

LUIS MAURICIO TREVISAN

**UM ALGORITMO DE LOCALIZAÇÃO DE
NÓS EM REDES SEM-FIO USANDO NÍVEL
DE POTÊNCIA DO SINAL**

Dissertação submetida ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito parcial para a obtenção do título de Mestre em Informática.

Curitiba PR
Fevereiro de 2009

LUIS MAURICIO TREVISAN

**UM ALGORITMO DE LOCALIZAÇÃO DE
NÓS EM REDES SEM-FIO USANDO NÍVEL
DE POTÊNCIA DO SINAL**

Dissertação submetida ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito parcial para a obtenção do título de Mestre em Informática.

Área de concentração: *Ciência da Computação*

Orientador: Prof. Dr. Marcelo Eduardo Pellenz

Curitiba PR
Fevereiro de 2009

Trevisan, Luis Mauricio.

Um Algoritmo de Localização de Nós em Redes Sem-fio Usando Nível de Potência do Sinal. Curitiba, 2008. 76p.

Dissertação - Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática.

1. Redes Sem-Fio 2. Algoritmos de Localização 3. Métodos de Localização. I. Pellenz, Marcelo Eduardo. II. Pontifícia Universidade Católica do Paraná. Centro de Ciências Exatas e de Tecnologia. Programa de Pós-Graduação em Informática II-Título.

*Esta folha deve ser substituída pela ata de defesa devidamente assinada,
que será fornecida pela secretaria do programa após a defesa.*

A minha amada esposa, Cintia. A meus amados pais, Luiz e Maris.

Resumo

As redes sem-fio são cada vez mais populares devido ao seu baixo custo, tolerância a falhas, facilidade de instalação e operação. Há três tipos principais: Redes *Ad-Hoc*, Redes em Malha e Redes de Sensores. Em todas elas a capacidade de se determinar a localização geográfica de cada nó é condição necessária, quando não essencial, para um bom desempenho. Exemplo disso são os protocolos de roteamento baseados em localização desenvolvidos para redes *Ad-Hoc* que melhoram a robustez e otimizam o consumo de energia. Uma rede de sensores sem-fio só pode operar se a posição dos sensores for conhecida, caso contrário os dados coletados perdem significado. Sistemas tradicionais de localização, como o GPS, são inadequados para uso em redes sem-fio principalmente devido ao custo e consumo extra de energia que acarretam. Este trabalho explora um algoritmo de localização de nós em redes sem-fio que utiliza apenas a potência média do sinal recebido entre pares de nós como medida de distância relativa. Esta informação é comumente disponibilizada nas interfaces de rádio e não há consumo adicional de energia devido ao seu uso. O algoritmo de localização proposto se baseia no método numérico de otimização não-linear de Nelder-Mead e foi implementado tanto na forma iterativa quanto conjunta para fins de estudo e comparação de desempenho. Diferentes funções custo foram elaboradas e uma métrica para ponderação foi implementada. O desempenho é avaliado através de simulações em diferentes cenários de rede. Medidas de potência de sinal recebidas numa rede *Ad-Hoc* IEEE 802.11 são utilizadas para demonstrar a eficácia do algoritmo numa aplicação real de localização. Iniciando a busca em um ponto qualquer aleatoriamente distribuído na rede, o algoritmo proposto apresenta erro médio de localização equivalente ao do algoritmo baseado em Gradiente Conjugado e que utiliza o Estimador de Máxima Verossimilhança-MLE.

Palavras-chave: Redes Sem-Fio, Algoritmos de Localização, Métodos de Localização.

Abstract

Wireless Networks are becoming even more popular thanks to its low cost, resilience, easy of deployment and operation. There are three main types: Ad-Hoc, Wireless Mesh Network-WMN and Wireless Sensor Network-WSN. In all of them, the capacity to determine the geographic location of each node is a needed condition, even a must condition, for a good performance of the whole network. For example, Location Based Routing-LBR protocols are used in Ad-Hoc networks to improve robustness and to optimize energy consumption. A WSN may only operate if nodes position are known, otherwise the collected data has no meaning. Traditional Locating Systems, like GPS, are inadequate to Wireless Networks due to its high cost and energy consumption. This work explores a locating algorithm that uses only the Received Signal Strength Indication-RSSI between nodes pairs to estimate relative distance. This data is already present in the radio interfaces and its use causes no additional energy consumption. The proposed algorithm is based on the Numerical Non-Linear Optimization Method from Nelder-Mead and was implemented in a joint and iterative logic for study and comparison reasons. Different cost functions are stressed and a weighting cost metric is implemented. The performance is evaluated via simulations in different network sceneries. Measurements of Received Signal Strength gathered from an *Ad-Hoc* IEEE802.11 network are used to demonstrate the algorithm effectiveness in a real network location application. Initiating the optimization in any point randomly distributed within the network, the proposed algorithm presents a mean location error close to the Conjugate Gradient based algorithm that uses the Maximum Likelihood Estimator-MLE.

Keywords: Wireless Networks, Locating Algorithms, Locating Methods.

Sumário

Resumo	ix
Abstract	xi
Lista de Figuras	xvi
Lista de Abreviações	xviii
1 Introdução	1
1.1 Motivação	2
1.2 Proposta	3
1.3 Organização	4
2 Sistemas de Localização	5
2.1 Técnicas de Estimativa de Distância	5
2.1.1 Técnica Baseada em Tempo de Chegada	6
2.1.2 Técnica Baseada em Ângulo de Chegada	7
2.1.3 Técnica Baseada na Potência do Sinal	8
2.1.4 Técnica Híbrida	8
2.2 Modelagem do Canal de Rádio	8
2.2.1 Desvanecimento em Larga Escala	8
2.2.2 Desvanecimento em Pequena Escala	10
2.2.3 Métodos de Localização	10
2.2.4 Trilateração	11
2.2.5 Triangulação	11
2.2.6 Multilateração	11
2.3 Sistemas Tradicionais de Localização	12
2.3.1 <i>Sistema de Posicionamento Global</i>	12
2.3.2 <i>Sistema de Posicionamento Móvel Celular</i>	13
2.4 Localização em Redes Sem-fio	13
2.4.1 Classificação dos Algoritmos de Localização	13
2.4.2 Quanto ao tipo do Algoritmo	15
2.4.3 Trabalhos Relacionados	18
2.5 Conclusão	19
3 Elaboração de Algoritmos de Estudo	21
3.1 Conclusão	26

4	Avaliação de Desempenho	27
4.1	Localização Conjunta	28
4.1.1	Estudo de Cenários com Um Nó Desconhecido	29
4.1.2	Estudo de Cenários com Diversos Nós Desconhecidos	31
4.1.3	Avaliação do Desempenho do Algoritmo de Localização Conjunta	33
4.2	Localização Iterativa	37
4.2.1	Influência da Sequência de localização	42
4.2.2	Utilização de Coeficientes de Ponderação	47
4.3	Conclusão	49
5	Algoritmo Proposto	51
5.1	Implementação Prática do Algoritmo	60
5.1.1	Estimativa do Expoente de Perda de Percurso	60
5.1.2	Aplicação Real do Algoritmo de Localização	61
5.2	Conclusão	69
6	Conclusões	71
A	O Método do Poliedro Flexível na Localização	75

Lista de Figuras

2.1	Técnica Baseada em Tempo de Chegada: (a) medição do ToA ; (b) medição do $TDoA$	7
2.2	Trilateração: (a) caso ideal; (b) caso real com erros	11
2.3	Triangulação	12
2.4	Multilateração	12
3.1	Nomenclatura adotada para as diferentes distâncias no processo de multilateração	23
4.1	Topologia de Rede Utilizada para Simulação em Matlab	28
4.2	Erro Médio de Localização de Um Nó em Redes Diferentes em Função de K_a : $m = 4, n = 1$	29
4.3	PDF e CDF do Erro Médio de Localização de Um Nó Fixo no Cenário de Referência	30
4.4	Comparação de Erro Médio de Localização Conjunta no Cenário de Referência em Função do Número de Nós para diferentes algoritmos	31
4.5	Erro Médio de Localização Conjunta para <i>Gradiente Conjugado-MLE</i> na rede de referência	32
4.6	Erro Médio de Localização Conjunta para <i>NM-Mínimos Quadrados</i> e <i>NM-MLE</i> na rede de referência	33
4.7	Erro Médio de Localização para <i>NM-Multilateração</i> na rede de referência . . .	34
4.8	Número Médio de Iterações para as Variantes do Algoritmo em Estudo na Lo- calização Conjunta na Rede de Referência	34
4.9	Número Médio de Iterações para Algoritmos de Localização Conjunta com custo em função do <i>MLE</i> na Rede de Referência	35
4.10	Tempo Médio de Convergência para as Variantes do Algoritmo em Estudo na Localização Conjunta na Rede de Referência	36
4.11	Tempo Médio de Convergência para Algoritmos de Localização Conjunta com custo em função do <i>MLE</i> na Rede de Referência	36
4.12	Erro Médio de Localização Iterativa no Cenário de Referência para <i>NM- Mínimos Quadrados</i> e <i>NM-Multilateração</i>	38
4.13	Número Médio de Iterações no Cenário de Referência para <i>NM-Mínimos Quadrados</i> e <i>NM-Multilateração</i> em Localização Iterativa	39
4.14	Tempo Médio de Convergência no Cenário de Referência para <i>NM-Mínimos Quadrados</i> e <i>NM-Multilateração</i> em Localização Iterativa	40
4.15	Erro Médio de Localização na Rede de Referência com $n = 8$ em Função de K_a para <i>NM-Mínimos Quadrados</i> e <i>NM-Multilateração</i> em Localização Iterativa .	40
4.16	Erro Médio de Localização no Cenário de Referência para a Variante <i>NM- Mínimos Quadrados</i> em Algoritmo do tipo Conjunto e Iterativo	41

4.17	Número Médio de Iterações no Cenário de Referência para a Variante <i>NM-Mínimos Quadrados</i> em Algoritmo do tipo Conjunto e Iterativo	41
4.18	Tempo Médio de Convergência no Cenário de Referência para a Variante <i>NM-Mínimos Quadrados</i> em Algoritmo do tipo Conjunto e Iterativo	42
4.19	Movimento Vertical e Diagonal a partir do centro da Rede de Referência	43
4.20	Erro Médio de Localização para o nó se afastando do centro da rede no Cenário de Referência	43
4.21	Rede de Referência $n = 15$ e $m = 4$	46
4.22	Comparação entre Localização Iterativa com e sem Sequenciamento dos nós no Cenário de Referência	46
4.23	Comparação dos Diferentes Tipos de Ponderação para Localização Iterativa no Cenário de Referência	47
4.24	Comparação do Erro Médio de Localização Iterativa das Ponderações β_1 e β_2 no Cenário de Referência	49
5.1	Método de seleção do ponto inicial de busca θ_{0_i}	53
5.2	Desempenho dos algoritmos proposto e <i>Gradiente Conjugado</i> para diferentes abordagens do ponto inicial de busca	54
5.3	Tempo médio de convergência para o algoritmo de <i>Gradiente Conjugado</i> para diferentes abordagens do ponto inicial de busca	55
5.4	Tempo médio de convergência para o algoritmo proposto para diferentes abordagens do ponto inicial de busca	55
5.5	Desempenho do algoritmo proposto em função das ponderações aplicadas	56
5.6	Desempenho dos algoritmos de Gradiente Conjugado e proposto para diferentes valores de K_a	57
5.7	Tempo médio de convergência em função de K_a para <i>Gradiente Conjugado</i>	57
5.8	Tempo médio de convergência em função de K_a para NM-LISPCP	58
5.9	Erro médio de localização em função da área da rede para os algoritmos NM-LIPCP e <i>Gradiente Conjugado</i>	58
5.10	Tempo médio de convergência para algoritmo de <i>Gradiente Conjugado</i> em função da área da rede	59
5.11	Tempo médio de convergência para o algoritmo proposto em função da área da rede	59
5.12	Variabilidade do Expoente de Perda de Percurso	61
5.13	Expoente Estimado de Perda de Percurso Usando os Valores Médios de SNR	61
5.14	Topologia de Rede Utilizada para as Medições	62
5.15	Comparação de desempenho entre <i>Gradiente Conjugado</i> e algoritmo proposto no cenário A	65
5.16	Comparação de desempenho entre <i>Gradiente Conjugado</i> e algoritmo proposto no cenário B	66
5.17	Comparação de desempenho entre <i>Gradiente Conjugado</i> e algoritmo proposto no cenário C	66
5.18	Comparação de desempenho entre <i>Gradiente Conjugado</i> e algoritmo proposto no cenário D	67
5.19	Comparação de desempenho entre <i>Gradiente Conjugado</i> e algoritmo proposto no cenário E	67

Lista de Tabelas

2.1	Tabela De Expoentes de Perda de Percurso	10
2.2	Classificação dos Trabalhos Relacionados	18
3.1	Simbologia Utilizada	22
4.1	Erro Médio de Localização de Um Nó Fixo no Cenário de Referência	30
4.2	Erro Médio de Localização de Um Nó Randomicamente Distribuído no Cenário de Referência	33
4.3	Fatores e Relações de Proximidade para o Nó da figura 4.19	45
5.1	Medidas Coletadas ao Ar Livre para Estimativa do Expoente de Perda de Percurso	60
5.2	Tabela De Potências Medidas	63
5.3	Tabela De Dados do Equipamento para uso no modelo de Espaço Livre	63
5.4	Tabela De Erros de Localização	64
5.5	Tabela De Estimativas De Distância	68

Lista de Abreviações

CDF	<i>Cumulative Density Function</i>
GPS	<i>Global Positioning System</i>
LBR	<i>Location Based Routing</i>
LOS	<i>Line Of Sight</i>
MLE	<i>Maximum Likelihood Estimator</i>
NM-LI	Localização Iterativa com otimização pelo Método do Poliedro Flexível
NM-LIS	NM-LI com Sequenciamento
NM-LISP	NM-LI com Sequenciamento Ponderado
NM-LISPCP	NM-LI com Sequenciamento e Custo Ponderado
PDA	<i>Personal Digital Assitant</i>
PDF	<i>Probability Density Function</i>
RF	<i>Radio Frequency</i>
RSSI	<i>Received Signal Strength Indication</i>
SNR	<i>Signal to Noise Ratio</i>
TDoA	<i>Time Difference Of Arrival</i>
ToA	<i>Time Of Arrival</i>
UWB	<i>Ultra Wide Band</i>
WSN	<i>Wireless Sensor Network</i>
WMN	<i>Wireless Mesh Network</i>

Capítulo 1

Introdução

As Redes Sem-fio tem cada vez mais espaço nas indústrias, escritórios, escolas e no cotidiano das pessoas. Computadores portáteis são equipados com interfaces IEEE802.11 de fábrica. Inúmeras novas aplicações continuam a emergir para as áreas: de entretenimento, de segurança, médica, industrial, de automação, entre outras. As principais Redes Sem-fio são: Redes *Ad-Hoc*, Redes de Sensores Sem-fio (*Wireless Sensor Network-WSN*) e as Redes em Malha Sem-fio (*Wireless Mesh Network-WMN*). Estas redes têm objetivos bem diferentes, mas o princípio de funcionamento é comum a todas elas: os pacotes são transportados em múltiplos saltos, sem a necessidade de um elemento mediador, como um ponto de acesso. São redes autônomas, flexíveis, tolerantes a falhas e de fácil instalação.

Com as Redes em Malha Sem-fio se estende a cobertura de serviços de banda larga na última milha [Alicherry et al., 2005], [Ramachandran et al., 2005],[Yang et al., 2005]. Seus nós não geram e nem terminam tráfego, mas cooperam para o transporte do tráfego entre os assinantes do serviço de banda larga de um lado e a rede da operadora prestadora do serviço de outro.

Redes *ad-hoc* têm por objetivo prover comunicação onde não haja qualquer outro tipo de infra-estrutura e são mais indicadas para a formação de redes transientes. Seu desenvolvimento foi dirigido por aplicações como: jogos iterativos, conferências, redes para campo de batalha e situações de emergência. Nestes tipos de aplicações os nós se conectam espontaneamente. Cada nó trata não apenas seu próprio tráfego, mas também retransmite o tráfego dos nós adjacentes. Celulares e computadores portáteis: *Lap Top*, *Palm Top*, *PDA*s são típicos terminais *ad-hoc*.

Enquanto as redes tradicionais com infra-estrutura servem como meio de comunicação entre um usuário e um serviço, as redes de sensores tem por objetivo fornecer o serviço: as informações coletadas. Os dados coletados isoladamente geralmente não fazem sentido, por isso o usuário da rede não interage com apenas um nó, mas com toda a rede. O conjunto de dados dos diferentes sensores precisa ser processado para ganhar significado. Redes de Sensores são muitas vezes formadas por centenas ou milhares de sensores lançados sobre vastas regiões que se pretende monitorar. Uma rede gigantesca pode ser formada por sensores lançados sobre uma floresta a fim de coletar informações de temperatura e umidade para estudos climáticos. Outras áreas de aplicação para Rede de Sensores são: monitoramento ambiental, militar, prevenção de desastres e automação: predial, comercial e residencial [Akyildiz et al., 2002].

Em todas as Redes Sem-fio a localização do nó é uma informação relevante. Em redes *Ad-Hoc* e Redes em Malha a informação de localização pode ser utilizada para suporte

ao funcionamento da própria rede. Por exemplo, o Roteamento Baseado em localização (*Location Based Routing-LBR*) utiliza a informação de localização para fazer o roteamento de pacotes, [Neto and Maia, 2008] e [Blazevic et al., 2001]. Esta técnica de roteamento é especialmente útil quando a origem e destino dos pacotes são distantes. Em Redes em Malha a informação de localização pode ser utilizada para se determinar os níveis de interferência à que os nós são submetidos através da adoção de um modelo de interferência como os propostos em [Gupta and Kumar, 2000]. A informação do nível de interferência pode ser utilizada para se criar métricas de roteamento que atribuam custos maiores às rotas sujeitas a maiores níveis de interferência e, desse modo, obter um melhor balanceamento de carga. Ou então para se elaborar mecanismos de escalonamento de transmissões que evitem ao máximo as colisões e interferências de transmissões simultâneas, [Gupta and Kumar, 2000]. É útil também para se melhorar a distribuição de canais às interfaces de rádio, promovendo o reuso de canais sem degradar a banda disponível. Outra aplicação é a localização de nós interferentes na rede, cujas transmissões comprometem a capacidade da rede mesmo sem terem acesso à mesma. Numa rede de sensores, o conhecimento da localização dos nós é primeiramente útil para gerenciamento da própria rede. A área de cobertura pode ser avaliada. Assim, numa rede formada por sensores aleatoriamente posicionados pode-se certificar de que as áreas de interesse estão suficientemente povoadas. A informação de posicionamento pode ser usada para a formação de *clusters* a fim de se usar roteamento hierárquico e processamento colaborativo das informações coletadas, [Krishnamachari, 2005]. Nas inúmeras aplicações de Redes de Sensores, uma ou mais grandezas físicas são avaliadas ou medidas: temperatura, umidade, pressão, luminosidade, nível de ruído sonoro e análise de compostos químicos entre outras. Qualquer que seja a grandeza monitorada, a medida feita por um sensor só tem significado se sua posição for conhecida.

Redes Sem-fio com poucos nós podem ser postas em operação de maneira que se conheça a localização de cada nó, sendo necessário manter uma relação entre a identificação do nó e sua posição. Redes em Malha podem ter proporções de grandes metrópoles e a tarefa de manter tal relação passa a ser inviável, principalmente se for levado em consideração a substituição de nós por defeito. Redes *Ad-Hoc* tem os nós móveis pela natureza de suas aplicações, [Yang et al., 2005], e tal controle não faria sentido. Muitas aplicações de Redes de Sensores utilizam nós fixos, mas como são distribuídos desordenadamente sobre vastas regiões não há como se ter controle de suas posições. Portanto, é necessário um mecanismo de obtenção da posição dos nós em uma rede Sem-fio sob demanda.

1.1 Motivação

O Sistema de Posicionamento Global (*Global Positioning System-GPS*) e o sistema de localização da Rede Móvel Celular (*Cell Based Positioning*) são sistemas tradicionais de localização e que provêm a localização de um receptor sob demanda, porém não podem ser utilizados em Redes Sem-fio. O Sistema de Posicionamento Global requer a utilização de receptores que consomem muita energia e têm dimensões grandes demais para a maioria das aplicações de Redes Sem-fio. Os nós de Redes *Ad-hoc* são computadores portáteis não equipados com receptores de *GPS*. Além disso o sinal de *GPS* pode não estar disponível em ambientes fechados, como em indústrias. Seu alto custo inviabiliza a adoção em cada um dos nós da Rede. Métodos de localização de redes móveis celular empregam triangulação do sinal das estações rádio base. Este método não fornece a precisão necessária às aplicações de Redes Sem-fio,

que podem ser formadas distantes de qualquer estação rádio base. A inadequação dos sistemas tradicionais de localização deu impulso ao desenvolvimento de novos métodos de localização para as Redes Sem-fio.

Um método de localização tem por objetivo localizar um receptor num sistema de coordenadas através da aplicação de uma métrica sobre medidas de distância relativa entre este receptor e elementos de referência no sistema. As métricas envolvidas no processo de localização em Redes Sem-fio são as mesmas utilizadas nos sistemas tradicionais de localização: triangulação ou a trilateração das medidas de distância relativa. As medidas de distância podem ser obtidas pela técnica baseada no tempo de chegada ou na potência do sinal recebido. Caso se utilize a triangulação a técnica de medida do ângulo de chegada do sinal deverá ser também aplicada. Os sistemas tradicionais de localização utilizam três ou mais elementos de referência para a localização de um nó receptor. Em Redes Sem-fio, poucos elementos de referência são utilizados para se localizar dezenas ou centenas de nós de posição desconhecida. Medidas de distância relativa, mesmo entre nós de posição desconhecida, são relevantes para obtenção de uma localização mais precisa. Com a técnica baseada em tempo de chegada se obtém melhores resultados do que com a técnica baseada na potência do sinal, pois está sujeita a menos interferências. Porém é uma técnica mais cara, pois exige a adoção de alguma forma de sincronia de relógios na rede, ou da implementação de sensores de ultra-som para medidas de tempo. As medidas de distância relativa entre nós da rede são processadas por algoritmos de localização que se baseiam em uma das métricas para otimizar a posição do nó procurado. Inúmeras pesquisas têm sido feitas em novos algoritmos de localização e geralmente o procedimento adotado é o de identificar a posição de maior probabilidade de um determinado nó se encontrar, ou então, o de minimizar uma função custo que compara grandezas medidas com as teoricamente esperadas dado um modelo para o canal, [Patwari et al., 2001], [Wann and Chin, 2007],[Patwari et al., 2005], [Patwari and Hero, 2003], [Patwari et al., 2003], [Savvides et al., 2001], [Costa et al., 2006]. O processamento pode se dar de forma conjunta ou iterativa. Algoritmos de localização conjunta otimizam a posição de todos os nós procurados simultaneamente e devem ser executados de forma centralizada. Algoritmos iterativos localizam um nó por vez e podem ser executados de forma distribuída na rede.

1.2 Proposta

Este trabalho propõe a elaboração e avaliação de diferentes algoritmos de localização baseados no método numérico de otimização não-linear de Nelder-Mead capazes de localizar nós em Redes Sem-fio utilizando apenas a técnica de Potência de Sinal Recebido, (*Received Signal Strength Indication*) *RSSI*. Apesar da técnica baseada em tempo de chegada proporcionar resultados de localização mais precisos, a informação de potência de sinal recebido é comumente disponibilizada nas interfaces de rádio e não requer funcionalidades de circuitos ou de programas adicionais para ser implementada. É comparado o desempenho das implementações do algoritmo de localização de forma conjunta e iterativa, como também são comparadas diferentes funções custo. As diferentes implementações dos algoritmos são avaliadas através de simulações e o desempenho é analisado em termos de erro médio de localização, número de iterações e tempo de convergência para variados cenários de rede. Os algoritmos baseados em Nelder-Mead são comparados com o algoritmo de localização conjunta que utiliza de Gradiente Conjugado proposto por Patwari, [Patwari et al., 2001].

1.3 Organização

Os demais capítulos desta dissertação estão organizados da seguinte forma:

- Capítulo 2: São apresentados os principais métodos e algoritmos de localização, métricas de localização e técnicas de estimativa de distância.
- Capítulo 3: São definidos o algoritmo proposto e as funções custo que dão origem à diferentes variantes. São apresentadas a simbologia e nomenclatura adotada no trabalho.
- Capítulo 4: O método numérico de otimização não-linear de Nelder-Mead é aplicado em algoritmos de localização conjunta e iterativa. As diferentes funções custo são avaliadas e é proposta uma metodologia para sequenciamento dos nós, como também é introduzido um coeficiente de ponderação para a função custo.
- Capítulo 5: O algoritmo proposto é apresentado e comparado com algoritmo de *Gradiente Conjugado* através de uma série de experimentos incluindo uma aplicação prática de localização numa rede *ad-hoc* IEEE 802.11.
- Conclusão: A contribuição é destacada, os principais resultados obtidos são apresentados assim como são sugeridas formas de prosseguimento deste trabalho.

Capítulo 2

Sistemas de Localização

Sistemas de Localização têm por objetivo identificar o endereço de um objeto móvel ou fixo num *sistema de coordenadas*. Pode-se utilizar diferentes sistemas de coordenadas conforme o sistema de localização. O Sistema de Posicionamento Global (*Global Positioning System-GPS*) usa latitude, longitude e altitude para referenciar a posição de um objeto no globo terrestre. Um sistema de localização predial, como numa fábrica, pode usar o mapeamento do edifício em quadrantes como sistema de coordenadas. Assim um objeto pode ser localizado no quadrante C4 por exemplo. Qualquer que seja o *sistema de coordenadas* adotado o *Método de Localização* requer o conhecimento prévio do posicionamento de pelo menos três elementos que servirão de referências no sistema. Os *Métodos de Localização* empregam *Métricas de Localização* como a trilateração ou triangulação. A métrica de trilateração estima a posição do objeto procurado a partir das distâncias medidas deste objeto até os elementos de referência do método de localização. A métrica de triangulação aplica relações trigonométricas sobre as distâncias conhecidas entre os elementos de referência e ângulos e distâncias medidos entre as referências e o objeto a ser localizado. As medições de distâncias e ângulos entre o objeto a localizar e os elementos de referência são obtidas por *Técnicas de Estimativa de Distância Relativa*. As redes sem-fio trouxeram uma complexidade maior aos *Métodos de Localização*, que evoluíram para o emprego de *Algoritmos de Localização* ao invés da aplicação direta das *Métricas de Localização*. São apresentados neste capítulo os principais *Métodos* e *Métricas* de *Localização*, assim como as *Técnicas* para *Estimativa de Distância Relativa*. Há um resumo de dois Sistemas Tradicionais de Localização e são descritos os algoritmos desenvolvidos para as *Redes Sem-Fio*.

2.1 Técnicas de Estimativa de Distância

Existem diferentes métodos para se estimar a distância entre dois objetos: é necessário avaliar alguma grandeza física que guarde uma relação conhecida com a distância. Esta grandeza pode ser: a potência de um sinal acústico, a potência de um sinal de Rádio Frequência (*Radio Frequency-RF*), o tempo que o sinal demora para percorrer a distância entre receptor e transmissor ou ainda o ângulo de recepção do sinal. É importante considerar que a medida de todas estas grandezas estão sujeitas a erros e dependendo da precisão requerida pela aplicação, a utilização de uma delas pode ser mais apropriada que outra. Além disso é requisito que se conheça a distribuição do erro nas medidas para as técnicas adotadas. As seções seguintes

apresentam as principais técnicas utilizadas, pode-se também escolher uma combinação das mesmas.

2.1.1 Técnica Baseada em Tempo de Chegada

A Técnica Baseada em Tempo de Chegada (*Time Of Arrival-ToA*) estima a distância entre dois nós baseada no tempo de trânsito do sinal desde o nó transmissor até um nó receptor. Uma vez que seja conhecida a velocidade de propagação para o tipo de onda utilizado é possível se estimar a distância percorrida num determinado intervalo de tempo. O relógio do nó receptor deve estar sincronizado com o relógio do nó transmissor para que o cálculo do tempo de trânsito seja preciso. O nó transmissor deve informar ao receptor o instante de tempo em que o sinal foi enviado. O tempo de chegada é o instante de tempo correspondente à detecção do Sinal de Linha de Visada (*Line Of Sight-LOS*). Matematicamente o *ToA* é o instante de tempo que maximiza a correlação cruzada entre o sinal recebido e o sinal transmitido, previamente conhecido. De acordo com Patwari [Patwari et al., 2005] experimentos demonstram que para distâncias curtas as medidas de erro no tempo de propagação são gaussianas e podem ser expressas por

$$f(T_{ij} = t | \theta) = \mathcal{N}(t, \bar{T}_{ij}, \sigma_T^2), \quad (2.1)$$

$$\theta = [x_1, y_1, z_1, x_2, y_2, z_2, \dots, x_n, y_n, z_n], \quad (2.2)$$

onde: $\mathcal{N}(t, \bar{T}_{ij}, \sigma_T^2)$ é a notação para o valor em t de uma Função Densidade de Probabilidade (*Probability Density Function-PDF*) gaussiana com média \bar{T}_{ij} e variância σ_T^2 . θ é o vetor de coordenadas dos nós de posição desconhecida, ou simplesmente nós desconhecidos, dado pela equação 2.2. Os erros das medições são causados principalmente por ruído aditivo e multi-percurso, seção 2.2.2. O erro causado por ruído aditivo pode ser combatido por múltiplas medidas ao longo do tempo [Patwari et al., 2005]. O erro causado por multi-percurso é mais difícil de ser combatido porque dependendo da forma que as ondas correspondentes aos múltiplos percursos se combinam pode interferir na identificação do *LOS*.

A necessidade de sincronia de relógio entre os nós é inconveniente e pode gerar uma complexidade indesejada na rede. Normalmente se utiliza uma maneira mais simples de se fazer a medida, conforme mostrado na figura 2.1(a). O sinal é ecoado pelo nó receptor assim que recebido, desta forma é possível se calcular o tempo de trânsito sem a necessidade de sincronizar os relógios na rede.

A distância estimada entre um receptor i e um transmissor j é dado por

$$\hat{d}_{ij} = \frac{((T_3 - T_0) - (T_2 - T_1)) \cdot V}{2}, \quad (2.3)$$

conforme apresentado em [Tseng et al., 2005], onde V é a velocidade de propagação do sinal de *RF*. Para maior precisão adota-se ondas de ultra-som ao invés de *RF*. Assim, erros nas medidas, como os devidos à atrasos ao se ecoar o sinal, passam a não ser tão significativos dado o maior tempo de trânsito. Para que o cálculo possa ser feito exclusivamente no transmissor, sem a necessidade de troca de informação com o receptor, é necessário que se conheça o intervalo de tempo transcorrido no receptor entre T_1 e T_2 e de que este intervalo seja constante.

Caso não se deseje ecoar o sinal pode-se utilizar o *TDoA* (*Time Difference Of Arrival*) (Diferença entre os Tempos de Chegada). É um método que utiliza duas ondas de naturezas

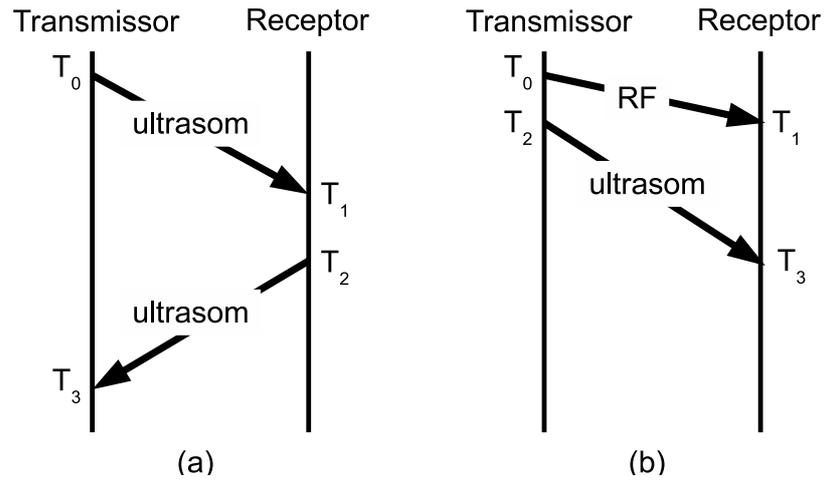


Figura 2.1: Técnica Baseada em Tempo de Chegada: (a) medição do *ToA* ; (b) medição do *TDoA*

diferentes e portanto com diferentes velocidades de propagação, como: *RF* e o ultra-som por exemplo. Este conceito é apresentado na figura 2.1(b). Não há a necessidade de se sincronizar os relógios visto que a distância será proporcional à diferença dos tempos de chegadas entre os dois sinais medidos no receptor. Neste caso a distância estimada é dada por

$$\hat{d}_{ij} = ((T_3 - T_1) - (T_2 - T_0)) \cdot \left(\frac{V_{RF} \cdot V_{US}}{V_{RF} - V_{US}} \right), \quad (2.4)$$

onde V_{RF} e V_{US} são as velocidades de propagação dos sinais de RF e ultra-som respectivamente. Em [Savvides et al., 2001] são apresentados resultados de um algoritmo de localização que adota o *TDoA* como estimador de distância: foi atingida precisão maior que $20cm$ numa rede de 50 nós, dos quais 5 de referência, distribuídos numa área de $15m \times 15m$. Os dados foram processados de forma distribuída por meio de sua proposta de localização colaborativa, seção 2.4.2.

2.1.2 Técnica Baseada em Ângulo de Chegada

A técnica baseada em Ângulo de Chegada (*Angle Of Arrival-AoA*) estima o ângulo de chegada do sinal ao invés da distância relativa entre dois nós. É utilizada em conjunto com *ToA*, *TDoA*. *AoA* requer o uso de um conjunto de duas ou mais antenas ou de receptores de ultra-som dispostos no sensor de tal forma que processando o *ToA* do sinal percebido por cada um deles seja possível determinar o ângulo e direção de chegada do sinal, [Patwari et al., 2005]. Assim como o *ToA* e *TDoA* os erros são causados por ruído aditivo e multi-percurso. Uma complexidade adicional ao se utilizar o *AoA* é que não se sabe a orientação de cada sensor na rede e portanto este é mais um parâmetro a ser determinado, por exemplo, por meio de algum método de otimização linear. É, portanto, uma técnica que agrega precisão às demais técnicas ao custo de mais capacidade de processamento do nó.

2.1.3 Técnica Baseada na Potência do Sinal

O *RSSI* (Indicação de Potência de Sinal Recebido) pode ser utilizado para se estimar a distância entre dois nós. Pode se utilizar tanto *RF* quanto ultra-som, mas é importante se conhecer o comportamento do decaimento da onda escolhida com a distância, assim como os erros a que as medidas de potência do sinal estão sujeitos, [Patwari et al., 2005]. *RF* é normalmente utilizada em redes sem-fio visto que todos os sensores já estão equipados com transmissores e receptores. A seção 2.2 apresenta a modelagem do canal de rádio, utilizada por métodos e algoritmos de localização que usam a potência de *RF* como estimativa de distância.

2.1.4 Técnica Híbrida

A literatura menciona a utilização combinada das técnicas de estimativa de distância, por exemplo, a técnica baseada no tempo de chegada com a técnica baseada no ângulo de chegada ou na potência do sinal. Um tipo de sinal bastante utilizado pela técnica híbrida é o sinal de Banda Ultralarga, *Ultra Wide Band-UWB*, que emprega pulsos extremamente curtos, menor que $1ns$, resultando em um sinal espalhado no espectro de frequências. Um sinal é considerado *UWB* se a razão entre a largura de banda e a portadora for maior que 0.2. Esta característica proporciona uma resolução temporal fina, ideal para aplicações de localização pela técnica baseada em tempo de chegada. O *RSSI* medido entre nós próximos pode ser usado como informação adicional para se melhorar a qualidade da localização, [Gezici et al., 2005]. Em [Wann and Chin, 2007] é descrito um algoritmo de localização que utiliza o *UWB* em técnica híbrida. Empregada-se uma função objetiva não linear onde as medidas de *TOA* recebem peso proporcional à sua confiabilidade. As medidas de *RSSI* influenciam nos valores dos pesos das medidas de *TOA*, inclusive determinando se a mesma corresponde ou não à um sinal de linha de visada.

2.2 Modelagem do Canal de Rádio

Existem diversos modelos de canal de rádio. Aqui são apresentados alguns modelos para o desvanecimento em pequena e larga escala.

2.2.1 Desvanecimento em Larga Escala

O desvanecimento em larga escala corresponde a uma variação lenta da potência do sinal que decai à medida que as antenas receptora e transmissora se afastam. Dois modelos bastante utilizados são os modelos de espaço livre e o de perda de percurso.

Modelo de Propagação de Espaço Livre

O modelo de propagação de espaço livre é usado para prever a potência de sinal recebido quando transmissor e receptor tem uma linha de visada clara e sem obstrução. Sistemas de comunicação via satélite e enlaces de microondas estão sujeitos ao modelo de espaço livre. Como muitos modelos de propagação em larga escala, o modelo de espaço livre prediz que a potência recebida decai como função da distância de separação do transmissor e receptor elevada ao quadrado da potência. A potência no espaço livre recebida por uma antena receptora

que esteja separada da antena transmissora da distância d_{ij} , é dada pela equação de espaço livre de Friis,

$$P_{ij} = \frac{P_j \cdot G_t \cdot G_r \cdot \lambda^2}{(4 \cdot \pi)^2 \cdot d_{ij}^2 \cdot C_L}, \quad (2.5)$$

onde P_j é a potência transmitida em Watts (W), P_{ij} é a potência recebida em Watts (W) e que é função da separação d_{ij} entre o transmissor j e o receptor i dada em metros (m), G_t é o ganho da antena transmissora, G_r é o ganho da antena receptora, λ é o comprimento de onda em metros (m) e C_L é o fator de perda do sistema não relacionado com a propagação, cabos por exemplo.

Log-Distance

Experimentos demonstram que a potência média do sinal decresce de forma logarítmica com o aumento da distância entre transmissor e receptor, quer em ambientes abertos ou fechados. O desvanecimento do sinal é proporcional à d^η , onde d é a distância entre transmissor e receptor e η é o expoente de perda de percurso. Para o espaço livre η é igual a 2.0 e pode variar entre 2.5 e 6.0 para ambientes fechados. Durante o trajeto, além da atenuação pela distância, o sinal eletromagnético está sujeito à diferentes fenômenos à medida que encontra obstáculos: reflexão, difração e dispersão [Rappaport, 1996]. A reflexão ocorre quando o sinal colide com objetos de dimensões muito maiores que o comprimento de onda, como por exemplo: edifícios, paredes e a própria superfície terrestre. A difração ocorre quando o sinal atravessa objetos com superfícies irregulares com arestas ou contornos agudos. Isto resulta em ondas secundárias após o obstáculo, mesmo que não haja linha de visada entre transmissor e receptor. A dispersão do sinal ocorre ao atravessar uma região abundante de objetos pequenos em relação ao comprimento de onda, como por exemplo folhagens. O modelo de perda de percurso estima as perdas do sinal ao percorrer o trajeto d_{ij} entre transmissor j e receptor i e é dado por,

$$PL_{(d_{ij})}^{dB} = PL_{(d_0)}^{dB} + 10 \cdot \eta \cdot \log \frac{d_{ij}}{d_0} + \chi^{dB}, \quad (2.6)$$

onde $PL_{(d_0)}$ são as perdas médias ocorridas na propagação até a distância d_0 próxima ao transmissor. É composto de duas partes: uma determinística, em função da distância; e outra aleatória representada pela variável χ , e que depende do ambiente e da disposição de objetos em volta do transmissor e receptor. Experimentos mostram que a potência do sinal no receptor i posicionado em diferentes pontos, porém à mesma distância d_{ij} do transmissor j , pode variar bastante. Constata-se que a variável aleatória χ , chamada de sombreamento, tem distribuição normal em dB com média zero e variância σ^2 constante com a distância.

Estatisticamente, a potência média de recepção no receptor i afastado de d_{ij} de um transmissor j pode ser modelada como a diferença entre a potência transmitida, P_j , e as perdas ocorridas no percurso, $PL_{(d_{ij})}$:

$$\bar{P}_{ij}^{dBm} = P_j^{dBm} - PL_{(d_{ij})}^{dB}. \quad (2.7)$$

Aplicando-se a equação 2.7 para a distância de referência, d_0 , e isolando-se as perdas no percurso obtém-se a equação

$$PL_{(d_0)}^{dB} = P_j^{dBm} - P_0^{dBm}, \quad (2.8)$$

onde P_0 é a potência à distância d_0 próxima do transmissor. Substituindo-se 2.6 e depois 2.8 em 2.7, obtêm-se a equação

$$\bar{P}_{ij}^{dBm} = P_0^{dBm} - 10 \cdot \eta \cdot \log_{10}\left(\frac{d_{ij}}{d_0}\right) + \chi^{dB}. \quad (2.9)$$

Trata-se de um modelo de grande valia, pois permite simular o comportamento do sinal em larga escala para diferentes tamanhos de redes com diferentes quantidades de nós e variados valores de perda de percurso. A título de exemplificação a tabela 2.1 apresenta uma lista de valores para o expoente de perda de percurso para enlaces de rádio celular, [Rappaport, 1996].

Tabela 2.1: Tabela De Expoentes de Perda de Percurso

ambiente	η
Espaço Livre	2
Área Urbana	2.7 à 3.5
Área Urbana com Sombra	3 à 5
Área fechada com linha de visada	1.6 à 1.8
Área fechada sem linha de visada	4 à 6
Área de Fábrica sem linha de visada	2 à 3

2.2.2 Desvanecimento em Pequena Escala

À medida que o sinal se propaga desde o transmissor até o receptor e reflete ou difrata nos obstáculos, novas ondas são geradas e podem chegar até o receptor com certo atraso em relação ao sinal de linha de visada. Estas diferentes ondas se combinam na antena receptora de forma construtiva ou destrutiva dependendo da frequência de transmissão, largura de banda e duração dos símbolos. O desvanecimento em pequena escala corresponde à uma variação rápida da potência do sinal devido à interferência das ondas correspondentes aos múltiplos percursos. O mesmo ocorre quando o receptor se movimenta em relação ao transmissor: ao longo do percurso o receptor percebe uma frequência diferente da transmitida e passa por vales e picos do sinal conforme ocorram interferências construtivas ou destrutivas devido ao seu movimento. Este é o chamado efeito *doppler* e quanto maior a velocidade mais rápida é a variação do canal e mais sujeito a erros está o receptor. Um canal que apresente estes tipos de interferência é dito seletivo em frequência. A fase, a amplitude e o atraso com que as diferentes ondas atingem o receptor possuem distribuição aleatória e é modelada de diferentes formas dependendo do tipo de canal que se pretende estudar. Os dois modelos de pequena escala mais comuns são: Rayleigh e o Rice. A distribuição Rayleigh é utilizada quando não há um sinal predominante de linha de visada e a distribuição Rice é usada se houver sinal de linha de visada. Este trabalho considera que os erros de medidas causados por multi-percurso são minimizados através de várias medidas no tempo conforme sugerido em [Patwari et al., 2005].

2.2.3 Métodos de Localização

Os métodos de localização processam as medidas efetuadas pelo nó desconhecido até três ou mais nós de referência do sistema, obtidas pelas técnicas de estimativa de distância. A

trilateração e a triangulação são os dois métodos básicos. O terceiro método, a multilateração, é uma generalização da trilateração e muito utilizada em algoritmos de localização em redes sem-fio.

2.2.4 Trilateração

Utilizando-se qualquer técnica para se estimar a distância entre o nó desconhecido e três nós de referência (d_{i1}, d_{i2}, d_{i3}), a trilateração determina a localização geográfica do nó desconhecido a partir da posição previamente conhecida dos nós de referência através da determinação da intersecção de 3 círculos com centro nas referências e raios iguais às estimativas de distâncias [Savvides et al., 2001]. A partir da estimativa da distância d_{i1} até o nó de referência R_1 sabe-se que o nó desconhecido deve estar localizado em algum ponto sobre o círculo de raio d_{i1} . O mesmo ocorrendo para as referências R_2 e R_3 . Sendo assim, a intersecção das circunferências corresponde à localização do nó. A figura 2.2 apresenta este cenário de localização utilizando a trilateração para o caso hipotético onde não haja erros nas estimativas de distância relativa e também para o caso real onde considera-se a ocorrência de erros de medição e então a localização mais provável do nó é em algum ponto da área de intersecção das três circunferências.

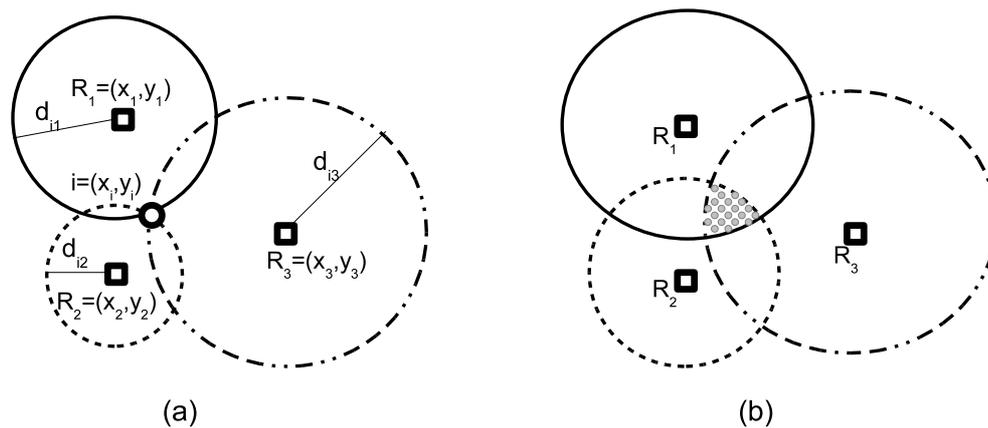


Figura 2.2: Trilateração: (a) caso ideal; (b) caso real com erros

2.2.5 Triangulação

Quando os nós desconhecidos utilizam a técnica baseada em ângulo de chegada, utiliza-se de triangulação para se determinar sua posição [Savvides et al., 2001]. Aplicando-se relações trigonométricas fundamentais de senos e cossenos os ângulos de chegada de um sinal podem ser utilizados para se estimar a posição de um nó desconhecido, se forem conhecidas as posições das referências [Savvides et al., 2001]. A figura 2.3 mostra este conceito.

2.2.6 Multilateração

No caso de um nó desconhecido ser vizinho de diversos nós de referência pode-se aplicar a multilateração que é uma generalização da trilateração [Patwari et al., 2005]. Utiliza-

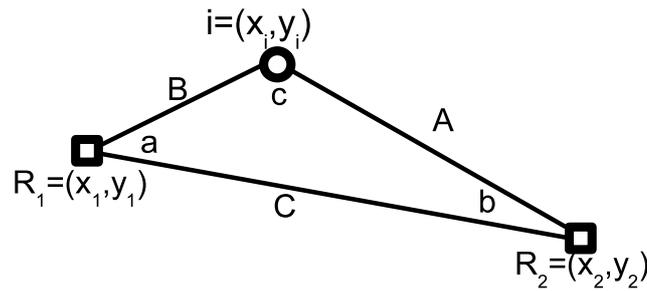


Figura 2.3: Triangulação

se de algum método de otimização para se minimizar a diferença entre as distâncias medidas e otimizadas desde o nó desconhecido i até cada referência, [Savvides et al., 2001]. A figura 2.4 ilustra este conceito.

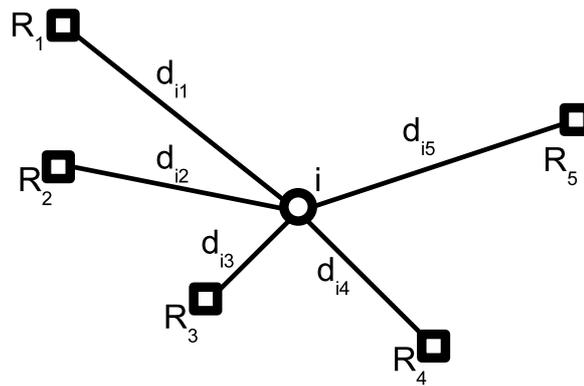


Figura 2.4: Multilateração

2.3 Sistemas Tradicionais de Localização

Os sistemas tradicionais de localização utilizam as técnicas de estimativa de distância relativa e as métricas apresentadas nas seções anteriores. O elemento a ser localizado, um aparelho móvel celular ou um receptor de *GPS*, efetua medidas de distância relativa até os elementos de referência do sistema de localização.

2.3.1 Sistema de Posicionamento Global

O Sistema de Posicionamento Global, *GPS*, foi desenvolvido pelo departamento de defesa americano com o objetivo de se determinar com precisão a posição de um terminal receptor em qualquer ponto da Terra 24 horas por dia, [Beutel, 2005]. Inicialmente foram colocados 28 satélites em órbita sendo que pelo menos quatro deles são vistos de qualquer ponto

do globo à qualquer momento. Cada satélite transmite sua posição, data e hora obtidas a partir de relógio atômico sincronizado entre todos os satélites da rede. O receptor de *GPS* estima sua distância até cada um dos satélites com base no tempo de trânsito do sinal. Usando métricas de triangulação ou trilateração o terminal receptor é capaz de determinar sua posição definida por: latitude, longitude e altitude. Nenhuma comunicação do receptor em direção aos satélites é necessária. O uso do *GPS* em redes sem-fio é limitado devido ao tamanho dos terminais receptores de *GPS* serem bem maiores que o tamanho requerido para a maioria das aplicações, principalmente de sensores. Além disso o consumo extra dos receptores *GPS* pode inviabilizar sua adoção em redes *ad-hoc*, pois utilizam terminais portáteis alimentados por baterias. Outro fator limitante é o fato de que muitas vezes redes sem-fio são instaladas em ambientes fechados onde o sinal de *GPS* pode não estar disponível.

2.3.2 Sistema de Posicionamento Móvel Celular

Cell Based Positioning é o sistema de localização utilizado em redes móveis celulares cujo desenvolvimento foi impulsionado principalmente pelos EUA para provisionamento de localização de assinantes para serviços de emergências. O terminal móvel deve ser capaz de avaliar o enlace até pelo menos 3 estações rádio base e estimar a distância até as mesmas. Sua posição é então estimada num processo de trilateração com base nas coordenadas fornecidas pelas estações rádio base. Atualmente aplicações de *LBS* (Location Based Services - Serviços Baseados em Localização) estão sendo pesquisados com o objetivo de oferecer serviços dependentes da localização do cliente. A aplicação deste sistema em redes sem-fio é bastante restrita devido à grande distância entre as estações rádio base, o que implica em erros nas estimativas de distância maiores que os requeridos para a maioria das aplicações. A localização em redes móveis celulares se dá com precisão de 50m a 100m, [Beutel, 2005].

2.4 Localização em Redes Sem-fio

Um dos maiores desafios dos algoritmos de localização em redes de sem-fio é a estimativa da posição de nós que estejam distantes de qualquer nó de referência e além disso o refinamento destas estimativas, [Patwari et al., 2005]. A localização em redes sem-fio utiliza as mesmas métricas e técnicas de estimativa de distância dos sistemas tradicionais de localização. Porém, com a diferença de que se utilizam medidas de distância relativa entre pares de nós de posição desconhecida. Este procedimento melhora a qualidade da localização e quanto maior a densidade de nós um menor erro de localização é observado. As seções seguintes apresentam a classificação dos algoritmos de localização em redes sem-fio, como também uma tabela resumo dos trabalhos relacionados.

2.4.1 Classificação dos Algoritmos de Localização

Os algoritmos de localização podem ser classificados quanto ao tipo de processamento e quanto à exatidão requerida.

Quanto à Precisão

Os algoritmos podem ser de baixa granularidade ou de alta granularidade, [Krishnamachari, 2005]. Os de baixa granularidade aproximam a posição de um nó processando uma quantidade mínima de níveis de quantização. De acordo com [Patwari and Hero, 2003] a aproximação binária é um caso particular da quantização que utiliza apenas dois níveis. Informações discretas que podem ser utilizadas para aproximar a localização de um nó de forma binária são:

- a existência ou não de conectividade entre dois nós.
- qual entre dois nós está mais próximo de um terceiro.
- um determinado nó está ao sul, norte, leste ou oeste de outro nó.

Apenas 3 níveis de quantização podem ser suficientes para aproximar o desempenho de um sistema contínuo (não-quantizado) em alguns casos [Patwari and Hero, 2003]. Se uma exatidão melhor for requerida mais informações são exigidas assim como a aplicação de algoritmos de alta granularidade que usam as técnicas de estimativa de distância e algoritmos de localização.

Quanto ao Tipo de Processamento

Os algoritmos de localização podem ser classificados como centralizados ou distribuídos, [Patwari et al., 2005]. As informações de estimativa de distância relativa medida até os nós de referência, assim como a identificação dos nós de referência utilizados, endereço de camada de enlace por exemplo, são transmitidas por cada nó desconhecido da rede até um nó central que acumula estas informações para o caso do algoritmo centralizado. Apenas quando o nó central tiver recebido os dados provenientes de todos os nós desconhecidos, é executado o algoritmo de localização. No caso distribuído cada nó procede à sua própria localização baseado nas estimativas de distância relativa medida até os nós de referência e nas coordenadas destes nós. Uma vez que o processo individual de localização finaliza o nó localizado transmite as suas coordenadas até um processador central da rede.

Algoritmos distribuídos necessitam que os nós desconhecidos da rede tenham capacidade de processamento para rodar o algoritmo de localização. Algoritmos centralizados podem sofrer com gargalos na rede devido à quantidade de informação que deve ser transmitida até o nó central para processamento. A precisão e exatidão obtida com cada tipo pode diferir: uma vez que o algoritmo centralizado tem informações de todos os nós desconhecidos e pode proceder a uma localização conjunta de todos eles, seu resultado tende a ser mais preciso. A escolha do tipo de algoritmo a adotar depende, portanto, além da exatidão esperada, da capacidade de processamento exigida de cada nó assim como da largura de banda disponível. Estes fatores devem ser considerados tendo-se em vista o consumo de energia requerido para processamento do algoritmo como também para a transmissão de informações. Sabe-se que as limitações de largura de banda, capacidade de processamento e consumo de energia são um desafio único e que distinguem a rede de sensores sem-fio das demais redes sem-fio. Caso o algoritmo distribuído utilize de refinamento sucessivo, seção 2.4.2, o problema se torna ainda mais complexo: há necessidade de mais capacidade de processamento e intensa troca de informações.

Um terceiro tipo de processamento é uma combinação dos dois anteriores e tenta extrair as vantagens de cada tipo. É utilizado em redes com grande quantidade de nós. A

rede é subdividida em redes menores ou *clusters* que possuem um nó central cada um para processamento da localização. Dentro do *cluster* os nós desconhecidos são localizados de forma conjunta e o nó central do *cluster* é responsável por transmitir as informações até o processador central da rede que executa o processo de localização global. Balanceando-se adequadamente o número de nós em cada *cluster* espera-se ser possível poupar energia através de um maior equilíbrio entre o processamento exigido e a largura de banda requerida em toda a rede sem comprometer o resultado da localização.

2.4.2 Quanto ao tipo do Algoritmo

À medida que pesquisas foram sendo realizadas e trabalhos foram sendo publicados uma nova classificação relacionada com a forma como os dados são processados foi criada para particularizar principalmente os diferentes tipos de algoritmos distribuídos. A localização do tipo conjunta [Patwari et al., 2001], [Patwari et al., 2003] localiza todos os nós simultaneamente enquanto que a localização do tipo multilateração iterativa localiza um nó por vez. Portanto a localização conjunta necessita de processamento centralizado, e localização iterativa pode ser implementada tanto de forma centralizada quanto distribuída [Savvides et al., 2001]. Os tipos multilateração colaborativa e refinamento sucessivo são melhorias realizadas sobre a localização iterativa. Apesar de não haver um impedimento para a implementação da localização iterativa e de seus tipos derivados de forma centralizada é recomendado que seja utilizado um processamento distribuído para estes tipos de algoritmos a fim de poupar energia e banda disponível, [Savvides et al., 2001], [Costa et al., 2006]. Por outro lado é possível obter um melhor resultado com a localização iterativa se for usado processamento centralizado e houver uma ordenação na localização dos nós que proporcione um menor erro de localização, [Savvides et al., 2001]. A seguir as classificações dos algoritmos por tipo são detalhadas.

Multilateração Atômica

Processo básico de localização descrito em [Savvides et al., 2001] onde um nó é capaz de estimar sua posição se atender o requisito básico de estar a um salto de distância de pelo menos 3 nós de referência. Assim que a rede entra em operação os nós de referência passam a fazer anúncios em *broadcast* de suas coordenadas cartesianas, (x_j, y_j, z_j) , a intervalos regulares. Qualquer nó desconhecido que esteja na vizinhança de pelo menos três nós de referência é capaz de estimar sua posição através de alguma das técnicas de estimativa. Caso o algoritmo adotado seja distribuído, os nós desconhecidos podem individualmente estimar suas posições através de trilateração ou multilateração baseados nas coordenadas anunciadas pelos nós de referência. No caso de algoritmo centralizado as medidas de distâncias relativas devem ser transmitidas até um nó central para processamento.

Multilateração Iterativa

O processo é descrito em [Savvides et al., 2001] e [Patwari et al., 2005] e tem a multilateração atômica como princípio básico de funcionamento. A multilateração iterativa é o algoritmo de localização no qual os nós são localizados um a um e uma vez que um nó desconhecido seja localizado ele passa a se comportar como referência para os demais nós. É possível, portanto, que um nó seja localizado mesmo que esteja distante de qualquer nó de referência. É um algoritmo que pode operar tanto de forma centralizada quanto distribuída ou

até mesmo em *cluster*, onde o nó central do *cluster* proceder à sua localização e à localização dos demais nós desconhecidos dentro do *cluster*. Os nós desconhecidos estimam sua distância até cada um dos nós de referência em sua vizinhança utilizando uma das técnicas de estimativa de distância relativa. O algoritmo utiliza grafos e adota diferentes pesos para as bordas conforme a distância estimada até os nós de referência, assim a informação de estimativa de distância proveniente de um nó de referência próximo tem peso maior que a de um nó de referência distante. Baseado nas estimativas de distância e nas coordenadas anunciadas pelos nós de referência, os nós desconhecidos podem estimar suas próprias posições. Uma vez que um nó desconhecido finaliza o seu processo de localização ele passa a se comportar como referência e também anuncia suas coordenadas em *broadcast* para que outros nós desconhecidos possam estimar suas coordenadas mesmo que não sejam vizinhos de nenhum nó de referência. Para o caso do algoritmo centralizado, o processo de localização se inicia pelo nó que tem o maior número de nós de referência ao seu redor. Já para o caso do algoritmo ser distribuído, a localização é paralela: cada nó desconhecido inicia o processo de localização assim que tiver recebido anúncio das coordenadas de pelo menos três nós de referência em sua vizinhança. O algoritmo iterativo tem a desvantagem de carregar um erro cumulativo na localização causado pelo fato dos nós desconhecidos serem usados como referência e o erro de sua localização acarretar erro na localização dos demais nós desconhecidos que utilizem suas coordenadas como referência. Este problema é bastante diminuído para o algoritmo centralizado um vez que é possível se estabelecer uma ordem mais conveniente para a localização dos nós.

Multilateração Iterativa com Refinamento Sucessivo

Este conceito descrito em [Patwari et al., 2005] estabelece que um nó já localizado pode ainda tentar refinar sua posição utilizando adicionalmente as coordenadas anunciadas de outros nós desconhecidos que tenham sido localizados recentemente e passado a se comportar como referências. Quando um nó estabelece sua nova posição ele volta a anunciá-la e os nós em sua vizinhança podem proceder ao refinamento de suas próprias coordenadas sucessivamente até que haja convergência.

Multilateração Colaborativa

A Multilateração Iterativa funciona enquanto o requisito básico da multilateração atômica for atendido, ou seja, se houver para cada iteração pelo menos um nó desconhecido que esteja na vizinhança de pelo menos três nós de referência. Normalmente esta condição é satisfeita, visto que o número de nós de referência aumenta a cada iteração. Mas existe a possibilidade de haver um ou mais nós desconhecidos que não tenham pelo menos três nós de referência como vizinhos. Neste caso aplica-se a multilateração colaborativa, [Savvides et al., 2001] que se utiliza da informação de localização dos nós de referência através de múltiplos saltos. Para isso se identifica um sub-grafo que possa ser escrito de forma que as distâncias entre nós e suas posições formem um conjunto de equações quadráticas redundantes com solução única para as posições dos nós desconhecidos e que possa ser resolvido por meio de MMSE ou outro algoritmo de busca. A multilateração colaborativa oferece suporte para localização nos pontos da rede onde a condição de multilateração atômica não seja satisfeita. Os autores concluem que a utilização do *TDMA* proporciona uma precisão muito maior do que o uso do *RSSI* e optam por um algoritmo de processamento do tipo distribuído devido às vantagens de melhor distribuição

do consumo de energia, maior tolerância a falhas, e menor possibilidade de formação de gargalos próximo ao nó central da rede.

Localização Multi-dimensional Escalonável

Em [Costa et al., 2006] é descrito um algoritmo distribuído e ponderado com base em *MDS* chamado de *dwMDS- (distribute weighted Multidimensional Scaling)*. O algoritmo adaptativamente enfatiza as medidas de distância relativa mais precisas e naturalmente considera as restrições de comunicação na rede. É um algoritmo de refinamento sucessivo, onde cada nó escolhe um conjunto de nós vizinhos, procede a sua localização e a retransmite em forma de anúncio aos nós vizinhos. O objetivo da *MDS Multidimensional Scaling* é o de encontrar uma representação, por exemplo a posição de sensores, para um grupo de objetos na menor dimensão possível de modo que *dissimilaridades* entre os objetos se enquadrem tão bem quanto possível num grupo de *diferenças medidas*, por exemplo por *RSSI*, entre os pares de nós. *MDS* pode ser usado para encontrar um mapa de posições de sensores quando as dissimilaridades são medidas de distâncias relativas obtidas por *RSSI*, *ToA*, *TDoA* ou uma combinação destas, [Costa et al., 2006].

Localização Conjunta

Em [Patwari et al., 2001] é utilizado o *RSSI* como estimador de distância relativa entre os nós. A posição de todos os nós é estimada simultaneamente. Através de simulações e medidas de laboratório é demonstrado que é possível se localizar nós numa rede, mesmo que distantes de nós de referência, com precisão da ordem do afastamento entre os nós. As medidas de *RSSI* mesmo entre nós desconhecidos melhora a qualidade da localização e são necessários poucos nós de referência para que a localização tenha qualidade. As simulações comprovam que em redes onde a densidade de nós desconhecidos é maior, o erro médio de localização diminui, mesmo sem aumentar o número de referências. O trabalho equaciona um Estimador de Máxima Verossimilhança (*Maximum Likelihood Estimator-MLE*) que indica a probabilidade de um dado nó se encontrar em determinada posição. Primeiramente é calculada a verossimilhança L_{in} que indica a probabilidade, dado que uma posição candidata é a verdadeira, de que a potência recebida \bar{P}_{ij} estaria dentro de um Δ_p do valor médio esperado naquela posição \tilde{P}_{ij} :

$$L_{in} = \prod_{i=1}^n \prod_{j \in H_i, j \neq i} \left\{ \exp\left[-\frac{1}{2} \cdot \left(\frac{\bar{P}_{ij} - \tilde{P}_{ij}}{\sigma_{dB}}\right)^2\right] \cdot \Delta_p \right\}, \quad (2.10)$$

onde H_i é o conjunto de nós vizinhos que o nó i é capaz de detectar. Assume-se que se a potência recebida estiver abaixo de um determinado valor P_{thr} o nó transmissor correspondente não pode ser detectado. A verossimilhança L_{out} é a probabilidade, dado que a posição candidata seja a verdadeira, de que a potência recebida para $j \notin H_i$ seja menor que P_{thr} :

$$L_{out} = \prod_{i=1}^n \prod_{j \in H_i, j \neq i} \left\{ Q\left[\frac{\bar{P}_{ij} - P_{thr}}{\sigma_{dB}}\right] \right\}, \quad (2.11)$$

onde $Q[x]$ é a área nos extremos da distribuição normal x distante da média. A função de verossimilhança total é dada pelo produto de L_{in} e L_{out} . As coordenadas ML são dadas por

$$X, Y, Z = \arg \min_{X, Y, Z} [f(x_k, y_k, z_k)] \quad (2.12)$$

$$f(x_k, y_k, z_k) = \frac{b^2}{8} \cdot \sum_{i=1}^n \sum_{j \in H_i, j \neq i} \ln^2 \frac{\hat{d}_{i,j}^2}{\tilde{d}_{i,j}^2} - \sum_{i=1}^n \sum_{j \in H_i, j \neq i} \ln \left[Q \left(\frac{b}{2} \ln \frac{d_{thr}^2}{\tilde{d}_{ij}^2} \right) \right], \quad (2.13)$$

onde

$$b = \frac{10 \cdot \eta}{\ln(10) \cdot \sigma^{dB}} \quad (2.14)$$

e

$$d_{thr} = d_0 \cdot 10^{\frac{p_0 - p_{thr}}{10 \cdot \eta}}. \quad (2.15)$$

As coordenadas que maximizem o *MLE* são as coordenadas de maior probabilidade de o nó se encontrar e a sua busca é feita utilizando-se de gradiente conjugado.

2.4.3 Trabalhos Relacionados

Tabela 2.2: Classificação dos Trabalhos Relacionados

	algoritmo	erro médio de localização(m)
processamento tipo método de solução referência	centralizado localização conjunta Gradiente Conjugado-MLE [Patwari et al., 2003]	rede A RSSI - 2.18 ToA - 1.23
processamento tipo método de solução referência	distribuído localização por refinamento sucessivo dwMDS [Costa et al., 2006]	rede A RSSI - 2.48 ToA - 1.12
processamento tipo método de solução referência	distribuído localização colaborativa MMSE [Savvides et al., 2001]	rede B TDoA - 0.20

A tabela 2.2 apresenta um resumo dos principais trabalhos na área e a classificação dos mesmos. Estes trabalhos têm foco em redes de sensores fixos, mas os resultados podem ser generalizados para redes sem-fio. Quanto à precisão, todos os trabalhos listados são classificados como de alta granularidade. Portanto a tabela apresenta a classificação do tipo do processamento e do tipo de algoritmo apenas. Para cada caso é destacado o método adotado para a busca de solução do problema de localização assim como a citação do trabalho. A terceira coluna apresenta o erro médio de localização em metros para cada técnica de medida de distância relativa adotada, *ToA*, *RSSI* ou *TDoA*. Os autores realizaram seus experimentos em redes com as seguintes características:

- Rede A: Rede real de 14m x 14m com 4 nós de referência e 40 nós desconhecidos. $\sigma_{dB}/\eta = 1.7$ e $\sigma_T = 6.1ns$;

- Rede B: Rede Simulada com 15m x 15m, 5 nós de referência e 45 nós desconhecidos. $\sigma_{cm} = 2cm$;

onde para a rede A σ é o desvio padrão das medidas de potência, η é o expoente de perda de percurso e $\sigma_T = 6.1ns$ é o desvio padrão das medidas de ToA observados. E para a rede B, σ_{cm} é o valor do desvio padrão de erro para as estimativas de distância entre dois nós dado em cm. O desempenho do algoritmo foi avaliado através de simulação, na qual se utilizou o desvio padrão obtido pelo autor através de medidas práticas entre os sensores. A maior precisão obtida pelo algoritmo de [Savvides et al., 2001] se deve ao uso de sensores equipados com emissores e receptores de ultra-som possibilitando o uso do $TDoA$.

2.5 Conclusão

Este capítulo fez uma revisão dos sistemas e métodos de localização, inclusive das técnicas de estimativa de distância e métricas de localização. Discuti os algoritmos utilizados em localização em redes sem-fio e apresentou trabalhos relacionados e que refletem o estado da arte na área. Foi discutido ainda, o modelo de larga escala para o canal de rádio, que se mostrou bastante importante neste trabalho. Os conceitos e termos aqui introduzidos são utilizados nos próximos capítulos onde se apresenta o algoritmo em estudo e se discutem os resultados obtidos.

Capítulo 3

Elaboração de Algoritmos de Estudo

Neste capítulo o problema da localização é formalmente expresso, as variáveis do problema são apresentadas, diferentes funções custo são analisadas e o algoritmo é detalhado.

Dada uma rede com m nós de coordenadas de referência e n nós de posição desconhecida, ou simplesmente nós desconhecidos, o objetivo do algoritmo é o de localizar os nós desconhecidos com a maior precisão possível. Assumimos que cada nó da rede utilize a técnica de estimativa de distância baseada na potência do sinal, *RSSI*, e que associe este valor ao endereço de camada de enlace, endereço *MAC*, do nó transmissor. Primeiramente se elegem coordenadas candidatas a cada nó desconhecido e então se calculam os valores esperados de algumas grandezas naquela posição. O algoritmo se fundamenta na comparação dos valores medidos e teoricamente calculados destas grandezas. As grandezas envolvidas são as potências medidas, distâncias estimadas e as potências e distâncias teóricas calculadas através do modelo *log-distance*. A comparação se dá por meio da chamada função custo e quanto menor seu valor uma melhor localização é esperada. O valor da função custo depende, portanto, das coordenadas candidatas a nó desconhecido, e localizar um nó compreende o processo de busca pelas candidatas que minimizem a função custo. Este processo é feito por meio do método numérico de otimização não-linear de Nelder-Mead [Nelder and Mead, 1965] ou Método do Poliedro Flexível: este método permite minimizar uma função custo em um espaço multi-dimensional, encontrando a solução ótima para um problema de N variáveis. Baseia-se no conceito simplex, que é um poliedro de $N + 1$ vértices em N dimensões. O anexo A traz um resumo do método focando no algoritmo proposto. A tabela 3.1 apresenta o significado das variáveis utilizadas pelo algoritmo de localização.

A figura 3.1 apresenta graficamente a notação para as diferentes distâncias envolvidas no processo de localização, e usada no cálculo da função custo nos diferentes algoritmos e variantes que serão discutidos posteriormente. A figura apresenta 3 nós de referência identificados por R_1, R_2, R_3 . As circunferências ao redor das referências representam a distância que o nó desconhecido i estima estar de cada uma delas. O círculo na área de intersecção das circunferências representa a posição real do nó desconhecido e o triângulo uma posição candidata ao mesmo. A distância entre uma referência j e a posição real do nó desconhecido i é identificada como d_{ij} . A distância entre a posição real do nó desconhecido e sua posição candidata corresponde ao erro de localização para a iteração corrente. A distância euclidiana entre uma referência j e a posição atualmente candidata ao nó desconhecido é identificada como \tilde{d}_{ij} e dada por

$$\tilde{d}_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \quad (3.1)$$

Tabela 3.1: Simbologia Utilizada

Símbolo	Significado
i	i -ésimo nó de posição desconhecida ou simplesmente nó desconhecido.
j	j -ésimo nó de coordenadas de referência ou nó de referência.
m	número de nós de referência
n	número de nós desconhecidos
p	para a localização iterativa é o número de nós desconhecidos já localizados
P_0	potência de referência obtida à uma distância d_0 do nó transmissor
d_0	distância de referência próxima ao nó transmissor
\bar{P}	matriz de Potências medidas
\bar{P}_{ij}	potência média experimentada pelo nó i devido às transmissões do nó j , $\bar{P}_{ij} \in \bar{P}$
\tilde{D}	matriz de distâncias euclidianas obtida a partir dos vetores θ e ω
\tilde{d}_{ij}	distância euclidiana calculada entre nó i e nó j , $\tilde{d}_{ij} \in \tilde{D}$
\hat{D}	matriz de distâncias estimadas obtidas a partir da equação 3.2
\hat{d}_{ij}	distância estimada entre nó i e nó j pelo modelo de Perda de Percurso, $\hat{d}_{ij} \in \hat{D}$
d_{ij}	$d_{ij} \in D$, distância real entre nó i e nó j
\tilde{P}	matriz de potências teóricas obtida a partir da equação 3.4
\tilde{P}_{ij}	potência média teórica esperada pelo nó j devido às transmissões de i , $\tilde{P}_{ij} \in \tilde{P}$
ω	vetor de coordenadas dos nós de referência
θ	vetor de coordenadas dos nós desconhecidos
P_t	potência de transmissão de cada nó da rede
χ	variável de sombreamento no modelo de perda de percurso em larga escala
σ	desvio padrão do sombreamento no modelo de perda de percurso
η	expoente de perda de percurso no modelo de perda de percurso
σ_{erro}	desvio padrão do erro de localização em metros
σ_{erro}^2	variância do erro de localização em metros quadrados

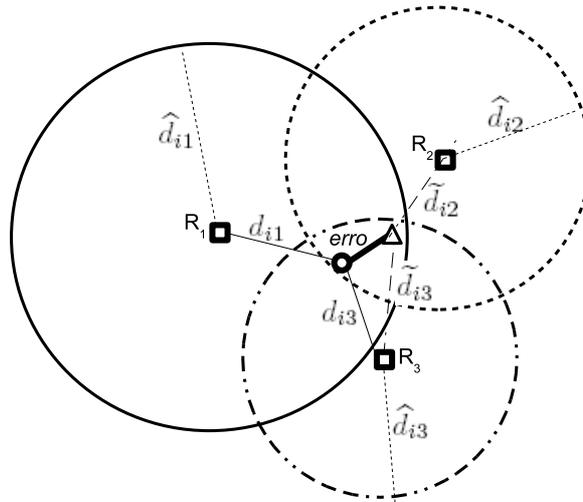


Figura 3.1: Nomenclatura adotada para as diferentes distâncias no processo de multilateração

A distância estimada entre o nó desconhecido i e cada referência j é designada por \hat{d}_{ij} e obtida a partir do modelo de perda de percurso determinístico:

$$\hat{d}_{ij} = 10^{\frac{P_0 - \bar{P}_{ij}}{10 \cdot \eta}}, \quad (3.2)$$

onde \bar{P}_{ij} é o valor médio de *RSSI* recebido em i devido às transmissões de j , P_0 é o valor de potência à distância de referência de $1m$ do transmissor dado em *dBm* e η é o expoente de perda de percurso.

Três diferentes funções custo foram usadas com objetivo de realizar comparações. A implementação de cada uma delas dá origem a três diferentes variantes do algoritmo de localização, cujos comportamentos serão comparados nos capítulos 4 e 5 em diferentes cenários com o algoritmo de localização conjunta baseado no Gradiente Conjugado. A razão de se utilizar o algoritmo de localização conjunta como parâmetro de comparação é a de que este algoritmo também utiliza apenas o *RSSI* para se estimar distância relativa entre pares de nós. As funções custo elaboradas para uso no algoritmo empregando otimização de Nelder-Mead são: Mínimos Quadrados, MLE, Multilateração.

A variante de Mínimos Quadrados compara a diferença entre as potências médias medidas, \bar{P} , e as potências médias teóricas esperadas, \tilde{P} :

$$f_{CMSE} = \sum_{i=1}^n \sum_{j=1, j \neq i}^{m+n} (\bar{P}_{ij} - \tilde{P}_{ij})^2, \quad (3.3)$$

onde \tilde{P}_{ij} é a potência teórica esperada em i devido à uma transmissão de j se estiverem separados de \tilde{d}_{ij} calculado com base no modelo de perda de percurso determinístico

$$\tilde{P}_{ij}^{dBm}(\tilde{d}_{ij}) = P_0^{dBm} - 10 \cdot \eta \cdot \log_{10}(\tilde{d}_{ij}). \quad (3.4)$$

Cabe ressaltar que é possível se utilizar dos *Mínimos Quadrados*, pois a distribuição de \bar{P}_{ij} é gaussiana. Com a finalidade de diferenciar as variantes do algoritmo em estudo, estabeleceu-se uma nomenclatura para as mesmas. À variante do algoritmo originada pelo uso desta função custo dá-se o nome de *NM-Mínimos Quadrados*.

A função custo da multilateração é definida como a diferença entre a distância estimada individualmente entre cada nó desconhecido i e cada nó de referência j , \hat{d}_{ij} , e a distância euclidiana obtida na posição atualmente candidata ao nó i pelo processo de otimização, \tilde{d}_{ij} ,

$$f_{c_{ML}} = \sum_{i=1}^n \sum_{j=1, j \neq i}^{m+n} |\hat{d}_{ij} - \tilde{d}_{ij}|, \quad (3.5)$$

onde \hat{d}_{ij} e \tilde{d}_{ij} são dados respectivamente por 3.2 e 3.1. Esta é a variante *NM-Multilateração*, pois este é o custo tradicionalmente definido como custo para a multilateração. O vetor θ que minimize a função custo corresponde às estimativas das coordenadas dos nós desconhecidos.

A terceira função custo, também em função da distância, foi derivada a partir de uma simplificação da proposta de Patwari [Patwari et al., 2001] para estimativa de localização com base na formulação do estimador *MLE*. A função custo utilizada considera que todos os nós se comunicam uns com os outros:

$$f_{c_{MLE}} = \sum_{i=1}^n \sum_{j=1, j \neq i}^{m+n} \ln^2 \frac{\hat{d}_{ij}^2}{\tilde{d}_{ij}^2}. \quad (3.6)$$

O mínimo de $f_{c_{MLE}}$ corresponde ao máximo do estimador *MLE* e portanto as coordenadas de θ que minimizem a equação 3.6 são as de maior probabilidade do nó se encontrar. A variante do algoritmo em estudo que utiliza o estimador *MLE* é chamada de *NM-MLE*. O algoritmo proposto por [Patwari et al., 2001] é referenciado por *Gradiente Conjugado-MLE*.

Os dados de entrada do algoritmo são as posições dos m nós de referência representados pelo vetor $\omega = [x_1, y_1, x_2, y_2, \dots, x_m, y_m]$, a matriz

$$\bar{\mathbf{P}} = \begin{pmatrix} \bar{P}_{11} & \bar{P}_{12} & \cdots & \bar{P}_{1(m+n)} \\ \bar{P}_{21} & \bar{P}_{22} & \cdots & \bar{P}_{2(m+n)} \\ \vdots & \vdots & \vdots & \vdots \\ \bar{P}_{n1} & \bar{P}_{n2} & \bar{P}_{n3} & \bar{P}_{n(m+n)} \end{pmatrix}, \quad (3.7)$$

de potências médias medidas e a potência na distância de referência, P_0 , que foi considerada a mesma para todos os nós e obtida na distância de 1m. A saída do algoritmo é o vetor θ de coordenadas dos nós desconhecidos $\theta = [x_1, y_1, x_2, y_2, \dots, x_n, y_n]$. Cada linha da matriz $\bar{\mathbf{P}}$ corresponde a um nó i , $1 \leq i \leq n$, que se pretende localizar e cada coluna corresponde a um nó j , $1 \leq j \leq m+n$, em sua vizinhança. Cada elemento \bar{P}_{ij} de $\bar{\mathbf{P}}$ corresponde à potência média experimentada em i devido às transmissões de j . Por esta razão $\bar{\mathbf{P}}$ é designada de matriz de medidas de potência média. Para as simulações em *Matlab*[®] os valores de \bar{P} foram gerados de acordo com a equação

$$\bar{P}_{ij}^{dBm}(d_{ij}) = P_0^{dBm} - 10 \cdot \eta \cdot \log_{10}(d_{ij}) + \chi. \quad (3.8)$$

É necessário que se estabeleça o ponto inicial de busca, assim requer-se a definição de um vetor $\theta_0 = [x_{1_0}, y_{1_0}, x_{2_0}, y_{2_0}, \dots, x_{n_0}, y_{n_0}]$, como ponto de partida para a função de otimização. Por questão de simplificação o algoritmo é apresentado para o caso bidimensional. A busca por cada um dos nós pode iniciar em um ponto aleatório da rede ou então próximo de um nó de referência.

De posse do vetor θ_0 calcula-se a distância euclidiana entre todos nós da rede: distância entre os próprios nós desconhecidos e distância entre nós desconhecidos e nós de referência. Os valores resultantes compõem a matriz de distâncias

$$\tilde{\mathbf{D}} = \begin{pmatrix} \tilde{d}_{11} & \tilde{d}_{12} & \cdots & \tilde{d}_{1(m+n)} \\ \tilde{d}_{21} & \tilde{d}_{22} & \cdots & \tilde{d}_{2(m+n)} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{d}_{n1} & \tilde{d}_{n2} & \cdots & \tilde{d}_{n(m+n)} \end{pmatrix}. \quad (3.9)$$

Dependendo da função custo a ser utilizada calcula-se a matriz de potência média teórica esperada, $\tilde{\mathbf{P}}$ ou a matriz de estimativas de distâncias, $\hat{\mathbf{D}}$. Caso se utilize a função $f_{c_{MSE}}$ constrói-se $\tilde{\mathbf{P}}$. Caso se utilize as funções custo $f_{c_{MLE}}$ ou $f_{c_{ML}}$ constrói-se $\hat{\mathbf{D}}$.

Para se construir $\tilde{\mathbf{P}}$ emprega-se o modelo de *perda de percurso*, equação 3.4, para obtenção da potência média teórica esperada em cada nó desconhecido devido ao sinal gerado por cada outro nó da rede, para isso são usadas a potência P_0 e a matriz $\tilde{\mathbf{D}}$ de distâncias. Aplicando-se o modelo para todas as combinações possíveis de pares de nós obtém-se a matriz de potências teóricas esperadas entre os nós de cada par:

$$\tilde{\mathbf{P}} = \begin{pmatrix} \tilde{P}_{11} & \tilde{P}_{12} & \cdots & \tilde{P}_{1(m+n)} \\ \tilde{P}_{21} & \tilde{P}_{22} & \cdots & \tilde{P}_{2(m+n)} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{P}_{n1} & \tilde{P}_{n2} & \cdots & \tilde{P}_{n(m+n)} \end{pmatrix}. \quad (3.10)$$

Para se construir a matriz de estimativas de distâncias,

$$\hat{\mathbf{D}} = \begin{pmatrix} \hat{d}_{11} & \hat{d}_{12} & \cdots & \hat{d}_{1(m+n)} \\ \hat{d}_{21} & \hat{d}_{22} & \cdots & \hat{d}_{2(m+n)} \\ \vdots & \vdots & \vdots & \vdots \\ \hat{d}_{n1} & \hat{d}_{n2} & \cdots & \hat{d}_{n(m+n)} \end{pmatrix}, \quad (3.11)$$

emprega-se o modelo de perda de percurso da equação 3.2 para obtenção das estimativas de distância relativa a partir da técnica baseada na potência de sinal.

O modelo de *perda de percurso* usado para construção de $\hat{\mathbf{D}}$ e $\tilde{\mathbf{P}}$, equações 3.2 e 3.4, considera apenas a parcela determinística referente ao valor médio esperado devido ao desvanecimento em larga escala. Não há como se inferir sobre o valor do sombreamento, χ , devido ao seu comportamento aleatório. Este é um fator determinante de erro para qualquer algoritmo de localização que utilize o *RSSI* como base para se estimar a distância relativa entre os nós.

A função de otimização irá realizar uma série de iterações, refinando as coordenadas de θ com o objetivo de se minimizar a função custo. A minimização desta função custo é realizada pelo método numérico de otimização não linear de Nelder-Mead. Diferentes variantes

de função custo são discutidas, mas o princípio básico é o mesmo para todas: o custo deve ser tanto menor quanto mais próxima as coordenadas atribuídas ao nó desconhecido estiverem de seu real valor.

O algoritmo de localização conjunta pode ser condensado nos passos apresentados a seguir:

1. parâmetros de entrada: $\bar{P}, \omega, P_0, \eta$
2. definir um vetor θ_0 como ponto de partida. $\theta = \theta_0$
3. usar θ para calcular a distância até as referências: \tilde{D}
4. Se função custo for $f_{C_{MSE}}$ calcular \tilde{P}
5. Se função custo for $f_{C_{ML}}$ ou $f_{C_{MLE}}$ calcular \hat{D}
6. minimizar a função custo otimizando o vetor θ através do método numérico não linear de Nelder-Mead
7. parâmetro de saída: θ

3.1 Conclusão

O problema de localização foi formalmente expresso, o algoritmo de localização conjunta foi apresentado e as funções custo a serem estudadas foram equacionadas. A simbologia utilizada, assim como a nomenclatura adotada foram definidas. O capítulo 4 utiliza a formulação aqui introduzida nas discussões dos resultados obtidos e reformulação do algoritmo para localização iterativa.

Capítulo 4

Avaliação de Desempenho

Neste capítulo, o desempenho dos diferentes algoritmos de localização elaborados, e das diferentes funções custo, são avaliados em termos de erro médio de localização, número de iterações e tempo até a convergência. A avaliação é feita através de simulações em *Matlab*[®] para diferentes cenários de rede. O desempenho dos algoritmos baseados no método numérico de Nelder-Mead foi comparado com o algoritmo de Patwari [Patwari et al., 2001] que usa Gradiente Conjugado e *MLE*. As seções apresentam-se assim divididas:

- A seção 4.1 - apresenta e discute os resultados do algoritmo em estudo elaborado de forma conjunta;
- A seção 4.2 - baseado nos passos do algoritmo de localização conjunta elaboram-se algoritmos de localização iterativa e se discute resultados de desempenho;
- A seção 4.3 resume os principais resultados obtidos.

Um cenário de rede é composto pela área da rede, número de nós de referência m , número de nós desconhecidos n , desvio padrão σ das medida de potência média \bar{P}_{ij} , expoente de perda de percurso η . A figura 4.1 mostra a topologia de rede utilizada nos testes. A rede tem formato quadrangular e foram utilizados 4 nós de referência: R_1, R_2, R_3 e R_4 simbolizados pelos quadrados e posicionados nos cantos da rede. Os nós desconhecidos são representados por círculos numerados de 1 a n e estão distribuídos uniformemente dentro da área da rede. O número de nós desconhecidos pode variar em cada cenário assim como as variáveis para simulação do ambiente: σ e η . Muitas das simulações foram executadas para a mesma rede para possibilitar comparações, chamada de rede de referência. Esta rede tem $15\text{m} \cdot 15\text{m}$ e 4 nós de referência dispostos nos cantos. De acordo com [Patwari et al., 2001] existe uma constante dada pela razão entre σ e η para cada ambiente. Por exemplo, dentro de uma indústria esta razão é em torno de 2.73, [Rappaport and McGillem, 1989]. A constante K_a ,

$$K_a = \frac{\sigma}{\eta}, \quad (4.1)$$

expressa esta relação. Se a rede de referência estiver inserida num ambiente onde $K_a = 2.73$, define-se esta combinação como cenário de referência.

- rede de referência: $m = 4$, $15 \cdot 15$ metros

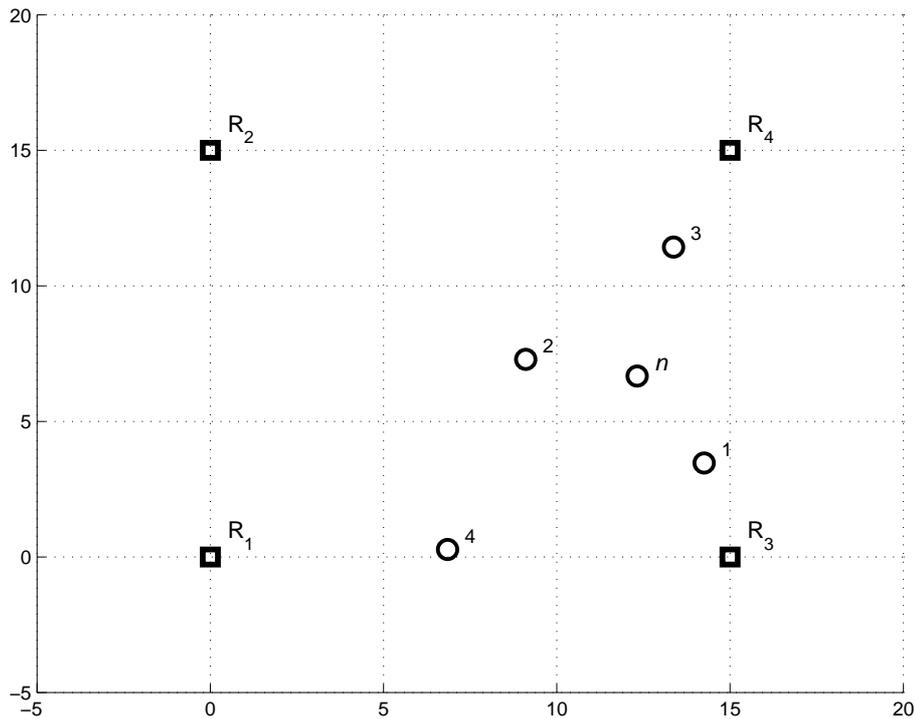


Figura 4.1: Topologia de Rede Utilizada para Simulação em Matlab

- cenário de referência: rede de referência onde $K_a = 2.73$

A avaliação dos algoritmos foi feita através de simulações em *Matlab*[®]. Para n igual a 1, 2, 3, 4, 5 são realizados 1000, 800, 400, 250, 200 experimentos respectivamente e para $n > 5$ são realizados 100 experimentos para que seja gerada uma amostra suficientemente grande de erros de localização. Um experimento compreende: fixar os 4 nós de referência nos cantos da rede, distribuir aleatoriamente os n nós desconhecidos, gerar a matriz de potências médias medidas, \bar{P} , utilizando o modelo de perda de percurso com sombreamento, equação 3.8, e então localizar o nó desconhecido. A menos que especificado em contrário, os nós desconhecidos permanecem na mesma posição ao longo de todos os experimentos. A cada experimento sorteia-se um novo sombreamento e assim cada nó desconhecido é submetido à diferentes valores de \bar{P}_{ij} apesar de permanecer imóvel. Deste modo é possível se simular diferentes ambientes em que a rede possa estar inserida. Por motivos de comparação com o algoritmo de *Gradiente Conjugado-MLE*, o ponto inicial de busca pelo nó desconhecido foi considerado a posição real do nó. É um ponto irreal de partida, mas o algoritmo disponibilizado por Patwari em [Patwari, 2009] e usado como parâmetro comparativo utiliza esta premissa.

4.1 Localização Conjunta

A localização conjunta deve ser executada necessariamente num nó central da rede, pois as coordenadas de todos os nós da rede são otimizadas simultaneamente. Assim é possível localizar as coordenadas de um nó, mesmo que ele esteja distante de qualquer nó de referência, desde que outros nós em sua vizinhança estejam próximos à nós de referência ou que tenham

vizinhos seus que estejam. Assume-se que cada nó desconhecido colete e envie ao nó central a informação de *RSSI* médio de cada um dos outros nós que lhe cause contenção.

4.1.1 Estudo de Cenários com Um Nó Desconhecido

A figura 4.2 compara o erro médio de localização de um único nó desconhecido em função de K_a . Foram feitas simulações para redes com diferentes áreas: $15m \times 15m$, $75m \times 75m$ e $150m \times 150m$. Três diferentes valores de desvio padrão σ foram usados: 3.0, 7.1, 10.0, e três diferentes valores de perda de percurso η também foram usados: 2.0, 2.6, 3.5. De posse desses valores de σ e η pode-se simular a potência média experimentada no nó desconhecido devido ao sinal dos nós de referência. E este experimento foi realizado 1000 vezes para cada valor de K_a para o nó desconhecido fixo. O erro médio é obtido pela média aritmética dos erros de localização observados nos experimentos realizados. As simulações foram realizadas para o algoritmo *Gradiente Conjugado-MLE* de Patwari e para as 3 variantes de custo do algoritmo baseado em Nelder-Mead.

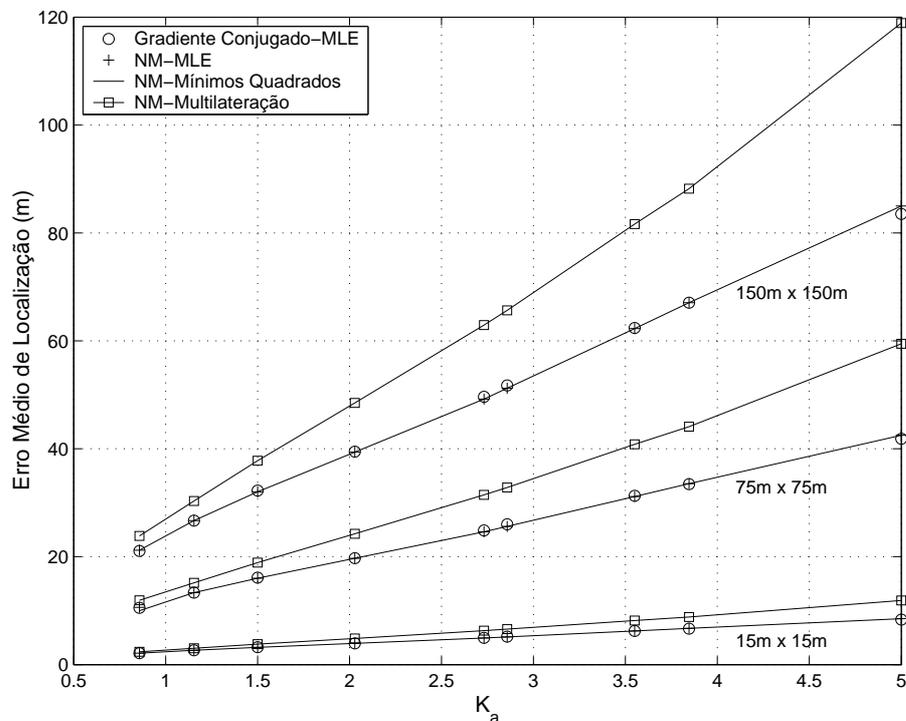


Figura 4.2: Erro Médio de Localização de Um Nó em Redes Diferentes em Função de K_a : $m = 4, n = 1$

Verifica-se que os algoritmos *Gradiente Conjugado-MLE* e o baseado em Nelder-Mead tem resultados muito semelhantes para as variantes *NM-Mínimos Quadrados* e *NM-MLE*. O erro médio é maior para a variante de custo *NM-Multilateração*. Além disso, este experimento comprova que o erro é multiplicativo com a distância: o erro médio de localização para rede de $150m \cdot 150m$ é o dobro do que para a de $75m \cdot 75m$ que por sua vez é 5 (cinco) vezes maior do que para a rede de $15m \cdot 15m$.

A seguir, na figura 4.3, são apresentadas as curvas de Função de Densidade Cumulativa e Função de Densidade de Probabilidade do erro médio de localização para as vari-

antes do algoritmo de localização em Nelder-Mead e também para o algoritmo em *Gradiente Conjugado-MLE* no cenário de referência. A posição sorteada randomicamente para o nó desconhecido e utilizada em todos os experimentos foi de $(x = 14.2519, y = 3.4671)$.

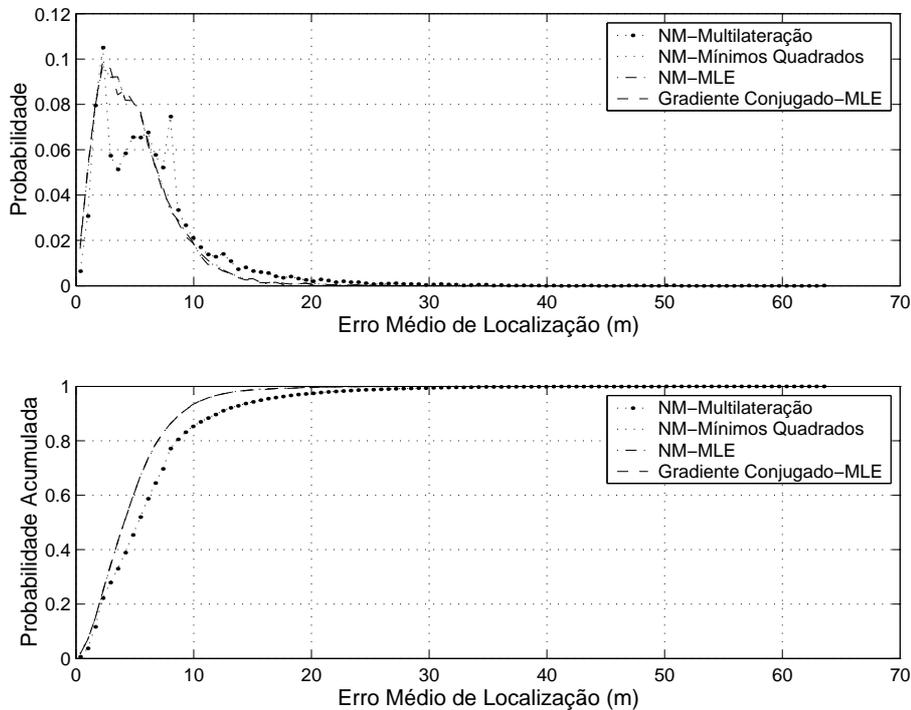


Figura 4.3: PDF e CDF do Erro Médio de Localização de Um Nó Fixo no Cenário de Referência

As curvas de PDF tem uma envoltória similar, se destacando o pico em torno de $2m$ para a variante *NM-Multilateração*. A CDF das variantes *NM-Mínimos Quadrados* e *NM-MLE* e do algoritmo em *Gradiente Conjugado-MLE* tem comportamento muito próximo enquanto que a variante *NM-Multilateração* apresenta uma subida um pouco mais lenta. Isto ocorre porque a PDF da variante *NM-Multilateração* possui um espalhamento maior: seu maior erro de localização foi de aproximadamente $35m$ enquanto que para as variantes *NM-Mínimos Quadrados* e *NM-MLE* foi de aproximadamente $26m$ e para o algoritmo em *Gradiente Conjugado-MLE* foi de $21m$. São tabelados em 4.1 os valores de desempenho dos algoritmos de localização.

Tabela 4.1: Erro Médio de Localização de Um Nó Fixo no Cenário de Referência

Algoritmo	Erro Médio m	σ_{erro}^2 m^2	σ_{erro} m
<i>Gradiente Conjugado-MLE</i>	4.9576	10.0013	3.1625
<i>NM-Mínimos Quadrados</i>	4.9208	10.078	3.1746
<i>NM-MLE</i>	4.9208	10.078	3.1746
<i>NM-Multilateração</i>	6.2912	24.1081	4.9100

Tanto pelos gráficos de *PDF* quanto pelos valores tabelados nota-se que erro de localização apresenta uma dispersão relativamente grande para todos os algoritmos: o desvio padrão é da mesma ordem de grandeza do erro de localização. Esta é uma característica dos algoritmos baseados em *RSSI*. A próxima seção, 4.1.2, mostra que a exatidão da localização baseada

em *RSSI* pode ser melhorada se houver um maior número de nós desconhecidos e se forem utilizadas as medidas de *RSSI* entre nós desconhecidos.

4.1.2 Estudo de Cenários com Diversos Nós Desconhecidos

Comparou-se o desempenho dos algoritmos para a rede de referência com número de nós desconhecidos entre 1 e 8. O erro médio é obtido pela média aritmética dos erros de localização observados nos experimentos realizados. Este estudo foi realizado para o algoritmo *Gradiente Conjugado-MLE* assim como para o algoritmo baseado em Nelder-Mead em suas 3 variantes de custo. A figura 4.4 apresenta as curvas obtidas para cada algoritmo.

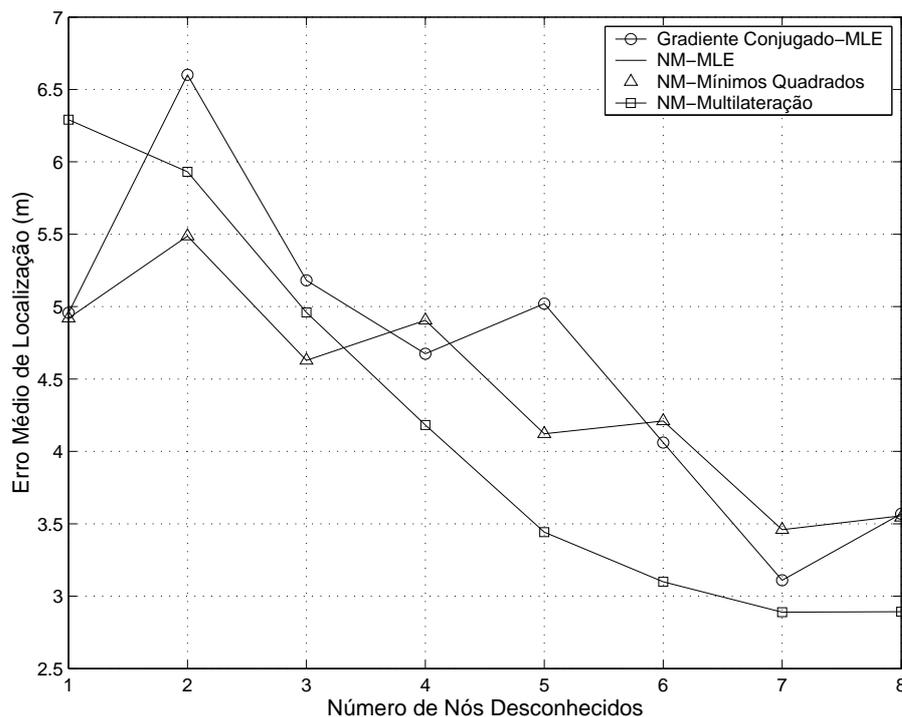


Figura 4.4: Comparação de Erro Médio de Localização Conjunta no Cenário de Referência em Função do Número de Nós para diferentes algoritmos

Verifica-se que tanto para o algoritmo *Gradiente Conjugado-MLE* quanto para o algoritmo baseado em Nelder-Mead com suas variantes de custo há uma tendência de diminuição do erro médio com o aumento do número de nós. Para o algoritmo baseado em Nelder-Mead as variantes de custo *NM-Mínimos Quadrados* e *NM-MLE* apresentaram comportamento idêntico, as curvas se sobrepõem. Interessante observar que as curvas destas variantes tem um comportamento dente de serra à medida que se adiciona um novo nó desconhecido na rede. Isto se deve ao fato de que a posição dos nós desconhecidos foi mantida durante todos os experimentos. À medida que se adicionam nós desconhecidos, a nova topologia pode se configurar desvantajosa para determinados algoritmos. Para o algoritmo baseado em *Gradiente Conjugado-MLE* este comportamento é ainda mais acentuado. A variante *NM-Multilateração* apresentou o maior erro médio de localização para $m = 1$, mas por outro lado este diminui continuamente e apresenta o melhor desempenho nos testes realizados para $n > 3$. Portanto a variante *NM-Multilateração*

tem um maior proveito da proximidade entre os nós e a curva apresenta uma queda suave com o aumento do número de nós desconhecidos.

O fato do algoritmo com base em *Gradiente Conjugado-MLE* e das variantes *NM-Mínimos Quadrados* e *NM-MLE* terem apresentado um erro médio maior para $n = 2$ do que para $n = 1$ pode ser melhor entendido observando-se os gráficos das figuras: 4.5, 4.6 e 4.7. Foram plotados três gráficos, sendo um para *Gradiente Conjugado-MLE* e outros dois para as variantes do algoritmo de localização em *Nelder-Mead*. Em cada gráfico foram traçadas duas curvas do comportamento do erro médio em função de K_a , uma para $n = 1$ e outra para $n = 2$.

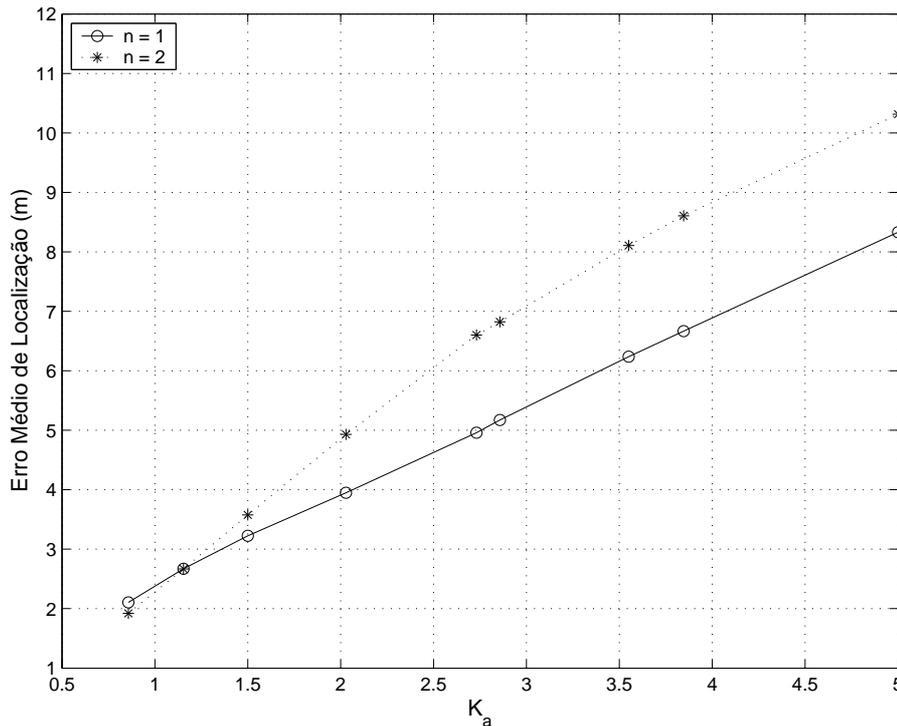


Figura 4.5: Erro Médio de Localização Conjunta para *Gradiente Conjugado-MLE* na rede de referência

Observa-se que dependendo dos valores de K_a as duas curvas de erro médio podem se cruzar. Isto ocorre para o algoritmo em *Gradiente Conjugado-MLE* e para as variantes *NM-MLE* e *NM-Mínimos Quadrados*, mas não ocorre para a variante *NM - Multilateração*. Esta diferença é um indicativo de que o algoritmo em *Gradiente Conjugado-MLE* e as variantes (*NM-Mínimos Quadrados*) e *NM-MLE* são mais sensíveis à qualidade da informação do nó desconhecido vizinho do que a variante *NM-Multilateração*.

A fim de avaliar se um maior valor de erro médio da variante *NM-Multilateração* para $n = 1$ é decorrência da posição em que o nó desconhecido foi fixado, o que poderia ser desfavorável apenas para esta variante, repetimos a análise de desempenho com uma diferença no procedimento: o nó passa a ter uma nova posição randomicamente distribuída a cada experimento. Os resultados obtidos para o cenário da rede de referência para $K_a = 2.73$ são apresentados na tabela 4.2

Analisando a tabela verifica-se que apesar da variante *NM-Multilateração* apresentar um erro médio maior para o nó fixo, esta diferença diminui bastante se o nó tiver uma posição diferente a cada experimento. A diferença de erro entre as variantes *NM-Multilateração* e *NM-*

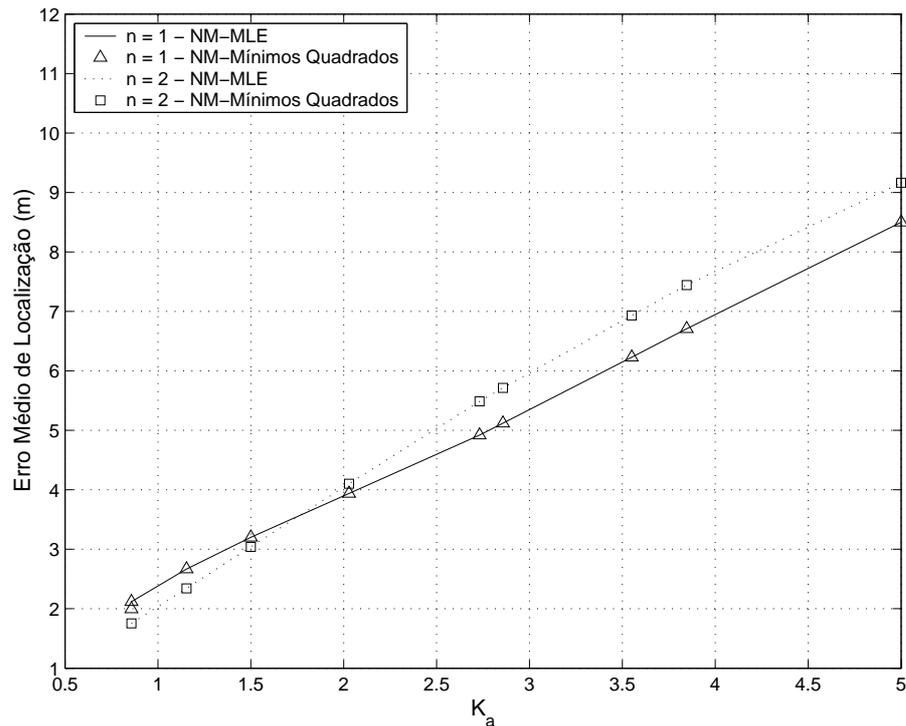


Figura 4.6: Erro Médio de Localização Conjunta para *NM-Mínimos Quadrados* e *NM-MLE* na rede de referência

Tabela 4.2: Erro Médio de Localização de Um Nó Randomicamente Distribuído no Cenário de Referência

Algoritmo	Erro Médio m	σ_{erro}^2 m^2	σ_{erro} m
<i>Gradiente Conjugado-MLE</i>	10.386	28.8311	5.3695
<i>NM-Mínimos Quadrados</i>	5.9381	15.2777	3.9087
<i>NM-MLE</i>	5.9381	15.2777	3.9087
<i>NM-Multilateração</i>	6.4656	25.7005	5.0696

Mínimos Quadrados cai de $1.3704m$ para $0.5275m$. Portanto a posição em que o nó foi fixado realmente penaliza a variante *NM-Multilateração*. Para as variantes do algoritmo em estudo observa-se também que o desvio padrão, σ_{erro} , e a variância, σ_{erro}^2 , do erro de localização são maiores para a variante *NM-Multilateração*. Na verdade, as variantes *NM-Mínimos Quadrados* e *NM-MLE* reproduziram exatamente os mesmos valores de erro médio e σ_{erro}^2 . O algoritmo baseado em *Gradiente Conjugado-MLE* apresentou o maior erro médio e σ_{erro}^2 para este teste, isto significa que o desempenho deste algoritmo varia mais que as variantes do algoritmo baseado em Nelder-Mead quando o nó muda de posição na rede.

4.1.3 Avaliação do Desempenho do Algoritmo de Localização Conjunta

Para a rede de referência foi computado também o número de iterações necessárias até a convergência. Para as variantes do algoritmo em Nelder-Mead o número de iterações

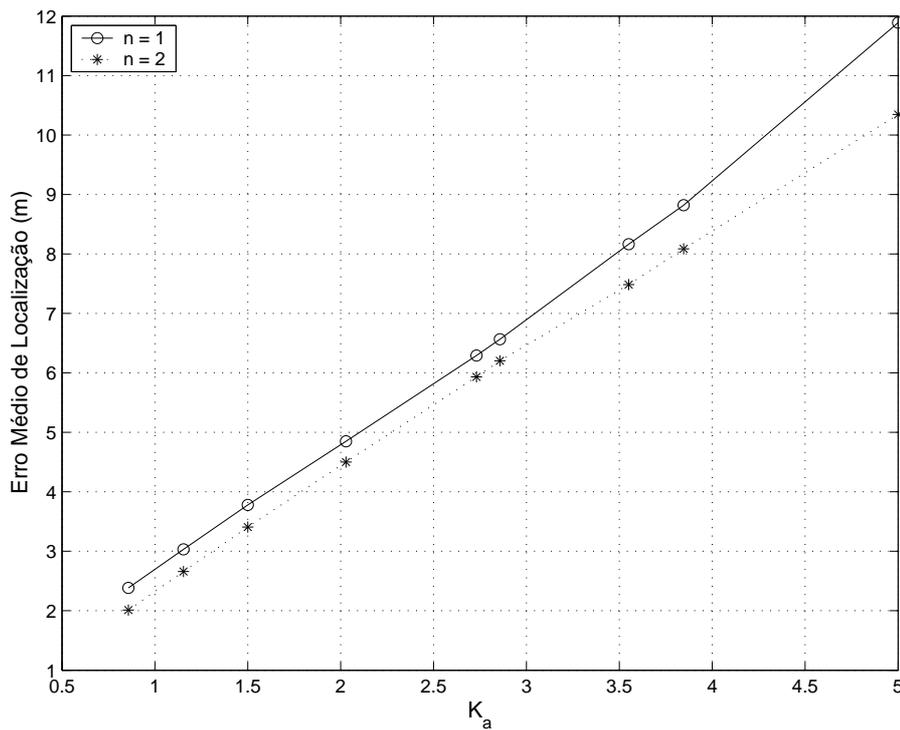


Figura 4.7: Erro Médio de Localização para *NM-Multilateração* na rede de referência

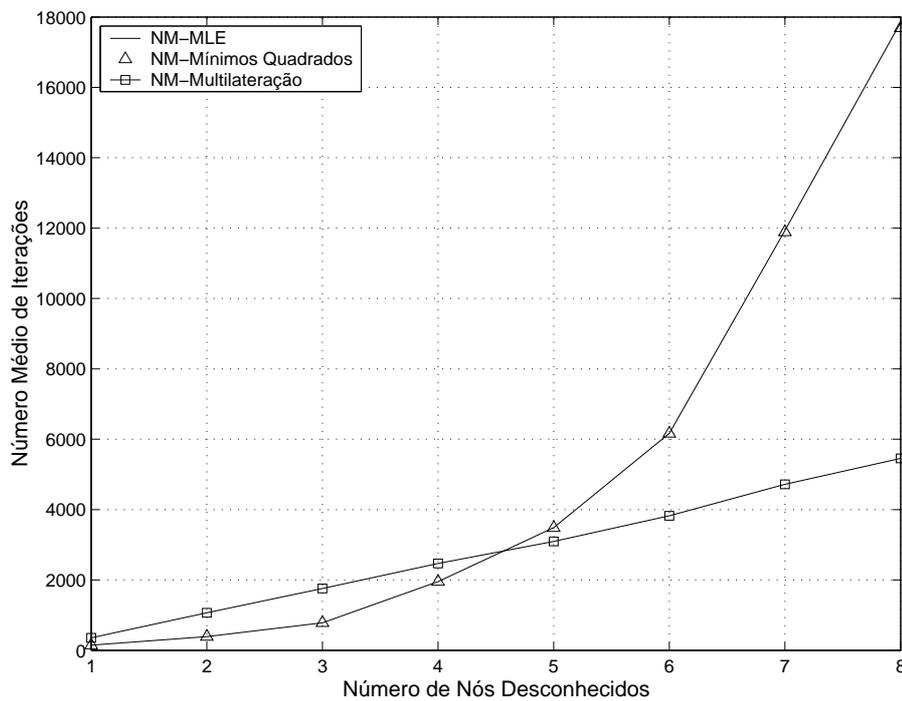


Figura 4.8: Número Médio de Iterações para as Variantes do Algoritmo em Estudo na Localização Conjunta na Rede de Referência

corresponde ao número de vezes que a função custo foi executada. O *Gradiente Conjugado-MLE* faz uso de derivação e o número de iterações computadas é a somatória do número de

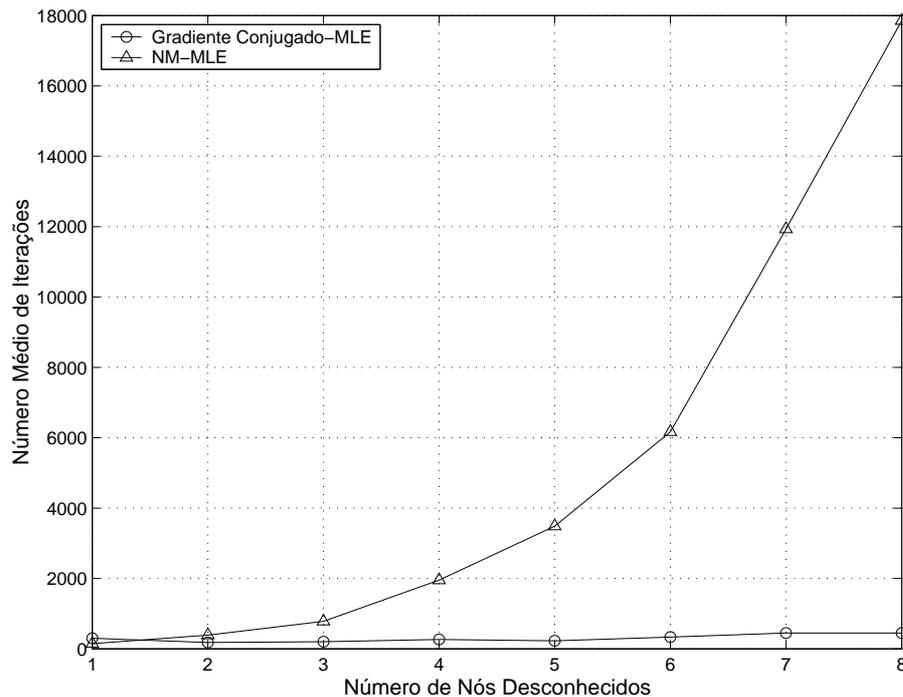


Figura 4.9: Número Médio de Iterações para Algoritmos de Localização Conjunta com custo em função do *MLE* na Rede de Referência

chamadas da função custo e de sua derivada. A figura 4.8 compara os resultados obtidos entre as variantes do algoritmo em estudo. Novamente os algoritmos *NM-Mínimos Quadrados* e *NM-MLE* apresentam comportamento idêntico: o aumento do número de iterações progride geometricamente com o aumento do número de nós. Já para a variante *NM-Multilateração* o aumento do número de iterações é mais lento quando comparado com *NM-Mínimos Quadrados* e *NM-MLE*. A figura 4.9 compara os algoritmos que utilizam *MLE* como função custo: *NM-MLE* e *Gradiente Conjugado-MLE*. Observa-se que o algoritmo por *Gradiente Conjugado-MLE* tem a melhor escalabilidade com o aumento do número de nós, o número de iterações cresce linearmente e muito lentamente quando comparado com o *NM-MLE*.

O tempo de convergência para cada variante do algoritmo em estudo também varia, vide gráfico da figura 4.10. O que este gráfico mostra de interessante é a única diferença observada entre as implementações *NM-Mínimos Quadrados* e *NM-MLE*: o tempo médio para convergência. A figura 4.11 compara o tempo de convergência para os algoritmos que utilizam *MLE* como função custo. O algoritmo de *Gradiente Conjugado-MLE* apresenta um tempo de convergência muito menor que o *NM-MLE*.

O número de iterações e o tempo despendido até a convergência é dependente da função custo, mas não está diretamente ligado ao custo computacional da execução da função custo em si, mas sim ao custo computacional que ela demanda do método numérico não-linear de Nelder-Mead. Os valores de tempos foram medidos em um computador portátil marca DELL modelo Inspiron 1525 equipado com processador *intel*[®] *core 2 duo* T5550, 1.83MHz e 3GB de RAM.

Observa-se que as variantes do algoritmo de localização baseado em Nelder-Mead apresentam um bom desempenho em termos de erro médio de localização quando comparado ao *Gradiente Conjugado-MLE*, visto que o erro médio é menor ou está muito próximo dos

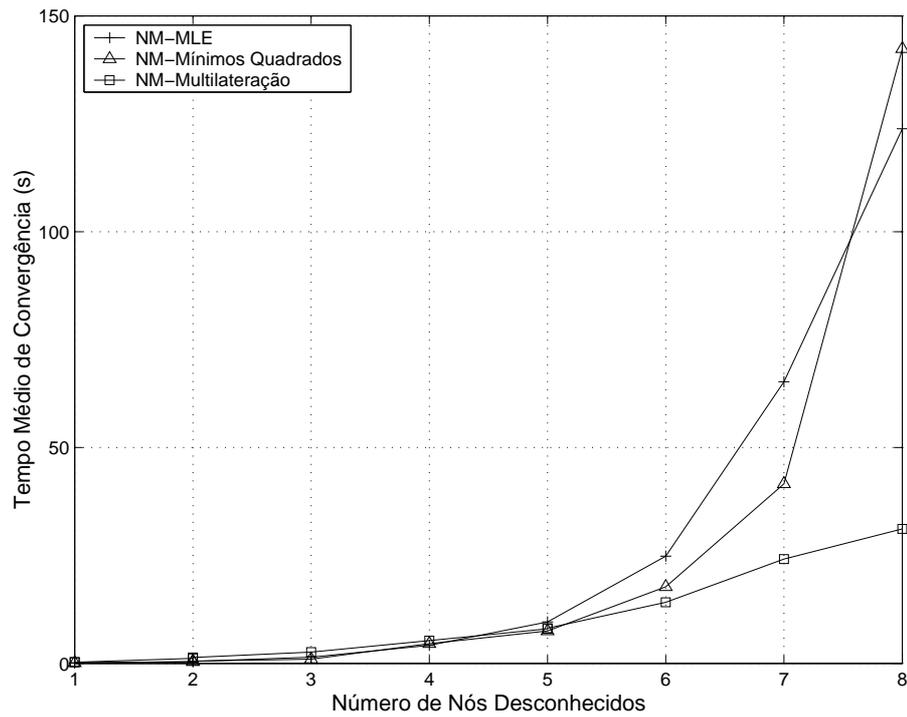


Figura 4.10: Tempo Médio de Convergência para as Variantes do Algoritmo em Estudo na Localização Conjunta na Rede de Referência

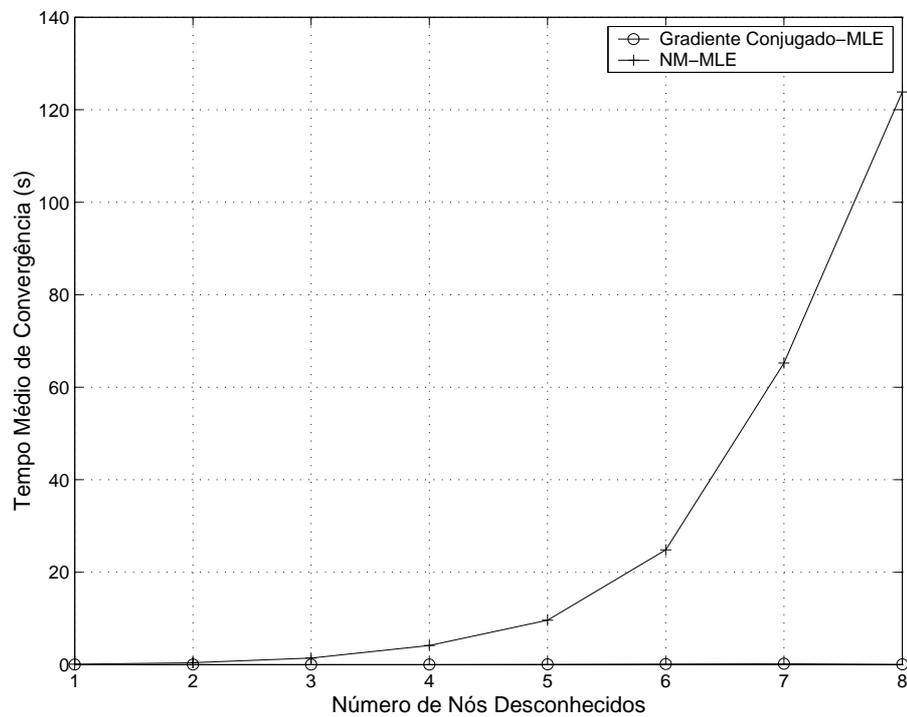


Figura 4.11: Tempo Médio de Convergência para Algoritmos de Localização Conjunta com custo em função do *MLE* na Rede de Referência

resultados obtidos por este. Porém o desempenho em termos de número de iterações e tempo até convergência da localização conjunta deixa a desejar: apesar dos testes terem sido feitos com um máximo de 8 nós desconhecidos, os resultados revelam que o algoritmo baseado em Nelder-Mead não tem escalabilidade na Localização Conjunta. Assim, efetuou-se mudanças no algoritmo para estudo da localização iterativa, seção 4.2, a fim de verificar se um menor tempo de convergência é observado.

4.2 Localização Iterativa

Como o tempo de convergência aumenta de forma bastante acentuada com o aumento do número de nós desconhecidos na localização conjunta, avaliou-se o algoritmo numa versão iterativa na expectativa de uma melhor escalabilidade. A localização iterativa é sujeita à propagação do erro de localização. Isto se deve à influência do erro de localização do primeiro nó, visto que passa a ser referência, na localização do segundo e assim sucessivamente.

A localização iterativa tem o mesmo princípio da Multilateração Iterativa descrita no *Capítulo 2*. Este trabalho utiliza o termo genérico localização iterativa já que a função custo usada nem sempre é o da multilateração. O algoritmo pode então ser resumido nos seguintes passos:

1. parâmetros de entrada: $\bar{P}, P_0, \eta, \omega$
2. para o primeiro nó desconhecido: $i = 1, p = 0$
 - (a) definir um vetor θ_0 , como ponto de partida. $\theta_i = \theta_0$
 - (b) usar θ_i e calcular um vetor de distância até as referências, $\tilde{D}_i = [\tilde{d}_{i1}, \tilde{d}_{i2}, \dots, \tilde{d}_{i(m+p)}]$
 - i. Se função custo for f_{cMSE} calcular o vetor $\tilde{P}_i = [\tilde{P}_{i1}, \tilde{P}_{i2}, \dots, \tilde{P}_{i(m+p)}]$
 - ii. Se função custo for f_{cML} ou f_{cMLE} calcular $\hat{D}_i = [\hat{d}_{i1}, \hat{d}_{i2}, \dots, \hat{d}_{i(m+p)}]$
 - iii. Calcular o valor da função custo para $\theta_i = [x_i, y_i]$
 - iv. Enquanto não houver convergência, refinar $\theta_i = [x_i, y_i]$ e voltar ao passo (i)
 - (c) Nó i encontrado. Fazer $i = i + 1, p = p + 1$
 - (d) Enquanto houverem nós a localizar, ($p < n$), voltar ao passo (a)
3. parâmetro de saída: $\theta = [x_1, y_1, x_2, y_2, \dots, x_n, y_n]$

Onde os parâmetros de entrada $\bar{P}, P_0, \eta, \omega$, são respectivamente a matriz de potências medidas, potência à distância $d_0 = 1m$ da referência, expoente de perda de percurso e coordenadas dos nós de referência. O parâmetro de saída θ é o vetor de coordenadas otimizadas dos n nós desconhecidos. i é o índice do i -ésimo nó desconhecido, cujas coordenadas são dadas por θ_i e p é o número de nós desconhecidos já localizados. Na localização iterativa pode-se usar o algoritmo tanto de forma centralizada quanto descentralizada. Para efeitos de simulação em *Matlab*[®] utilizou-se a localização centralizada onde os nós desconhecidos são localizados em sequência. Como apenas um nó é localizado de cada vez, deve-se usar a variante do algoritmo que tenha o menor erro médio e o melhor desempenho em termos de tempo de convergência para a localização de um único nó: observando-se a tabela 4.2 nota-se que a variante *NM-Mínimos Quadrados* e *NM-MLE* apresentam o mesmo erro médio de localização, e tem

desempenho bastante semelhantes conforme avaliados nos casos de teste da seção 4.1.3. São, portanto, equivalentes e não há a necessidade de se empregar ambas as variantes na implementação da localização iterativa. Optou-se pelo uso da variante *NM-Mínimos Quadrados* por ter um tempo de convergência menor para redes densas. A variante *NM-Multilateração* tem um erro médio aproximadamente $0.5m$ maior que a *NM-Mínimos Quadrados*, porém, isto não justifica seu descarte de emprego numa localização iterativa visto que o ganho que ela tem com proximidade entre nós vizinhos é alta conforme mostra o gráfico da figura 4.7, e por isso também foi utilizada esta variante de custo na localização iterativa.

A gráfico da figura 4.12 compara o desempenho das duas variantes de custo, *NM-Mínimos Quadrados* e *NM-Multilateração* para o cenário de referência, num algoritmo iterativo com base em Nelder-Mead. Apesar da variante de custo *NM-Multilateração* ter um desempenho melhor do que a variante *NM-Mínimos Quadrados* na localização conjunta, isto não se confirma para a localização iterativa. A variante *NM-Mínimos Quadrados* tem um desempenho em torno de $0.75m$ melhor que a variante *NM-Multilateração* para $n = 30$.

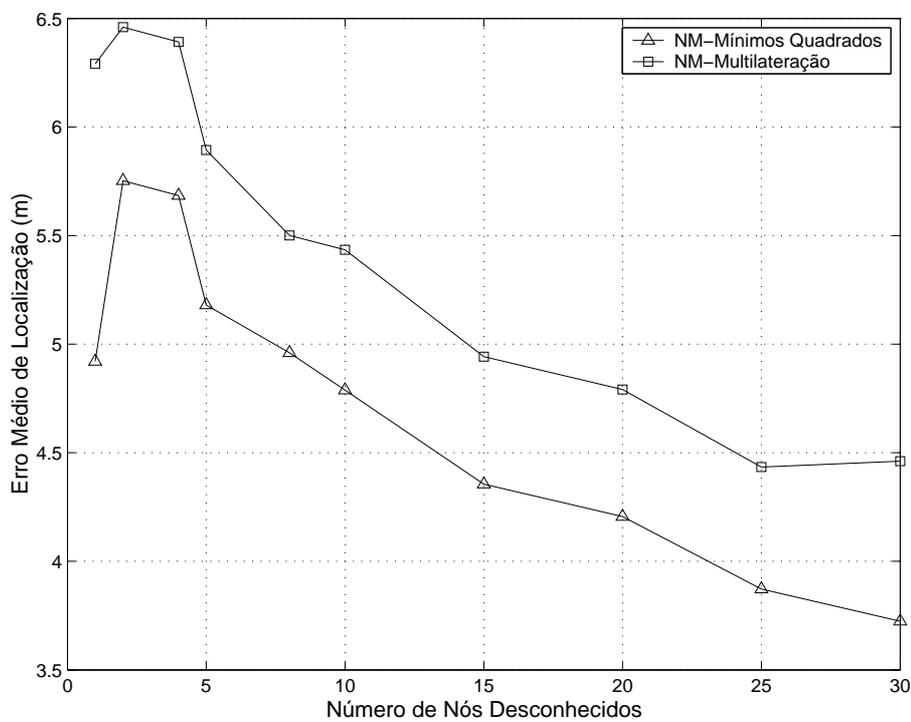


Figura 4.12: Erro Médio de Localização Iterativa no Cenário de Referência para *NM-Mínimos Quadrados* e *NM-Multilateração*

Um investimento na melhoria da qualidade da localização da variante *NM-Multilateração* se justifica apenas se esta tiver um tempo de convergência menor do que a da variante *NM-Mínimos Quadrados*. As figuras 4.13 e 4.14 comparam respectivamente o desempenho em número de iterações e o tempo despendido até a convergência para ambas as variantes em análise. O gráfico 4.13 mostra que o número de iterações aumenta linearmente com o número de nós para ambas as variantes, conforme esperado. As curvas iniciam próximas, mas a variante *NM-Multilateração* apresenta um ângulo de subida mais acentuado. O gráfico da figura 4.14 mostra que apesar do aumento do número de iterações ser linear, o tempo despendido au-

menta em progressão geométrica com o número de nós e a variante *NM-Multilateração* apresentou um aumento mais acentuado do tempo de convergência.

Os estudos de localização iterativa realizados a seguir utilizam a variante *NM-Mínimos Quadrados* por apresentar um melhor desempenho em termos de erro médio de localização, número de iterações e tempo até convergência. Quando o texto mencionar o algoritmo baseado em Nelder-Mead sem referenciar a variante de custo, é a esta que se está referindo.

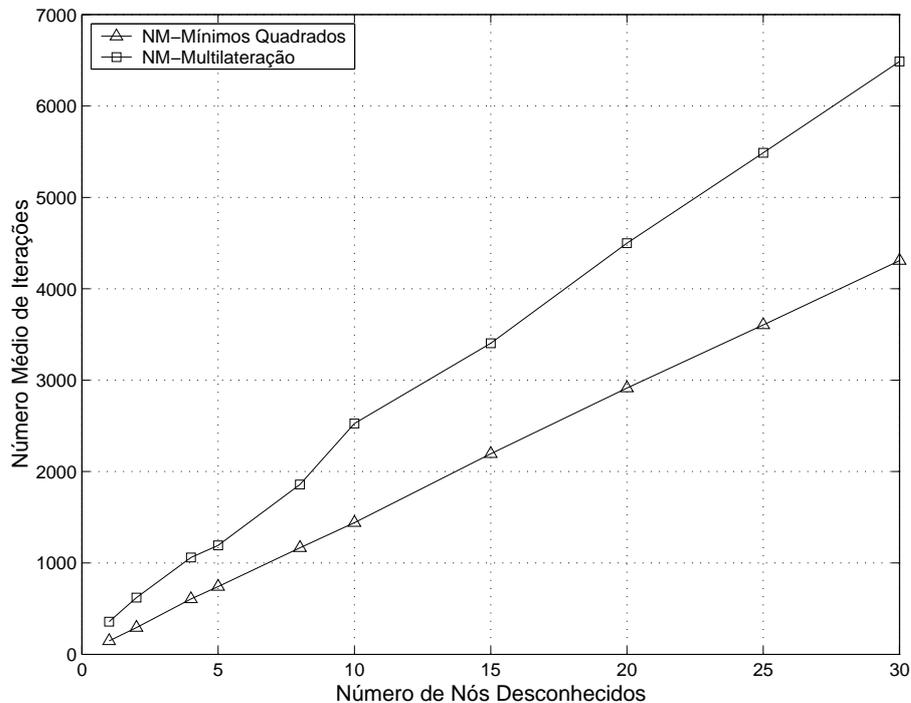


Figura 4.13: Número Médio de Iterações no Cenário de Referência para *NM-Mínimos Quadrados* e *NM-Multilateração* em Localização Iterativa

A figura 4.15 compara o desempenho da *Localização Iterativa* com o da *Localização Conjunta* para $n = 8$ em função de K_a . A figura 4.16 compara o desempenho entre as localizações *Iterativa* e *Conjunta* no cenário de referência. Os gráficos mostram que apesar do desempenho do algoritmo iterativo melhorar à medida que n aumenta, o ganho não é tão expressivo como quanto para a localização conjunta. A razão disto é a propagação do erro de localização.

Para o cenário de referência foi comparado o desempenho das versões de *Localização Conjunta* e *Iterativa* em termos de número de iterações e tempo médio de convergência, vide gráfico das figuras 4.17 e 4.18 respectivamente. Enquanto o desempenho em termos de tempo médio e número de iterações até a convergência da localização conjunta degrada em progressão geométrica, o desempenho da localização iterativa oscila em patamares muito menores.

Verifica-se que o tempo médio de convergência da localização iterativa é bem menor do que da localização conjunta. Para $n = 9$ a localização iterativa converge em média em 2.777s enquanto a localização conjunta leva 154.2896s para convergir. Porém o erro médio da localização iterativa é maior que o da localização conjunta. Para o mesmo valor de n o erro médio da localização iterativa é de 4.6723m enquanto que para a localização conjunta é de apenas 3.0105m. Este valor pode ser melhorado se a localização dos nós ocorrer numa sequência tal

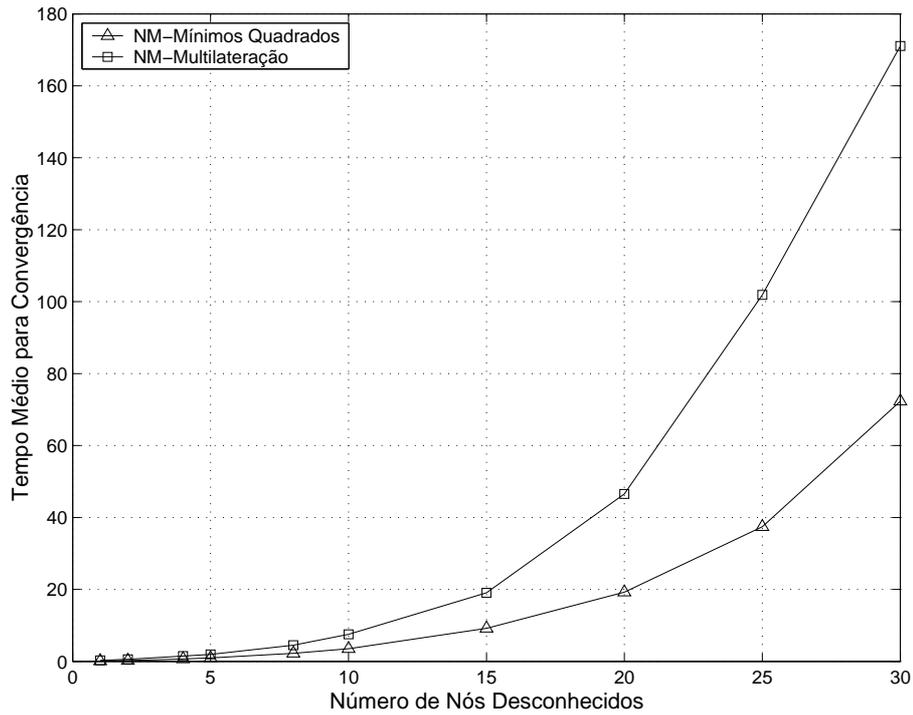


Figura 4.14: Tempo Médio de Convergência no Cenário de Referência para *NM-Mínimos Quadrados* e *NM-Multilateração* em Localização Iterativa

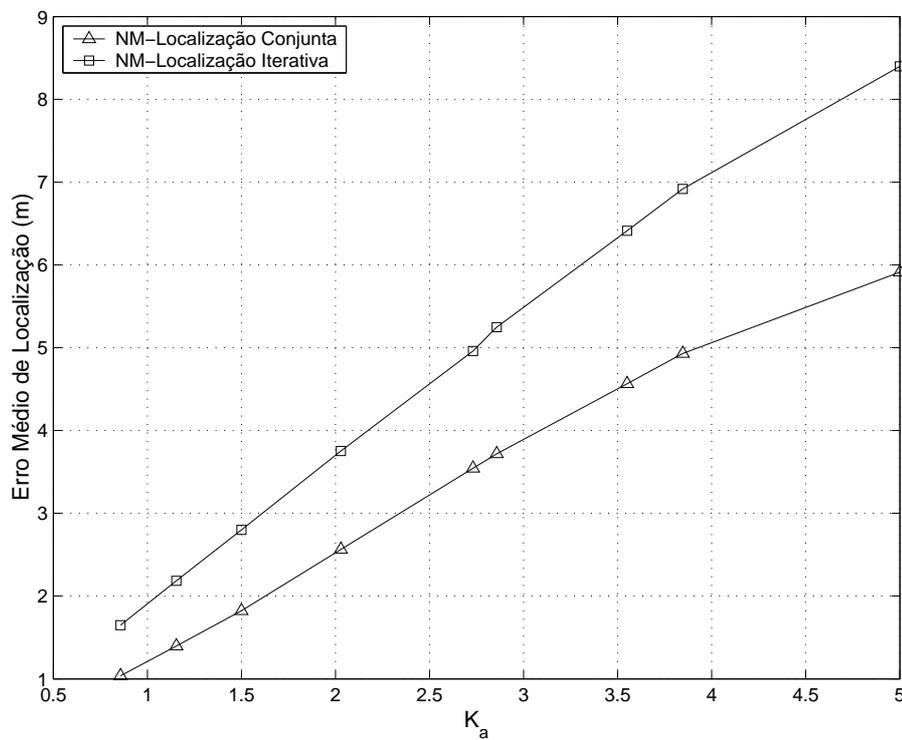


Figura 4.15: Erro Médio de Localização na Rede de Referência com $n = 8$ em Função de K_a para *NM-Mínimos Quadrados* e *NM-Multilateração* em Localização Iterativa

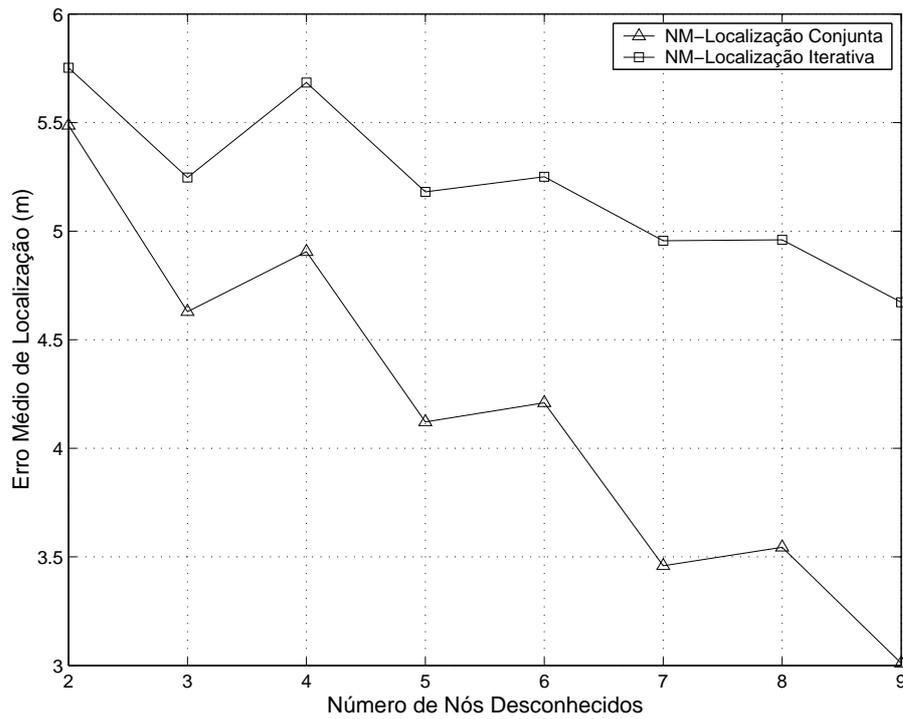


Figura 4.16: Erro Médio de Localização no Cenário de Referência para a Variante *NM-Mínimos Quadrados* em Algoritmo do tipo Conjunto e Iterativo

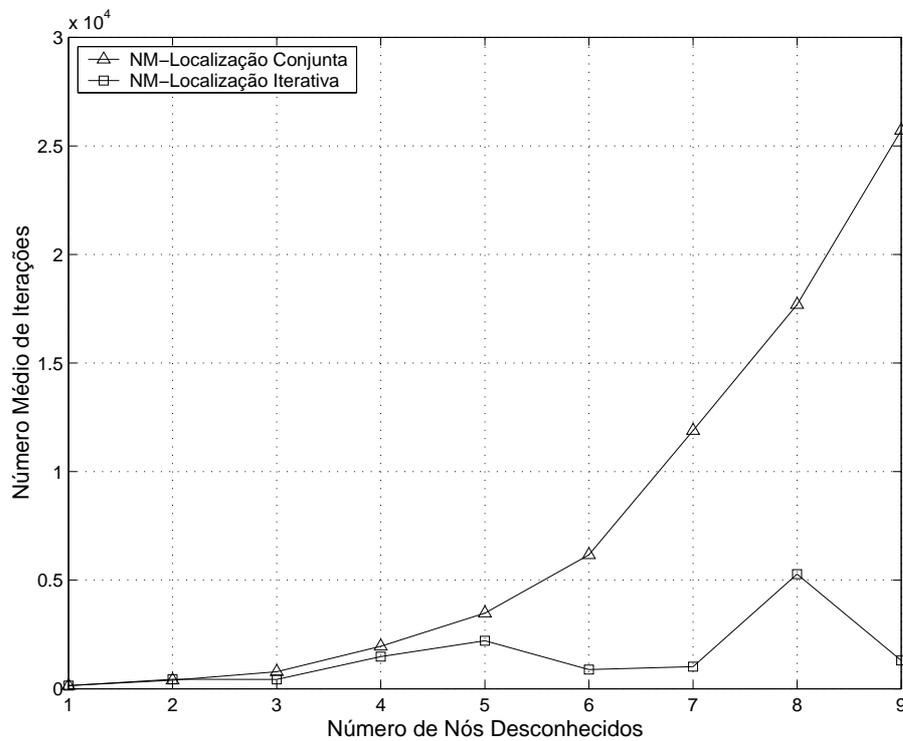


Figura 4.17: Número Médio de Iterações no Cenário de Referência para a Variante *NM-Mínimos Quadrados* em Algoritmo do tipo Conjunto e Iterativo

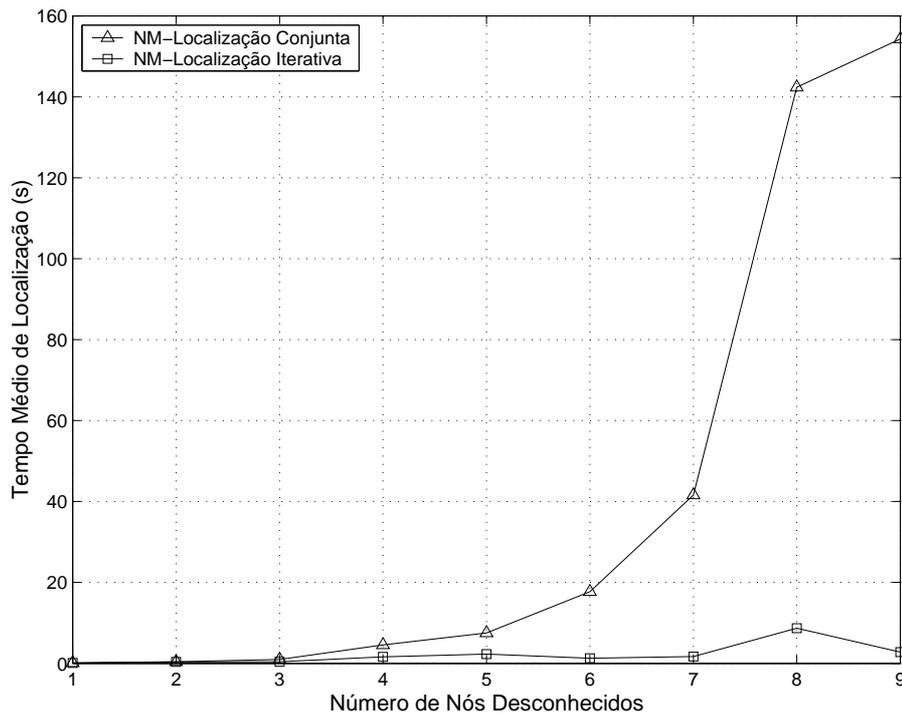


Figura 4.18: Tempo Médio de Convergência no Cenário de Referência para a Variante *NM-Mínimos Quadrados* em Algoritmo do tipo Conjunto e Iterativo

que produza a menor propagação de erro possível. A seção 4.2.1 apresenta a influência das posições relativas dos nós e da sequência de localização no erro médio de localização.

4.2.1 Influência da Sequência de localização

Avaliou-se a influência que a posição do nó exerce sobre o erro médio de sua localização. Para isto mediu-se o erro médio de localização para um nó desconhecido fixo em posições pré-determinadas conforme mostra a figura 4.19. Inicialmente o nó foi posicionado no centro da rede, e uma vez tendo sido localizado ele foi posicionado $1m$ acima e assim sucessivamente até $7m$ de distância do centro. Em seguida repetiu-se o mesmo procedimento, porém com o nó movendo-se em diagonal a partir do centro em passos de $1m$ também.

A figura 4.20 apresenta duas curvas de evolução do erro médio à medida que o nó se move a partir do centro. Uma curva corresponde ao movimento vertical e a outra curva ao movimento em diagonal do nó. Ao lado de cada curva é indicada a posição do nó associada ao valor de erro médio. Quando o nó se move a partir do centro da rede na vertical o seu erro médio de localização tem uma pequena variação se comparado quando o nó se move na diagonal em direção à um nó de referência. Isto está bastante de acordo com o fato de que quanto mais próximos dois nós estiverem, mais precisa é a estimativa da distância relativa entre eles.

Visto que a posição do nó é fator determinante do erro médio de sua localização, é razoável assumir que a sequência de localização pode influenciar no erro médio de localização dos nós. Definiu-se uma métrica que atribui pesos relacionados com a posição do nó na rede. As únicas informações de que o algoritmo dispõe são as medidas de *RSSI* e a matriz \hat{D} resultante. Portanto a métrica considera apenas estas informações. Considerando-se que deve-se localizar

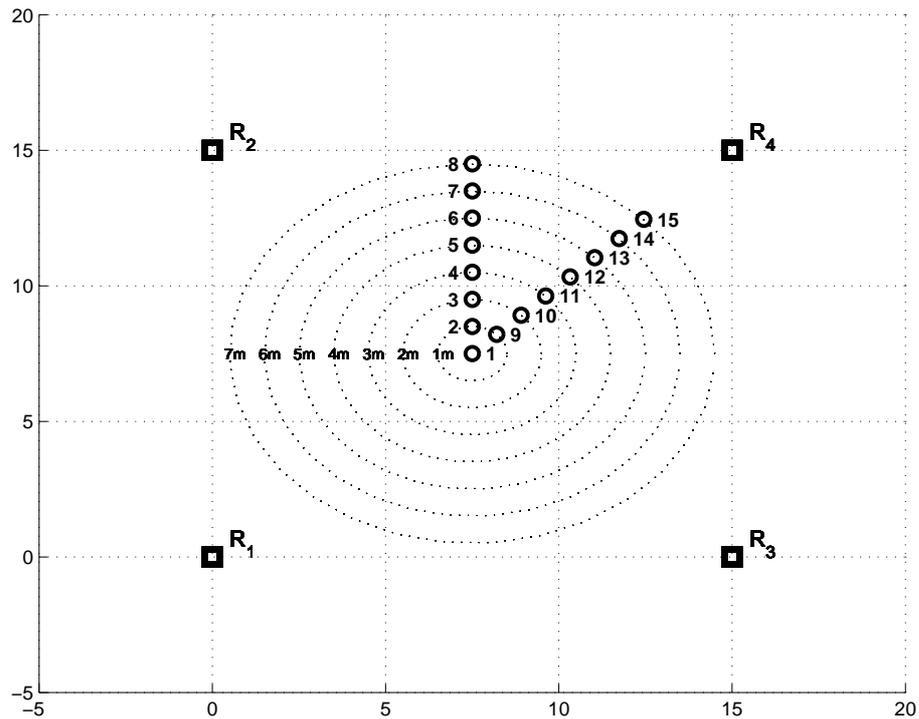


Figura 4.19: Movimento Vertical e Diagonal a partir do centro da Rede de Referência

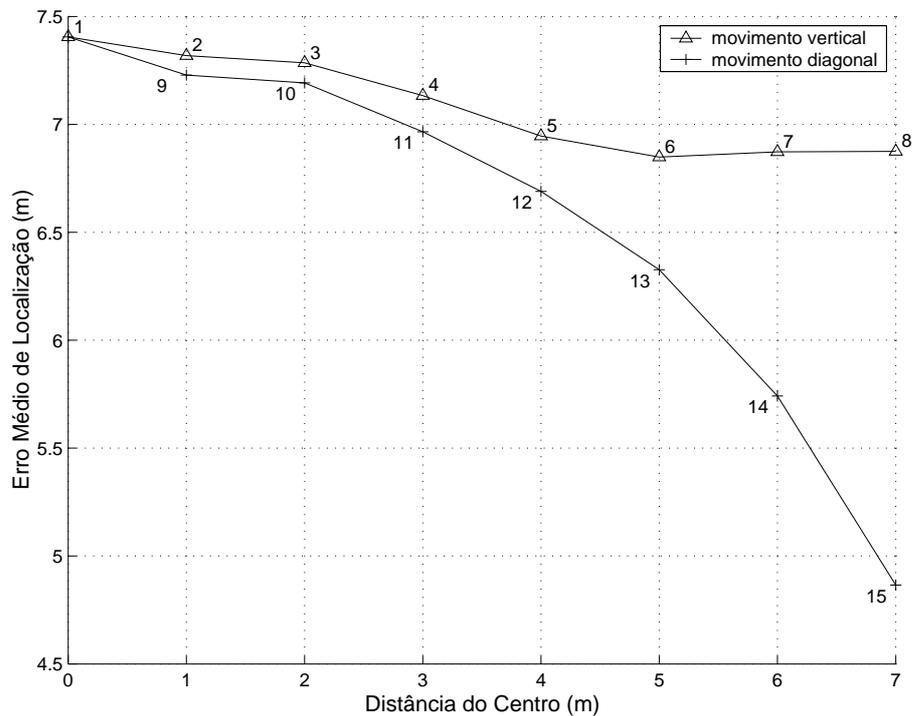


Figura 4.20: Erro Médio de Localização para o nó se afastando do centro da rede no Cenário de Referência

primeiro aquele nó mais próximo do maior número de referências [Savvides et al., 2001], elaborou-se uma métrica que procura quantificar esta proximidade. Adotou-se como ponto

de partida a somatória das estimativas de distância realizadas por um dado nó desconhecido até suas referências. A proximidade de um nó i dos demais nós é chamada de fator de proximidade e é dado por

$$F_{proximidade}(i) = \sum_{j=1}^{j \leq m+p} \widehat{d}_{ij}, \quad (4.2)$$

onde p é o número de nós desconhecidos já localizados. Aplicando-se $F_{proximidade}$ ao nó da figura 4.19 obtém-se os valores apresentados na tabela 4.3. O cálculo foi feito sem considerar o sombreamento, de modo que possa ser estabelecida uma relação visual entre o fator de proximidade e a posição do nó. A primeira coluna, *Sequência Alvo*, apresenta as posições em ordem crescente de erro médio de localização. Os valores de $F_{proximidade}$ estão tabelados em ordem decrescente na segunda coluna e as posições associadas são apresentadas à direita na coluna *Sequência*. A métrica de seleção da sequência de localização deve se aproximar tanto quanto possível da *Sequência Alvo* a fim de se localizar primeiramente aqueles nós em posições associadas à menores valores de erro médio de localização. Comparando-se a sequência alvo com a sequência obtida por $F_{proximidade}$, verifica-se que este objetivo não foi atingido. O nó na posição 8 tem um erro de localização aproximadamente 2m maior do que para o nó na posição 15. O nó da figura 4.19 se aproxima de algumas referências e se afasta de outras à medida que se movimenta na vertical ou diagonal. Mas quando ele está o mais próximo de uma única referência é que o erro médio de sua localização foi o menor. Baseando-se nesta observação, elaborou-se uma outra métrica, chamada de relação de proximidade e dada pela razão entre $F_{proximidade}$ e a menor distância medida pelo nó desconhecido até qualquer nó de referência,

$$R_{proximidade}(i) = \frac{F_{proximidade}(i) + \gamma}{\min(\widehat{d}_{ij})}, \forall j \in \{\omega \cup \theta^*\} \quad (4.3)$$

$$\gamma = (n - p) * \max(\widehat{D}), \quad (4.4)$$

onde θ^* é o vetor de coordenadas dos p nós desconhecidos já localizados e γ é uma constante atribuída como a distância entre dois nós desconhecidos ainda não localizados. É esta constante que torna indiferente a proximidade do nó i que se quer localizar de um outro nó desconhecido. Através de experimentos se comprovou que ao invés de se desprezar a estimativa de distância entre dois nós desconhecidos quaisquer, substituí-la por um valor constante γ e igual à maior estimativa realizada produz resultados de sequenciamento melhores. Os valores de $R_{proximidade}$ para o nó nas diferentes posições da figura 4.19 estão tabeladas em 4.3. Comparando-se os valores da $R_{proximidade}$ com a posição do nó observa-se que este tem um valor maior para as posições que se relacionam com um menor erro médio de localização na maioria dos casos. A ordenação desejada não é satisfeita para o nó nas posições: 11, 10 e 3. Nestes casos porém, o erro de localização dos mesmos é apenas pouco maior do que os nós que tem uma relação-proximidade menor.

O algoritmo para localizar individualmente os nós iniciando pelo mais próximo de qualquer nó de referência é composto dos seguintes passos:

1. parâmetros de entrada: \bar{P} , ω, P_0, η
2. determinar a $R_{proximidade}$ de cada nó desconhecido
3. $p = 0$

Tabela 4.3: Fatores e Relações de Proximidade para o Nó da figura 4.19

Sequência alvo	$F_{proximidade}$	Sequência	$R_{proximidade}$	Sequência
15	47.6830	8	81.2738	15
14	46.6297	15	63.4042	14
13	46.1839	7	51.9312	13
12	45.5853	14	43.9526	12
8	44.9661	6	39.1365	8
7	44.6653	13	38.2657	7
6	44.0120	5	38.0920	11
5	43.8848	12	36.8669	6
11	43.2998	4	35.1000	5
4	43.2586	11	33.6129	10
10	42.8085	3	33.1326	4
9	42.8002	10	31.1058	3
3	42.5210	2	30.0848	9
2	42.5205	9	29.1206	2
1	42.4264	1	27.2395	1

4. $i =$ nó desconhecido que tiver a maior $R_{proximidade}$. $i = arg \max[R_{proximidade}]$
 - (a) definir um vetor θ_i como ponto de partida. $\theta_i = \theta_{0_i}$
 - (b) usar θ_i e calcular um vetor de distância até as referências, $\tilde{D}_i = [\tilde{d}_{i1}, \tilde{d}_{i2}, \dots, \tilde{d}_{i(m+p)}]$
 - i. calcular o vetor $\tilde{P}_i = [\tilde{P}_{i1}, \tilde{P}_{i2}, \dots, \tilde{P}_{i(m+p)}]$
 - ii. Calcular o valor da função custo para $\theta_i = [x_i, y_i]$
 - iii. Enquanto não houver convergência, refinar $\theta_i = [x_i, y_i]$ e voltar ao passo (b)
 - (c) Nó i encontrado, fazer $p = p + 1$
 - (d) Enquanto houverem nós a localizar, ($p < n$), voltar ao passo (2)
5. parâmetro de saída: $\theta = [x_1, y_1, x_2, y_2, \dots, x_n, y_n]$

Onde \bar{P} é a matriz de potências medidas, P_0 é a potência à distância de $d_0 = 1m$ da referência, η é o desvio padrão do sombreamento, ω são as coordenadas dos nós de referência. O parâmetro de saída θ é o vetor de coordenadas otimizadas dos n nós desconhecidos. i é o índice do i -ésimo nó desconhecido, cujas coordenadas são dadas por θ_i e p é o número de nós desconhecidos já localizados.

Aplicando-se a métrica relação-proximidade para sequenciamento dos nós desconhecidos à rede da figura 4.21 os nós são localizados na seguinte sequência: 9, 3, 7, 12, 13, 15, 11, 8, 14, 1, 4, 5, 2, 6, 10. Cada nó localizado que se torna referência passa a influir na métrica dos próximos nós. Nós próximos aos já localizados são localizados antes de nós mais distantes de qualquer referência. A localização inicia pelos nós próximos aos m nós de referência, portanto nos cantos da rede, nós próximos a estes são os próximos a serem localizados, e por último são localizados aqueles mais distantes de qualquer nó.

A figura 4.22 apresenta a comparação da curva obtida pelo algoritmo de localização iterativa com e sem o sequenciamento. Até $n = 15$ a localização sequenciada apresenta uma

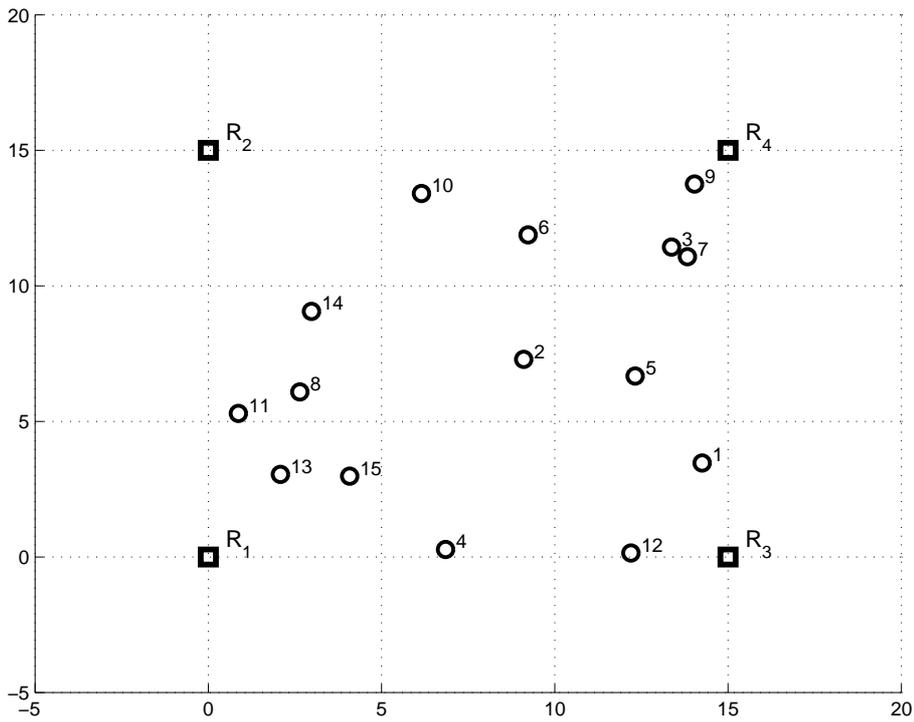


Figura 4.21: Rede de Referência $n = 15$ e $m = 4$

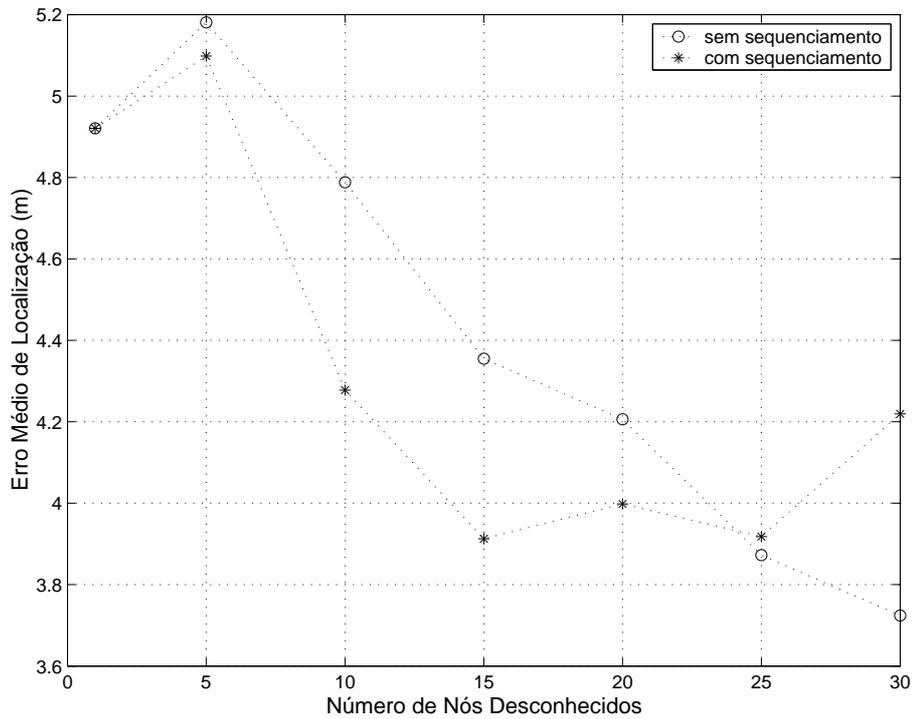


Figura 4.22: Comparação entre Localização Iterativa com e sem Sequenciamento dos nós no Cenário de Referência

queda bem mais acentuada do erro médio de localização, a partir daí até $n = 25$ seu desempenho é ainda melhor que a localização sem o sequenciamento. E para $n = 30$ a localização não

sequenciada apresenta um desempenho melhor. À medida que a densidade de nós aumenta, o sequenciamento perde influência no erro médio de localização. Isto ocorre porque em distâncias mais curtas as medidas de *RSSI* são menos sujeita aos erros de sombreamento.

4.2.2 Utilização de Coeficientes de Ponderação

A figura 4.23 apresenta uma série de curvas de desempenho utilizando a variante *NM-Mínimos Quadrados* num algoritmo iterativo. As curvas são de diferentes tipos de ponderações realizadas tanto sobre o sequenciamento dos nós quanto sobre a função custo. Cada curva recebeu uma identificação diferente, conforme uma ponderação tenha sido usada ou não e dependendo da ponderação aplicada:

1. NM-LI: Localização Iterativa sem ponderação;
2. NM-LIS: Localização Iterativa com Sequenciamento de localização sem ponderação;
3. NM-LISP: Localização Iterativa com Sequenciamento de Localização Ponderado;
4. NM-LISPCP: Localização Iterativa com Sequenciamento de localização Ponderado e Custo Ponderado.

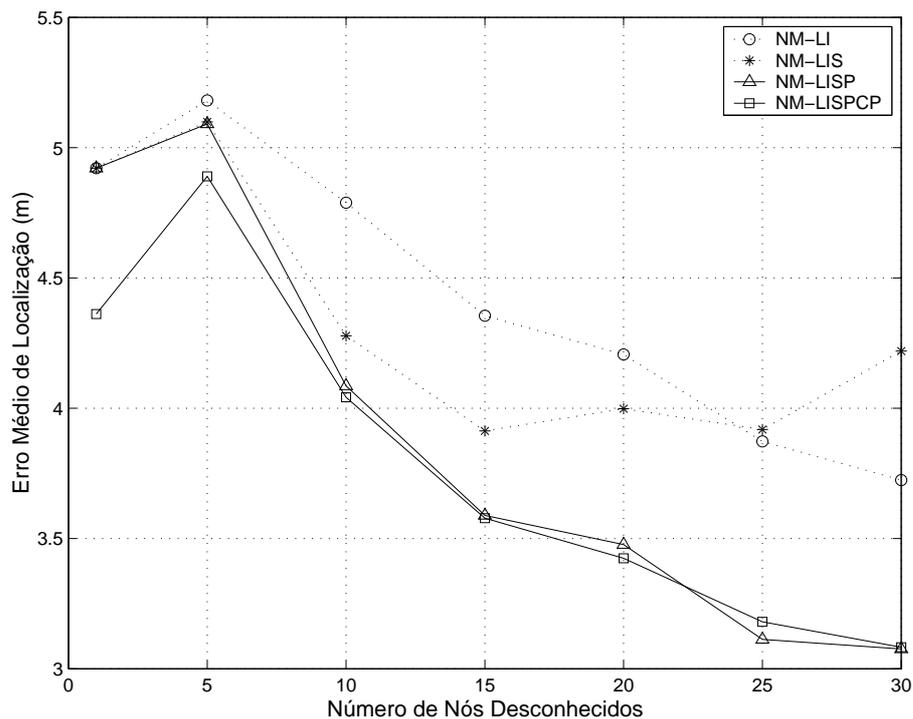


Figura 4.23: Comparação dos Diferentes Tipos de Ponderação para Localização Iterativa no Cenário de Referência

As curvas *NM-LI* e *NM-LIS* são aquelas obtidas com o algoritmo iterativo e iterativo com sequenciamento conforme apresentado nas seções 4.2 e 4.2.1. Foram criados 2 coeficientes de ponderação: α e β . α atua sobre a sequência de localização dos nós e β atua sobre o custo da função a ser minimizada.

A ponderação no sequenciamento através do coeficiente α tem por objetivo localizar antes aqueles nós desconhecidos que estejam mais próximos do maior número de nós de referência. Assim, se houverem dois nós desconhecidos com o mesmo valor de $R_{proximidade}$ aquele que tem um dos m nós de referência mais próximo é localizado antes. É uma forma de se valorizar a confiabilidade da localização dos m nós de referência. A relação proximidade ponderada é dada pela equação

$$Rp_{proximidade}(i) = \frac{F_{proximidade}(i) \cdot \alpha + \gamma}{\min(\widehat{d}_{ij})}, \forall j \in \{\omega \cup \theta^*\}, \quad (4.5)$$

onde, $\gamma = (n - p) * \max(\widehat{D})$, e

$$\alpha = \begin{cases} 1.15, & \text{para } 1 \leq j \leq m \\ 1.00, & \text{para } j > m. \end{cases} \quad (4.6)$$

O desempenho para $\alpha = 1.15$ é descrito pela curva *NM-LISP*. Outros valores foram testados para α : 1.50 e 3.00. Não houve diferença de desempenho para testes realizados sob as mesmas condições com até 30 nós desconhecidos.

A ponderação da função custo pelo coeficiente β se baseia no fato do erro nas estimativas de distância relativa, \widehat{d}_{ij} , ser maior quanto maior for a distância entre i e j . Através da utilização deste coeficiente se pondera a contribuição de cada parcela $(\bar{P}_{ij} - \tilde{P}_{ij})^2$ em função de \widehat{d}_{ij} , conforme equação

$$f_{c_{ponderado}}(i) = \sum_{j=1}^{m+p} \beta(ij) \cdot (\bar{P}_{ij} - \tilde{P}_{ij})^2 \quad (4.7)$$

As diferentes implementações foram elaboradas de maneira evolutiva de modo que à medida que os fatores sejam aplicados melhores desempenhos sejam observados. A implementação *NM-LI* não tem sequenciamento nem ponderação. A implementação *NM-LIS* tem $\alpha = 1$, $\beta = 1$. A implementação *NM-LISP* tem α conforme equação 4.6 e $\beta = 1$. *NM-LISPCP* usa α conforme equação 4.6 e β conforme equação 4.8. A ponderação β usa da a razão entre $F_{proximidade}(i)$, equação 4.2, e a distância estimada desde o nó desconhecido i e cada outro nó j , de referência, d_{ij} como uma forma de atribuir pesos maiores quanto mais próximo o nó desconhecido estiver de sua referência. As curvas da figura 4.23 mostram que os fatores de ponderação melhoram o desempenho do algoritmo em relação à implementação iterativa sequenciada *NM-LIS*. O uso do coeficiente α no sequenciamento provoca uma melhora significativa do desempenho, enquanto que o coeficiente β produz uma melhora principalmente para $n < 10$ onde as distâncias relativas entre os nós são maiores e a ponderação do custo em função da distância tem mais efeito. Dois cálculos diferentes para β foram empregados:

$$\beta_1(ij) = \rho \cdot \log\left(\frac{F_{proximidade}(i)}{\widehat{d}_{ij}}\right) \quad (4.8)$$

e

$$\beta_2(ij) = \rho \cdot \frac{F_{proximidade}(i)}{\widehat{d}_{ij}}, \quad (4.9)$$

onde ρ é uma constante que privilegia a proximidade de um nó desconhecido até qualquer referência e cujo valor ótimo, obtido por meio de testes, é dado por

$$\rho = \begin{cases} 2.7, & \text{para } 1 \leq j \leq m \\ 0.0, & \text{para } j > m. \end{cases} \quad (4.10)$$

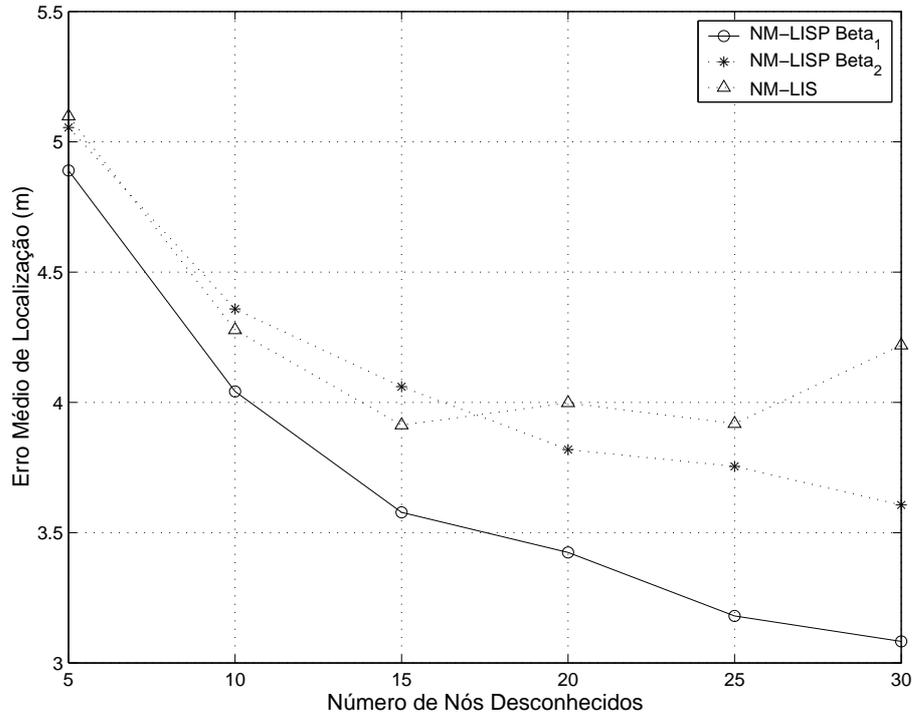


Figura 4.24: Comparação do Erro Médio de Localização Iterativa das Ponderações β_1 e β_2 no Cenário de Referência

A comparação entre o desempenho obtido pelo algoritmo com as ponderações β do custo é mostrado no gráfico da figura 4.24. Enquanto que o desempenho da ponderação β_1 é sempre melhor que o desempenho sem ponderação, o desempenho da ponderação β_2 só é melhor que o algoritmo sem ponderação para $n > 20$. O melhor desempenho de β_1 pode ser explicado pelo fato de capturar o decaimento logarítmico da potência com a distância. Isto é relevante porque o sombreamento χ , maior fonte de erro, é invariante com a mesma.

4.3 Conclusão

Empregou-se o método numérico de otimização não-linear de Nelder-Mead como base para elaboração de diferentes algoritmos de localização. Foram realizados testes com diferentes funções custo, das quais se destacam o custo usando Multilateração em algoritmo de localização conjunta e o custo por Mínimos Quadrados num algoritmo de localização iterativa. No algoritmo de localização iterativa foram usados coeficientes de ponderação tanto para sequenciamento da localização (α), quanto para a função custo (β). Avaliou-se o desempenho destes algoritmos em diferentes cenários de rede obtidos pela variação da área da rede, do desvio padrão das medidas de RSSI, pela variação do expoente de perda de percurso e pelo

número de nós desconhecidos. O coeficiente α demonstrou que melhora a localização em até 19% numa rede de 15m X 15m num cenário de chão de fábrica. Normalmente as redes têm grande quantidade de nós e tendem a crescer continuamente. Portanto a escalabilidade é fator decisivo para a escolha da variante a ser utilizada. No processamento centralizado a localização conjunta do algoritmo *Gradiente Conjugado-MLE* requer um tempo menor até a convergência, porém a variante *NM-Mínimos Quadrados Iterativo* pode ser também implementada por processamento distribuído. Neste caso o tempo de convergência para $n > 1$ deixa de ser relevante, pois cada nó desconhecido é responsável por processar sua própria localização. Portanto a variante *NM-Mínimos Quadrados Iterativo - LISPCP* (com sequenciamento dos nós e ponderação do custo) é a variante mais indicada para uso em um algoritmo de localização com base no método numérico não-linear de Nelder-Mead. O capítulo 5 define o algoritmo proposto e apresenta comparações de desempenho com o algoritmo de *Gradiente Conjugado* tanto em cenários simulados quanto numa aplicação prática de localização numa rede *ad-hoc*.

Capítulo 5

Algoritmo Proposto

Este capítulo apresenta o algoritmo proposto em seu formato final e explora comparações de desempenho com o algoritmo de *Gradiente Conjugado* em diferentes cenários de rede. Um caso de estudo prático é realizado numa rede *ad-hoc* IEEE 802.11b com ambos os algoritmos. Será demonstrado que o algoritmo proposto tem desempenho equivalente ao algoritmo de *Gradiente Conjugado* em termos de erro médio de localização para quando a busca inicia em um ponto aleatório da rede. Tendo em vista os resultados obtidos através dos experimentos realizados no capítulo 4, o algoritmo proposto adota a variante *NM-Mínimos Quadrados-LISPCP* (Localização Iterativa com Sequenciamento Ponderado e Custo Ponderado) dentre os algoritmos de localização com base em Nelder-Mead estudados. É a variante que tem o menor tempo médio de convergência e apresenta um erro médio de localização menor que o algoritmo de *Gradiente Conjugado*. O algoritmo proposto é descrito pelos seguintes passos:

1. parâmetros de entrada: $\bar{P}, \omega, P_0, \eta$
2. $p = 0$
3. $i =$ nó desconhecido que tiver a maior $Rp_{proximidade}$:

$$i = \arg \max_{i \in \theta} [Rp_{proximidade}(i)]$$
$$Rp_{proximidade}(i) = \frac{F_{proximidade}(i) \cdot \alpha + \gamma}{\min(\widehat{d}_{ij})}, \forall j \in \omega \cup \theta^*,$$

onde $\gamma = (n - p) * \max(\widehat{D})$ e

$$\alpha = \begin{cases} 1.15, & \text{para } 1 \leq j \leq m \\ 1.00, & \text{para } j > m. \end{cases}$$

- (a) definir o vetor θ_0 , como ponto de partida. $\theta_i = \theta_0$
- (b) Calcular o vetor de distâncias desde o i -ésimo nó desconhecido até as referências, $\widetilde{D}_i = [\widetilde{d}_{i1}, \widetilde{d}_{i2}, \dots, \widetilde{d}_{i(m+p)}]$
 - i. calcular o vetor de potências médias teóricas esperadas em θ_i , $\widetilde{P}_i = [\widetilde{P}_{i1}, \widetilde{P}_{i2}, \dots, \widetilde{P}_{i(m+p)}]$;

ii. calcular o fator de ponderação $\beta(ij) = \rho \cdot \log\left(\frac{F_{proximidade}(i)}{\hat{d}_{ij}}\right)$, onde

$$\rho = \begin{cases} 2.7, & \text{para } 1 \leq j \leq m \\ 0.0, & \text{para } j > m; \end{cases}$$

iii. calcular a função custo $f_{c_{ponderado}}(i) = \sum_{j=1}^{m+p} \beta(ij) \cdot (\bar{P}_{ij} - \tilde{P}_{ij})^2$;

iv. Enquanto não houver convergência, refinar $\theta_i = (x_i, y_i)$ e voltar ao passo (b);

$$\theta_i = \arg \min_{(x_i, y_i)} [f_{c_{ponderado}}(i)]$$

(c) Nó i encontrado, fazer $p = p + 1$

(d) Enquanto houverem nós a localizar, ($p < n$), voltar ao passo (2)

4. parâmetro de saída: $\theta = [x_1, y_1, x_2, y_2, \dots, x_n, y_n]$

A seguir serão apresentadas uma série de simulações para comparação entre o algoritmo proposto, *NM-LISPCP*, e o algoritmo de *Gradiente Conjugado*. O procedimento das simulações realizadas neste capítulo difere do adotado nos capítulos anteriores. A fim de evitar variações abruptas para cima ou para baixo do valor de erro médio, estabeleceu-se a execução de determinado número de experimentos para diferentes arranjos. Chama-se de arranjos diferentes disposições da mesma quantidade de nós. Para a realização de um experimento é necessário que os 4 nós de referência estejam dispostos nos cantos da rede e os n nós desconhecidos tenham sido uniformemente distribuídos pela área da rede. Cada experimento consiste em se gerar os valores médios de *RSSI* entre pares de nós e efetuar a localização de cada um dos n nós desconhecidos. Para se gerar os valores médios de *RSSI* utiliza-se do valor η de perda de percurso e considera-se sombreamento com distribuição normal de média zero e desvio padrão σ . Após a realização de determinado número de experimentos para este arranjo, um novo arranjo é sorteado, redistribuindo-se os n nós desconhecidos e assim sucessivamente. Para cada novo arranjo de nós realiza-se uma nova sequência de experimentos. O valor de erro médio de localização é a média do erro de localização de cada um dos experimentos para os diferentes arranjos. Desse modo obtém-se um erro médio de localização válido para n nós desconhecidos e independente das distâncias relativas entre os nós ou de alguma disposição específica. O comportamento dos algoritmos é avaliado para até 20 nós desconhecidos, acima disto o tempo de convergência exigido pelo algoritmo *NM-LISPCP* torna os testes muito longos. As experiências apresentadas a seguir foram realizadas utilizando-se de 20 arranjos diferentes do mesmo número de nós desconhecidos. Para cada um dos arranjos foram executados 200, 40 e 20 experimentos para 1, 5 e mais de 5 nós desconhecidos respectivamente.

Avaliação do desempenho para diferentes abordagens de início de busca

Nos capítulos anteriores foi usada a própria posição real do nó como ponto de partida para a busca, θ_0 . Duas abordagens mais realistas e práticas são o sorteio do ponto inicial de busca e o método equacionado em [Song, 1994]. A segunda abordagem seleciona o ponto inicial de busca mais próximo da posição real do nó desconhecido. Este ponto inicial é obtido a partir da geometria entre 3 nós de referências. Na figura 5.1 as 3 referências são representadas pelos quadrados e o ponto inicial de busca pelo triângulo.

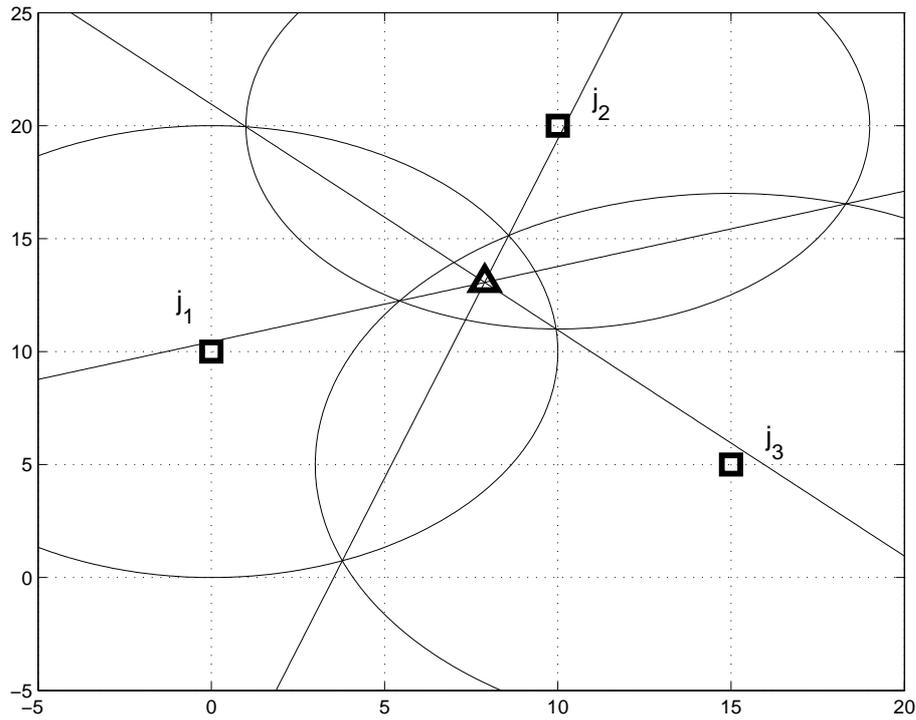


Figura 5.1: Método de seleção do ponto inicial de busca θ_{0_i}

As circunferências ao redor das referências, e sobre as quais se estima estar o nó i , têm raio igual às estimativas de distância e podem ser equacionadas como

$$(x_i - x_{j_1})^2 + (y_i - y_{j_1})^2 = d_{ij_1}^2, \quad (5.1)$$

$$(x_i - x_{j_2})^2 + (y_i - y_{j_2})^2 = d_{ij_2}^2, \quad (5.2)$$

$$(x_i - x_{j_3})^2 + (y_i - y_{j_3})^2 = d_{ij_3}^2. \quad (5.3)$$

Combinando as equações 5.1 e 5.2 acima, chega-se à

$$(x_{j_2} - x_{j_1}) \cdot x_i + (y_{j_2} - y_{j_1}) \cdot y_i = \frac{\widehat{d}_{ij_1}^2 - \widehat{d}_{ij_2}^2 + x_{j_2}^2 - x_{j_1}^2 + y_{j_2}^2 - y_{j_1}^2}{2}, \quad (5.4)$$

que é a equação da reta que passa pelos dois pontos de intersecção das circunferências de centro em j_1 e j_2 . Combinando-se as demais equações de circunferências duas a duas obtém-se outras duas retas:

$$(x_{j_3} - x_{j_1}) \cdot x_i + (y_{j_3} - y_{j_1}) \cdot y_i = \frac{\widehat{d}_{ij_1}^2 - \widehat{d}_{ij_3}^2 + x_{j_3}^2 - x_{j_1}^2 + y_{j_3}^2 - y_{j_1}^2}{2} \quad (5.5)$$

$$(x_{j_3} - x_{j_2}) \cdot x_i + (y_{j_3} - y_{j_2}) \cdot y_i = \frac{\widehat{d}_{ij_2}^2 - \widehat{d}_{ij_3}^2 + x_{j_3}^2 - x_{j_2}^2 + y_{j_3}^2 - y_{j_2}^2}{2}. \quad (5.6)$$

A intersecção das 3 retas define o ponto de início de busca para o i -ésimo nó a ser localizado conforme mostra a figura 5.1. Acaso as 3 referências estejam alinhadas as retas definidas acima são paralelas. Assim, em [Song, 1994], define-se que o ponto de partida é a

média das coordenadas das 3 intersecções entre a reta definida pelas 3 referências com as 3 paralelas.

A figura 5.2 compara o desempenho de ambos os algoritmos para as duas abordagens de seleção de início de busca. Na primeira a busca inicia num ponto uniformemente distribuído na rede e na segunda o ponto inicial é definido conforme procedimento baseado na geometria entre as referências exemplificado acima. A legenda da figura apresenta o nome do algoritmo seguido de A1 para quando a busca inicia pela primeira abordagem e A2 quando inicia pela segunda. A área da rede é de 15 metros·15 metros, $\sigma = 7.1$ e $\eta = 2.6$.

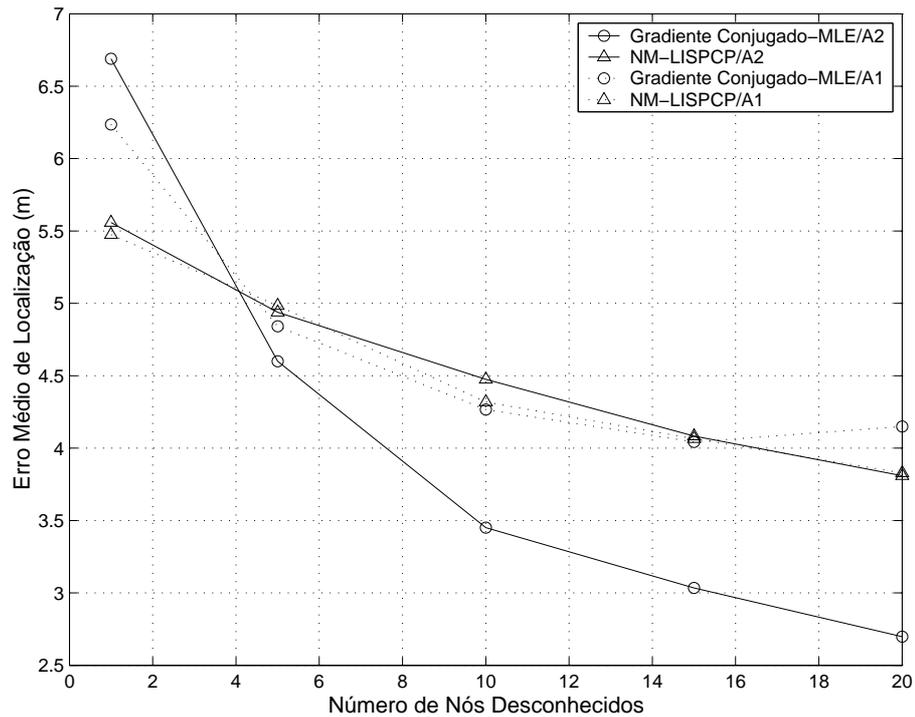


Figura 5.2: Desempenho dos algoritmos proposto e *Gradiente Conjugado* para diferentes abordagens do ponto inicial de busca

Verifica-se que o algoritmo proposto apresenta desempenho equivalente ao algoritmo de gradiente conjugado para quando a busca inicia em ponto uniformemente distribuído. O desempenho do algoritmo NM-LISPCP praticamente não varia ao se mudar a abordagem de seleção do início das buscas. Ao contrário, o algoritmo de *Gradiente Conjugado* apresenta um desempenho bastante melhor se a abordagem do método geométrico for utilizada. A figura 5.3 mostra que o tempo médio de convergência inclusive é menor neste caso. O tempo médio de convergência para o algoritmo proposto mostrou uma ligeira melhora para a segunda abordagem de início de busca, conforme apresenta a figura 5.4.

Efeito das ponderações sobre o algoritmo proposto

A seguir, a influência do coeficiente de ponderação β e do sequenciamento dos nós é explorada. A eficácia dos mesmos é demonstrada através de experimento que compara o desempenho do algoritmo proposto em quatro situações diferentes. A primeira sem sequenciamento nem ponderação. A segunda com sequenciamento e ponderação β apenas. A terceira com se-

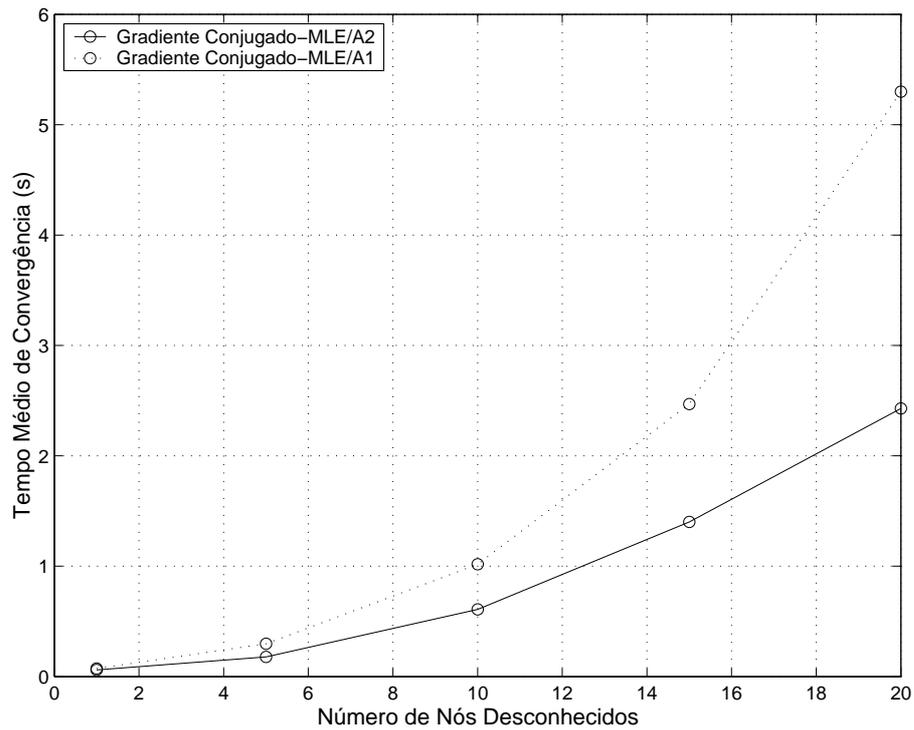


Figura 5.3: Tempo médio de convergência para o algoritmo de *Gradiente Conjugado* para diferentes abordagens do ponto inicial de busca

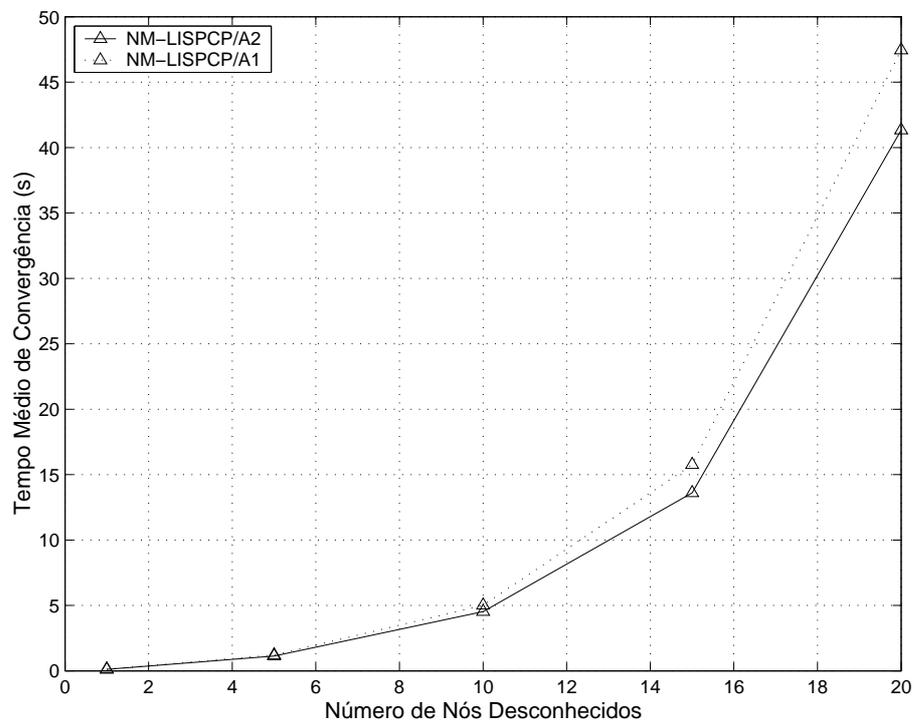


Figura 5.4: Tempo médio de convergência para o algoritmo proposto para diferentes abordagens do ponto inicial de busca

quenciamento e ponderação α apenas. A quarta utiliza sequenciamento e ambas as ponderações (NM-LISPCP). A figura 5.5 apresenta os resultados obtidos. Verifica-se que cada ponderação contribui para um menor valor de erro médio, sendo que o melhor resultado é obtido pela aplicação de ambas.

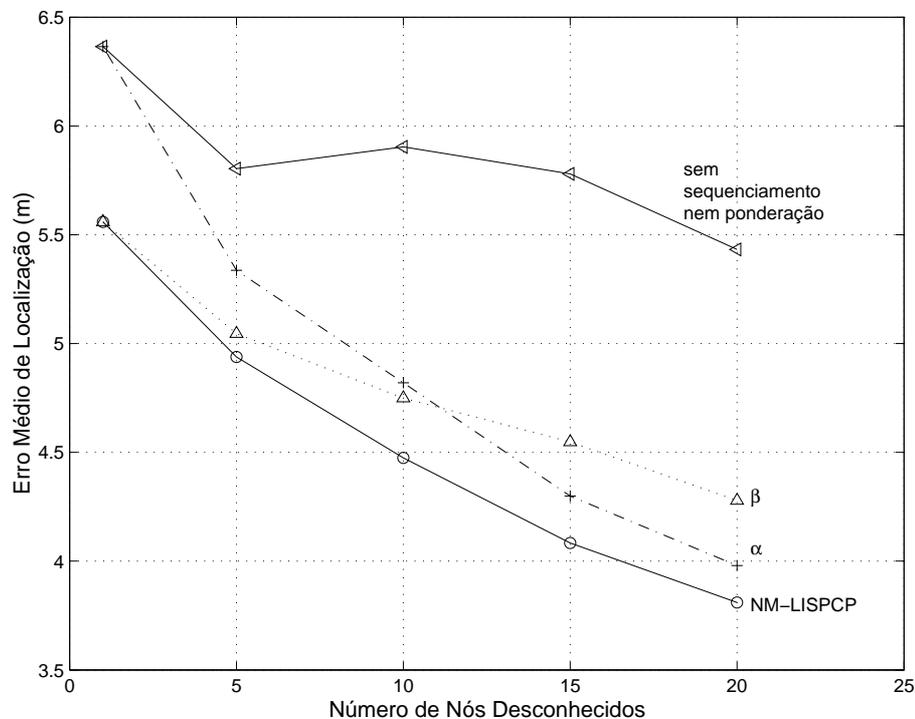


Figura 5.5: Desempenho do algoritmo proposto em função das ponderações aplicadas

Avaliação de desempenho em função de K_a

O próximo experimento compara o desempenho dos algoritmos para a mesma rede de $15m \cdot 15m$, com $\sigma = 7.1$, mas com diferentes valores de η sendo que a busca inicia em um ponto aleatório uniformemente distribuído pela área da rede. Os seguintes valores de η foram utilizados: 1.5, 2.6, 3.5 e 5.5 e que deram origem à diferentes valores de K_a com conseqüente impacto no erro médio de localização conforme observa-se na figura 5.6.

Quanto menor K_a menores valores de erro médio são observados e portanto o erro médio de localização é menor para valores maiores do expoente de perda de percurso, η . Possivelmente os algoritmos convergem com maior precisão se variações no posicionamento dos nós forem acompanhadas de grandes variações nos níveis de potência experimentados, o que ocorre para maiores valores de η . As figuras 5.7 e 5.8 mostram que o desempenho dos algoritmos em termos de tempo médio de convergência pouco varia com o expoente de perda de percurso.

Desempenho em função da área

O desempenho em redes de diferentes áreas, mas com os mesmos valores de $\sigma = 7.1$ e $\eta = 2.6$ é comparado na figura 5.9. O ponto inicial de busca é selecionado por sorteio.

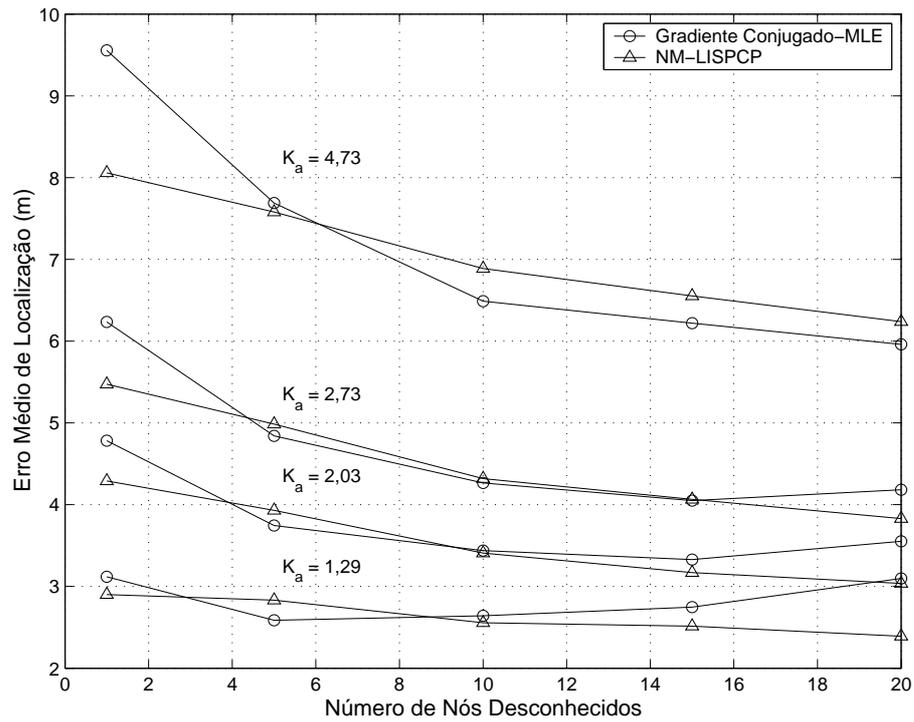


Figura 5.6: Desempenho dos algoritmos de Gradiente Conjugado e proposto para diferentes valores de K_a

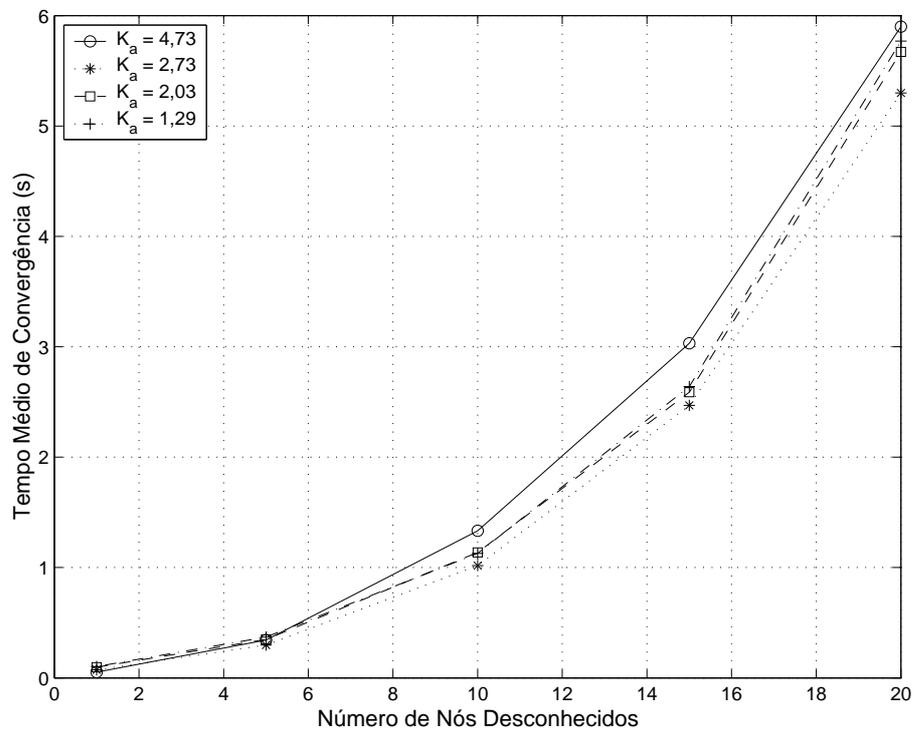


Figura 5.7: Tempo médio de convergência em função de K_a para *Gradiente Conjugado*

Observa-se que ambos os algoritmos têm desempenho bastante próximos. Constatam-se também o efeito multiplicativo do erro de localização para medidas de *RSSI*, os erros médios

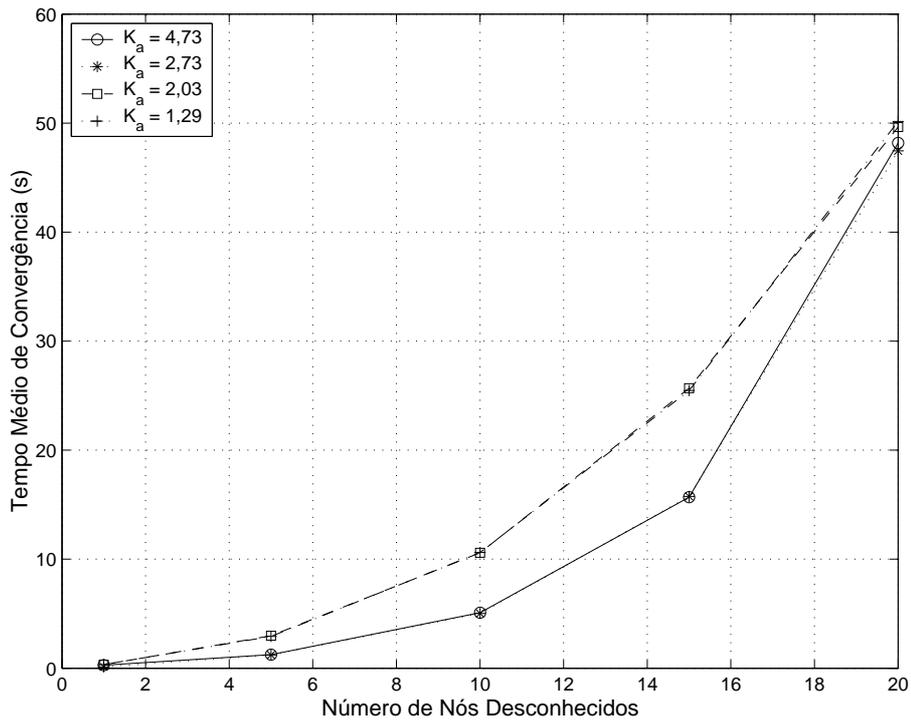


Figura 5.8: Tempo médio de convergência em função de K_a para NM-LISPCP

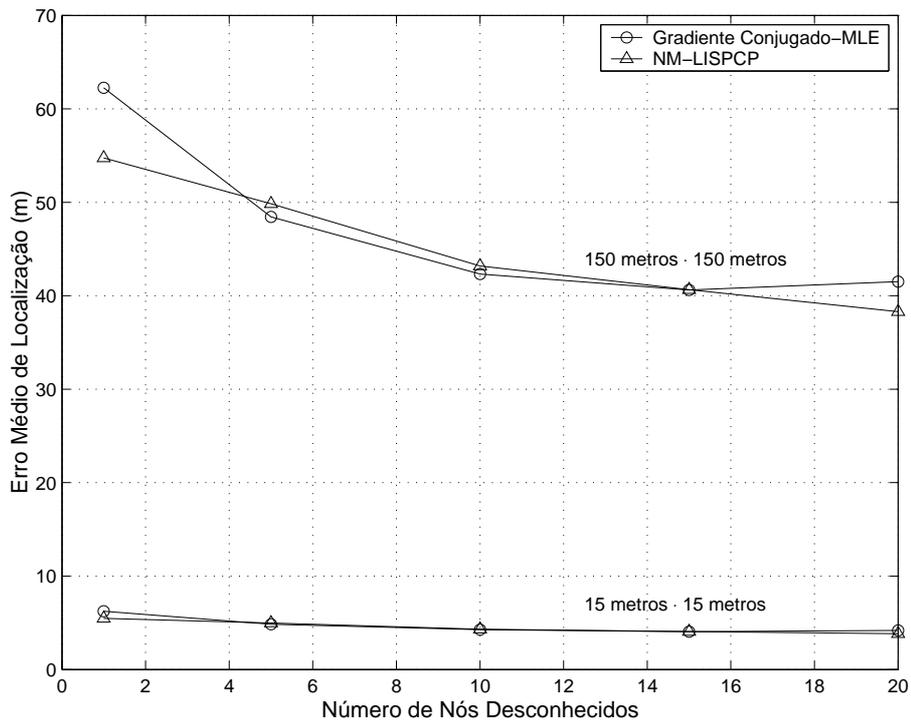


Figura 5.9: Erro médio de localização em função da área da rede para os algoritmos NM-LIPCP e *Gradiente Conjugado*

para a rede de 150 metros de lado é aproximadamente 10 vezes maior do que para a rede de 15

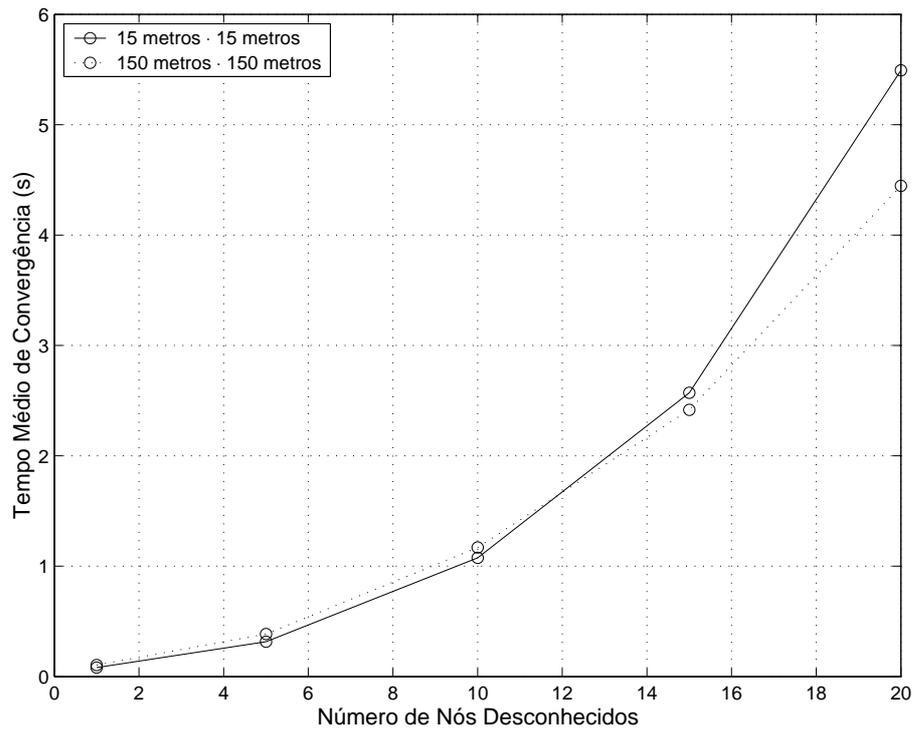


Figura 5.10: Tempo médio de convergência para algoritmo de *Gradiente Conjugado* em função da área da rede

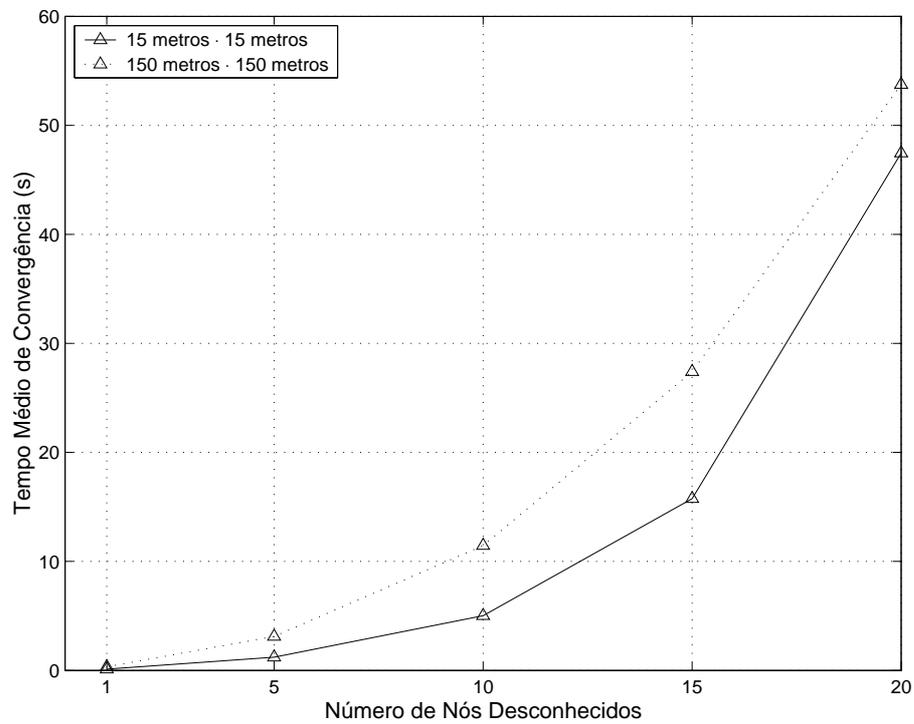


Figura 5.11: Tempo médio de convergência para o algoritmo proposto em função da área da rede

metros de lado. As figuras 5.10 e 5.11 mostram que o tempo médio de convergência não guarda relação com a área da rede.

5.1 Implementação Prática do Algoritmo

O algoritmo *NM-LISPCP* foi utilizado para a localização de um nó numa rede ad-hoc real trabalhando em ambiente aberto. Um dos parâmetros de entrada do algoritmo é o expoente de perda de percurso que foi determinado com base nos valores médios de SNR obtidos no ambiente da rede, conforme descreve a seção 5.1.1. A apresentação dos dados coletados relativos à localização do nó desconhecido, assim como a discussão dos mesmos é parte da seção 5.1.2.

5.1.1 Estimativa do Expoente de Perda de Percurso

As medições ao ar livre foram realizadas no Campus da PUC-PR, para distâncias d_{ij} entre transmissor e receptor de 10, 20, 30, 40, 50, 60, 70, 80, 90 e 100 metros. Para cada uma delas foi medida a Relação Sinal-Ruído (*Signal to Noise Ratio-SNR*) no receptor i e no transmissor j durante transferência de arquivo. Estes valores de SNR foram designados respectivamente de \overline{SNR}_i e \overline{SNR}_j . Os testes foram realizados com dois computadores portáteis e para evitar influência externa nas medidas selecionou-se o canal 4 (frequência de 2427 MHz) por estar livre e não apresentar interferência adjacente. O nível de ruído foi considerado constante no valor de $-100dBm$.

Tabela 5.1: Medidas Coletadas ao Ar Livre para Estimativa do Expoente de Perda de Percurso

d_{ij} (m)	\overline{SNR}_i (dB)	$\sigma_{\overline{SNR}_i}^2$ (dB)	\overline{SNR}_j (dB)	$\sigma_{\overline{SNR}_j}^2$ (dB)
10	52.03	11.96	55.56	13.84
20	54.68	11.70	59.08	16.68
30	49.34	30.80	54.04	10.77
40	44.44	18.06	48.14	13.18
50	40.04	30.63	43.65	35.05
60	39.24	16.90	42.32	11.81
70	28.80	16.70	31.58	15.91
80	32.02	19.54	34.19	39.25
90	34.81	4.69	37.26	17.59
100	24.08	10.74	26.51	17.99

A Tabela 5.1 apresenta o valor médio de SNR para cada separação d_{ij} e a respectiva variância das medidas. A Figura 5.12 apresenta os valores de SNR medidos para cada distância e também apresenta duas curvas de perda de percurso teóricas para $\eta = 2$ e $\eta = 3$, assim visualmente é possível ter uma idéia da grande variabilidade a que estão sujeitas as medidas de potência mesmo em ambiente externo.

As curvas com os valores otimizados através de MMSE para o expoente de perda de percurso são mostrados na figura 5.13 e são respectivamente de $\eta = 2.4$ para o receptor e de $\eta = 2.2$ para o transmissor. Na aplicação do algoritmo utilizou-se a média $\eta = 2.3$ dos valores em cada sentido.

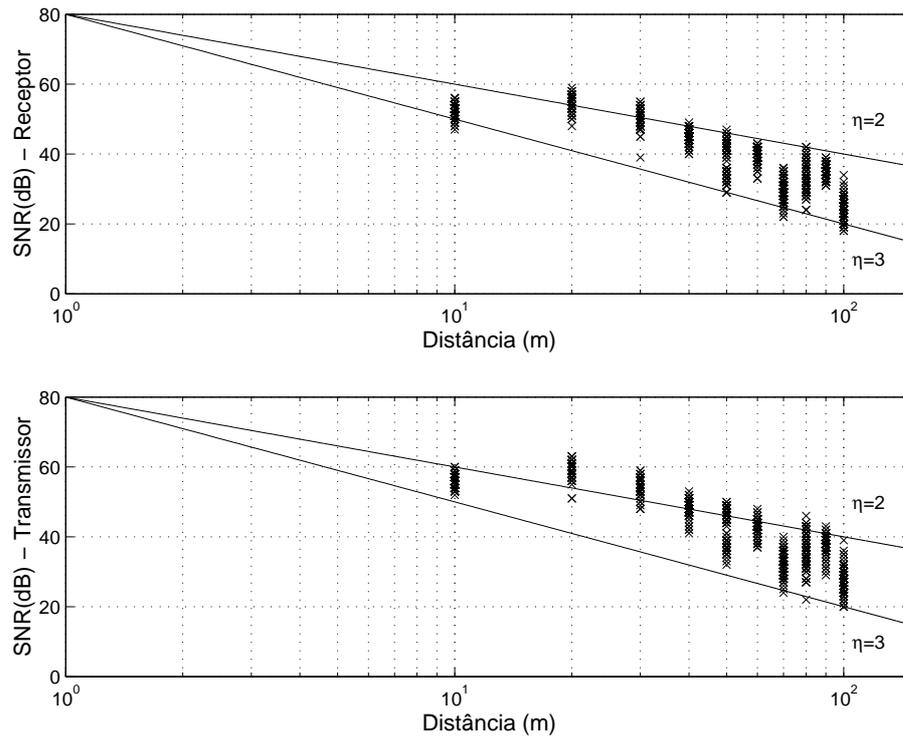


Figura 5.12: Variabilidade do Exponente de Perda de Percurso

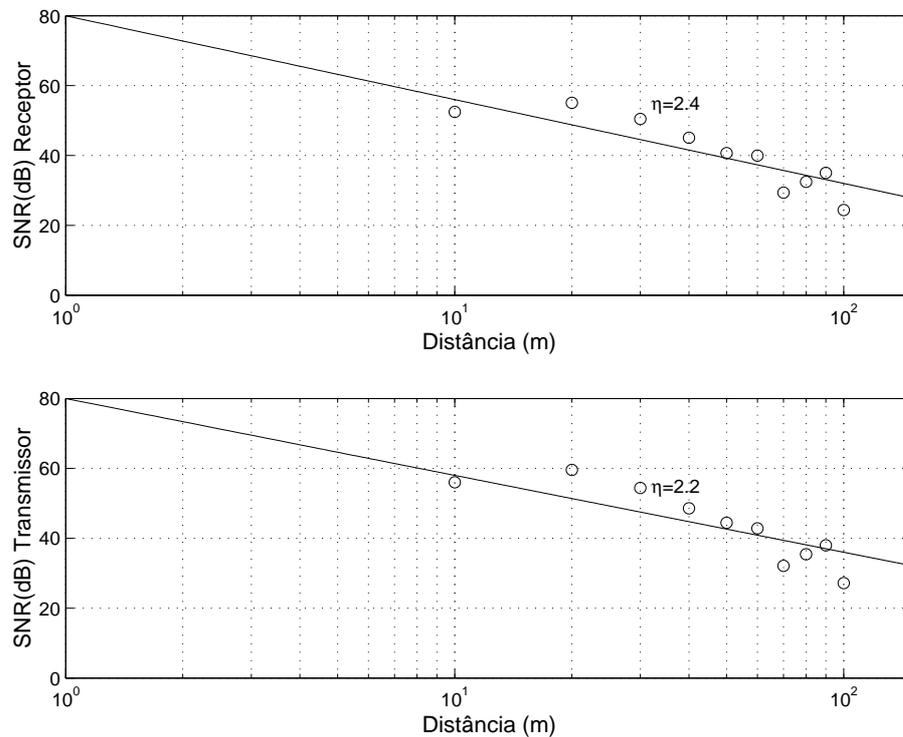


Figura 5.13: Exponente Estimado de Perda de Percurso Usando os Valores Médios de SNR

5.1.2 Aplicação Real do Algoritmo de Localização

Validou-se o algoritmo proposto também através de medidas realizadas numa rede *Ad-Hoc* IEEE802.11b montada com quatro computadores portáteis conforme cenário de rede

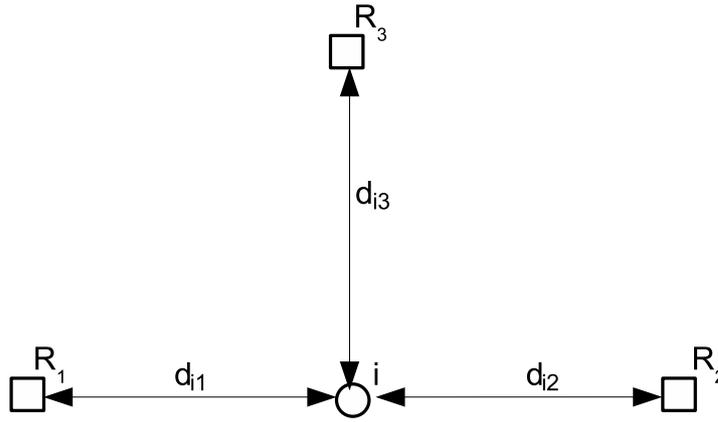


Figura 5.14: Topologia de Rede Utilizada para as Medições

apresentado pela figura 5.14. Para criar variações no cenário, as distâncias entre os nós foram alteradas conforme mostra a tabela 5.2. Cinco cenários diferentes foram utilizados e estão identificados pelas letras de A à E na primeira coluna. As medidas de potência foram coletadas de forma bidirecional: foi coletada tanto a potência que o nó desconhecido experimenta devido às transmissões dos nós de referência, quanto a potência que os nós de referência experimentam devido às transmissões do nó desconhecido. A tabela apresenta os valores médios de 1000 amostras de potência coletadas em cada sentido. \bar{P}_{ij} é a notação para a potência média coletada no nó desconhecido, \bar{P}_{ji} é a notação para a potência média coletada nos nós de referência e \bar{P} é a média dos valores considerando ambos os sentidos. O expoente de perda de percurso η utilizado é de 2.3 e que corresponde à media dos valores obtidos na seção 5.1.1. As colunas identificadas por R_j contém os valores de medidas entre o nó desconhecido e nó de referência j , onde $1 \leq j \leq 3$. A tabela apresenta também a variância para os dados coletados na coluna $\sigma_{R_j}^2$.

Para aplicação do algoritmo é necessária a determinação da potência P_0 à uma distância de referência d_0 do nó transmissor. Utilizou-se o modelo de espaço livre de Friis, equação 5.7 para determinação das perdas ocorridas até a distância $d_0 = 1m$ do transmissor. A potência do sinal em um ponto afastado de d_0 do transmissor pode ser calculada pela diferença entre a potência transmitida e as perdas impostas pelo trajeto do sinal, equação 5.8.

$$L_0^{dB} = 20 \cdot \log\left(\frac{4\pi d_0}{\lambda}\right) \quad (5.7)$$

$$P_0^{dBm} = P_t^{dBm} + G_t^{dBi} + G_r^{dBi} - 2 \cdot L_c^{dB} - L_0^{dB} \quad (5.8)$$

Foram considerados os dados de ganhos das antenas e perdas nos cabos fornecidos pelo fabricante, conforme tabela 5.3 e foi utilizado o canal 5 para as medidas de potência de sinal.

As simulações apresentadas nas seções anteriores consideram a potência média de recepção do sinal emitido pelas referências avaliada no nó desconhecido, \bar{P}_{ij} , para aplicação no modelo de perda de percurso. Isto é necessário sempre que os algoritmos considerarem as me-

Tabela 5.2: Tabela De Potências Medidas

cenário	d_{i1} m	d_{i2} m	d_{i3} m	sentido \bar{P}_{ij}	R_1 dBm	$\sigma_{R_1}^2$ m^2	R_2 dBm	$\sigma_{R_2}^2$ m^2	R_3 dBm	$\sigma_{R_3}^2$ m^2
A	5	5	5	\bar{P}_{ij}	-36.30	0.34	-36.58	0.39	-38.83	1.21
				\bar{P}_{ji}	-37.84	0.80	-38.04	0.66	-41.83	1.32
				\bar{P}	-37.07		-37.31		-40.33	
B	10	10	10	\bar{P}_{ij}	-43.65	0.40	-44.42	6.57	-47.27	7.94
				\bar{P}_{ji}	-45.38	0.57	-45.57	0.63	-50.31	1.32
				\bar{P}	-44.52		-45.00		-48.79	
C	20	20	20	\bar{P}_{ij}	-48.74	1.42	-48.95	6.12	-52.61	1.08
				\bar{P}_{ji}	-50.93	1.48	-50.65	0.49	-54.70	1.16
				\bar{P}	-49.84		-49.80		-53.66	
D	50	50	50	\bar{P}_{ij}	-53.07	17.17	-52.80	0.58	-55.66	1.17
				\bar{P}_{ji}	-54.86	1.04	-55.47	0.64	-57.76	1.54
				\bar{P}	-53.97		-54.14		-56.71	
E	20	10	50	\bar{P}_{ij}	-49.60	11.11	-42.71	7.74	-54.36	4.07
				\bar{P}_{ji}	-51.01	1.69	-44.42	0.78	-56.03	0.62
				\bar{P}	-49.60		-42.71		-54.36	

Tabela 5.3: Tabela De Dados do Equipamento para uso no modelo de Espaço Livre

P_t^{dBm}	15	Potência de transmissão
G_t^{dBi}	4	Ganho da antena transmissora
G_r^{dBi}	4	Ganho da antena receptora
L_c^{dB}	2	Atenuação dos cabos
$f_{canal 5}$	2.432GHz	frequência de operação no canal 5

didadas de potência entre pares de nós desconhecidos. Por meio deste ensaio prático demonstra-se que o inverso também é válido, a localização pode ser feita através das medidas efetuadas pelos nós de referência, \bar{P}_{ji} , e possibilita algumas outras aplicações, por exemplo, a detecção de nós interferentes na rede. Nós interferentes são nós não autorizados à fazer parte da rede, mas que competem pela banda disponível, causam interferências nas transmissões legítimas da rede e acabam degradando sua capacidade. A localização do nó interferente é possível porque não é necessário o suporte de nenhum protocolo dedicado por parte deste. Ao tentar ingressar na rede seu sinal será detectado e monitorado nos nós de referência. Após amostragem, o valor médio de *RSSI* é transmitido até o nó de referência central para processamento da localização do nó intruso. Portanto, nesta aplicação, a comunicação necessária à localização é restrita aos nós de referência.

Tabela 5.4: Tabela De Erros de Localização

cenário	sentido	Erro (<i>m</i>)		Redução do Erro (%)
		$\beta = 1$ Gradiente Conjugado	β dado pela equação 4.8	
A	\bar{P}_{ij}	0.70	0.68	2.86
	\bar{P}_{ji}	2.52	2.44	3.17
	\bar{P}	1.56	1.51	3.21
B	\bar{P}_{ij}	3.11	2.99	3.86
	\bar{P}_{ji}	6.84	6.54	4.39
	\bar{P}	4.97	4.79	3.62
C	\bar{P}_{ij}	2.11	1.86	11.84
	\bar{P}_{ji}	6.82	6.36	6.74
	\bar{P}	4.11	3.69	10.22
D	\bar{P}_{ij}	10.21	9.03	11.56
	\bar{P}_{ji}	6.93	6.89	0.58
	\bar{P}	8.37	7.37	11.95
E	\bar{P}_{ij}	5.74	3.93	31.53
	\bar{P}_{ji}	7.17	5.80	20.11
	\bar{P}	6.37	4.63	27.31

A tabela 5.4 compara os algoritmos proposto e de *Gradiente Conjugado* em termos de erros de localização para os cenários descritos na tabela 5.2. Da mesma forma há uma linha para cada sentido da potência medida e uma terceira para a média de ambas. Há duas colunas de erro de localização: a terceira coluna corresponde à função custo não ponderada, $\beta = 1$, enquanto que a quarta coluna considera a ponderação *beta* conforme equação 4.8. O desempenho do algoritmo de *Gradiente Conjugado* é o mesmo que do algoritmo proposto para o case de $\beta = 1$ e por motivo de clareza ambos são apresentados na mesma coluna. A quinta coluna apresenta a variação do erro de localização pela utilização de β conforme 4.8 ao invés de $\beta = 1$. O erro de localização foi maior no sentido *i*->*j* em alguns cenários e em outros foi maior no sentido inverso. Os valores de variância foram calculados com o objetivo de criar uma ponderação em função da confiabilidade das medidas. Porém, não foi possível estabelecer uma relação entre a variância das medidas e o erro de localização. Medidas de potência média associadas a variâncias maiores nem sempre acarretam um maior erro de localização. Dependendo da aplicação da

localização e do algoritmo utilizado pode se optar por utilizar um sentido de medição ou outro, ou até mesmo a média de ambos a fim de não incorrer num erro muito grande de localização. Comparando-se as colunas de erro de localização e observando-se a coluna de variação do erro, verifica-se que a utilização do coeficiente de ponderação β produz uma melhoria considerável na localização em todos os cenários e para qualquer que tenha sido a direção da medida de *RSSI*. Esta melhora é de menos de 5% para os cenários A e B, chega a quase 12% para o cenário C e D, e supera os 30% para o cenário E.

A sequência de figuras 5.15, 5.16, 5.17, 5.18, 5.19 mostra para cada um dos cinco cenários a posição de início da busca e a posição onde o nó desconhecido foi encontrado tanto pelo algoritmo de *Gradiente Conjugado* quanto pelo algoritmo proposto. As referências são representadas pelos quadrados, a posição real do nó desconhecido pelo triângulo, o ponto inicial de busca pelo círculo, a posição onde o nó foi localizado pelo algoritmo proposto pelo losango e a posição onde o nó foi localizado pelo algoritmo de *Gradiente Conjugado* pela cruz. O segmento de reta desde a posição real do nó desconhecido até a posição onde o mesmo foi localizado pelo algoritmo proposto corresponde ao erro de localização. As coordenadas dos pontos iniciais de busca e as coordenadas onde os nós foram localizados são identificadas ao lado direito da sinalização correspondente. Quando as coordenadas do nó desconhecido determinadas por ambos os algoritmos forem muito próximas, apenas as coordenadas dadas pelo algoritmo proposto são apresentadas. Apenas os resultados relativos às medidas de *RSSI* efetuadas pelo nó desconhecido, P_{ij} , são apresentadas para maior clareza de cada figura.

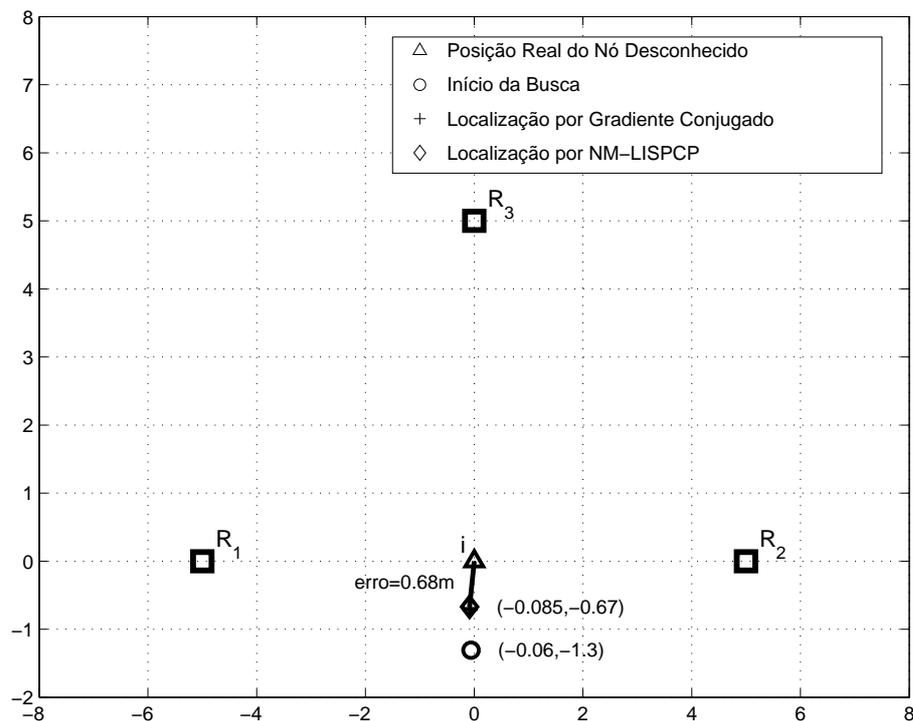


Figura 5.15: Comparação de desempenho entre *Gradiente Conjugado* e algoritmo proposto no cenário A

A tabela 5.5 ajuda a entender os resultados obtidos. Para cada cenário esta tabela apresenta os valores reais das distâncias entre o nó desconhecido e os nós de referência. Há uma linha para a estimativa de distância em cada sentido da medição: \hat{d}_{ij} e \hat{d}_{ji} , e uma quarta

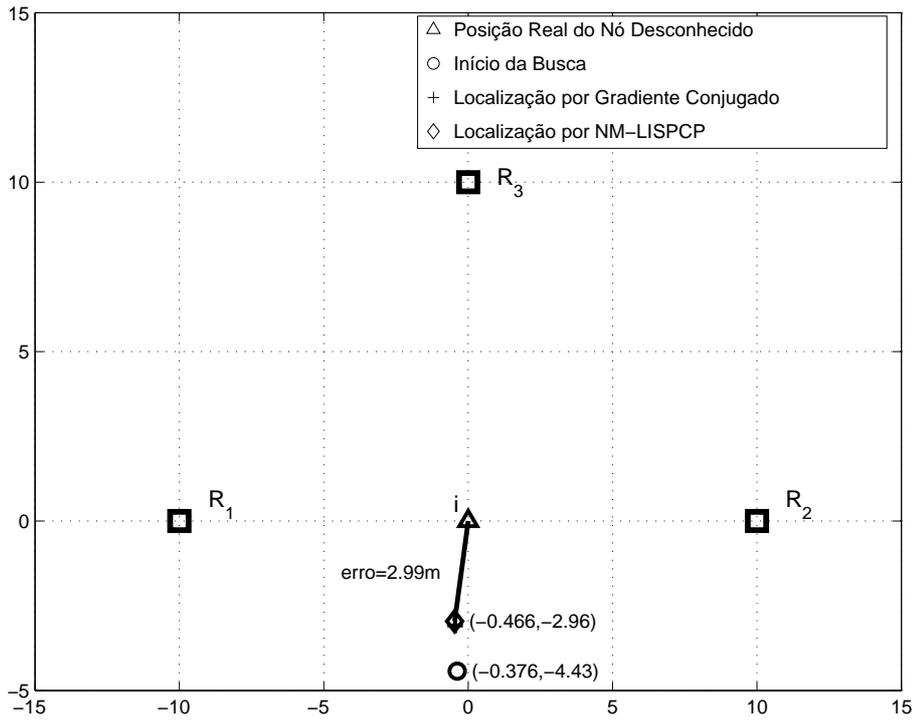


Figura 5.16: Comparação de desempenho entre *Gradiente Conjugado* e algoritmo proposto no cenário B

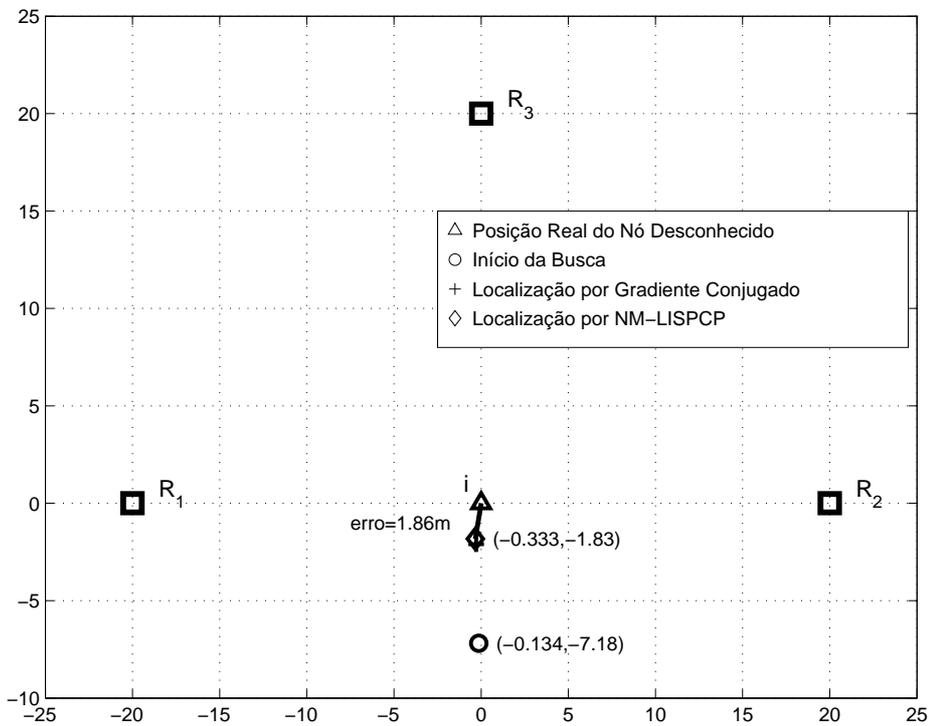


Figura 5.17: Comparação de desempenho entre *Gradiente Conjugado* e algoritmo proposto no cenário C

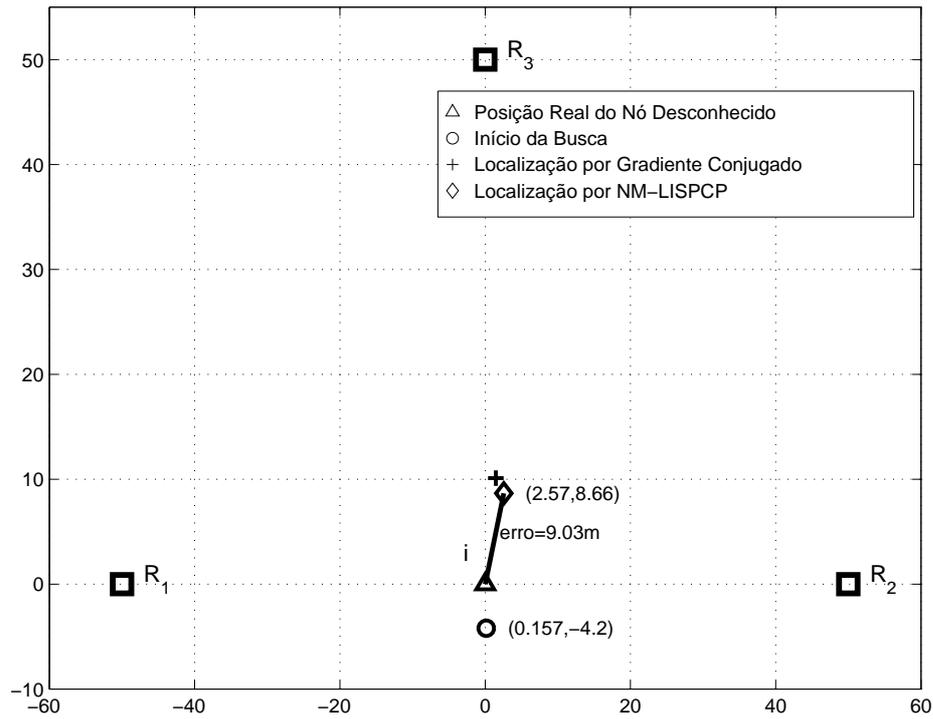


Figura 5.18: Comparação de desempenho entre *Gradiente Conjugado* e algoritmo proposto no cenário D

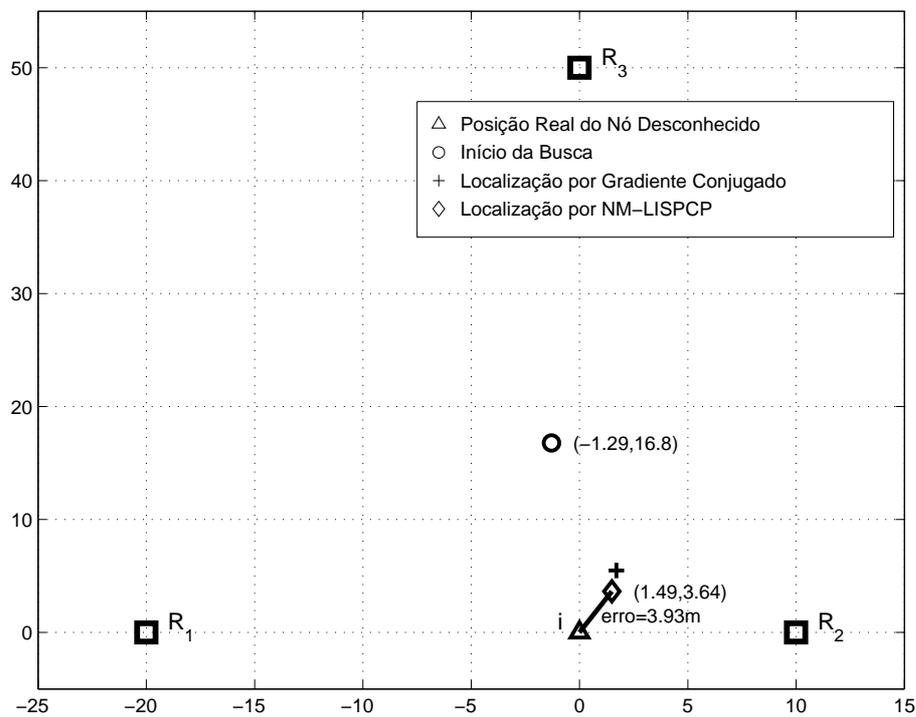


Figura 5.19: Comparação de desempenho entre *Gradiente Conjugado* e algoritmo proposto no cenário E

linha para a estimativa de distância a partir do valor médio de *RSSI* em ambos os sentidos da medição: \hat{d} .

Tabela 5.5: Tabela De Estimativas De Distância

cenário	distância	R_1	R_2	R_3
A	d_{ij}	5	5	5
	\widehat{d}_{ij}	4.5521	4.6815	5.8643
	\widehat{d}_{ji}	5.3109	5.4184	7.9186
	\widehat{d}	4.9169	5.0365	6.8145
B	d_{ij}	10	10	10
	\widehat{d}_{ij}	9.5012	10.2626	13.6512
	\widehat{d}_{ji}	11.2979	11.5149	18.5074
	\widehat{d}	10.3607	10.8707	15.8949
C	d_{ij}	20	20	20
	\widehat{d}_{ij}	15.8156	16.1516	23.2995
	\widehat{d}_{ji}	19.6926	19.1482	28.7221
	\widehat{d}	17.6479	17.5862	25.8691
D	d_{ij}	50	50	50
	\widehat{d}_{ij}	24.3975	23.7469	31.6195
	\widehat{d}_{ji}	29.1858	31.0237	39.0175
	\widehat{d}	26.6845	27.1425	35.1242
E	d_{ij}	20	10	50
	\widehat{d}_{ij}	17.2376	8.6479	27.7609
	\widehat{d}_{ji}	19.8509	10.2626	32.8127
	\widehat{d}	18.4982	9.4207	30.1813

Quando o coeficiente β é aplicado à função custo, os maiores valores de *RSSI* reportados recebem um peso maior, pois correspondem às medições mais precisas por serem provenientes dos nós mais próximos do nó desconhecido. Por isso a melhora pelo uso de β foi maior para o cenário E, onde as distâncias entre os três nós de referência e o desconhecido são diferentes. Para os cenários A, B e C a melhora se deve ao fato de que os valores médios de *RSSI* reportados causaram erro positivo nas estimativas de distância com maior intensidade, principalmente para as medidas efetuadas em relação à referência R_3 . O nó desconhecido percebe estar mais afastado desta referência do que realmente está. Quanto mais distante i e j estão, menor o peso de seu valor de *RSSI* na função custo e portanto as medidas errôneas de *RSSI* tiveram menor influência na otimização que as mais precisas e isto produziu uma melhora de quase 12% na localização. O cenário D apresenta grandes valores de erro negativo nas estimativas de distância, o nó desconhecido percebe estar mais próximo dos nós de referência do que realmente está. As estimativas de distância relativa \hat{d}_{i1} , \hat{d}_{1i} , \hat{d}_{i2} e \hat{d}_{2i} ficaram com valores bastante próximos, mas com erros que chegaram a mais de 25m ou 50%. A estimativa de distância relativa até R_3 foi mais próxima da distância real da separação entre os nós, $d_{i3} = 50m$. Observando-se a figura 5.14 nota-se que o nó i está posicionado exatamente entre R_1 e R_2 . Portanto, mesmo que os valores de *RSSI* entre o nó i e os nós de referência R_1 e R_2 estejam errados haverá uma tendência do algoritmo de otimizar a localização de i equidistante de R_1 e R_2 desde que as potências médias reportadas até estas duas referências tenham valores próximos, como foi o caso. A ponderação β acaba diminuindo o erro da localização ainda mais ao atribuir pesos maiores aos valores de *RSSI* reportados até R_1 e R_2 , visto que a referência R_3 reportou um valor de *RSSI* menor, apesar de mais preciso. Sendo assim o maior erro de localização para o cenário D foi de 9.03m o que não é considerado um erro grande para a técnica de estimativa de distância por potência de sinal recebido dada a separação entre os nós de 50m.

5.2 Conclusão

O capítulo apresentou o algoritmo proposto e mostrou uma série experimentos de comparações entre este e o algoritmo de *Gradiente Conjugado*. Foi validada a eficácia do sequenciamento e das ponderações α e β . Os resultados obtidos são equivalentes em termos de erro médio para quando a busca se inicia em um ponto uniformemente distribuído na rede. A utilização do método geométrico de seleção do ponto inicial de busca proporciona tanto um menor erro médio de localização quanto uma convergência mais rápida para o algoritmo de *Gradiente Conjugado*. O tempo de convergência independe da área ou de K_a . Quanto aos resultados da aplicação prática dos algoritmos na rede *ad-hoc*, pode-se dizer que o algoritmo proposto teve um desempenho até 30% melhor que o de *Gradiente Conjugado* pelo uso do coeficiente β . Apresentou um erro relativamente pequeno, menor que 10m, para uma rede com apenas 3 referências distantes de até 50m do nó desconhecido.

Capítulo 6

Conclusões

As redes sem-fio têm um papel cada vez mais constante e importante no modo de vida das populações e dentro dos escritórios, fábricas e escolas. As inúmeras aplicações a que se destinam requerem o conhecimento da localização dos nós participantes. Espera-se que mecanismos baratos possam fazer a localização sob demanda dos nós de grandes redes e que assim ofereçam o suporte necessário aos protocolos de roteamento baseados em localização, ao gerenciamento da rede e às aplicações em que a localização é informação vital. Este trabalho apresentou o estudo do uso do método de otimização não-linear de Nelder-Mead como base na elaboração de algoritmos de localização que utilizam apenas a potência média de sinal recebido (*RSSI*) como medida de distância relativa. Os dados para pesquisa foram obtidos tanto através de simulações de diversos cenários típicos de redes sem-fio quanto na aplicação do algoritmo para processamento de medidas coletadas numa rede *ad-hoc* real. Três diferentes funções custo foram analisadas em algoritmos do tipo conjunto e iterativo. Demonstrou-se que os algoritmos do tipo conjunto fornecem uma melhor precisão da localização, enquanto que os iterativos têm menor tempo de convergência. Para rede com poucos nós, a adição de mais um nó nem sempre melhora a qualidade da localização, a posição relativa dos nós é um dos fatores determinantes do erro de localização. Por isso é importante se calcular o erro médio a partir de experimentos realizados em diferentes arranjos de nós. Um nó no centro da rede apresenta erro de localização maior que um nó na periferia. Foi proposta uma métrica para sequenciamento dos nós na localização iterativa que leva em consideração as distâncias relativas medidas entre nós e que diminui o erro de localização consideravelmente. Implementou-se uma métrica para ponderação da função custo que produz uma diminuição adicional do erro de localização demonstrado no cenário de rede real. A técnica de estimativa de distância baseada na potência do sinal está sujeita a diversos erros, principalmente o sombreamento, por isso é recomendada para redes densas com grande quantidade de nós onde o espaçamento entre nós seja pequeno. O uso do algoritmo de tipo iterativo com a função custo dada pelo Erro Médio Quadrático (variante *NM-LISPCP*) é o que oferece boa precisão à um custo de tempo de convergência relativamente baixo. Demonstrou-se que o erro médio de localização do algoritmo proposto é equivalente ao do algoritmo *Gradiente Conjugado-MLE* quando a busca inicia em um ponto qualquer randomicamente distribuído na rede. Recomenda-se novos estudos para a elaboração deste algoritmo numa versão distribuída e sua implementação em cada nó da rede. Desse modo um nó em movimento pode recalculer sua posição sem a necessidade de recálculo da posição dos demais. Os coeficientes de ponderação merecem uma pesquisa mais detalhada a fim de avaliar a possibilidade de diminuição adicional do erro de localização. A continuidade deste

trabalho pode ainda considerar cenários onde as referências estejam em quantidades variadas e distribuídas desordenadamente na rede.

Referências Bibliográficas

- [Akyildiz et al., 2002] Akyildiz, I., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422.
- [Alicherry et al., 2005] Alicherry, M., Bhatia, R., and Li, L. E. (2005). Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *Proceedings of ACM Mobile Communications (MobiCom)*.
- [Beutel, 2005] Beutel, J. (2005). *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, pages 20:1–23. CRC PRESS.
- [Blazevic et al., 2001] Blazevic, L., Buttyan, L., Capkun, S., Giordano, S., Hubaux, J., and Boudec, J. L. (2001). Self-organization in mobile ad hoc networks: The approach of terminodes. *IEEE Communications Magazine*, 39(6):166–174.
- [Costa et al., 2006] Costa, J. A., Patwari, N., and Hero, A. O. (2006). Distributed weighted-multidimensional scaling for node localization in sensor networks. In *Proceedings of ACM Transactions on Sensor Networks*.
- [Gezici et al., 2005] Gezici, S., Tian, Z., Giannakis, G. B., Kobayashi, H., Molisch, A. F., Poor, H. V., and Sahinoglu, Z. (2005). Localization via ultra-wideband radios. *IEEE Signal Processing Magazine*, 22(4):70–84.
- [Gupta and Kumar, 2000] Gupta, P. and Kumar, P. R. (2000). The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404.
- [Krishnamachari, 2005] Krishnamachari, B. (2005). *Networking Wireless Sensors*. Cambridge.
- [Nelder and Mead, 1965] Nelder, J. A. and Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, 7:308–313.
- [Neto and Maia, 2008] Neto, J. A. C. and Maia, M. A. (2008). Roteamento baseado em localização em redes ad-hoc. In *Anais SBRT 2008*.
- [Patwari, 2009] Patwari, N. (2009). Wireless sensor network localization repository. Disponível em <http://www.eecs.umich.edu/hero/localize>. Acesso em: 17 de maio de 2009.
- [Patwari and Hero, 2003] Patwari, N. and Hero, A. O. (2003). Using proximity and quantized rssi for sensor localization in wireless networks. In *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications (WSNA)*.

- [Patwari et al., 2003] Patwari, N., Hero, A. O., O’Dea, R. J., and Perkins, M. (2003). Relative location in wireless sensor networks. *IEEE Transactions on Signal Processing*, 51(8):2137–2148.
- [Patwari et al., 2005] Patwari, N., Joshua, N., Kyperountas, S., Hero, A. O., Moses, R. L., and Correal, N. S. (2005). Locating the nodes - cooperative localization in wireless sensor networks. *IEEE Signal Processing Magazine*, 22(4):54–69.
- [Patwari et al., 2001] Patwari, N., O’Dea, R. J., and Wang, Y. (2001). Relative location in wireless networks. *IEEE Vehicular Technology Conference (VTC-Spring)*, 2:1149–1153.
- [Ramachandran et al., 2005] Ramachandran, K. N., Buddhikot, M. M., Chandranmenon, G., Miller, S., Belding-Royer, E. M., and Almeroth, K. C. (2005). On the design and implementation of infrastructure mesh networks. In *Proceedings of the IEEE Workshop on Wireless Mesh Networks (WiMesh)*.
- [Rappaport, 1996] Rappaport, T. S. (1996). *Wireless Communication: Principles & Practice*. Prentice Hall.
- [Rappaport and McGillem, 1989] Rappaport, T. S. and McGillem, C. D. (1989). Uhf fading in factories. *IEEE Journal on Selected Areas in Communications*, 7(1):40–48.
- [Savvides et al., 2001] Savvides, A., Han, C. C., and Srivastava, M. B. (2001). Dynamic fine grained localization in ad-hoc sensor networks. In *Proceedings of ACM Mobile Communications (MobiCom)*.
- [Song, 1994] Song, H. L. (1994). Automatic vehicle location in cellular communication systems. *IEEE Transactions on Vehicular Technology*, 43:902–9908.
- [Tseng et al., 2005] Tseng, Y.-C., Huang, C.-F., and Kuo, S.-P. (2005). *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, pages 21:1–15. CRC PRESS.
- [Wann and Chin, 2007] Wann, C.-D. and Chin, H.-C. (2007). Hybrid toa/rssi wireless location with unconstrained nonlinear optimization for indoor uwb channels. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*.
- [Yang et al., 2005] Yang, Y., Wang, J., and Kravets, R. (2005). Designing routing metrics for mesh networks. In *Proceedings of the IEEE Workshop on Wireless Mesh Networks (WiMesh)*.

Apêndice A

O Método do Poliedro Flexível na Localização

O procedimento de otimização adotado neste trabalho é chamado "O método do Poliedro Flexível"[Nelder and Mead, 1965], um método de busca para a solução de problemas de programação não-linear sem restrição. O mecanismo básico de funcionamento é determinação da direção de busca baseado nas melhores soluções das iterações anteriores. Uma figura geométrica chamada de poliedro é formada por um conjunto de vértices, onde cada vértice corresponde à uma solução θ . O número de vértices é o número de variáveis a serem otimizadas incrementado de um. Para o caso da localização iterativa no espaço bidimensional duas variáveis são definidas (x_i, y_i) . A cada iteração uma nova solução é computada e os piores vértices substituídos. Considerando uma sucessão de iterações, a tendência é o poliedro se ajustar em torno de uma solução ótima e reduzir sua dimensão (distância entre os vértices) até que um dado critério de convergência seja atingido.

Seja Θ o espaço de possíveis localizações para um dado nó i . O problema de localização pode ser definido como um problema de otimização expresso em termos de uma função custo $f_c(\theta)$, que consiste em encontrar um vetor $\theta = \theta^* \in \Theta$ que minimize f_c . Matematicamente, isto pode ser expresso como:

$$\theta^* \in \Theta \quad e \quad f_c(\theta^*) = \min_{\theta \in \Theta} f_c(\theta) \quad (\text{A.1})$$

Para cada elemento θ pertencente ao conjunto de soluções Θ , $f_c(\theta)$ representa seu nível de rejeição. Para a aplicação de localização foram elaboradas quatro diferentes funções custo equacionadas em: (3.3), (3.5), (3.6), (4.7). Para ser uma solução válida θ deve satisfazer um conjunto de critérios relacionados com a natureza do problema a ser resolvido.

1. Dada uma solução inicial θ_{i_1} , inicializar os vértices do poliedro como vetores linearmente independentes como indicado em A.2. Os coeficientes d_m^+ e d_m^- representam, respectivamente, uma grande e uma pequena distância com respeito aos elementos do vetor inicial θ_{i_1} . A escolha destes parâmetros está relacionada ao espaço de busca inicial.

$$\begin{aligned} \theta_{i_2} &= \theta_{i_1} + [d_1^+ \ d_2^-]^T \\ \theta_{i_3} &= \theta_{i_1} + [d_1^- \ d_2^+]^T \end{aligned} \quad (\text{A.2})$$

2. Computar a função custo $f_c(\theta)$ para cada vertex, ordenando e identificando os vértices de acordo com o valor da função custo: $f_c(\theta_{i_1}) \leq f_c(\theta_{i_2}) \leq f_c(\theta_{i_3})$.
3. Computar a centróide excluindo o pior vértice como segue:

$$\bar{\theta}_i = \frac{1}{2} \cdot \sum_{i=1}^2 (\theta_{i_1} - \theta_{i_3}) \quad (\text{A.3})$$

4. Compute θ_{i_r} usando a operação de reflexão: $\theta_i = \bar{\theta}_i - a(\theta_{i_3} - \bar{\theta}_i)$, onde a representa o passo de busca de reflexão. Se $f_c(\theta_{i_1}) \leq f_c(\theta_{i_r}) < f_c(\theta_{i_2})$, substituir θ_{i_3} por θ_{i_r} e ir para o passo 10. Se $f_c(\theta_{i_r}) < f_c(\theta_{i_1})$, ir para o passo 5. Se $f_c(\theta_{i_2}) \leq f_c(\theta_{i_r}) < f_c(\theta_{i_3})$, ir para o passo 6. Ou então, se $f_c(\theta_{i_r}) \geq f_c(\theta_{i_3})$, ir para o passo 7.
5. Computar θ_{i_e} usando a operação de expansão: $\theta_{i_e} = \bar{\theta}_i - b(\bar{\theta}_i - \theta_{i_r})$, onde b representa o passo de busca por expansão. Se $f_c(\theta_{i_e}) < f_c(\theta_{i_r})$, substituir θ_{i_3} por θ_{i_e} e ir para o passo 9. Em outro caso aceitar θ_{i_r} como nova solução e ir para passo 8.
6. Computar θ_{i_c} usando a operação de contração: $\theta_{i_c} = \bar{\theta}_i - c(\bar{\theta}_i - \theta_{i_r})$, onde c representa o passo de busca de contração. Se $f_c(\theta_{i_c}) \leq f_c(\theta_{i_r})$, substituir θ_{i_3} por θ_{i_c} e ir para o passo 9. Em outro caso, ir para passo 8.
7. Computar θ_{i_c} usando a operação de contração interna: $\theta_{i_c} = \bar{\theta}_i - c(\bar{\theta}_i - \theta_{i_3})$, onde c representa o passo de busca de contração. Se $f_c(\theta_{i_c}) < f_c(\theta_{i_3})$, substituir θ_{i_3} por θ_{i_c} e ir para o passo 9. Em outro caso, ir para passo 8.
8. Executar um passo de redução recalculando todos os vértices e excluindo os melhores vértices: $\theta_i = \theta_{i_1} - s(\theta_{i_1} - \theta_{i_k})$, para $k=2,3$, onde s representa o passo de redução.
9. Usar A.4 para reconstruir o poliedro. Se a condição não for satisfeita, retornar ao passo 2. Ir para o passo 10 em outro caso.

$$\| \theta_{i_2} - \theta_{i_1} \| \leq \varepsilon_1 \quad (\text{A.4})$$

10. Se $\theta_{k_1 old}$ não tiver sido inicializada ainda, fazer $\theta_{k_1 old} = \theta_{k_1}$ e retornar ao passo 1. Em outro caso, usar A.5 para testar o critério de convergência. Se o critério for satisfeito, fazer $\theta_i^* = \theta_{i_1}$ e terminar o procedimento de otimização. Senão, fazer $\theta_{k_1 old} = \theta_{k_1}$ e retornar ao passo 1.

$$\| \theta_{i_2} - \theta_{i_1} \| \leq \varepsilon_2 \quad (\text{A.5})$$