

Universidade Federal do Rio Grande
Modelagem Computacional

**Uma Abordagem Topológica Híbrida
Baseada em Mapas Espacial e
Perceptivo para Robôs**

Celina Häffele da Rocha

Rio Grande, 27 de Setembro de 2011

Universidade Federal do Rio Grande
Modelagem Computacional

Uma Abordagem Topológica Híbrida Baseada em Mapas Espacial e Perceptivo para Robôs

Celina Häffele da Rocha

Dissertação apresentada ao Programa de Pós-Graduação em Modelagem Computacional, como requisito parcial à obtenção do título de Mestre em Modelagem Computacional.

Orientador(a): Profa. Dra. Silvia Silva da Costa Botelho

Co-orientador(a): Prof. Msc. Paulo Lilles Jorge Drews Junior

Rio Grande, 27 de Setembro de 2011

Este trabalho foi analisado e julgado adequado para a obtenção do título de Mestre em Modelagem Computacional e aprovado em sua forma final pelo orientador.

Profa. Dra. Silvia Silva da Costa Botelho

Banca Examinadora:

Profa. Dra. Silvia Silva da Costa Botelho

Centro de Ciências Computacionais – FURG (Orientadora)

Prof. Msc. Paulo Lilles Jorge Drews Junior

Centro de Ciências Computacionais – FURG (Co-orientador)

Prof. Dr. Hugo Vieira Neto

Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial
– UTFPR

Prof. Dr. Adriano Velasque Werhli

Centro de Ciências Computacionais – FURG

Para meus pais,
que estiveram sempre presentes e em agradecimento pela vida
e pelos ensinamentos para vivê-la com dignidade e
honestidade.

AGRADECIMENTOS

À minha orientadora, Prof^a Dr^a Silvia Silva da Costa Botelho, pela paciência, pelo comprometimento, seriedade, pela competente orientação e pelo conhecimento demonstrado durante a elaboração desse trabalho.

Ao meu co-orientador, Prof^o Msc. Paulo Lilles Jorge Drews Junior, pelo profissionalismo e encorajamento, pela disponibilidade dispensada em todas as situações e pelas suas sugestões que foram preciosas na construção deste trabalho.

A todos os Professores, pelo convívio e pelas aprendizagens durante as aulas, nos atendimentos e nos corredores.

Ao programa de Pós-Graduação em Modelagem Computacional, por todo o suporte.

À CAPES, pelo apoio financeiro.

À amiga Gisele Simas, pelo coleguismo no laboratório que se transformou em amizade - um constante apoio durante a caminhada.

À Mônica Figueiredo e Gabriel Oliveira, pela parceira no projeto, sempre incansáveis.

A todos os colegas do NAUTEC, pelos momentos de construção de conhecimento, pela preciosa vivência diária.

À minha irmã e sua família, por todo amor e apoio em todos os momentos.

Aos meus avós, por todas as orações.

À Dilma Machado e Francisco Machado pelo apoio, carinho e orações.

A Gabriel Soares Machado, por ser meu porto seguro, pelo amor incondicional que por ele tenho, pela honestidade e bondade infinita que existe em seu coração, por ser mais do que amigo e companheiro.

À minha mãe, pelo apoio, amor, cuidado, disponibilidade, por abrir mão de parte de sua vida para que eu pudesse construir a minha. Obrigada mamãe.

Ao meu pai, homem incomparável, incansável, exemplo de honestidade, amor e dedicação, alguém que tenho orgulho de ser filha.

À Deus (A e Ω) por estar sempre ao meu lado.

*”Eu pedi forças...
E Deus me deu dificuldades para me fazer forte.
Eu pedi sabedoria...
E Deus me deu problemas para resolver.
Eu pedi prosperidade...
E Deus me deu cérebro e músculos para trabalhar.
Eu pedi coragem...
E Deus me deu perigos para superar.
Eu pedi amor...
E Deus me deu pessoas com problemas para ajudar.
Eu pedi favores...
E Deus me deu oportunidades.
Não recebi exatamente o que pedi...
Mas recebi tudo aquilo que mais precisava...”*

(Maria Júlia Paes da Silva)

RESUMO

Neste trabalho será descrito um estudo sobre a aplicação de mapas topológicos, mais precisamente mapas auto-organizáveis, e seu uso em diferentes níveis semânticos para o tratamento de informações sensoriais. É proposto um método para mapeamento robótico utilizando mapas auto-organizáveis dispostos em níveis, que sejam capazes de amenizar os problemas de escalabilidade, correspondência, presença de ruídos, dinamicidade do ambiente e estratégias de exploração, possibilitado o uso de diferentes tipos de informações sensoriais. O sistema foi implementado em C++ e validado em simulação, na qual o robô é dotado com uma câmera omnidirecional e um GPS. O algoritmo proposto mostrou-se eficiente, conseguindo compactar informação e reconhecendo dados semelhantes.

Palavras-chaves: Mapeamento Robótico, Redes Neurais Artificiais, Mapas Topológicos, Redes de Kohonen, Estruturas Celulares Crescentes.

ABSTRACT

This work describes an application study about topological maps to deal with sensorial information using different semantic levels. It is proposed a method to robotic mapping using self-organizing maps disposed in levels. It is capable to deal with the scalability problem, correspondence matching, noise sensory data, environment dynamicity and different exploration strategies. It allows using different sensory sources. The system is implemented using C++ language and it is validated using robotic simulation. The robot is simulation has an unidirectional camera and GPS. The results show the capacities to information compression and similar data recognition.

Keywords: Robotic Mapping, Artificial Neural Networks, Topological Maps, Kohonen Networks, Growing Cell Structures.

Lista de Figuras

2.1	Abordagem para Mapeamento.	10
2.2	Exemplo de mapa topológico proposto em [Kuipers and Byun, 1988].	15
3.1	Estrutura celular com diferentes dimensionalidade de k	22
3.2	Remoção de neurônio em uma rede bidimensional.	24
4.1	Arquitetura Proposta.	37
4.2	Redução Dimensional da Extração de Características.	38
4.3	Processo de Extração de Características [Hafner, 2000].	40
5.1	Ambiente de simulação Webots	44
5.2	Robô do Simulador Webots	45
5.3	Imagem capturada pela câmera omnidirecional	45
5.4	Imagem retificada	46
5.5	Cenário Base.	46
5.6	Cenário Correspondência	47
5.7	Cenário Escalabilidade	47
5.8	Cenário Dinâmico - ocorreu a remoção de alguns obstáculos, o ambiente completo é semelhante ao cenário base, conforme Figura 5.5.	48
5.9	Trajeto percorrido pelo Robô	49
5.10	Mapa do Cenário Base utilizando arestas para a integração entre os mapas.	55
5.11	Mapa do Cenário Base utilizando cores, no qual os nodos do mapa espacial representam a cor do grupo do mapa perceptivo que melhor representa a informação encontrada na posição do robô.	56

LISTA DE FIGURAS

5.12	Mapa do Cenário Base utilizando cores, no qual os nodos do mapa espacial representam a(s) cor(es) do grupo(s) do mapa perceptivo que representa a informação encontrada na posição do robô.	57
5.13	Mapa do Cenário de Correspondência	58
5.14	Trajeto percorrido pelo robô.	59
5.15	Trajeto percorrido pelo robô.	59
5.16	Mapa do Cenário de Escalabilidade	60
5.17	Mapa do Cenário de Escalabilidade	61
5.18	Mapa do Cenário Dinâmico Completo	62
5.19	Mapa do Cenário Dinâmico com Retirada dos Obstáculos	63

Lista de Tabelas

2.1	Comparação do Mapeamento Métrico com o Mapeamento Topológico [Thrun, 1998]	14
3.1	Comparação das diferentes abordagens de redes neurais auto-organizáveis .	33
5.1	Resultados de Rede 10x10 para Mapa Perceptivo da Navegação no Cenário Base	51
5.2	Resultados de Rede 50x50 para Mapa Perceptivo da Navegação no Cenário Base	52
5.3	Resultados de Rede 100x100 para Mapa Perceptivo da Navegação no Cenário Base	53
5.4	Estudo dos Parâmetros do Mapa Espacial no Cenário Base	54
5.5	Desempenho do Sistema para o Cenário Base	55
5.6	Desempenho do Sistema para o Cenário Correspondência	58
5.7	Desempenho do Sistema para o Cenário Escalabilidade	60
5.8	Desempenho do Sistema para o Cenário Dinâmico	62

LISTA DE SIGLAS

2D (*Bidimensional*)

3D (*Tridimensional*)

BMU (*Best Matching Unit*)

EQ (*Erro de Quantização*)

ET (*Erro Topográfico*)

FURG (*Universidade Federal do Rio Grande*)

NAUTEC (*Núcleo de Automação e Tecnologia Computacional*)

GCS (*Growing Cell Structures*)

GNG (*Growing Neural Gas*)

GPS (*Global Positioning System*)

GWR (*Grow When Required*)

OpenCV (*Intel Open Source Computer Vision Library*)

OpenGL (*Open Graphics Library*)

SLAM (*Simultaneous Localization and Mapping*)

SOM (*Self Organizing-maps*)

Conteúdo

Resumo

Abstract

Lista de ilustrações

Lista de tabelas

Lista de Siglas

1	Introdução	1
1.1	Contexto e Motivação	1
1.2	Objetivos	3
1.3	Contribuição Relevante	4
1.4	Estrutura do Texto	5
2	Mapeamento de Ambientes	6
2.1	A Síntese de Ambientes	8
2.2	Mapeamento Métrico	11
2.3	Mapeamento Topológico	12
2.4	Mapeamento Híbrido	14
2.4.1	Trabalhos Correlatos	14
2.5	Considerações do Capítulo	17
3	Redes Neurais Auto-Organizáveis	19
3.1	Redes de Kohonen	20

3.2	Estruturas Celulares Crescentes - GCS (<i>Growing Cell Structures</i>)	21
3.3	Rede Neural Gas - GNG (<i>Growing Neural Gas</i>)	25
3.4	Cresce Quando Necessário - GWR (<i>Grow When Required</i>)	28
3.5	Considerações do Capítulo	32
4	Metodologia	34
4.1	Requisitos do Projeto	34
4.2	Visão Geral da Arquitetura	35
4.3	Implementação da Arquitetura	37
4.3.1	Implementação dos Mapas Perceptivos	38
4.3.2	Implementação do Mapa Espacial	40
4.3.3	Integração do Mapa Espacial com os Mapas Perceptivos	41
4.4	Considerações do Capítulo	41
5	Resultados Experimentais	43
5.1	Implementação do Sistema	43
5.1.1	Plataforma de Experimentação	44
5.2	Cenário de Testes	46
5.3	Cenário Base	48
5.3.1	Medidas de Avaliação	48
5.3.2	Parâmetros de Inicialização	50
5.3.3	Utilização dos Mapas Perceptivos e Espacial	55
5.4	Cenário de Correspondência	58
5.5	Cenário de Escalabilidade	59
5.6	Cenário de Ambientes Dinâmicos	61
5.7	Considerações Finais	63
6	Conclusão e Trabalhos Futuros	65
	Referências Bibliográficas	75

Capítulo 1

Introdução

Este capítulo descreve a contextualização e as motivações deste trabalho. Em seguida, são definidos os objetivos da proposta, bem como a sua contribuição científica. Por fim, a estrutura do texto, na qual apresenta-se como estão dispostos os capítulos que compõem essa dissertação.

1.1 Contexto e Motivação

Além de ser uma área consolidada de pesquisa, a utilização de sistemas robóticos no cotidiano cresce a cada dia. Muitos são ainda os desafios algorítmicos e tecnológicos associados ao desenvolvimento de robôs com autonomia para a realização de suas atividades principais: a percepção do ambiente, a tomada de decisão e a atuação. Uma classe destes desafios está ligada à questão da navegação destes em ambientes reais visando a execução de diferentes tipos de tarefas. Durante a navegação, os robôs devem ser capazes de tratar uma ampla variedade de informações incertas e dinâmicas, interagir inteligentemente com humanos e outros sistemas robóticos, e garantir sua própria segurança, assim como das pessoas ao seu redor [Bensalem et al., 2009].

Para tal, sistemas de navegação autônoma devem ser desenvolvidos de forma a guiar um robô sem nenhuma interferência externa, bem como definir ações para o mesmo, tais como atingir alvos, desviar de obstáculos e explorar o ambiente [Calvo and Figueiredo, 2003], [Cazangi and Figueiredo, 2002], [Crestani et al., 2002], [Borenstein et al., 1996]. Neste contexto, a pesquisa na área de navegação autônoma de

robôs móveis envolve basicamente três desafios fundamentais, sendo eles: a localização, o mapeamento de ambientes, e o planejamento de trajetórias.

A localização de robôs móveis consiste na tarefa de estimar a posição de um robô, por meio de informações obtidas de seus sensores, em um mapa do ambiente no qual o robô está inserido. O mapeamento tem o objetivo de criar, também através dos dados adquiridos de seus sensores, uma representação do ambiente navegado pelo robô. Por fim, o planejamento de trajetória é responsável por enviar comandos motores ao robô fazendo com que ele navegue de forma segura, desviando de obstáculos, até um local determinado [Oliveira, 2010].

O mapeamento robótico tem sido uma área bastante explorada em robótica e inteligência artificial há pelo menos duas décadas [Thrun, 2002]. Em suma, pode-se defini-lo como a obtenção de um modelo representativo do ambiente percorrido pelo robô. A obtenção de tal modelo dependerá das características do ambiente e dos sensores utilizados para a sua percepção. Segundo Thrun [Thrun, 2002], existem vários aspectos que tornam esse problema complexo, dentre eles pode-se citar:

- erros nas medições dos sensores;
- alta dimensionalidade das entidades que estão sendo mapeadas;
- problema da associação de dados (correspondência);
- ambientes que mudam ao longo do tempo; e
- dependência de uma estratégia de exploração.

Na literatura, diferentes abordagens de mapeamento são propostas tratando as questões acima. No que tange modelos de representação do ambiente, abordagens geométricas são propostas visando a representação exata do espaço de configuração. É crescente o uso de técnicas probabilísticas de SLAM (Simultaneous Localization and Mapping) as quais associam o problema de mapeamento ao de localização visando minimizar os efeitos cumulativos de imprecisão de localização na obtenção do mapa, e vice-versa [Thrun et al., 2005], [Durrant-Whyte and Bailey, 2006a], [Durrant-Whyte and Bailey, 2006b], [Castellanos et al., 2007], [Grisetti et al., 2007], [Kim et al., 2008], [Blanco et al., 2008], [Blanco et al., 2009], [Tovar et al., 2006].

Métodos bio-inspirados buscam desenvolver modelos de representação que permitam fornecer subsídios à exploração e ao registro do ambiente navegável sem a necessidade de descrições exatas de seus componentes [Hafner, 2000], [Hafner, 2008],[Menegaz, 2009], [Oliveira, 2010], [Zimmer and von Puttkamer, 1994], [Zimmer, 1996]. Para tal, pode-se citar o uso de Mapas Topológicos baseados em Redes Auto-Organizáveis (Self-organized Maps - SOM). Tais mapas permitem a representação de grupos de informações de dimensão m em representações topológicas de menor dimensão, n . No caso de $n = 2$ grades bidimensionais congregam topologicamente as percepções advindas da exploração do robô, fornecendo mapas representativos do ambiente navegável.

A capacidade de adaptabilidade, tolerância a ruídos e generalização, próprias de sistemas inteligentes, das SOMs são características relevantes para a sua utilização. A complexidade decorrente de valores de m muito grandes, devido a escalabilidade e diversidade dos multi-sensores (característica intrínseca das tecnologias para percepção) e a dificuldade em relacionar os mapas obtidos com informações espaciais são aspectos a serem tratados em tais abordagens. Trabalhos recentes exploram diferentes níveis semânticos de representação [Hafner, 2008], dividindo a informação a ser mapeada em diferentes níveis de abstração, indo de informações geométricas espaciais de baixo nível a informações cognitivas de alto nível dependentes do contexto [Hafner, 2008], [Hafner, 2000].

Espera-se que as características inteligentes das SOMs, associadas às recentes organizações das informações sensoriais em diferentes níveis hierárquicos, permitam avanços na obtenção de sistemas com maior capacidade de tratamento dos problemas de escalabilidade, correspondência, dinamicidade, presença de ruídos e estratégia de exploração referentes ao mapeamento.

1.2 Objetivos

O principal objetivo deste trabalho é o estudo de Mapas Topológicos, mais precisamente Mapas Auto-organizáveis, e seu uso em diferentes níveis semânticos para o tratamento das informações sensoriais, visando a obtenção de modelos representativos de ambientes navegáveis por robôs móveis dotados de diferentes sensores. O estudo objetiva fornecer uma arquitetura para mapeamento robótico baseada em diferentes níveis de Mapas Auto-

organizáveis. De forma mais precisa, focaliza-se nas seguintes questões:

- Realizar estudo bibliográfico sobre algumas técnicas de Mapeamento Robótico, entendendo seus principais desafios e limitações;
- Realizar revisão bibliográfica envolvendo os principais tipos de Mapas Auto-organizáveis, verificando suas características;
- Especificar e projetar SOMs capazes de representar diferentes categorias de informações sensoriais (peceptivas) e de localização (espaciais), tratando os diferentes desafios próprios do problema de mapeamento;
- Propor uma arquitetura capaz de integrar os diferentes níveis semânticos associados à percepção advinda de multi-sensores presentes em um robô móvel;
- Estudar e desenvolver uma interface de simulação capaz de permitir a realização de testes com robôs dotados de diferentes sensores (GPS, câmera) em ambientes 3D, utilizando o simulador Webots (<http://www.cyberbotics.com/>).
- Realizar testes e análises da proposta de modo a verificar seu desempenho e limitações frente aos desafios existentes.

1.3 Contribuição Relevante

A principal contribuição dessa pesquisa é propor um método para mapeamento robótico utilizando mapas auto-organizáveis dispostos em níveis semânticos, espacial e perceptivos, que seja capaz de tratar, ao menos parcialmente, os problemas de escalabilidade, correspondência, presença de ruídos, dinamicidade do ambiente e estratégias de exploração, possibilitado o uso de diferentes tipos de informações sensoriais em um arcabouço extensível, compacto e robusto.

Além dessas, tem-se o desenvolvimento do arcabouço de software, com uso de dados do simulador robótico Webots, permitindo a simulação física tridimensional de diferentes tipos de robôs dotados de sensores de tipos variados.

1.4 Estrutura do Texto

Neste capítulo foi apresentada uma introdução sobre o mapeamento robótico justificando assim, o problema tratado neste trabalho. No capítulo seguinte é realizada uma revisão dos trabalhos encontrados na literatura sobre mapeamento robótico.

No Capítulo 3 são apresentados os principais conceitos e técnicas associadas aos Mapas auto-organizáveis, com suas respectivas vantagens e desvantagens para a aplicação ao problema do mapeamento robótico.

No Capítulo 4 é descrita, em detalhes, a metodologia proposta para mapeamento topológico utilizando múltiplas camadas de SOM.

No Capítulo 5 são descritas as instâncias experimentais utilizadas e discutidos os resultados alcançados, onde a metodologia é validada. Utilizando dados de robôs simulados, os resultados mostram a capacidade da metodologia em resolver o problema proposto, seu desempenho e restrições.

O Capítulo 6 apresenta as conclusões finais da proposta. Algumas possibilidades futuras também são apresentadas, que permitirão diversas extensões e melhorias ao presente trabalho.

Capítulo 2

Mapeamento de Ambientes

Robôs móveis têm como característica intrínseca a possibilidade de navegar no ambiente em que estão imersos. Para tal, a habilidade de estimar sua localização, mapear o ambiente navegável e planejar a sua trajetória são pré-condições básicas para a concepção de veículos móveis autônomos.

O processo de estimar a posição de um robô dentro do ambiente é denominado de localização. A localização estabelece-se a partir de um mapa que modela o mundo ao seu redor e estime através deste modelo sua posição corrente e sua orientação. Tais mapas podem ser conhecidos *a priori* ou obtidos automaticamente pelo robô. A seguir, apresentam-se os principais paradigmas associados à localização de robôs móveis:

- Paradigmas Locais × Globais: Métodos locais rastreiam apenas a posição do robô, de forma a compensar os erros odométricos. Nos métodos globais não se faz necessário informar a posição inicial do robô, permitindo re-localização no caso de falha.
- Paradigmas Estáticos × Dinâmicos: Nos ambientes estáticos o robô é o único agente em movimento. Métodos que filtram objetos em movimento, excluindo-os do mapa, são desenvolvidos para o tratamento de ambientes dinâmicos.
- Paradigmas Passivos × Ativos: No paradigma passivo a localização baseia-se exclusivamente em informações sensoriais, sem controle da pose dos sensores de forma a melhorar o desempenho da localização. Já no paradigma ativo a função de estima-

tiva de posição tem o controle parcial, ou total do robô, melhorando a eficiência e robustez do método de localização.

- Único robô \times multirobôs: A um maior custo de comunicação, sistemas multi-robôs podem compartilhar crenças e ter um melhor desempenho associado à exploração do ambiente.
- Indoor \times Outdoor: Diferentes informações deverão ser manipuladas, dependendo da natureza do ambiente: indoor (interiores) ou outdoor (exteriores). Em geral, estes últimos não são conhecidos a priori.

Métodos probabilísticos vem sendo utilizados para o tratamento das questões envolvidas na Tabela 2.4, representando as incertezas através de probabilidades. Destes pode-se citar:

- Filtros de Kalman: rastreiam a localização de marcas e de características enquanto o robô se move no ambiente. É um dos poucos métodos que computacionalmente podem manter uma completa representação da incerteza em todo o mapa, restringindo-se uma representação gaussiana da incerteza (tratada pelas abordagens estendidas).
- Métodos de Expectativa de Maximização: complementa ao Filtro de Kalman, sendo capaz de resolver problemas de correspondência em ambientes cíclicos. O método encontra o mapa que maximiza as medidas sensoriais e o caminho estimado pelo robô. Não permite o registro do mapa completo, sendo computacionalmente complexo.
- Filtros de Partículas: o robô representa a crença sobre sua pose, usando um conjunto de amostras, denominadas partículas. Na localização robótica, geralmente são referenciados como Localização de Monte Carlo. Uma vez ponderadas, as partículas representam a distribuição estimada da pose do robô.

Recentemente o problema de localização vem sendo tratado em conjunto com as questões de mapeamento, recebendo a denominação de Mapeamento e Localização Simultâneos (SLAM - *Simultaneous Localization and Mapping*). Este pode ser definido como “construir alguma forma de representação espacial do mundo por meio de uma

sequência de dados capturados pelo robô, assumindo ainda que não haja nenhuma informação a priori do ambiente, e simultaneamente localizar o robô usando essa representação”. Ao mapear o ambiente o robô deverá realizar duas tarefas: sintetização do mapa e sua exploração.

A síntese de mapa é passiva, caracterizando-se pela construção de um modelo que represente o ambiente, a partir das observações percebidas. A exploração é um processo ativo, no qual o robô é controlado de forma a mapear todo o ambiente. Esta dissertação focaliza-se na etapa de síntese, descrita em detalhes a seguir.

2.1 A Síntese de Ambientes

Como mencionado no Capítulo 1, o processo de sintetização do mapa pode ser resumido na aquisição de um modelo do ambiente percorrido pelo robô, criado por meio de informações sensoriais, que representam a lista destas impressões e suas localizações no ambiente. O mapa m pode ser descrito formalmente através de suas N impressões, conforme Equação 2.1:

$$M = \{m_1, m_2, \dots, m_n, \dots, m_N\}, \quad (2.1)$$

onde cada m_i , com $1 < n < N$, está associada a uma descrição de cada impressão sensorial específica - característica. Mapas são, normalmente, baseados em características e baseados em localização. Em Mapas de Características i é o índice da característica mapeada. O valor de m_i contém as características e a localização. Suas representações especificam apenas a forma do ambiente, nos locais específicos, ou seja, a localização dos objetos contidos no mapa.

Em mapas baseados na localização, o índice i corresponde a localizações específicas. Em *mapas planares* é comum a indicação de um elemento no mapa por $m_{x,y}$, ao invés de m_i , tornando explícito que $m_{x,y}$ é uma coordenada no mundo, identificando qualquer local no ambiente, não contendo apenas informações sobre objetos no ambiente, mas também sobre a ausência deles (por exemplo, o espaço livre).

Busca-se encontrar o conjunto M que melhor represente o ambiente navegado. Nas últimas três décadas, muitas têm sido as abordagens para tal. Atualmente, encontra-se

na bibliografia métodos robustos para o mapeamento de ambientes estáticos, estruturados e de tamanho limitado. No entanto, o mapeamento de ambientes não estruturados, dinâmicos e de grande escalabilidade é um problema em aberto, englobando as seguintes questões [Thrun, 2002]:

- **A alta dimensionalidade das entidades a serem mapeadas** - para entender esse problema, pode-se considerar o seguinte exemplo, supondo que os diferentes cômodos de uma casa são representados por cotas associadas à sua largura e comprimento, quantas cotas são necessárias para construir um mapa de uma casa, limitando-se a descrição das principais entidades topológicas, tais como: corredores, salas, quartos, portas. Para construir esse mapa, por exemplo poderia ser suficiente algumas dezenas de cotas. No caso de um edifício, utilizando-se o mesmo elemento descritivo (cotas) representado em um mapa bidimensional da sua planta baixa, o que é bastante utilizado também em robótica, geralmente se faz necessário algo em torno de milhares de cotas. Além disso, se desejarmos um mapa visual detalhado 3D deste prédio (ou por exemplo do fundo do oceano) facilmente podem se fazer necessários milhões de elementos descritivos. Do ponto de vista estatístico, cada cota corresponde a uma dimensão. Assim, o problema de mapeamento pode ter uma alta dimensionalidade [Thrun, 2002].
- **O problema de correspondência** - também é conhecido como o problema de associação de dados [Thrun, 2002]. O problema de correspondência está em determinar se as impressões sensoriais tomadas em diferentes instantes de tempo são iguais ou distintas, ou seja, o problema está relacionado em reconhecer percepções como semelhantes ou diferentes.
- **Mudança do ambiente ao longo do tempo** - esse problema está relacionado com ambientes dinâmicos. A dinamicidade dos ambientes robóticos cria um relevante desafio, já que a mudança do ambiente, por exemplo, o estado de uma porta (aberta ou fechada), é de difícil percepção a partir das medidas dos sensores. Tal exemplo pode ser explicado por duas hipóteses, ou o estado da porta mudou, ou o robô não está no lugar em que ele acredita estar [Thrun, 2002] [Vasudevan et al., 2007].

- **Natureza das medidas dos sensores** os sensores não são ideais, existindo incerteza nos valores de suas medições. Tal fato se agrava devido ao erro na odometria do veículo acumular-se ao longo do tempo, afetando a maneira como, no futuro, as medições serão interpretadas e dispostas sobre o mapa.
- **Estratégias de Exploração Adaptativas** considerando que um ambiente desconhecido e dinâmico deva ser mapeado pelo robô, não é uma tarefa simples o planejamentos de rotas que garantam, frente a recursos limitados, a obtenção de mapas completos e consistentes deste.

No que tange os aspectos específicos de estrutura de representação do ambiente, mais fortemente abordados neste trabalho, pode-se categorizar as diferentes propostas em três principais grupos: abordagens métricas, topológicas ou híbridas (derivada da abordagem métrica e topológica) [Choset et al., 2005] [Pedrosa, 2006] [Souza et al., 2008] ¹, a Figura 2.1 ilustra a divisão dessas abordagens, apresentadas a seguir.

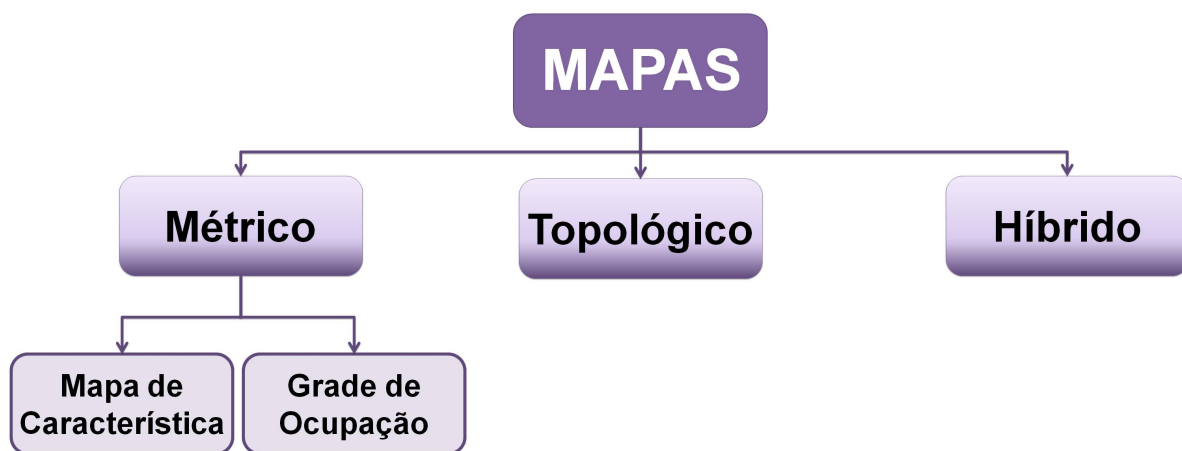


Figura 2.1: Abordagem para Mapeamento.

¹Outra classificação também presente na bibliografia considera as abordagens classificadas conforme o referencial escolhido para a construção do mapa. Nesta classificação os mapas podem ser centrados no robô ou centrados no ambiente, maiores detalhes podem ser vistos em [Thrun, 2002]

2.2 Mapeamento Métrico

O mapeamento métrico produz uma representação geométrica do ambiente [Thrun, 2002]. Nessa abordagem o ambiente é representado de maneira detalhada incluindo os objetos que o compõem como paredes, obstáculos, passagens, entre outros [Pedrosa, 2006]. Os mapas métricos podem ser representados de duas maneiras [Pedrosa, 2006]:

- **Grade de ocupação** - esta abordagem foi proposta por Alberto Elfes e Hans Moravec, em 1985 [Moravec and Elfes, 1985] e posteriormente formalizada em [Elfes, 1989]. Esse tipo de mapeamento é definido como um campo aleatório multidimensional que contém estimativas estocásticas dos estados das células em um espaço reticulado [Elfes, 1989]. A grade de ocupação tem como objetivo fornecer mapas consistentes a partir de dados sensoriais, a partir de uma posição x conhecida do robô. A grade de ocupação é representada sob a forma de uma grade (grid) ou matriz, geralmente 2D ou 3D. Cada elemento da matriz ou do grid, também denominado de célula, representa uma porção fixa do espaço e armazena um valor $m_{x,y}$ que indica o estado desta célula: livre ou ocupada [Thrun, 1998]. Na proposta inicial, a construção do mapa foi realizada a partir de informações sensoriais obtidas de um sonar [Moravec and Elfes, 1985]. No entanto, outros tipos de sensores também podem ser utilizados, tais como: laser, infravermelhos, entre outros.

As células podem possuir valores no intervalo de 0 a 1, onde 0 significa que a célula está vazia e 1 que a célula está ocupada. Assim, conhecendo a priori a posição (x, y) e as medições z realizadas pelo robô, a grade de ocupação utiliza essas informações para calcular a probabilidade posterior de cada célula do mapa, a partir da Equação 2.2 [Elfes, 1987], [Thrun et al., 2005], onde M representa o mapa adquirido e $z_{1:t}$ é o conjunto das medições até o instante de tempo t .

$$p(m_{x,y}|z_{1:t}) = \frac{p(z_t|m_{x,y})p(m_{x,y}|z_{1:t-1})}{p(z_t|z_{1:t-1})} \quad (2.2)$$

- **Mapas de Características** - um dos primeiros a propor esse tipo de abordagem foi James Crowley em 1985 [Crowley, 1985]. Esse tipo de mapa armazena informações geométricas de formas encontradas no ambiente, como por exemplo, círculos, retân-

gulos, retas, dentre outras. Tais características são encontradas principalmente em ambientes internos e estruturados, por exemplo, casas, corredores, escritórios e laboratórios.

2.3 Mapeamento Topológico

Mapas topológicos representam ambientes como uma lista de lugares significativos que estão conectados por arestas [Thrun, 2002]. Esse tipo de mapa, também é denominado mapa qualitativo. Devido ao uso desta abordagem neste trabalho, a seguir detalha-se suas características e trabalhos relevantes.

Estrutura, Processo de Registro e Tolerância a Ruídos O Mapeamento Topológico faz uso, comumente, de grafos para a sua construção. Normalmente, esses grafos descrevem os espaços livres para a execução de tarefas, no qual os seus nodos correspondem a regiões que possuem informações sensoriais e suas arestas são as relações entre elas. A abordagem topológica consiste na simples memorização das informações sensoriais enquanto o robô navega pelo ambiente, de forma que seja possível identificar situações em que o robô já esteve, apenas comparando a informação sensorial corrente com as já registradas. As informações sensoriais são memorizadas ao longo da navegação, possibilitando a identificação de situações já conhecidas, apenas comparando as informações sensoriais registradas. Como não existe a necessidade de se extrair significado geométrico das impressões sensoriais, pode-se utilizar sensores bem menos precisos e com custos reduzidos, além de não necessitar de um poder computacional elevado. Obtendo-se ainda um mapa com uma boa confiabilidade, embora possam ocorrer inconsistências [Zimmer and von Puttkamer, 1994]. Esse tipo de mapeamento pode ser feito utilizando qualquer sensor influenciável por alguma característica relevante do ambiente, ou seja, características que mudam quando o robô muda de posição e permanecem constantes quando o robô está parado, fazendo com que seja possível a localização local e global do robô.

Capacidade de Representação e Correspondência de Medidas Claramente, a quantidade de informações sensoriais diferentes, considerando as suas sutis variações, en-

contradas em um ambiente é infinita e, portanto, impossível de ser completamente representada. É, então, necessário registrar somente um conjunto finito associado às mais relevantes. Tais situações, devem ser capazes de representar adequadamente, por similaridade, uma quantidade maior de informações sensoriais, de forma que haja uma correspondência de um para muitos entre as situações registradas e as adquiridas [Drews Jr, 2007].

Para tratar as questões referentes à correspondência, onde leituras sensoriais podem acontecer em localizações distintas, acrescenta-se, ao processo, informações relativas à estimativa da posição do robô dentro do ambiente. A fusão das informações sensoriais com estimativas de posição (sujeitas a erros cumulativos) permitem a localização do robô no cenário, uma vez que as impressões sensoriais ao apresentarem menores erros de posição conduzem a uma melhor estimativa de localização. Por exemplo, existem descritores visuais robustos que pouco se alteram ao longo da navegação, mesmo com variações de iluminação e ponto de vista, tornando possível a localização do robô mesmo após um alto acúmulo de erro dos sensores de posição.

A escolha correta das situações sensoriais a serem registradas é muito importante, para que possa-se assumir situações sensoriais semelhantes correspondem a posições semelhantes no ambiente, no qual o robô pode transitar. A semelhança entre as duas situações é representada por relações de vizinhança a serem também registradas. As relações de vizinhança, correspondendo a possíveis transições entre situações sensoriais, bem como as próprias circunstâncias registradas, constituem um grafo que mapeia o ambiente.

Escalabilidade do Sistema De forma a tratar a alta dimensionalidade associada às medidas dos sensores, as abordagens baseadas em Mapas Topológicos utilizam técnicas de redução dimensional. O uso de Mapas Auto-organizáveis é comum na literatura, uma vez que estes utilizam uma estrutura de células e arestas para o mapeamento de uma dimensão m para n , onde $n \ll m$, com um baixo custo computacional. A abordagem topológica apresenta a vantagem, em relação a métrica, de necessitar de um menor poder computacional para a construção do mapa, além dos requisitos de processamento da informação sensorial serem reduzidos e o treinamento de mapas auto-organizáveis é pouco custoso. Os algoritmos envolvidos no mapeamento topológico são também de mais simples implementação que os algoritmos utilizados no mapeamento métrico [Zimmer, 1996].

2.4 Mapeamento Híbrido

A partir das vantagens e desvantagens do mapeamento métrico e topológico, expostos na Tabela 2.1, surgiu o mapeamento híbrido, que visa utilizar as vantagens de cada abordagem citada anteriormente. Embora, ele apresente boas características, ainda assim tem um custo computacional mais elevado que o topológico em si, além de lidar não tão bem quanto os mapas topológicos com imprecisões sensoriais.

Tabela 2.1: Comparação do Mapeamento Métrico com o Mapeamento Topológico [Thrun, 1998]

Tipo de Mapeamento	Vantagens	Desvantagens
Métrico	<ul style="list-style-type: none"> ✓ Fácil de construir, representar e manter; ✓ Reconhecimento de lugares (baseado na geometria); ✓ Facilita a computação de caminhos curtos 	<ul style="list-style-type: none"> ✓ Planejamento ineficiente e alto custo computacional; ✓ Requer a determinação precisa da pose do robô; ✓ Linguagem simbólica pobre para a solução de problemas
Topológico	<ul style="list-style-type: none"> ✓ Permite planejamento eficiente e baixo custo computacional; ✓ Não requer a determinação precisa da localização do robô; ✓ Representação sub-simbólica conveniente para a solução de problemas 	<ul style="list-style-type: none"> ✓ Dificuldade de construir e manter para ambientes grandes; ✓ Dificuldades no reconhecimento de lugares; ✓ Pode produzir caminhos sub-ótimos.

Conceitualmente, um mapa híbrido é definido como uma estrutura $H = \langle M, C \rangle$, onde $M = \{M_1, M_2, \dots, M_n\}$ é um conjunto de mapas e $C = \{c_1, c_2, \dots, c_p\}$ é um conjunto de elos. Cada elo corresponde a um par $\langle m_i, m_j \rangle$, onde m_i é um objeto do mapa M_i e, de forma similar, m_j é um objeto de M_j , com $i \neq j$. Este elo trata da correspondência existente entre objetos distintos de ambientes distintos [Pedrosa, 2006].

2.4.1 Trabalhos Correlatos

Como este trabalho pretende averiguar o uso de diferentes implementações de Mapas Topológicos, de forma a associar a estes distintas semânticas relacionadas às diferentes informações do ambiente, a seguir serão apresentados os principais trabalhos correlatos

encontrados na bibliografia.

O mapeamento topológico utilizando mapas de Kohonen [Kohonen, 2001] foi realizado inicialmente por Benjamin Kuipers e Yung-Tai Byun em [Kuipers and Byun, 1988]. A Figura 2.2 apresenta o mapa topológico utilizado pelos autores, onde os nodos representam os máximos locais obtidos a partir das medidas dos sensores e as arestas apresentam como se deslocar de um ponto a outro, através de informações métricas como a distância do percurso. A atualização do mapa ocorria enquanto o robô percorria o ambiente.

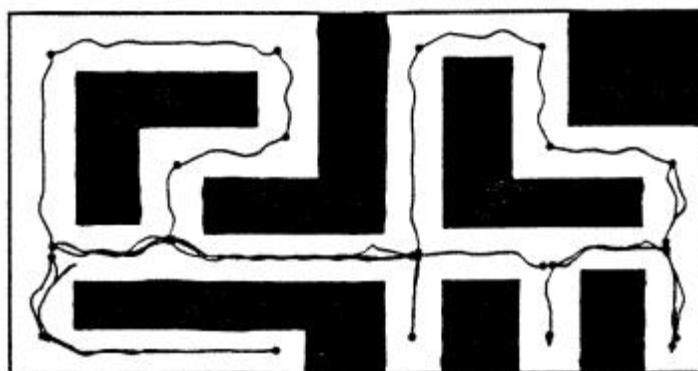


Figura 2.2: Exemplo de mapa topológico proposto em [Kuipers and Byun, 1988].

[Nehmzow and Smithers, 1991] e [Nehmzow, 1992] propuseram a utilização de mapas auto-organizáveis para construir uma representação do ambiente baseada na detecção de canto. Essa informação foi adquirida através de dois sensores de toque. Uma abordagem semelhante foi usada no trabalho de [Owen and Nehmzow, 1998], mas utilizando sensores mais complexos: um sonar e uma bússola em um robô Nomad 2000.

No trabalho Zimmer [Zimmer, 1995], o peso de cada neurônio de uma SOM é relacionado com a posição métrica e com a percepção sensorial. Essas informações sensoriais são concatenadas com a posição do robô, constituindo assim, vetores sensoriais. Para determinar o neurônio vencedor (BMU) e para a métrica de atualização efetua-se diferentes ponderações para cada informação (métrica e sensorial).

Outras abordagens foram as propostas em [Kurz, 1996] e [Zimmer, 1996]. Na primeira é proposta a utilização de mapas auto-organizáveis em um robô dotado de odometria nas rodas e sonares. Nesse trabalho, a localização é estimada por um filtro de Kalman estendido (EKF), com esta informação os dados do sonar são usados para o aprendizado do mapa. No trabalho de [Zimmer, 1996] estima-se e corrige-se a posição, caso necessário,

além de construir o mapa do ambiente usando uma abordagem semelhante a estrutura de células de crescimento (GCS)[Fritzke, 1993b]. O método permite alterar dinamicamente o tamanho do mapa. As informações utilizadas nesse trabalho são de odometria de rodas associadas com sensores de toque e de distância por infravermelho. Uma característica interessante da abordagem de Zimmer é o atraso no sistema de percepção, para evitar instabilidade. Outros trabalhos também propuseram o uso do SOM para fazer mapeamento de robôs no mesmo período histórico [Vlassis et al., 1997] [Gerecke, 1999]. No trabalho de [Jun and Duckett, 2002] os mapas auto-organizáveis e as redes neurais foram utilizadas para aprender o comportamento de um robô.

[Hafner, 2000] propôs a utilização de mapas auto-organizáveis para construir mapas perceptivos baseados em dados obtidos de uma câmera omnidirecional e uma bússola. Os mapas gerados por essa abordagem não contém informação espacial, permitindo a navegação e localização com base apenas em informação sensorial, não existe o conhecimento de informações métricas, utiliza-se apenas dados de orientação, que são inseridos ao mapa sob forma de angulação das arestas do grafo formado pela SOM. As arestas entre neurônios são baseados na ordem temporal. Portanto, para construir um mapa com significado métrico é incorporado ao modelo a orientação (robô se locomove no plano). O aprendizado é baseado em percepções do ambiente, o peso da aresta decai com o tempo, e a angulação entre os movimentos interfere no aprendizado. Além disso, o uso do princípio da força de atração e repulsão, permite determinar uma "posição" espacial para o neurônio. recentemente, [Hafner, 2008] estendeu o trabalho anterior.

No trabalho proposto em [Drews Jr et al., 2008] e, depois, em [Botelho et al., 2009], foi apresentada uma abordagem de mapeamento topológico, baseada em informações visuais aplicada a veículos submarinos, usando SOM, semelhante ao GCS, onde foram gerados mapas subaquáticos durante a navegação do veículo, que localiza o robô e, posteriormente, corrige o mapa.

[Koutník et al., 2006] propuseram um método para gerar mapas 3D usando dados obtidos a partir de dois laser scanner SICK LMS. A pose é estimada utilizando o método de registo ICP. Após isso, um mapa global é gerado e agrupados usando agrupamento K-means. Cada grupo é representado pela SOM e fundido a fim de obter um mapa 3D robusto. Em [Miranda Neto et al., 2008] é apresentado um método que permite a

navegação autônoma e semi-autônoma, através do uso de dois SOM distintos, o primeiro é usado para aprender um mapa das regiões navegáveis usando uma câmera monocular. Já o outro SOM é usado para aprender os possíveis comandos enviados para o robô durante a navegação. Os resultados são obtidos utilizando o banco de dados DARPA Grand Challenge.

Outro método para a construção de mapas topológicos são os que utilizam as Cadeias de Markov. No trabalho de [Shatkay and Kaelbling, 2002] os estados no modelo de Markov correspondem aos lugares no ambiente, as arestas entre esses lugares correspondem ao caminho que deve ser percorrido pelo robô. Por ser um modelo probabilístico, já que o movimento do robô é visto como um processo de Markov parcialmente observável nas quais as ações correspondem a transições probabilísticas de estados, esse método consegue lidar com as incertezas nos atuadores do robô.

No trabalho de [Pedrosa, 2006], um dos objetivos é criar uma representação topológica, onde os nodos representem as salas e corredores de um determinado ambiente interno e as arestas representam as passagens entres as salas, tanto os nós quanto as arestas armazenam informações métricas relativas ao ambiente.

Em [Thrun, 1998] é apresentado um modelo de mapeamento híbrido. Para a construção desse mapa, primeiramente, o robô explora o ambiente e cria um mapa métrico, através da abordagem de grade de ocupação. Em seguida gera-se um mapa topológico a partir do mapa métrico. A construção do mapa topológico diminui o problema de ambiguidade e correspondência [Thrun, 1998].

2.5 Considerações do Capítulo

Neste capítulo foi apresentado um panorama geral do problema de Mapeamento Robótico. Foram apresentados os principais conceitos e paradigmas associados a Localização e SLAM.

O problema da síntese de ambientes foi formalizado, tendo sido apresentadas algumas técnicas utilizadas para a sua solução. Como o trabalho a ser apresentado é baseado no uso de Mapas Topológicos, estes foram descritos em detalhe.

Pode-se afirmar que o custo (computacional, sensorial, etc.) e a aplicação irão deter-

minar qual a abordagem melhor se adapta ao sistema. Se colisões são críticas e a rapidez de reação ao ambiente é fundamental, a abordagem métrica é mais vantajosa. Entretanto, quando a robustez e simplicidade forem necessárias, é conveniente utilizar o mapeamento topológico, principalmente se o ambiente for dinâmico. Modelos híbridos que apresentem representações espaciais são abordagens que buscam aliar as vantagens de ambos.

Finalmente, foram apresentado um conjunto de trabalhos correlatos os quais utilizam Mapas Topológicos ou Mapeamento Híbrido, de forma a situar a proposta aqui apresentada frente a literatura existente.

Capítulo 3

Redes Neurais Auto-Organizáveis

Tendo em vista a adoção, neste trabalho, de Mapas Auto-organizáveis como modelo de representação de mapas robóticos, este capítulo apresentará tal abordagem apresentando suas características e principais conceitos.

Originalmente, os Mapas Auto-organizáveis (SOM, Self Organizing Maps) foram propostos com o objetivo principal de transformar um padrão de sinal incidente de dimensão arbitrária n em um mapa discreto geralmente uni ou bidimensional, realizando esta transformação adaptativamente de uma maneira topologicamente ordenada [Haykin, 2007]. Para tal, SOMs são implementados através de grades de neurônios de dimensão m . Cada nodo da grade é composto de uma unidade de processamento (neurônio); sendo atribuído a esse um peso de dimensão n . Os diferentes nodos têm seus pesos adaptados (e de sua vizinhança) de forma a melhor representar o espaço de entrada também de dimensão n , a ser apresentado ao mapa. A adaptação da grade é baseada na aprendizagem competitiva e não-supervisionada [Russel and Norvig, 2003]. Assim, a rede é definida como:

- A é um conjunto de neurônios. Cada nodo $c \in A$ tem um vetor de peso sinápticos associados $w_c \in R^n$, onde n é a dimensão do espaço de entrada.
- C é um conjunto de arestas entre os pares de neurônios. O objetivo dessas ligações é definir a estrutura topológica.

A seguir, serão apresentadas algumas variações de mapas auto-organizáveis.

3.1 Redes de Kohonen

As redes de Kohonen foram propostas, inicialmente, por Teuvo Kohonen em 1982 [Kohonen, 1982]. A arquitetura proposta consiste de um tipo de SOM com topologia fixa de células com relação de vizinhança, formando uma grade multi-dimensional, porém as mais comuns são 2D e 3D.

O treinamento da rede se dá iterativamente, a partir da inicialização aleatória dos pesos sinápticos da grade. Para Haykin [Haykin, 2007], após a inicialização da grade, há três processos essenciais envolvidos na formação do mapa auto-organizável: competição, cooperação e adaptação sináptica, descritos a seguir:

1. **Inicialização:** estabelece os valores de pesos iniciais da rede, através dos seguintes passos:
 - definir os parâmetros de aprendizagem e de vizinhança;
 - inicializar o peso w_c (normalmente aleatoriamente).
2. **Competição:** determina o neurônio vencedor (s), ou seja, aquele que melhor se assemelha com o vetor de entrada (x), como mostra a Equação 3.1.

$$s = \operatorname{argmin}_A \|x - w_c\|. \quad (3.1)$$

3. **Cooperação:** define a função de vizinhança centrada no neurônio vencedor, uma função utilizada para esse definição de vizinhança é a função gaussiana, conforme Equação 3.2.

$$h_{c,s} = \exp\left(-\frac{\|r_c - r_s\|^2}{2\sigma^2}\right). \quad (3.2)$$

onde σ é a largura da vizinhança, r_c é posição de um dado neurônio e r_s é a posição do neurônio vencedor s . Como σ decai com o tempo, pode-se utilizar uma função de decaimento exponencial, conforme mostra a Equação 3.3.

$$\sigma(n) = \sigma_0 \exp\left(-\frac{n}{\tau_1}\right) \quad (3.3)$$

onde σ_0 é o valor de σ quando a rede é inicializada, τ_1 é uma constante de tempo e n é a iteração atual. A vizinhança topológica assume uma forma variável no tempo, conforme mostra a Equação 3.4, representando como o padrão de entrada interfere na vizinhança do nodo vencedor.

$$h_{c,s}(t) = \exp\left(-\frac{\|r_c - r_s\|}{2\sigma^2(t)}\right), \quad t = 1, 2, \dots \quad (3.4)$$

4. **Adaptação:** adaptação dos pesos à entrada, de forma iterativa. A Equação 3.5 é aplicada a todos os neurônios da grade dentro da região de vizinhança $h_{c,s}$.

$$w_c(t+1) = w_c(t) + \eta(t)h_{c,s}(t)(x(t) - w_c(t)). \quad (3.5)$$

onde $\eta(t)$ é a taxa de aprendizagem que decai com o tempo.

3.2 Estruturas Celulares Crescentes - GCS (*Growing Cell Structures*)

Essa estrutura foi proposta por Bernd Fritzke [Fritzke, 1993b], buscando resolver a maior limitação das redes de Kohonen, a topologia fixa, visto que para definir o tamanho ótimo da rede é necessário ter algum conhecimento a priori, o que normalmente não está disponível [Fritzke, 1993a], sem esse conhecimento pode-se limitar a capacidade da rede.

A topologia inicial da rede é composta por um simplexo k -dimensional, onde k representa o número de dimensões que a estrutura possui, por exemplo, para $k = 1$, o simplexo k -dimensional é um segmento de reta; para $k = 2$ um triângulo; para $k = 3$ um tetraedro, conforme Figura 3.1; para $k > 3$ a estrutura obtida é denotada hipertetraedro. Os $(k+1)$ vértices de um simplexo correspondem aos neurônios, e as $(k+1)k/2$ arestas, às relações de vizinhança topológica entre os neurônios. Durante o treinamento, novos neurônios são adicionados à rede, e algumas vezes neurônios são removidos. Qualquer alteração feita na rede, como inserção ou remoção de neurônios, é realizada de maneira que após a operação a rede continue formada apenas por simplexos k -dimensionais [Fritzke, 1993a].

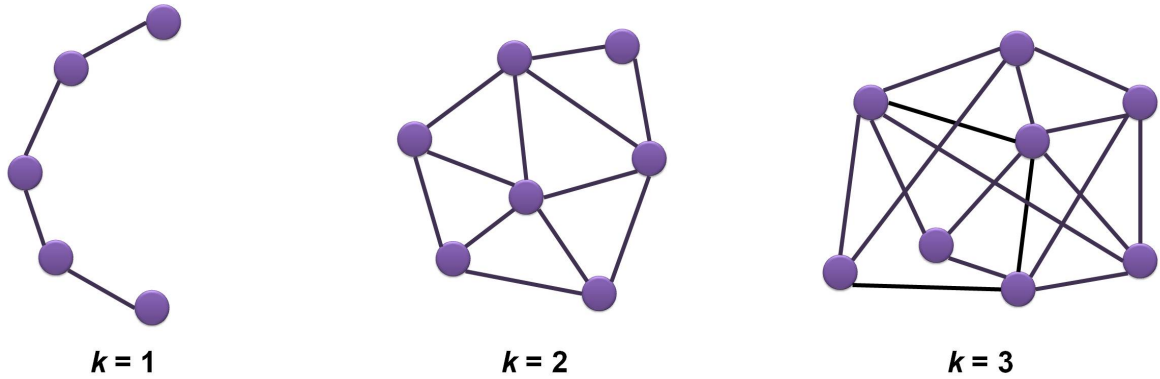


Figura 3.1: Estrutura celular com diferentes dimensionalidade de k .

O treinamento ocorre de maneira muito semelhante às redes de Kohonen. Para cada caso apresentado, uma célula vencedora é eleita, deslocando o vetor da célula vencedora e das vizinhas uma fração em relação ao caso apresentado. Porém, existe duas diferenças: a taxa de aprendizagem permanece constante durante processo de treinamento, e apenas o neurônio vencedor e seus vizinhos diretos tem seus vetores de peso sináptico atualizados.

O processo de aprendizado da GCS é descrito a seguir:

1. Inicializar o treinamento com um simplexo $k - dimensional$. Os $(k + 1)$ nodos são inicializados com valores aleatórios de vetores de pesos sinápticos.
2. Selecionar um sinal de entrada x , de acordo com o espaço de entrada $P(x)$.
3. Determinar o neurônio vencedor (*BMU - best-matching unit*) s que possui o vetor peso w_i mais similar com a entrada x . Geralmente, utiliza-se a medida da distância Euclidiana, conforme Equação 3.6.

$$s = \operatorname{argmin}_A \{ \|w_c - x\| \} (\forall c \in A). \quad (3.6)$$

4. Somar o quadrado da distância entre a entrada x e o vetor peso sináptico do neurônio vencedor s , para obter a variável de erro local E_s , como mostra a Equação 3.7.

$$\Delta E_s = \|w_s - x\|^2. \quad (3.7)$$

5. Mover s e seus vizinhos topológicos diretos na direção de x , por frações ϵ_b e ϵ_n , parâmetros da rede, como mostram as equações 3.8 e 3.9.

$$\Delta w_s = \epsilon_b (x - w_s), \quad (3.8)$$

$$\Delta w_i = \epsilon_n (x - w_i) (\forall i \in N_s). \quad (3.9)$$

onde N_s é o conjunto de vizinhos topológicos diretos da unidade c e $\epsilon_b > \epsilon_n$.

6. Se o número de sinais de entrada gerados, até então, for um inteiro múltiplo de um parâmetro λ , onde λ é um valor estipulado pelo usuário que representa o intervalo de iterações entre a inserção de um novo nodo, então insere-se um novo neurônio, seguindo os seguintes passos:

- Determinar o nodo q que possui o maior erro acumulado, de acordo com a Equação 3.10.

$$q = \operatorname{argmax}_i \{E_c\} (\forall c \in A). \quad (3.10)$$

- Inserir um novo neurônio r dividindo a maior aresta que emana de q . Considerando que esta aresta leva ao nodo f , inserir duas novas conexões (q, r) e (r, f) , e remover a conexão original (q, f) . Para reconstruir a estrutura de forma que ela continue consistindo apenas de simplexes $k - dimensionais$, o novo neurônio r deve estar conectado com todos os vizinhos comuns de q e f , ou seja, com todos os nodos dentro do conjunto $N_q \cap N_f$.
- Interpolar o vetor de r dos vetores de q e f , conforme Equação 3.11:

$$w_r = 0,5 (w_q + w_f). \quad (3.11)$$

- Decrementar o valor da variável de erro de toda a vizinhança de r , pela Equação 3.12.

$$\Delta E_i = -\frac{\alpha}{|N_i|} E_i (\forall i \in N_r). \quad (3.12)$$

- Definir a variável de erro do novo neurônio r , a partir do valor médio de seus vizinhos, conforme Equação 3.13.

$$E_i = -\frac{\alpha}{|N_i|} \sum_{i \in N_r} E_i. \quad (3.13)$$

7. Decrementar a variável de erro de todos os neurônios, de acordo com a Equação 3.14.

$$\Delta E_c = -\beta E_c (\forall c \in A). \quad (3.14)$$

8. Se o critério de parada não for satisfeito, voltar ao passo 2.

É possível incluir um critério para remover neurônios. No trabalho de [Hodge and Austin, 2001] esse processo é detalhado.

Remoção: a partir de um tamanho de grade pré-definido, células podem ser excluídas. O processo de exclusão permite que nodos muito distantes dos seus vizinhos sejam eliminados do mapa. A Figura 3.2 apresenta um exemplo de rede bidimensional, portanto, a topologia dessa rede é triangular. Ao excluir um neurônio, é necessário percorrer os demais neurônios de sua vizinhança, caso restem neurônios com apenas uma conexão (assim não mantendo a topologia triangular, fundamental dessa rede), estes também devem ser removidos.

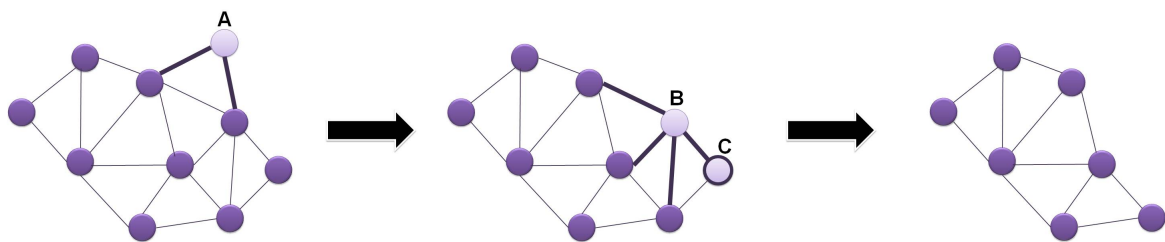


Figura 3.2: Remoção de neurônio em uma rede bidimensional.

3.3 Rede Neural Gas - GNG (*Growing Neural Gas*)

Essa estrutura foi proposta por Bernd Fritzke [Fritzke, 1995], e é uma modificação da Neural Gas (NG) apresentado por Martinetz e Schulten [Martinetz and Schulten, 1991].

Para iniciar o treinamento da rede são necessários dois nodos conectados entre si. Os dois nodos mais próximos são selecionados segundo uma função de distância Euclidiana para cada padrão apresentado à rede. Se não existir uma aresta entre elas o métodos trata de criá-la. O nó vencedor e a sua vizinhança sofrem um processo de adaptação dos pesos sinápticos.

Nessa técnica há um contador que controla a idade de cada aresta da rede. No caso de cada aresta, se o neurônio que ela conectar for o vencedor, então seu contador de idade é zerado, por outro lado se seu contador atingir um valor limiar esta aresta é eliminada, assim priorizando as arestas mais novas em relação as antigas. Esta técnica visa atualizar a cada instante a topologia da rede, mantendo apenas arestas que sejam úteis.

No final do processo de treinamento, o arranjo da rede é chamado de triangulação induzida de Delaunay [Fritzke, 1995], uma vez que mantém de uma maneira ótima a topologia dos dados de entrada. Esse tipo de rede é capaz de recuperar uma estrutura de dados bastante complexa, com diferentes dimensões e regiões desconexas. Porém, a convergência é lenta, pois os neurônios que são inseridos são adicionados em cada iteração múltipla de uma constante (λ).

Nessa abordagem, a topologia da rede é gerada de forma incremental, pela Aprendizagem Hebbiana Competitiva. A dimensionalidade de rede depende dos dados de entrada e podem variar localmente. Em [Fritzke, 1997], a construção da rede GNG segue os seguinte passo:

1. Inicializar um conjunto A de duas unidades a e b em posições aleatórias w_a e w_b em R_n , conforme Equação 3.15.

$$A = \{a, b\}. \quad (3.15)$$

Inicializar o conjunto de conexões C , de forma a conter uma aresta ligando a e b , e inicializar a idade, que mede a atividade das conexões. Definir a primeira idade em

zero, conforme Equação 3.16 e Equação 3.17.

$$C = \{(a, b)\}, \quad (3.16)$$

$$idade_{(a,b)} = 0. \quad (3.17)$$

2. Determinar um padrão de entrada x .
3. Determinar os dois neurônios s_1 e s_2 , com pesos sinápticos mais similares a x , conforme Equação 3.18 e Equação 3.19.

$$s_1 = \operatorname{argmin}_i \{\|x - w_i\|\} (\forall i \in A), \quad (3.18)$$

$$s_2 = \operatorname{argmin}_i \{\|x - w_i\|\} (\forall i \in A \setminus s_1). \quad (3.19)$$

4. Se ainda não existir, inserir uma conexão entre s_1 e s_2 em C , conforme 3.20.

$$C = C \cup \{(s_1, s_2)\}. \quad (3.20)$$

Em qualquer caso, definir a idade da conexão entre s_1 e s_2 como zero, segundo Equação 3.21.

$$idade_{(s_1,s_2)} = 0. \quad (3.21)$$

5. Adicionar o quadrado da distância entre o sinal de entrada x e o vetor peso mais similar s_1 em uma variável de erro local, a partir da Equação 3.22.

$$\Delta E_{s_1} = \|w_{s_1} - x\|^2. \quad (3.22)$$

6. Mover s_1 e sua vizinhança topológica direta para x , pelos parâmetros ϵ_b e ϵ_n , respectivamente, a distância total, conforme Equação 3.23 e Equação 3.24, sendo N_{s_1} a vizinhança do neurônio s_1 .

$$\Delta w_{s_1} = \epsilon_b (x - w_{s_1}), \quad (3.23)$$

$$\Delta w_i = \epsilon_n (x - w_i) (\forall i \in N_{s_1}). \quad (3.24)$$

7. Incrementar a idade de todas as conexões que não emanam de s_1 , utilizando a Equação 3.25.

$$idade_{(s_1,i)} = idade_{(s_1,i)} + 1 (\forall i \in N_{s_1}), \quad (3.25)$$

onde N_c é o conjunto de vizinhos topológicos diretos de c .

8. Remover as arestas com idade maior que $idade_{max}$, e se esta operação resultar em unidades sem arestas, removê-las também.
9. Se o número de sinais de entrada gerados até o momento for maior que um inteiro múltiplo de um parâmetro λ , inserir um novo neurônio, como segue:
- Determinar a unidade q com o maior erro acumulado, a partir da Inequação 3.26.

$$E_q \geq E_c (\forall c \in A). \quad (3.26)$$

- Interpolar um novo neurônio r a partir de q e seu vizinho f com a maior variável de erro, conforme Equação 3.27 e Equação 3.28.

$$A = A \cup \{r\}, \quad (3.27)$$

$$w_r = 0,5 (w_q + w_f). \quad (3.28)$$

- Inserir arestas conectando a nova unidade r com as unidades q e f , e remover a aresta original entre q e f , conforme Equação 3.29 e Equação 3.30.

$$C = C \cup \{(r, q), (r, f)\}, \quad (3.29)$$

$$C = C \setminus (r, f). \quad (3.30)$$

- Diminuir a variável de erro de q e f , de acordo com a Equação 3.31 e a Equação 3.32.

$$\Delta E_q = -\alpha E_q, \quad (3.31)$$

$$\Delta E_f = -\alpha E_f. \quad (3.32)$$

Determinar a variável de erro de r interpolando as variáveis de erro de q e f , conforme a Equação 3.33.

$$E_r = 0,5 (E_q + E_f). \quad (3.33)$$

10. Diminuir a a variavel erro de todos os neurônios, através da Equação 3.34

$$\Delta E_c = -\beta E_c. \quad (3.34)$$

11. Se o critério de parada não for satisfeito, continuar no passo 2.

Uma das principais deficiências do método é a incapacidade de se adaptar de modo rápido a dados de ambientes dinâmicos.

3.4 Cresce Quando Necessário - GWR (*Grow When Required*)

Marsland, Shapiro e Nehmzow [Marsland et al., 2002] propuseram essa abordagem, na qual neurônios são inseridos à rede de forma automática. Assim, a rede cresce rapidamente quando novos dados são apresentados a ela, mas interrompe o processo de crescimento uma vez que tenha convergido para uma solução estável.

Nessa abordagem a estrutura só crescerá se necessário, devido a presença de uma entrada que não corresponda a nenhum nodo, dentro de um limiar de erro estipulado.

Esse novo nó tem seus pesos inicializados para representar esta entrada. A rede para de crescer quando estiver adequadamente dimensionada as entradas e volta a crescer se a distribuição dos dados se modificarem.

Nessa estrutura tanto nós como arestas podem ser inseridas e removidas. Para adicionar uma aresta é usado o aprendizado Hebbiano, estabelecendo uma ligação entre os dois nós mais representativos para cada entrada, o nó vencedor e o segundo colocado. A remoção da conexão é realizada de modo igual da rede GNG, pelo critério de idade.

Para adicionar um nodo, nesse tipo de estrutura são utilizados dois tipo de conceitos: atividade e habituação [Marsland et al., 2002]. A atividade do nodo vencedor $a(s)$ é função da distância desse nodo ao vetor de entrada, e mede o grau de semelhança entre o vetor de entrada e o vetor de pesos do nodo. Um limiar de atividade a_x é responsável por definir se o nodo é ou não um bom representante da entrada. Caso a atividade do nodo vencedor não seja suficientemente alta, novos nodos podem ser adicionados à rede. Tal operação pode ser feita a qualquer momento, não apenas a cada λ iterações como na rede GNG. O conceito de habituação completa a decisão de acrescentar ou não um nodo. Disparo de um novo nodo é uma função que mede quão frequentemente o nodo é utilizado, com decréscimo exponencial de 1 para 0. Se o valor do disparo do nodo vencedor $h(s)$ estiver acima de um certo limiar h_x , o nodo é considerado novo e ainda precisa ser treinado. Caso contrário, isso significa que o nodo já foi suficientemente treinado; e é apenas nesse caso que a rede cresce.

Com esse método, consegue-se determinar em que situação a rede realmente precisa crescer, e isso ocorre de modo confiável. Também é possível determinar quando os nós precisam apenas de mais treino, ou quando nada precisa ser feito. Logo, esse tipo de rede é eficiente e econômica, sendo capaz de representar adequadamente conceitos que mudam com o tempo [Klinkenberg, 2004].

O treinamento da GWR segue os seguintes passos [Marsland et al., 2002]:

1. Inicializar a rede com dois neurônios, de acordo com Equação 3.35.

$$A = \{c_1, c_2\}. \quad (3.35)$$

2. Obter um padrão x para a entrada da rede.

3. Para cada nodo i da rede calcula-se da distância euclidiana com a entrada, conforme Equação 3.36 e selecionar o primeiro e segundo neurônio que melhor represente a entrada, de acordo com Equação 3.37 e Equação 3.38, respectivamente.

$$d = \|x - w_i\|, \quad (3.36)$$

$$s = \operatorname{argmin}_{c \in A} \|x - w_c\|, \quad (3.37)$$

$$t = \operatorname{argmin}_{c \in A \setminus \{s\}} \|x - w_c\|. \quad (3.38)$$

onde w_c é o vetor peso do nodo n

4. Se não houver uma conexão entre s e t , então criar uma, conforme Equação 3.39.

$$C = C \cup \{(s, t)\}. \quad (3.39)$$

Caso contrário, definir a idade da conexão como 0.

5. Calcular a atividade do neurônio vencedor, a partir da Equação 3.40.

$$a = \exp(-\|x - w_c\|)^2. \quad (3.40)$$

6. Se a atividade a for menor que um limiar de atividade a_x e o contador de disparo maior que um limiar de disparo h_T , um novo nodo deve ser adicionado entre os dois melhores nós correspondentes s e t .

- Adicionar um novo nodo r , conforme Equação 3.41.

$$A = A \cup \{r\}. \quad (3.41)$$

- Criar um novo vetor peso, a definição desse peso é a partir do cálculo da média entre x e w_s [Marsland et al., 2002].

- Inserir uma conexão entre os nodos r e s e entre r e t , a partir da Equação 3.42.

$$C = C \cup \{(r, s) (r, t)\}. \quad (3.42)$$

- Remover a ligação entre s and t , conforme Equação 3.43.

$$C = C / \{(s, t)\}. \quad (3.43)$$

7. Se um novo neurônio não é adicionado, adaptar os pesos do neurônio vencedor e de seus vizinhos, de acordo com a Equação 3.44 e Equação 3.45.

$$\Delta w_s = \epsilon_b \times h_s \times (x - w_s), \quad (3.44)$$

$$\Delta w_i = \epsilon_n \times h_i \times (x - w_i), \quad (3.45)$$

onde $0 < \epsilon_n < \epsilon_b < 1$ e h_s é o valor do contador de disparo para o nodo s .

8. Atualizar a idade das arestas com final em s , conforme Equação 3.46.

$$age_{(s,i)} = age_{(s,i)} + 1. \quad (3.46)$$

9. Reduzir o contador de frequência do nodo vencedor s de acordo com as equações 3.47 e 3.48.

$$h_s(t) = h_0 - \frac{S(t)}{\alpha_v} (1 - e^{-\alpha_b t / \tau_b}), \quad (3.47)$$

e os contadores de seus vizinhos (i).

$$h_s(t) = h_0 - \frac{S(t)}{\alpha_c} (1 - e^{-\alpha_c t / \tau_c}). \quad (3.48)$$

onde $h_i(t)$ é o tamanho da variável de habituação para o nó, h_0 é a intensidade inicial e $S(t)$ é a intensidade de estímulo, geralmente 1, α_c , α_b , τ_c e τ_b são constantes que controlam o comportamento da curva. O contador de disparo do vencedor diminui mais rápido que de seus vizinhos.

10. Verifique se existem nodos ou arestas para eliminar, ou seja, se existem nodos que não têm vizinhos, ou arestas que são mais velhos do que a maior idade permitida, caso isso aconteça deverão ser excluídos.
11. Se existir entradas, voltar ao passo (1) a menos que algum critério de parada tenha sido atingido.

3.5 Considerações do Capítulo

Neste capítulo, foram relatados os principais conceitos associados a Mapas Auto-organizáveis. Então, apresentaram-se as principais abordagens para a sua implementação (ver tabela 3.1). Em função das características descritas, a Rede de Kohonen e a Rede GCS foram escolhidas para construir os Mapas Topológicos que representaram os ambientes robóticos.

Com base no Modelo de Mapa Topológico adotado, o próximo capítulo apresenta a arquitetura proposta para um sistema de Representação de Ambientes, através de Mapas Topológicos dispostos em diferentes camadas taxonômicas.

Tabela 3.1: Comparação das diferentes abordagens de redes neurais auto-organizáveis

Abordagem	Vantagens	Desvantagens
Redes de Kohonen	Agrupa informações semelhantes de maneira simplificada, porém de maneira eficiente.	A topologia fixa, limita o resultado do mapeamento. A arquitetura precisa ser definida antes de iniciar o processo de treinamento, o que é uma tarefa não trivial quando se trabalha com uma grande quantidade de dados desconhecidos.
GCS	Não é necessário definir antes do treinamento o tamanho da rede. Além disso, todos os parâmetros são constantes, não tendo a necessidade de terem os seus valores diminuídos à medida que o treinamento se desenvolve. Geralmente, a configuração dos parâmetros não é problemática (Fritzke, 1997).	Devido à estrutura de grafo empregada ser dinâmica e irregular, o modelo é mais difícil de implementar que modelos com uma estrutura regular pré-definida (Fritzke, 1997).
GNG	O tamanho da rede não precisa ser definido. Todos os parâmetros são constantes e também o número total de passos de adaptação não precisa ser definido a priori. O processo de crescimento pode ser interrompido quando um critério definido pelo usuário for alcançado. Devido ao crescimento semelhante a um fractal todos os passos intermediários da rede são descrições muito boas da distribuição de dados com uma resolução dependendo do tamanho da rede atual (Fritzke, 1997).	A visualização direta dos dados só é possível se os dados forem de baixa dimensionalidade. A razão disso é que o modelo não realiza uma redução de dimensionalidade, ou seja, o modelo construído possuirá a mesma dimensionalidade dos dados originais (Fritzke, 1997). Além disso, devido à possibilidade de criar redes desconexas, pode dificultar o aprendizado.
GWR	O tamanho da rede também não é definido <i>a priori</i> . O crescimento da rede ocorre somente quando necessário.	Apresenta um número maior de parâmetros que as redes anteriores e de difícil ajuste.

Capítulo 4

Metodologia

Este capítulo descreve a metodologia utilizada para tratar dos problemas de mapeamento robótico. A proposta é baseada em Mapas auto-organizáveis para representação das impressões do ambiente, dispostos em camadas com diferentes significados taxonômicos. Primeiramente, apresenta-se uma visão geral da arquitetura proposta, posteriormente, uma descrição detalhada da abordagem utilizada.

4.1 Requisitos do Projeto

Dada a complexidade do problema de mapeamento, a proposta apresentada visa tratar, de forma mais específica, as seguintes questões:

- incertezas das medições: deve ser considerada a presença de ruídos nos sinais captados. A dependência probabilística das medições conduz a erros incrementais relativos a espacialidade das impressões sensorias;
- escalabilidade: a solução apresentada deverá considerar a alta dimensionalidade das informações sensoriais capturadas, permitindo a sua tratabilidade em termos de tempo de processamento e espaço de armazenamento;
- correspondência: o sistema deverá tratar os aspectos de percepção similares ao longo do ambiente, detectando correspondências e otimizando seu tratamento;
- dinamicidade: a arquitetura deverá apresentar adaptabilidade ao longo de processo

de construção dos mapas. Tal plasticidade permitirá a melhoria contínua do registro do ambiente, bem como será tolerante às possíveis modificações do ambiente navegado;

- a representação obtida deverá permitir a adoção de estratégias de navegabilidade ao sistema, permitindo o planejamento de trajetórias com vistas à realização de missões de alto nível pelo robô

Com base em tais requisitos, apresenta-se a seguir o detalhamento da implementação da arquitetura proposta.

4.2 Visão Geral da Arquitetura

Como apresentado nos capítulos anteriores, o problema de mapeamento robótico possui uma série de desafios. O problema de mapeamento pode ser resumido em se obter uma representação que descreva a lista de percepções realizadas e seu local no espaço. Ou seja, surgem duas dimensões a serem tratadas, uma associada ao conjunto de impressões, e outra à sua pose no espaço navegado pelo robô. Além disso, constata-se que a natureza das informações advindas dos sensores é diversificada, seja pela diferença intrínseca do sinal percebido pelo sensor i frente ao conjunto de sensores embarcados no robô, seja pelo nível de abstração da informação resultante de pós-processamentos e manipulações realizadas.

Neste trabalho, estas diferenciações das informações associadas à percepção são explicitamente tratadas na arquitetura, conduzindo a um conjunto de mapas distintos, organizados conforme as diversas naturezas das percepções realizadas. As diferentes impressões serão tratadas por representações individuais, as quais serão integradas por um tipo específico de representação associada à espacialidade do ambiente navegado (informações métricas de localização). Devido às características inteligentes, (ver Capítulo 3), dos mapas auto-organizáveis, estes serão escolhidos como estrutura para registro dos mapas.

Assim, propõe-se uma arquitetura capaz de fornecer diferentes mapas descritivos do ambiente, a partir de diferentes fontes sensoriais, utilizando mapas auto-organizáveis. De forma a integrar informações métricas de localização com abstrações topológicas advindas

de diferentes percepções do ambiente, a Figura 4.1 apresenta uma visão geral do sistema proposto. Nela divide-se o problema de mapeamento em duas categorias:

- mapas perceptivos (Mp): congregam informações perceptivas do ambiente navegado utilizando mapas auto-organizáveis. Estabelece-se que cada fonte sensorial, S_i , terá um respectivo mapa perceptivo Mp_i . Diferentes Mapas Perceptivos poderão existir em função das distintas naturezas perceptivas (dimensão e significado), permitindo a representação de sistemas multi-sensoriais (um mapa perceptivo representaria o mapeamento do sensor laser e outro seria responsável pela representação do mapeamento de uma câmera). Também diferentes Mapas Perceptivos poderão representar camadas de processamento de informação conduzindo a representações cognitivas de maior nível de abstração. Por exemplo, num nível teriam-se as representações dos descritores dos objetos pertencentes a cena, e num segundo nível estariam representados apenas o conjunto de objetos existentes no ambiente. Assim, cada mapa perceptivo Mp_i estará associado a um significado semântico próprio (tipo de sensor, nível de processamento da informação, etc).
- mapa espacial (Me): responsável por registrar informações espaciais associadas ao espaço de configuração navegável pelo robô. Utiliza abordagem auto-organizável crescente, visto que não é possível determinar *a priori* o número de nodos que o mapa terá. Estas representações estarão associadas ao local no espaço onde as impressões situam-se¹. Além de tratar a questão de espacialidade das percepções, o Mapa Espacial serve como elemento integrador entre os diferentes Mapas Perceptivos existentes no sistema.
- conexões entre mapas: são responsáveis por fazer a associação entre a informação espacial e as diferentes percepções obtidas no mesmo instante de tempo, ou seja, que representam a percepção de um dado lugar. Essas conexões feitas por meio de arestas entre os mapas permite minimizar ambiguidades, tanto para a criação dos mapas quanto a para a localização utilizando os mapas. A associação, também pode ser feita por meio de outros dois métodos de visualização. O primeiro, e padrão, desenha no nodo espacial a cor do grupo perceptivo que possui menor erro

¹Aqui admite-se metodologias de mapeamento baseadas no robô, bem como baseadas no ambiente.

de quantização, ou seja, será escolhido o nodo perceptivo, dentre os que o espacial está conectado, que é mais próximo de sua entrada original. Sendo assim, o nodo espacial receberá a cor do grupo ao qual pertence o melhor nodo perceptivo.

Já o segundo método, desenha no nodo espacial as cores de todos os grupos perceptivos ao qual ele está conectado. Essa outra abordagem respeita o número de conexões que o nodo espacial tem com cada grupo, ou seja, a parcela de cor de um grupo no nodo espacial será proporcional a sua porcentagem de conexões.

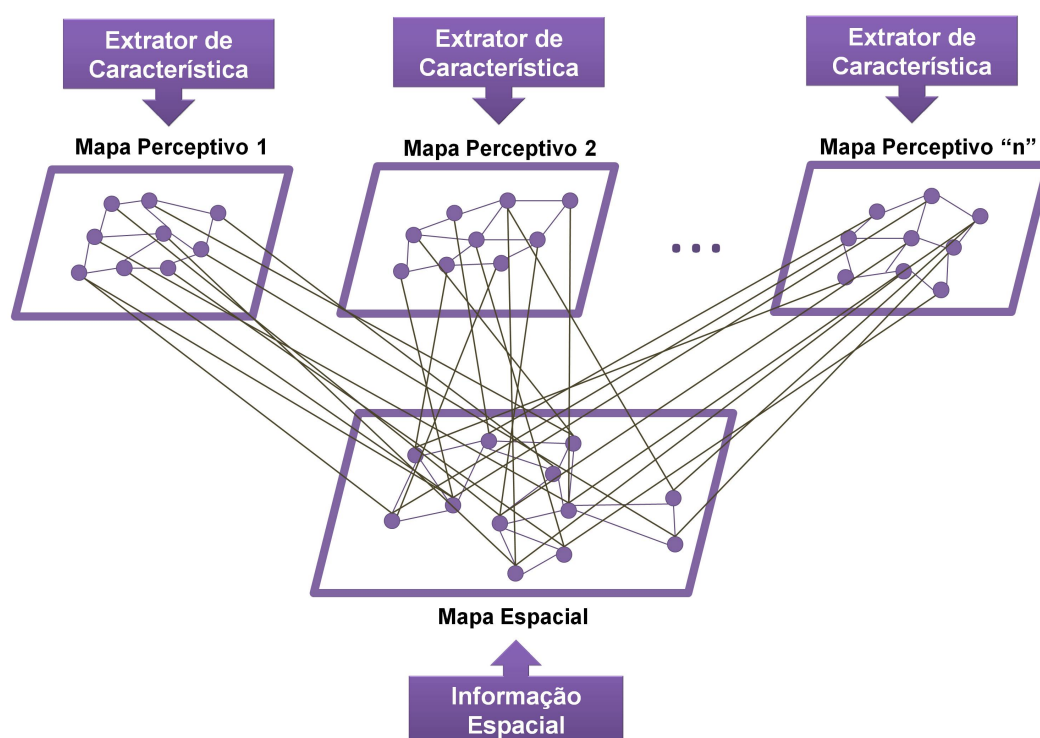


Figura 4.1: Arquitetura Proposta.

As diversas informações perceptivas registradas em cada mapa perceptivo Mp_i são associadas a diferentes regiões espaciais no mapa espacial Me por meio de arestas.

4.3 Implementação da Arquitetura

A implementação da arquitetura faz uso de técnicas e algoritmos capazes de tratar a escalabilidade, a correspondência, a presença de ruídos e a dinamicidade, conduzindo a

modelos de representação adaptativos. A arquitetura final poderá ser utilizada para a implementação de estratégias de navegação associadas a identificação, busca e registro de diferentes naturezas e níveis de descrição do ambiente, além de propiciar uma representação espacial do espaço de configuração deste.

O sistema apresentado tem como base sensores de uso comum em robótica terrestre como câmera de vídeo omnidirecional e escâner laser, além de estimativa de posição baseada em odometria das rodas do veículo. Entretanto, a proposta é genérica o suficiente para que a arquitetura seja usada para o mapeamento com outros dispositivos sensoriais. Descreve-se a seguir em detalhes a implementação dos mapas perceptivos e espaciais.

4.3.1 Implementação dos Mapas Perceptivos

Extração de Características de Imagens Informações sensoriais são utilizadas para o mapeamento autônomo por robôs. Diferentes sensores, tais como: câmeras, laser, sonares, fornecem informações do ambiente no qual o robô navega. Geralmente, o tratamento de tais informações, de elevada dimensão m , é um problema complexo a ser resolvido. O aumento da escalabilidade do sistema decorrente da evolução da exploração pelo robô impõe o emprego de técnicas de seleção de informação pertinentes, frente ao conjunto total de percepções captadas pelo robô.

Assim, neste trabalho estabelece-se inicialmente, como seletor de informações relevantes, um módulo de Extração de Características. Este receberá informações sensoriais de dimensões m reduzindo-as a n , com $n \ll m$, impressões sensoriais relevantes, a Figura 4.2 ilustra esse processo. Desta forma, cada mapa perceptivo Mp_i^n terá associado n dimensões para representação das informações do sensor.

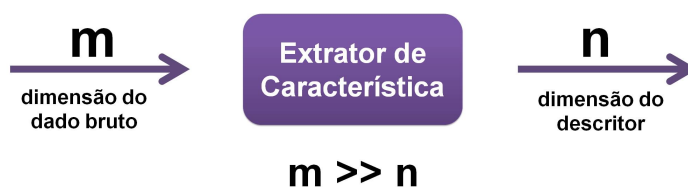


Figura 4.2: Redução Dimensional da Extração de Características.

Na implementação atual, o módulo de extração de características, é semelhante ao

realizado em [Hafner, 2000]. A Figura 4.3 ilustra a abordagem. Primeiramente, obtêm-se imagens omnidirecionais do ambiente percorrido pelo robô. Após retifica-se essas imagens transformando-as em panorâmicas [Spacek, 2003], como mostrado por A na Figura 4.3. No processo, mostrado em B na figura, é efetuada a média dos pixels das linhas para cada coluna. Após, em C, é aplicado um filtro passa baixa para suavização, no caso do presente trabalho é utilizado um filtro média com uma janela de 7 colunas. Por fim, em D é feita uma reamostragem de modo a reduzir o tamanho do descritor para 16 amostras, que servem de entrada perceptiva para o mapa.

Outros extratores de características em imagens de imagens poderiam ser utilizados, como o SURF [Bay et al., 2006], SIFT [Lowe, 2004] ou modelo de saliência [Itti et al., 1998] [Itti and Koch, 2001]. Porém, visto a similaridade entre as abordagens proposta neste trabalho e no trabalho de [Hafner, 2008], a ideia de utilizar um descritor semelhante facilita a validação dos resultados, mostrados no Capítulo 5. Para o caso de utilização de outros sensores como nuvem de pontos adquiridas com laser escâner, pode-se utilizar o descritor proposto em [Steder et al., 2011] para dados tridimensionais e os descritores propostos em [Nunez et al., 2006] ou [Zhao and Chen, 2011] para dados bidimensionais.

Representação dos Mapas Perceptivos Com vistas a tratar a dinamicidade e redundância associadas às informações sensoriais, propõe-se a implementação dos mapa perceptivo Mp_i^n como mapas auto-organizáveis, baseados em redes de Kohonen.

Os mapas auto-organizáveis foram apresentados no Capítulo 3. Na arquitetura proposta, o SOM será responsável por agrupar informações sensoriais semelhantes, bem como lidar com a dinamicidade das percepções sensoriais. Cada nodo do mapa terá igual dimensão dos descritores de entrada. Ao longo da exploração serão formados grupos responsáveis por caracterizar as diferentes entradas sensoriais existentes em categorias sensoriais relevantes, permitindo assim a compactação da informação.

Cada mapa perceptivo Mp_i^n recebe um conjunto de descritores S_i^n de seu sensor i do sistema a cada instante. Para cada entrada S_i^n , o mapa Mp_i^n é adaptado de forma a melhor representar a categoria sensorial associada a tal impressão sensorial.

O uso de mapas auto-organizáveis de Kohonen se justifica por apresentar uma maior

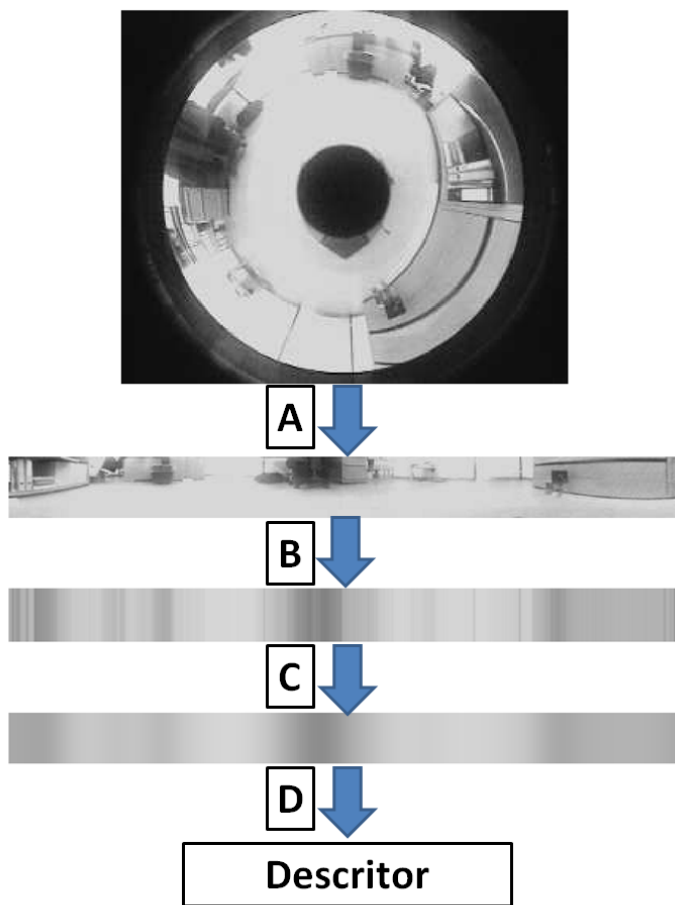


Figura 4.3: Processo de Extração de Características [Hafner, 2000].

simplicidade de parâmetros, apesar de apresentar uma topologia fixa. Essa topologia não chega a ser uma restrição, visto que o espaço de percepção, embora possa ser de grandes dimensões, pode ser limitado a um dado conjunto de percepções parecidas.

4.3.2 Implementação do Mapa Espacial

Representação do Mapa Espacial Informações associadas à localização do robô e suas percepções sensoriais no espaço de configuração são tratadas pelo Mapa Espacial. Como no caso dos Mapas Perceptivos, o Mapa Espacial é implementado através de SOMs, porém utilizado a abordagem de Estruturas Celulares Crescentes (GCS) [Fritzke, 1993a].

Os nodos da SOM representaram diferentes poses de dimensão p , onde $p = 2$ no espaço de configuração. Assim, associado a cada informação sensorial, um par $S_i^n \times G^p$, onde G^p

fornece a pose do robô no momento da captura do sinal. A informação G^p é passada ao mapa espacial que se adapta para melhor representar a informação espacial apresentada.

O uso de mapas auto-organizáveis fornece ao mapa espacial uma robustez a falhas sensoriais, bem como efetua uma compactação da informação de modo a permitir escalabilidade. O uso de mapas com tamanho crescente é necessário devido ao fato de que durante o aprendizado do mapa, não se pode definir qual o tamanho da área a ser explorada pelo robô.

4.3.3 Integração do Mapa Espacial com os Mapas Perceptivos

Os mapas perceptivos são associados ao mapa espacial, visando localizar no espaço de configuração as diferentes percepções do ambiente. Tal associação é implementada através de arestas que são estabelecidas entre os nodos vencedores dos diferentes mapas envolvidos. Assim, a cada nova percepção S_i^n obtida pelo robô cuja pose é G^p , uma aresta é inserida entre os vencedores que melhor representam em Mp_i^n e Me^p tais informações.

Essas arestas criadas permitem relacionar a informação espacial com as diferentes percepções de modo que ambiguidades sejam tratadas no espaço, enquanto dificuldades de localização podem ser resolvidas utilizando a informação perceptiva.

4.4 Considerações do Capítulo

Diferente de outras abordagens utilizando mapas auto-organizáveis para mapeamento topológico espacial e perceptivo [Zimmer, 1996], a presente proposta estabelece explicitamente em diferentes módulos da arquitetura, Mapa Espacial e Mapas Perceptivos, as diferentes naturezas das informações a serem mapeadas. As informações espaciais serão registradas no seu próprio Mapa, enquanto que cada tipo de sensor ou nível abstrato de informação pode possuir seu próprio Mapa Perceptual.

Desta forma a arquitetura proposta apresenta as seguintes características:

- **SOM**: como em abordagens semelhantes, a utilização de Mapas Auto-organizáveis trata as questões de escalabilidade permitindo que o mapeamento se dê de uma maneira mais compacta, minimizando a replicação de informação, muito comum no âmbito de percepções sensoriais;

- **SOM:** a característica adaptativa dos SOM permite tratar a dinamicidade do ambiente, através da adaptação da rede à dinâmica do universo de entrada;
- **SOM:** como métodos de clusterização SOM apresentam capacidade de lidar com incerteza nos dados, conduzindo a grupos representativos dos diferentes sinais a estes apresentados;
- **Mapa Espacial + Mapas Perceptivos:** a presença do Mapa Espacial auxilia no tratamento de ambiguidades, estabelecendo diferentes associações espaço-perceptivas a impressões sensoriais similares;
- **Mapa Espacial + Mapas Perceptivos:** a existência de um mapa perceptivo para cada tipo de informação sensorial permite o uso de diferentes técnicas para a sua implementação de acordo com as características de seu domínio;
- **Mapa Espacial + Mapas Perceptivos:** camadas cognitivas resultantes do pós-processamento de Mapas Perceptivos de nível de abstração inferiores poderão ser adicionadas ao sistema, mantendo relações espaciais com o ambiente. Desta forma, conduzindo percepções mais detalhadas e *inteligentes* do ambiente;
- **Mapa Espacial + Mapas Perceptivos:** diferentes estratégias de navegação poderão fazer uso da topologia dos registros existentes no Mapa Espacial. Também a disponibilização de diferentes tipos de informações perceptivas permite a implementação de técnicas de exploração de alto nível cognitivo.

Em função das características apresentadas, parece promissora a proposta apresentada para a representação de Mapas Perceptivos obtidos por robôs ao longo da sua navegação. No próximo capítulo será apresentado um ambiente de simulação desenvolvido, onde foram realizados experimentos de forma a verificar o desempenho e limitações do sistema.

Capítulo 5

Resultados Experimentais

A metodologia apresentada foi utilizada para o mapeamento em um conjunto de testes experimentais simulados. Neste capítulo, serão descritos o modo como a proposta foi implementada, bem como a plataforma de experimentação desenvolvida constituída de ambiente de simulação e dos robôs utilizados para a construção de diferentes cenários de teste. Além disso, serão apresentados e analisados os resultados obtidos indicando características e restrições do método.

5.1 Implementação do Sistema

A metodologia foi implementada em um sistema de software capaz de fornecer uma estrutura de representação do ambiente navegável por um robô a partir de sinais sensorizados. A estrutura possui dois principais tipos de mapas: Espacial e Perceptivo. O sistema foi implementado em C++ e a interface foi desenvolvida utilizando OpenGL com o toolkit GLUT. Para a leitura das imagens o sistema utiliza a biblioteca OpenCV.

A implementação do método proposto consistiu, principalmente, na construção das seguintes bibliotecas:

- obtenção dos descritores - responsável pelo pré-processamento dos sinais sensoriais de forma a fornecer descritores a serem apresentados aos mapas perceptivos.
- mapa espacial - que envolve a implementação da rede de estruturas celulares crescentes.

- mapa perceptivo - que envolve a implementação da rede de kohonen.
- conexão entre os mapas - estabelece relações entre os mapas, conectando os nodos do mapa espacial com os do mapa perceptivo.
- interface - funções referentes à visualização dos resultados e interação com o usuário.
- tratamento de arquivos - operações sobre imagens e gerenciamento de arquivos (textos).

5.1.1 Plataforma de Experimentação

Para a criação das simulações empregadas nos testes utilizou-se o ambiente de desenvolvimento chamado Webots, o qual permite simulações fisicamente realistas em ambientes 3D configuráveis. A plataforma apresenta uma biblioteca de robôs e sensores composta de vários modelos comerciais. A Figura 5.1 ilustra o ambiente de simulação utilizado nesse trabalho.

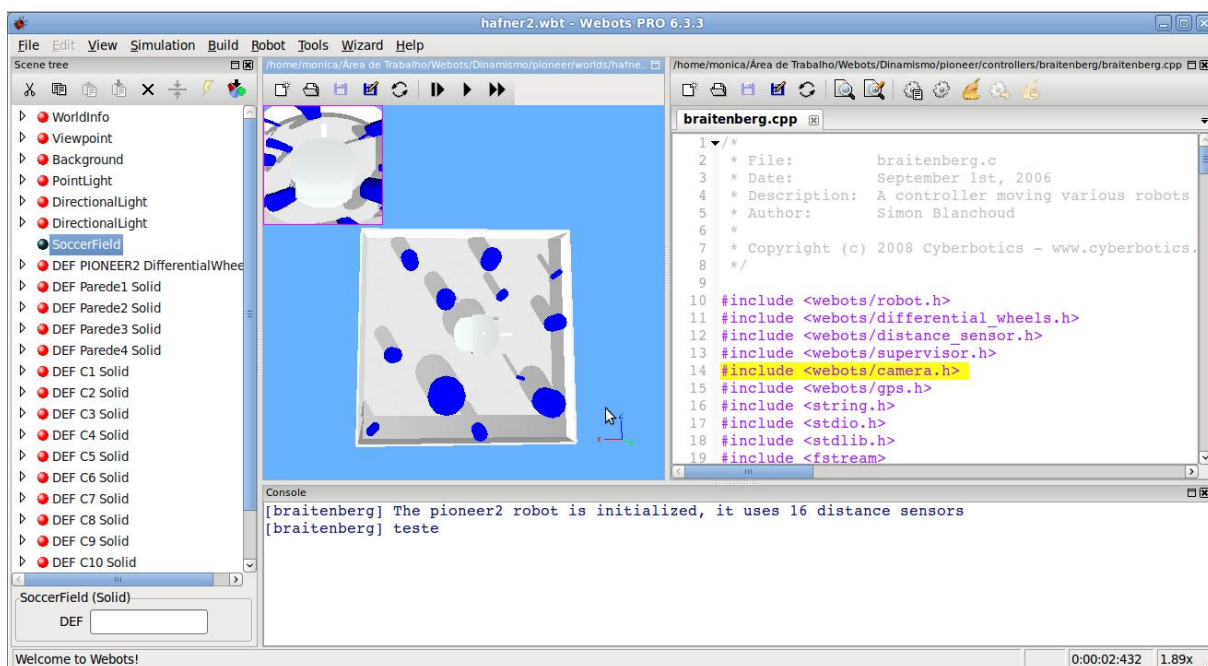


Figura 5.1: Ambiente de simulação Webots

Dentre os modelos de robôs, o simulador Webots disponibiliza o robô Pioneer, ver Figura 5.2, utilizado nos testes. Foi desenvolvido um sistema de controle de navegação

que permite o controle remoto do veículo pelo usuário.

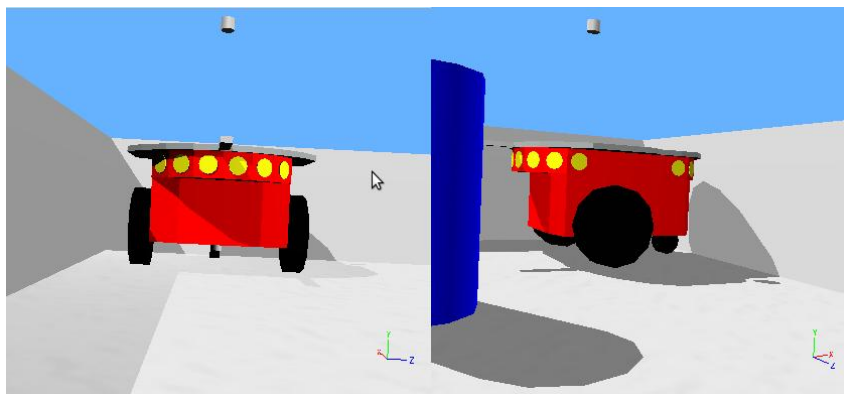


Figura 5.2: Robô do Simulador Webots

Para adquirir as informações do ambiente, foram utilizados dois tipos de sensores: uma câmera e um GPS. A câmera é utilizada em uma estrutura omnidirecional, ver Figura 5.3. As imagens omnidirecionais são retificadas, resultando em imagens panorâmicas, ver Seção 4.3. As imagens retificadas, ver Figura 5.4, serão as entradas do módulo de extração de descritores. Nos exemplos realizados, cada imagem foi subdividida em colunas, as quais serão representadas por um vetor descritor de dimensão 16. Cada elemento do vetor descreve o valor médio dos pixels da coluna [Hafner, 2008]. Já as informações advindas do GPS, correspondem às entradas do mapa espacial, uma vez que retornam a posição (x, y) do robô.

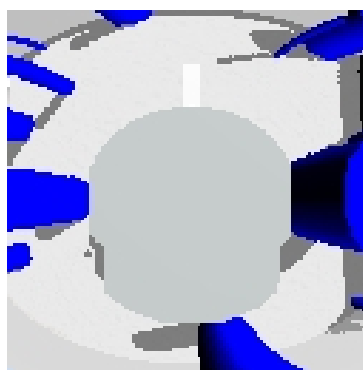


Figura 5.3: Imagem capturada pela câmera omnidirecional



Figura 5.4: Imagem retificada

5.2 Cenário de Testes

O sistema foi testado em um conjunto de cenários distintos, construídos de forma a verificar o desempenho da proposta frente a algumas das características. Os ambientes são constituídos de cilindros de diferentes dimensões, dispostos em vários arranjos espaciais, conduzindo a diferentes impressões perceptivas ao longo da navegação. Os cenários são descritos a seguir:

- Cenário Básico: ambiente desenvolvido para apresentação em detalhe da metodologia, suas características de inicialização e definição de parâmetros do sistema, ver figura 5.5.

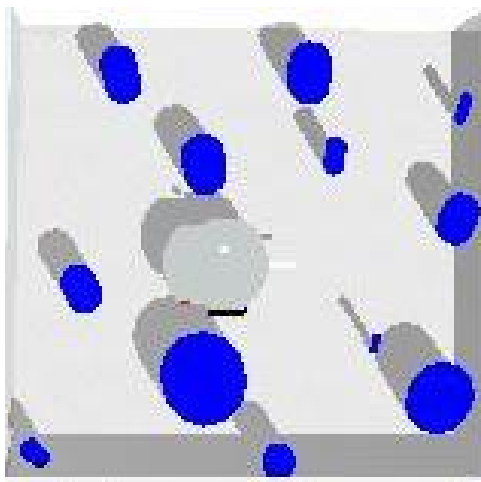


Figura 5.5: Cenário Base.

- Cenário Correspondência: ambiente povoado de elementos cilíndricos de igual formato, dispostos de maneira uniforme no cenário, apresentando elevada correspondência perceptiva em diferentes situações espaciais, ver figura 5.6.
- Cenário Escalabilidade: ambiente maior que o Cenário Básico, apresentando obstáculos dispostos de forma mais espaçada. As suas amplas dimensões permitirão

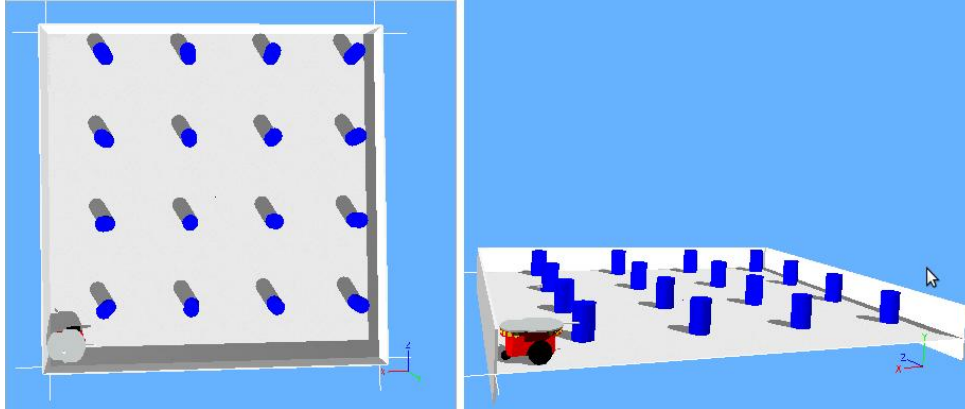


Figura 5.6: Cenário Correspondência

um maior espaço de configuração livre, conduzindo a uma navegabilidade livre de obstáculos, ver figura 5.7.

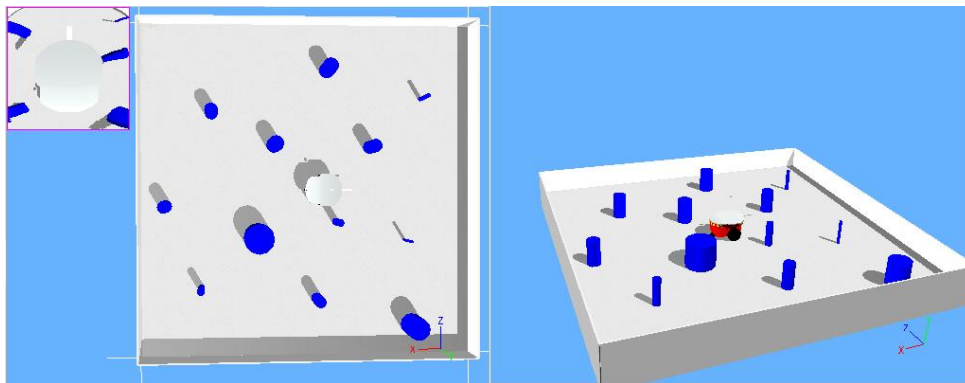


Figura 5.7: Cenário Escalabilidade

- Cenário Dinâmico: foi desenvolvido um cenário cujos elementos são removidos do ambiente ao longo do tempo, ver figura 5.8, conduzindo a diferentes percepções sensoriais em um mesmo local espacial ao longo do tempo.

A seguir será apresentada a metodologia de uso do sistema, com detalhes associados a suas ferramentas, parâmetros e inicialização. Os cenários serão utilizados para verificar várias características da proposta, tais como tolerância a ruídos, correspondência de pontos, escalabilidade, dinâmica do sistema, dentre outras.

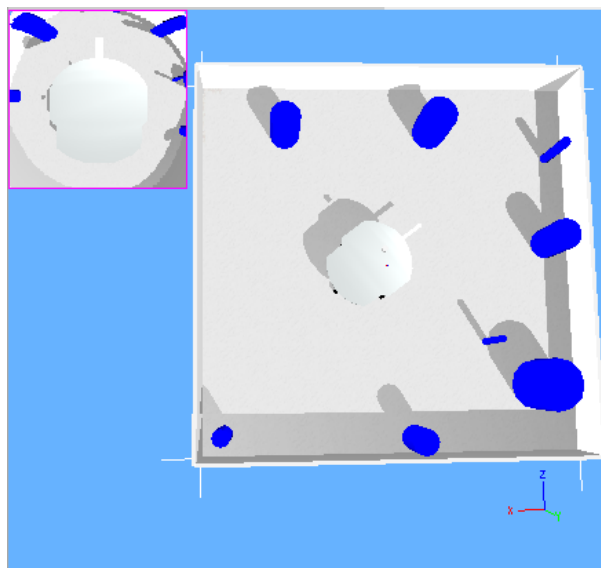


Figura 5.8: Cenário Dinâmico - ocorreu a remoção de alguns obstáculos, o ambiente completo é semelhante ao cenário base, conforme Figura 5.5.

5.3 Cenário Base

Utilizou-se o Cenário Base para apresentar as ferramentas desenvolvidas e o conjunto de parâmetros a serem definidos, bem como os critérios de avaliação da qualidade do mapa. Para tal, foi estabelecida uma missão de navegação via operação remota apresentada na Figura 5.9. Tal navegação gerou um total de 335 amostras sensoriais, decorrentes dos sinais da câmera e do GPS.

5.3.1 Medidas de Avaliação

Na literatura encontram-se alguns métodos de avaliação da qualidade de Mapas Auto-organizáveis [Wu and Takatsuka, 2005]. Neste trabalho serão utilizados as medidas associadas ao Erro de Quantização e ao Erro Topográfico, descritos a seguir.

Erro de Quantização

Fornece uma medida de erro associada ao quão um padrão de entrada é bem representado pelo seu neurônio vencedor (BMU). O cálculo dessa avaliação é dado a partir da Equação 5.1.

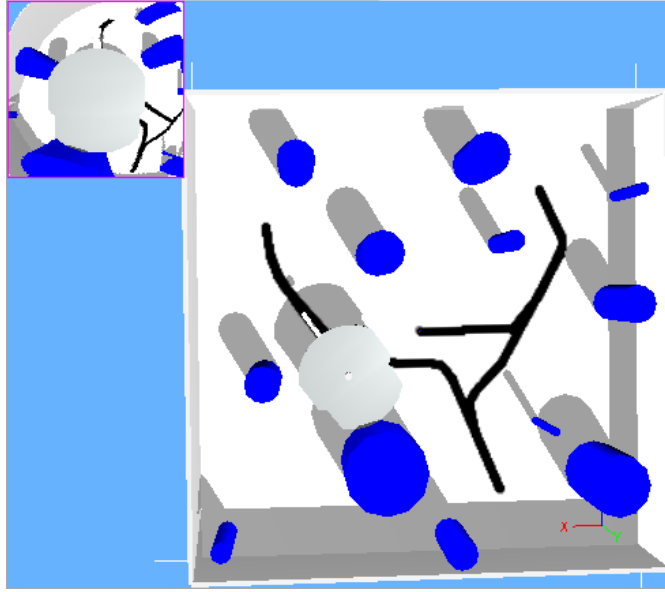


Figura 5.9: Trajeto percorrido pelo Robô

$$\varepsilon_q = \frac{E}{N} \quad (5.1)$$

onde N é o número de entradas e E é obtido pelo Equação 5.2.

$$E = \sum_{i=1}^N d(X_i, w_k) \quad (5.2)$$

onde d é a distância euclidiana entre um vetor X_i do espaço de entrada e o vetor de pesos w_k do neurônio vencedor k para a respectiva entrada.

Erro Topográfico

A medida da continuidade do mapeamento é denominada de erro topográfico ε_t . Esta é obtida considerando-se a adjacência dos campos receptivos e a proporção dos vetores de treinamento, os quais indicam uma descontinuidade local do mapeamento. O erro topográfico é obtido pela soma dos erros topográficos locais para todos os vetores da amostra de treinamento conforme Equação 5.3.

$$\varepsilon_t = \frac{1}{N} \sum_{i=1}^N u(X_i) \quad \text{onde } u(X_i) = \begin{cases} 1, & \text{se } n_k \text{ e } n_j \text{ não são adjacentes} \\ 0, & \text{caso contrário} \end{cases} \quad (5.3)$$

onde $\frac{1}{N}$ fornece uma idéia de proporção da vizinhança local que é mapeada corretamente, sem descrever o tipo de mapeamento incorreto, uma vez que dois vetores próximos no espaço de entrada podem ser mapeados por neurônios que distem de apenas uma unidade no espaço de saída ou que estejam em cantos opostos do mapa [Kiviluoto, 1996].

5.3.2 Parâmetros de Inicialização

Como apresentado no Capítulo 3, os Mapas Auto-organizáveis apresentam um conjunto de parâmetros a serem definidos empiricamente, destes podem-se citar as dimensões da rede, taxas de aprendizado, topologia da vizinhança, para citar alguns^{1 2}. Apresenta-se a seguir um conjunto de testes que conduzem a escolha de parâmetros adequados para as simulações realizadas.

Mapa Perceptivo Foram realizados testes com *i.* diferentes dimensões de redes (10x10, 50x50 e 100x100); *ii.* diferentes taxas de aprendizado (η)(0.05, 0.1, 0.2 e 0.5) e diferentes topologias de vizinhança. Diferentes mapas perceptivos foram gerados, para cada qual foram calculados os Erros de Quantização e Topográficos. As Tabelas 5.1, 5.2 e 5.3 apresentam os resultados das diversas simulações.

Mapa Espacial: para o mapeamento espacial da navegação no Cenário Básico foram realizados testes com diferentes taxas de aprendizagem e limiar de inserção de nodos, cujos resultados são apresentados na tabela 5.4.

Escolha dos Parâmetros: calculou-se a média e o desvio padrão, de uma amostras de dados, entre os valores obtidos e realizou-se a escolha dos parâmetros que obtiveram a menor média de erro de quantização e menor erro topográfico. Portanto, para a determinação dos parâmetros do Mapa Perceptivo, o tamanho $m \times m = 50 \times 50$ foi escolhido, a taxa de aprendizagem $\eta = 0.5$ e a largura da vizinhança = 8. Já para o Mapa Espacial, foram escolhidos a taxa de aprendizagem $\eta = 0.5$ e o limiar de inserção= 0.005.

¹Na literatura encontram-se alguns estudos relacionados a definição de heurísticas e formalismos probabilísticos para auxílio a sua definição

²Para cada experimento foram realizados três testes, com diferentes valores de pesos iniciais.

Tabela 5.1: Resultados de Rede 10x10 para Mapa Perceptivo da Navegação no Cenário Base

m x m	η	Vizinhança	EQ	ET
10	0.1	8	0.405615	0.979104
			0.404653	0.916418
			0.411222	0.961194
			$\mu = 0.407163$	$\mu = 0.952238$
			$s = 0.003547$	$s = 0.032288$
10	0.1	16	0.410993	0.901493
			0.409142	0.844776
			0.412429	0.979104
			$\mu = 0.410854$	$\mu = 0.908457$
			$s = 0.001647$	$s = 0.067434$
10	0.05	8	0.410138	0.988060
			0.409267	0.883562
			0.410096	0.886567
			$\mu = 0.409833$	$\mu = 0.919396$
			$s = 0.000491$	$s = 0.059483$
10	0.05	16	0.408567	0.991045
			0.409903	0.880597
			0.403001	0.892537
			$\mu = 0.407157$	$\mu = 0.921393$
			$s = 0.003660$	$s = 0.060615$
10	0.2	8	0.405944	0.985075
			0.410177	0.919403
			0.411979	0.841791
			$\mu = 0.409366$	$\mu = 0.915422$
			$s = 0.003098$	$s = 0.071724$
10	0.2	16	0.408369	0.901493
			0.411662	0.847761
			0.407825	0.916418
			$\mu = 0.409285$	$\mu = 0.888557$
			$s = 0.002076$	$s = 0.361101$
10	0.5	8	0.398206	0.143284
			0.399312	0.119403
			0.394621	0.122388
			$\mu = 0.397379$	$\mu = 0.128358$
			$s = 0.002452$	$s = 0.013011$
10	0.5	16	0.398138	0.137313
			0.400753	0.140299
			0.400808	0.161194
			$\mu = 0.399899$	$\mu = 0.146268$
			$s = 0.001525$	$s = 0.013011$

Tabela 5.2: Resultados de Rede 50x50 para Mapa Perceptivo da Navegação no Cenário Base

m x m	η	Vizinhança	EQ	ET
50	0.1	8	0.386609	0.994030
			0.364711	0.890774
			0.385065	0.894030
			$\mu = 0.378795$	$\mu = 0.892944$
			$s = 0.012221$	$s = 0.058697$
50	0.1	16	0.384558	0.895636
			0.390733	0.882090
			0.392339	0.873137
			$\mu = 0.389210$	$\mu = 0.883621$
			$s = 0.004107$	$s = 0.011327$
50	0.05	8	0.387614	0.891045
			0.385094	0.891697
			0.382323	0.897015
			$\mu = 0.385001$	$\mu = 0.893252$
			$s = 0.002646$	$s = 0.003274$
50	0.05	16	0.387392	0.897015
			0.386305	0.897604
			0.385150	0.891045
			$\mu = 0.386282$	$\mu = 0.895221$
			$s = 0.001121$	$s = 0.003628$
50	0.2	8	0.389294	0.887015
			0.386083	0.870149
			0.383630	0.897017
			$\mu = 0.386335$	$\mu = 0.888660$
			$s = 0.002840$	$s = 0.013579$
50	0.2	16	0.384441	0.873134
			0.393319	0.897437
			0.394637	0.885761
			$\mu = 0.390799$	$\mu = 0.885444$
			$s = 0.005545$	$s = 0.012154$
50	0.5	8	0.379331	0.123749
			0.377574	0.122975
			0.380987	0.120047
			$\mu = 0.379297$	$\mu = 0.122257$
			$s = 0.001706$	$s = 0.001952$
50	0.5	16	0.381475	0.143881
			0.382829	0.136866
			0.387956	0.149104
			$\mu = 0.384086$	$\mu = 0.143283$
			$s = 0.003418$	$s = 0.006140$

Tabela 5.3: Resultados de Rede 100x100 para Mapa Perceptivo da Navegação no Cenário Base

m x m	η	Vizinhança	EQ	ET
100	0.1	8	0.378075	0.997435
			0.376434	0.990853
			0.377889	0.974395
			$\mu = 0.377466$	$\mu = 0.987561$
			$s = 0.000898$	$s = 0.011867$
100	0.1	16	0.373997	0.984533
			0.378104	0.984734
			0.384405	0.980066
			$\mu = 0.378835$	$\mu = 0.983111$
			$s = 0.005242$	$s = 0.002638$
100	0.05	8	0.378970	0.990775
			0.381290	0.988756
			0.875508	0.980646
			$\mu = 0.371718$	$\mu = 0.986725$
			$s = 0.002909$	$s = 0.005361$
100	0.05	16	0.372853	0.988704
			0.378297	0.990574
			0.387441	0.988453
			$\mu = 0.379530$	$\mu = 0.989243$
			$s = 0.007371$	$s = 0.001158$
100	0.2	8	0.389773	0.899435
			0.389488	0.890674
			0.390677	0.899452
			$\mu = 0.389979$	$\mu = 0.896520$
			$s = 0.000620$	$s = 0.005063$
100	0.2	16	0.380874	0.879974
			0.378740	0.883567
			0.379366	0.897331
			$\mu = 0.379660$	$\mu = 0.886957$
			$s = 0.001096$	$s = 0.009161$
100	0.5	8	0.373453	0.216774
			0.388497	0.223551
			0.388563	0.216834
			$\mu = 0.383504$	$\mu = 0.219053$
			$s = 0.008704$	$s = 0.003895$
100	0.5	16	0.399743	0.244356
			0.398055	0.245415
			0.388810	0.256647
			$\mu = 0.395536$	$\mu = 0.248806$
			$s = 0.005885$	$s = 0.006811$

Tabela 5.4: Estudo dos Parâmetros do Mapa Espacial no Cenário Base

η	Limiar de Inserção	EQ	ET
0.1	0.01	0.006727	0.032836
		0.006412	0.023881
		0.006461	0.053731
		$\mu = 0.006533$	$\mu = 0.036816$
		$s = 0.000169$	$s = 0.015317$
0.1	0.005	0.005590	0.017910
		0.005020	0.023881
		0.005370	0.041791
		$\mu = 0.005320$	$\mu = 0.027860$
		$s = 0.000028$	$s = 0.012427$
0.05	0.01	0.007213	0.080597
		0.006841	0.065672
		0.008208	0.080597
		$\mu = 0.007420$	$\mu = 0.075622$
		$s = 0.000706$	$s = 0.008616$
0.05	0.005	0.005967	0.035821
		0.005053	0.038806
		0.005360	0.050746
		$\mu = 0.005460$	$\mu = 0.041791$
		$s = 0.000465$	$s = 0.007897$
0.2	0.01	0.007242	0.032836
		0.006587	0.032837
		0.007179	0.023881
		$\mu = 0.007002$	$\mu = 0.029851$
		$s = 0.000361$	$s = 0.005170$
0.2	0.005	0.005201	0.053731
		0.005882	0.047761
		0.005546	0.041791
		$\mu = 0.005543$	$\mu = 0.047761$
		$s = 0.000340$	$s = 0.005970$
0.5	0.01	0.006971	0.071642
		0.006458	0.089552
		0.006096	0.076720
		$\mu = 0.006508$	$\mu = 0.079304$
		$s = 0.000439$	$s = 0.009232$
0.5	0.005	0.005479	0.029851
		0.005769	0.023881
		0.004217	0.023782
		$\mu = 0.005155$	$\mu = 0.025838$
		$s = 0.000825$	$s = 0.003475$
1.0	0.01	0.005334	0.208955
		0.005683	0.283582
		0.005930	0.211940
		$\mu = 0.005649$	$\mu = 0.234825$
		$s = 0.000299$	$s = 0,042250$
1.0	0.005	0.004998	0.038806
		0.004891	0.020896
		0.005016	0.035821
		$\mu = 0.004968$	$\mu = 0.031841$
		$s = 0,000067$	$s = 0,009595$

5.3.3 Utilização dos Mapas Perceptivos e Espacial

Os parâmetros definidos na seção anterior, durante os experimento no Cenário Base, conduziram às representações espaciais e perceptivas apresentadas na figura 5.10. O mapa superior (mapa perceptivo) é visualizado utilizando a matriz de distâncias unificadas. Já o mapa inferior (mapa espacial) é a representação da estrutura celular crescente. As conexões existentes entre os mapas representam que a informação perceptiva está na posição (x, y) do mapa espacial. A tabela 5.5 apresenta o desempenho da representação obtida.

Mapa	EQ	ET
Espacial	0,004784	0,029737
Perceptivo	0,385198	0,397015

Tabela 5.5: Desempenho do Sistema para o Cenário Base

O mapa espacial final apresentou 182 nodos. Enquanto que o mapa perceptivo apresentou 9 grupos principais de percepção, conforme figura 5.10.

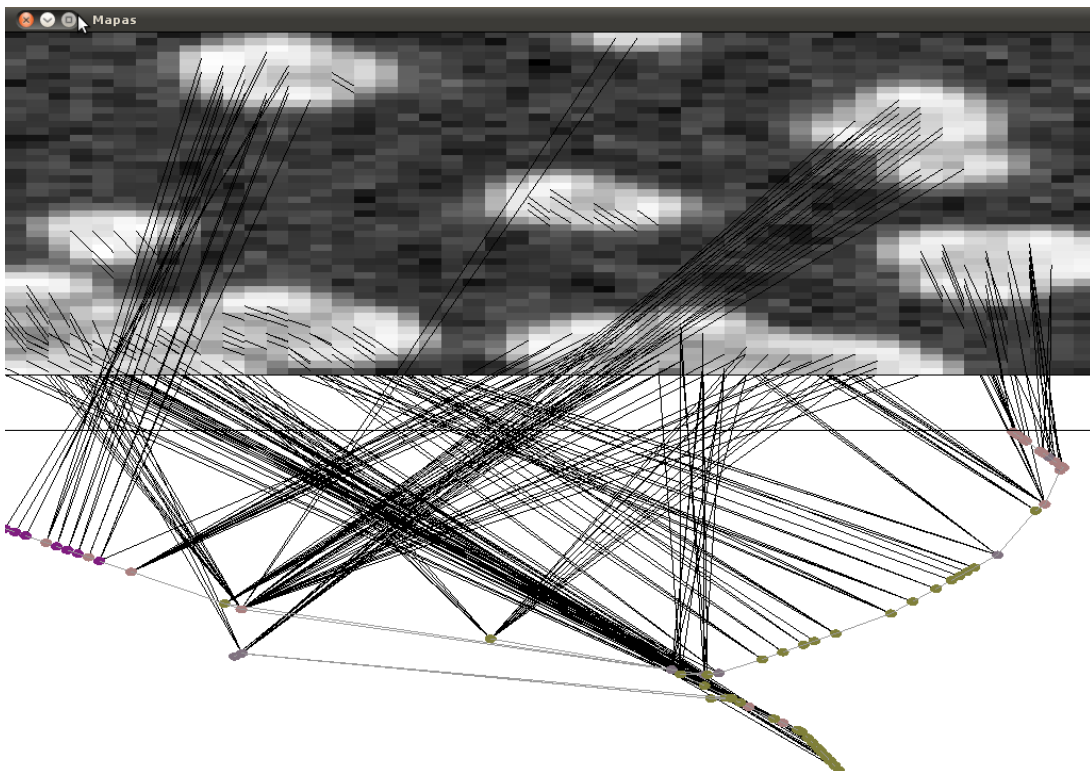


Figura 5.10: Mapa do Cenário Base utilizando arestas para a integração entre os mapas.

Para facilitar a representação da integração entre os mapas, além das conexões, o sistema também possibilita outros dois modos de relacionar os mapas, como mencionado no Capítulo 4, conforme Figuras 5.11 e 5.12.

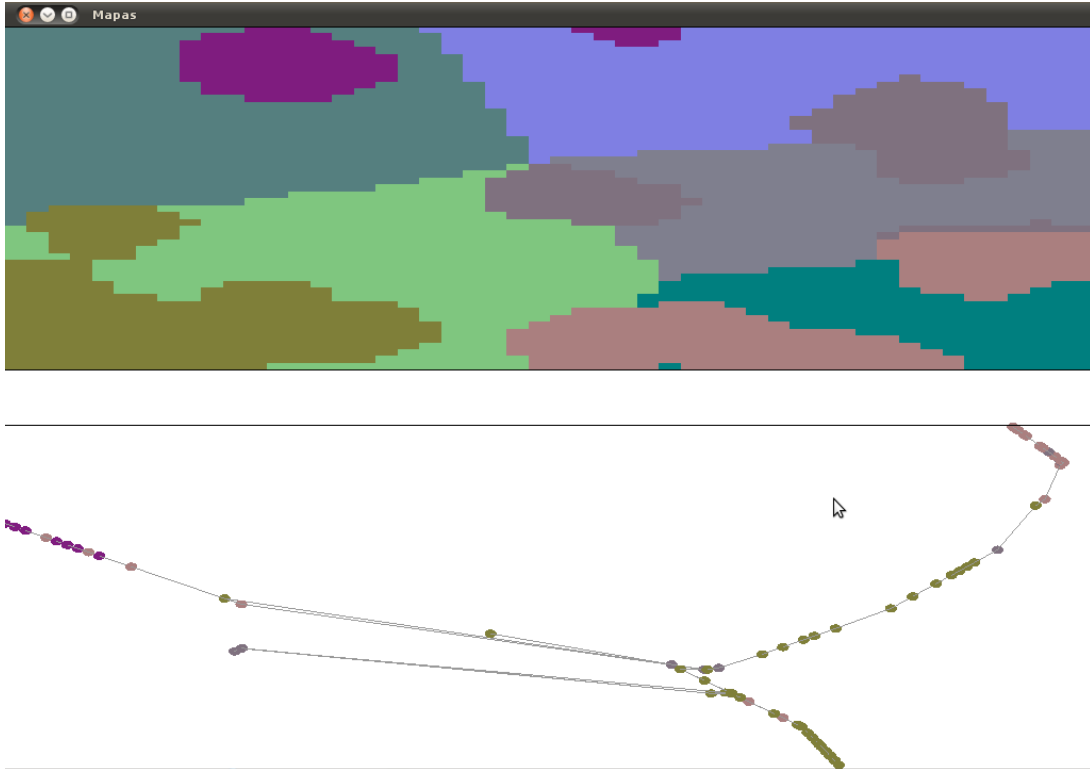


Figura 5.11: Mapa do Cenário Base utilizando cores, no qual os nodos do mapa espacial representam a cor do grupo do mapa perceptivo que melhor representa a informação encontrada na posição do robô.

Na Figura 5.11, os nodos do mapa espacial são apresentados pelo método padrão; e na Figura 5.12, os nodos espaciais associados a mais de um grupo perceptivo são coloridos com diferentes cores (uma cor para cada um desses grupos). Percebe-se tal situação sensorial dúbia na figura 5.12, onde, próximo a bifurcação, os nodos espaciais apresentam mais de uma representação sensorial, devido a rotação do robô para adotar a segunda ramificação (no mesmo local diferentes percepções decorrentes da modificação da direção do robô).

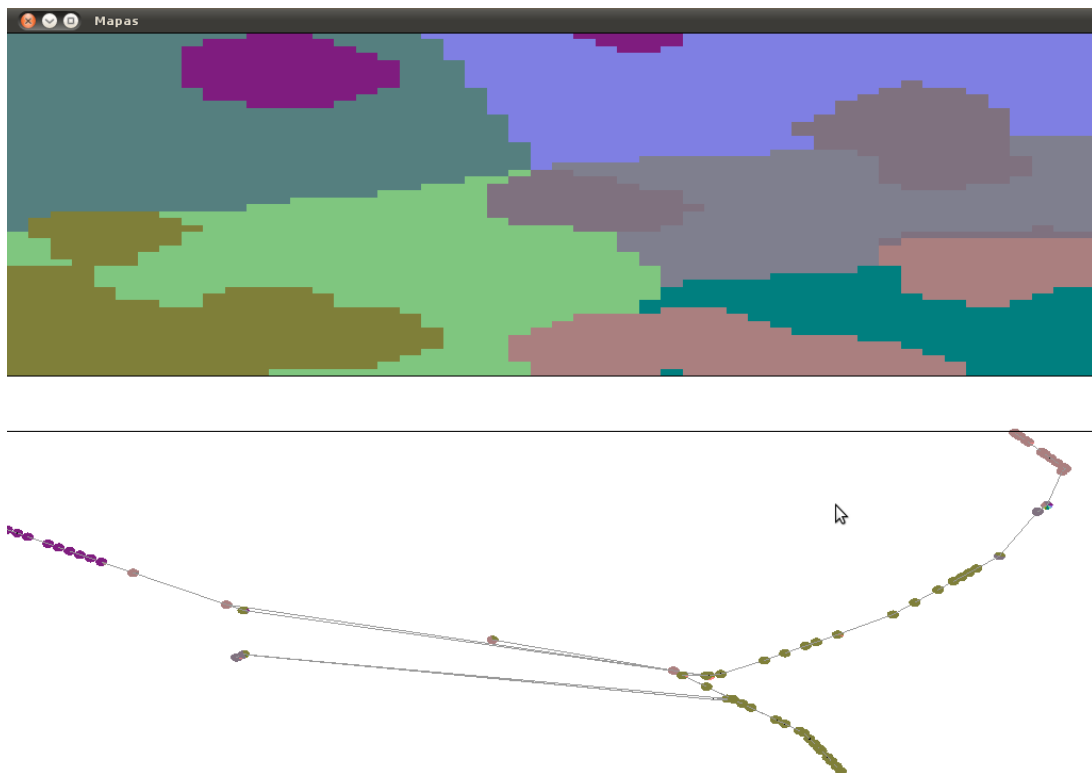


Figura 5.12: Mapa do Cenário Base utilizando cores, no qual os nodos do mapa espacial representam a(s) cor(es) do grupo(s) do mapa perceptivo que representa a informação encontrada na posição do robô.

5.4 Cenário de Correspondência

Por apresentar uma disposição uniforme de obstáculos semelhantes, este cenário é propício a existência de muitas impressões sensoriais semelhantes. Ao total foram capturadas 605 amostras sensoriais, a tabela 5.6 apresenta o desempenho do sistema.

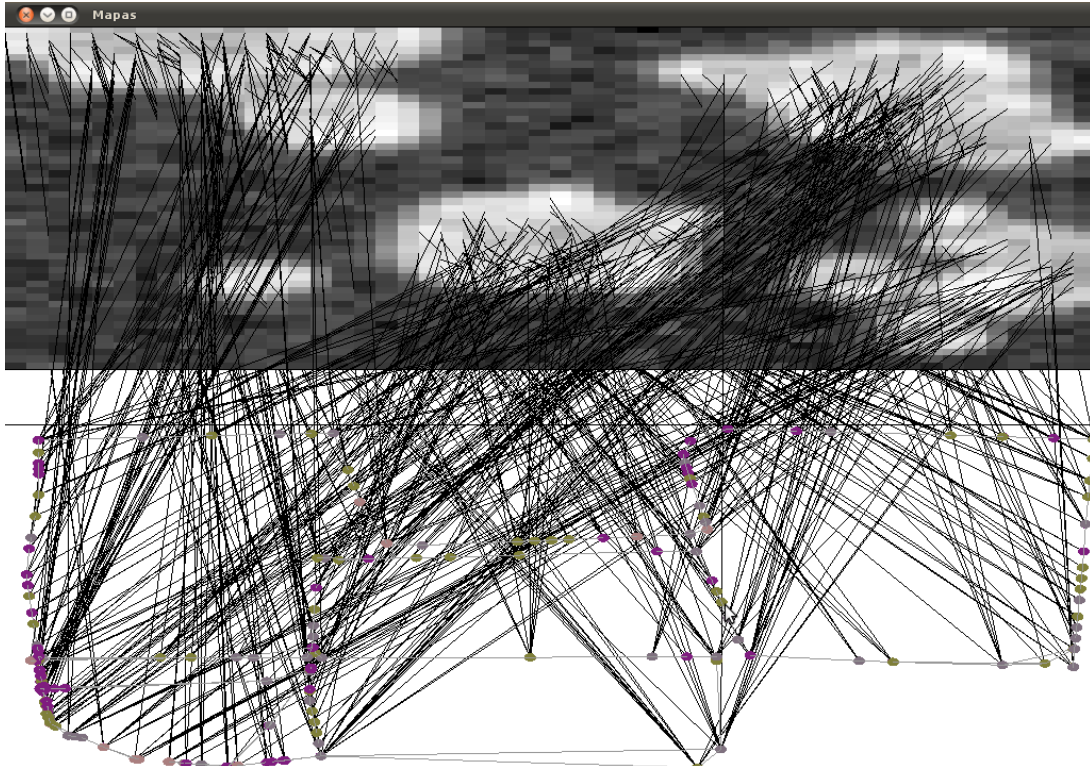


Figura 5.13: Mapa do Cenário de Correspondência

Mapa	EQ	ET
Espacial	0,013316	0,029752
Perceptivo	0,416541	0,345455

Tabela 5.6: Desempenho do Sistema para o Cenário Correspondência

Para os experimentos realizados, o robô percorreu a trajetória apresentada na Figura 5.14. As impressões espaciais foram representadas em 480 nodos. Percebe-se que o mapa perceptivo condensa a quantidade similar de percepções em um número reduzido de 5 grupos, ver Figura 5.13, que são conectados aos diferentes nodos do Mapa Espacial.

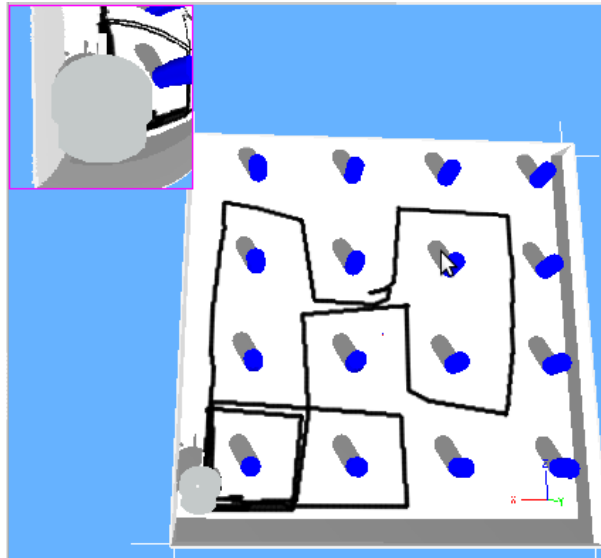


Figura 5.14: Trajeto percorrido pelo robô.

5.5 Cenário de Escalabilidade

Este cenário congrega um ambiente de maiores dimensões que os anteriores. Os obstáculos estão mais espaçados permitindo ao robô navegar mais afastado dos objetos, conforme pode ser visto na trajetória apresentada na figura 5.15.

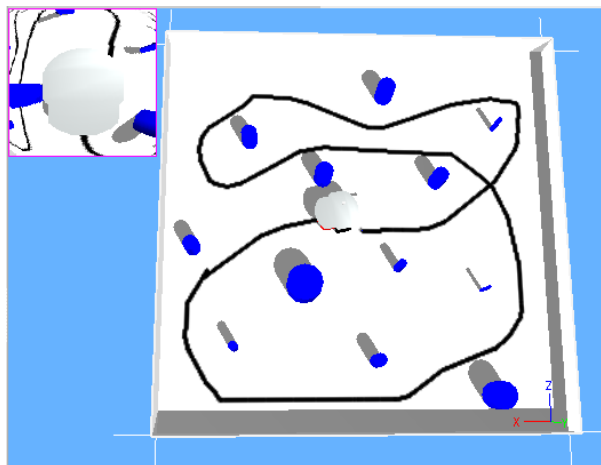


Figura 5.15: Trajeto percorrido pelo robô.

Foram apresentadas ao sistema 868 amostras e obteve-se o desempenho demonstrado na tabela 5.7. O mapa espacial convergiu para 696 nodos. Apesar do ambiente ser maior, foram obtidos no mapa perceptivo apenas 5 grupos, ver Figuras 5.16 (uma classe por nodo espacial) e 5.17 (multi-classes). Como era de se esperar, o aumento da escalabilidade do

ambiente proporcionou um maior deslocamento do robô e isto implicou em um incremento no número de nodos do mapa espacial. Por outro lado, devido a abrangência limitada da visão omnidirecional e a navegação mais afastada dos obstáculos, durante a maioria do trajeto, o robô não enxergava nenhum obstáculo, diminuindo a diversidade de situações sensoriais.

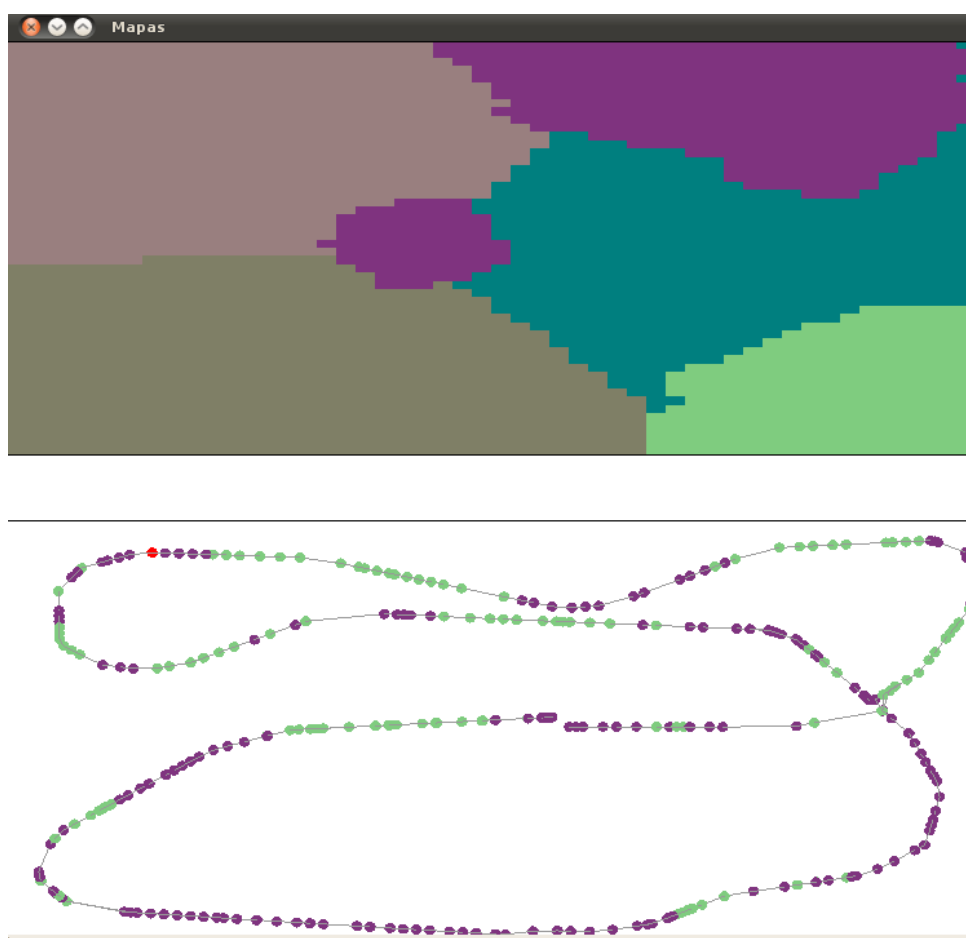


Figura 5.16: Mapa do Cenário de Escalabilidade

Mapa	EQ	ET
Espacial	0,006345	0,003456
Perceptivo	0,252695	0,095622

Tabela 5.7: Desempenho do Sistema para o Cenário Escalabilidade

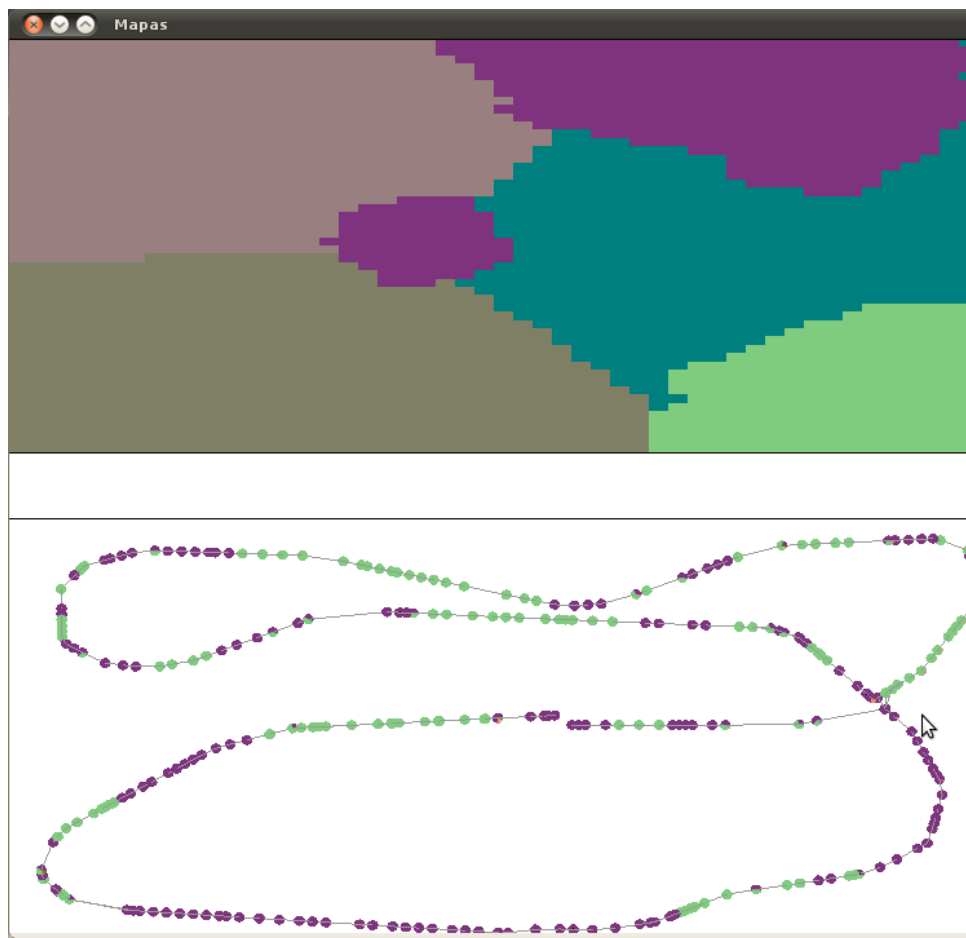


Figura 5.17: Mapa do Cenário de Escalabilidade

5.6 Cenário de Ambientes Dinâmicos

Esse ambiente é idêntico ao cenário base, no entanto, são retirados, em determinado momento, alguns obstáculos. Nesse teste, são apresentadas duas situações: primeiramente, o robô percorre o ambiente com todos os obstáculos presentes; após, alguns desses são removidos. O cenário resultante da remoção dos obstáculos é apresentado na Figura 5.8.

Para o ambiente completo, foram apresentadas 276 amostras, o sistema criou 77 nodos no mapa espacial e, no mapa perceptivo, formaram-se 7 grupos. As figuras 5.18 e 5.19 apresentam os mapas perceptivos e espacial antes e após a remoção. Percebe-se a adaptação do mapa perceptivo à remoção dos obstáculos, manifestando-se através da redução do número de grupos do Mapa Perceptivo. Entretanto, os erros apresentam-se maiores, tendo em vista que, em sua implementação atual, o mapa é o resultado da totalidade das impressões sensoriais obtidas ao longo de toda a navegação (etapa com todos os ob-

stáculos e após a retirada de alguns), ou seja, a redução no número de classes, resultantes da dinâmica da entrada, implicou em um maior erro de representação (ver tabela 5.8 em comparação com a tabela 5.5), isto deve-se ao fato de que os padrões retirados, ainda fazem parte do universo de aprendizado.

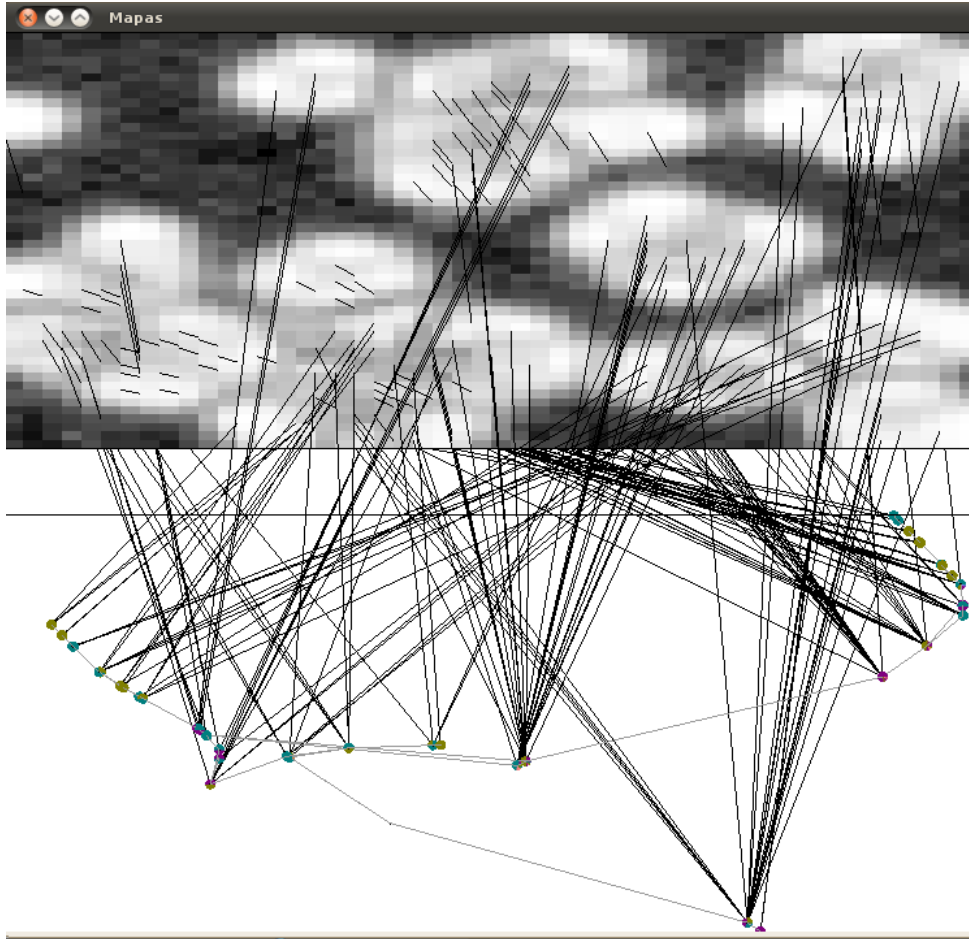


Figura 5.18: Mapa do Cenário Dinâmico Completo

Mapa	EQ	ET
Espacial	0,018236	0,126482
Perceptivo	0,517431	0,464427

Tabela 5.8: Desempenho do Sistema para o Cenário Dinâmico

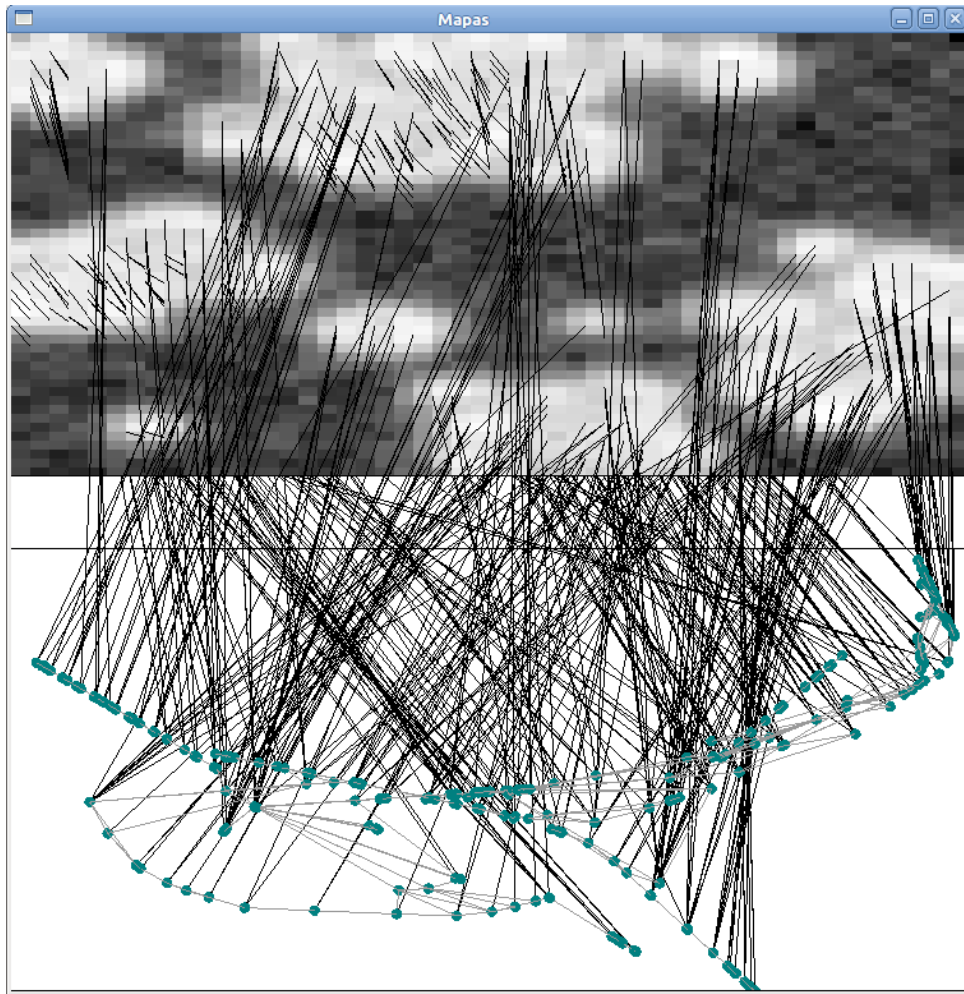


Figura 5.19: Mapa do Cenário Dinâmico com Retirada dos Obstáculos

5.7 Considerações Finais

Neste capítulo, foram mostrados os resultados obtidos com a aplicação do modelo proposto.

Inicialmente foi apresentada a atual implementação do sistema com suas principais bibliotecas. O ambiente de experimentação webots foi também detalhado. A partir deste, foram gerados quatro principais cenários.

O cenário base foi utilizado para apresentar a metodologia de uso da proposta, bem como sua sensibilidade a diferentes escolhas paramétricas e inicializações. Os demais cenários propiciaram a verificação de algumas características da proposta. Percebe-se que a separação semântica entre espaço e demais níveis perceptivos permite, além de diferentes abordagens específicas as suas entradas (por exemplo Kohonen para perceptivo

visual e GCS para espacial), o tratamento adequado de situações de correspondências. Também o aumento de escalabilidade espacial, não reflete-se em aumento de complexidade perceptiva, caso a diversidade das impressões sensoriais mantenham-se constantes. As características de aprendizado e adaptação dos Mapas Topológicos garantem a tolerância necessária a imprecisões sensoriais, além de serem uma promissora abordagem para a detecção e tratamento de mudanças temporais no ambiente.

Ressalta-se que mesmo não tendo sido validada em toda a sua potencialidade neste trabalho (testes com diferentes níveis de abstração e envolvendo outros sensores não foram realizados pela restrição temporal), o sistema desenvolvido e a plataforma disponibilizada já habilitam seu uso para estas e outras diferentes aplicações que possam vir a ocorrer. Por exemplo, os diferentes níveis de Mapas Perceptivos podem ser utilizados diretamente para a especificação de missões em diferentes níveis de abstração. A integração destes com o Mapa Espacial, fornecem o vínculo necessário para a obtenção das trajetórias para a sua execução.

Capítulo 6

Conclusão e Trabalhos Futuros

Neste trabalho foi desenvolvido um método de mapeamento robótico utilizando mapas auto-organizáveis, dispostos em níveis semânticos: espacial e perceptivos. Além disso desenvolveu-se uma plataforma de simulação utilizando o simulador Webots, que auxiliou na execução dos testes apresentados.

Foi apresentado um panorama geral do problema de Mapeamento Robótico, os principais conceitos e paradigmas associados a Localização e SLAM. Foram formalizados os problemas da síntese de ambientes e apresentaram-se as principais técnicas utilizadas na bibliografia, para resolver este problema.

Um enfoque maior foi dado aos Mapas Topológicos, uma vez que estes foram utilizados nesse trabalho. Os Mapas Topológicos apresentam como vantagens, em relação as outras abordagens, a robustez e simplicidade principalmente se o ambiente tratado for dinâmico. Então, relatou-se um conjunto de trabalhos correlatos sobre mapeamento topológicos e mapeamento híbrido, de forma a situar a proposta desse trabalho. Além disso, foram descritos os principais conceitos associados a Mapas Auto-organizáveis. Então, foram apresentadas as principais abordagens para a sua implementação.

Com base na revisão bibliográfica realizada, apresentou-se a metodologia proposta, para tratar os problemas de mapeamento robótico. Utilizou-se uma abordagem baseada em Mapas Auto-organizáveis para a representação das impressões do ambiente, dispostos em camadas com diferentes significados taxonômicos. Diferentemente de outros trabalhos que utilizam mapas auto-organizáveis para mapeamento topológico espacial e perceptivo, a presente proposta estabeleceu explicitamente os diferentes módulos da arquitetura,

Mapa Espacial e Mapas Perceptivos (as diferentes naturezas das informações a serem mapeadas). As informações espaciais são armazenadas no Mapa Espacial, enquanto que cada tipo de sensor (ou nível abstrato de informação) possui seu próprio Mapa Perceptivo.

A proposta foi utilizada em experimentos realizados em diferentes tipos de cenários. Uma plataforma de simulação utilizando a ferramenta webots foi desenvolvida e integrada ao sistema implementado. O desempenho do sistema nos diferentes testes realizados indicam que a abordagem é viável, e suas extensões são promissoras.

A abordagem proposta buscou averiguar a possibilidade de tratar o problema de mapeamento robótico através do uso de diferentes mapas associados a informações de diversas naturezas, analisando seu uso frente a alguns dos seus principais problemas:

- escalabilidade - a divisão em diferentes níveis reduz a complexidade associada a representação em nodos, restringindo o aumento escalar apenas aos níveis/abstrações relacionados. Também o uso de Mapas Auto-organizáveis possibilitam o agrupamento automático de informações semelhantes, favorecendo a compactação.
- correspondência - impressões sensoriais idênticas são associadas a diferentes locais espaciais através das arestas de integração entre os mapas.
- dinamismo e tratamento de ruídos - a utilização de SOMs para a representação das informações sensoriais traz consigo as características de adaptabilidade e aprendizado destes sistemas.
- navegabilidade - os diferentes níveis de Mapas Perceptivos associados à possibilidade de obtenção de trajetórias a partir do Mapa Espacial, permitem a especificação de missões de diferentes níveis de abstração.

Mesmo que restrito a um conjunto limitado de experimentos, a utilização da proposta mostrou-se promissora, conduzindo a um conjunto de possíveis extensões. Dentre estas, pode-se citar:

- Realizar um conjunto maior de testes, principalmente associados a multi-sensores;
- Aplicar a proposta a robôs reais existentes no laboratório, com uso de câmera e kinect;

- Explorar melhor os conceitos envolvidos com as arestas de ligação entre os mapas, de modo a repercutir na adaptação destes em função dos níveis de confiança das informações manipuladas. Bem como, desenvolver algoritmos para criação e deleção dessas arestas;
- Explorar a hierarquia semântica da proposta, por exemplo, realizando a clusterização do mapa perceptivo de modo que um dado conjunto de impressões similares sejam clusterizados e entendidos de modo a determinar informação de mais alto nível;
- Avaliar as diferenças geradas no mapeamento pelas redes como GNG e GWR, além de estabelecer uma adaptação de redes existentes para o problema de mapeamento robótico;
- Realizar uma comparação mais aprofunda com outros algoritmos de mapeamento utilizando mapas auto-organizáveis também pode ser efetuado, permitindo assim uma validação mais completa do modelo proposto.

Bibliografia

- [Bay et al., 2006] Bay, H., Tuytelaars, T., and Gool, L. J. V. (2006). SURF: Speeded up robust features. In Leonardis, A., Bischof, H., and Pinz, A., editors, *European Conference on Computer Vision*, volume 3951 of *Lecture Notes in Computer Science*, pages 404–417. Springer.
- [Bensalem et al., 2009] Bensalem, S., Gallien, M., Ingrand, F., Kahloul, I., and Thanh-Hung, N. (2009). Designing autonomous robots - toward a more dependable software architecture. *IEEE Robotics and Automation Magazine*, 16(1):67–77.
- [Blanco et al., 2008] Blanco, J., Fernández-Madrigal, J., and González, J. (2008). Toward a unified bayesian approach to Irbid Metric-Topological SLAM. *IEEE Transactions on Robotics*, 24:259–270.
- [Blanco et al., 2009] Blanco, J., Fernández-Madrigal, J., and González, J. (2009). Toward a unified bayesian approach to Irbid Metric-Topological SLAM. *Robotics and Autonomous Systems*, 57:64–74.
- [Borenstein et al., 1996] Borenstein, J., Everett, B., and I. Feng (1996). Navigating mobile robots: Systems and techniques. *A. K. Peters Ltd.*
- [Botelho et al., 2009] Botelho, S. S. C., Drews Jr, P. L. J., Figueiredo, M., Haffele, C., and Oliveira, G. L. (2009). Appearance-based odometry and mapping with feature descriptors for underwater robots. *Journal of the Brazilian Computer Society*, 15:47–54.
- [Calvo and Figueiredo, 2003] Calvo, R. and Figueiredo, M. (2003). Extensão da lei de Hebb e aprendizagem por reforço em redes neurais aplicadas à sistema de navegação

- autônoma. In *XXIII Congresso da Sociedade Brasileira de Computação, Encontro Nacional de Inteligência Artificial*, Campinas, Brasil.
- [Castellanos et al., 2007] Castellanos, J., Martinez-Cantin, R., Tardós, J., and Neira (2007). Robocentric map joining: Improving the consistency of EKF-SLAM. *Robotics and Autonomous Systems*, 55:21–29.
- [Cazangi and Figueiredo, 2002] Cazangi, R. and Figueiredo, M. (2002). Simultaneous emergence of conflicting basic behaviors and their coordination in an evolutionary autonomous navigation system. In *Proceedings of 2002 Congress on Evolutionary Computation*, Hawaii, USA.
- [Choset et al., 2005] Choset, H., Lynch, K., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L., and Thrun, S. (2005). *Principles of Robot Motion: Theory, Algorithms and Implementation*. MIT Press.
- [Crestani et al., 2002] Crestani, P., Figueiredo, M., and Zuben, F. (2002). A hierarchical neuro-fuzzy approach to autonomous navigation. In *Proceedings of 2002 Congress on Evolutionary Computation*, Hawaii, USA.
- [Crowley, 1985] Crowley, J. L. (1985). Navigation for an intelligent mobile robot. *IEEE Journal on Robotics and Automation*, 1(1).
- [Drews Jr, 2007] Drews Jr, P. (2007). Robótica subaquática: Uma abordagem baseada em visão computacional aplicada a SLAM. Technical report, Monografia de Conclusão do Curso de Engenharia de Computação - Fundação Universidade Federal do Rio Grande.
- [Drews Jr et al., 2008] Drews Jr, P., Botelho, S. S. C., and Oliveira, G. L. (2008). NLMAP - visual-based self localization and mapping for autonomous underwater vehicles. In *MTS/IEEE Oceans*.
- [Durrant-Whyte and Bailey, 2006a] Durrant-Whyte, H. and Bailey, T. (2006a). Simultaneous localisation and mapping (SLAM): Part I the essential algorithms. *IEEE ROBOTICS AND AUTOMATION MAGAZINE*, 2:2006.

- [Durrant-Whyte and Bailey, 2006b] Durrant-Whyte, H. and Bailey, T. (2006b). Simultaneous localization and mapping (SLAM): Part II state of the art. *IEEE ROBOTICS AND AUTOMATION MAGAZINE*, 2:2006.
- [Elfes, 1987] Elfes, A. (1987). Sonar-based real-world mapping and navigation. *IEEE Journal of Robotics and Automation*, 3(3):249–265.
- [Elfes, 1989] Elfes, A. (1989). Using occupancy grids for mobile robot perception and navigation. *Computer*, 22:46–57.
- [Fritzke, 1993a] Fritzke, B. (1993a). Growing cell structures - a self-organizing network for unsupervised and supervised learning. Technical report, University of California.
- [Fritzke, 1993b] Fritzke, B. (1993b). Kohonen feature maps and growing cell structures - a performance comparison. In *Advances in Neural Information Processing Systems 5*, pages 123–130, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- [Fritzke, 1995] Fritzke, B. (1995). A growing neural gas network learns topologies. In *Advances in Neural Information Processing Systems 7*, pages 625–632. MIT Press.
- [Fritzke, 1997] Fritzke, B. (1997). A self-organizing network that can follow non-stationary distributions. In Gerstner, W., Germond, A., Hasler, M., and Nicoud, J.-D., editors, *Artificial Neural Networks ICANN 97*, volume 1327 of *Lecture Notes in Computer Science*, pages 613–618. Springer Berlin / Heidelberg.
- [Gerecke, 1999] Gerecke, U. (1999). *Robot localization using self-organizing maps with evidence shifting*. Advanced Course on Artificial Intelligence.
- [Grisetti et al., 2007] Grisetti, G., Tipaldi, G., Stachniss, C., Burgard, W., and Nardi, D. (2007). Fast and accurate SLAM with Rao-Blackwellized particle filters. *Robotics and Autonomous Systems*, 55:30–38.
- [Hafner, 2000] Hafner, V. (2000). Learning places in newly explored environments. In *International Symposium for Adaptive Behavior*, pages 111–120.
- [Hafner, 2008] Hafner, V. (2008). Robots as tools for modelling navigation skills a neural cognitive map approach. In Jefferies, M. and Yeap, W.-K., editors, *Robotics and*

- Cognitive Approaches to Spatial Mapping*, volume 38 of *Springer Tracts in Advanced Robotics*, pages 315–324. Springer Berlin / Heidelberg.
- [Haykin, 2007] Haykin, S. (2007). *Redes Neurais: Princípios e Práticas*. Bookman.
- [Hodge and Austin, 2001] Hodge, J. V. and Austin, J. (2001). Hierarchical growing cell structures: TreeGCS. *IEEE Transactions on knowledge and data engineering*, 13:207–218.
- [Itti and Koch, 2001] Itti, L. and Koch, C. (2001). Computational modelling of visual attention. *Nature Review Neuroscience*, 2(3):194–203.
- [Itti et al., 1998] Itti, L., Koch, C., and Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:1254–1259.
- [Jun and Duckett, 2002] Jun, L. and Duckett, T. (2002). Learning robot behaviours with self-organizing maps and radial basis function networks. In *Second Swedish Workshop on Autonomous Robotics*, Stockholm, Sweden.
- [Kim et al., 2008] Kim, C., Sakthivel, R., and Chung, W. (2008). Unscented FastSLAM: A robust and efficient solution to the SLAM Problem. *IEEE Transactions on Robotics*, 24:808–820.
- [Kiviluoto, 1996] Kiviluoto, K. (1996). Topology preservation in self-organizing maps. *Proc. IEEE International Conference on Artificial Neural Networks*, 1:249–254.
- [Klinkenberg, 2004] Klinkenberg, R. (2004). Learning drifting concepts: Example selection vs. example weighting. *Intelligent Data Analysis*, 8:281–300.
- [Kohonen, 1982] Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69.
- [Kohonen, 2001] Kohonen, T. (2001). *Self-Organizing Maps*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

- [Koutník et al., 2006] Koutník, J., Mázl, R., and Kulich, M. (2006). Building of 3d environment models for mobile robotics using self-organization. In Runarsson, T. P., Beyer, H.-G., Burke, E. K., Guervós, J. J. M., Whitley, L. D., and Yao, X., editors, *Parallel Problem Solving from Nature*, volume 4193 of *Lecture Notes in Computer Science*, pages 721–730. Springer.
- [Kuipers and Byun, 1988] Kuipers, B. and Byun, Y. (1988). Robust qualitative method for robot spatial learning. In *Proceedings of the National Conference on Artificial Intelligence*, pages 774–779.
- [Kurz, 1996] Kurz, A. (1996). Constructing maps for mobile robot navigation based on ultrasonic range data. *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, 26(2):233–242.
- [Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91.
- [Marsland et al., 2002] Marsland, S., Shapiro, J., and Nehmzow, U. (2002). A self-organising network that grows when required. *Neural Networks - New developments in self-organizing maps*, 15(8):1041–1058.
- [Martinetz and Schulten, 1991] Martinetz, T. and Schulten, K. (1991). A Neural-Gas Network Learns Topologies. *Artificial Neural Networks*, 1:397–402.
- [Menegaz, 2009] Menegaz, M. (2009). *Aplicação da rede GTSOM para navegação de robôs móveis utilizando aprendizado por reforço*. PhD thesis, Universidade Federal do Rio Grande do Sul.
- [Miranda Neto et al., 2008] Miranda Neto, A., Zampieri, D., Victorino, A., and Fantoni, I. (2008). Self-organizing maps for environments and states mapping of an autonomous navigation system. In *International Congress of Mechanical Engineering*.
- [Moravec and Elfes, 1985] Moravec, H. P. and Elfes, A. (1985). High resolution maps from wide angle sonar. In *1985 IEEE International Conference on Robotics and Automation*, pages 116–121, St. Louis, Missouri. IEEE Computer Society Press.

- [Nehmzow, 1992] Nehmzow, U. (1992). *Experiments in competence acquisition for autonomous mobile robots*. PhD thesis, University of Edinburgh.
- [Nehmzow and Smithers, 1991] Nehmzow, U. and Smithers, T. (1991). Mapbuilding using self-organising networks in "really useful robots". In *Proceedings of the First International Conference on Simulation of Adaptive Behaviour*, pages 152–159. MIT Press.
- [Nunez et al., 2006] Nunez, P., Martín, R. V., del Toro Lasanta, J. C., Bandera, A., and Hernandez, F. S. (2006). Feature extraction from laser scan data based on curvature estimation for mobile robotics. In *International Conference in Robotics and Automation*, pages 1167–1172. IEEE.
- [Oliveira, 2010] Oliveira, J. (2010). *Um sistema integrado para navegação autônoma de robôs móveis*. PhD thesis, Universidade de São Paulo.
- [Owen and Nehmzow, 1998] Owen, C. and Nehmzow, U. (1998). Landmark-based navigation for a mobile robot. In *Proceedings of the fifth international conference on simulation of adaptive behavior on*, pages 240–245.
- [Pedrosa, 2006] Pedrosa, D. (2006). *Mapeamento de ambientes Estruturados com extração de informações Geométricas através de Dados Sensoriais*. PhD thesis, Universidade Federal do Rio Grande do Norte.
- [Russel and Norvig, 2003] Russel, S. and Norvig, P. (2003). *Inteligência Artificial*. Campus.
- [Shatkay and Kaelbling, 2002] Shatkay, H. and Kaelbling, L. P. (2002). Learning geometrically-constrained hidden markov models for robot navigation: Bridging the geometrical-topological gap. *Journal of AI Research*, 16:16–167.
- [Souza et al., 2008] Souza, A. A., Santana, A. M., Britto, R. S., Gonçalves, L. M., and Medeiros, A. A. (2008). Representation of odometry errors on occupancy grids. In *V ICINCO - International Conference on Informatics in Control, Automation and Robotics*, Funchal, Portugal.
- [Spacek, 2003] Spacek, L. (2003). Omnidirectional catadioptric vision sensor with conical mirrors. In *Towards Intelligent Mobile Robots*.

- [Steder et al., 2011] Steder, B., Rusu, R. B., Konolige, K., and Burgard, W. (2011). Point feature extraction on 3d range scans taking into account object boundaries. In *International Conference on Robotics and Automation*.
- [Thrun, 1998] Thrun, S. (1998). Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21–71.
- [Thrun, 2002] Thrun, S. (2002). Robotic mapping: A survey. In Lakemeyer, G. and Nebel, B., editors, *Exploring Artificial Intelligence in the New Millenium*, pages 163–175. Morgan Kaufmann.
- [Thrun et al., 2005] Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press.
- [Tovar et al., 2006] Tovar, B., noz Gómez, L. M., Murrieta-Cid, R., Alencastre-Miranda, M., Monroy, R., and Hutchinson, S. (2006). Planning exploration strategies for simultaneous localization and mapping. *Robotics and Autonomous Systems*, 54:314–331.
- [Vasudevan et al., 2007] Vasudevan, S., Gächter, S., Nguyen, V., and Siegwart, R. (2007). Cognitive maps for mobile robots-an object based approach. *Robotics and Autonomous Systems*, 55:359–371.
- [Vlassis et al., 1997] Vlassis, N., Papakonstantinou, G., and Tsanakas, P. (1997). Robot map building by Kohonen’s Self-Organizing Neural Networks. In *Proc. 1st Mobinet Symposium on Robotics for Health*.
- [Wu and Takatsuka, 2005] Wu, Y. and Takatsuka, M. (2005). The geodesic self-organizing map and its error analysis. In *ACSC’05*, pages 343–352.
- [Zhao and Chen, 2011] Zhao, Y. and Chen, X. (2011). Prediction-based geometric feature extraction for 2d laser scanner. *Robotics and Autonomous Systems*, 59(6):402–409.
- [Zimmer, 1995] Zimmer, U. (1995). Self-localization in dynamic environments. In *IEEE/SOFT International Workshop*.
- [Zimmer, 1996] Zimmer, U. (1996). Robust world-modelling and navigation in a real world. *NeroComputing*, 13(2):247–260.

- [Zimmer and von Puttkamer, 1994] Zimmer, U. and von Puttkamer, E. (1994). Comparing world-modelling strategies for autonomous mobile robots. In *Internationalen Wissenschaftlichen Kolloquium - IWK*, Ilmenau, Germany.