

**ROBÔS MÓVEIS ROTEADORES APLICADOS À  
CONSTRUÇÃO DE REDES AD-HOC**



ELERSON RUBENS DA SILVA SANTOS

ROBÔS MÓVEIS ROTEADORES APLICADOS À  
CONSTRUÇÃO DE REDES AD-HOC

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

ORIENTADOR: MARCOS AUGUSTO MENEZES VIEIRA

Belo Horizonte  
Fevereiro de 2015

© 2015, Elerson Rubens da Silva Santos.  
Todos os direitos reservados.

S237r Santos, Elerson Rubens da Silva  
Robôs Móveis Roteadores Aplicados à Construção de  
Redes Ad-Hoc / Elerson Rubens da Silva Santos. —  
Belo Horizonte, 2015  
xxii, 60 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de  
Minas Gerais

Orientador: Marcos Augusto Menezes Vieira

1. Computação - Teses. 2. Sistemas de comunicação  
móvel - Teses. 3. Robótica - Teses. I. Título.

CDU 519.6\*22(043)



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

## FOLHA DE APROVAÇÃO

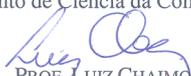
Robôs móveis roteadores aplicados à construção de redes ad-hoc

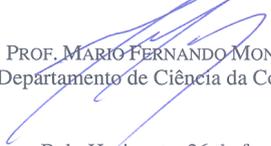
**ELERSON RUBENS DA SILVA SANTOS**

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

  
PROF. MARCOS AUGUSTO MENEZES VIEIRA - Orientador  
Departamento de Ciência da Computação - UFMG

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

  
PROF. LUIZ CHAIMOWICZ  
Departamento de Ciência da Computação - UFMG

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

Belo Horizonte, 26 de fevereiro de 2015.



*Aos meus pais*



# Agradecimentos

Agradeço aos meus pais, Nilton e Íris, e ao meu irmão, Erlon, por todo incentivo e apoio que me ofereceram durante toda minha vida e em especial durante o período do curso de mestrado.

Agradeço ao meu orientador Marcos Vieira, pelas várias oportunidades durante todo período que tive a oportunidade de trabalhar com ele. Agradeço também pelos seus ensinamentos e paciência, algo que tornou sua orientação muito agradável e útil ao meu crescimento pessoal e acadêmico.

Por último agradeço a todo pessoal do VeRLab que tive oportunidade de conhecer e trabalhar. Aos professores, Mário, Chaimo, Luiz Filipe e Douglas, pelas disciplinas que assisti e artigos publicados em parceria. Aos alunos Vinícius Graciano, Cláudio Santos, Fernando Carvalho, que me ensinaram muito de computação gráfica quando eu ainda era novo no VeRLab. E aos meus colegas mais recentes de laboratório David, Levi, Ramon, Igor, Anderson Tavares, Hector, Omar e outros colegas mais antigos Balbino, Anderson Pires, Rafael Colares, que tive o privilégio de conhecer e ter várias conversas e discussões sobre os mais diversos temas.



*“É sempre o aventureiro que consegue grandes coisas”*  
(Montesquieu)



# Resumo

A Robótica Móvel pode ajudar a humanidade em várias tarefas, incluindo auxiliar no resgate em desastres. Em ambientes sem infra-estrutura de comunicação, é importante estabelecer uma rede de comunicação ad-hoc para membros da equipe de resgate poderem se comunicar. Neste trabalho é investigado o problema de alocar um conjunto de robôs roteadores para criar uma rede ad-hoc interconectando um conjunto de clientes estáticos, permitindo que esses comuniquem entre si. A abordagem proposta é composta por duas fases. Primeiro uma árvore de Steiner é construída interconectando os clientes. Essa árvore é construída mesmo na presença de obstáculos e, por definição, é a menor árvore Euclidiana interconectando o conjunto de clientes. Na segunda fase, cada robô executa uma máquina de estados, permitindo que a rede ad-hoc seja instalada autonomamente. A abordagem proposta é descentralizada, autônoma e capaz de tratar fenômenos de propagação de sinal. Para validar a abordagem são utilizados uma análise teórica e experimentos reais e simulados. A análise teórica prevê um limite máximo de robôs roteadores para a criação da solução. Os experimentos reais, através da métrica de vazão da rede, validam a abordagem em ambientes reais.



# Abstract

Mobile Robotics can help humanity in many tasks, including assist rescue operations in disasters. In environments with no communication infrastructures, it is important to establish an ad-hoc network to allow rescue teams to communicate. This work investigates the problem of deploying a set of networked robots to create an ad-hoc network interconnecting a set of static clients, allowing the clients to communicate among themselves. The proposed approach has two phases.

First, a Steiner tree is built interconnecting the clients. This tree is constructed even in the presence of obstacles and, by definition, is the smallest Euclidean tree interconnecting the set of clients. In a second phase, each robot runs a state machine, allowing the ad-hoc network to be created autonomously. The approach is decentralized, autonomous and able to deal with signal propagation phenomena. We validate our approach through physical and simulated experiments and theoretical analysis. The theoretical analysis provides a bound on the maximum number of networked robots to create the solution. The real experiments, using the throughput metric, validate the approach in physical environments.



# Lista de Figuras

1.1	Representação do problema de interconectar um conjunto de clientes estáticos em uma rede ad-hoc utilizando robôs roteadores. Nessa representação, é necessário conectar um conjunto de 4 clientes. A solução utiliza 3 robôs roteadores que são posicionados no ambiente. . . . .	4
3.1	Exemplo de uma árvore de Steiner Eucliana. A árvore de Steiner apresentada conecta um conjunto de 532 cidades dos Estados Unidos da América. (Exemplo retirado do trabalho de Warme et al. [2000]) . . . . .	18
3.2	Comparação da solução de uma MST (árvore da esquerda) com a solução de uma EST (árvore da direita). Na figura, os pontos terminais são representados pelos círculos e os ponto de Steiner por um triângulo. . . . .	20
3.3	Exemplo de uma árvore de Steiner Euclidiana Mínima com Desvio de Obstáculos. A árvore de Steiner apresentada conecta 10, 50 e 150 pontos respectivamente. (Exemplo retirado do trabalho de Zachariasen & Winter [1999]) . . . . .	21
3.4	Exemplo de uma árvore de Steiner em grafo. A árvore de Steiner apresentada conecta 23 pontos terminais (Pontos de cor mais escura). (Exemplo retirado do trabalho de Bayati et al. [2008]) . . . . .	22
3.5	Exemplo de uma árvore de Steiner com o Número Mínimo de Pontos de Steiner. A árvore de Steiner apresentada conecta 9 pontos terminais utilizando 3 pontos de Steiner. (Exemplo retirado do trabalho de Lin & Xue [1999]) . . . . .	24
3.6	A perda de caminho ocorre pela propagação da onda no meio. O sombreamento ocorre pela presença de obstáculos, na figura uma árvore. Os caminhos múltiplos ocorrem pelas múltiplas ondas do mesmo sinal que chegam ao receptor, na figura causado pela reflexão do sinal na parede. . . . .	26
4.1	Diagrama de estados da CEFMSM. . . . .	32

4.2	Representação do funcionamento da CEFSM. . . . .	34
4.3	O robô no estado <i>move</i> (1) move em direção ao cliente desconectado até perder conexão (limiar inferior) mudando para o estado <i>desconectado</i> . No estado <i>desconectado</i> o robô move para trás até reconectar com a rede perdida (limiar superior), mudando para o estado <i>conectado</i> (2). . . . .	35
4.4	Ilustração de variáveis. . . . .	37
4.5	CEFSM Contra Alocação Ótima. . . . .	38
5.7	Média, valor máximo e mínimo do RSS em relação a distância. . . . .	44
5.1	Sequência de imagens de uma simulação. . . . .	46
5.2	Simulação com um ambiente contendo obstáculos e 5 clientes (pontos). A linha preta indica a árvore de Steiner. A solução final necessitou de 26 robôs (quadrados). . . . .	47
5.3	Ambiente e mapa dos experimentos. . . . .	47
5.4	Topologia final de experimento simulado com 4 clientes. . . . .	48
5.5	Etapas da abordagem na comparação das árvores de Steiner. . . . .	48
5.6	Robô utilizados em experimentos. . . . .	48
5.8	Topologia de experimento real com 3 clientes. . . . .	49
5.9	Vazão da rede em experimento real com 3 clientes. . . . .	49
5.10	Topologia de experimento real com 2 clientes. . . . .	50
5.11	Vazão da rede em experimento real com 2 clientes. . . . .	50
5.12	Topologia final e layout de experimento simulado . . . . .	51
5.13	Topologia final e layout de experimento real. . . . .	51
5.14	Vazão da rede em experimento real com dois clientes. . . . .	52

# Lista de Tabelas

2.1	Trabalhos relacionados a criação de redes ad-hoc utilizando robôs roteadores	15
4.1	Tabela de estado do CEFSM. . . . .	32
4.2	Tabela de Ações. . . . .	33
4.3	Tabela de Eventos. . . . .	33
4.4	Tabela de informações compartilhadas. . . . .	36
4.5	Variáveis utilizadas na análise teórica. . . . .	38
5.1	Comparação da árvore de Steiner Euclidiana e da árvore de Steiner em grafo.	43
5.2	Topologia final dos experimentos de raio 38 u.m. da Tabela 5.1 . . . . .	43



# Sumário

<b>Agradecimentos</b>	<b>ix</b>
<b>Resumo</b>	<b>xiii</b>
<b>Abstract</b>	<b>xv</b>
<b>Lista de Figuras</b>	<b>xvii</b>
<b>Lista de Tabelas</b>	<b>xix</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	3
1.2 Definição do Problema . . . . .	4
1.3 Objetivo . . . . .	4
1.4 Contribuições . . . . .	5
1.5 Organização do Trabalho . . . . .	5
<b>2 Revisão Bibliográfica</b>	<b>7</b>
2.1 Clientes Estáticos . . . . .	7
2.2 Clientes Móveis . . . . .	8
2.3 Cobertura . . . . .	9
2.4 Cobertura e Gateway . . . . .	10
2.5 Navegação . . . . .	10
2.6 Formação . . . . .	11
2.7 Protocolo de comunicação . . . . .	12
2.8 Visão Geral e Discussão . . . . .	13
<b>3 Fundamentação Teórica</b>	<b>17</b>
3.1 Árvores de Steiner . . . . .	17
3.1.1 Árvore de Steiner Euclidiana Mínima . . . . .	18

3.1.2	Árvore de Steiner Euclidiana Mínima com Desvio de Obstáculos	21
3.1.3	Árvore de Steiner em Grafo	22
3.1.4	Árvore de Steiner com o Número Mínimo de Pontos de Steiner	23
3.2	Propagação de Sinal	25
3.2.1	Modelo de propagação em espaço livre	26
3.2.2	Modelo de propagação log-distância	27
3.2.3	Modelo de fator de atenuação	27
<b>4</b>	<b>Metodologia</b>	<b>29</b>
4.1	Modelagem	29
4.2	Planejador de caminho	30
4.2.1	Árvore de Steiner Euclideana com Desvio Obstáculos	30
4.2.2	Árvore de Steiner em Grafos	31
4.3	Alocação dos Robôs Roteadores	31
4.4	Comunicação	36
4.5	Análise Teórica	36
<b>5</b>	<b>Resultados</b>	<b>41</b>
5.1	Experimentos Simulados	41
5.1.1	Comparação das árvores de Steiner	42
5.2	Experimentos Reais	44
5.3	Comparação Experimentos Reais e Simulados	45
<b>6</b>	<b>Conclusão e Trabalhos Futuros</b>	<b>53</b>
6.1	Conclusão	53
6.2	Trabalhos Futuros	54
	<b>Referências Bibliográficas</b>	<b>55</b>

# Capítulo 1

## Introdução

Com o aprimoramento da capacidade de sensoriamento e atuação dos robôs, além da evolução dos dispositivos de comunicação, podemos criar os chamados robôs roteadores, também conhecidos como *networked robots*. Esses robôs são capazes de, além de perceber e atuar em um ambiente, comunicar de forma explícita com outros robôs, podendo se comportar como dispositivos finais em uma rede, apenas enviando e/ou recebendo mensagens, ou agir como roteadores, retransmitindo mensagens destinadas a outros dispositivos.

Robôs podem auxiliar operações de busca e salvamento, sendo capazes de visitar locais inóspitos para seres humanos. Mais especificamente, robôs roteadores podem ser utilizados para fornecer serviços de comunicação. Nesse contexto, se considerarmos um cenário de desastre onde as infraestruturas de comunicação foram totalmente ou parcialmente destruídas, robôs roteadores podem interconectar um conjunto de clientes fixos no ambiente, como estações base de bombeiros e policiais, em uma rede ad-hoc, permitindo que esses comuniquem entre si. No mesmo cenário, a rede criada pode servir de apoio para outros robôs, que podem executar tarefas específicas de busca e salvamento sem a necessidade de tratar problemas de comunicação.

Esta dissertação estuda o problema de utilizar robôs roteadores para criar uma rede ad-hoc conectando um conjunto de clientes. Os clientes representam estações base de entidades de resgate como policiais, bombeiros e paramédicos. No problema é considerado que os clientes são estáticos e o ambiente é conhecido. Para criar a rede, os robôs roteadores precisam ser alocados no ambiente, permitindo que os pares de clientes sejam capazes de se comunicar.

Neste trabalho é proposta uma abordagem composta por duas etapas para criar a rede ad-hoc. Primeiro é calculado uma árvore de Steiner Euclideana conectando o conjunto de clientes. Por definição, a árvore de Steiner interconectará os clientes

com a menor árvore possível, sendo utilizada com o intuito de reduzir o número de robôs necessários para criação da rede. Segundo, utilizando a árvore de Steiner como um planejador de caminho global, cada robô executa uma máquina de estados, sendo guiados para conectar o cliente desconexo da rede mais próximo, e quando necessário servir de suporte para outros robôs seguirem esse objetivo. O comportamento gerado é a criação da rede, permitindo que os clientes comuniquem entre si. A abordagem proposta é descentralizada, autônoma, e é capaz de lidar com fenômenos de propagação de sinal. Para validar o trabalho utilizamos uma análise teórica e experimentos simulados e reais.

O problema de interconectar um conjunto de clientes estáticos em uma rede ad-hoc é estudado em outros trabalhos. No trabalho apresentado por Vieira et al. [2011], uma solução *offline* baseada em campos potenciais e modelos de propagação de sinal é apresentado. Enquanto isso, Chiu et al. [2009] utiliza uma solução bio-inspirada onde os robôs roteadores criam uma rede na direção dos clientes desconectados. Já Williams et al. [2014] define fluxos para pares de clientes, repartindo o número de robôs com objetivo a priorizar determinados fluxos. Nesses trabalhos o número de robôs necessários para criar uma solução não é estudado. Dessa forma, a abordagem apresentada nesta dissertação se difere dos demais trabalhos dado que a solução proposta é *online*, e um dos objetivos é apresentar um limite superior no número de robôs necessários para a solução.

Em outros trabalhos, como os apresentados por Gil et al. [2012] e Feldman et al. [2013], o objetivo é interconectar um conjunto de clientes móveis utilizando outro conjunto de robôs roteadores. Nesta dissertação, o objetivo é conectar um conjunto de clientes estáticos, sendo possível apresentar um limite máximo de robôs necessários para criar a solução.

Diversos trabalhos que consideram o problema de criar rede entre robôs, como os trabalhos de Gil et al. [2012] e Giordano et al. [2013], utilizam como métrica de conectividade a distância entre os dispositivos, sendo essa métrica não realista dado que a intensidade do sinal não depende apenas da distância. Outros trabalhos já consideram a conectividade com métricas reais de intensidade de sinal, por exemplo Rizzo et al. [2013] e Chiu et al. [2009], apresentando determinada garantia que a abordagem pode ser aplicada em ambientes reais. Nesta dissertação é utilizado a métrica de RSS (do inglês, *Received Signal Strength*), possibilitando que a abordagem considere fenômenos reais de propagação de sinal.

## 1.1 Motivação

As motivações para este trabalho são cenários de desastre onde infraestruturas de comunicação são inexistentes ou foram destruídas. Como exemplo desse cenário, é possível citar o furacão Katrina, que atingiu a costa leste dos Estados Unidos em 2005, destruindo as infraestruturas de comunicação, incluindo as torres de transmissão sem fio, deixando os primeiros times de resgate incapazes de se comunicar apropriadamente (Portmann & Pirzada [2008]). Em uma situação similar, robôs roteadores podem ser utilizados para criar uma rede conectando um conjunto de estações base, fornecendo suporte para os times de resgate até que as infraestruturas de comunicação sejam reparadas.

Outro exemplo, onde os principais meios de comunicação foram afetados após um desastre, é o terremoto de magnitude 8.8 que ocorreu no sul do Chile em 2010. Na ocasião, a energia elétrica foi cortada, além da quebra de cabos de fibra óptica, afetando os meios de comunicação, incluindo redes de telefone celular. Com isso, as comunicações foram afetadas em diversas áreas e por um longo período de tempo (Eiselt & Marianov [2012]). Em um cenário similar, as infraestruturas podem ser temporariamente substituídas por uma rede ad-hoc, permitindo que a comunicação seja restabelecida.

Apesar da existência de tecnologia capaz de prover comunicação a dezenas de quilômetros, como os rádios UHF, cenários de desastre em grande escala podem sofrer problemas de comunicação sem fio, existindo o limite de alcance dos rádios de comunicação. Cenários de menor escala, por exemplo interior de prédios ou estações de metrô subterrâneas, onde a propagação do sinal pode ser limitada, também podem apresentar um desafio em relação a comunicação. Nesses cenários, os robôs podem fornecer pontos de retransmissão de sinal, permitindo que a comunicação entre os agentes não seja prejudicada.

Em geral, a capacidade de comunicação entre times de resgate pode ser um elemento importante em cenários de desastre, sendo possível a transmissão de informações cruciais como: locais não seguros para os times de resgate; estado e localização de sobreviventes; e informações importantes relacionadas a saúde e segurança pública. Nesse sentido, também é possível utilizar a rede criada para transmissão de vídeos e imagens em tempo real, permitindo que a área de desastre seja analisada por especialistas que não estão presentes na área do desastre.

## 1.2 Definição do Problema

O objetivo deste trabalho é alocar um conjunto de robôs roteadores  $\mathcal{R} = \{r_1, r_2, \dots, r_n\}$  de forma a estabelecer uma rede ad-hoc que permita a comunicação dos clientes  $\mathcal{C} = \{c_1, c_2, \dots, c_m\}$ . Esses clientes são posicionados em um ambiente conhecido e representado por um mapa  $M$ . Os clientes têm sua posição conhecida e são estáticos, não podendo locomover pelo ambiente. Além disso, o ambiente pode conter obstáculos e os clientes são dispositivos finais da rede, não sendo capazes de agir como roteadores.

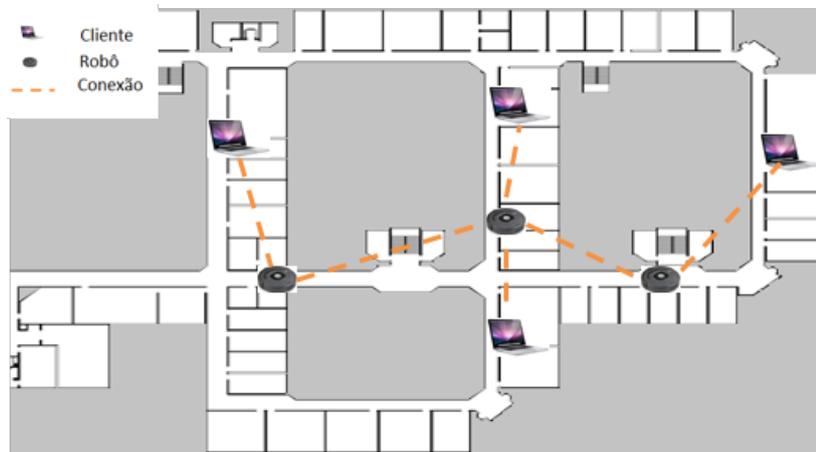
O problema pode ser definido como:

**Entrada:** Dado a posição dos clientes  $\mathcal{C}$  e o mapa do ambiente  $M$ .

**Objetivo:** Alocar os robôs roteadores para estabelecer uma rede com conectividade entre os pares de clientes.

**Saída:** Cada robô  $r_i$  move autonomamente da posição inicial até uma posição final no ambiente.

A Figura 1.1 ilustra o problema abordado nesta dissertação.



**Figura 1.1.** Representação do problema de interconectar um conjunto de clientes estáticos em uma rede ad-hoc utilizando robôs roteadores. Nessa representação, é necessário conectar um conjunto de 4 clientes. A solução utiliza 3 robôs roteadores que são posicionados no ambiente.

## 1.3 Objetivo

O objetivo desta dissertação é estudar o problema de utilizar robôs roteadores para criar uma rede ad-hoc interconectando um conjunto de clientes. Os clientes são estáticos e o

ambiente é conhecido e possui obstáculos. A abordagem proposta deve ser autônoma e capaz de ser utilizada em ambientes reais, tratando fenômenos de propagação de sinal. Além disso, outro objetivo deste trabalho é prever a quantidade máxima de robôs necessários para uma solução.

## 1.4 Contribuições

As principais contribuições deste trabalho são:

- É apresentado um sistema completo capaz de criar uma rede ad-hoc conectando um conjunto de clientes. Por sistema completo entende-se que é determinado a localização e é feita a alocação dos robôs.
- A abordagem proposta leva em consideração fenômenos de propagação de sinal (perda de caminho, sombreamento, multi-caminhos).
- Através de um estudo analítico, é apresentado um limite superior para o número de robôs necessários para criar uma solução.

O projeto desta dissertação, além de outros projetos relacionados, foram apresentados em artigos publicados em conferências nacionais e internacionais.

- **Santos, Elerson R. S.**, Vieira, Marcos A. M. Autonomous wireless backbone deployment with bounded number of networked robots *International Conference on Intelligent Robots and Systems (IROS)*, 2014. [Qualis-CC A1]
- Macharet, D. G., Assis, N. N., Valle, D., **Santos, Elerson R. S.**, Vieira, M. A. M., Mario Campos A Graph-based Algorithm for Minimum Router Deployment. *Latin American Robotics Symposium (LARS)*, 2014. [Qualis-CC B4]

## 1.5 Organização do Trabalho

Este trabalho está organizado da seguinte forma: o Capítulo 3 apresenta a fundamentação teórica utilizada para o desenvolvimento do trabalho. O Capítulo 2 apresenta a revisão bibliográfica dos trabalhos que estão relacionados ao problema estudado. O Capítulo 4 apresenta a metodologia desenvolvida para criação de uma solução para o problema estudado. O Capítulo 5 apresenta e discute os resultados obtidos com a metodologia proposta. Para finalizar, o Capítulo 6 apresenta uma conclusão e as direções futuras do trabalho apresentado.



# Capítulo 2

## Revisão Bibliográfica

Este capítulo apresenta os principais trabalhos relacionados à criação de redes ad-hoc utilizando robôs roteadores. Os trabalhos apresentados possuem como característica comum o objetivo de manter a conectividade entre robôs roteadores e em alguns casos, como no problema abordado neste trabalho, clientes.

### 2.1 Clientes Estáticos

O problema de alocação de robôs roteadores para conectar clientes estáticos, problema estudado nesta dissertação, é abordado em diversos trabalhos. Chiu et al. [2009] apresenta uma abordagem bio-inspirada que é capaz de interconectar entidades críticas de um ambiente. No trabalho proposto, os robôs são alocados para formar "tentáculos" em direção aos clientes que devem ser conectadas na solução. Se a direção para o cliente não é conhecido, os robôs seguem uma direção aleatória, e em caso de um "tentáculo" ser verificado como desnecessário, esse é desfeito, permitindo que os robôs dessa parte da solução sejam utilizados em outra região da rede.

Considerando um ambiente e posição dos clientes conhecidos, Vieira et al. [2011] propõe uma estratégia *offline* baseada em campos potenciais artificiais e modelos de propagação de sinal, obtendo ao final a posição onde um conjunto de robôs precisa ser alocado para que a rede seja criada. A abordagem proposta é capaz de ser utilizada em ambiente contendo obstáculos.

Chiu & Shen [2011] utiliza de medições de sinal RSS (do inglês, *Received Signal Strength*) feitas em um ambiente para criar uma solução. Na abordagem proposta, um conjunto de pontos é previamente definido no ambiente e um conjunto de robôs faz a medição do sinal entre pares dessas localizações. A solução apresentada é ótima em relação a discretização feita, sendo esse resultado obtido utilizando uma solução

de uma árvore de Steiner em grafos. Além disso, a solução apresentada é dependente da granularidade da discretização, quanto menor a granularidade melhor a solução, entretanto maior o tempo para fazer as medições do RSS.

O trabalho apresentado por Rizzo et al. [2013] investiga o problema de conectividade em um ambiente com forte taxa de decaimento de sinal, por exemplo túneis e galerias, sendo considerado apenas a conexão entre pares de clientes. Para lidar com o problema, a solução proposta utiliza a medição da qualidade de sinal, e ao formar a rede, os robôs são alocados um a um, permitindo que a alocação seja refeita caso uma alocação de maior distância seja percebida posteriormente.

Já Williams et al. [2014] considera o problema onde a qualidade das rotas pode ser priorizada, garantindo qualidade entre determinados clientes pertencentes a rede. Para isso, na abordagem são definidos fluxos para conjunto de clientes, esses fluxos determinam uma razão de quantos robôs devem ser alocados para cada par de clientes. Para interconectar os clientes, robôs são alocados equidistantes uns dos outros na linha reta que interconecta os clientes.

Abordando o problema onde o número de robôs roteadores não é o suficiente conectar os clientes estáticos em uma rede, temos o trabalho de Abbas & Younis [2013]. Nesse trabalho, uma rede tolerante a atraso (DTN, do inglês *Delay-tolerant networking*) é formada para conectar os clientes, sendo determinado um *tour* que os robôs precisam percorrer para fornecer conexão a subconjunto de clientes.

## 2.2 Clientes Móveis

Considerando o problema onde um conjunto de robôs roteadores prove uma rede para um cliente móvel, Tekdas et al. [2009] modela o problema como um jogo de "pega ladrão" (do inglês, *Pursuit-evasion*). No modelo proposto, o pior cenário é considerado, onde o cliente irá tentar evadir da rede o mais rápido possível. Para resolver o problema, a abordagem proposta discretiza o ambiente e os robôs roteadores são alocados para cobrir as rotas de fuga, garantindo a conectividade do cliente.

Já Gil et al. [2012] considera um conjunto de clientes móveis que se movimenta por um ambiente. Nesse trabalho, os autores utilizam uma árvore geradora de menor largura (MBST do inglês, *Minimal Bottleneck Spanning Tree*) para gerar as configurações onde os robôs roteadores devem ser alocados para conectar os clientes em uma rede de topologia fixa. Dado restrições de velocidade, a abordagem ainda prevê o tempo que uma determinada configuração será válida, permitindo a escolha da configuração que reduza a mobilidade dos robôs roteadores.

Com o intuito de aprimorar a abordagem de Gil et al. [2012], o trabalho de Feldman et al. [2013], além de conectar os clientes móveis em uma rede, permite uma rede de topologia dinâmica. Na solução proposta, a MBST é utilizada como limite superior para o algoritmo, e para formar a rede dinamicamente, dado um conjunto de possíveis soluções, que são geradas pelo movimento dos clientes, ou de forma aleatória, a configuração que melhor atende os clientes é selecionada.

O problema onde um robô cliente precisa manter uma conexão com uma estação de trabalho é abordado por Stump et al. [2008] e Fink et al. [2012]. Stump et al. [2008] utiliza o valor de Fiedler, com métrica de conectividade RSS, para garantir que os robôs roteadores sejam capazes de manter a conectividade entre o robô cliente e a estação base. Na abordagem, os robôs roteadores se movimentam na direção que otimiza o valor de Fiedler da rede. Enquanto isso, Fink et al. [2012] utiliza um modelo estocástico de propagação de sinal para garantir a conectividade e definir qual a movimentação que os robôs devem realizar para que a conexão fim-a-fim seja mantida.

## 2.3 Cobertura

O problema de cobertura, que é classificado por Gage [1992] como problema de *blanket coverage*, tem por objetivo maximizar a área de cobertura de um conjunto de sensores, maximizando as chances da detecção de eventos que ocorrem em um determinado ambiente. De forma geral, o problema de cobertura não requer que os sensores mantenham conectividade, entretanto essa restrição é adicionada em alguns trabalhos, permitindo que os sensores formem uma rede onde informações podem ser compartilhadas.

Alguns dos trabalhos que abordam o problema de cobertura com conectividade utilizam campos potenciais artificiais, modelando o problema para que os robôs se espalhem pelo ambiente ao mesmo tempo que a conectividade é garantida. Na abordagem apresentada por Howard et al. [2002], que estuda o problema de cobertura em ambientes não conhecidos, são modelados campos de atração e repulsão entre robôs e obstáculos, com isso, a conectividade é garantida pela força atração e o espalhamento dos robôs é feita pelas forças de repulsão.

Enquanto isso, Poduri & Sukhatme [2004] estuda o problema considerando o número de vizinhos de cada robô, se o número de vizinhos entre os robôs é menor que um valor  $k$ , uma nova força de atração surge no sistema, garantindo que uma rede  $k$ -conexa seja formada ao mesmo tempo que o ambiente é coberto. Howard et al. [2002] e Poduri & Sukhatme [2004] utilizam a distância para definir as forças do sistema, já Guan et al. [2008] propõe uma abordagem que utiliza campos potências baseados no

valor do SNR (do inglês, *Signal-to-Noise Ratio*) medido, permitindo que uma medida de conectividade realística seja utilizada em um problema de cobertura. Além disso, para garantir que uma rede seja capaz de se regenerar ao perder conectividade, a abordagem inicia com uma rede conexa e mantém uma memória contendo os locais onde cada robô possuiu conectividade, fazendo com que ao quebrar a conectividade o robô se locomova para um local onde existia conectividade.

Outros trabalhos abordam o problema de cobertura considerando partições Voronoi, por exemplo Cortes et al. [2004] e Pimenta et al. [2008], entretanto o problema de conectividade não é abordado nesses trabalhos.

## 2.4 Cobertura e Gateway

Uma variação do problema de cobertura é estudado por Correll et al. [2009] e Derbakova et al. [2011]. No problema estudado, além de tentar aumentar a área coberta, é necessário que um *gateway* esteja conectado a solução final. Correll et al. [2009] ainda adiciona a restrição de que os robôs possuam capacidades mínimas de locomoção e sensoriamento, sendo possível para os robôs apenas locomover e avaliar o estado da rede, permitindo que robôs de baixo custo possam ser utilizados. Na solução proposta, os robôs se locomovem pelo ambiente utilizando um *random walk* (Camp et al. [2002]), havendo uma probabilidade de se juntar ou evadir da rede dependendo do número local de robôs estão conectados.

Já no trabalho proposto por Derbakova et al. [2011], os robôs possuem informação de localidade, e em caso de possuir conectividade com o *gateway*, campos potenciais artificiais são utilizados para aumentar a cobertura da rede. Caso contrário, um algoritmo de planejamento de caminho é utilizado para conectar cada robô desconexo ao robô mais próximo do *gateway*.

## 2.5 Navegação

Considerando o problema de navegação, onde um conjunto de robôs navega por um ambiente conhecido para execução de tarefas, a conectividade entre os robôs pode ser um dos requisitos de uma missão. De forma geral, os problemas que estudam conectividade na navegação podem ser separados em dois grupos. No primeiro, onde existe um grupo homogêneo de robôs roteadores, o objetivo é manter a conectividade do grupo enquanto os robôs navegam pelo ambiente. No segundo, onde existe um conjunto de clientes e um conjunto de robôs roteadores, o objetivo dos robôs roteadores é manter

a conectividade dos clientes enquanto esses navegam pelo ambiente.

Considerando um conjunto homogêneo de robôs roteadores, Esposito & Dunbar [2006] propõe uma função de navegação para um enxame de robôs que considera um objetivo global e a conectividade dos robôs. Na abordagem proposta, cada robô seleciona, dentre um conjunto de possíveis valores, a melhor variável de controle que minimiza essa função de navegação proposta. Essa abordagem é descentralizada, utiliza linha de visão e distância para determinar a conectividade par-a-par dos robôs e permite uma topologia de rede dinâmica.

Enquanto isso, Sabbatini et al. [2012] utiliza um controlador baseado no valor de Fiedler (Fiedler [1973]) para manter a conectividade do grupo. Sendo o valor de Fiedler uma medida algébrica de conectividade. No trabalho, a topologia da rede é dinâmica, sendo necessários apenas que os robôs mantenham o valor Fiedler da rede maior que um limiar pré-definido. Giordano et al. [2013] também utiliza o valor de Fiedler para manter a conectividade do grupo, entretanto, indo além, o trabalho propõe um modelo de conectividade que leva em consideração requisitos locais, como controle de formação e restrições para evitar colisão. Em um trabalho com propósito similar, e também com topologia dinâmica, Yao & Gupta [2009] utiliza um algoritmo de agrupamentos para eleger líderes em um grupo de robôs. Na abordagem, os robôs líderes são responsáveis por manter a conectividade, enquanto os outros robôs continuam suas missões e selecionam os líderes mais próximos, pelos quais devem manter a conectividade. Nesses trabalhos, a conectividade é determinada pela distância entre os robôs.

Já abordando o problema considerando que os objetivos dos robôs são conhecidos, e com uma topologia pré-definida, temos a abordagem proposta por Hsieh et al. [2006]. Nesse trabalho, campos artificiais são utilizados para manter a conectividade e guiar os robôs para o objetivo mantendo a conectividade nos links locais pré-determinados.

O trabalho proposto por Hollinger & Singh [2010] relaxa o problema, definindo que a conectividade entre os robôs pode ser periódica, sendo permitido aos robôs perder a conectividade por curtos períodos de tempo. Na solução proposta, a abordagem considera que é possível determinar um conjunto de configurações conexas intermediárias entre a configuração inicial e a final, sendo o objetivo determinar os caminhos entre as configurações que minimizam o tempo que os robôs perdem conectividade.

## 2.6 Formação

O trabalho de Spanos & Murray [2005] estuda o problema de formação e conectividade de grupos de robôs. Nesse trabalho, um conjunto de agentes em uma configuração

conexa tem por objetivo atingir outra configuração, também conexa, mantendo sempre as condições de conectividade global da rede. Na solução, o planejamento das posições dos robôs é feito através de configurações intermediárias que permitem os robôs se mantenham conexos.

## 2.7 Protocolo de comunicação

Em outros trabalhos, o objetivo é o estudo de protocolos de comunicação para conjuntos de robôs roteadores. Zeiger et al. [2008] faz uma análise de protocolos de roteamento existentes em uma tarefa de teleoperação. Nesse trabalho é considerado o caso onde apenas existe uma rota possível para roteamento. São analisados os protocolos de roteamento: AODV S. Das & Belding-Royer [2003], OLSR Clausen [2003], DSR Y.-C. Hu & Maltz. [2004], e B.A.T.M.A.N<sup>1</sup>. Os resultados mostraram que o protocolo B.A.T.M.A.N não conseguiu criar uma rota para o ambiente descrito, já os protocolos AODV e OLSR levaram até 30 segundos para conseguir criar a rota do robô para um teleoperador. O DSR apresentou os melhores resultados, necessitando pouco tempo para recriar as rotas, conseqüentemente apresentando a menor taxa de perda de pacotes.

Em Kudelski et al. [2014], uma abordagem para estimar a qualidade de sinal é apresentada. No método, inicialmente a qualidade de sinal (RSS ou SNR) é medida para um conjunto topologias, em seguida, é utilizado um algoritmo de aprendizagem e-SVR, similar ao SVM, para criação de um modelo capaz de prever a qualidade de sinal no ambiente. Os experimentos mostram que a abordagem proposta apresenta alta correlação entre os valores medidos e estimados. A abordagem é posteriormente testada com o algoritmo de roteamento OLSR, criando o OLSR-LQE, melhorando a qualidade dos resultados apresentados pelo OLSR.

Já o trabalho de Stephan et al. [2014] tem por objetivo explorar a camada de transporte. No trabalho é proposto o MCTP, protocolo que tem por objetivo substituir o TCP e o UDP em redes sem fio. O MCTP tem por objetivo apresentar garantia na entrega de pacotes, e superar o problema do TCP que considera que todas as perdas de pacotes são causadas por congestionamentos. O MCTP mistura atributos da camada de rede e camada de transporte, utilizando a qualidade do link fornecida pela camada de rede e afetando as rotas de roteamento de acordo com as falhas detectadas nos links.

---

<sup>1</sup><https://www.open-mesh.net/batman>

## 2.8 Visão Geral e Discussão

Comparando as abordagens existentes para criação de uma rede conectando clientes estáticos com o trabalho desta dissertação, é possível verificar algumas diferenças. A abordagem proposta neste trabalho não possui uma etapa *offline* que determina a posição onde os robôs devem ser alocados, também não utiliza de uma etapa anterior a alocação para medição do RSS. Outra diferença é que este trabalho apresenta um limite superior no número de robôs necessários para criar uma solução, sendo que outros trabalhos encontrados na literatura não fazem essa consideração.

Já a diferença desse trabalho para os problemas onde os clientes são móveis está na possibilidade da definição de um número de robôs necessários para criação de uma solução. Sendo que determinar um valor máximo para o caso dos clientes móveis é complexo devido a existência de obstáculos.

O trabalho estudado nesta dissertação tem como objetivo utilizar robôs roteadores para criar uma rede conectando um conjunto de clientes estáticos. Assim, o problema de cobertura é diferente do problema estudado nesta dissertação dado que esse problema não define clientes que devem se conectados a uma rede, além disso, seu objetivo é maximizar uma área de cobertura. No problema estudado nesta dissertação, dado que o conjunto de clientes e suas posições são conhecidos, é possível limitar o número de robôs necessários para a criar uma solução.

Já o problema de cobertura com a adição de um *gateway* mostra que é possível adicionar clientes fixos à definição do problema de cobertura, entretanto o problema ainda se difere do problema desta dissertação pelo número de robôs necessários para criar uma solução.

Enquanto isso, o problema estudado nesta dissertação difere dos problemas de navegação de grupos homogêneos de robôs dado que, por definição, esse problema não define clientes que devem ser conectados a uma rede, sendo o objetivo manter um grupo conectado ao realizar uma tarefa. No problema estudado nesta dissertação, os clientes são fixos, possuem localização conhecida e são definidos como dispositivos finais, não possuindo a capacidade de rotear mensagens para outros dispositivos.

A Tabela 2.1 resume e classifica os principais trabalhos relacionados ao problema de criação de redes ad-hoc utilizando robôs roteadores. A Tabela está dividida nos seguintes campos: autores; tipos de robôs utilizados, robôs roteadores e/ou robôs clientes; problema abordado; capacidade de lidar com obstáculos; clientes móveis ou estáticos; ambiente conhecido; topologia da rede, estática ou dinâmica; métrica de conectividade. As métricas de conectividade são: *Received Signal Strength* (RSS), que mede a potência de sinal que chega ao receptor; *Signal-to-Noise Ratio* (SNR), que mede

a razão entre a potência do sinal recebido e do ruído; e distância.

**Tabela 2.1.** Trabalhos relacionados a criação de redes ad-hoc utilizando robôs roteadores

Autores	Robôs	Problema	Obstáculos	Clientes Móveis	Ambiente Conhecido	Topologia	Conectividade
Howard et al. [2002]	Roteadores	Cobertura	Sim	-	Não	Dinâmica	Distância
Poduri & Sukhatme [2004]	Roteadores	Cobertura	Sim*	-	Não	Dinâmica	Distância
Guan et al. [2008]	Roteadores	Cobertura	Não	-	Não	Dinâmica	SNR
Correll et al. [2009]	Roteadores	Conectividade com gateway e Cobertura	Sim	-	Não	Dinâmica	RSS
Derbakova et al. [2011]	Roteadores	Conectividade com gateway e Cobertura	Sim	-	Sim	Dinâmica	Distância
Esposito & Dumar [2006]	Roteadores	Navegação	Sim	-	Sim	Dinâmica	Distância e Linha de Visão
Sabbatini et al. [2012]	Roteadores	Navegação	Não*	-	Não*	Dinâmica	Distância
Giordano et al. [2013]	Roteadores	Navegação	Sim	-	Sim	Dinâmica	Distância e Linha de Visão
Yao & Gupta [2009]	Roteadores	Navegação	Sim	-	Não*	Dinâmica	Distância
Hsieh et al. [2006]	Roteadores	Navegação	Sim	-	Sim*	Estática	SNR ou Bandwidth
Hollinger & Singh [2010]	Roteadores	Navegação	Sim	-	Sim	Dinâmica	Distância e Linha de Visão
Tekdas et al. [2009]	Roteadores e Clientes	Clientes Móveis	Sim	-	Sim	Dinâmica	Linha de Visão
Gil et al. [2012]	Roteadores e Clientes	Clientes Móveis	Não	Sim	Não*	Estática	Distância
Feldman et al. [2013]	Roteadores e Clientes	Clientes Móveis	Não	Sim	Não	Dinâmica	Distância
Stump et al. [2008]	Roteadores e Clientes	Clientes Móveis	Sim	Sim	Não*	Dinâmica*	RSS
Fink et al. [2012]	Roteadores e Clientes	Clientes Móveis	Sim	Sim	Sim	Dinâmica*	Modelo de Propagação
Spanos & Murray [2005]	Roteadores	Formação	Sim	-	Sim	Estática	Distância
Vieira et al. [2011]	Roteadores e Clientes	Clientes Estáticos	Sim	Não	Sim	Estática	Modelo de Propagação
Chiu & Shen [2011]	Roteadores e Clientes	Clientes Estáticos	Sim	Não	Sim	Estática	RSS
Chiu et al. [2009]	Roteadores e Clientes	Clientes Estáticos	Sim	Não	Não	Dinâmica	RSS
Rizzo et al. [2013]	Roteadores e Clientes	Clientes Estáticos	Não	Não	Não	Estática	RSS
Williams et al. [2014]	Roteadores e Clientes	Clientes Estáticos	Não	Não	Não	Dinâmica	Distância
Abbas & Younis [2013]	Roteadores e Clientes	Clientes Estáticos	Não*	Não	Não	Dinâmica	Distância
Zeiger et al. [2008]	-	Protocolo de Comunicação	-	-	-	Dinâmica	-
Stephan et al. [2014]	-	Protocolo de Comunicação	-	-	-	Dinâmica	-
Kudelski et al. [2014]	-	Protocolo de Comunicação	-	-	-	Dinâmica	-

- não se aplica, \* não mencionado explicitamente



# Capítulo 3

## Fundamentação Teórica

Este capítulo descreve os fundamentos teóricos deste trabalho. A primeira parte descreve o problema da árvore de Steiner com suas variantes e algoritmos. A árvore de Steiner é utilizada para definir o menor caminho Euclidiano que interconecta um conjunto de clientes. A segunda parte descreve a propagação de sinal de ondas de radiofrequência. Essa descrição é utilizada para discutir algumas características do espaço de propagação de sinal.

### 3.1 Árvores de Steiner

Dado um conjunto de pontos  $\mathcal{C}$ , uma árvore de Steiner é definida como uma árvore que interconecta os pontos  $\mathcal{C}$ , podendo a solução ainda conter outro conjunto de pontos  $\mathcal{S}$ . O conjunto de pontos  $\mathcal{C}$  são chamados de pontos terminais e os pontos  $\mathcal{S}$  são chamados de pontos de Steiner. Os pontos de Steiner não precisam necessariamente pertencer a uma solução, enquanto os pontos terminais precisam. De forma geral, os problemas relacionados a criação de árvores de Steiner procuram interconectar o conjunto  $\mathcal{C}$ , utilizando os pontos  $\mathcal{S}$  para atingir algum objetivo específico, como por exemplo minimizar o custo da árvore gerada.

O problema das árvores de Steiner possui aplicações em problemas relacionados a confecção de circuitos integrados, posicionamento de sensores, algoritmos de roteamento, dentre outros. Nesse contexto, diversas variantes do problema podem ser encontradas, cada uma com diferentes objetivos de otimização. Apesar da sua utilidade, a maioria das variantes do problema da árvore de Steiner são provados pertencer a classe de algoritmos NP-completo. Contudo, é possível encontrar soluções aproximadas, heurísticas e soluções ótimas de instâncias pequenas para os problemas da árvore de Steiner.

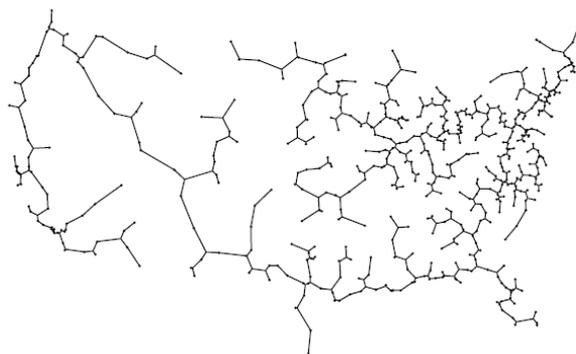
Esta seção tem por objetivo definir e detalhar as seguintes variantes do problema das árvores de Steiner:

- **Árvore de Steiner Euclidiana Mínima:** problema que busca a menor árvore Euclidiana que interconecta um conjunto de pontos.
- **Árvore de Steiner Euclidiana Mínima com Desvio de Obstáculos:** problema que busca a menor árvore Euclidiana que interconecta um conjunto de pontos em um ambiente com obstáculos.
- **Árvore de Steiner em Grafo:** problema que tem por objetivo encontrar a menor árvore que interconecta um subconjunto de vértices de um grafo.
- **Árvores de Steiner com o Número Mínimo de Pontos de Steiner e Tamanho Máximo de Aresta:** problema cujo objetivo é encontrar a árvore de Steiner com o menor número de pontos de Steiner. Sendo as arestas da árvore limitadas por um tamanho máximo.

Para conhecer outras variantes do problema da árvore de Steiner, são sugeridos os trabalhos de Du & Hu [2008], e Hwang et al. [1992].

### 3.1.1 Árvore de Steiner Euclidiana Mínima

#### 3.1.1.1 Definição e Propriedades



**Figura 3.1.** Exemplo de uma árvore de Steiner Euclidiana. A árvore de Steiner apresentada conecta um conjunto de 532 cidades dos Estados Unidos da América. (Exemplo retirado do trabalho de Warme et al. [2000])

O problema da árvore de Steiner Euclidiana mínima (EST, do inglês *Euclidean Steiner Tree Problem*) busca a menor árvore que interconecta, no espaço Euclidiano,

um conjunto de pontos terminais. Em uma EST, pontos de Steiner são gerados e inseridos a solução com o objetivo de reduzir o custo total da árvore. A Figura 3.1 apresenta um exemplo de uma árvore de Steiner Euclidiana. Formalmente o problema pode ser definido como:

**Definição 1.** Dado um conjunto de pontos  $\mathcal{C} = \{c_1, c_2, \dots, c_m\}$  no espaço Euclidiano. A solução é uma árvore de Steiner  $\mathcal{T}$  que interconecta todos os  $\mathcal{C}$ . O objetivo é encontrar a menor árvore Euclidiana que interconecta  $\mathcal{C}$ .

Matematicamente, o problema da EST possui as seguintes propriedades:

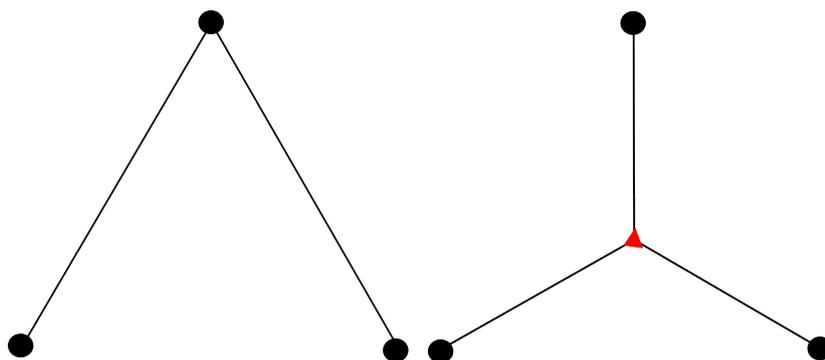
- Os pontos de Steiner são incidentes de exatamente três arestas, sendo o ângulo entre essas arestas  $120^\circ$ .
- Duas arestas nunca se cruzam.
- Se a árvore possui  $n$  pontos terminais, o número máximo de pontos de Steiner dessa árvore é  $n - 2$ .
- Uma EST é composta pela união de componentes completos. Um componente completo é uma árvore de Steiner de um subconjunto dos pontos terminais.

Um problema relacionado, e que possui uma formulação similar a EST, é o problema da árvore geradora mínima (MST, do inglês *Minimal Spanning Tree*). Uma MST procura a menor árvore que interconecta um conjunto de pontos, entretanto uma MST não insere pontos a solução com o objetivo de reduzir o tamanho da árvore. Além disso, enquanto a MST possui algoritmos de custo polinomial, sendo os mais conhecidos os algoritmos de Prim e Kruskal (Cormen et al. [2009]), uma EST é da classe de complexidade NP-completo.

A maior razão de ganho de uma EST em comparação com uma MST é (Du & Hwang [1992]):

$$\frac{|EST(P)|}{|MST(P)|} \geq \frac{\sqrt{3}}{2} \approx 0.866$$

A figura 3.2 apresenta um caso onde a razão entre EST e a MST é  $\frac{\sqrt{3}}{2}$ .



**Figura 3.2.** Comparação da solução de uma MST (árvore da esquerda) com a solução de uma EST (árvore da direita). Na figura, os pontos terminais são representados pelos círculos e os ponto de Steiner por um triângulo.

### 3.1.1.2 Algoritmos

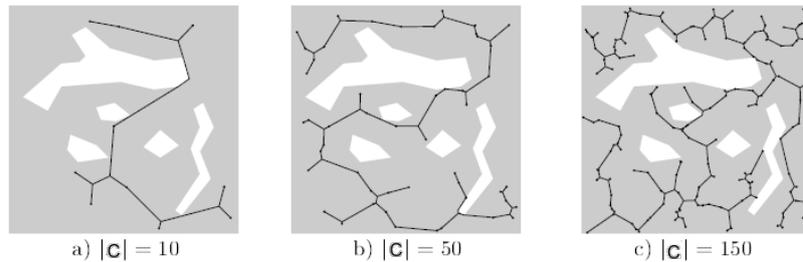
O primeiro algoritmo exato para resolver o problema da EST foi proposto por Melzak [1961]. Na metodologia, o objetivo é encontrar a menor árvore de Steiner para todas as topologias possíveis, selecionando a menor árvore como solução para o problema da EST. Nesse algoritmo, uma topologia é formada pela ordem de seleção dos pontos terminais.

Já no algoritmo proposto por Winter [1985], o algoritmo mais eficiente para o problema encontrado na literatura, duas fases são utilizadas para criação da árvore. Primeiro, todos os componentes completos são gerados para todos os subconjuntos de pontos terminais. Em uma segunda fase, a EST é construída a partir da concatenação dos componentes completos gerados, sendo a menor árvore produzida a solução final. Na fase de geração, o algoritmo proposto por Melzak [1961] é utilizado, entretanto na metodologia proposta, antes da geração dos componentes completos é verificado se a topologia poderá fazer parte da EST. Essa verificação poda o máximo possível de componentes completos. Sendo a segunda fase do algoritmo, a concatenação, o gargalo da abordagem.

Na fase de concatenação, como mais de um componente completo possui um mesmo terminal, é necessário utilizar um método não trivial para criar a menor árvore. Para isso, um algoritmo de *backtracking* pode ser utilizado, entretanto, como proposto em Warne [1998], um algoritmo para o problema da árvore geradora mínima em um hipergrafo resolve o mesmo problema e apresenta uma solução mais eficiente. Nessa solução, uma árvore de Steiner pode ser gerada para instâncias contendo mais de 2000 pontos terminais.

### 3.1.2 Árvore de Steiner Euclidiana Mínima com Desvio de Obstáculos

#### 3.1.2.1 Definição e Propriedades



**Figura 3.3.** Exemplo de uma árvore de Steiner Euclidiana Mínima com Desvio de Obstáculos. A árvore de Steiner apresentada conecta 10, 50 e 150 pontos respectivamente. (Exemplo retirado do trabalho de Zachariasen & Winter [1999])

Apesar do problema da EST encontrar a menor árvore interconectando o conjunto de pontos terminais, sua utilização prática é limitada devido ao problema não considerar obstáculos. Uma formulação que constrói uma árvore de Steiner no espaço Euclidiano considerando obstáculos é chamada de problema da árvore de Steiner Euclidiana Mínima com Desvio de Obstáculos (ESMTO, do inglês *Obstacle-Avoidance Minimal Euclidean Steiner Tree*), definido como:

**Definição 2.** Dado um conjunto de pontos  $\mathcal{C} = \{c_1, c_2, \dots, c_m\}$  no espaço Euclidiano, e um conjunto de obstáculos  $\Omega$ . O objetivo é encontrar a menor árvore Euclidiana que interconecta  $\mathcal{C}$  evitando os obstáculos  $\Omega$ .

A Figura 3.3 apresenta um exemplo de uma árvore de Steiner Euclidiana com desvio de obstáculos.

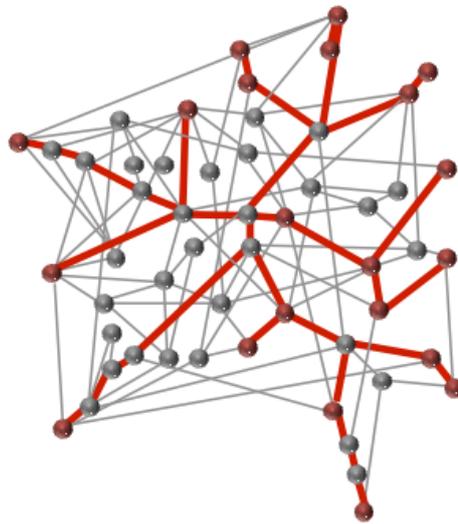
Apesar da similaridade, as propriedades da ESMTO são diferentes da EST. Primeiro, nessa definição, os pontos de Steiner podem ter de duas a três arestas incidentes, não necessariamente formando ângulos de  $120^\circ$ . Segundo, dado a existência de obstáculos, o número máximo de pontos de Steiner não está limitado pelo número de pontos terminais. Apesar das diferenças, a definição e propriedades relacionadas aos componentes completos continuam válidas.

#### 3.1.2.2 Algoritmos

Dado que a propriedade relacionada aos componentes completos é válida para o problema, Zachariasen & Winter [1999] propõe um algoritmo similar ao algoritmo apresentado por Winter [1985]. Sendo esse composto por uma fase de geração de componentes

completos e outra fase de concatenação. Como algumas propriedades dos problemas EST e ESMTO são diferentes, os autores propõem outras técnicas para a poda dos componentes completos. Além disso, ainda na fase de geração, para que o algoritmo gere todos os componentes completos que a árvore mínima pode conter, os pontos pertencentes aos obstáculos também são considerados pontos terminais. Na fase posterior, na concatenação, os pontos pertencentes aos obstáculos são considerados pontos de Steiner.

### 3.1.3 Árvore de Steiner em Grafo



**Figura 3.4.** Exemplo de uma árvore de Steiner em grafo. A árvore de Steiner apresentada conecta 23 pontos terminais (Pontos de cor mais escura). (Exemplo retirado do trabalho de Bayati et al. [2008])

O problema da árvore de Steiner em grafo (STG, do inglês *Steiner Tree On Graph*) tem por objetivo encontrar a menor árvore que interconecta um subconjunto de vértices de um grafo. Formalmente o problema é definido como:

**Definição 3.** Dado um grafo  $G = (V, A)$ , e um conjunto de vértices terminais  $\mathcal{C} \subseteq V$ . O objetivo é encontrar a menor árvore no grafo  $G$  que interconecta os vértices  $\mathcal{C}$ .

A Figura 3.4 apresenta um exemplo de uma árvore de Steiner em grafo.

Para esse problema, uma MST, na definição para o problema em grafos, possui uma formulação diferente da STG. Na MST o objetivo é encontrar a menor árvore que interconecta todos os vértices  $V$ , enquanto que na definição da STG apenas um

subconjunto dos vértices  $\mathcal{C} \subseteq V$  necessita ser conectado. Além disso, o problema da STG é provado ser da classe NP-completo (Karp [1972]).

### 3.1.3.1 Algoritmos

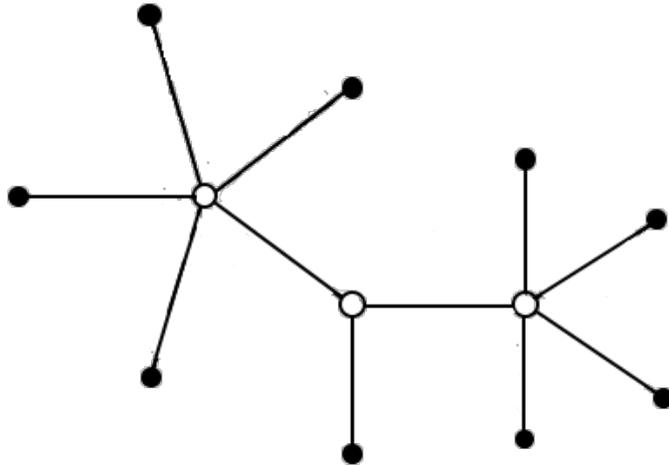
A solução da MST, considerando o subgrafo completo  $G = (\mathcal{C}, A')$ , onde  $A'$  representa as arestas conectando os clientes  $\mathcal{C}$ , é uma solução 2 – *aproximado* para o problema da STG (Gilbert & Pollak [1968]). Caso o subgrafo conectando  $\mathcal{C}$  não seja completo, sem perda de generalidade, um grafo completo pode ser construído a partir do grafo original. Nesse novo grafo, o custo das arestas interconectando os vértices terminais será o custo do menor caminho que interconecta esses vértices no grafo original.

Diversas aproximações já foram propostas para o problema da STG (Robins & Zelikovsky [2005]). A maioria dessas soluções utiliza alguma abordagem para substituir componentes completos para subconjuntos de vértices terminais. Nessa definição do problema da árvore de Steiner, um componente completo é definido como uma árvore que interconecta um subconjunto de terminais  $\mathcal{C}$ . A abordagem que apresenta a melhor aproximação é o algoritmo de Robins & Zelikovsky [2005], sendo o algoritmo apresentado pelo trabalho uma solução 1.55-aproximado para o problema. Para conseguir essa aproximação o algoritmo proposto seleciona os componentes completos que apresentam a melhor relação entre o ganho e a perda ao adicionar um componente completo a uma solução. O ganho é definido como a diferença entre o custo da árvore atual e da árvore com o novo componente completo. A perda é o tamanho da MST que conecta os vértices do componente completo.

### 3.1.4 Árvore de Steiner com o Número Mínimo de Pontos de Steiner

A árvore de Steiner com o número mínimo de pontos de Steiner (STP-MSP, do inglês *Steiner Tree Problem for the Minimal number of Steiner Points*) busca pela árvore com o menor número de pontos de Steiner interconectando um conjunto de pontos terminais. Além disso, no problema as arestas da árvore são limitadas por um valor máximo. A Figura 3.5 apresenta um exemplo de uma árvore de Steiner com o número mínimo de pontos de Steiner. problema pode ser definido como:

**Definição 4.** Dado um conjunto de pontos no espaço Euclidiano  $\mathcal{C} = \{c_1, c_2, \dots, c_m\}$ , e  $d$ , o tamanho máximo permitido de uma aresta. O objetivo é encontrar a árvore de Steiner Euclidiana interconectando os pontos terminais com o menor número de pontos de Steiner e com arestas de tamanho máximo  $d$ .



**Figura 3.5.** Exemplo de uma árvore de Steiner com o Número Mínimo de Pontos de Steiner. A árvore de Steiner apresentada conecta 9 pontos terminais utilizando 3 pontos de Steiner. (Exemplo retirado do trabalho de Lin & Xue [1999])

Uma STP-MSP possui as seguintes propriedades:

- Duas arestas não se cruzam.
- Duas arestas se encontram formando um ângulo de pelo menos  $60^\circ$
- Em uma árvore ótima, se duas arestas formam um ângulo de  $60^\circ$ , essas arestas possuem o mesmo tamanho.

Diferente das outras árvores de Steiner apresentadas, uma STP-MSP não é diretamente comparável com uma MST.

### 3.1.4.1 Algoritmos

Lin & Xue [1999] apresenta o primeiro trabalho com uma prova de um algoritmo aproximado para o problema da STP-MSP. Nesse trabalho, uma MST com suas arestas repartidas, de forma que essas não tenham tamanho superior a  $d$ , é provado ser uma solução 5-aproximado para o problema.

Já, em um trabalho posterior, Cheng et al. [2008] apresenta algoritmos com aproximações 3 e 2.5 para o problema. Para conseguir uma aproximação 3, a solução inicialmente conecta todos os pontos que estão a uma distância menor que  $d$ , os pontos restantes são conectados utilizando o algoritmo de Prim para construção da MST, sendo as arestas maiores que  $d$  repartidas. Já na solução 2.5 aproximado, um algoritmo que determina a árvore geradora mínima em um hipergrafo (MSTHG, do inglês *Minimal Spanning Tree on a Hypergraph*) é utilizado. Para isso, inicialmente é construído

todas as MSTHGs para os subconjuntos de 3 terminais, sendo as arestas das MSTHGs que possuem tamanho maior que  $d$  repartidas. A solução final é a menor árvore da concatenação dessas MSTHG.

## 3.2 Propagação de Sinal

A partir da década de 80, com o surgimento das primeiras redes sem fio comerciais, o interesse em prever os efeitos e características da propagação de sinal de rádio aumentou consideravelmente. Hoje, modelos de propagação de sinal analíticos e empíricos podem ser encontrados, modelando fenômenos de propagação para ambientes internos e externos.

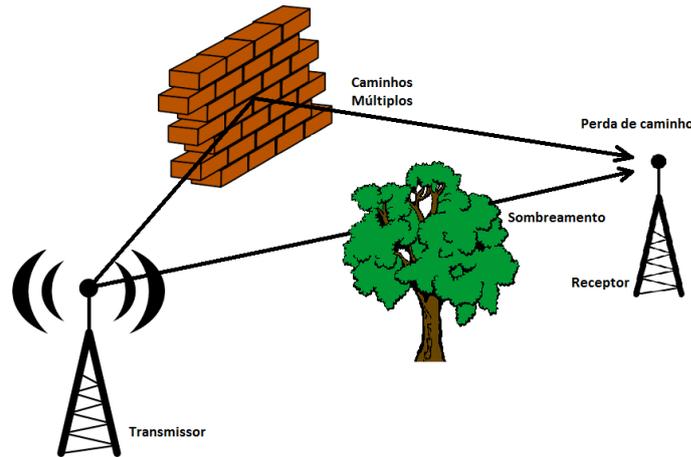
Os fenômenos da propagação de sinal são similares aos fenômenos que ocorrem na propagação de outras ondas eletromagnéticas. Por exemplo, efeitos como refração, reflexão e difração, observados na propagação da luz, podem ser verificados na propagação das ondas de rádio. Assim, ondas que são emitidas por um transmissor sofrem diversos efeitos ao percorrerem um ambiente. Dessa forma, dependendo das características do ambiente, o decaimento da intensidade de sinal pode não ser monotônico decrescente em relação a distância do transmissor.

A propagação de sinal em um ambiente apresenta os seguintes fenômenos:

- **Perda do caminho (*path loss*):** caracterizado pela atenuação da intensidade de uma onda eletromagnética que propaga pelo um meio em relação a distância percorrida.
- **Sombreamento (*shadowing*):** efeito da variação da intensidade de sinal devido a presença de objetos obstruindo o caminho entre o transmissor e receptor.
- **Caminhos múltiplos (*multi-path*):** caracterizado pela interferência causada por duas ou mais versões de uma mesmo sinal que chegam a antena do receptor em momentos diferentes.

A Figura 3.6 apresenta os fenômenos de propagação mencionados. Dados esses fenômenos, o espaço de propagação de sinal não é Euclidiano, significando que a desigualdade triangular não é válida.

A seguir serão apresentados alguns modelos de propagação que modelam a alguns dos fenômenos de propagação de sinal.



**Figura 3.6.** A perda de caminho ocorre pela propagação da onda no meio. O sombreamento ocorre pela presença de obstáculos, na figura uma árvore. Os caminhos múltiplos ocorrem pelas múltiplas ondas do mesmo sinal que chegam ao receptor, na figura causado pela reflexão do sinal na parede.

### 3.2.1 Modelo de propagação em espaço livre

O modelo de propagação em espaço livre considera um ambiente ideal, onde a linha de visão entre o transmissor e receptor não é obstruída. Nesse modelo, apenas o efeito de perda de caminho é considerado, assim, a intensidade de sinal se propaga de forma uniforme e com decaimento quadrático. A equação de Friis, representando o cenário descrito, e que expressa a potência recebida por um receptor a uma distância  $d$  do transmissor, é dado por:

$$P_r = \frac{P_t G_r G_t \lambda^2}{(4\pi)^2 d^2 L}$$

Respectivamente,  $P_r$  e  $P_t$  são as potências de recepção e transmissão,  $G_r$  e  $G_t$  são os ganhos das antenas de recepção e transmissão,  $\lambda^1$  está relacionada a frequência do sinal e  $L$  é um fator de perda não relacionado a propagação de sinal (perda que causada pelo equipamento). O ganho de sinal de uma antena está relacionado a sua abertura.<sup>2</sup>

Dado o modelo de Friis para o cálculo da potência, a perda de caminho, que pode ser calculada como a diferença em dB entre a potência de transmissão e recepção é dado por:

$$PL(dB) = 10n \log \frac{P_t}{P_r} = -10 \log \left[ \frac{P_t G_r G_t \lambda^2}{(4\pi)^2 d^2} \right]$$

<sup>1</sup> $\lambda = \frac{c}{f}$ , onde  $c$  é a velocidade da luz em metros por segundo e  $f$  a frequência da onda em Hertz.

<sup>2</sup> $G = \frac{4\pi A_e}{\lambda^2}$ , onde  $A_e$  é a abertura efetiva da antena e está relacionado ao tamanho da antena.

Dado um ambiente aberto, o valor dado por  $PL(dB)$  pode ser utilizado para indicar a conectividade entre dois dispositivos.

### 3.2.2 Modelo de propagação log-distância

Tanto os modelos de propagação teóricos quanto os baseados em medições predizem que, para ambientes internos e externos, a atenuação média do sinal decai de forma logarítmica em relação a distância entre o transmissor e receptor. Sendo que para diferentes ambiente existe é um expoente de decaimento  $n$  diferente, dessa forma, o modelo de propagação log-distância determina que a perda de caminho média pode ser representada como:

$$PL(dB) = PL(d_0) + 10n \log \frac{d}{d_0}$$

O expoente de perda de caminho  $n$  pode se ser obtido de forma experimental ou por meio de tabelas de materiais. Nessa formula também é necessário a medida do sinal  $PL(d_0)$  a uma distância  $d_0$ . A distância  $d_0$  varia de acordo com o ambiente e as características do sinal transmitido, podendo variar de um a centenas de metros.

Para considerar o efeito de sombreamento, uma variável aleatória normal  $X_\alpha$  com um desvio padrão  $\alpha$  pode ser adicionada a equação do modelo de propagação log-distância. O desvio padrão  $\alpha$  é determinado pelas características dos ambiente, similar ao expoente de perda de caminho. Ambos os valores podem ser obtidos através de tabelas construídas a partir de medições reais. A equação final é representada por:

$$PL(dB) = PL(d_0) + 10n \log \frac{d}{d_0} + X_\alpha$$

### 3.2.3 Modelo de fator de atenuação

Um modelo mais completo, que apresenta a atenuação do sinal causada pelos obstáculos existentes entre o transmissor e receptor é dado pelo modelo de fator de atenuação:

$$PL(dB) = PL(d_0) + 10n \log \frac{d}{d_0} + \sum PAF[dB]$$

Esse modelo adiciona  $\sum PAF[dB]$  ao modelo de log-distância. Sendo que  $\sum PAF[dB]$  apresenta a atenuação de sinal que cada obstáculo no caminho entre os transmissor e receptor causa. Para calcular  $\sum PAF[dB]$ , um linha é traçada entre o transmissor e receptor, e para cada obstáculo que cruza a linha, um fator de atenuação é somado ao valor de atenuação. Os valores de atenuação causada pelos obstáculos

depende do material.

# Capítulo 4

## Metodologia

Este capítulo apresenta a abordagem proposta para interconectar um conjunto de clientes em uma rede ad-hoc utilizando robôs roteadores. A abordagem proposta é composta por dois passos. Primeiro, dado o mapa do ambiente contendo os clientes e obstáculos, uma árvore de Steiner Euclidiana é calculada. Por definição essa árvore conecta os clientes com a menor distância possível. Segundo, utilizando a árvore de Steiner construída no planejador global de caminhos, os robôs, através de uma máquina de estado finita comunicante, criam de forma autônoma uma rede ad-hoc interconectando os clientes.

### 4.1 Modelagem

Uma das propriedades pouco abordadas nos trabalhos de criação de redes ad-hoc utilizando robôs roteadores é o número de robôs necessários para criar uma solução. Assim, para reduzir o número de robôs necessários para uma solução, neste trabalho são utilizadas árvores de Steiner Euclidianas, essas retornam a árvore de menor distância interconectando um conjunto de pontos. O objetivo disso é limitar a alocação dos robôs à menor árvore que interconecta os clientes, reduzindo o número de robôs necessários para criar a solução.

A árvore de Steiner Euclidiana mínima (EST) não pode ser utilizada no problema proposto dado que não considera obstáculos. Assim, neste trabalho são utilizadas as árvores de Steiner Euclidiana com desvio de Obstáculos (ESMTO) (Seção 3.1.2) e árvore de Steiner em grafos (STG) (Seção 3.1.3). A STG, apesar de não retornar a menor árvore interconectando os clientes, é capaz de tratar obstáculos através do grafo de entrada. Neste trabalho, a STG é utilizada como uma comparação para a ESMTO. Além disso, no problema desta dissertação, os pontos terminais definidos nas árvores

de Steiner representam os clientes.

Apesar da ESMTO apresentar a menor árvore que interconecta os clientes, a melhor alocação sobre essa árvore poderá não ser a alocação ótima. A solução ótima para o problema, pela própria definição, seria dado pela árvore STP-MSP (Seção 3.1.4), entretanto os algoritmos existentes para esse problema de otimização não consideram ambientes com obstáculos, não sendo possível apresentar uma solução mesmo aproximada para o problema. Além disso uma solução ótima para o problema deve considerar, além de obstáculos, os fenômenos de propagação de sinal, onde o espaço não é Euclidiano, dificultando a criação de uma solução para o problema da STP-MSP para ambientes reais. Com isso, é importante mencionar que a abordagem que é proposta não depende da árvore de Steiner Euclidiana, sendo apenas necessário uma árvore que interconecta os clientes.

Dado que o caminho onde os robôs podem ser alocados é definido por uma árvore de Steiner, é necessário um método para criar a rede autonomamente. Neste trabalho, é utilizada uma máquina de estados comunicante estendida para fazer a alocação dos robôs. Essa máquina de estados é construída utilizando uma técnica de histerese, possibilitando que a alocação dos robôs considere os fenômenos de propagação de sinal.

A seguir serão detalhados o planejador de caminhos global utilizando a árvore de Steiner e a máquina de estados utilizada para fazer a alocação dos robôs roteadores.

## 4.2 Planejador de caminho

O primeiro passo é determinar uma árvore para o planejador global, determinando o local onde os robôs podem mover pelo ambiente, e por consequência limitando o formato da rede ao formato da árvore utilizada.

Neste trabalho, um planejador local também é utilizado mas com o objetivo de evitar os obstáculos que não foram considerados pela árvore do planejador global (ex. robôs e pessoas).

A seguir são apresentados duas árvores de Steiner utilizadas como planejador global. Uma árvore de Steiner Euclidiana com desvio de obstáculos, e um árvore de Steiner em grafo. A árvore de Steiner em grafo é utilizada como comparação para solução do árvore de Steiner Euclidiana.

### 4.2.1 Árvore de Steiner Euclideana com Desvio Obstáculos

Como apresentado no Capítulo 3, a árvore de Steiner Euclideana com desvio de obstáculos (ESMTO) interconecta um conjunto de pontos, em um ambiente com obstáculos,

com a menor árvore Euclidiana possível. Nesse caso, os clientes são os pontos terminais que devem ser conectados a árvore, sendo adicionados pontos de Steiner à solução com o objetivo da redução da árvore.

Neste trabalho, para construir a ESMTO, a abordagem apresentada por Zachariasen & Winter [1999] é utilizada. De forma geral, a abordagem consiste em gerar todos os componentes completos para os subconjuntos de nós terminais. Esses componentes completos são construídas evitando os obstáculos. Em uma etapa posterior, a menor árvore é gerada através da concatenação de componentes completos. Para concatenar os componentes completos, é utilizado um algoritmo de *backtrack*.

### 4.2.2 Árvore de Steiner em Grafos

Uma árvore de Steiner em grafo (STG) tem por objetivo encontrar a menor árvore que interconecta um subconjunto de vértices em um grafo. Neste trabalho, um grafo de visibilidade, construído a partir do ambiente de entrada e os clientes, é utilizado para gerar uma árvore de Steiner em grafo conectando os clientes. Para gerar esse grafo de visibilidade, é determinado o menor caminho que interconecta um par de clientes, cada caminho gerado é adicionado ao grafo de visibilidade.

A solução utilizada nesse trabalho é uma MST do subgrafo dos clientes. Essa é uma solução 2-aproximado para o problema. Não existe uma relação de aproximação entre a solução 2-aproximado da STG e a ESMTO.

## 4.3 Alocação dos Robôs Roteadores

O planejador de caminho é responsável por limitar o local onde os robôs podem ser alocados, sendo necessário alguma abordagem para fazer a alocação dos robôs, criando a rede ad-hoc interconectando os clientes. Nesta seção será apresentado a máquina de estados comunicante estendida (CEFSM, do inglês *Communicating Extended Finite State Machine*) utilizada pelos robôs para criar de forma autônoma a rede ad-hoc. Na abordagem proposta, a partir do compartilhamento de informações pela rede, cada robô executa uma instância da CEFSM.

A CEFSM pode ser definido formalmente como uma 6-tupla  $(S, I_0, E, f, O, V)$ , onde  $S$  representa o conjunto de estados,  $I_0$  é o estado inicial,  $E$  é um conjunto de eventos,  $f$  representa uma função de transição,  $O$  é um conjunto de sinais de saída,  $V$  é um conjunto de variáveis. A função de transição  $f$  retorna, dado o estado atual e um evento, um novo estado, um conjunto de sinais de saída  $O$  e um conjunto de variáveis  $V$ . Por exemplo,  $f(S_0, e_1) \rightarrow (S_1, o_1, o_2, (action_3))$  descreve que, dado um estado  $S_0$  e

um evento  $e_1$ , a saída é um estado  $S_1$  com um conjunto de sinais de saída  $o_1, o_2$  e com a ação  $action_3$  sendo executada. As variáveis  $V$  armazenam valores que também são utilizadas nos predicados.

A CEFSM pode ser representado visualmente através de um grafo, onde cada nó é um estado e as arestas representam as transições. As transições são tuplas Evento/Ação, quando um evento é acionado, uma transição ocorre e um ação correspondente é gerada. A Figura 4.1 mostra a CEFSM em uma representação de grafo.



Figura 4.1. Diagrama de estados da CEFSM.

A CEFSM é composto por 4 estados: *espera*, *move*, *desconectado*, e *conectado*. Os estados da CEFSM e seus papéis são apresentados na Tabela 4.1. Basicamente, no estado *espera* os robôs esperam até que seja detectado um cliente desconectado. Enquanto que no estado *move* o robô se move na direção do cliente desconectado mais próximo. O estado *desconectado* é utilizado para reconectar um robô a um cliente que foi desconectado pelo movimento do robô. Por último, o objetivo do estado *conectado* é estender a rede ad-hoc, provendo suporte para os outros robôs continuarem a criação da rede.

Estado	Papel
Espera	Espera para entrar na rede.
Move	Move em direção a cliente desconectado.
Desconectado	Move para trás para até conectar com a rede que foi perdida.
Conectado	Fica parado para fornecer ponto de conexão.

Tabela 4.1. Tabela de estado do CEFSM.

Os eventos e ações da CEFISM são apresentados nas Tabelas 4.3 e 4.2, respectivamente. Os eventos representam as mudanças na rede que irão ocorrer durante a execução da CEFISM. O consenso é utilizado para evitar condição de corrida.

Ação	Significado
Move	Move em direção ao cliente desconectado mais próximo pela árvore de Steiner.
Move para trás	Move em direção a um cliente que foi perdido.
Espera	Fica parado

**Tabela 4.2.** Tabela de Ações.

Evento	Significado
Todos clientes Conectados	Verifica se todos os clientes estão conectados
Desconectado de cliente	Algum cliente perdeu conexão com a rede.
Reconectado a cliente	O cliente foi reconectado a rede.
Pertence a solução	Irá ocorrer uma partição da rede se o robô deixar a rede.
Rota para cliente mudou	O cliente se reconectou à rede mas sua rota mudou.
Vence consenso	Vence a competição para o estado <i>conectado</i>
Perde consenso	Perde a competição para o estado <i>conectado</i>

**Tabela 4.3.** Tabela de Eventos.

São definidas três ações para os robôs *Move*, *Move para trás* e *Espera*. Respectivamente, essas ações são responsáveis por fazer os robôs seguirem na direção do cliente desconectado mais próximo, mover na direção de um cliente que a conexão foi perdida, e ficar em modo de espera.

Para avaliar os eventos, cada robô mantém uma matriz de adjacência representando a rede. Para manter essa matriz de adjacência, os robôs avaliam e compartilham em mensagens periódicas suas conexões locais. A conectividade local é verificada através da medição do RSS (do inglês, Received Signal Strength) juntamente com uma técnica de histerese. Lembrando que uma técnica de histerese utiliza dois limiares para separar dois estados distintos.

A Figura 4.3 apresenta o funcionamento da CEFISM (a seguir será utilizado (#) para indicar a sub-figura que representa a ação descrita). Inicialmente, todos os robôs começam juntos e podem comunicar entre si. Em seguida, todos os robôs começam no estado *espera* (**1**). Quando um robô detecta um cliente desconectado, ele muda para o estado *move* (**2**). No estado *move*, o robô vai para o cliente desconectado mais próximo (**3-4**). Eventualmente, enquanto se movimenta para o cliente desconectado,

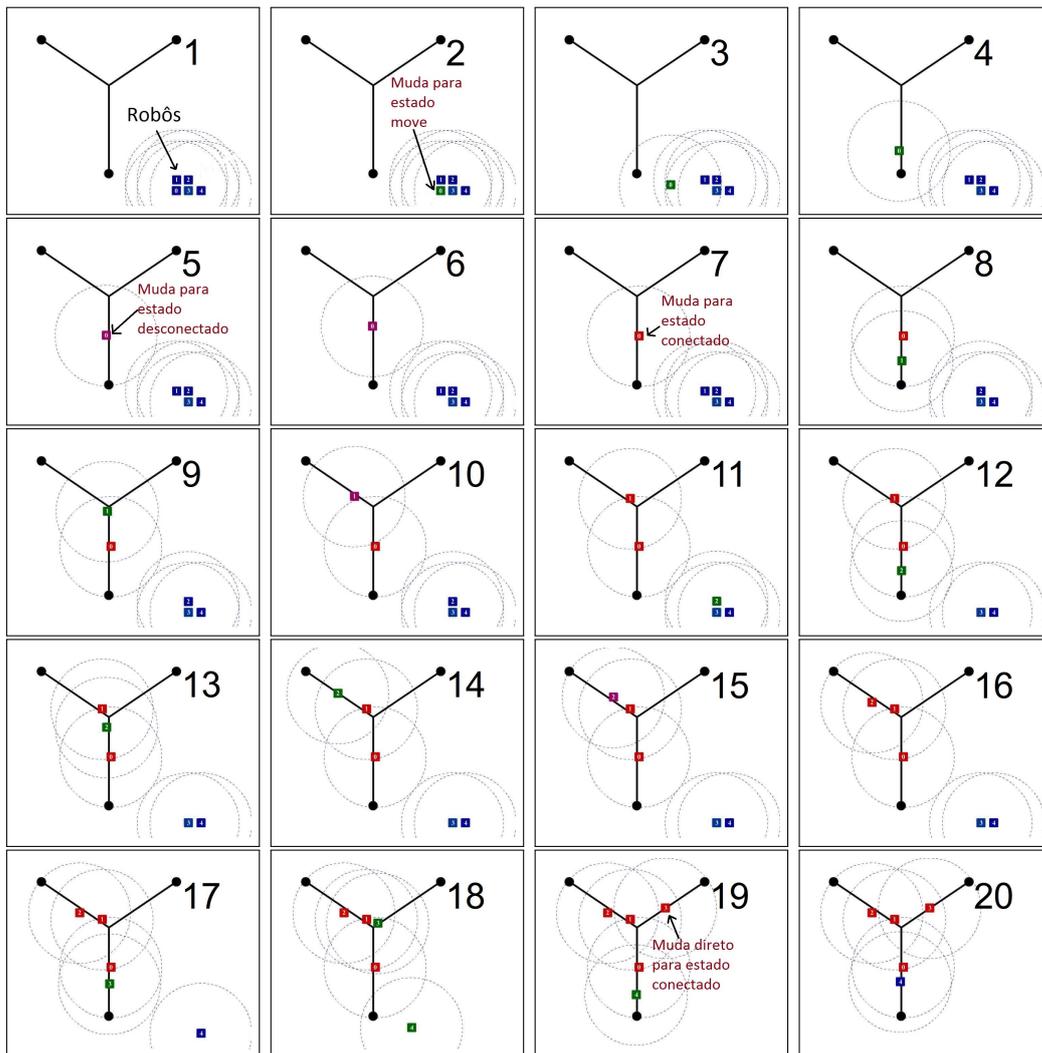
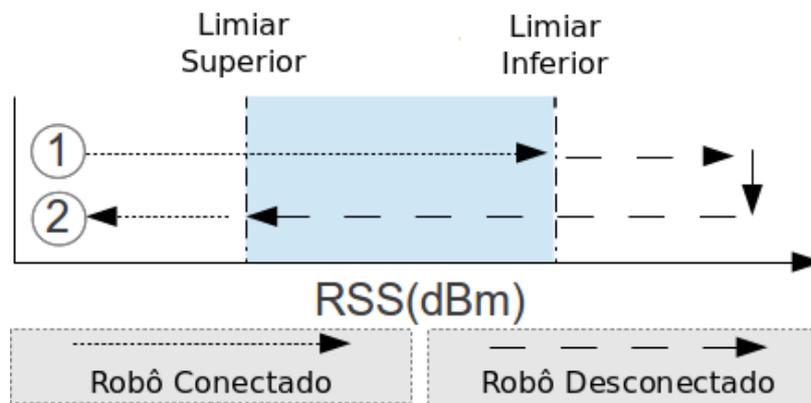


Figura 4.2. Representação do funcionamento da CEFSM.

o robô pode perder a conexão com um cliente já conectado. Quando isso acontece, o robô muda para o estado *desconectado* e abandona o seu último objetivo (5).

No estado *desconectado*, o robô se move para trás, para o cliente perdido (6). Quando este cliente for reconectado, o robô muda para o estado *conectado*, mantendo a rede sem fio e provendo suporte para outros robôs interligarem o restante dos clientes desconectados (7). O processo de conexão e desconexão é mostrado na Figura 4.3. No entanto, mais de um robô pode estar no estado *desconectado* ao mesmo tempo e próximos uns dos outros. Para evitar condição de corrida, os robôs executam um algoritmo de consenso. Os robôs que perdem o consenso, mudam para o estado *move*. O algoritmo de consenso é apresentado a seguir.

Antes da execução da CEFSM, cada robô é associado a um número de identificação (ID) único. Ao executar o algoritmo de consenso, o robô analisa se há um outro



**Figura 4.3.** O robô no estado *move* (1) move em direção ao cliente desconectado até perder conexão (limiar inferior) mudando para o estado *desconectado*. No estado *desconectado* o robô move para trás até reconectar com a rede perdida (limiar superior), mudando para o estado *conectado*(2).

robô perto e nos estados *desconectado* ou *move*. Quando este não é o caso, o robô ganha o consenso. Se houver outros robôs que atendam às condições, o robô avalia os IDs destes outros robôs. O robô com o menor ID ganha o consenso.

A CEFMSM possui um caso onde um robô no estado *desconectado* reconecta-se a um cliente perdido através de outro robô, neste caso, uma nova rota para este cliente é adicionado na rede, permitindo que o robô mude para o estado *move*.

O processo de alocação continua para cada robô (8-18). No final, irá permanecer apenas um cliente para ser conectado na rede. Neste caso, o último robô necessário para ligar este cliente muda diretamente do estado *move* para *conectado* (19). Para evitar concorrência, o esquema de consenso é usado quando o robô muda para o estado *conectado*. Os robôs remanescentes, que não estão na solução, mudam para o estado *espera*. Os robôs no estado *espera* são utilizados para robustez. No caso de um robô no estado *conectado* falhar, um cliente desconectado será detectado, permitindo que os robôs no estado *espera* executem a CEFMSM, re-estabelecendo a rede.

O algoritmo proposto é descentralizado, completo e considera os fenômenos de propagação de sinal. É descentralizado dado que não utiliza um agente centralizador controlando os robôs e cada robô pode gerar sua árvore de Steiner dado que os algoritmos utilizados são determinísticos. É completo no sentido que ele para quando a rede está concluída. Neste caso, os robôs param nos estados *espera* ou *conectado*. Se a rede não está concluída e todos os robôs estão no estado *conectado*, significa que mais robôs são necessários para concluir a solução. O algoritmo também lida com fenômenos de propagação de sinal: perda de caminho, sombreamento, e multi-caminho. Isto ocorre

uma vez que é utilizado a medida real do RSS para avaliar as conexões.

## 4.4 Comunicação

A capacidade de estimar o estado da rede apresenta um papel importante neste trabalho. No sistema proposto, para estimar o estado da rede, informações do ID, estado atual da CEFMS, posição no mapa e lista de vizinhos são compartilhados pelos robôs. As mensagens contendo essas informações são disseminadas na rede através de mensagens *broadcast*. A Tabela 4.4 mostra as informações compartilhadas e o propósito dessas.

Campo	Propósito
Id	Identificação do robô que envia a mensagem.
Número da mensagem	Identificação de mensagens antigas
Estado	Estado atual da CEFMS
Vizinhos	Lista de vizinhos do robô na rede.
Posição	Posição atual no mapa.

**Tabela 4.4.** Tabela de informações compartilhadas.

Os robôs utilizam uma matriz de adjacência, construída a partir das informações compartilhadas, para determinar os robôs que estão conectados na rede. Isso permite que cada robô mantenha uma memória da estrutura de rede, sendo possível determinar os eventos apresentados na tabela 4.3.

## 4.5 Análise Teórica

Nesta seção, dado certas suposições, será provado que o algoritmo CEFMS necessita de um número máximo de robôs para estabelecer uma rede ad-hoc conectando o conjunto de clientes. Para isso, comparamos o número de robôs necessário para formar a rede com o tamanho da árvore de Steiner.

Para a prova fazemos duas pressuposições. Essas pressuposições são feitas apenas para prova, não afetando o funcionamento do algoritmo. Primeiro, o sinal sofre com o problema de perda de caminho, sombreamento, e multi caminho. Dado que prever todos os potenciais refletores e refratores é intratável em prática, omitimos o sombreamento e o multi caminho. Se levarmos em conta o sombreamento e multi-caminho, então não teríamos um espaço métrico, tornando difícil uma análise matemática para o problema. A partir do modelo propagação em espaço livre, o RSS é monotonamente decrescente. Quanto maior for a distância entre os dispositivos, menor é o RSS.

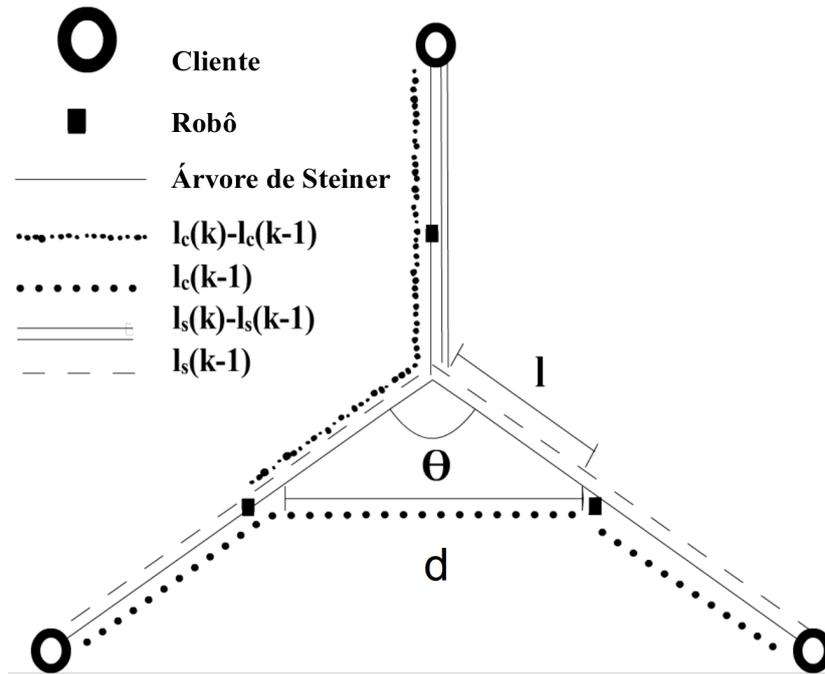
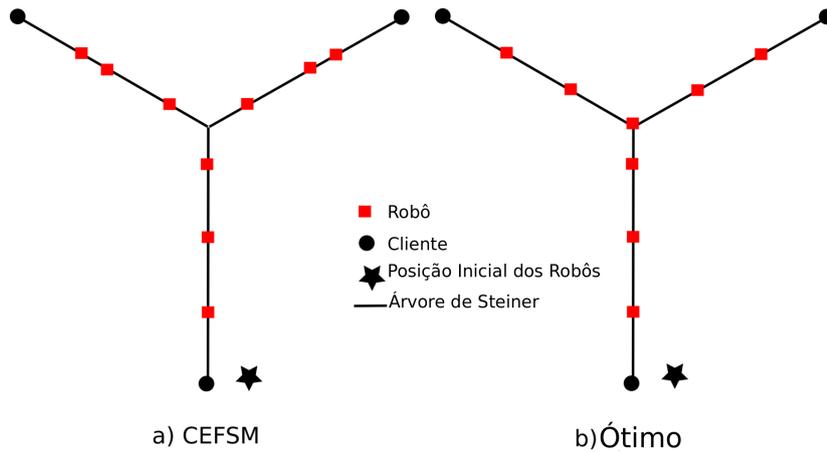


Figura 4.4. Ilustração de variáveis.

Segundo, é pressuposto que a distância de comunicação de rádio  $d$  é muito menor do que a distância entre clientes  $D$ , ou seja,  $D \gg d$ . Essa suposição é feita para que seja fornecido alguns limites para a análise, sendo o algoritmo válido sem essa suposição.

Uma característica da abordagem proposta, que é importante ser discutida, é que a CEFMSM não retorna sempre a melhor solução para o problema. A Figura 4.5 mostra um exemplo onde a CEFMSM usa mais robôs do que o ótimo. No algoritmo proposto, robôs podem ficar perto uns dos outros quando um cliente é conectado a rede, enquanto que na solução ótima esta configuração pode não ocorrer. Apesar disso, o algoritmo apresenta um número máximo de robôs necessário para criar a rede. A implementação ótima em cenários reais não é viável já que a distância de conexão pode variar devido a fenômenos de propagação de sinal. Por outro lado, a CEFMSM considera os fenômenos de propagação de sinal, e, para a maioria das conexões criadas, maximiza a distância entre os pares de robôs.

Considerando  $l_s(k)$  o comprimento da árvore Steiner para conectar  $k$  clientes e  $l_c(k)$  o comprimento necessário para o CEFMSM conectar  $k$  clientes, e definindo  $\theta$  como o ângulo do ponto de Steiner para conectar duas folhas de uma árvore. Quando não há obstáculos, este ângulo é de  $120^\circ$ . Por último, definindo  $l$  como o lado do triângulo de base  $d$  e ângulo interno  $\theta$ . A Figura 4.4 ilustra essas variáveis em um exemplo com 3 clientes. A Tabela 4.5 apresenta as variáveis utilizadas na análise teórica e seus



**Figura 4.5.** CEFMSM Contra Alocação Ótima.

significados.

Variável	Significado
$l$	Lado do triângulo de base $d$ e ângulo $\theta$
$\theta$	Ângulo do ponto de Steiner para conectar duas folhas da árvore
$d$	Distância de comunicação
$C$	Número de clientes
$l_c(k)$	Comprimento necessário para o CEFMSM conectar $k$ clientes
$l_s(k)$	Comprimento da árvore de Steiner para conectar $k$ clientes

**Tabela 4.5.** Variáveis utilizadas na análise teórica.

Supondo que  $k - 1$  clientes estão conectados na rede,  $l_c(k - 1)$  representa o tamanho da CEFMSM para conectar  $k - 1$  clientes, e  $l_s(k - 1)$  representa o tamanho da árvore de Steiner para conectar  $k - 1$  clientes. A diferença entre  $l_c(k) - l_c(k - 1)$  representa o tamanho necessário para conectar o  $k$ -ésimo cliente utilizando a CEFMSM. A diferença  $l_s(k) - l_s(k - 1)$  representa o tamanho necessário para conectar o  $k$ -ésimo cliente utilizando a árvore de Steiner.

Existem três casos para serem considerados. No caso (1),  $\theta = 120^\circ$ . A Figura 4.4 mostra o pior caso, onde a diferença entre a conexão atual e a conexão que deverá ser feita é a maior possível. Para conectar o cliente  $k$ , considerando que  $k - 1$  clientes estão conectados pela árvore de tamanho  $l_c(k - 1)$ , é necessário conectar  $l_s(k) - l_s(k - 1)$  somando a  $l$ , sendo que, por propriedades geométricas,  $l = \sqrt{3}/4 * d$ . Assim, como  $l < d$ , no pior caso, além dos  $\lfloor \frac{l_s(k) - l_s(k-1)}{d} \rfloor + 1$  robôs, é necessário 1 robô adicional para conectar o  $k$ -ésimo cliente.

No caso 2,  $\theta < 120^\circ$ . Como  $l < \sqrt{3}/4 * d$  o resultado anterior ainda é válido.

No caso 3,  $\theta > 120^\circ$ . Para conectar o  $k$ -ésimo cliente, é necessário adicionar

$l_c(k) - l_c(k - 1) = l_s(k) - l_s(k - 1) + l$  a rede. Sendo necessário preencher o buraco deixado na conexão anterior, entretanto, se um buraco foi deixado por uma conexão anterior então essa conexão utilizou menos robôs, assim  $l_c(k) = l_s(k) - l$ . Não sendo necessário mais robôs do que o estimado pela árvore de Steiner para conectar o  $k$ -ésimo cliente.

No pior caso, nos três casos, 2 robôs extras são necessário para cada cliente além do tamanho da árvore de Steiner dividido por  $d$ . Assim, considerando  $C$  clientes, para criar a rede são necessários  $\lfloor l_s(C)/d \rfloor + 2C$  robôs.

Portanto, é provado um limite máximo de robôs roteadores necessários para o algoritmo proposto criar uma rede ad-hoc interconectando um conjunto de clientes.



# Capítulo 5

## Resultados

Os experimentos reais e simulados foram realizados utilizando o framework de robótica ROS<sup>1</sup>. Na implementação da abordagem proposta, foram utilizados os pacotes de navegação e localização presentes no ROS. A máquina de estados proposta foi implementada como um pacote do ROS, e o navegador global do pacote de navegação, que pode ser alterado a partir da criação de plugins, foi modificado para que os caminhos percorridos pelo robô passem sempre pela árvore de Steiner (ou outra árvore fornecida pelo sistema). A navegação local não foi modificada, sendo utilizado o algoritmo padrão *dwa* (do inglês, *Dynamic Window Approach*) para desvio de obstáculos. Para localização, foi utilizado o algoritmo AMCL (do inglês, *Adaptive Monte-Carlo Localization*).

### 5.1 Experimentos Simulados

Para simular a abordagem proposta, foi utilizado um computador desktop com as seguintes configurações: Processador Amd Fx-6300 Bulldozer, 4 GB RAM, Sistema Operacional Ubuntu 12.04 64bits.

As simulações foram realizadas utilizando o simulador Stage (Gerkey et al. [2003]). Como o Stage não possui um simulador de redes interno, simulando os fenômenos de rede sem fio, possibilitando uma resposta realista da conectividade entre os robôs, nos experimentos simulados, para determinar a conectividade entre os robôs, foi utilizando o modelo de disco unitário. Se dois robôs que estão a uma distância menor que  $d$ , esses são considerados conectados.

A Figura 5.1 mostra a execução de uma simulação. Inicialmente uma árvore de Steiner Euclidiana é criada interconectando os clientes. Em seguida, cada robô executa a CEFSM, sendo a solução final formada por 6 robôs no estado *conectado*. A simulação

---

<sup>1</sup>[www.ros.org](http://www.ros.org)

mostra também a robustez da solução, na sequência 17 um robô falha e os robôs no estado *espera* são utilizados para reparar a solução.

A Figura 5.2 mostra a configuração final de uma simulação utilizando 5 clientes (pontos pretos) em um ambiente com obstáculos. A linha preta representa a árvore de Steiner Euclidiana (EMSTO). Os quadrados em vermelho são os robôs em estado conectado, os triângulos são os robôs em estado de espera. Executando a abordagem proposta, para construir a rede foram necessários 26 robôs. Os dois robôs em estado de espera são utilizados para robustez da rede. Caso algum robô em estado conectado falhar, os robôs em estado de espera executarão o CEFSM. Esse experimento mostra que a abordagem funciona em ambientes grandes com muitos robôs.

Com o objetivo de criar uma simulação realista, um ambiente real medindo 50x30 metros foi mapeado utilizando um laser. A Figura 5.1 apresenta o mapa e o ambiente dos experimentos. A área cinza indica os locais bloqueados e a área branca representa os corredores presentes no ambiente.

A Figura 5.4 apresenta uma simulação em um ambiente realista utilizando 4 clientes. Nessa instância, foram necessários 5 robôs para conectar os clientes. Os robôs restantes ficaram em estado de espera.

### 5.1.1 Comparação das árvores de Steiner

Nessa seção são comparadas as soluções de duas abordagens para a criação da árvore conectando os clientes: árvore de Steiner Euclidiana e árvore de Steiner em Grafo. O algoritmo para árvore de Steiner em grafo utilizado é a solução 2-aproximado com o grafo de visibilidade conectando os clientes. A Figura 5.5 mostra as configuração dos experimentos por etapa.

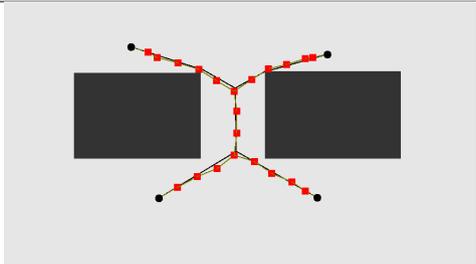
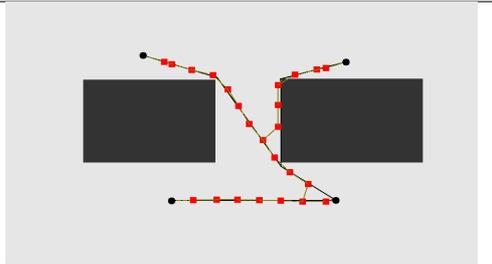
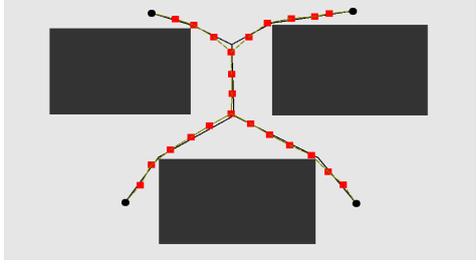
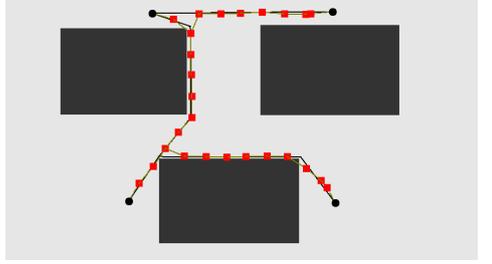
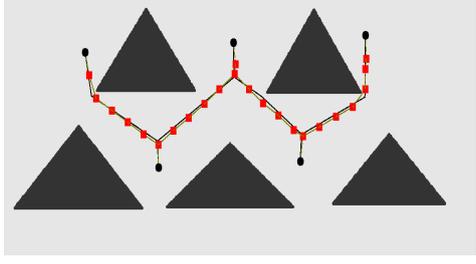
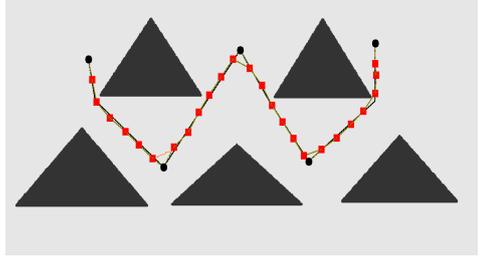
A Tabela 5.1 apresenta 3 experimentos comparando os métodos de criação de árvore. Os experimentos 1 e 2 conectam 4 clientes e o último experimento conecta 5 clientes. Em todos os experimentos, as distâncias de comunicação utilizadas foram 155, 77, 38 unidades de medida. O número máximo de robôs foi calculado utilizando a formula apresentada no Capítulo 4. A topologia final dos experimentos com raio de comunicação 38 unidades de medida é apresentado nas Tabela 5.2.

Os experimentos mostram que o limite máximo teórico é conservativo considerando o número efetivo de robôs utilizados para conectar os clientes. Além disso é possível verificar que a diferença entre o limite máximo e número efetivo de robôs utilizado se torna menor ao diminuir o raio de comunicação.

No terceiro experimento, com 77 unidades de medida como raio de comunicação,

Exp.	Raio	Árvore de Steiner Euclidiana		Árvore de Steiner em Grafo		
		Tamanho da árvore	#Robôs		Tamanho da árvore	#Robôs Utilizados
			Limite Superior	Utilizados		
1	155	777.28	13	3	957.869	3
	77		18	8		10
	38		28	21		24
2	155	903.161	13	4	1029.34	4
	77		19	10		10
	38		31	23		25
3	155	906.375	15	4	943.161	4
	77		21	11		10
	38		33	23		25

**Tabela 5.1.** Comparação da árvore de Steiner Euclidiana e da árvore de Steiner em grafo.

Exp.	Árvore de Steiner Euclidiana	Árvore de Steiner em Grafo
1		
2		
3		

**Tabela 5.2.** Topologia final dos experimentos de raio 38 u.m. da Tabela 5.1

o número de robôs utilizado pela árvore de Steiner em Grafo foi menor que o utilizado pela árvore de Steiner Euclidiana. A árvore de Steiner prove a menor conexão para o conjunto de clientes, entretanto a alocação da CEFMS não é ótima sempre, além do

algoritmo ser sensível ao formato da árvore.

## 5.2 Experimentos Reais

Para fazer os experimentos foram utilizadas as plataformas iRobot Create equipadas com o laser Hokuyo URG-04LX-UG01 e um giroscópio de dois eixos ADXRS613 para auxiliar na localização e navegação. A Figura 5.6 mostra a plataforma utilizada nos experimentos. Os clientes utilizados foram netbooks rodando o algoritmo de roteamento OLSR.

Para escolher a métrica para avaliar a conexão de um salto, foram avaliados as medidas *Received Signal Strength* (RSS) e OLSR Link Quality juntamente com a vazão da rede, variando as distâncias. Pelas medições o RSS apresenta 0.87 de correlação enquanto o OLSR Link Quality prove 0.76. Assim, o RSS foi escolhido como métrica de conectividade. A Figura 5.7 mostra a média, valor máximo e mínimo do RSS medido em relação a distância. Esses valores foram utilizados para determinar os limiares de conexão e desconexão dos experimentos reais.

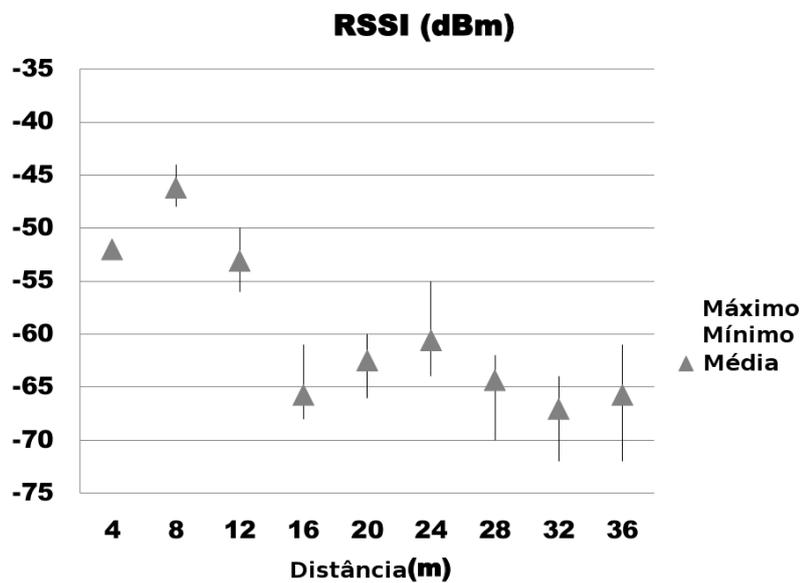


Figura 5.7. Média, valor máximo e mínimo do RSS em relação a distância.

A Figura 5.8 apresenta um experimento utilizando 3 clientes. Nesse experimento o limiar de desconexão foi -65 dBm e conexão -60 dBm. A topologia final necessitou de 3 robôs para conectar a rede. Foram avaliados também a vazão da rede para conexões TCP. Essa medida avalia quantos bytes podem ser enviados de um cliente a outro. A

vazão obtida nesse experimento é apresentada na Figura 5.9. Isso mostra que a rede foi efetivamente conectada.

Já a Figura 5.10 apresenta um experimento real com 2 clientes. Nesse experimento o limiar de desconexão foi -60 dBm e conexão -55 dBm. O experimento necessitou de 3 robôs roteadores para criar a solução. A Figura 5.11 mostra a vazão da rede para esse experimento. Como pode ser visto nos experimentos simulados, alguns robôs, ao utilizar métrica de distância, acabam equidistantes uns dos outros pela forma que o CEFSM faz a alocação da rede, entretanto isso não ocorre nos experimentos reais devido a utilização do RSS, que não é monotonicamente decrescente em relação a distância dos robôs.

### 5.3 Comparação Experimentos Reais e Simulados

O simulador Stage e o framework ROS( Quigley et al. [2009]) não são projetados para simular redes sem fio de forma realistas, não sendo capazes de considerar os fenômenos de propagação de sinal. Assim, enquanto nos experimentos simulados é utilizado como métrica distância, nos experimentos reais é utilizado a métrica RSS.

A Figura 5.12 apresenta a topologia final da para um experimento simulado com 2 clientes. A simulação inicia com três robôs, entretanto apenas 2 são necessários para conectar os clientes. O terceiro robô, estando no estado de espera, não faz parte da rede de comunicação.

A Figura 5.13 apresenta a configuração final do experimento real. O experimento utiliza o limiar de desconexão de -65 dBm e de conexão -60 dBm. Nesse experimento foi necessário dois robôs para estabelecer a rede. Comparado com a simulação, os robôs terminaram em posições diferentes. Isso ocorre pois a medição do RSS não depende apenas da distância entre os dispositivos, sendo a medida influenciada pelo decaimento de sinal, sombreamento, multi-caminho. Entretanto, o experimento mostra que a abordagem proposta funciona com medidas reais de qualidade de sinal.

A Figura 5.14 apresenta a vazão da rede para o par de clientes, mostrando que a rede foi efetivamente estabelecida.

Portanto, através dos experimentos reais é possível concluir que a abordagem proposta é capaz de lidar com fenômenos de propagação de sinal, criando uma rede ad-hoc interconectando um conjunto de clientes em um ambiente real.

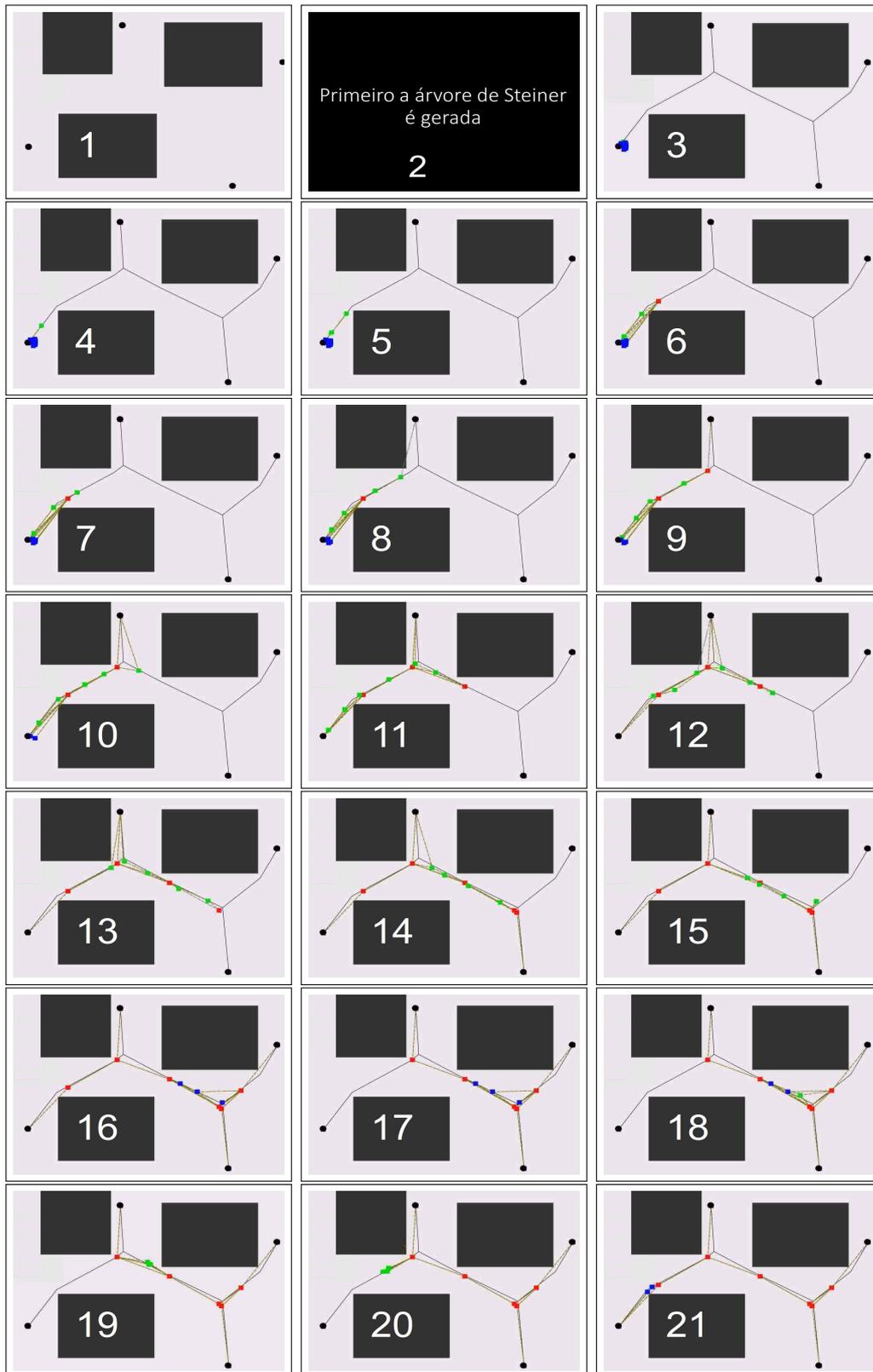
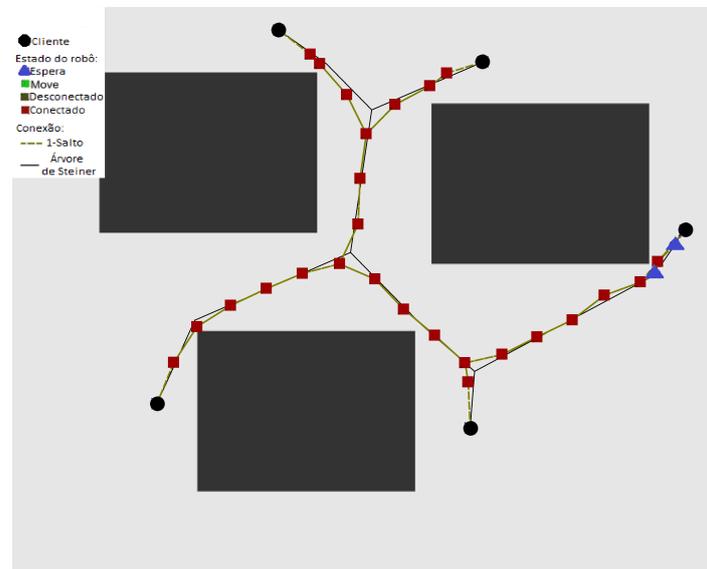


Figura 5.1. Sequência de imagens de uma simulação.



**Figura 5.2.** Simulação com um ambiente contendo obstáculos e 5 clientes (pontos). A linha preta indica a árvore de Steiner. A solução final necessitou de 26 robôs (quadrados).



**Figura 5.3.** Ambiente e mapa dos experimentos.

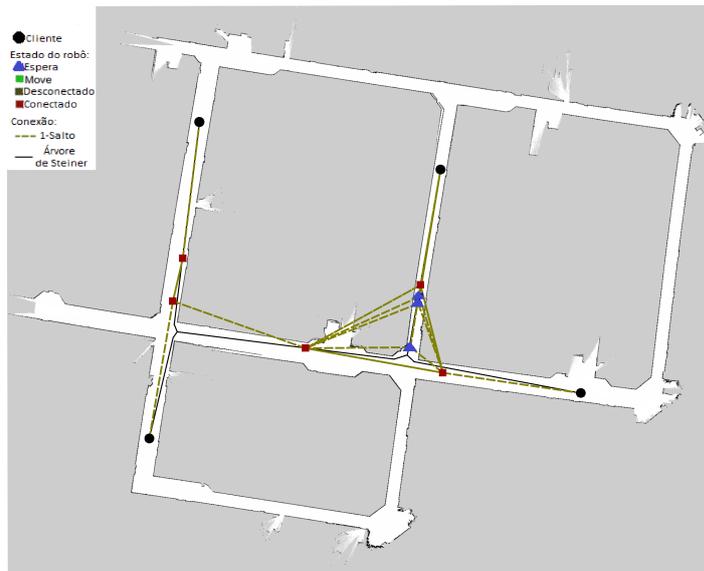


Figura 5.4. Topologia final de experimento simulado com 4 clientes.

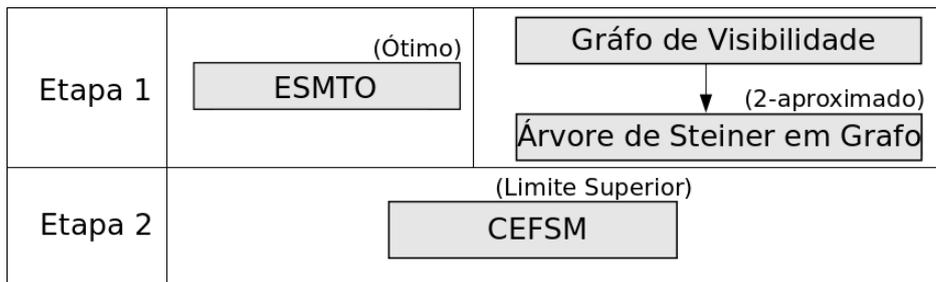


Figura 5.5. Etapas da abordagem na comparação das árvores de Steiner.



Figura 5.6. Robô utilizados em experimentos.

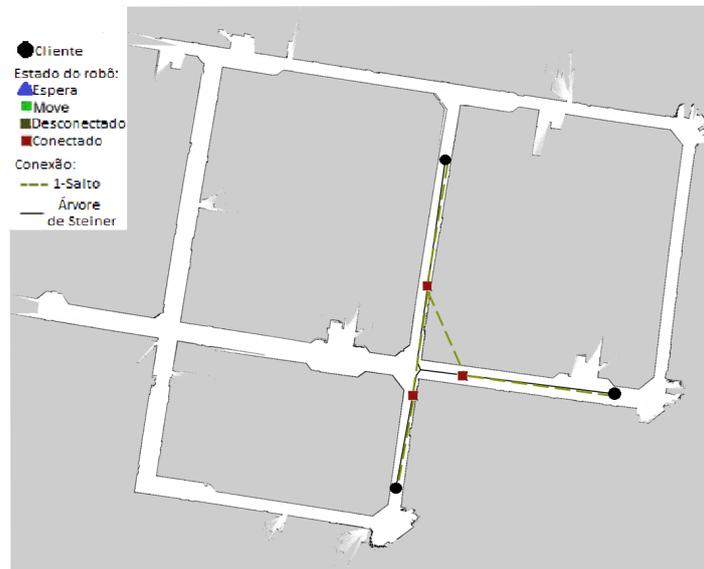


Figura 5.8. Topologia de experimento real com 3 clientes.

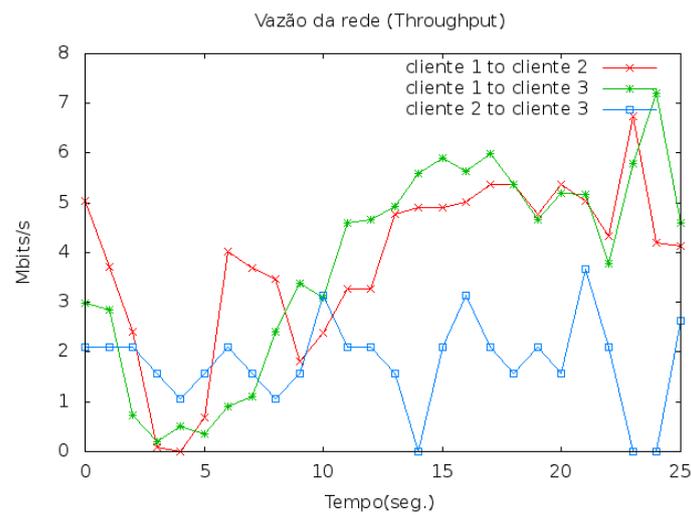


Figura 5.9. Vazão da rede em experimento real com 3 clientes.

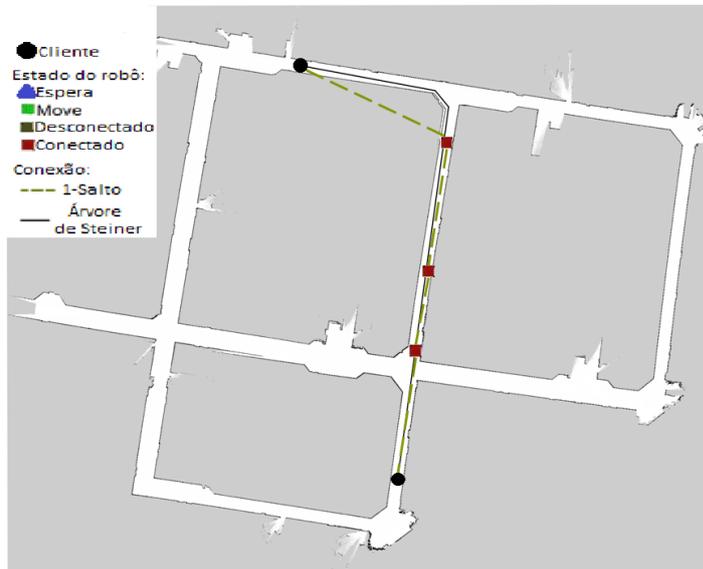


Figura 5.10. Topologia de experimento real com 2 clientes.

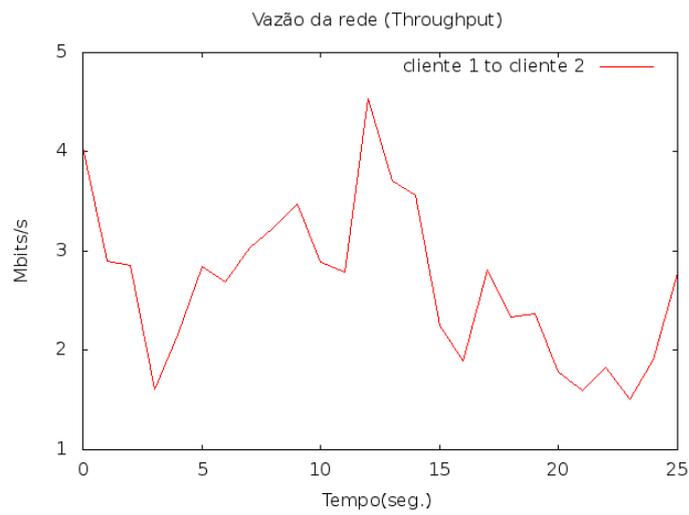


Figura 5.11. Vazão da rede em experimento real com 2 clientes.



Figura 5.12. Topologia final e layout de experimento simulado

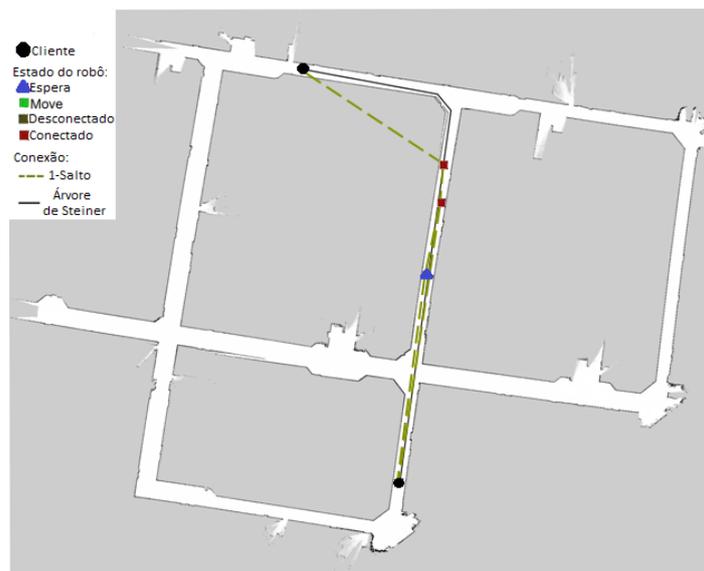


Figura 5.13. Topologia final e layout de experimento real.

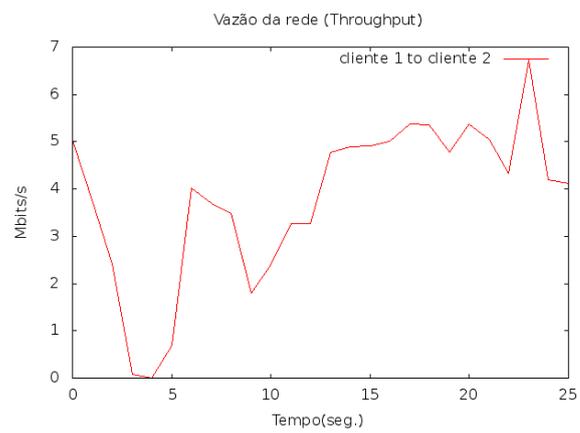


Figura 5.14. Vazão da rede em experimento real com dois clientes.

# Capítulo 6

## Conclusão e Trabalhos Futuros

### 6.1 Conclusão

Neste trabalho foi estudado o problema de alocação de robôs roteadores para criação de uma rede ad-hoc interconectando um conjunto de clientes. A solução proposta utiliza uma máquina de estados que conecta os robôs roteadores de forma autônoma. Com o objetivo de reduzir a quantidade de robôs utilizados para criar a solução, foi proposta a utilização de uma árvore de Steiner Euclidiana para determinar a posição onde os robôs podem ser alocados.

A abordagem proposta é autônoma, descentralizada, e considera fenômenos reais de propagação de sinal. É autônoma dado que os robôs são capazes de criar a rede através da execução de uma máquina de estados. É descentralizado, dado que cada robô executa uma máquina de estados individual. Considera fenômenos reais de propagação de sinal dado que a qualidade de sinal RSS é medida para determinar os enlaces locais.

A partir da análise teórica da abordagem foi possível apresentar um limite superior para a abordagem apresentada. O limite obtido foi verificado através de experimentos simulados, sendo o número de robôs necessário para criação da rede menor que o valor máximo predito. Na comparação das árvores de Steiner Euclidiana e árvore de Steiner em grafo, foi apresentado que a CEFMS pode não apresentar a melhor solução para o problema.

Para avaliar o algoritmo, foram utilizados experimentos reais e simulados. Os experimentos simulados mostram a abordagem criando redes com muitos robôs, mostrando que o algoritmo apresenta escalabilidade. Nos experimentos reais, que possibilitou a utilização da métrica RSS, o objetivo foi mostrar que a abordagem é capaz de criar a rede em um ambiente real. O método proposto foi construído utilizando uma abordagem de histerese, sendo capaz de lidar com variação do sinal. Assim, a partir de

métricas reais de conectividade, o algoritmo proposto foi validado, demonstrando ser efetivo na criação de uma solução mesmo em cenários reais.

## 6.2 Trabalhos Futuros

Na solução apresentada neste trabalho, após alocado em uma posição, um robô não é realocado em outra posição no futuro. Em um trabalho futuro essa restrição pode ser relaxada através da alteração da máquina de estados, permitindo que um robô que já foi alocado seja realocado caso necessário. Por exemplo, isso poderia ser utilizado para reduzir a concorrência entre os robôs, permitindo que os robôs determinem o local onde a alocação deverá ser feita, sendo a alocação feita através de, por exemplo, leilões de posicionamentos.

Neste trabalho também foi considerado que o ambiente estudado é conhecido. Em um trabalho futuro, um ambiente parcialmente conhecido pode ser abordado, permitindo que o mapa do ambiente esteja desatualizado, algo que pode estar associado a um cenário de real desastres. Para criar uma solução para esse problema, o mapeamento do ambiente terá que ser abordado de forma simultânea a alocação dos robôs. Além disso, a máquina de estados terá que permitir que os robôs sejam realocados e a construção da árvore de Steiner terá que ser eficiente para determinar em tempo real a árvore interconectando os clientes.

O problema onde os clientes são móveis também pode ser abordado em um trabalho futuro. Para isso, uma árvore de Steiner Euclidiana pode ser utilizada para determinar o posicionamento dos robôs roteadores. Além disso, dado o número de clientes e o ambiente que será utilizado, poderá ser proposto uma abordagem que apresente um limite máximo de robôs que serão necessários para manter a rede sempre conectada.

# Referências Bibliográficas

- Abbas, A. & Younis, M. (2013). Establishing connectivity among disjoint terminals using a mix of stationary and mobile relays. *Computer Communications*, 36(13):1411-1421.
- Bayati, M.; Borgs, C.; Braunstein, A.; Chayes, J.; Ramezanpour, A. & Zecchina, R. (2008). Statistical mechanics of steiner trees. *Physical Review Letters*, 101(3):37208.
- Camp, T.; Boleng, J. & Davies, V. (2002). A survey of mobility models for ad hoc network research. *Wireless communications and mobile computing (wcmc): special issue on mobile ad hoc networking: research, trends and applications*, 2:483--502.
- Cheng, X.; Du, D.-Z.; Wang, L. & Xu, B. (2008). Relay sensor placement in wireless sensor networks. *Wireless networks*, 14(3):347--355.
- Chiu, H. C. H.; Ryu, B.; Zhu, H.; Szekely, P.; Maheswaran, R.; Rogers, C.; Galstyan, A.; Salemi, B.; Rubenstein, M. & Shen, W.-M. (2009). Tentacles: self-configuring robotic radio networks in unknown environments. Em *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1383--1388, Piscataway, NJ, USA. IEEE Press.
- Chiu, H. C. H. & Shen, W.-M. (2011). ANCHOR - Self-Configuring Robotic Network. Em *IEEE International Conference on Robotics and Automation*.
- Clausen, P. J. T. (2003). Optimized link state routing protocol (olsr), ietf rfc 3626.
- Cormen, T. H.; Leiserson, C. E.; Rivest, R. L. & Stein, C. (2009). *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edição.
- Correll, N.; Bachrach, J.; Vickery, D. & Rus, D. (2009). Ad-hoc wireless network coverage with networked robots that cannot localize. Em *IEEE International Conference on Robotics and Automation*.

- Cortes, J.; Martinez, S.; Karatas, T. & Bullo, F. (2004). Coverage control for mobile sensing networks. *20(2):243–255*.
- Derbakova, A.; Correll, N. & Rus, D. (2011). Decentralized self-repair to maintain connectivity and coverage in networked multi-robot systems. Em *IEEE International Conference on Robotics and Automation*, pp. 3863–3868. IEEE.
- Du, D. & Hu, X. (2008). *Steiner Tree Problems In Computer Communication Networks*. World Scientific Publishing Co., Inc., River Edge, NJ, USA.
- Du, D.-Z. & Hwang, F. K. (1992). A proof of the gilbert-pollak conjecture on the steiner ratio. *Algorithmica*, 7(2-3):121–135.
- Eiselt, H. A. & Marianov, V. (2012). Mobile phone tower location for survival after natural disasters. *European Journal of Operational Research*, 216(3):563–572.
- Esposito, J. M. & Dunbar, T. W. (2006). Maintaining Wireless Connectivity Constraints for Swarms in the Presence of Obstacles. Em *IEEE International Conference on Robotics and Automation*, pp. 946–951.
- Feldman, D.; Gil, S.; Knepper, R. a.; Julian, B. & Rus, D. (2013). K-robots clustering of moving sensors using coresets. *2013 IEEE International Conference on Robotics and Automation*, pp. 881–888.
- Fiedler, M. (1973). Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(98):298–305.
- Fink, J.; Ribeiro, A. & Kumar, V. (2012). Robust control for mobility and wireless communication in cyber-physical systems with application to robot teams. *Proceedings of the IEEE*, 100(1):164–178.
- Gage, D. W. (1992). *Command control for many-robot systems*. Naval command control and ocean surveillance center rdt and e div.
- Gerkey, B.; Vaughan, R. & Howard, A. (2003). The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. Em *11th International Conference on Advanced Robotics, 2003*.
- Gil, S.; Feldman, D. & Rus, D. (2012). Communication coverage for independently moving robots. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1:4865–4872.

- Gilbert, E. & Pollak, H. (1968). Steiner minimal trees. *SIAM Journal on Applied Mathematics*, 16(1):1--29.
- Giordano, P. R.; Franchi, A.; Secchi, C. & Bühlhoff, H. H. (2013). A Passivity-Based Decentralized Strategy for Generalized Connectivity Maintenance. 3.
- Guan, K.; Imbrenda, D.; Ghanadan, R. & Hsu, J. (2008). Distributed sensing and communications in tactical robotic networks. Em *IEEE International Conference for Military Communications*, pp. 1–7.
- Hollinger, G. & Singh, S. (2010). Multi-robot coordination with periodic connectivity. Em *IEEE International Conference on Robotics and Automation, ICRA 2010, Anchorage, Alaska, USA, 3-7 May 2010*, pp. 4457–4462. IEEE.
- Howard, A.; Matarić, M. J. & Sukhatme, G. S. (2002). Mobile Sensor Network Deployment using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem. Em *Distributed Autonomous Robotic Systems*, pp. 299--308.
- Hsieh, M. A.; Cowley, A.; Kumar, V. & Taylor, C. J. (2006). Towards the deployment of a mobile robot network with end-to-end performance guarantees. Em *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pp. 2085–2091.
- Hwang, F. K.; Richards, D. S. & Winter, P. (1992). *The Steiner tree problem*. Annals of discrete mathematics. North-Holland, Amsterdam, New York.
- Karp, R. (1972). Reducibility among combinatorial problems. Em Miller, R.; Thatcher, J. & Bohlinger, J., editores, *Complexity of Computer Computations*, The IBM Research Symposia Series, pp. 85–103. Springer US.
- Kudelski, M.; Gambardella, L. M. & Di Caro, G. a. (2014). A mobility-controlled link quality learning protocol for multi-robot coordination tasks. *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5024--5031.
- Lin, G.-H. & Xue, G. (1999). Steiner tree problem with minimum number of steiner points and bounded edge-length. *Information Processing Letters*, 69(2):53--57.
- Melzak, Z. A. (1961). On the problem of Steiner. 4:143--148.
- Pimenta, L. C. A.; Kumar, V.; Mesquita, R. C. & Pereira, G. A. S. (2008). Sensing and coverage for a network of heterogeneous robots. Em *IEEE Conference on Decision and Control, 2008*, pp. 3947–3952. IEEE.

- Poduri, S. & Sukhatme, G. S. (2004). Constrained Coverage for Mobile Sensor Networks. Em *In IEEE International Conference on Robotics and Automation*, pp. 165--172.
- Portmann, M. & Pirzada, A. (2008). Wireless mesh networks for public safety and crisis management applications. *IEEE Internet Computing*, 12(1):18–25.
- Quigley, M.; Conley, K.; Gerkey, B. P.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R. & Ng, A. Y. (2009). Ros: an open-source robot operating system. Em *ICRA Workshop on Open Source Software*.
- Rizzo, C.; Tardioli, D.; Sicignano, D.; Riazuelo, L.; Villarroel, J. L. & Montano, L. (2013). Signal-based deployment planning for robot teams in tunnel-like fading environments. *The International Journal of Robotics Research*, 32(12):1381--1397.
- Robins, G. & Zelikovsky, A. (2005). Tighter bounds for graph steiner tree approximation. *SIAM Journal on Discrete Mathematics*, 19(1):122--134.
- S. Das, C. E. P. & Belding-Royer, E. M. (2003). Ad hoc on-demand distance vector (aodv) routing, ietf rfc 3561.
- Sabattini, L.; Secchi, C. & Chopra, N. (2012). Decentralized connectivity maintenance for networked lagrangian dynamical systems. Em *IEEE International Conference on Robotics and Automation (ICRA), 2012*, pp. 2433 –2438.
- Spanos, D. & Murray, R. (2005). Motion planning with wireless network constraints. Em *Proceedings of the American Control Conference, 2005.*, pp. 87 – 92.
- Stephan, J.; Fink, J.; Charrow, B.; Ribeiro, A. & Kumar, V. (2014). Robust routing and Multi-Confirmation Transmission Protocol for connectivity management of mobile robotic teams. *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, (Iros)*:3753--3760.
- Stump, E.; Jadbabaie, A. & Kumar, V. (2008). Connectivity Management in Robot Networks . Em *IEEE International Conference on Robotics and Automation (ICRA)*, "Pasedena.
- Tekdas, O.; Kumar, Y.; Isler, V. & Janardan, R. (2009). Algorithmic aspects of wireless sensor networks. capítulo Building a Communication Bridge with Mobile Hubs, pp. 179--190.

- Vieira, M. A. M.; Govindan, R. & Sukhatme, G. S. (2011). Towards Autonomous Wireless Backbone Deployment in Highly-Obstructed Environments. Em *IEEE International Conference on Robotics and Automation*.
- Warne, D. (1998). Spanning Trees in Hypergraphs with Applications to Steiner Trees.
- Warne, D. M.; Winter, P. & Zachariasen, M. (2000). Exact Algorithms for Plane Steiner Tree Problems: A Computational Study. Em Du, D.-Z.; Smith, J. M. & Rubinstein, J. H., editores, *Advances in Steiner Trees*, pp. 81--116. Kluwer Academic Publishers, Boston.
- Williams, R. K.; Gasparri, A. & Krishnamachari, B. (2014). Route swarm: Wireless network optimization through mobility. *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, (Iros):3775--3781.
- Winter, P. (1985). An algorithm for the steiner problem in the euclidean plane. *Networks*, 15(3):323--345.
- Y.-C. Hu, D. B. J. & Maltz., D. A. (2004). The dynamic source routing protocol (dsrc) for mobile ad hoc networks for ipv4, ietf rfc 4728.
- Yao, Z. & Gupta, K. (2009). Backbone-based connectivity control for mobile networks. Em *IEEE International Conference on Robotics and Automation, 2009.*, pp. 1133--1139.
- Zachariasen, M. & Winter, P. (1999). Obstacle-avoiding euclidean steiner trees in the plane: An exact algorithm. Em Goodrich, M. & McGeoch, C., editores, *Algorithm Engineering and Experimentation*, volume 1619 of *Lecture Notes in Computer Science*, pp. 286--299. Springer Berlin Heidelberg.
- Zeiger, F.; Kraemer, N. & Schilling, K. (2008). Commanding mobile robots via wireless ad-hoc networks; a comparison of four ad-hoc routing protocol implementations. Em *IEEE International Conference on Robotics and Automation, 2008.*, pp. 590--595.

