposta nesse trabalho foi comparada com as estratégias que buscam alcançar e explorar sempre:

- **as fronteiras mais próximas (NFE)**: Através do cálculo da distância euclidiana do robô para cada fronteira, ignorando a existência de obstáculos e áreas desconhecidas.

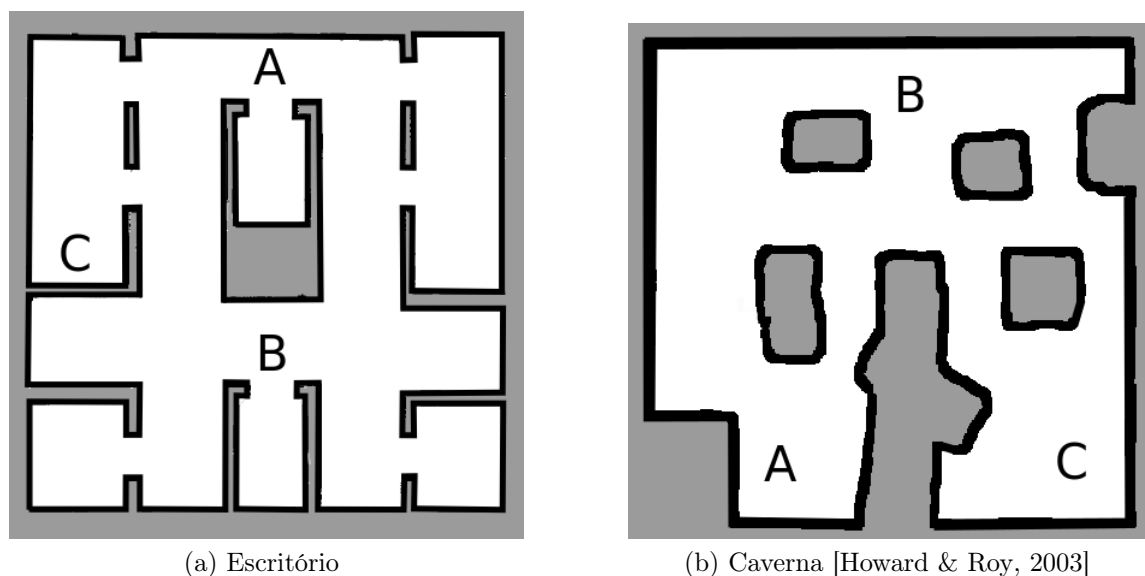


Figura 4.1: Mapas de referência para cada ambiente ($20m \times 20m$). As três posições iniciais para cada mapa estão representadas pelas letras A, B e C.

- **as maiores fronteiras:** Através do número total de células do mapa que compõem cada fronteira.
- **e as que trazem os maiores ganhos estimados de informação:** Através da técnica apresentada nessa dissertação que calcula o ganho estimado usando a predição das regiões além das fronteiras.

Essas três estratégias são classificadas como gulosas pois buscam otimizar o problema da exploração selecionando sempre os destinos que aparentam ser mais promissores, que maximizam ou minimizam algum fator em análise. Nesse caso, os fatores das três estratégias são a distância, o tamanho e o ganho estimado de informação. A estratégia conhecida como NFE, apesar de ser uma abordagem simples, é uma das técnicas mais usadas na comparação com as novas abordagens publicadas na literatura por obter resultados bastante interessantes.

De modo geral, a metodologia proposta nesse trabalho corresponde a um ciclo que deve ser repetido enquanto existir fronteiras a serem exploradas. Esse ciclo está representado pelo diagrama da Figura 3.11. Assim, quando iniciado o experimento, o robô deve realizar as primeiras observações do ambiente e iniciar a construção do mapa. Com o modelo atualizado, o robô deve identificar as fronteiras, realizar as predições para cada fronteira com o objetivo de calcular o ganho estimado de informação e, por fim, selecionar e gerar um caminho para uma região de interesse do mapa com base na técnica descrita na Seção 3.2.2 dessa dissertação. Após alcançar o destino desejado,

o processo é repetido até que o ambiente seja explorado por completo, não existindo mais fronteiras no mapa.

Por fim, as métricas comuns usadas para avaliar trabalhos com o tema exploração autônoma de ambientes, como completude e qualidade do mapa, não foram utilizadas nesse trabalho devido à alta precisão na localização do agente durante todo o processo de exploração. Conseqüentemente, os experimentos foram todos finalizados com mapas completos, bastante semelhantes e com uma boa acurácia. Para avaliar a metodologia, os critérios usados foram a taxa referente ao ganho real de informação pelo tempo e o custo total no deslocamento do robô. Como todos os experimentos foram finalizados com valores similares em relação ao ganho total de informação, foi calculado um valor médio para cada ambiente para representar a completude do mapa. Assim, se tornou possível definir o percentual de completude do mapa em relação ao tempo de exploração para cada experimento.

4.2 Exemplo Ilustrativo

Nessa seção, dois exemplos de exploração baseados na metodologia proposta serão demonstrados. Imagens geradas em diversos momentos dos experimentos serão usadas para ilustrar as principais etapas discutidas no Capítulo 3 dessa dissertação, como a identificação das fronteiras, a criação das zonas de predição, as predições e a criação das árvores RRT para seleção dos melhores caminhos.

4.2.1 Escritório

Nesse experimento o robô foi inicializado na posição representada pela letra A da Figura 4.1a. No primeiro instante, o robô realiza uma observação de 360° no ambiente e gera um mapa inicial apresentado na Figura 4.2. Para esse mapa foram identificadas três fronteiras. E, assim, três zonas de predição foram projetadas com base no tamanho das próprias fronteiras. Como pode ser visto, as zonas de predição 1 e 2 são maiores que a 3, e a zona 1 é ligeiramente maior que a 2.

Para essas fronteiras, o ganho estimado de informação foi proporcional ao tamanho das zonas de predição. Assim, a fronteira que obteve o maior ganho estimado foi a de número 1, seguida pela 2 e 3. A Figura 4.3 apresenta as predições realizadas para cada fronteira com os valores exibidos que correspondem ao ganho estimado de informação. Nesse momento, as fronteiras 1 e 3 geraram regiões mais abertas enquanto que a de número 2 continuou a propagação das paredes em linha reta gerando um espaço semelhante à região que antecede a própria fronteira.

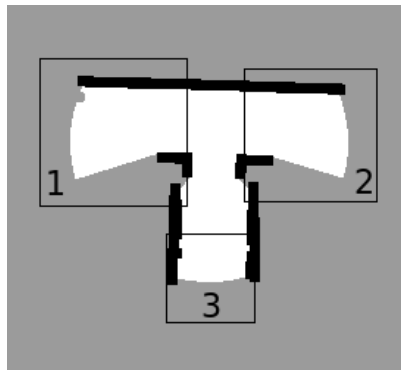


Figura 4.2: Identificação das fronteiras 1, 2 e 3 e criação das zonas de predição na etapa inicial de exploração para o ambiente Escritório.

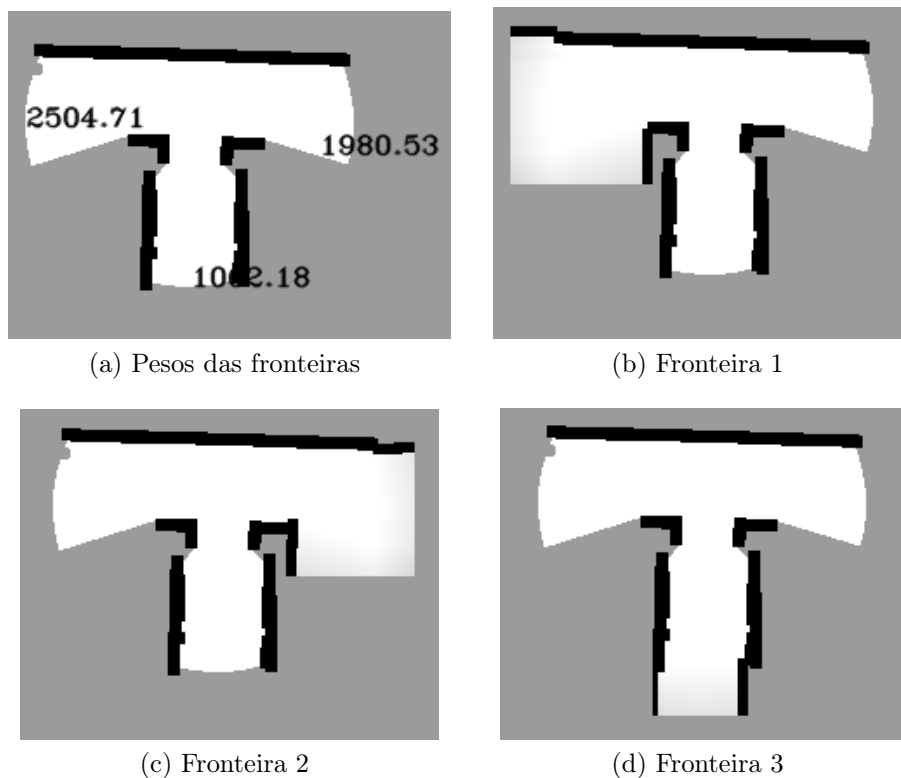


Figura 4.3: Os ganhos de informação baseado nas predições. Em (a) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas predições realizadas. (b) Predição da fronteira 1. (c) Predição da fronteira 2. (d) Predição da fronteira 3.

Para finalizar o processo de tomada de decisão, o robô gera uma árvore RRT que identifica um caminho para uma região de interesse do mapa (Figura 4.4). Nesse instante, o caminho gerado foi direcionado à fronteira de número 1. Na seleção das amostras, as fronteiras 1 e 3 tiveram grande efeito expandindo a árvore nas duas di-

reções. Apesar de um ganho estimado de informação elevado da fronteira 2, pode ser observado uma expansão de apenas um nodo na direção desse destino. Isso ocorre porque os nodos da árvore não podem ser anexados muito próximos aos obstáculos. Assim, partindo do nodo mais próximo dessa árvore em direção ao destino da fronteira 2, não foi possível identificar um espaço livre o suficiente para a inserção de um novo nodo. Além disso, o tamanho do setor aleatório nesse momento foi de apenas 10% aproximadamente, sendo um valor baixo para permitir a expansão da árvore para a fronteira de número 2. Por fim, a expansão foi interrompida quando um novo nodo próximo à posição de destino da fronteira 1 foi anexado à árvore.

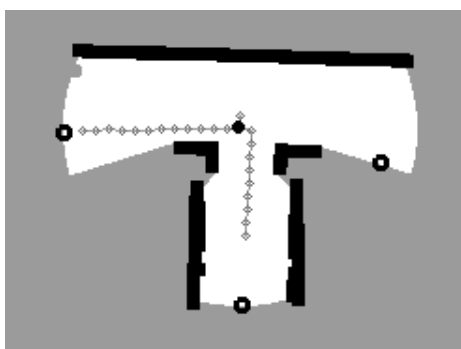


Figura 4.4: Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os três destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto. O caminho final geral parte do robô para a fronteira de número 1.

Após 5 minutos do início da exploração, o robô atualiza o mapa e identifica quatro fronteiras. As zonas de predição das fronteiras foram projetadas conforme Figura 4.5. Como pode ser observado, o robô avançou mais pelo lado esquerdo do mapa. As novas zonas de predição criadas possuem dimensões e características um pouco semelhantes.

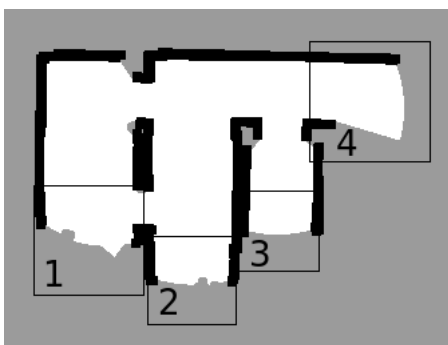


Figura 4.5: Identificação das fronteiras 1, 2, 3 e 4 e criação das zonas de predição cinco minutos após iniciar a exploração.

Nessa etapa, o processo de predição propagou as três primeiras fronteiras (1,2 e 3) usando linhas retas apenas (Figuras 4.6a, 4.6b e 4.6d). A última fronteira (4) foi propagada usando uma quina em uma das paredes (Figura 4.6e). A predição com quina apresentou novamente um espaço mais amplo. Assim, essa fronteira foi definida com um maior ganho estimado de informação (Figura 4.6c). As fronteiras de número 1, 2 e 3 obtiveram um ganho estimado, nessa ordem, menores.

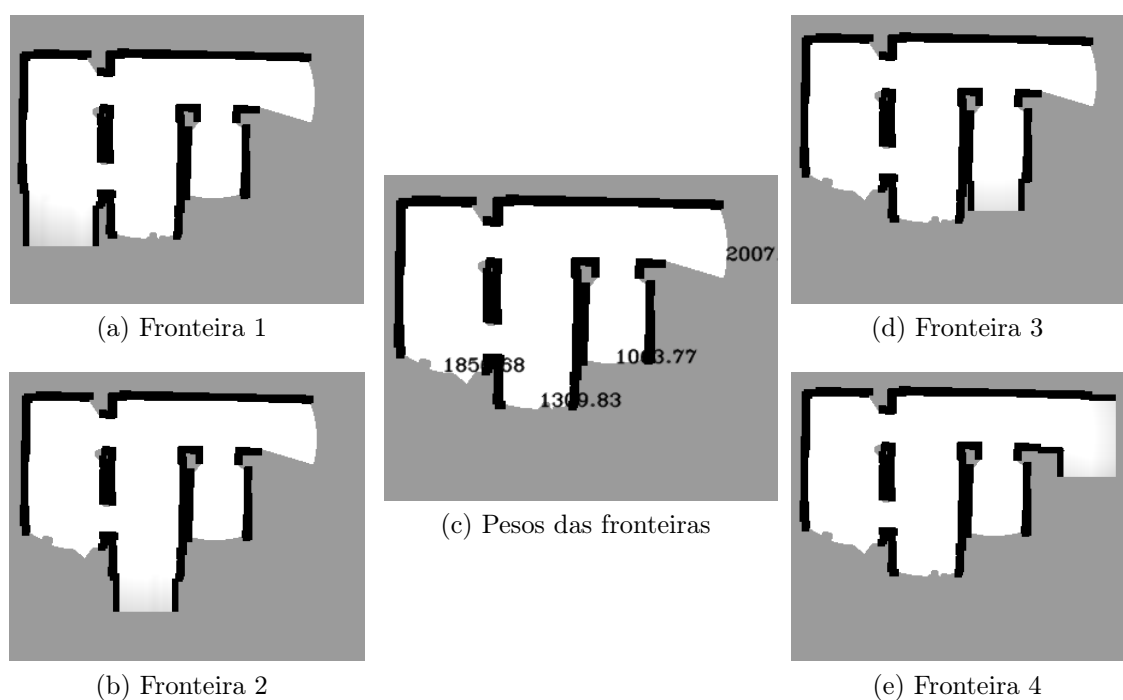


Figura 4.6: Os ganhos de informação baseado nas predições. Em (c) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas predições realizadas. (a) Predição da fronteira 1. (b) Predição da fronteira 2. (c) Predição da fronteira 3. (d) Predição da fronteira 4. (e) Predição da fronteira 4.

Novamente, para finalizar o processo de tomada de decisão, o robô gera uma árvore RRT que identifica um caminho para uma região de interesse do mapa (Figura 4.7). Nesse instante, o caminho gerado foi direcionado à fronteira de número 1. A fronteira de número 4 obteve o maior ganho estimado de informação no mapa atual, no entanto, partindo da posição do robô na direção dessa fronteira (4), qualquer nodo gerado não pode ser anexado à árvore devido a colisão com os obstáculos. Não só a fronteira de número 4, mas o mesmo ocorre para as fronteiras 2 e 3. Como resultado, a árvore expande com uma maior frequência para a fronteira de número 1 que possui um ganho estimado um pouco menor que a 4. A expansão foi interrompida quando

um novo nodo próximo a posição de destino da fronteira 1 foi anexado à árvore.

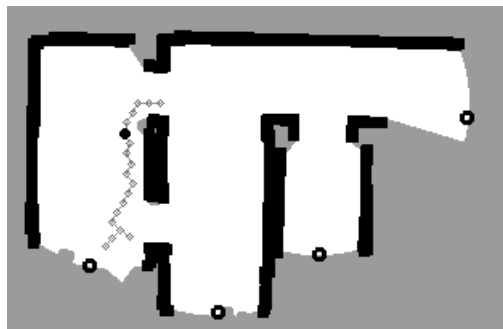


Figura 4.7: Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os quatro destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto. O caminho final geral parte do robô para a fronteira de número 1.

Comparando os resultados das estratégias após 15 minutos de exploração é possível identificar os efeitos de cada abordagem nos mapas gerados. Por exemplo, o mapa com a estratégia que busca explorar sempre as fronteiras mais próximas (Figura 4.8a) possui poucas lacunas com baixa informação espalhadas pelo mapa. É comum, nessa estratégia, o robô explorar bem uma região antes de buscar novas áreas inexploradas, diferente das estratégias que buscam o maior ganho de informação (Figura 4.8c) e a maior fronteira (Figura 4.8b). No entanto, o robô gasta muito tempo navegando pelas regiões com pouca informação, o que resulta em uma taxa de cobertura do ambiente baixa. As outras duas estratégias ignoram o custo de deslocamento do robô. Assim, diversas fronteiras pequenas vão sendo desconsideradas durante a navegação com o objetivo de explorar as que devem maximizar a cobertura do mapa. Apesar de terem uma rápida expansão nos primeiros minutos com um custo de deslocamento elevado, essas estratégias reduzem a velocidade de cobertura quando o mapa passa a não identificar novas fronteiras. Nesse momento, o robô começa a percorrer o mapa diversas vezes para preencher as lacunas que restaram. Por fim, o mapa resultado da metodologia proposta nesse trabalho (Figura 4.8d) demonstra que a estratégia busca alcançar as fronteiras de maior ganho estimado sem deixar muitas fronteiras próximas ao robô a serem exploradas.

Para finalizar o exemplo ilustrativo do ambiente escritório, após 30 minutos o mapa chega próximo a etapa final da exploração. Nesse momento, foram identificadas três fronteiras pequenas conforme Figura 4.9. Normalmente, as últimas fronteiras representam regiões com baixa informação estimada ou de difícil alcance para uma árvore RRT. Durante a exploração de todo o ambiente, a fronteira de número 1 se caracterizou pelo difícil acesso quando o robô não se localizava próximo a ela. Quando existia

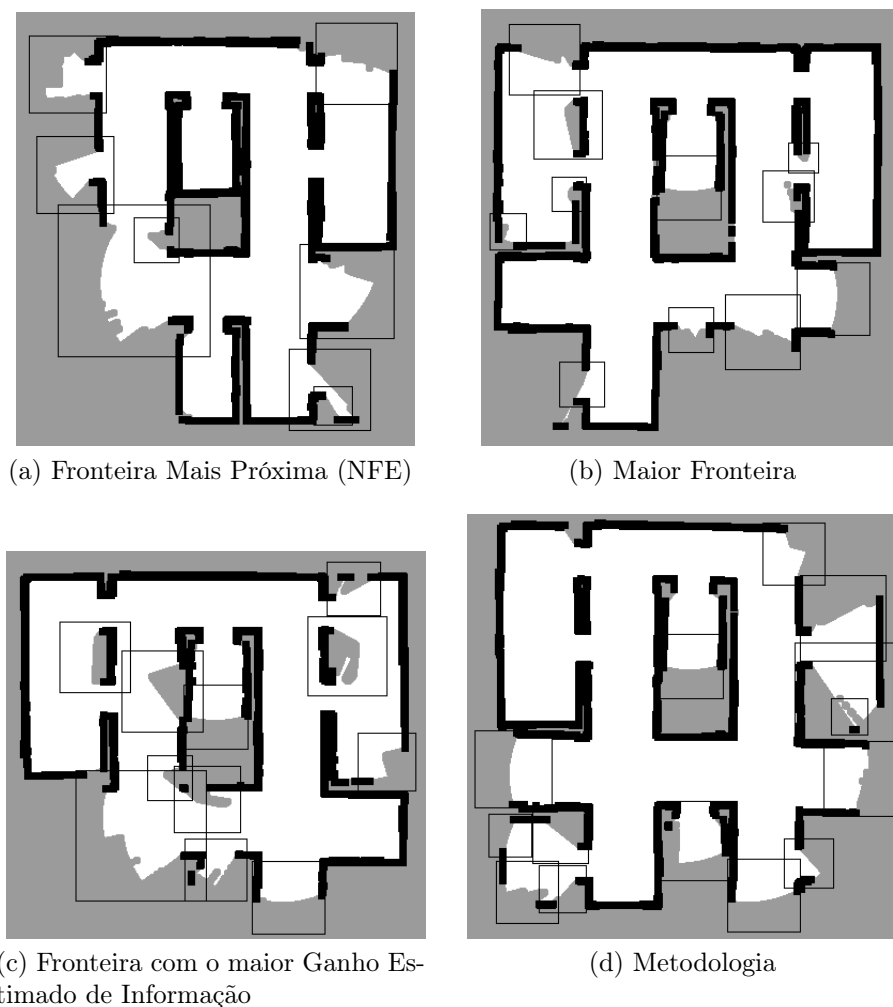


Figura 4.8: Exemplos de mapas formados após 15 minutos de execução dos experimentos para cada estratégia.

a aproximação, haviam sempre fronteiras com maiores ganhos para serem exploradas. Dessa forma, a fronteira foi deixada para ser explorada ao término da tarefa.

As fronteiras de número 2 e 3 representam cantos de uma sala com baixa informação, como pode ser visto na Figura 4.10. Essas regiões são comuns por não trazerem informação relevante ao mapa. A fronteira de número 1 representa um ambiente semelhante a um corredor ou a uma sala longa. Comparando com as outras opções de destino, a de número 1 é a mais interessante pois possui um ganho estimado de informação maior que as restantes.

Por fim, nesse momento, o robô gera uma árvore RRT que identifica um caminho para uma região de interesse do mapa (Figura 4.11). Nesse instante, o caminho gerado foi direcionado à fronteira de número 3. A árvore consiste apenas em dois nodos: a raiz indicando a posição do robô e o nodo próximo ao destino marcado na fronteira 3.

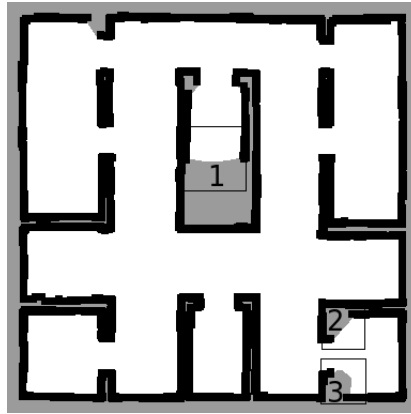


Figura 4.9: Identificação das fronteiras 1, 2 e 3 e criação das zonas de predição trinta minutos após iniciar a exploração.

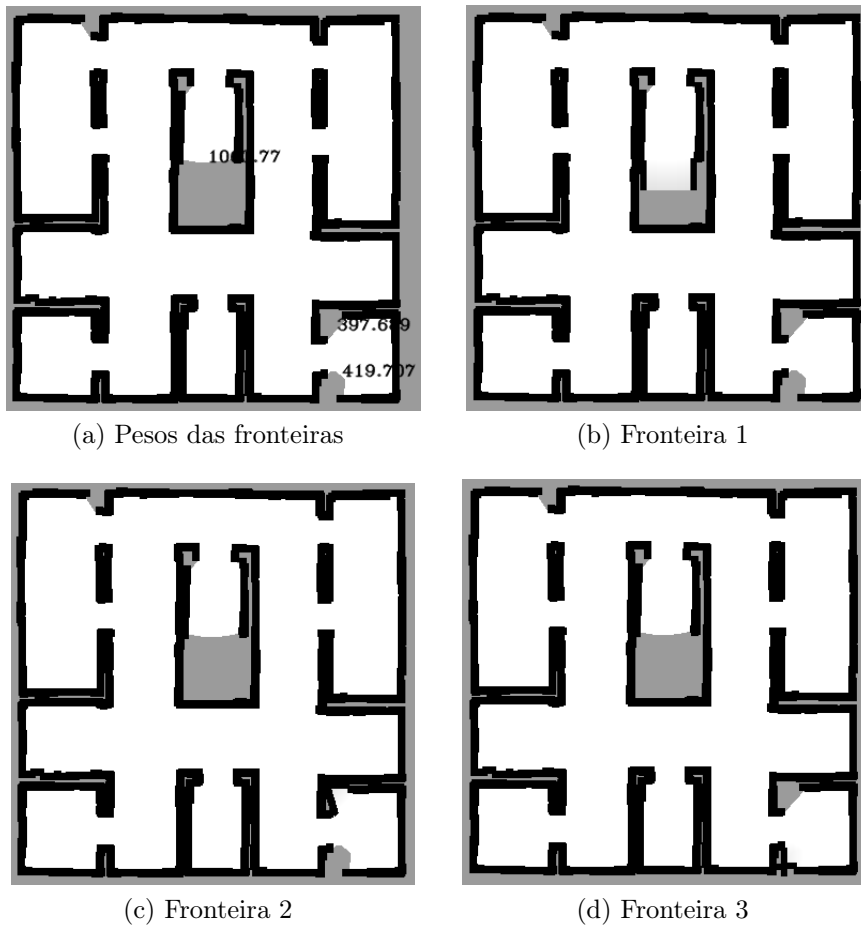


Figura 4.10: Os ganhos de informação baseado nas predições. Em (a) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas predições realizadas. (b) Predição da fronteira 1. (c) Predição da fronteira 2. (d) Predição da fronteira 3.

Devido à grande aproximação do robô com as fronteiras 2 e 3, a exploração é finalizada com a navegação por essas fronteiras seguida pela fronteira de número 1.

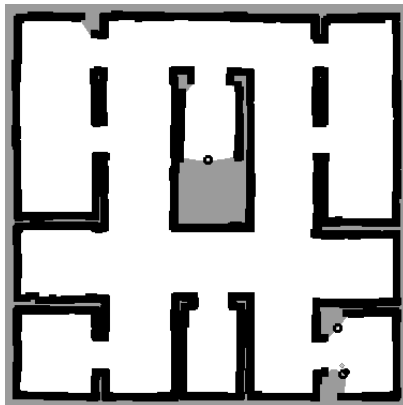


Figura 4.11: Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os três destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto. O caminho final geral parte do robô para a fronteira de número 3.

4.2.2 Caverna

Nesse experimento o robô foi inicializado na posição representada pela letra A da Figura 4.1b. Diferente do exemplo anterior, é observado que nesse novo ambiente não existem muitos obstáculos e, dessa forma, as opções de caminhos diferentes para exploração devem ser menores. No primeiro instante, o robô realiza uma observação de 360° no ambiente e gera um mapa inicial apresentado na Figura 4.12. Para esse mapa foram identificadas duas fronteiras. E, assim, duas zonas de predição foram projetadas com base no tamanho das próprias fronteiras. Nessa imagem, já pode ser observado um padrão um pouco diferente nos obstáculos do mapa em relação ao ambiente anterior. Por exemplo, algumas paredes possuem curvaturas no seu prolongamento e algumas quinas deixam de ser perpendiculares. Ainda assim, as fronteiras foram identificadas corretamente e as duas zonas de predição foram projetadas incorporando todo o espaço das fronteiras.

A fronteira de número 1 representa a região de maior interesse para ser explorada. O ganho estimado de informação para essa fronteira foi maior que a fronteira de número 2. A Figura 4.13 apresenta as predições realizadas para cada fronteira com os valores exibidos que correspondem ao ganho estimado de informação além da árvore gerada em direção à região de interesse. Nesse instante, o caminho gerado foi direcionado à fronteira de número 1. Na seleção das amostras, essa fronteira gerou um grande

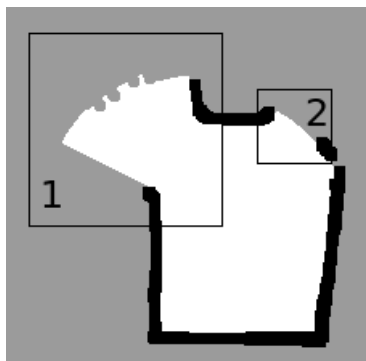


Figura 4.12: Identificação das fronteiras 1 e 2 e criação das zonas de predição na etapa inicial de exploração para o ambiente Cave.

efeito expandindo quase todos os nodos da árvore nessa direção. A árvore possui uma ramificação pequena devido a baixa taxa de aleatoriedade das amostras. A expansão foi interrompida quando um novo nodo próximo à posição de destino da fronteira 1 foi anexado à árvore.

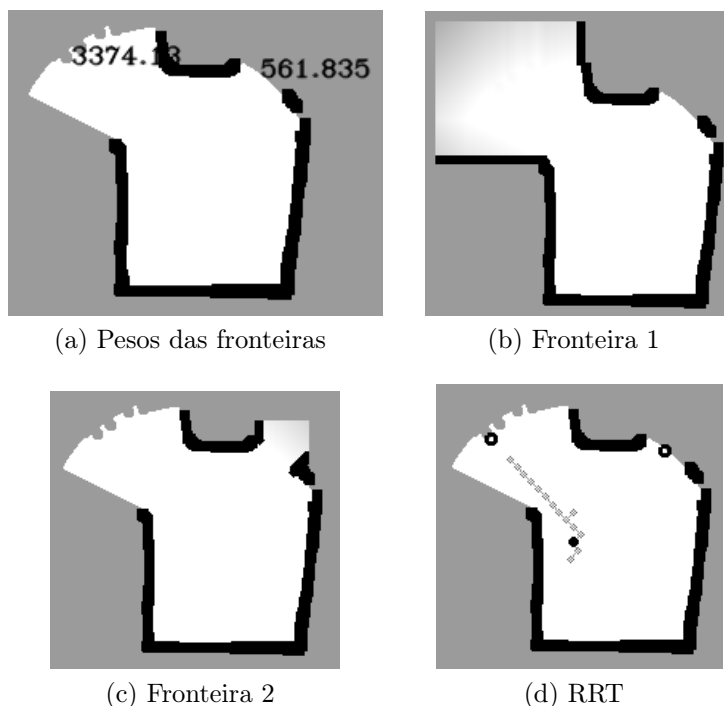


Figura 4.13: Os ganhos de informação baseado nas predições. Em (a) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas predições realizadas. (b) Predição da fronteira 1. (c) Predição da fronteira 2. (c) Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os dois destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto.

Após 5 minutos do início da exploração, o robô atualiza o mapa e identifica quatro fronteiras. As zonas de predição das fronteiras foram projetadas conforme Figura 4.14. Como pode ser observado, duas das fronteiras possuem zonas de predição projetadas maiores que as demais. As fronteiras 1 e 4 representam regiões com características que não devem trazer muita informação ao serem exploradas.

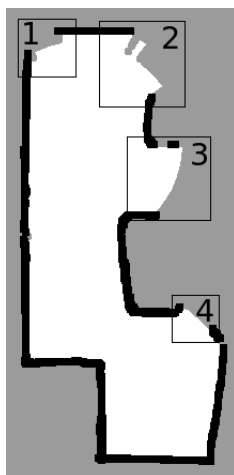


Figura 4.14: Identificação das fronteiras 1, 2, 3 e 4 e criação das zonas de predição cinco minutos após iniciar a exploração.

Usando linhas retas apenas, as paredes da fronteira de número 1 foram expandidas limitando a propagação da predição e, conseqüentemente, reduzindo o ganho estimado de informação. A fronteira de número 4, apesar da expansão das paredes não bloquearem a propagação da predição, as dimensões da zona de predição não permitem obter um ganho de informação relevante. As duas principais fronteiras nesse instante da exploração foram a de número 2 e 3 (Figura 4.15a). A fronteira de número 2 representou o maior ganho estimado de informação. É possível ver claramente nessa fronteira a redução na certeza da predição pela propagação das células brancas (Figura 4.15b). Essa característica é importante para evitar predições com baixa acurácia em fronteiras grandes. Semelhante às outras etapas, a árvore expandiu em direção a região de maior ganho estimado com ramificações para as outras fronteiras (Figura 4.15d).

Comparando os resultados das estratégias após 15 minutos de exploração é possível identificar os efeitos de cada abordagem nos mapas gerados (Figura 4.16). Por exemplo, o mapa com a estratégia que busca explorar sempre as regiões mais próximas (Figura 4.16a) possui fronteiras com um grande potencial de ganho de informação ainda inexploradas. Os mapas dos outros métodos gulosos (Figuras 4.16b e 4.16c) são caracterizados por diversas lacunas inexploradas em todas as regiões do mapa. Fronteiras menores e com baixo ganho estimado de informação são deixadas por último na

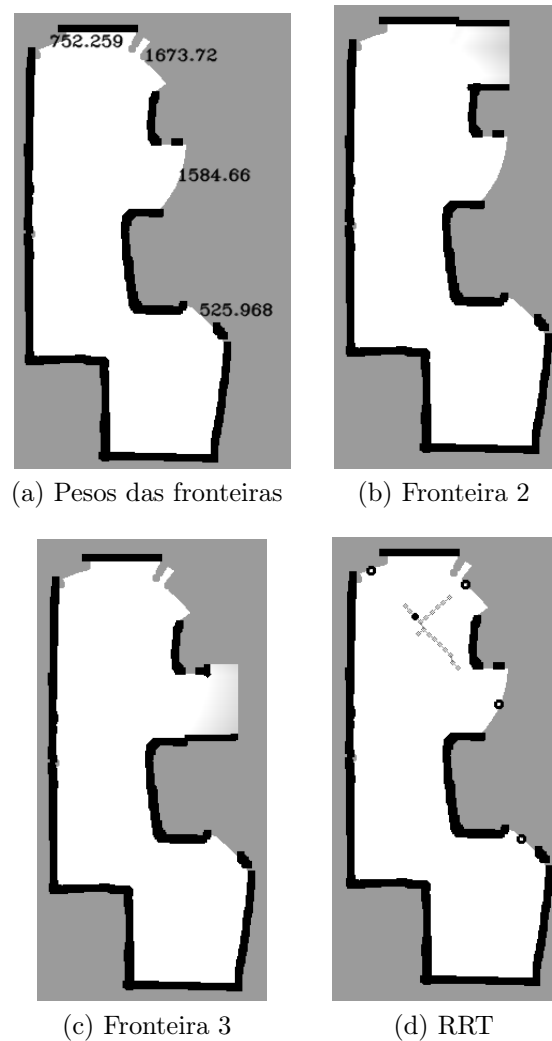


Figura 4.15: Imagens formadas com as previsões, os valores dos ganhos de informação e o caminho gerado para exploração. Em (a) são exibidos os valores do ganho estimado de informação para cada fronteira com base nas previsões realizadas. (b) Predição da fronteira 1. (c) Predição da fronteira 2. (c) Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os quatro destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto.

lista dos melhores destinos. Por fim, o mapa resultado da metodologia proposta nesse trabalho (Figura 4.16d) demonstra que a estratégia busca ainda alcançar as fronteiras de maior ganho estimado sem deixar muitas fronteiras próximas ao robô sem serem exploradas.

Para finalizar o exemplo do ambiente caverna, após 25 minutos o mapa chega próximo a etapa final da exploração. Nesse momento, foram identificadas duas fronteiras conforme Figura 4.17. Normalmente, as últimas fronteiras representam regiões com

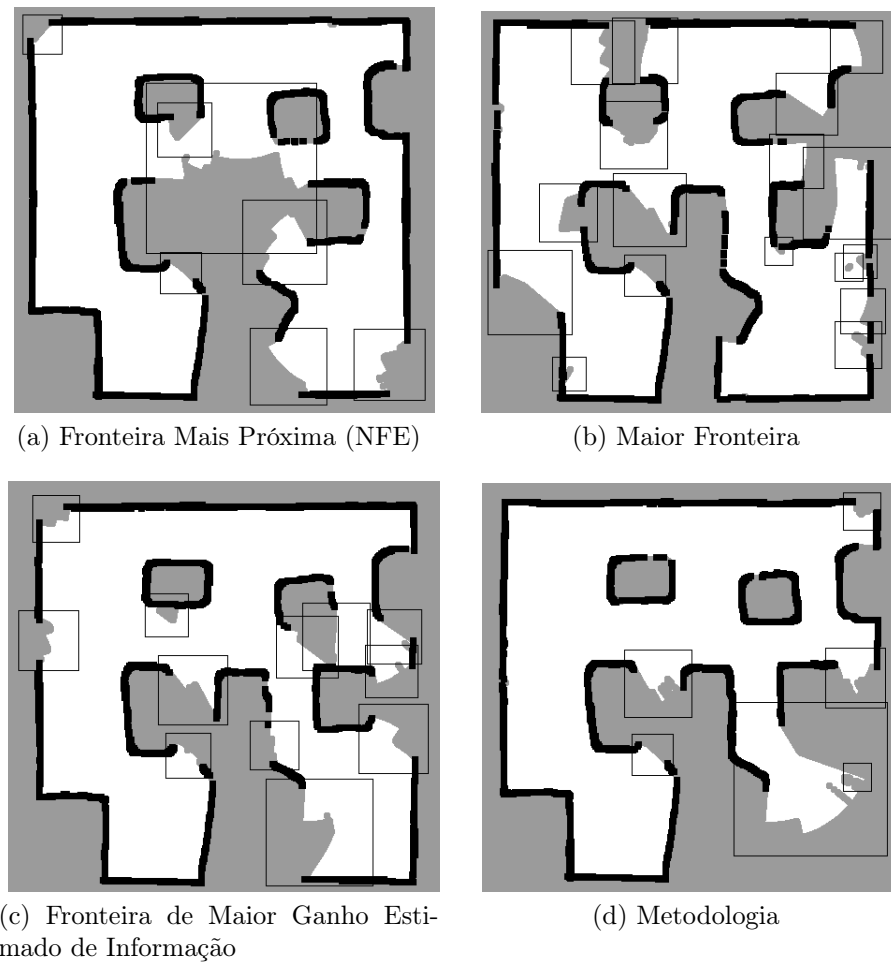


Figura 4.16: Exemplos de mapas formados após 15 minutos de experimentos para cada estratégia.

baixa informação estimada ou de difícil alcance para uma árvore RRT. Nesse caso, o ganho estimado de informação dessas regiões foram ignorados por existir sempre fronteiras próximas com ganhos maiores.

Por fim, nesse momento, o robô gera uma árvore RRT que identifica um caminho para uma região de interesse do mapa (Figura 4.18). Como existe um bloqueio para os principais destinos, diversas amostras aleatórias são geradas. Devido a uma maior aproximação com a fronteira 2, a exploração final ocorre através da navegação por essa fronteira seguida pela fronteira de número 1.

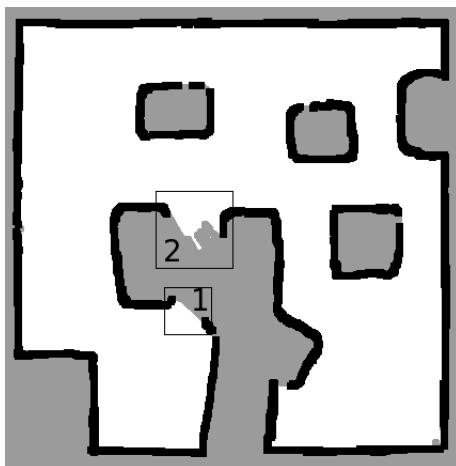


Figura 4.17: Identificação das fronteiras 1 e 2 e criação das zonas de predição na etapa final de exploração.

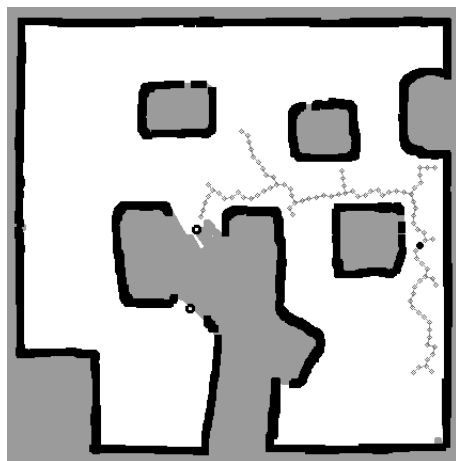


Figura 4.18: Árvore gerada para identificação de uma rota para exploração. Os círculos nas fronteiras indicam os dois destinos definidos como bias na expansão da árvore. A árvore está representada em cinza com o nó raiz, indicando a posição do robô, em preto. O caminho final geral parte do robô para a fronteira de número 2.

4.3 Comparativo Geral

Nessa seção são comparadas as soluções entre as abordagens discutidas anteriormente com a metodologia proposta nesse trabalho. Além disso uma análise é feita entre as métricas gulosas que buscam as maiores fronteiras e as de maior ganho estimado de informação. As métricas aplicadas na análise dos resultados foram a completude do mapa pelo tempo de exploração e o custo total no deslocamento do robô por todo o ambiente na construção de um mapa completo. O objetivo principal dessa análise consiste em mostrar o efeito da cobertura do ambiente pelo robô nos minutos iniciais

em cada cenário de teste. Assim, para aplicações em que possa existir uma restrição de tempo, a abordagem mais apropriada pode ser selecionada.

4.4 Comparativo da Metodologia

Os experimentos e suas respectivas análises estatísticas foram feitas em dois ambientes. Para cada estratégia, 45 experimentos foram executados, 15 para cada posição inicial (Figura 4.1). Cada experimento teve como meta explorar autonomamente todo o ambiente. Assim, os gráficos gerados apresentam a evolução da expansão do mapa para cada estratégia. Foi utilizada apenas a média dos valores para exibir os resultados. Como o nível de dispersão entre os experimentos foi pequeno, não foram inseridas as faixas relativas ao desvio padrão.

4.4.1 Escritório

Com o robô iniciado nas posições A, B e C (Figura 4.1a), é possível identificar uma evolução muito menor do NFE na cobertura do mapa quando comparado com as outras técnicas. Isso ocorre devido a constante exploração de lacunas no mapa próximas ao robô com baixo ganho de informação. Outras abordagens gulosas demonstraram rápida expansão na cobertura do mapa com resultados bastante semelhantes. Apesar de atingirem resultados pouco melhores que a metodologia proposta em alguns momentos, os gráficos que apresentam os resultados relativos ao custo total de deslocamento do agente mostram que a metodologia reduz bastante a navegação do agente pelo ambiente. Isso prova que o agente consegue ser próximo dos algoritmos gulosos reduzindo bastante o custo e energia necessária para deslocar o robô pelo ambiente.

No gráfico com os resultados da posição A (Figura 4.19), pode ser observado que entre os 10 a 20 minutos de exploração a taxa de dispersão dos resultados é levemente aumentada. Essa característica ocorre pelo aumento do número de fronteiras no mapa e, conseqüentemente, o aumento nas possíveis formas de se explorar o ambiente. Além disso, é possível notar que os resultados das técnicas passam um pouco da completude de 100% nos últimos segundos do gráfico. Isso ocorre porque os experimentos executados terminam a tarefa com um valor de informação total do mapa um pouco diferente um dos outros. Assim, foi definido um valor fixo como base para determinar a total completude do mapa.

O custo no deslocamento total do robô (Figura 4.20) para a estratégia NFE foi o menor, como já era esperado. Essa estratégia busca explorar todas as regiões do mapa em partes. Assim, realiza poucos retornos às regiões já exploradas para preencher

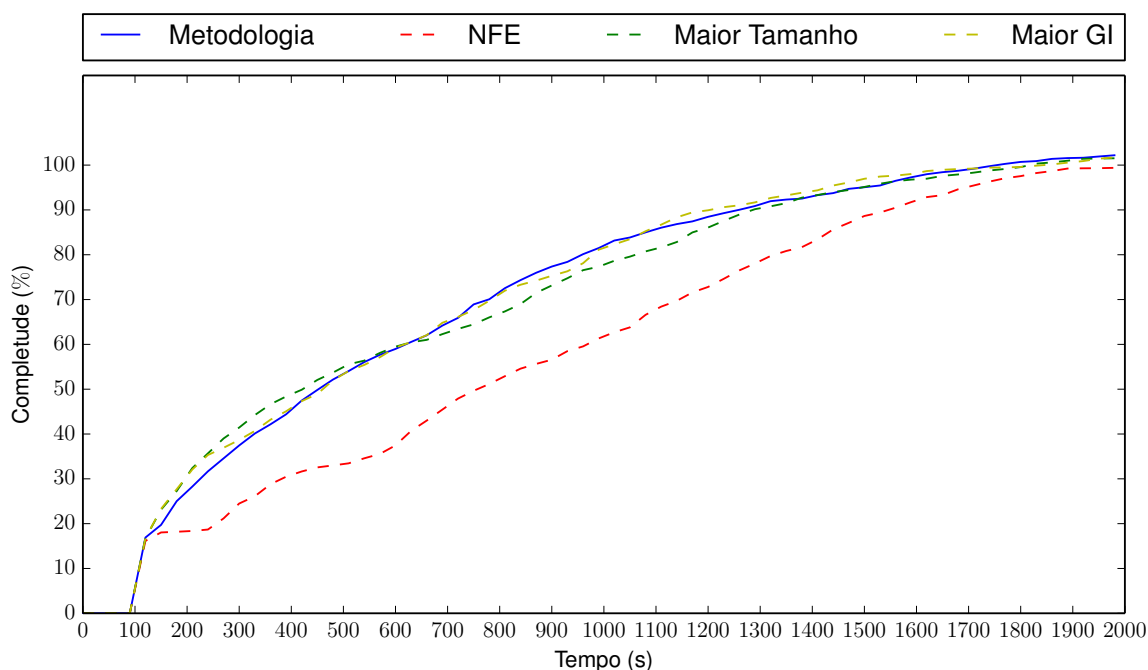


Figura 4.19: Gráfico com a média do percentual de completude do mapa pelo tempo para a posição A do mapa Escritório.

alguns espaços ignorados anteriormente. Os outros algoritmos gulosos foram os que obtiveram maior custo de deslocamento. Como já explicado, diversas vezes o robô precisa retornar para regiões já navegadas para explorar fronteiras consideradas como de maior interesse num determinado instante. A metodologia conseguiu trazer um resultado próximo ao NFE com uma evolução no mapa próxima aos outros métodos gulosos.

Os experimentos para a posição B demonstraram características similares à posição A em relação aos dois critérios (Figuras 4.21 e 4.22). No entanto, nessa posição, os resultados da estratégia referente a metodologia foram iguais ou melhores, em quase todo o processo de exploração, comparado com as demais abordagens exceto pela NFE.

A última posição (C) apresentou um resultado diferente dos demais (Figura 4.23). Nessa posição inicial, existiam poucas opções de destino e a tomada de decisão para cada estratégia foram bastante semelhantes. Ou seja, as fronteiras mais próximas correspondiam as de maior tamanho e as de maior ganho estimado de informação. Após explorar as primeiras fronteiras e com o aumento do número de destinos, as estratégias começaram a desmonstrar resultados mais parecidos com os anteriores. Após 20 minutos de exploração, os experimentos se aproximam da etapa final de completude do mapa. Noca fronteiras deixam de existir e inicia a etapa de preencher lacunas do mapa. Os resultados dessa posição são importantes pois mostram a variação que pode

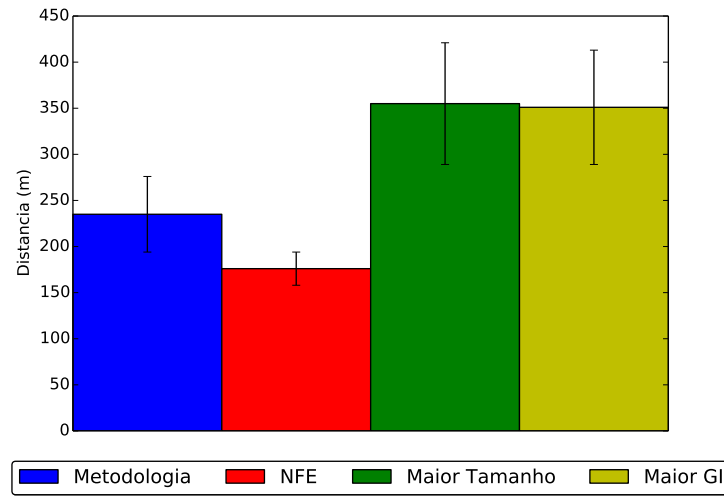


Figura 4.20: Deslocamento total do agente durante a tarefa de exploração partindo da posição A do mapa Escritório.

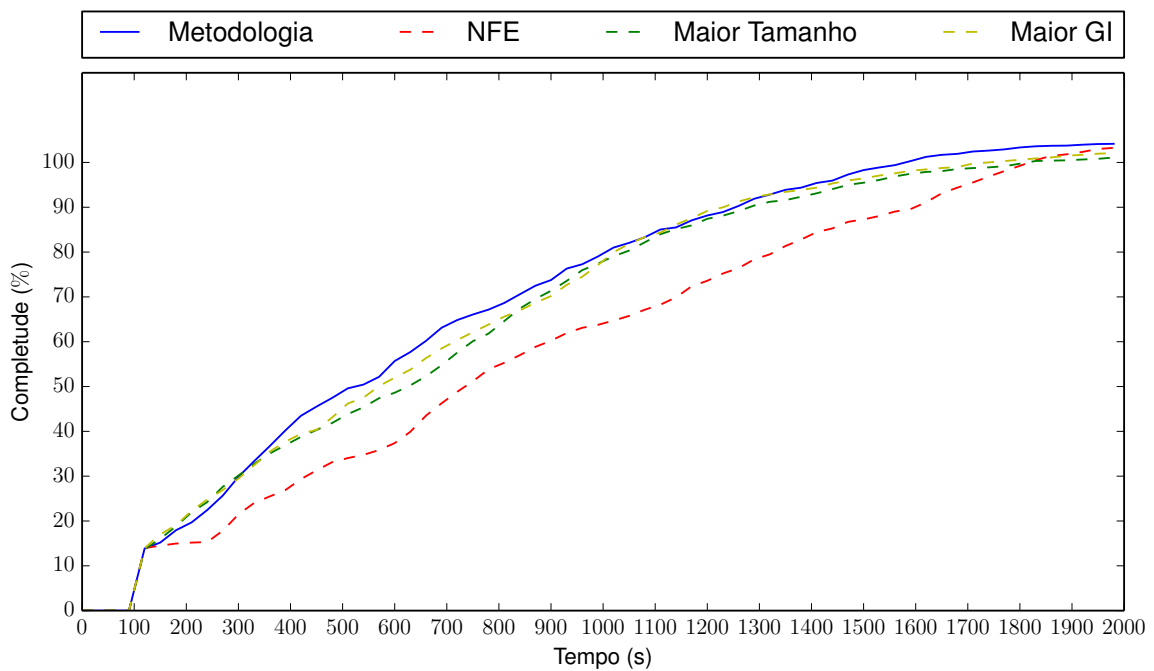


Figura 4.21: Gráfico com a média do percentual de completude do mapa pelo tempo para a posição B do mapa Escritório.

existir dos resultados com a mudança na posição inicial do robô na tarefa de exploração autônoma. Apesar de um resultado um pouco mais próximo do NFE, o deslocamento da metodologia demonstrou um resultado bastante próximo ao NFE (Figura 4.24).

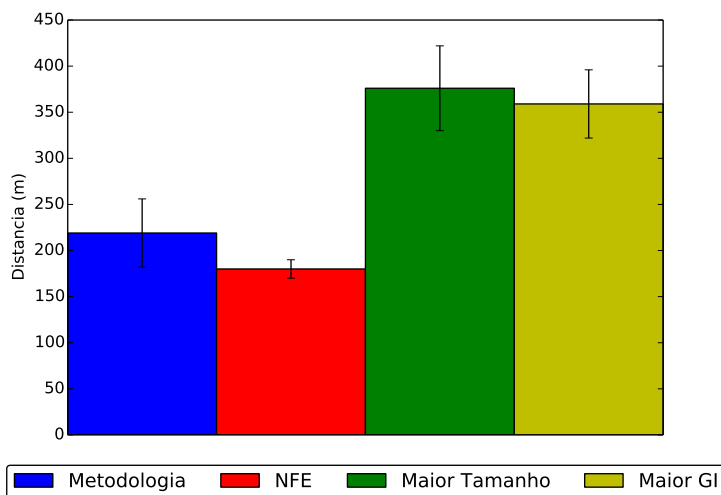


Figura 4.22: Deslocamento total do agente durante a tarefa de exploração partindo da posição B do mapa Escritório.

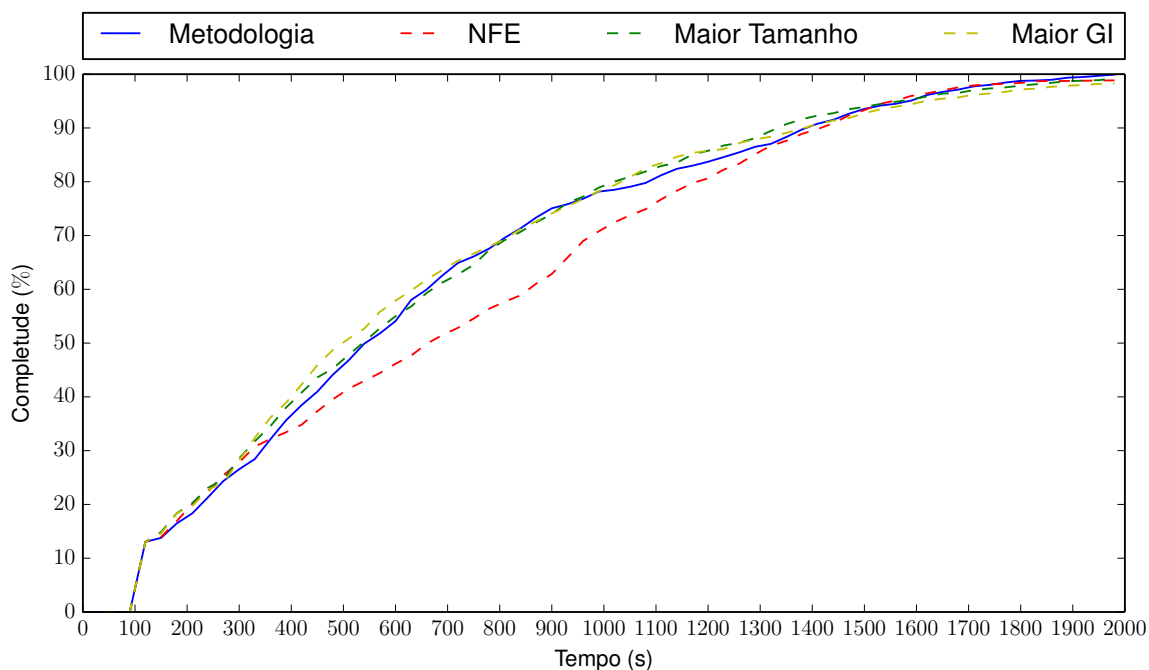


Figura 4.23: Gráfico com a média do percentual de completude do mapa pelo tempo para a posição C do mapa Escritório.

4.4.2 Caverna

Para esse novo ambiente, o método NFE obteve resultados um pouco melhores ao serem comparados aos resultados no ambiente anterior. Isso ocorreu por ser um ambiente mais aberto, com poucas quinas que demandam uma exploração com baixo ganho de informação. Contudo, as outras abordagens ainda foram melhores que o NFE, ex-

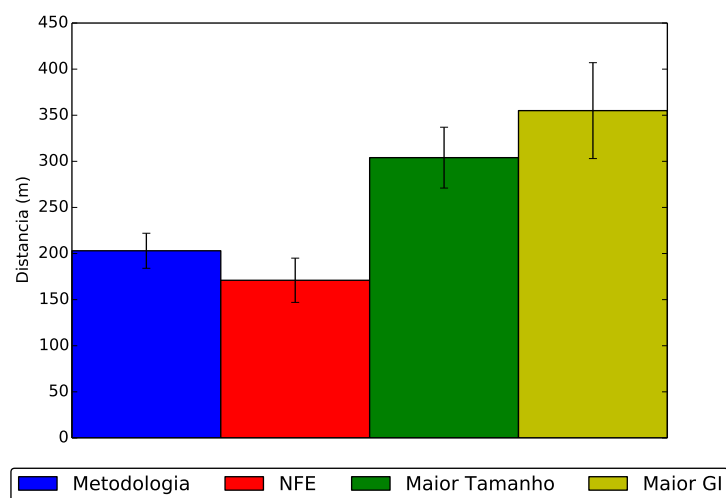


Figura 4.24: Deslocamento total do agente durante a tarefa de exploração partindo da posição C do mapa Escritório.

andando mais rapidamente o mapa. Essas abordagens gulosas demonstraram uma expansão um pouco mais rápida na cobertura do mapa com resultados semelhantes ao ambiente anterior. Devido a um acesso mais fácil às lacunas com baixo ganho de informação próximas ao robô, a expansão do mapa através da metodologia foi um pouco mais lenta que as abordagens gulosas, exceto pelo NFE (Figuras 4.25, 4.27 e 4.29). Contudo, apesar de atingirem resultados um pouco melhores que a metodologia proposta, os gráficos que apresentam os resultados relativos ao custo total de deslocamento do agente mostram que a metodologia ainda reduz bastante a navegação do agente pelo ambiente. Isso prova que o agente consegue ser próximo dos algoritmos gulosos reduzindo o custo e energia necessária para deslocar o robô pelo ambiente.

As observações realizadas nos resultados do ambiente anterior também são aplicadas aos resultados nesse ambiente. A principal diferença que pode ser observado entre os gráficos apresentados é uma maior aproximação no custo de deslocamento da metodologia com o NFE (Figuras 4.26, 4.28 e 4.30). Ou seja, a metodologia nesse ambiente buscou mais explorar as regiões próximas ao robô sem deixar de guiar o agente pelas regiões de maior interesse.

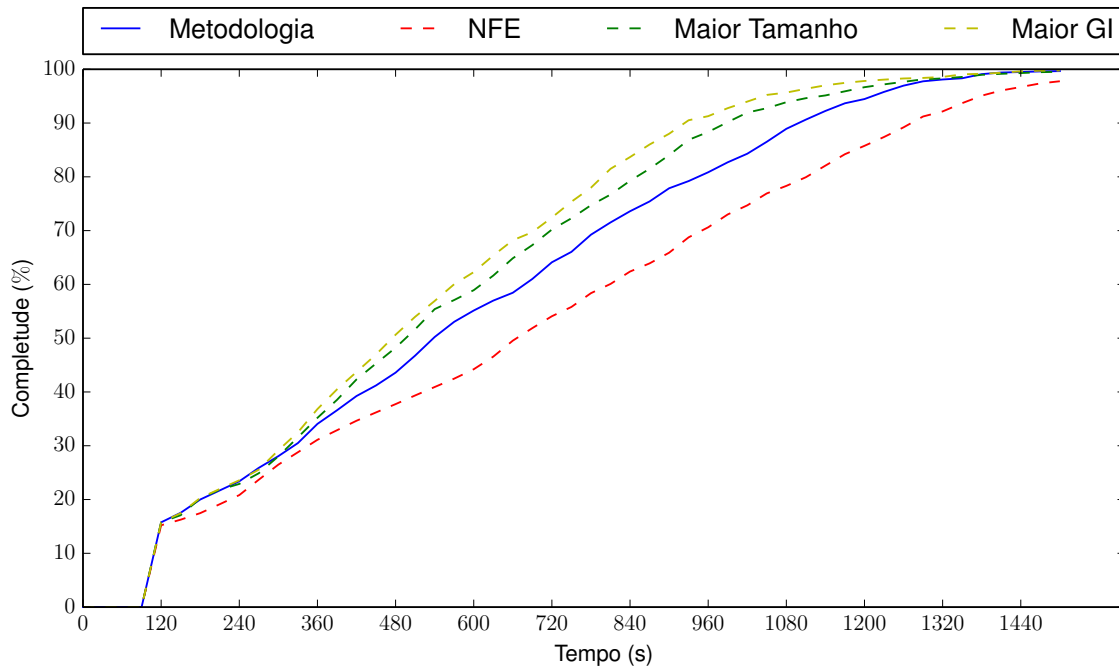


Figura 4.25: Gráfico com a média do percentual de completude do mapa pelo tempo para a posição A do mapa Caverna.

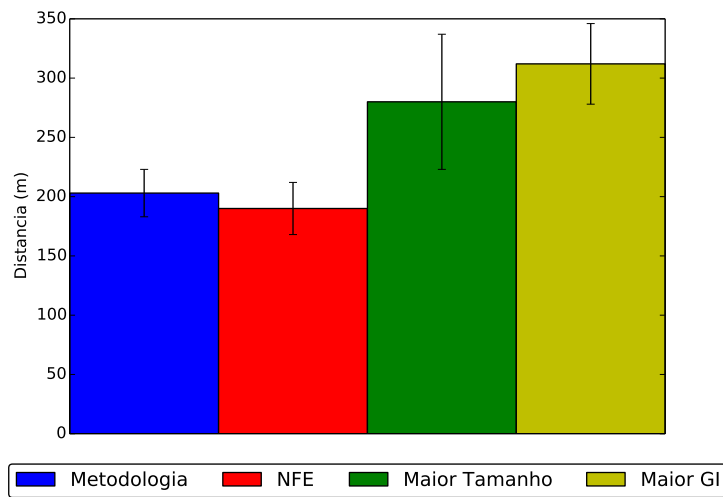


Figura 4.26: Deslocamento total do agente durante a tarefa de exploração partindo da posição A do mapa Caverna.

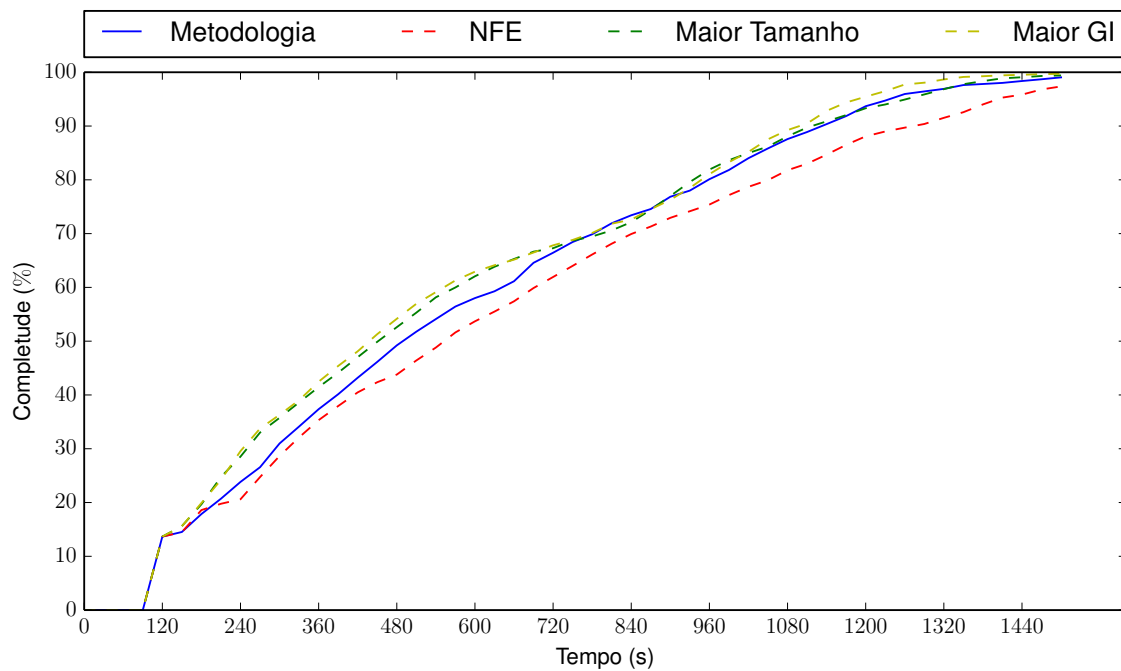


Figura 4.27: Gráfico com a média do percentual de completude do mapa pelo tempo para a posição B do mapa Caverna.

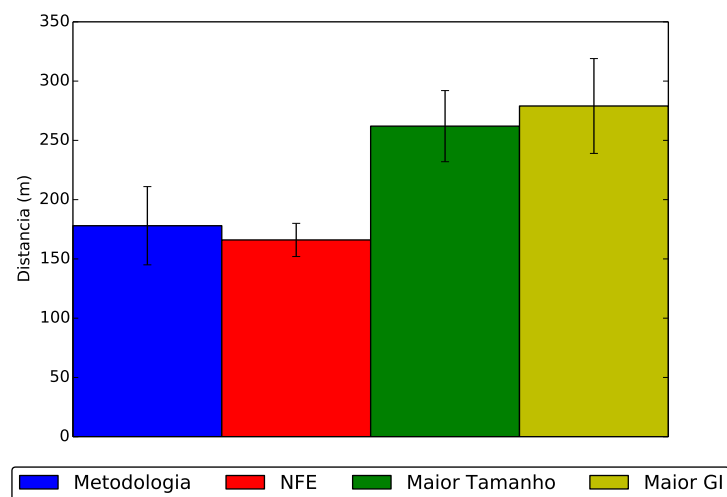


Figura 4.28: Deslocamento total do agente durante a tarefa de exploração partindo da posição B do mapa Caverna.

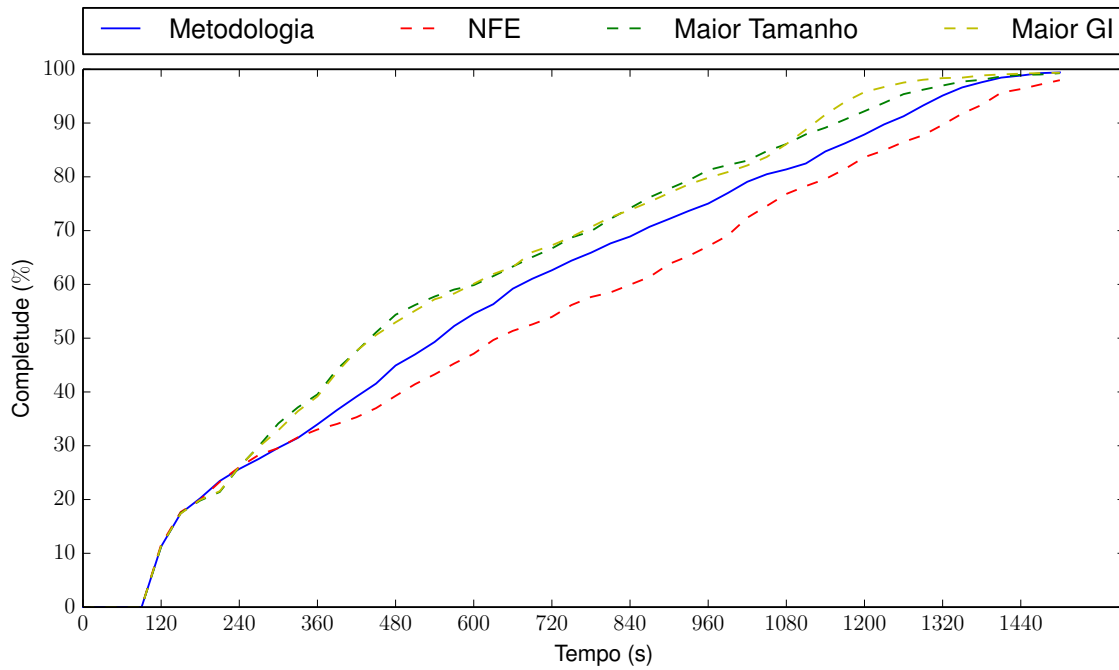


Figura 4.29: Gráfico com a média do percentual de completude do mapa pelo tempo para a posição C do mapa Caverna.

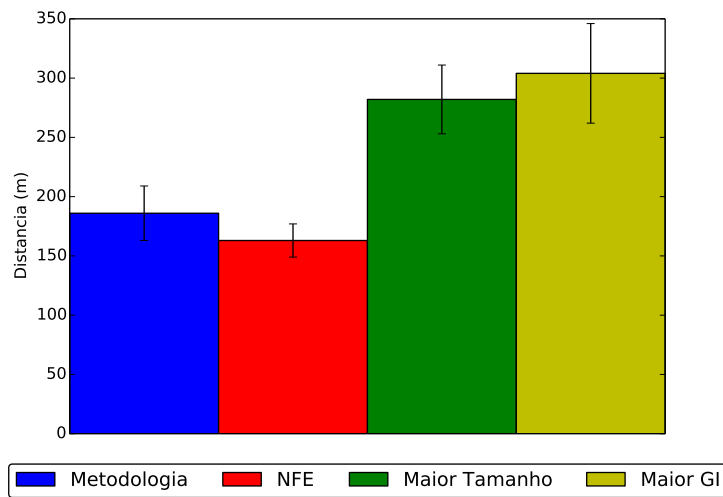


Figura 4.30: Deslocamento total do agente durante a tarefa de exploração partindo da posição C do mapa Caverna.

4.5 Comparativo: Maiores Fronteiras vs. Maiores Ganhos Estimados de Informação

Existem casos em que uma fronteira longa pode não trazer um ganho estimado de informação grande, como seria o esperado. Os exemplos da Figura 4.31, demonstram alguns casos em que as fronteiras longas obtiveram um ganho estimado de informação pequeno. Nesses casos, os prolongamentos das paredes limitaram a possibilidade de expansão das células livres e, assim, o valor do ganho foi reduzido.

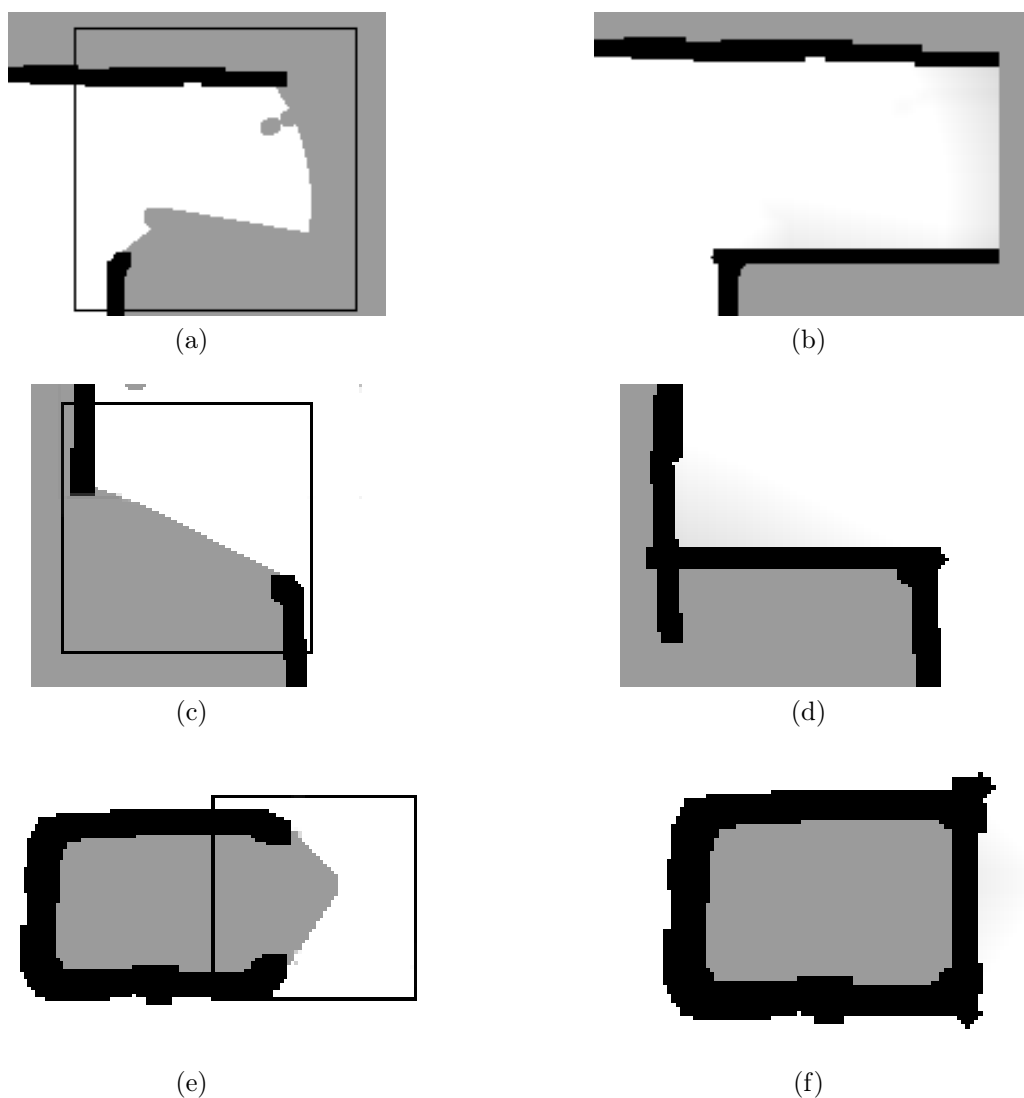


Figura 4.31: Exemplos de fronteiras em que o ganho estimado de informação foi reduzido devido ao prolongamento das paredes. As imagens do lado esquerdo representam partes do mapa com as zonas de predição projetadas e as imagens do lado direito exemplos de fronteiras em que existiu uma limitação da expansão das áreas livres pelo prolongamento das paredes.

No entanto, esses casos não são muito comuns e, por consequência, os resultados entre essas duas abordagens foram similares para todos os cenários de teste. Mesmo em ambientes diferentes, onde os obstáculos apresentaram características diferentes, ainda foram semelhantes. Isso prova que o ganho estimado de informação das regiões desconhecidas próximas às fronteiras são proporcionais ao tamanho das próprias fronteiras.

4.6 Exemplo de Exploração com Erro na Construção do Mapa

Qualquer algoritmo usado na solução do problema SLAM depende dos dados coletados pelos sensores para a construção do mapa e localização do agente. Se os sensores possuírem uma taxa de erro elevada, o mapa poderá ser formado com baixa acurácia em relação ao estado real do ambiente. A Figura 4.32 demonstra exemplos de mapas construídos usando sensores de odometria com 0%, 10% e 20% de erro linear e angular. Como pode ser visto, em ambientes pequenos, já é possível visualizar a má formação do mapa mesmo com uma taxa de erro não muito elevada.

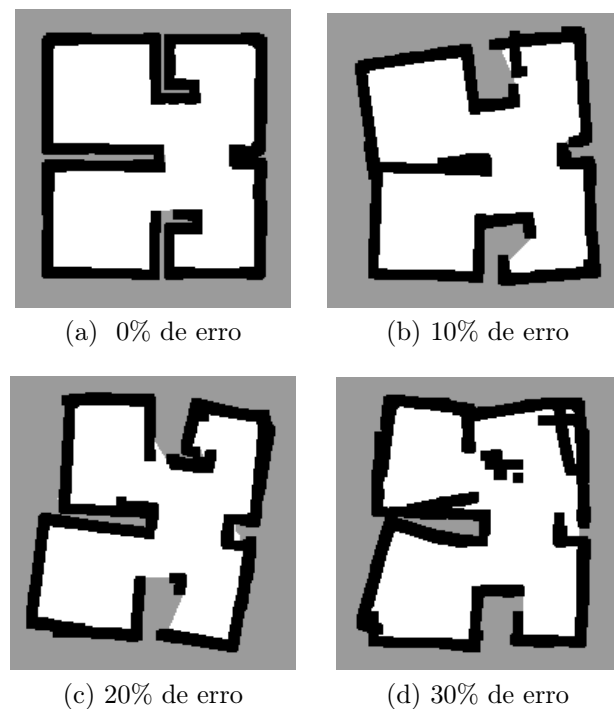


Figura 4.32: Exemplos de mapas após a exploração do mesmo ambiente com diferente taxas de erro.

Um mapa com qualidade ruim pode prejudicar a metodologia de diferentes formas. Primeiro, fronteiras que não existem podem ser criadas tornando-as impossíveis de serem exploradas. Paredes plotadas em lugares que deveriam ser livres para navegação podem bloquear a expansão da árvore RRT e, assim, não permitir alcançar regiões de maior interesse para serem exploradas. Além disso, com o erro, o traçado das paredes começa a realizar curvas que não existem. Para a metodologia atual, os traçados precisam ser retos ou com uma leve curvatura. O contrário deve gerar previsões completamente incorretas. Desse modo, a metodologia proposta nesse trabalho depende do bom funcionamento do módulo responsável pelo problema SLAM no sistema.

O exemplo a seguir (Figura 4.33) apresenta a evolução de um experimento realizado no mapa Escritório com o sensor de odometria a uma taxa de erro de 10%. Como a taxa de erro não é muito alta, foi possível, nesse experimento, obter uma exploração cobrindo praticamente todo o ambiente. No entanto, a Figura 4.34 demonstra que o mapa gerado nas etapas finais possui uma qualidade inferior aos mapas gerados sem erro. Devido aos problemas já citados anteriormente, a evolução da exploração usando a metodologia com erro foi mais lenta que a média apresentada anteriormente.

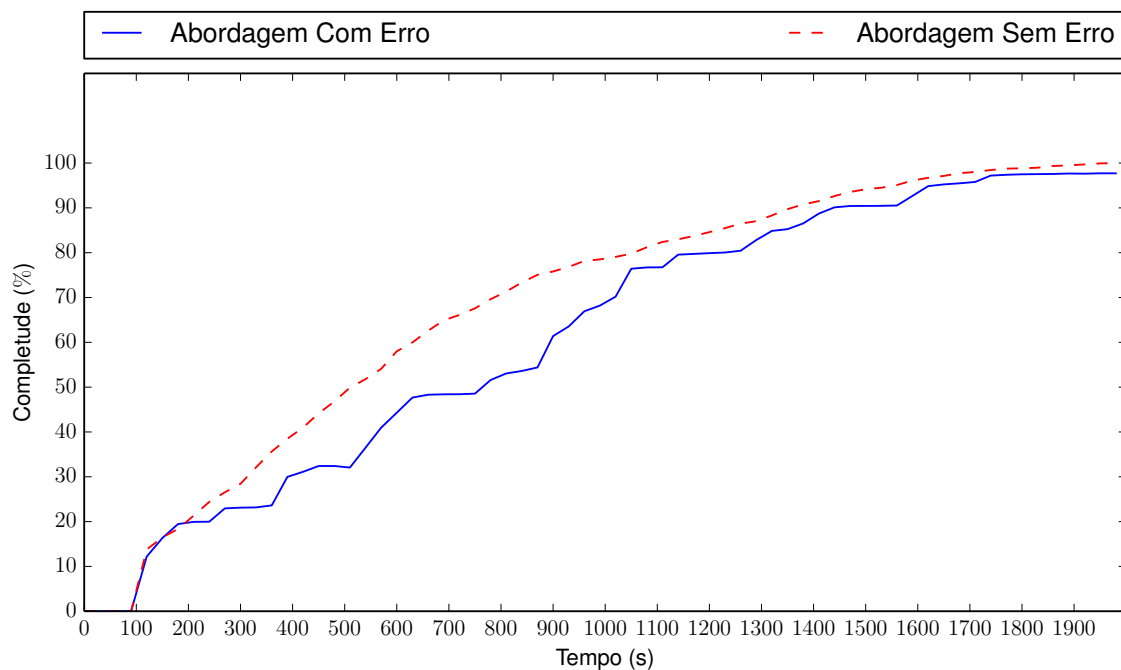


Figura 4.33: Comparação entre experimentos de uma abordagem com e sem erro no sensor de odometria.

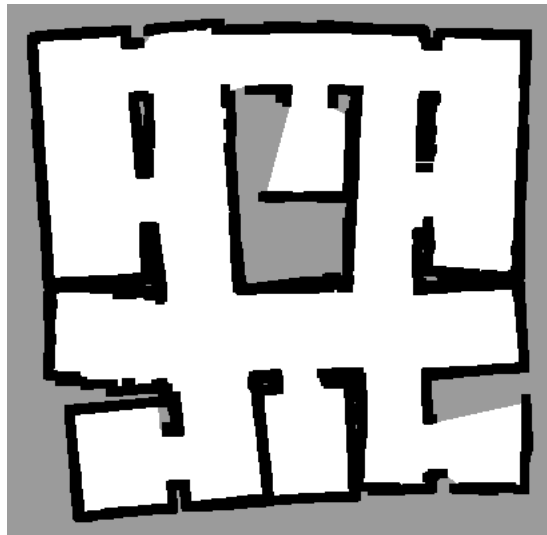


Figura 4.34: Mapa formado com qualidade baixa na etapa final de uma exploração.

Capítulo 5

Conclusão e Trabalhos Futuros

5.1 Conclusão

Nesse trabalho foi apresentada uma nova abordagem com o objetivo de otimizar o problema da exploração autônoma de ambientes usando um robô móvel. Esse problema consiste na criação de uma representação do ambiente completa através da navegação do robô pelas regiões inexploradas. Assim, uma boa estratégia deve identificar as melhores regiões do mapa para serem cobertas, ou seja, selecionar os destinos que devem agregar mais informação ao mapa em construção. Além disso, é interessante que o custo energético relativo ao deslocamento do robô seja reduzido.

Com o desenvolvimento do trabalho, duas contribuições foram geradas: a predição e o ganho estimado de informação com base na propagação das paredes que limitam as fronteiras e o método unificado para identificação dos melhores destinos com a formação das rotas desejadas. Apesar de ser uma análise local, as predições permitem selecionar destinos que beneficiam a expansão do mapa, como pode ser visto nos resultados dos experimentos desse trabalho. Além disso, a formação das rotas através do método adaptado da RRT permite uma análise das fronteiras através do ganho estimado de informação junto com outros critérios de forma indireta.

A metodologia proposta demonstrou uma expansão mais rápida do mapa em relação ao método clássico NFE para ambos os ambientes e em todos os cenários apresentados. Em relação às outras duas abordagens gulosas, maior fronteira e maior ganho estimado de informação, a metodologia demonstrou um resultado semelhante na expansão do mapa com algumas variações. Por exemplo, a metodologia apresentou resultados mais interessantes quando o ambiente possuía características mais complexas e com áreas abertas menores. No entanto, o custo total no deslocamento do agente por todo o ambiente através do método proposto foi bastante reduzido para qualquer

cenário, chegando próximo ao NFE. O custo total de deslocamento das demais abordagens foram bem elevados, chegando a valores próximos ao dobro do NFE. Dessa forma, é possível aplicar a metodologia proposta nesse trabalho em diversas aplicações reais e obter um mapa com mais informação em poucos minutos de exploração e com custo energético menor para deslocar o agente.

Apesar dos resultados positivos, a metodologia ainda possui algumas limitações, como depender de mapas regulares e com boa qualidade para a realização das predições com sucesso. Se o ambiente a ser explorado possuir muitos elementos que podem trazer uma irregularidade no traçado das paredes ou se a taxa de erro do sensor não gerar mapas com uma boa qualidade, a tarefa de exploração não será otimizada. Assim, para que a metodologia desenvolvida seja eficiente, é importante a formação de um mapa com traços retos e regulares. Além disso, para que a exploração possa ser realizada por completa, é importante a formação de um mapa próximo do estado real do ambiente. Caso contrário, o robô tentará se deslocar continuamente para fronteiras que não existem com o objetivo de reduzir a incerteza do mapa.

5.2 Trabalhos Futuros

O uso de multi-robôs representa uma característica importante para a tarefa de exploração autônoma de ambientes desconhecidos. Com um número maior de agentes, é possível reduzir o tempo de cobertura do ambiente além de garantir um sistema mais tolerante a falhas. Por exemplo, os erros de leitura dos sensores podem ser corrigidos, quando os mapas gerados por cada robô forem analisados para a formação do mapa final, além de aumentar as chances de obter um mapa final mesmo com a perda de alguns agentes. Dessa forma, é interessante a extensão do trabalho para o uso de multi-robôs.

Conseqüentemente, novos problemas deverão ser tratados, como a coordenação dos agentes em ambientes em que não é possível uma comunicação efetiva entre os robôs. Assim, uma estratégia deve ser aplicada para evitar a exploração de regiões iguais por diferentes robôs. Além da coordenação dos agentes em diferentes regiões do ambiente, cada robô deve identificar os melhores caminhos para serem explorados visto que o tempo disponível deve ser otimizado. O processo de predição pode ser feito localmente com base nas informações conhecidas próximas às regiões inexploradas, semelhante a metodologia aqui proposta. Contudo, com o avançar da exploração, as informações obtidas do ambiente pelo robô podem ser usadas a seu favor com o objetivo de melhorar a acurácia das predições. Assim, durante uma exploração, elementos visuais podem

ser identificados e empregados como critérios importantes nas predições. Por exemplo, o robô poderá identificar o cômodo como um corredor ou sala e, com base no histórico de observações do laser, poderá inferir as dimensões desse ambiente.

Por fim, visando melhorar a metodologia proposta nesse trabalho, alguns parâmetros poderiam ser analisados e configurados de forma diferente para obter uma maior acurácia nas predições. Por exemplo, as zonas de predição poderiam ser projetadas com tamanhos e formas adaptativas de acordo com a análise das observações passadas pelo robô. A função que modela o tamanho do setor aleatório da roleta poderia ser melhor investigada, analisando o efeito da variação nos resultados. Os traçados das paredes poderiam ser analisados com base nas observações passadas e diferentes modelos de propagação poderiam ser estabelecidos. Além disso, diferentes técnicas da RRT poderiam ser aplicadas para melhorar os resultados na navegação do agente, como a RRT* que busca gerar caminhos entre dois pontos com um custo menor de deslocamento.

Referências Bibliográficas

- Amigoni, F. & Caglioti, V. (2010). An information-based exploration strategy for environment mapping with mobile robots. *Robotics and Autonomous Systems*, 54(58):684--699.
- Cipolleschi, R.; Giusto, M.; Li, A. Q. & Amigoni, F. (2013). Semantically-informed coordinated multirobot exploration of relevant areas in search and rescue settings. Em *Mobile Robots (ECMR), 2013 European Conference on*, pp. 216--221.
- Colares, R. G. & Chaimowicz, L. (2014). Information window: uma nova abordagem para exploração de fronteiras. *Anais do XX Congresso Brasileiro de Automática*.
- Colares, R. G. & Chaimowicz, L. (2015). A novel distance cost approach for multi-robot integrated exploration. Em *2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR)*, pp. 192--197.
- Colares, R. G. & Chaimowicz, L. (2016). The next frontier: Combining information gain and distance cost for decentralized multi-robot exploration. Em *Proceedings of the 31st Annual ACM Symposium on Applied Computing, SAC '16*, pp. 268--274, New York, NY, USA. ACM.
- Cover, T. M. & Thomas, J. A. (2006). *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience. ISBN 0471241954.
- Denny, J.; Morales, M.; Rodriguez, S. & Amato, N. M. (2013). Adapting rrt growth for heterogeneous environments. Em *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1772--1778. ISSN 2153-0858.
- Devol, J. (1961). Programmed article transfer. US Patent 2,988,237.
- Dissanayake, M.; Newman, P.; Clark, S.; Durrant-Whyte, H. & Csorba, M. (2002). A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotics and Automation*, 17(3):229--241.

- Doria, N. S. F.; Freire, E. O. & Basilio, J. C. (2013). An algorithm inspired by the deterministic annealing approach to avoid local minima in artificial potential fields. Em *Advanced Robotics (ICAR), 2013 16th International Conference on*, pp. 1–6.
- Doucet, A.; de Freitas, N.; Murphy, K. & Russell, S. (2000). Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks. *Conference on Uncertainty in Artificial Intelligence*, pp. 176--183.
- Droeschel, D.; Nieuwenhuisen, M.; Beul, M.; Holz, D.; Stücker, J. & Behnke, S. (2016). Multilayered mapping and navigation for autonomous micro aerial vehicles. *J. Field Robotics*, 33(4):451--475.
- Durrant-Whyte, H. & Bailey, T. (2006). Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms. *Robotics & Automation Magazine*, 13(2):99--110.
- Durrant-Whyte, H.; Rye, D. & Nebot, E. (1995). Localisation of automatic guided vehicles. *7th International Symposium of Robotics Research*.
- Elara, M. R.; Rojas, N. & Chua, A. (2014). Design principles for robot inclusive spaces: A case study with roomba. Em *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5593–5599. ISSN 1050-4729.
- Emter, T. & Petereit, J. (2014). Integrated multi-sensor fusion for mapping and localization in outdoor environments for mobile robots. *Proceedings of the International Society for Optical Engineering*, 9121.
- Fang, Z. & Zhang, L. (2013). A Multi-Objective Strategy based on Frontier-based Approach and Fisher Information Matrix for Autonomous Exploration. *IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems*, pp. 96--101.
- Freda, L. & Oriolo, G. (2005). Frontier-Based Probabilistic Strategies for Sensor-Based Exploration. *IEEE International Conference on Robotics and Automation*, pp. 3881-3887.
- Früh, C. & Zakhor, A. (2003). Constructing 3D City Models by Merging Aerial and Ground Views. *Computer Graphics and Applications*, 23(6):52--61.
- Gerkey, B. P.; Vaughan, R. T. & Howard, A. (2003). The player/stage project: Tools for multi-robot and distributed sensor systems. Em *In Proceedings of the 11th International Conference on Advanced Robotics*, pp. 317--323.

- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edição. ISBN 0201157675.
- González-Baños, H. H. & Latombe, J. (2002). Navigation strategies for exploring indoor environments. *I. J. Robotic Res.*, 21(10-11):829--848.
- Gupta, S.; Hare, J. & Zhou, S. (2012). Cooperative coverage using autonomous underwater vehicles in unknown environments. Em *2012 Oceans*, pp. 1–5. ISSN 0197-7385.
- Holz, D.; Basilico, N.; Amigoni, F. & Behnke, S. (2010). Evaluating the Efficiency of Frontier-based Exploration Strategies. *41st International Symposium on Robotics and 6th German Conference on Robotics (ROBOTIK)*, pp. 1--8.
- Hough, P. V. C. (1962). Method and means for recognizing complex patterns. US Patent 3,069,654.
- Howard, A. & Roy, N. (2003). The Robotics Data Set Repository (Radish).
- Huang, Y. & Gupta, K. (2008). RRT-SLAM for motion planning with motion and map uncertainty for robot exploration. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1077--1082.
- Husain, A.; Jones, H.; Kannan, B.; Wong, U.; Pimentel, T.; Tang, S.; Daftry, S.; Huber, S. & Whittaker, W. L. (2013). Mapping planetary caves with an autonomous, heterogeneous robot team. Em *Aerospace Conference, 2013 IEEE*, pp. 1–13. ISSN 1095-323X.
- Islam, F.; Nasir, J.; Malik, U.; Ayaz, Y. & Hasan, O. (2012). Rrt star;-smart: Rapid convergence implementation of rrt star; towards optimal solution. Em *2012 IEEE International Conference on Mechatronics and Automation*, pp. 1651–1656. ISSN 2152-7431.
- Jadidi, M. G.; Miro, J. V. & Dissanayake, G. (2015). Mutual information-based exploration on continuous occupancy maps. Em *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 6086–6092.
- Jebari, I.; Bazeille, S. & Filliat, D. (2011). Combined Vision and FrontierBased Exploration Strategies for Semantic Mapping. *International Asia Conference on Informatics in Control, Automation and Robotics*, pp. 237--244.

- Jorge, V. A. M.; Maffei, R.; Franco, G. S.; Daltrozo, J.; Giambastiani, M.; Kolberg, M. & Prestes, E. (2015). Ouroboros: Using potential field in unexplored regions to close loops. Em *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2125–2131. ISSN 1050-4729.
- Juliá, M.; Gil, A. & Reinoso, O. (2012). A comparison of path planning strategies for autonomous exploration and mapping of unknown environments. *Autonomous Robots*, 33(4):427–444. ISSN 1573-7527.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45.
- Karaman, S. & Frazzoli, E. (2011). Sampling-based algorithms for optimal motion planning. *Int. J. Rob. Res.*, 30(7):846–894. ISSN 0278-3649.
- Kasperski, S.; Wollenberg, J. & Eich, M. (2013). Evaluation of cooperative exploration strategies using full system simulation. Em *Advanced Robotics (ICAR), 2013 16th International Conference on*, pp. 1–6.
- Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. *Int. J. Rob. Res.*, 5(1):90–98. ISSN 0278-3649.
- Korrapati, H. & Mezouar, Y. (2014). Vision-based sparse topological mapping. *Robotics and Autonomous Systems*, 62(9):1259 – 1270. ISSN 0921-8890. Intelligent Autonomous Systems.
- Lavalle, S. M. (1998). Rapidly-exploring random trees: A new tool for path planning. Relatório técnico.
- Lee, K. S.; Ovinis, M.; Nagarajan, T.; Seulin, R. & Morel, O. (2015). Autonomous patrol and surveillance system using unmanned aerial vehicles. Em *Environment and Electrical Engineering (EEEIC), 2015 IEEE 15th International Conference on*, pp. 1291–1297.
- Li, Y.; Wang, J.; Rizos, C.; Mumford, P. & Ding, W. (2006). Low-cost Tightly Coupled GPS/INS Integration Based on a Nonlinear Kalman Filtering Design. *National Technical Meeting of the Institute of Navigation*, pp. 958–966.
- Li, Y.; Yuan, P.; Han, W.; Lai, T.; Chen, D.; Tan, H.; Shi, Z. & Wang, T. (2015). A mapping method based on EKF and point-segment matching for indoor environment. *IEEE 10th Conference on Industrial Electronics and Applications*, pp. 945–950.

- Lu, F. & Milios, E. (1997). Robot pose estimation in unknown environments by matching 2D range scans. , *Journal of Intelligent and Robotic Systems*, 18(3):249–275.
- Melo, R. S.; Macharet, D. G. & Campos, M. F. M. (2015). Multi-object transportation using a mobile robot. Em *2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR)*, pp. 234–239.
- Moravec, H. & Elfes, A. (1985). High resolution maps from wide angle sonar. *International Conference on Robotics and Automation*, pp. 116–121.
- Nagatani, K.; Otake, K. & Yoshida, K. (2012). Three-dimensional thermography mapping for mobile rescue robots. Em Yoshida, K. & Tadokoro, S., editores, *FSR*, volume 92 of *Springer Tracts in Advanced Robotics*, pp. 49–63. Springer.
- Newman, P.; Bosse, M. & Leonard, J. (2003). Autonomous feature-based exploration. *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1234–1240.
- Oh, T. J. & Chung, M. J. (2012). Global Pose Estimation with Adaptive GPS/IMU Fusio. *Proceedings of the 12th International Conference IAS-12*, 1:26–29.
- Oßwald, S.; Bennewitz, M.; Burgard, W. & Stachniss, C. (2016). Speeding-up robot exploration by exploiting background information. *IEEE Robotics and Automation Letters*, 1(2):716–723. ISSN 2377-3766.
- Paz, L. M.; Jensfelt, P.; Tardos, J. D. & Neira, J. (2007). EKF slam updates in $O(n)$ with divide and conquer slam. Em *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 1657–1663. ISSN 1050-4729.
- Rasche, C.; Stern, C.; Kleinjohann, L. & Kleinjohann, B. (2011). A distributed multi-uav path planning approach for 3d environments. Em *Automation, Robotics and Applications (ICARA), 2011 5th International Conference on*, pp. 7–12.
- Schmid, K.; Tomic, T.; Ruess, F.; Hirschmuller, H. & Suppa, M. (2013). Stereo Vision based indoor/outdoor Navigation for Flying Robots. *Intelligent Robots and Systems (IROS)*, pp. 3955–3962.
- Schmidt, D.; Luksch, T.; Wettach, J. & Berns, K. (2006). Autonomous behavior-based exploration of office environments. *Proceedings of the International Conference on Informatics in Control, Automation and Robotics*, pp. 235–240.

- Shannon, C. E. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27(3):379–423.
- Siegwart, R. & Nourbakhsh, I. R. (2004). *Introduction to Autonomous Mobile Robots*. Bradford Company, Scituate, MA, USA. ISBN 026219502X.
- Sim, R. & Dudek, G. (2003). Effective exploration strategies for the construction of visual maps. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3224–3231.
- Sim, R.; Dudek, G. & Roy, d. N. (2004). Online control policy optimization for minimizing map uncertainty during exploration. *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1758–1762.
- Simhon, S. & Dudek, G. (1998). A global topological map formed by local metric maps. Em *Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on*, volume 3, pp. 1708–1714 vol.3.
- Souto, D.; Faiña, A.; Peña, F. L. & Duro, R. J. (2015). Morphologically intelligent underactuated robot for underwater hull cleaning. Em *Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2015 IEEE 8th International Conference on*, volume 2, pp. 879–886.
- Stachniss, C. & Burgard, W. (2003). Exploring Unknown Environments with Mobile Robots using Coverage Maps. *International Conference on Artificial Intelligence*, pp. 1127–1132.
- Stephen, J. (2000). *Development of a multisensor GNSS based vehicle navigation system*. Tese de doutorado, University of Calgary.
- Stephen, J. & Lachapelle, G. (2001). Development and testing of a GPS-augmented multisensor vehicle navigation system. *J. of Navigation*, 54(2):297–319.
- Ström, D. P.; Nenci, F. & Stachniss, C. (2015). Predictive exploration considering previously mapped environments. Em *IEEE International Conference on Robotics and Automation*, pp. 2761–2766. ISSN 1050-4729.
- Vallve, J. & Andrade-Cetto, J. (2015). Active pose SLAM with rrt*. Em *IEEE International Conference on Robotics and Automation, ICRA 2015, Seattle, WA, USA, 26-30 May, 2015*, pp. 2167–2173.

- Vishal, K.; Jawahar, C. V. & Chari, V. (2015). Accurate Localization by Fusing Images and GPS Signals. *CVPR Workshops*, pp. 17--24.
- Visser, A. & Slamet, B. A. (2008). Including communication success in the estimation of information gain for multi-robot exploration. *International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, pp. 680--687.
- Wang, Q.; Wang, W. & Li, Y. (2012). A multi-rrt based hierarchical path planning method. Em *Communication Technology (ICCT), 2012 IEEE 14th International Conference on*, pp. 971--975.
- Wen, C.; Qin, L.; Zhu, Q.; Wang, C. & Jonathan Li, J. (2014). Three-Dimensional Indoor Mobile Mapping With Fusion of Two-Dimensional Laser Scanner and RGB-D Camera Data. *Geoscience and Remote Sensing Letters*, 11(4):843--847.
- Wurm, K. M.; Stachniss, C. & Burgard, W. (2008). Coordinated multi-robot exploration using a segmentation of the environment. Em *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1160--1165. ISSN 2153-0858.
- Yamauchi, B. (1997). A frontier-based approach for autonomous exploration. *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pp. 146--151.
- Yang, K.; Gan, S. K. & Sukkariéh, S. (2013). A gaussian process-based rrt planner for the exploration of an unknown and cluttered environment with a uav. *Advanced Robotics*, 27(6):431--443.
- Zhang, P.; Gu, J.; Milios, E. E. & Huynh, P. (2005). Navigation with IMU/GPS/Digital Compass with Unscented Kalman Filter. *Proceedings of IEEE International Conference on Mechatronics & Automation*, 3:1497--1502.