

GUSTAVO CAMPOS MENEZES

**MODELO E ALGORITMOS PARA A
DEFINIÇÃO DA DENSIDADE, COBERTURA E
CONNECTIVIDADE EM UMA REDE DE
SENSORES SEM FIO**

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

Belo Horizonte
13 de Agosto de 2004



UNIVERSIDADE FEDERAL DE MINAS GERAIS



FOLHA DE APROVAÇÃO

Modelo e Algoritmos para a Definição da Densidade, Cobertura e Conectividade
em uma Rede de Sensores Sem Fio

GUSTAVO CAMPOS MENEZES

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

Prof. GERALDO ROBSON MATEUS - Orientador
Departamento de Ciência da Computação - UFMG

Prof. MAURÍCIO CARDOSO DE SOUZA
Departamento Engenharia de Produção - EE- UFMG

Prof. RICARDO HIROSHI CALDEIRA TAKAHASHI
Departamento de Matemática - UFMG

Profa LINNYER BEATRY S RUIZ
Departamento de Engenharia Elétrica - UFMG

Belo Horizonte, 13 de agosto de 2004.

Resumo

As Redes de Sensores Sem Fio (RSSF) vêm propondo grandes desafios. Um destes desafios é o problema de cobertura, que consiste na garantia de uma qualidade de monitoramento para uma determinada área ou ambiente. Outro desafio, é o problema de controle da densidade dos nós sensores. Este problema consiste em determinar o menor número de nós sensores ativos dispostos em uma área de monitoramento de forma a garantir a cobertura e conectividade da rede. Este trabalho apresenta um modelo de programação linear inteira mista que tem por objetivo resolver estes problemas. Além do modelo de otimização, é proposta uma heurística baseada na utilização da Relaxação Lagrangeana e do método de sub-gradientes. Os resultados computacionais mostram que a heurística utilizada é capaz de fornecer soluções ótimas para um grande número de instâncias, além de fornecer soluções com um esforço computacional muito menor que o utilizado por pacotes de otimização como o CPLEX.

Abstract

The Wireless Sensor Networks have been carrying great challenges. One of them is the coverage problem which consists in assuring the monitoring quality for a specific area or environment. Another challenge, is to control the density of sensors. That is, from a set of sensors deployed in an area of monitoring, determine the less number of sensors that must be activated in order to guarantee the coverage and connectivity in the network. This paper presents a mixed integer linear programming model which aims to solve these problems. Besides the optimization model, it is also proposed a heuristic based on Lagrangean Relaxation and sub-gradient methods. The computational results show that the heuristic used is able to provide optimal solutions to a large range of instances. Moreover, a significant reduction in computational effort is achieved, as compared to optimization packages such as CPLEX.

Aos meus pais, Mário e Zilda.

Agradecimentos

Muito obrigado ao Professor Geraldo Robson Mateus pelo apoio, paciência e orientação para o desenvolvimento deste trabalho. Em especial, agradeço ao Zenilton, meu professor e orientador na graduação, com quem realizei projetos de graduação e iniciação científica que foram fundamentais para a minha formação e preparação para a entrada no curso de mestrado.

Agradeço também aos meus pais Mário e Zilda, a minha irmã Laura, ao meu primo-irmão Heleno, a Simone e ao Thiago, a todos os meus amigos, pelo apoio, votos de confiança e suporte nos momentos difíceis.

A galera do LaPo e redondezas: ao João, Pedrop, Silvana, Filipe, Kíssia, Lena, Gurban, Cris, Martín, Habib, Vinícius, Dri, Jayme, Olga, André (aldo), Fabíola, Zenilton, Gilberto, Fumega, Ricardo Poley, Ricardo Camargo, Wagner, Djavan, Fred, pessoas com quem estive mais próximo não só nos laboratórios e corredores do DCC, mas também nos churrascos, festas de fim de semana e viagens. Em especial ao André (aldo) e Fabíola, colegas de trabalho e amigos por toda a ajuda recebida.

Aos amigos do Mestrado e Doutorado da turma de 2002 pelo apoio e companherismo.

Ao CNPq e CAPES pelo apoio financeiro indispensável a realização deste trabalho.

Sumário

Lista de Figuras	ix
Lista de Tabelas	x
1 Introdução	1
1.1 Objetivo	2
1.2 Resultados Obtidos	3
1.3 Estrutura do Documento	3
2 Redes de Sensores Sem Fio	4
2.1 Classificação das RSSFs	4
2.2 Localização em RSSFs	5
2.3 Cobertura em RSSFs	6
2.4 Densidade nas RSSFs	7
2.5 Conectividade em RSSFs	7
2.6 Trabalhos Relacionados	8
2.7 Considerações Finais	9
3 Formulação do Problema	10
3.1 Definição do Problema	10
3.2 Modelo de Programação Matemática	11
3.2.1 Formulação	11
3.2.2 Flexibilidade	12
3.3 Complexidade	14
3.3.1 Problemas de Recobrimento	14
3.4 Considerações Finais	15
4 Algoritmo de Solução	16
4.1 Apresentação da Técnica	16
4.2 Obtenção do Limite Inferior	17
4.2.1 Lagrangeano	17
4.2.2 Lagrangeano Dual	19
4.3 Limite Superior	21
4.3.1 Limite Superior Inicial	21
4.3.2 Heurística Lagrangeana	23
4.4 Considerações Finais	23

5	Resultados Computacionais	24
5.1	Gerador de Instâncias	24
5.1.1	Localização dos Nós Sensores	25
5.1.2	Localização dos Nós Sorvedouros	25
5.1.3	Instâncias de Teste	25
5.2	Experimentos	28
5.2.1	Avaliação do Algoritmo de Solução	28
5.2.2	Apresentação dos Resultados	29
5.2.3	Bateria de testes 1	29
5.2.4	Bateria de testes 2	31
5.2.5	Bateria de testes 3	32
5.2.6	Bateria de testes 4	33
5.3	Considerações Finais	38
6	Considerações Finais	39
6.1	Conclusão	39
6.2	Trabalhos Futuros	39

Lista de Figuras

1.1	Rede de sensores sem fio.	2
2.1	Raio de Sensoriamento.	6
2.2	Cobertura em RSSFs.	6
2.3	Raio de Comunicação.	7
3.1	Pontos	11
3.2	Nós Sorvedouros	11
3.3	Nós Sensores	11
3.4	Solução para o PDCC-RSSF.	14
5.1	Disposição dos nós sensores.	25
5.2	Primeira disposição dos nós.	25
5.3	Segunda disposição dos nós.	25
5.4	Obstáculos para as instâncias p45 e p46.	28
5.5	Obstáculos para as instâncias p47 e p48.	28
5.6	Obstáculos para as instâncias p49 e p50.	28
5.7	Obstáculos para as instâncias p51 e p52.	28
5.8	40 nós sem obstáculo 30X30.	34
5.9	40 nós com obstáculo 30X30.	34
5.10	40 nós sem obstáculo 60X30.	35
5.11	40 nós com obstáculo 60X30.	35
5.12	60 nós sem obstáculo 30X30.	35
5.13	60 nós com obstáculo 30X30.	35
5.14	60 nós sem obstáculo 60X30.	36
5.15	60 nós com obstáculo 60X30.	36
5.16	80 nós sem obstáculo 30X30.	36
5.17	80 nós com obstáculo 30X30.	36
5.18	80 nós sem obstáculo 60X30.	37
5.19	80 nós com obstáculo 60X30.	37
5.20	100 nós sem obstáculo 30X30.	37
5.21	100 nós com obstáculo 30X30.	37
5.22	100 nós sem obstáculo 60X30.	38
5.23	100 nós com obstáculo 60X30.	38

Lista de Tabelas

5.1	Conjunto de instâncias da primeira bateria.	26
5.2	Conjunto de instâncias da segunda bateria.	26
5.3	Conjunto de instâncias da terceira bateria.	27
5.4	Conjunto de instâncias da quarta bateria.	27
5.5	Experiência Computacional - Bateria 1.1.	30
5.6	Experiência Computacional - Bateria 1.2.	30
5.7	Experiência Computacional - Bateria 2.	31
5.8	Comparação CPLEX com 4 e 2 nós sorvedouros.	32
5.9	Comparação Heurística com 4 e 2 nós sorvedouros.	32
5.10	Comparação CPLEX com Heurística - 2 nós sorvedouros.	33
5.11	Comparação CPLEX com Heurística considerando obstáculos.	33

Capítulo 1

Introdução

O avanço na área de micro-processadores, materiais de sensoriamento, sistemas eletromecânicos (Micro Electro-Mechanical Systems - MEMS), tem tornado possível o desenvolvimento de dispositivos dotados de capacidade de processamento, comunicação e sensoriamento chamados de nós sensores [35]. Estes dispositivos, associados à tecnologia de redes sem fio, permitiram a criação de uma área de pesquisa multidisciplinar conhecida pelo nome de Rede de Sensores Sem Fio (RSSF).

RSSFs são dispositivos dotados de limitada capacidade de processamento e energia conectados por uma rede sem fio com o objetivo de detectar objetos ou fenômenos químicos, físicos, biológicos em uma determinada região ou ambiente.

Estes nós sensores apresentam vantagens como tamanho reduzido, baixo custo o que possibilita a disposição de inúmeros destes sensores em regiões onde o acesso por máquinas maiores ou pessoas é muito difícil ou perigoso [16]. Embora com recursos limitados, a disposição destes sensores em uma determinada área e a criação de uma rede sem fio, permite a execução de uma função colaborativa onde tarefas maiores podem ser realizadas. RSSF apresenta um vasto campo de aplicação:

- Meio Ambiente: para monitorar variáveis ambientais em locais internos como prédios e residências ou locais externos como florestas, desertos, oceanos;
- Tráfego: para monitorar tráfego de veículos em rodovias e vias urbanas;
- Medicina/Biologia: os nós sensores podem ser introduzidos no corpo humano para monitoramento de órgãos como coração, detectar a presença de substâncias que podem indicar o surgimento de um problema biológico;
- Militar: para detectar a presença de inimigos, explosões, presença de material perigoso: gases venenosos e radiação.

Alguns outros exemplos de aplicações podem ser encontrados em [16], [1].

Os nós sensores consistem basicamente de cinco componentes: bateria, memória, processador, sensor e rádio [38]. A bateria é o armazenador de energia e possui capacidade limitada, o que restringe a quantidade de energia disponível na rede. A memória e o processador são utilizados para armazenamento e processamento das informações coletadas na área de monitoramento e possuem capacidade reduzida em virtude das dimensões do nó. O sensor é responsável pelo monitoramento da área e pode ser de temperatura, sísmico, detector de movimento, entre outros. O rádio inclui o sistema de transmissão, recepção, amplificador e antena.

Em uma aplicação em RSSF estes nós são depositados em uma área para monitorar o comportamento de um fenômeno e disseminar as informações coletadas para outros nós ou para um observador [39] conforme mostrado na figura (1.1). Os nós que geram os dados são denominados nós fontes e estes dados chegam ao observador através de pontos de acesso na rede. Estes pontos podem ser estações rádio base e os próprios nós sensores, denominados nós sorvedouros ou monitores.

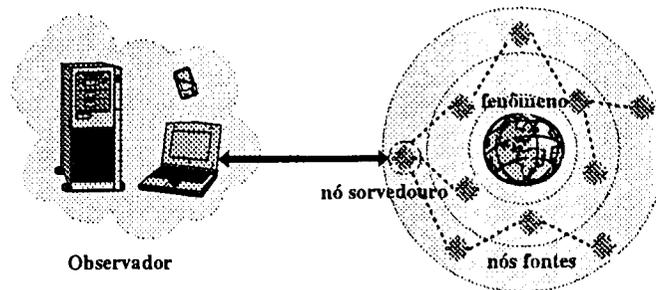


Figura 1.1: Rede de sensores sem fio.

Em determinados ambientes o nó sensor pode apresentar mobilidade. Ou seja, após distribuídos sobre uma determinada região, os nós podem se deslocar pelo ambiente com o objetivo de cobrir uma área de monitoração descoberta ou para detectar a presença de algum fenômeno ou objeto. Quando os nós são lançados em uma área pode ocorrer que alguns sejam destruídos e outros estejam defeituosos. Considerando também que a energia (capacidade da bateria) é o fator mais crítico em RSSF, é necessário que um número maior que o necessário de nós sensores sejam distribuídos na área a ser monitorada de forma a garantir cobertura e prolongar o tempo de vida da rede. Em relação a densidade a rede pode estar densa, quando a concentração de nós por área é alta, balanceada, quando o número de nós por área é a mesma em toda a área de monitoração e esparsa, quando a concentração por nós é baixa.

Além da densidade, outros aspectos a serem considerados são a cobertura e a conectividade da rede. A cobertura consiste na habilidade da rede em detectar e observar um elemento na área de monitoramento. A conectividade consiste em definir para cada um dos nós sensores um caminho de menor custo até um dos nós sorvedouros. Maiores detalhes sobre densidade, cobertura e conectividade serão abordados no próximo capítulo.

1.1 Objetivo

O objetivo deste trabalho é desenvolver um modelo de programação linear inteira mista (PLIM) para determinação da configuração inicial de uma RSSF e desenvolver algoritmos que sejam capazes de resolvê-lo em tempo viável para o projetista. Ou seja, o modelo e os algoritmos devem ser projetados de tal forma a definir a densidade da rede e, ao mesmo tempo, garantir a cobertura e conectividade considerando presença e ausência de obstáculos na área de monitoramento. O modelo e algoritmos propostos podem ser inseridos em um ambiente de gerenciamento de RSSFs como proposto por Ruiz em [32].

1.2 Resultados Obtidos

Os principais resultados obtidos com este trabalho foram:

1. Definição dos cenários e aplicações para o trabalho.
2. Desenvolvimento de um modelo de Programação Linear Inteira Mista (PLIM) para definição da densidade da rede, garantindo cobertura e conectividade.
3. Desenvolvimento de algoritmos que permitam a obtenção de soluções de qualidade e em tempo viável.
4. Resultados computacionais considerando a presença e ausência de obstáculos na área de monitoramento.

1.3 Estrutura do Documento

Esta dissertação é constituída de 5 capítulos. O Capítulo 2 apresenta conceitos e classificação das RSSFs, além de listar os trabalhos relacionados. O Capítulo 3 traz o modelo de Programação Linear Inteira Mista. O Capítulo 4 apresenta as técnicas de otimização utilizadas bem como a descrição do algoritmo desenvolvido para resolver o PLIM. O Capítulo 5 apresenta os resultados obtidos bem como as análises. E, finalmente, o capítulo 6 as conclusões e trabalhos futuros.

Capítulo 2

Redes de Sensores Sem Fio

As RSSFs são utilizadas para executar tarefas colaborativas tais como: detecção de eventos, classificação e rastreamento de objetos, além de encontrar aplicações em áreas como produção e monitoramento industrial, segurança, medicina, tráfego, aviação e militar. As RSSFs podem auxiliar no controle de incêndios florestais, pois ao ser estabelecida em uma área de incêndio os nós fontes poderiam disseminar dados como temperatura, pressão barométrica, umidade, luz e velocidade do vento. Estes dados, quando combinados com dados sobre vegetação e topografia do local, poderiam auxiliar na detecção de focos de incêndio e, desta forma, deter o seu avanço.

Para ser utilizada em cada uma das aplicações acima, as RSSFs possuem diversas características. Neste capítulo, serão descritas algumas destas características, bem como os trabalhos relacionados.

2.1 Classificação das RSSFs

Conforme descrito por Loureiro et. al, em [16], o ciclo de vida das RSSFs pode ser caracterizado em cinco fases: configuração, manutenção, sensoriamento, processamento e comunicação. Na fase de configuração ocorre o estabelecimento da rede na região de monitoração. A fase de manutenção, que permanece ativa durante todo o tempo de operação das redes, visa prolongar o tempo de vida da RSSF. O sensoriamento possui a tarefa de observar o ambiente e coletar os dados requisitados pela aplicação. O processamento consiste na fase de análise dos dados coletados pelos nós sensores e também está relacionado ao gerenciamento, comunicação e manutenção da rede. A comunicação abrange as operações de transmissão e recepção de dados.

Dentre as fases citadas podem ser destacadas características que ajudam a classificar as RSSFs. Para este trabalho serão consideradas as características relacionadas a fase de configuração extraídas de Loureiro et. al, em [16] e definidas a seguir.

- Fase de configuração
 - Composição
 - * Homogêneas : todos os nós apresentam as mesmas características de hardware.
 - * Heterogêneas : os nós apresentam dispositivos de comunicação, processamento e sensoriamento diferenciados.

– Organização

- * Planas: redes sem agrupamentos.
- * Hierárquicas: redes com agrupamentos.

– Distribuição

- * Regular: os nós são distribuídos de maneira equidistante na área de monitoramento.
- * Irregular: os nós estão distribuídos de maneira aleatória na área de monitoramento.

– Densidade

- * Balanceada: a concentração de nós por área é a mesma em toda a área de monitoração.
- * Densa: a concentração de nós por área é alta.
- * Esparsa: a concentração de nós por área é baixa.

– Controle

- * Aberta: a rede apenas monitora a região.
- * Fechada: a rede monitora e atua na região.

2.2 Localização em RSSFs

A localização nas RSSFs consiste em se definir a posição geográfica ou uma posição relativa para cada nó sensor que compõem a rede. Para o processo de localização ocorrer alguns nós da rede devem conhecer antecipadamente sua posição. Estes nós são denominados nós âncoras ou beacons. Eles conhecem sua localização porque foram colocados em locais pré-definidos ou possuem um equipamento GPS (Global Position System) acoplado. A informação de localização de nós é importante neste trabalho pois o modelo de otimização desenvolvido, leva em conta a distância entre os nós sensores.

Os trabalhos desenvolvidos por Niculescu e Nath em [28], por Savarese, Langedoen e Rabaey em [33], Savvides, Park e Srivastava em [34] serviram de base para a criação de uma estrutura para algoritmos de localização desenvolvidos por Langendoen e Reijers em [13] que constituem de três fases:

1. Determinação da distância entre nós âncoras e nós que não conhecem sua localização.
2. Determinação da localização dos nós a partir de sua distância para os nós âncoras.
3. Refinamento da localização através da informação de distância e posição dos nós vizinhos.

Entre os métodos utilizados nestas fases estão a determinação da distância baseada no ângulo de chegada (angle of arrival - AOA), no tempo de chegada (time of arrival - TOA), na diferença do tempo de chegada (time difference of arrival - TDOA) e na potência do sinal. São utilizados também no processo métodos de lateração e triangulação. Para cada método é determinado o número mínimo de nós âncoras que os nós sensores devem ter como vizinhos para que seja possível definir a posição de todos os nós sensores. Quanto maior este número, mais precisa será a localização encontrada. Trabalhos relacionados com estes métodos podem ser obtidos em [24] e [25].

2.3 Cobertura em RSSFs

A área de cobertura em RSSF corresponde a área coberta pelo dispositivo de sensoriamento do nó e o seu cálculo está relacionado ao raio de alcance dos nós que se encontram ativos na rede [41, 12]. Em Megerian et al [20], a cobertura é definida como uma medida da habilidade da rede em detectar e observar um elemento na área de monitoramento. Neste trabalho, o cálculo da área de cobertura em uma RSSF foi feito através da discretização da área de monitoramento. Desta forma, procura-se pelo melhor conjunto de nós sensores que devem estar ativos para garantir a cobertura total da área. As figuras abaixo representam o raio de sensoriamento do nó (figura 2.1) e a cobertura (figura 2.2).

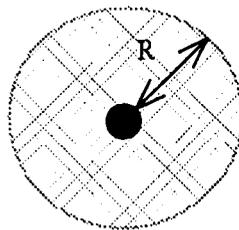


Figura 2.1: Raio de Sensoriamento.

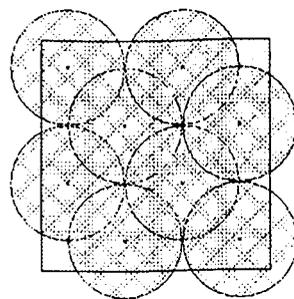


Figura 2.2: Cobertura em RSSFs.

Outro aspecto relativo a cobertura, está relacionado com a disposição dos nós sensores na área. Tal disposição pode ser feita de maneira determinística (nós colocados em posições pré-definidas) e não-determinística (onde os nós são lançados na área). Uma análise da influência desta disposição na cobertura da rede pode ser encontrada em Nakamura [26].

2.4 Densidade nas RSSFs

Tendo em vista o vasto campo de aplicação das RSSFs, bem como os diversos ambientes físicos onde os nós sensores podem estar sendo dispostos, torna-se necessário que um número maior destes elementos seja utilizado afim de garantir cobertura e qualidade de monitoramento. Como consequência, a RSSF se forma mais densa do que o necessário, causando redundância de recursos.

Ter uma alta redundância, embora possa aumentar a precisão dos dados coletados, pode causar efeitos negativos como colisões, interferências e atrasos que impactam o desempenho da rede. Desta forma, possuir recursos que permitam o controle da densidade da rede, constitui de uma ferramenta de extrema necessidade para as RSSFs. A formulação que será proposta no Capítulo 3, procura reduzir os impactos negativos da alta densidade diminuindo ao mínimo o número de nós sensores em atividade, mas comprometendo-se a garantir a qualidade de monitoramento desejada.

2.5 Conectividade em RSSFs

Além de garantir a cobertura, conforme apresentado na seção anterior, outro aspecto a ser considerado neste trabalho consiste na conectividade dos nós sensores. Uma vez que os nós sensores foram dispostos na área de monitoramento e ativados, o próximo passo consiste em receber as informações coletadas por estes nós. Para realizar esta tarefa, além dos nós sensores, é posicionado na área de monitoramento um outro nó conhecido pelo nome de nó sink ou nó sorvedouro. Este nó possui a função de receber as informações coletadas pelos demais nós sensores e enviá-las para uma estação base.

As informações podem chegar a este nó sorvedouro vindas diretamente do nó que coletou a informação, neste caso a comunicação é dita ser (single-hop), ou podem ser enviadas através dos nós sensores ativos até o nó sorvedouro (multi-hop). Assim, o problema de conectividade em RSSF consiste em definir para cada nó sensor ativo uma rota de menor custo que seja capaz de enviar cada pacote de informação gerado pelos nós sensores até um nó sorvedouro. Da mesma forma que para sensoriar, o nó sensor apresenta um raio de sensoriamento, para comunicar, o nó sensor possui um alcance, representado pelo seu raio de comunicação (figura 2.3).

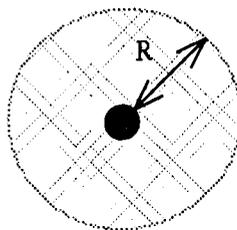


Figura 2.3: Raio de Comunicação.

2.6 Trabalhos Relacionados

O modelo de otimização proposto lidará com os problemas de cobertura, conectividade e controle de densidade de nós em RSSFs. O problema de cobertura está relacionado com a qualidade de monitoramento da área. O problema de conectividade está relacionado à comunicação entre nós e ao fluxo de dados na rede. Por último, o controle da densidade determina quais nós sensores devem estar ativos e que funções estão realizando.

Megerian e Potkonjak em [21] propõem diversos modelos de Programação Linear Inteira (PLI) para resolver o problema de cobertura em RSSFs. Esta abordagem é similar à proposta, porém não lida com o problema da conectividade e roteamento.

Chakrabarty et al. em [3] apresenta um modelo de programação linear inteira que minimiza o custo de posicionamento de nós sensores heterogêneos e garante a cobertura da área de monitoramento. A abordagem também é semelhante, embora não lida com com os problemas de conectividade e não considera o posicionamento aleatório dos nós sensores.

Outro trabalho relacionado com o problema de cobertura é apresentado por Vieira et al. em [40]. Neste trabalho, é proposto um planejamento ou scheduling dos nós sensores através de algoritmos baseados nos diagramas de Voronoi [2].

O problema de cobertura é definido em [12] por Huang e Tseng como uma determinação de quão bem monitorada está a área. Neste trabalho o problema foi formulado como um problema de decisão, cujo objetivo é determinar se cada ponto na área de monitoramento está coberto por pelo menos k nós sensores, porém a conectividade também não é um ponto de estudo. Outro trabalho relacionado com cobertura é apresentado por Li, Wan e Fireder em [15]. Neste trabalho é proposto um algoritmo distribuído utilizando um grafo da vizinhança do nó para resolver o problema de cobertura definido em [22] por Meguerdichian et al.

A relação entre cobertura e conectividade é apresentada por Wang et. al. em [41]. Este trabalho explora a economia de energia da rede mantendo a cobertura e conectividade. A abordagem utilizada consiste de uma análise geométrica das áreas cobertas e do estudo da relação entre os raios de sensoriamento e comunicação. De acordo com o trabalho, a solução do problema de cobertura, permite a conectividade quando o raio de sensoriamento é menor que a metade do raio de comunicação. Em Nakamura [26] e Oliveira [29] são propostas formulações semelhantes a deste trabalho. Nakamura propõe um modelo para o planejamento dinâmico de RSSFs considerando cobertura e conectividade. Já Oliveira, fornece um modelo e algoritmos baseados na Relaxação Lagrangeana para organização de RSSFs hierárquicas.

Xu et. al em [42] apresentam um protocolo de controle de topologia para redes de sensores planas e densas que visam prolongar o tempo de vida da rede, mantendo a conectividade entre os nós. A abordagem do trabalho é reconhecer nós redundantes e desligá-los. Este protocolo foi denominado como GAF (Geographic Adaptive Fidelity). O protocolo assume que todos os nós conhecem sua localização e utiliza um modelo de rádio idealizado.

Em otimização combinatória, outros trabalhos correlatos são: problema da localização de postes de iluminação pública, em que os locais onde se deseja providenciar iluminação são representados por pontos que devem ser cobertos. Problema da localização de radares de vigilância, a alocação arma x alvo, a exploração de petróleo em poços submarinos e o clássico problema da Galeria de Arte. Neste problema, procura-

se pelo número mínimo de observadores (guardas de segurança) necessários para cobrir uma sala de uma galeria de arte de forma que cada ponto da sala seja visto por pelo menos um observador. Estes são alguns exemplos de problemas em que a formulação matemática é similar a que será proposta neste trabalho. Um estudo mais detalhado destes problemas, bem como algoritmos de solução podem ser obtidos em [17], [9] e [18].

2.7 Considerações Finais

Este capítulo apresentou de maneira sucinta algumas das características e configurações possíveis para as RSSFs. Aspectos relacionados com a localização dos nós sensores, cobertura e conectividade foram expostos, bem como os trabalhos relacionados. No capítulo seguinte, será feita uma definição do problema, bem como será proposto um modelo de programação linear inteira mista capaz de representar os problemas de definição da densidade, cobertura e conectividade para as RSSFs.

Capítulo 3

Formulação do Problema

Um dos grandes desafios no planejamento das RSSFs, consiste no controle da densidade dos nós sensores. Ou seja, de um determinado conjunto de nós sensores dispostos em uma área, qual é o menor número de sensores que devem estar ativos para garantir a cobertura e a conectividade. Este capítulo apresenta a definição do problema, bem como uma formulação matemática para o mesmo.

3.1 Definição do Problema

O problema em estudo neste trabalho pode ser definido como: Seja uma área de sensoriamento onde os sensores foram distribuídos de maneira aleatória (simulando um lançamento feito por aeronave). Quais dos nós sensores lançados deverão estar ativos de forma a cobrir toda a área e qual a rota cada pacote de informação deverá percorrer para alcançar um dos nós sorvedouros. O cenário adotado para este estudo constitui de uma rede plana, homogênea, nós estacionários, área de monitoração sem obstáculos e 2D. Durante o restante deste trabalho, este problema será tratado como PDCC-RSSF (Problema de Densidade, Cobertura e Conectividade em Rede de Sensores Sem Fio).

Para garantir que toda a área estará coberta, utiliza-se o conceito de ponto de demanda. É feita uma discretização da área a ser sensoriada e procura-se pelo melhor conjunto de nós sensores que devem estar ativos para garantir que todos os pontos de demandas estarão cobertos. Conforme já apresentado no capítulo anterior, existem diversos trabalhos tratando o problema de cobertura como em [3], [22], [37]. Entretanto, trabalhos que apresentam uma formulação em programação linear inteira mista e a discretização da área em pontos de demanda, solucionando ao mesmo tempo o problema de densidade, cobertura e conectividade não foram encontrados.

Uma definição mais formal é a seguinte: Seja uma área de sensoriamento A , um conjunto de pontos de demanda D dispostos em A , um conjunto de nós sensores S , um conjunto de nós sorvedouros M e um parâmetro de cobertura m . O problema consiste em garantir que pelo menos m nós sensores em S estão cobrindo cada ponto de demanda em D na área de sensoriamento A e que existe uma rota para cada nó sensor a um dos nós sorvedouros. As Figuras (3.1), (3.2) e (3.3) representam respectivamente os pontos de demanda, os nós sorvedouros e os nós sensores.

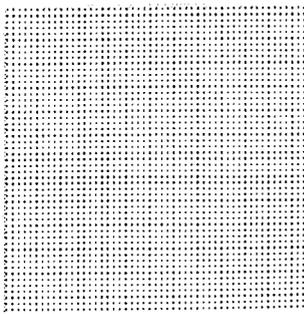


Figura 3.1: Pontos

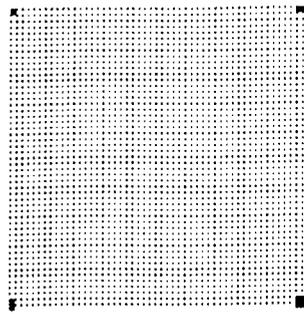


Figura 3.2: Nós Sorvedouros

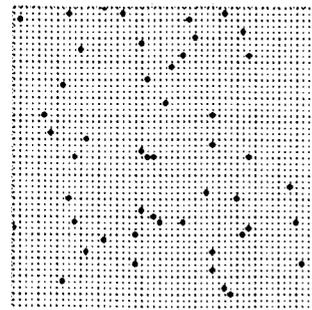


Figura 3.3: Nós Sensores

3.2 Modelo de Programação Matemática

O problema apresentado na seção anterior foi formulado de forma a ser possível a utilização de técnicas de programação linear Inteira. A solução para o modelo que será apresentado abaixo, fornece um conjunto de nós sensores a serem ativados de forma a garantir a total cobertura da área e, ao mesmo tempo, apresenta um caminho entre os sensores ativos e os nós sorvedouros.

3.2.1 Formulação

O modelo pode ser representado através de um grafo direcionado $G = (N, A)$ onde:

N conjunto de todos os nós

A conjunto de todos os arcos entre nós que estão no raio de comunicação dos sensores.

A partir dos conjuntos acima, temos os seguintes sub-conjuntos de nós:

N^s sub-conjunto de nós que são sensores

N^m sub-conjunto de nós que são sorvedouros

N^d sub-conjunto de nós que são pontos de demanda

Sejam os seguintes sub-conjuntos de arcos:

A^c Conjunto de arcos que conectam sensores a pontos de demanda

A^s Conjunto de arcos que conectam sensores com outros sensores

A^m Conjunto de arcos que conectam sensores com os nós sorvedouros

$E_j(A^s)$ Conjunto de arcos $(i, j) \in A^s$ que entram em um nó sensor $j \in N^s$

$S_j(A^s)$ Conjunto de arcos $(j, k) \in A^s$ que saem de um nó sensor $j \in N^s$

Seja a seguinte matriz:

D matriz de energia onde cada d_{ij} representa a energia de comunicação entre o nó sensor i e o nó sensor ou nó sorvedouro j . Essa energia depende da distância entre os nós.

Sejam os parâmetros:

c_i energia para ativar e manter o nó sensor i .

m parâmetro de cobertura que define o número de nós sensores que devem cobrir cada ponto de demanda.

T^1 número de pontos de demanda.

T^2 número de nós sensores.

Além dos parâmetros acima, o modelo utiliza as seguintes variáveis:

x_{ij} variáveis que possuem o valor 1 se o nó sensor i cobre o ponto de demanda j e 0 caso contrário

z_{ij}^l variáveis que possuem o valor 1 se o arco $(ij) \in (A^s \cup A^m)$ está no caminho entre o nó sensor $l \in N^s$ e um nó sorvedouro e 0 caso contrário.

t_i é igual a 1 se o nó sensor i está ativo e 0 caso contrário.

h_j variável que representa a não cobertura de um ponto de demanda j .

3.2.2 Flexibilidade

A formulação que será proposta, além de fornecer o número mínimo de nós sensores para cobrir a área e garantir a conectividade, permite representar diversas características das redes de sensores: presença ou ausência de obstáculos, redes homogêneas ou heterogêneas, lançamento aleatório ou em posições específicas.

No caso de existirem obstáculos, pode ser feita uma modificação no conjunto de arcos A^d , A^s e A^m de forma que só haverá um arco entre os nós sensores e pontos de demanda, entre os próprios nós sensores e entre nós sensores e nós sorvedouros que não estão nas áreas de influência ou cobertura de cada nó devido a estes obstáculos. Para representar uma rede heterogênea, é necessário apenas alterar os parâmetros de raio de comunicação e sensoriamento para cada nó sensor. A distribuição dos nós na área (determinística ou não-determinística) também não afeta o modelo de otimização proposto, mas afeta a solução obviamente.

Desta forma, ao combinar as características acima, é possível através do modelo obter soluções para diversas aplicações de RSSFs. Podem ser consideradas coleta de dados contínua ou discreta e a mobilidade dos nós sensores, isto facilita a utilização do modelo e dos algoritmos em aplicações como medição de temperatura, detecção de objetos, medição de substâncias químicas, entre outras.

Assim, o modelo de programação linear inteiro misto que representa o PDCC-RSSF pode ser definido como:

$$\min \sum_{l \in N^s} \sum_{(i,j) \in A^s \cup A^m} d_{ij} z_{ij}^l + \sum_{l \in N^s} c_l t_l \quad (3.1)$$

sujeito a:

$$\sum_{(l,j) \in A^c} x_{lj} \geq m \quad \forall j \in N^d \quad (3.2)$$

$$\sum_{j \in N^d} x_{lj} \leq T^1 t_l, \quad \forall l \in N^s \quad (3.3)$$

$$\sum_{j \in N^s \cup N^m} z_{ij}^l \leq T^2 t_i, \quad \forall l \in N^s, \forall i \in N^s \quad (3.4)$$

$$\sum_{i \in N^s} z_{ij}^l \leq T^2 t_j, \quad \forall l \in N^s, \forall j \in N^s \quad (3.5)$$

$$\sum_{(ij) \in E_j^s(A^s)} z_{ij}^l - \sum_{(jk) \in S_j^s(A^s \cup A^m)} z_{jk}^l = 0 \quad \forall j \in (N^s - l), \forall l \in N^s \quad (3.6)$$

$$\sum_{(ij) \in E_j^s(A^s)} z_{ij}^l - \sum_{(jk) \in S_j^s(A^s \cup A^m)} z_{jk}^l = -t_l \quad j = l, \forall l \in N^s \quad (3.7)$$

$$\sum_{l \in N^s} t_l \geq A/\pi r^2 \quad (3.8)$$

$$0 \leq x \leq 1 \quad (3.9)$$

$$z, t \in \{0, 1\} \quad (3.10)$$

A função Objetivo (3.1) minimiza a energia para transmissão entre os nós sensores e sorvedouros e o gasto de energia em ativar e manter um nó sensor. O conjunto de restrições (3.2) garante que pelo menos m sensores precisam estar cobrindo cada ponto de demanda. As restrições (3.3) definem que um nó que está inativo para sensoriamento não deve estar atendendo um ponto de demanda. As restrições (3.4) e (3.5) especificam que o fluxo de informação só é possível entre os nós que estão ativos. Já as restrições (3.6) garantem a conservação de fluxo entre cada sensor ativo e o sorvedouro, (3.7) representam a origem do roteamento do nó l para o nó sorvedouro. As restrições (3.8) são redundantes e garantem um número mínimo de nós sensores ativos para cobrir a área. Finalmente, as restrições (3.9) definem as variáveis de cobertura como contínuas entre 0 e 1, uma vez que um ponto de demanda pode estar sendo atendido por mais de um nó sensor. Cabe ressaltar que uma variável x_{lj} diferente de zero, já é uma garantia que o ponto de demanda j é coberto pelo nó sensor l . Já as restrições (3.10), definem as demais variáveis como binárias. A solução ótima para o modelo define um conjunto de nós sensores que devem estar ativos (variáveis t_l), garante a cobertura total da área (variáveis x_{lj}) e também o caminho entre um nó sensor ativo e o nó sorvedouro (z_{ij}^l).

Em alguns casos, dependendo da disposição dos nós sensores, pode não ser possível garantir a cobertura total (conforme as restrições (3.2) especificam). Para contornar este problema, as restrições (3.2) podem ser alteradas para:

$$\sum_{(l,j) \in A^c} x_{lj} + M h_j \geq m \quad \forall j \in N^d \quad (3.11)$$

bem como a função objetivo para:

$$\min \sum_{l \in N^s} \sum_{(i,j) \in A^s \cup A^m} d_{ij}^z z_{ij}^l + \sum_{l \in N^s} c_l t_l + \sum_{j \in N^d} h_j \quad (3.12)$$

Desta forma, é aplicado uma penalidade M na função Objetivo para cada ponto de demanda não coberto. Cabe ressaltar que um ponto só não será coberto se não houver nenhum nó sensor na região de alcance. A Figura (3.4) representa uma solução para o PDCC-RSSF. As linhas ligando os pontos representam as rotas que cada pacote de informação deve percorrer para chegar a um dos nós sorvedouros e os círculos representam a área de sensoriamento coberta por cada nó sensor ativo.

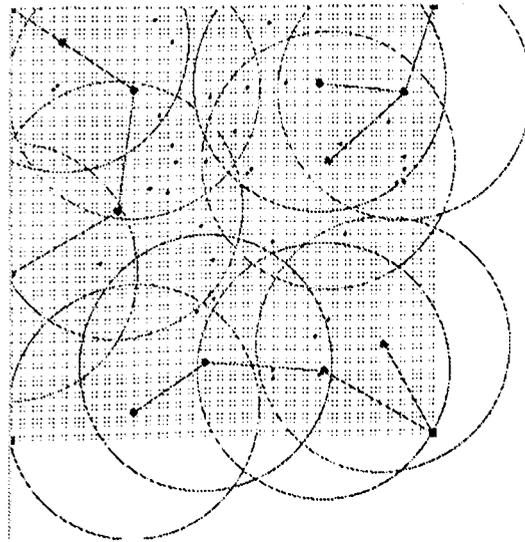


Figura 3.4: Solução para o PDCC-RSSF.

3.3 Complexidade

Problemas de Otimização Inteira e Mista como o PDCC-RSSF são normalmente difíceis de resolver. Obter soluções ótimas e em um tempo considerado aceitável não é uma tarefa fácil mesmo para pequenas instâncias, como será visto no capítulo que trata dos experimentos computacionais.

Uma das maneiras de analisar a complexidade em otimização combinatória consiste na comparação com outros problemas e modelos. Através de uma análise mais profunda dos modelos, é possível identificar semelhanças com formulações existentes na literatura que já são conhecidamente difíceis de resolver. Nesta seção, será feita uma comparação da formulação proposta para o PDCC-RSSF com outra existente na literatura com o objetivo de destacar a complexidade envolvida em obter soluções para o PDCC-RSSF.

3.3.1 Problemas de Recobrimento

Os problemas de Recobrimento (PR) (set covering problem) são problemas de programação linear e inteira que se enquadram na classe dos mais difíceis problemas de otimização combinatória existentes [17]. É possível formular o PR como um problema de programação linear e inteira da seguinte forma:

$$\min \sum_{l=1}^n c_l t_l \quad (3.13)$$

sujeito a:

$$\sum_{l=1}^n a_{jl} t_l \geq 1 \quad j = 1, \dots, m \quad (3.14)$$

$$t_l \in \{0, 1\}, \forall l \quad (3.15)$$

onde:

- A é a matriz de cobertura que define quais elementos podem estar cobrindo cada ponto.
- t_l é a variável binária que assume o valor 1 quando a coluna l de A está na solução e 0, caso contrário.
- c_l é o custo unitário associado a variável t_l .

A função Objetivo (3.13) procura minimizar o custo de cobertura, enquanto as restrições (3.14) garantem que pelo menos uma coluna l está cobrindo cada linha j .

É possível notar a similaridade do PR com um dos sub-problemas do PDCC-RSSF que corresponde ao problema de garantir um número mínimo de nós sensores ativos para cobrir a área. Os custos fixos c_l nos dois problemas representam exatamente um custo associado a cobertura da área. As restrições (3.14) em (PR) e (3.2) em (PDCC-RSSF) estão relacionadas com a garantia de que os pontos de demanda representados por j em ambos os modelos estarão sendo cobertos.

Outro problema relacionado, é o problema da Galeria de Arte. Neste problema, procura-se pelo número mínimo de observadores (guardas de segurança) necessários para cobrir uma sala de uma galeria de arte de forma que cada ponto da sala seja visto por pelo menos um observador. A demonstração de que este problema é NP-Completo pode ser obtida em Garey e Johnson [18]. Assim, é possível concluir com relação a complexidade, que obter soluções para o PDCC-RSSF é pelo menos tão difícil quanto obter soluções para o PR e para o problema da Galeria de Arte.

3.4 Considerações Finais

Este capítulo tratou da definição do problema em estudo neste trabalho: Problema de Densidade, Cobertura e Conectividade em Rede de Sensores sem Fio (PDCC-RSSF), bem como foi apresentado um modelo de programação linear inteira mista (PLIM) para o mesmo. No capítulo seguinte, serão expostos os algoritmos e a técnica de otimização utilizada para resolver o PDCC-RSSF.

Capítulo 4

Algoritmo de Solução

Conforme apresentado no Capítulo 3, encontrar boas soluções para problemas de otimização combinatória não é uma tarefa trivial. Soluções de qualidade e computacionalmente viáveis, que sejam obtidas em no máximo algumas horas de processamento não são fáceis de se obter. Desta forma, cabe ao projetista de algoritmos um estudo mais detalhado do modelo de otimização e das técnicas existentes para procurar encontrar soluções que, se não sejam ótimas para o problema, estejam o mais perto possível do ótimo e sejam computacionalmente viáveis. No restante deste capítulo, serão apresentados uma técnica e os algoritmos utilizados para resolver o PDCC-RSSF.

4.1 Apresentação da Técnica

Encontrar soluções para problemas de otimização combinatória como o PDCC-RSSF podem ser obtidas através da seguinte estratégia: obtenção de um limite superior que esteja o mais próximo possível do ótimo. Este limite superior deve ser uma solução viável para o problema e pode ser obtido através da utilização de heurísticas como algoritmos genéticos, grasp, simulated annealing, busca tabu. E da obtenção de um limite inferior que também deve estar o mais próximo possível da solução ótima. Para obtenção do limite inferior, a técnica mais utilizada é a relaxação linear. Nesta técnica, as restrições de integralidade são relaxadas e o modelo de programação mista ou inteira é resolvido como um modelo de programação linear (PL). O modelo relaxado pode ser resolvido através de algoritmos como simplex e pontos interiores. A solução obtida é o limite inferior para o modelo original.

Outra forma de obter um limite inferior é a relaxação Lagrangeana. Esta técnica tem sido utilizada com grande sucesso na obtenção de limites para uma grande variedade de problemas de otimização combinatória ([7], [27] e [6]). A relaxação Lagrangeana consiste resumidamente em relaxar as restrições complicadas do problema e penalizar a violação destas restrições adicionando-as à função objetivo com o uso de multiplicadores. A solução do problema relaxado, conhecido como lagrangeano, fornece um limite inferior para o problema de programação inteira ou mista.

Enquanto um bom limite superior fornece uma solução viável para o problema, um bom limite inferior representa uma forma de avaliar a qualidade da solução obtida no limite superior. Para um dado limite superior Z_{LS} e um dado limite inferior Z_{LI} , uma forma muito utilizada para avaliar a qualidade da solução Z_{LS} é dado por $\alpha = 100(Z_{LS} - Z_{LI})/Z_{LS}$, que mede em termos percentuais a distância relativa entre os limites superior e inferior. Quando os limites se igualam ($Z_{LS} = Z_{LI}$, $\alpha = 0$) e a solução

encontrada pelo limite superior é ótima. Quando ($Z_{LS} > Z_{LI}$), $\alpha > 0$ e a melhor solução encontrada esta a menos de $\alpha\%$ do valor ótimo. As seções seguintes irão apresentar os algoritmos utilizados para obtenção dos limites superior e inferior, bem como o método utilizado para resolver o problema Lagrangeano.

4.2 Obtenção do Limite Inferior

Muitos problemas difíceis de programação inteira e mista são formados por um subproblema (ou um conjunto de subproblemas) relativamente fácil (fáceis), que é (são) complicado(s) por um conjunto de restrições complicantes. A relaxação lagrangeana, surgiu da observação destas características. O que a relaxação Lagrangeana propõe é relaxar este conjunto de restrições complicantes do problema de forma que o problema relaxado seja formado por um problema fácil, ou possa ser decomposto em um conjunto de subproblemas fáceis. A cada restrição relaxada é associado um multiplicador de Lagrange de forma que as restrições relaxadas possam ser inseridas na função objetivo. Para a obtenção de bons limites para o problema torna-se necessário resolver duas questões estratégicas.

A primeira questão refere-se a quais restrições devem ser relaxadas. A escolha destas restrições deve ser feita de tal maneira a obter subproblemas que sejam fáceis de se resolver. A relaxação Lagrangeana sempre fornece um limite inferior que é maior ou igual ao limite inferior da relaxação linear. Durante o desenvolvimento deste trabalho, várias escolhas de restrições foram realizadas. Enquanto em algumas os subproblemas eram muito fáceis e não geravam bons limites, em outras, o subproblema obtido mostrou-se ser extremamente complexo de se resolver, o que invalidava a sua utilização. Após vários testes no modelo, foram relaxadas o conjunto de restrições (3.2).

A segunda questão é como ajustar os multiplicadores de Lagrange de forma que o problema relaxado forneça o maior limite inferior possível. O problema de calcular os multiplicadores de Lagrange que maximizem o problema relaxado é chamado de Lagrangeano dual. Existem diversas abordagens que podem ser aplicadas para resolver o Lagrangeano dual, tais como: método de subgradiente [36], método de plano de corte [4], método de centros analíticos [8] e método de feixe (ou bundle) [11]. Cada um desses métodos possui vantagens e desvantagens que estão relacionadas com o problema a ser resolvido. Pode-se encontrar em [14] uma descrição de cada um deles. Neste trabalho a abordagem utilizada foi o método de subgradiente. Esta escolha foi feita devido a sua simplicidade de implementação e do seu sucesso na solução de diversos problemas como [19], [30] e [6].

4.2.1 Lagrangeano

Após analisar diversas relaxações, foram escolhidas para relaxar as restrições (3.2). Estas restrições estão relacionadas com o problema de garantir que toda a área estará coberta. Ao relaxar estas restrições, os subproblemas obtidos tornaram-se fáceis de se resolver. Um dos subproblemas pode ser resolvido por inspeção e o outro, através de um algoritmo de caminho mínimo. Seja o modelo apresentado no capítulo anterior. Relaxando as restrições (3.2) e associando multiplicadores $\alpha_j, \forall j \in N^d$ o problema relaxado pode ser definido como:

$$Z_{RL}(\alpha) = \min \sum_{l \in N^s} (c_l t_l + \sum_{(i,j) \in A^s \cup A^m} d_{ij} z_{ij}^l - \sum_{j \in N^d} \alpha_j x_{lj}) + \sum_{j \in N^d} m \alpha_j \quad (4.1)$$

sujeito a:

restrições 3.3 a 3.10

Para qualquer conjunto de multiplicadores não negativos α_j , a solução Z_{RL} do problema relaxado RL fornece um limite inferior para a solução do modelo original que representa o PDCC-RSSF. Reorganizando os termos da função objetivo o modelo Z_{RL} torna-se separável para cada l , podendo ser reescrito como:

$$Z_{RL}(\alpha) = \min \sum_{l \in N^s} [P_{zl}(\alpha) + P_{zl}] t_l + \sum_{j \in N^d} m \alpha_j \quad (4.2)$$

sujeito a:

$$\sum_{l \in N^s} t_l \geq A/\pi r^2 \quad (4.3)$$

$$t_l \in \{0, 1\}, \forall l \in N^s \quad (4.4)$$

onde o subproblema P_{zl} pode ser representado como:

$$P_{zl}(\alpha) = \min \sum_{j \in N^d} -\alpha_j x_{lj} \quad (4.5)$$

sujeito a:

$$\sum_{j \in N^d} x_{lj} \leq T^1 \quad (4.6)$$

$$0 \leq x_{lj} \leq 1 \quad (4.7)$$

O subproblema P_{zl} pode ser resolvido por inspeção da seguinte forma:

- para cada $l \in N^s$ faça: $x_{lj} = 1$ se $-\alpha_j < 0, \forall j \in N^d$

O subproblema P_{zl} é representado através do seguinte modelo:

$$P_{zl} = \min c_l + \sum_{i \in N^s - l} c_i t_i + \sum_{(i,j) \in A^s \cup A^m} d_{ij} z_{ij}^l \quad (4.8)$$

sujeito a:

restrições 3.4, 3.5, 3.6, 3.7 e 3.10.

O subproblema P_{zl} pode ser resolvido obtendo o caminho mínimo para cada l da seguinte forma:

- para cada $l \in N^s$, encontrar o caminho mínimo de l aos sorvedouros considerando os custos:
 - $d_{ij} \forall (i, j) \in A^s \cup A^m$ e t_i em cada nó $i \in (N^s - l)$.

- para isto, desdobre o nó i em nós i e i' , fazendo $d_{ii'} = c_i$ e $z_{ii'}^l = t_i$, e fazendo $A^{s'} = A^s \cup (i, i')$, $N^{s'} = N^s \cup (i')$, $\forall i \in N^s$.

Desta forma, P_{zi} pode ser representado como: $P_{zi} = c_i + C^l$, onde C^l é o custo do caminho mínimo de l a um sorvedouro. Logo, o valor do limite inferior para Z_{RL} pode ser calculado da seguinte forma: seja $f_l = P_{zi}(\alpha) + c_i + C^l$ e o seguinte modelo:

$$Z_{RL}(\alpha) = \min \quad \sum_{l \in N^s} f_l t_l + \sum_{j \in N^d} m \alpha_j \quad (4.9)$$

sujeito a:

$$\sum_{l \in N^s} t_l \geq A/\pi r^2 \quad (4.10)$$

$$t_l \in \{0, 1\}, \forall l \in N^s \quad (4.11)$$

Obtem-se o valor do limite inferior da seguinte forma:

- Selecione os sensores $l \in N^s$ em ordem crescente de f_l .
 - Se $f_l < 0$ então $t_l = 1$;
 - se $f_l \geq 0$ faça $t_l = 1$ até que a restrição $\sum_{l \in N^s} t_l \geq A/\pi r^2$ seja atendida.

4.2.2 Lagrangeano Dual

Conforme mostrado anteriormente, para qualquer conjunto de multiplicadores de Lagrange α_j não negativos, o Lagrangeano fornece uma solução Z_{RL} que é sempre igual ou inferior à solução do problema PDCC-RSSF. Portanto, quanto maior for a solução Z_{RL} , melhor será o limite inferior para o problema PDCC-RSSF. O problema de encontrar o conjunto de multiplicadores de Lagrange que maximizem a solução Z_{RL} do Lagrangeano é chamado de Lagrangeano dual (LD), que é definido como:

$$Z_{LD} = \max \quad Z_{RL}(\alpha) \quad (4.12)$$

sujeito a:

$$\alpha \geq 0 \quad (4.13)$$

O Lagrangeano dual é um problema de otimização não diferenciável, onde a função Z_{LD} é uma função côncava e linear por partes. Conforme apresentado anteriormente, existem diversos algoritmos para obter a solução do Z_{LD} tais como: método de subgradiente, método de plano de corte, método de centros analíticos e método de feixe (ou bundle). Destes, será utilizado o método do sub-gradiente. O método do sub-gradiente consiste em iterativamente ajustar os multiplicadores de Lagrange, sendo que na n -ésima iteração os multiplicadores são calculados por:

$$\alpha_j^{n+1} = \text{Max}(0, \alpha_j^n + p^n g_{\alpha_j}^n(x^n)), \forall j \in N^d \quad (4.14)$$

onde $g_{\alpha_j}^n$ é o vetor sub-gradiente na iteração n , com respeito a α . Este vetor é definido como:

$$g_{\alpha_j}^n(x^n) = m - \sum_{(l,j) \in A^c} x_{lj}, \forall j \in N^d \quad (4.15)$$

O parâmetro p^n , representa o tamanho do passo a ser tomado na direção do sub-gradiente. Este passo é definido por:

$$p^n = \pi \frac{1.05Z_{LS} - Z_{RL}(\alpha^n)}{\|g_{\alpha}^n\|^2} \quad (4.16)$$

onde π é um escalar, definido no intervalo $0 < \pi \leq 2$, que é utilizado para regular o tamanho do passo, e portanto, a velocidade de convergência do método. Z_{LS} representa o valor do melhor limite superior conhecido para o problema PDCC-RSSF, o qual pode ser obtido a partir de alguma heurística. Seu valor é multiplicado pela constante 1.05 a fim de acelerar a convergência do algoritmo quando Z_{LS} se aproxima de Z_{LI} . $Z_{RL}(\alpha^n)$ fornece o valor da solução ótima do problema relaxado na iteração n , e a função $\|g\|$ define a norma euclidiana do vetor. Desta forma, o procedimento utilizado para implementação do método, pode ser organizado da seguinte forma:

1. Inicialize parâmetros e variáveis:

- Faça $\pi = 2$.
- Faça $n = 0$.
- Faça $N = 0$ (número de iterações consecutivas sem que haja melhora no limite inferior Z_{LI}).
- Inicialize o limite superior Z_{LS} com uma heurística e o limite inferior Z_{LI} com um valor suficientemente pequeno.
- Inicialize os multiplicadores de Lagrange $\alpha_j = 0, \forall j \in N^d$

2. Faça $n = n + 1$, e resolva o Lagrangeano $Z_{RL}(\alpha^n)$ para os valores correntes dos multiplicadores, de forma a obter a solução (x^n, t^n, z^n) de valor Z_{RL} .

3. Se $Z_{RL} > Z_{LI}$:

- Atualize o limite inferior ($Z_{LI} = Z_{RL}$)
- Utilize uma heurística que tente obter uma nova solução viável para o PDCC-RSSF a partir da solução fornecida pelo problema relaxado (x^n, t^n, z^n) . Se uma solução viável for encontrada, atualize o limite superior se este for melhorado.
- se $Z_{LI} = Z_{LS}$, a solução Z_{LS} é a solução ótima para o PDCC-RSSF, e o procedimento pode ser finalizado.
caso contrário,
- Faça $N = N + 1$.
- Se o Z_{LI} não é reduzido há 200 iterações consecutivas, ou seja, se $N = 200$:
 - Faça $N = 0$;
 - Faça $\pi = \pi/2$
 - Se $\pi \leq 1 \times 10^{-03}$, conclua o procedimento.

$$g_{\alpha_j}^n(x^n) = m - \sum_{(l,j) \in A^c} x_{lj}, \forall j \in N^d \quad (4.15)$$

O parâmetro p^n , representa o tamanho do passo a ser tomado na direção do sub-gradiente. Este passo é definido por:

$$p^n = \pi \frac{1.05Z_{LS} - Z_{RL}(\alpha^n)}{\|g_a^n\|^2} \quad (4.16)$$

onde π é um escalar, definido no intervalo $0 < \pi \leq 2$, que é utilizado para regular o tamanho do passo, e portanto, a velocidade de convergência do método. Z_{LS} representa o valor do melhor limite superior conhecido para o problema PDCC-RSSF, o qual pode ser obtido a partir de alguma heurística. Seu valor é multiplicado pela constante 1.05 a fim de acelerar a convergência do algoritmo quando Z_{LS} se aproxima de Z_{LI} . $Z_{RL}(\alpha^n)$ fornece o valor da solução ótima do problema relaxado na iteração n , e a função $\|g\|$ define a norma euclidiana do vetor. Desta forma, o procedimento utilizado para implementação do método, pode ser organizado da seguinte forma:

1. Inicialize parâmetros e variáveis:

- Faça $\pi = 2$.
- Faça $n = 0$.
- Faça $N = 0$ (número de iterações consecutivas sem que haja melhora no limite inferior Z_{LI}).
- Inicialize o limite superior Z_{LS} com uma heurística e o limite inferior Z_{LI} com um valor suficientemente pequeno.
- Inicialize os multiplicadores de Lagrange $\alpha_j = 0, \forall j \in N^d$

2. Faça $n = n + 1$, e resolva o Lagrangeano $Z_{RL}(\alpha^n)$ para os valores correntes dos multiplicadores, de forma a obter a solução (x^n, t^n, z^n) de valor Z_{RL} .

3. Se $Z_{RL} > Z_{LI}$:

- Atualize o limite inferior ($Z_{LI} = Z_{RL}$)
- Utilize uma heurística que tente obter uma nova solução viável para o PDCC-RSSF a partir da solução fornecida pelo problema relaxado (x^n, t^n, z^n) . Se uma solução viável for encontrada, atualize o limite superior se este for melhorado.
- se $Z_{LI} = Z_{LS}$, a solução Z_{LS} é a solução ótima para o PDCC-RSSF, e o procedimento pode ser finalizado.
caso contrário,
- Faça $N = N + 1$.
- Se o Z_{LI} não é reduzido há 200 iterações consecutivas, ou seja, se $N = 200$:
 - Faça $N = 0$;
 - Faça $\pi = \pi/2$
 - Se $\pi \leq 1 \times 10^{-03}$, conclua o procedimento.

4. Calcule os vetores de sub-gradiente g_α^n a partir das equações (4.15)
5. Calcule o valor do passo p^n a partir da equação (4.16).
6. Atualize os multiplicadores de Lagrange α^{n+1} e volte ao passo 2.

4.3 Limite Superior

Conforme apresentado na seção anterior, o procedimento para o cálculo do limite inferior necessita a cada iteração de uma solução viável. Este limite superior pode ser obtido de duas formas: utilizar heurísticas como grasp, algoritmos genéticos ou métodos gulosos que sejam capazes de gerar soluções viáveis para o PDCC-RSSF, ou através de uma heurística que procure, a partir da solução fornecida pelo limite inferior, gerar soluções viáveis para o PDCC-RSSF. Esta heurística é conhecida na literatura pelo nome de heurística Lagrangeana, uma vez que sua solução baseia-se em informações obtidas através da relaxação Lagrangeana.

Neste trabalho foram adotadas as duas estratégias para obtenção de um limite superior para o PDCC-RSSF. O limite superior inicial, foi obtido através de uma heurística gulosa e, durante o processo iterativo do método subgradiente, foi utilizada uma heurística que procura viabilizar as soluções obtidas pelo limite inferior fornecido pela relaxação Lagrangeana. As próximas seções irão descrever cada uma das heurísticas utilizadas para obtenção do limite superior.

4.3.1 Limite Superior Inicial

Como solução inicial para o PDCC-RSSF foi utilizado um método guloso que procura inicialmente atender as restrições de cobertura e, em seguida, as restrições de caminho referentes a formulação do PDCC-RSSF. A heurística inicial pode ser dividida em duas fases: na primeira fase, procura-se resolver o problema de cobertura e, na segunda, o problema de conectividade. Abaixo segue o pseudo-código da heurística.

Definindo os seguintes conjuntos e variáveis:

S^d nós sensores desligados.

S^a nós sensores ativos.

P conjunto de pontos de demanda em que se conhece pelo menos $m + 1$ nós sensores que podem ser ativos para cobertura.

c_i energia necessária para ativar o nó sensor $i \in N^s$.

C_i custo do menor caminho de i ao sorvedouro mais próximo, considerando os custos d_{ij}^z , $\forall (i, j) \in (A^s \cup A^m)$ e c_l , $\forall l \in N^s$.

A_i número de pontos de demanda que podem ser cobertos pelo nó sensor i .

D_j número de sensores ativos cobrindo o ponto de demanda j .

uma solução para garantir a cobertura pode ser obtida através do seguinte algoritmo:

1. Inicialize variáveis e conjuntos com:

- $S^a = \emptyset$;
- $S^d = N^s$;
- $D_j = \emptyset$;
- $Iteracoes = 1$.

2. Enquanto $P \neq \emptyset$ e $Iteracoes \leq N^s$ faça:

- (a) Selecione $i \in S^d$, o nó sensor de menor custo c'_i calculado da seguinte forma:

$$c'_i = \frac{c_i + C_i}{A_i}$$
- (b) $S^a = S^a \cup i$
- (c) $S^d = S^d - i$
- (d) Se i cobre j então faça $D_j = D_j + 1$, $\forall j \in N^d$
- (e) Se $D_j \geq m$ e $j \in P$ (cobertura já foi garantida), $\forall j \in N^d$ faça:
 - i. $P = P - j$
- (f) $Iteracoes = Iteracoes + 1$

Para os pontos de demanda que não pertencem ao conjunto P , basta ativar todos os sensores disponíveis para aquele ponto e, $\forall j \in P$, caso em que o parâmetro m não foi atendido, inserir a variável de penalidade h_j . Desta forma, considerando o modelo matemático proposto para o PDCC-RSSF (ver Capítulo 3), temos até o momento as seguintes restrições atendidas: 3.2 e 3.3.

A restrição (3.8), embora não definida explicitamente na heurística, não irá interferir no cálculo do limite superior. Sendo uma restrição redundante, sua finalidade é apenas procurar melhorar o limite inferior do problema. A heurística seguinte, resolve o problema de conectividade e garante o atendimento das demais restrições para o PDCC-RSSF.

1. Para cada $l \in S^a$ faça:

- (a) $t_l = 1$
- (b) Calcular o caminho mínimo de t_l ao nó sorvedouro mais próximo.
- (c) Ative as variáveis z_{ij}^l ($z_{ij}^l = 1$) referentes ao caminho.
- (d) Ative as variáveis t_i ($t_i = 1$) referentes aos nós sensores i pertencentes ao caminho de l ao nó sorvedouro mais próximo.

No final destes passos, teremos o problema de conectividade resolvido e uma solução viável, ou seja, um limite superior para o PDCC-RSSF.

4.3.2 Heurística Lagrangeana

A heurística que será apresentada nesta seção, conhecida na literatura como heurística Lagrangeana, procura a partir da solução fornecida pelo limite inferior, gerar soluções viáveis para o PDCC-RSSF. A heurística funciona da seguinte maneira: a solução ótima obtida para o problema relaxado, já ativa um conjunto de nós sensores (variáveis t_i). O objetivo da heurística abaixo, é checar se a cobertura total já foi garantida, caso não tenha sido, procura-se ativar o menor número de sensores para atender aos pontos ainda não cobertos e, no final, garante a conectividade. Assim, temos também um limite superior para o PDCC-RSSF. Abaixo, segue o pseudo-código da heurística.

Considerando os mesmos conjuntos e variáveis da seção anterior. Uma solução para garantir a cobertura pode ser obtida através do seguinte algoritmo:

1. Inicialize variáveis e conjuntos com:

- S^a = Nós sensores ativos na solução do problema relaxado;
- $S^d = N^s - N^a$;
- D_j = Número de nós sensores ativos que cobrem os pontos j ;
- $Iteracoes = 1$.

2. Retire de P os pontos cobertos por pelo menos m nós sensores obtidos através do problema relaxado.

3. Enquanto $P \neq \emptyset$ e $Iteracoes \leq N^s$ faça:

- (a) Selecione $i \in S^d$, o nó sensor de menor custo c'_i calculado da seguinte forma:

$$c'_i = \frac{c_i + C_i}{A_i}$$
- (b) $S^a = S^a \cup i$
- (c) $S^d = S^d - i$
- (d) Se i cobre j então faça $D_j = D_j + 1$, $\forall j \in N^d$
- (e) Se $D_j \geq m$ e $j \in P$ (cobertura já foi garantida), $\forall j \in N^d$ faça:
 - i. $P = P - j$
- (f) $Iteracoes = Iteracoes + 1$

Desta forma, teremos também no final as restrições (3.2 e 3.3) atendidas. O restante do algoritmo para obtenção do limite superior é idêntico ao apresentado anteriormente.

4.4 Considerações Finais

Neste capítulo foram expostos a técnica e os algoritmos utilizados para obtenção dos limites inferior e superior para o PDCC-RSSF. No capítulo seguinte serão apresentados o gerador de instância e a experiência computacional realizada através da implementação destes algoritmos.

Capítulo 5

Resultados Computacionais

Neste capítulo serão apresentados o gerador de instâncias desenvolvido e os resultados computacionais obtidos através da implementação dos algoritmos descritos no Capítulo 4 para a solução do PDCC-RSSF.

5.1 Gerador de Instâncias

Com o objetivo de validar o modelo de otimização proposto, foi desenvolvido um gerador de instâncias. Com este gerador, é possível simular diversas características das redes de sensores: redes homogêneas ou heterogêneas, lançamento aleatório ou em posições específicas, presença ou ausência de obstáculos.

O gerador recebe como entrada os seguintes parâmetros:

- Número e coordenadas (x,y) dos nós sensores;
- Número e coordenadas (x,y) dos nós sorvedouros;
- Raio de Comunicação;
- Raio de Sensoriamento;
- Dimensão da área;
- Número de pontos de demanda;
- Precisão de sensoriamento (número mínimo de nós sensores ativos para cobrir determinada área);
- Posição dos obstáculos (coordenadas (x,y)) presentes na área de monitoramento.

Como saída, o gerador fornece um arquivo que representa o modelo de otimização proposto no Capítulo 3. Com este arquivo, é possível utilizar dos diversos pacotes de otimização existentes no mercado para obter a solução do modelo. Ou seja, o número mínimo de nós a serem ativados para cobrir a área, bem como o caminho que cada pacote de informação deve percorrer para chegar ao nó sorvedouro. Além de ser utilizado pelos pacotes de otimização, o gerador de instâncias é utilizado para gerar os parâmetros de entrada da heurística. As próximas seções irão mostrar como foi obtida a localização dos nós sensores e sorvedouros, bem como as características das instâncias criadas para o PDCC-RSSF.

5.1.1 Localização dos Nós Sensores

Os nós sensores foram dispostos na área simulando um lançamento feito por aeronave. Ou seja, as coordenadas (x,y) dos nós sensores foi obtida aleatoriamente. A figura (5.1) representa como os nós sensores foram dispostos.

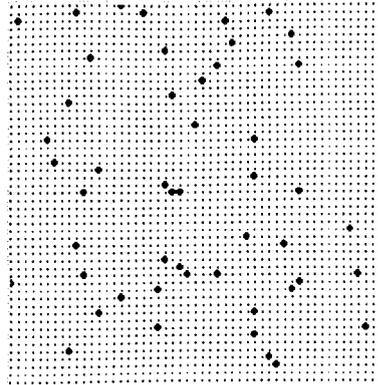


Figura 5.1: Disposição dos nós sensores.

5.1.2 Localização dos Nós Sorvedouros

Os experimentos foram realizados considerando duas configurações de posição para os nós sorvedouros: na primeira configuração, os nós sorvedouros foram posicionados nos extremos da área. Já na segunda, foram definidos dois nós sorvedouros situados um no canto superior esquerdo e, o outro, no canto inferior direito. As figuras (5.2) e (5.3) representam respectivamente estas configurações.

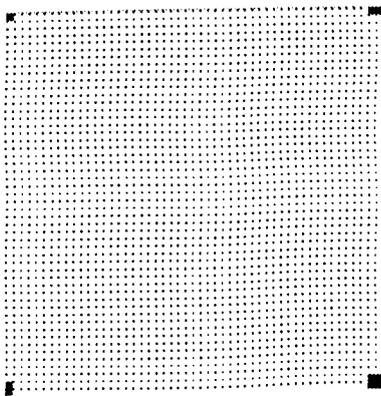


Figura 5.2: Primeira disposição dos nós.

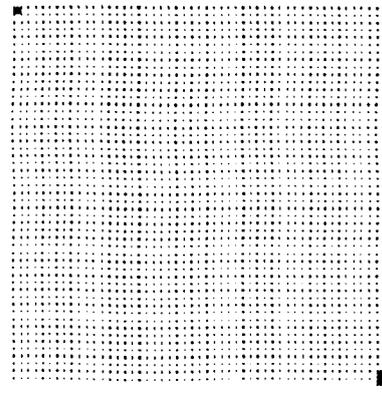


Figura 5.3: Segunda disposição dos nós.

5.1.3 Instâncias de Teste

Para realização dos experimentos computacionais, foram geradas quatro baterias de testes: Na primeira bateria, foram criadas quatorze instâncias conforme a tabela (5.1) especifica. Para todas estas instâncias, foram definidos que o número de pontos de demanda seria 10000 (dez mil) o que significa que a no máximo $0.09m^2$ existe um ponto de demanda que deve estar sendo coberto por pelo menos um nó sensor. Foram definidos que o número de nós sorvedouros seria quatro (um sorvedouro em cada extremo da área), foi considerado também que cada nó sensor consome uma unidade

de energia para ativação. Os campos Rc e Rs representam respectivamente os raios de comunicação e sensoriamento.

Instância	Nós Sensores	Rc	Rs	Área
p1	15	15	15	20x20
p2	15	30	15	20x20
p3	20	15	15	20x20
p4	20	30	15	20x20
p5	25	15	15	25X25
p6	25	30	15	25X25
p7	30	20	20	25X25
p8	30	40	20	25X25
p9	40	15	15	25X25
p10	40	30	15	25X25
p11	40	20	20	25X25
p12	40	40	20	25X25
p13	40	20	20	30X30
p14	40	40	20	30X30

Tabela 5.1: Conjunto de instâncias da primeira bateria.

A segunda bateria de testes (tabela (5.2)), correspondente ao conjunto de instâncias ($p15 - p36$), representam uma rede de sensores disposta em uma área maior (foram realizados experimentos simulando uma área de até $25600m^2$). Para estas instâncias, também foi definido quatro nós sorvedouros e 10000 pontos de demanda.

Instância	Nós Sensores	Rc	Rs	Área
p15	15	20	20	25x25
p16	15	40	20	25X25
p17	20	20	20	50X50
p18	20	40	20	50X50
p19	40	30	30	100X100
p20	40	60	30	100X100
p21	60	30	30	100X100
p22	60	60	30	100X100
p23	80	30	30	100X100
p24	80	60	30	100X100
p25	100	30	30	100X100
p26	100	60	30	100X100
p27	120	30	30	140X140
p28	120	60	30	140X140
p29	140	30	30	140X140
p30	140	60	30	140X140
p31	160	30	30	140X140
p32	160	60	30	140X140
p33	180	30	30	140X140
p34	180	60	30	140X140
p35	200	30	30	160X160
p36	200	60	30	160X160

Tabela 5.2: Conjunto de instâncias da segunda bateria.

A terceira bateria de testes (tabela (5.3), instâncias $p37 - p44$), correspondem as instâncias $p19 - p26$ da tabela (5.2). Para estas instâncias, a única alteração é a consideração de apenas dois nós sorvedouros na área, conforme ilustra a figura (5.3).

Instância	Nós Sensores	Rc	Rs	Área
p37	40	30	30	100X100
p38	40	60	30	100X100
p39	60	30	30	100X100
p40	60	60	30	100X100
p41	80	30	30	100X100
p42	80	60	30	100X100
p43	100	30	30	100X100
p44	100	60	30	100X100

Tabela 5.3: Conjunto de instâncias da terceira bateria.

A quarta bateria de testes consiste de oito instâncias (tabela 5.4) que foram geradas considerando a presença de obstáculos na área de monitoramento. Para a geração de instâncias considerando obstáculos, basta realizar uma modificação no conjunto de arcos A^d , A^s e A^m (da formulação que representa o PDCC-RSSF), de forma que só haverá um arco entre os nós sensores e pontos de demanda, entre os próprios nós sensores e entre nós sensores e nós sorvedouros que não estão nas áreas de influência ou cobertura de cada nó devido a estes obstáculos.

Instância	Nós Sensores	Rc	Rs	Área
p45	40	30	30	100X100
p46	40	60	30	100X100
p47	60	30	30	100X100
p48	60	60	30	100X100
p49	80	30	30	100X100
p50	80	60	30	100X100
p51	100	30	30	100X100
p52	100	60	30	100X100

Tabela 5.4: Conjunto de instâncias da quarta bateria.

As figuras abaixo (5.4, 5.5, 5.6 e 5.7) mostram como os obstáculos foram distribuídos na área de monitoramento. As barras horizontais e verticais inseridas na figura representam os obstáculos. Desta forma, os nós sensores posicionados abaixo (à direita) de uma destas barras não consegue estabelecer comunicação e nem monitorar a região situada acima (à esquerda) desta mesma barra. Cabe ressaltar, que embora na prática as capacidades de sensoriamento e comunicação de um nó sensor não sejam totalmente reduzidas na presença de obstáculos, neste trabalho, com o intuito de facilitar a geração das instâncias, foi considerado que os obstáculos eliminam ou barram por completo o sinal dos nós sensores.

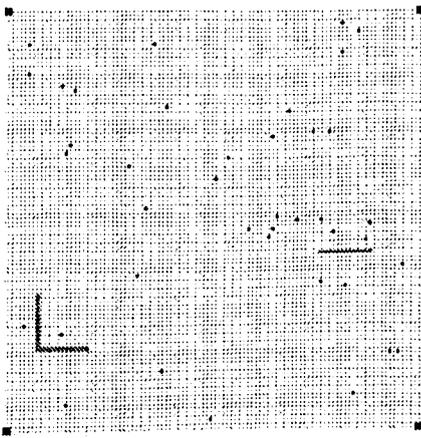


Figura 5.4: Obstáculos para as instâncias p45 e p46.

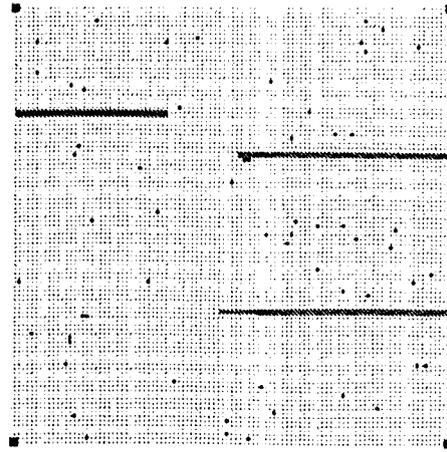


Figura 5.5: Obstáculos para as instâncias p47 e p48.

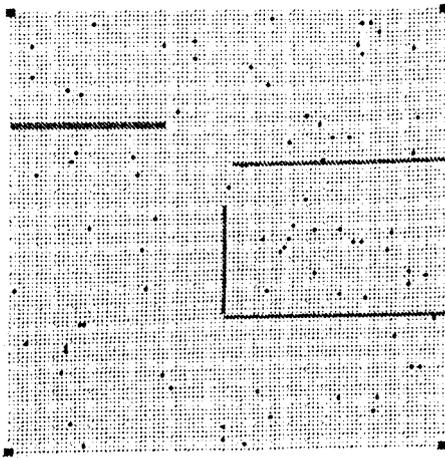


Figura 5.6: Obstáculos para as instâncias p49 e p50.

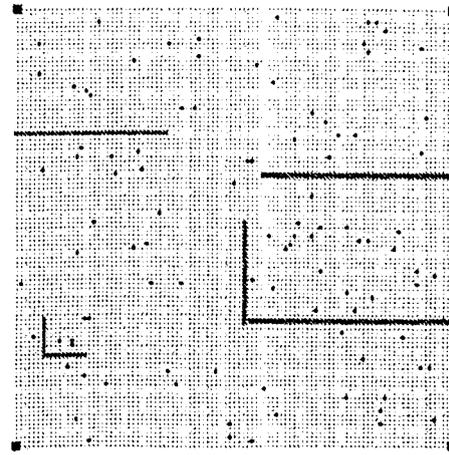


Figura 5.7: Obstáculos para as instâncias p51 e p52.

5.2 Experimentos

Nesta seção, várias instâncias do problema PDCC-RSSF de tamanhos diferentes são resolvidas utilizando a heurística baseada na relaxação Lagrangeana. O objetivo é avaliar a qualidade da heurística proposta. Para isto, são considerados o *gap* de dualidade obtido através dos valores do limite inferior e superior, e através da comparação dos resultados da heurística e do pacote de otimização CPLEX 7.0.

A relaxação Lagrangeana foi implementada em linguagem C ANSI e as instâncias foram resolvidas em uma estação SunBlade 100, ultraspace IIe com 500MHz e 1GB de memória RAM.

5.2.1 Avaliação do Algoritmo de Solução

Existem três maneiras de se avaliar a qualidade da solução heurística proposta: na primeira, quando a solução ótima do problema é conhecida, basta realizar sua comparação com esta solução. Desta forma, sabe-se qual a diferença entre a solução ótima

e a solução heurística. Entretanto, devido à complexidade envolvida em obter soluções ótimas para o PDCC-RSSF, o conhecimento de soluções ótimas para o PDCC-RSSF é limitado a pequenas instâncias.

Uma segunda maneira, quando a solução ótima não é conhecida, consiste em utilizar o valor do *gap* de dualidade. Esta informação fornece uma estimativa de quão longe se está da solução ótima. Este cálculo é baseado no valor obtido pelos limites inferior e superior do problema.

Outra forma de se avaliar a solução heurística encontrada, consiste em avaliar a diferença existente entre a solução heurística e a melhor solução viável conhecida para o problema. Embora esta análise não forneça informações sobre a distância que a solução encontra-se do ótimo, ela ajuda a avaliar a qualidade da solução heurística obtida com a melhor solução viável obtida para o problema. Como o modelo para o PDCC-RSSF é um modelo novo, não existem outros resultados disponíveis que possam ser utilizados para comparar com a heurística. Desta forma, procurou-se utilizar os resultados fornecidos pelo pacote CPLEX como referência para o problema.

5.2.2 Apresentação dos Resultados

Os resultados serão divididos em quatro partes. Cada parte corresponde aos resultados obtidos com uma das quatro baterias de testes utilizadas.

As tabelas de resultados apresentam as seguintes colunas: Instância (nome da instância que representa o PDCC-RSSF), Nós Sensores (número de nós lançados na área), CPLEX(s) (tempo gasto, em segundos pelo pacote CPLEX para resolver a instância), Z_{LS} (valor do limite superior obtido pelo pacote CPLEX), *gap* (distância que a melhor solução obtida pelo pacote encontra-se da solução ótima, com base na relaxação linear), Ativos (número de nós sensores ativos para cobertura e conectividade), RL(s) (tempo gasto, em segundos pelo algoritmo da relaxação Lagrangeana), Z_{LS} (valor do limite superior obtido pela heurística Lagrangeana), Iterações (número de iterações do algoritmo de relaxação Lagrangeana), *gap* (gap de dualidade calculado a partir da expressão $gap = 100(Z_{LS} - Z_{LI})/Z_{LS}$ que avalia em termos percentuais (%) o *gap* entre o limite superior e inferior) e *GAP* que serve para analisar a qualidade da solução heurística com a melhor solução fornecida pelo pacote CPLEX. Calculado da seguinte forma: $GAP = 100(Z_{LS}(Rel.Lagr) - Z_{LS}(CPLEX))/Z_{LS}(CPLEX)$, para os casos em que o limite superior obtido pelo CPLEX é melhor que aquele obtido pela relaxação Lagrangeana.

5.2.3 Bateria de testes 1

A primeira bateria de testes foi dividida em duas tabelas de resultados. Na primeira tabela de resultados, os experimentos foram conduzidos da seguinte forma: limitou-se o tempo de resolução das instâncias em 3600 segundos tanto para o pacote de otimização quanto para a heurística. Os resultados são apresentados na tabela (5.5).

Instância	Nós Sensores	Cplex(s)	Z_{LS}	gap	Ativos	RL(s)	Z_{LS}	Iteracoes	gap	Ativos	GAP
p1	15	879.41	17.59	0	2	250	17.59	4612	0	2	0
p2	15	871.84	17.59	0	2	251	17.59	4612	0	2	0
p3	20	10808	40.5	86.54	5	153	14.54	2210	0	2	0
p4	20	10808	40.36	75.24	4	152	14.54	2210	0	2	0
p5	25	6436.92	20.13	0	4	180	20.13	2227	0	4	0
p6	25	6601.52	20.13	0	4	180	20.13	2219	0	4	0
p7	30	10813	13.82	20.01	3	233	13.82	2225	0	3	0
p8	30	10814.17	49.86	77.84	2	233	13.82	2225	0	3	0

Tabela 5.5: Experiência Computacional - Bateria 1.1.

Com relação aos resultados obtidos e apresentados na tabela (5.5), os seguintes comentários podem ser feitos: o pacote de otimização CPLEX conseguiu encontrar a solução ótima apenas para a metade das instâncias propostas, sendo que o tempo para encontrar as soluções variou de aproximadamente 14 minutos (instâncias p1 e p2) a aproximadamente 2 horas (instâncias p5 e p6). Para as demais instâncias, com exceção da instância p7 (que embora o pacote tenha encontrado a solução ótima, foi incapaz de provar a sua otimalidade), o pacote de otimização não conseguiu obter a solução ótima, fornecendo apenas soluções viáveis com *gap* elevado. Este resultado já mostra que o limite inferior obtido pela relaxação linear não é bom.

Por outro lado, a heurística baseada na relaxação Lagrangeana conseguiu encontrar a solução ótima para todas as instâncias com um tempo de processamento muito abaixo que o realizado pelo resolvidor. O pior tempo da heurística para obter a solução ótima foi de aproximadamente 4 minutos (instâncias p1 e p2).

Já na segunda tabela de resultados, permitiu-se que o pacote de otimização CPLEX obtivesse a solução ótima das instâncias independente do tempo de processamento. Os resultados são apresentados na tabela (5.6)

Instância	Nós Sensores	Cplex(s)	Z_{LS}	gap	Ativos	RL(s)	Z_{LS}	Iteracoes	gap	Ativos	GAP
p9	40	96714.26	24.96	0	3	290	24.96	2222	0	3	0
p10	40	112831.83	24.96	0	3	289	24.96	2222	0	3	0
p11	40	149024.13	14.01	0	2	310	14.01	2221	0	2	0
p12	40	128801.41	14.01	0	2	311	14.01	2220	0	2	0
p13	40	61995.67	19.3	0	4	313	19.3	2344	0	4	0
p14	40	62552.78	19.3	0	4	313	19.3	2346	0	4	0

Tabela 5.6: Experiência Computacional - Bateria 1.2.

Com base nos resultados obtidos para as instâncias da tabela (5.6) as seguintes análises podem ser feitas: o pacote de otimização embora tenha obtido a solução ótima para todas as instâncias, consumiu muito tempo. O melhor resultado do cplex foi a instância p13 aproximadamente (17 horas) enquanto que o pior foi para resolver a instância p11 aproximadamente (41 horas). Tal resultado invalida a utilização do CPLEX como ferramenta para obter soluções para o PDCC-RSSF. Pois um dos objetivos do modelo e alvo dos trabalhos futuros é o de embutir o modelo que representa o PDCC-RSSF em um ambiente de gerenciamento de RSSFs. Tal ambiente de gerenciamento necessita de soluções em intervalos de tempo menores e, em geral, para problemas de maior dimensão.

Embora o resolvidor tenha demonstrado ineficiência em resolver estas instâncias, a heurística baseada na relaxação Lagrangeana conseguiu encontrar a solução ótima para todas as instâncias e de maneira muito eficiente. As soluções foram obtidas em no máximo 5 minutos. Tal resultado sugere que a heurística possa ser utilizada como ferramenta para auxiliar os mecanismos de gerenciamento de RSSFs.

5.2.4 Bateria de testes 2

A segunda bateria de resultados, que corresponde ao conjunto de instâncias p15 a p36, representa instâncias de maior porte para o PDCC-RSSF. Para estas instâncias, o tempo limite de processamento foi de 3 horas e os resultados são apresentados na tabela (5.7)

Instância	Nós Sensores	Cplex(s)	Z _{LS}	gap	Ativos	RL(s)	Z _{LS}	Iteracoes	gap	Ativos	GAP
p15	15	9125	17.19	0	3	246	17.19	2212	9.04	3	0
p16	15	10238	17.19	0	3	245	17.19	2212	9.04	3	0
p17	20	1942.14	66.72	0	4	280	66.72	2212	0	4	0
p18	20	2727.6	66.72	0	4	282	66.72	2217	0	4	0
p19	40	-	-	-	-	639	429.37	2321	35.92	15	-
p20	40	9661.47	286.75	0	5	907	309.45	3403	14.12	9	7.9
p21	60	10810.48	374.68	53.72	10	1237	384.58	2272	32.23	13	2.64
p22	60	-	-	-	-	2633	271.57	2263	7.46	9	-
p23	80	-	-	-	-	1901	377.46	2270	33.14	13	-
p24	80	10843.47	1156.44	85.29	14	1909	308.17	2260	20.84	10	-
p25	100	-	-	-	-	2294	325.98	2259	28.75	12	-
p26	100	-	-	-	-	2291	277.19	2275	18.32	9	-
p27	120	-	-	-	-	3352	1143.18	2310	43.9	28	-
p28	120	-	-	-	-	3339	901.76	2321	34.18	21	-
p29	140	-	-	-	-	3610	1229.4	2279	52.02	33	-
p30	140	-	-	-	-	3608	896.67	2265	36.7	21	-
p31	160	-	-	-	-	5340	1271.66	2296	54.23	33	-
p32	160	-	-	-	-	4980	890.7	2279	37.14	21	-
p33	180	-	-	-	-	6060	1315.64	2264	56.02	35	-
p34	180	-	-	-	-	5400	889.41	2271	37.3	22	-
p35	200	-	-	-	-	8940	1990.09	2291	56.76	42	-
p36	200	-	-	-	-	7860	1425.3	2283	42.19	29	-

Tabela 5.7: Experiência Computacional - Bateria 2.

Com base na tabela acima, os seguintes comentários podem ser feitos: para as instâncias p15 a p18, o pacote de otimização conseguiu encontrar a solução ótima e provar a otimalidade (gap de dualidade zero) mas com um tempo relativamente elevado. Já a heurística, também conseguiu encontrar a solução ótima para todas estas instâncias e com um tempo muito inferior ao resolvidor. Para as instâncias p15 e p16 a heurística não provou a otimalidade (gap de dualidade maior que zero), mas a solução viável obtida é a ótima, o que pode ser observado pela coluna que representa o GAP entre a solução viável da heurística e a solução do pacote de otimização.

Para as demais instâncias, com exceção das instâncias p20, p21 e p24 o resolvidor não foi capaz de obter nenhuma solução viável para o problema. Enquanto a heurística conseguiu obter soluções viáveis que, embora não haja garantia de serem soluções ótimas, foram obtidas em no máximo 3 horas de processamento.

As instâncias p20, p21 e p24 merecem ser analisadas individualmente: para a instância p20, o CPLEX foi capaz de obter uma solução ótima (gap zero) em aproximadamente 9661 segundos (duas horas e sessenta e oito minutos). Enquanto a heurística, conseguiu obter uma solução viável em 15 (quinze) minutos e que é apenas 7,9% pior que a solução ótima. Já para a instância p21, o CPLEX conseguiu obter uma solução viável melhor que a obtida pela heurística, o que pode ser observado pelo número de nós sensores que foi ativo. Enquanto o CPLEX obteve uma solução em que 10 (dez) nós sensores necessitam estar ativos, a heurística encontrou uma solução que necessita de 13 (treze) nós sensores. Para a instância p24, ocorreu o contrário, o resolvidor obteve uma solução que necessita de 14 (quatorze) nós sensores, enquanto a heurística forneceu uma solução que precisa de 10 (dez) nós sensores. Tal comportamento pode indicar que a heurística utilizada para cálculo do limite superior pode ser melhorada,

ou seja, que é possível reduzir ainda o valor do limite superior do problema, bem como elevar o limite inferior pela adição de restrições redundantes e cortes no conjunto viável. Cabe ressaltar também que para todas as instâncias, a heurística forneceu soluções em melhor tempo.

5.2.5 Bateria de testes 3

A terceira bateria de testes (instâncias p37 a p44) foram criadas considerando apenas dois nós sorvedouros dispostos na área. Conforme ilustra a figura (5.3). Os resultados desta bateria serão analisados da seguinte forma: primeiro, será feita uma comparação das soluções obtidas pelo pacote CPLEX considerando quatro e dois nós sorvedouros (tabela 5.8). Em seguida, serão realizadas comparações entre as soluções heurísticas para quatro e dois nós sorvedouros (tabela 5.9). E, por último, serão comparados os resultados do CPLEX com a heurística considerando dois nós sorvedouros (tabela 5.10). Para todos os experimentos desta bateria foi definido o tempo limite de 3 horas. As colunas $Cplex_4(s)$ e $Cplex_2(s)$ representam o tempo de solução encontrado pelo resolvidor considerando quatro e dois nós sorvedouros respectivamente. O mesmo raciocínio vale para as colunas $RL_4(s)$ e $RL_2(s)$ que correspondem ao tempo de solução obtido pela heurística.

Instância	Nós Sensores	$Cplex_4(s)$	Z_{LS}	gap	Ativos	$Cplex_2(s)$	Z_{LS}	gap	Ativos
p37	40	-	-	-	-	10810	675.39	0	12
p38	40	9661	286.75	0	5	3170	419.65	0	9
p39	60	10810.48	374.78	53.72	10	10808	799.3	60.19	15
p40	60	-	-	-	-	10820	633.31	53.06	13
p41	80	-	-	-	-	-	-	-	-
p42	80	10843.47	1156.44	85.29	14	-	-	-	-
p43	100	-	-	-	-	-	-	-	-
p44	100	-	-	-	-	-	-	-	-

Tabela 5.8: Comparação CPLEX com 4 e 2 nós sorvedouros.

De acordo com a tabela (5.8), pode-se perceber um aumento no número de nós sensores ativos quando é reduzido as opções de nós sorvedouros, o que já era esperado uma vez que com menor opções de caminhos, mais nós sensores precisam estar ativos para alcançar um dos nós sorvedouros. Quanto a eficácia do pacote de otimização, este conseguiu encontrar soluções ótimas apenas para uma instância (p38) considerando quatro nós sorvedouros e para duas instâncias (p37 e p38) quando eram definidos apenas 2 nós sorvedouros. Para as demais, foi possível encontrar solução viável apenas para as instâncias (p39 e p42) para quatro sorvedouros e para (p39 e p40) quando o número de nós sorvedouros disponíveis eram dois. A próxima tabela apresenta uma comparação das soluções obtidas pela heurística para quatro e dois nós sorvedouros.

Instância	Nós Sensores	$RL_4(s)$	Z_{LS}	gap	Ativos	GAP	$RL_2(s)$	Z_{LS}	gap	Ativos	GAP
p37	40	639	429.37	35.92	15	-	1054	762.3	33.61	16	12.86
p38	40	907	309.45	14.12	9	6.42	2338	435.89	4.66	9	3.86
p39	60	1237	384.58	32.23	13	-	1221	656.14	36.94	16	-
p40	60	2633	271.57	7.46	13	-	1269	548.4	28.07	12	-
p41	80	1901	377.46	33.14	13	-	1786	724.6	45.66	18	-
p42	80	1909	308.17	20.84	10	-	1735	470.21	20.57	10	-
p43	100	2294	325.98	28.75	12	-	2367	745.16	50.74	18	-
p44	100	2291	277.19	18.32	9	-	2323	503.7	29.53	11	-

Tabela 5.9: Comparação Heurística com 4 e 2 nós sorvedouros.

de processamento do resolvidor mostrou-se muito mais elevado do que o obtido pela heurística. Outro aspecto a ser considerado é que embora a heurística tenha apresentado *gaps* de dualidade elevados, quando sua solução é comparada com a solução ótima obtida pelo CPLEX, observa-se para as instâncias (p46 e p47) que a solução obtida encontra-se bem perto da solução ótima. Para a instância p45, observa-se também que a solução apresenta-se mais perto do ótimo do que a coluna *gap* (que representa a distância entre os limites inferior e superior) apresenta. Tais resultados indicam, conforme já apresentado anteriormente, que a heurística para obtenção do limite superior pode ser melhorada. Para as demais instâncias, o resolvidor não foi capaz de obter uma solução viável após 5 horas de processamento.

As figuras seguintes (5.8 à 5.23) ilustram as soluções obtidas pela heurística considerando a presença e ausência de obstáculos tendo como base as instâncias da tabela (5.11). As linhas ligando os pontos representam as rotas que cada pacote de informação deve percorrer para chegar a um dos nós sorvedouros, os círculos representam a área de sensoriamento coberta por cada nó sensor ativo e as barras representam os obstáculos existentes na área de monitoramento. Os títulos existentes abaixo de cada figura representam o número de nós sensores dispostos na área de monitoramento, se a área apresenta ou não obstáculos e o alcance (raios) de comunicação e sensoriamento.

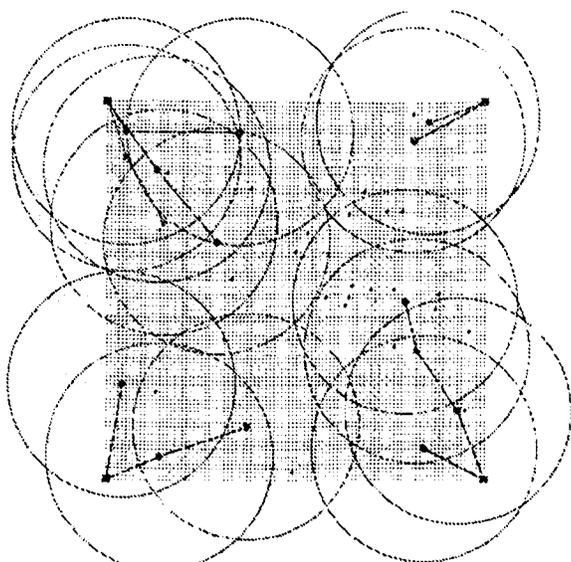


Figura 5.8: 40 nós sem obstáculo 30X30.

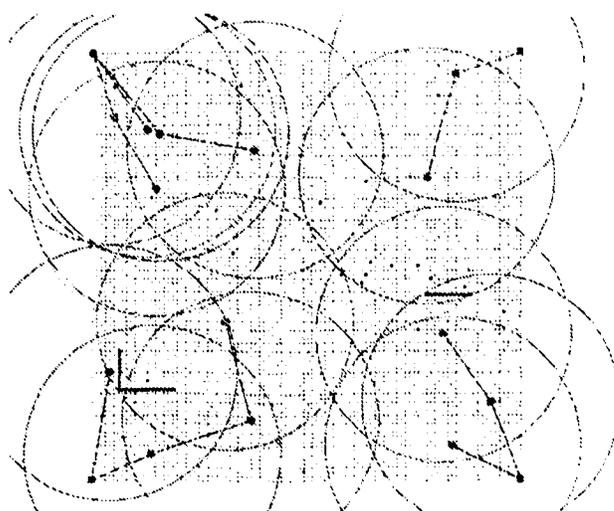


Figura 5.9: 40 nós com obstáculo 30X30.

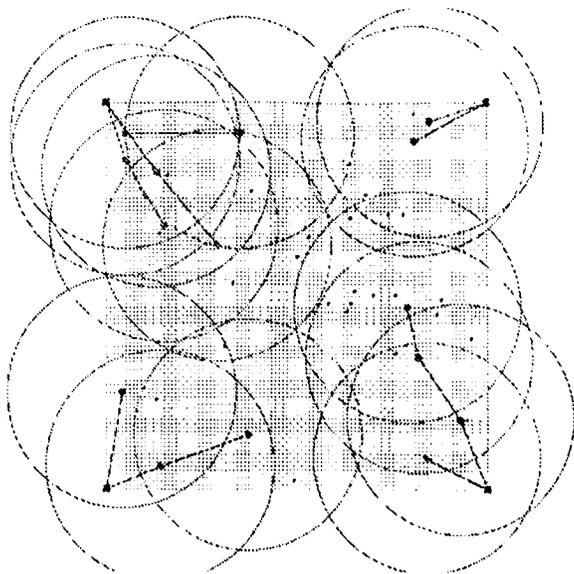


Figura 5.10: 40 nós sem obstáculo 60X30.

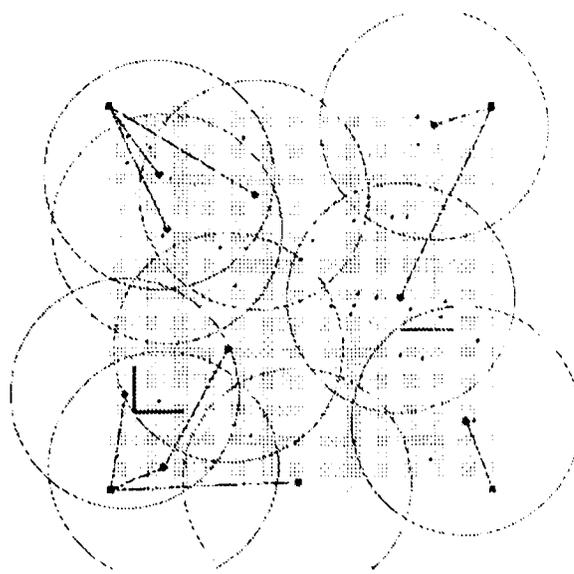


Figura 5.11: 40 nós com obstáculo 60X30.

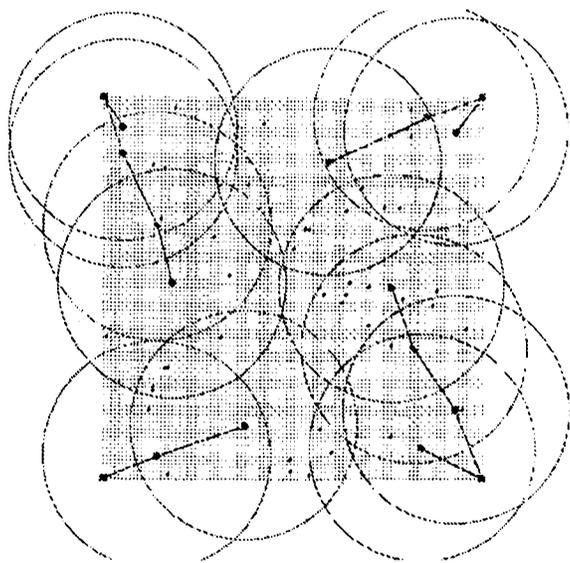


Figura 5.12: 60 nós sem obstáculo 30X30.

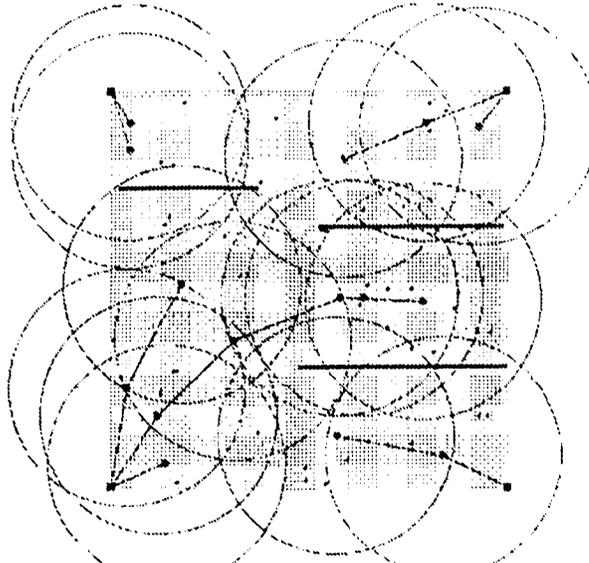


Figura 5.13: 60 nós com obstáculo 30X30.

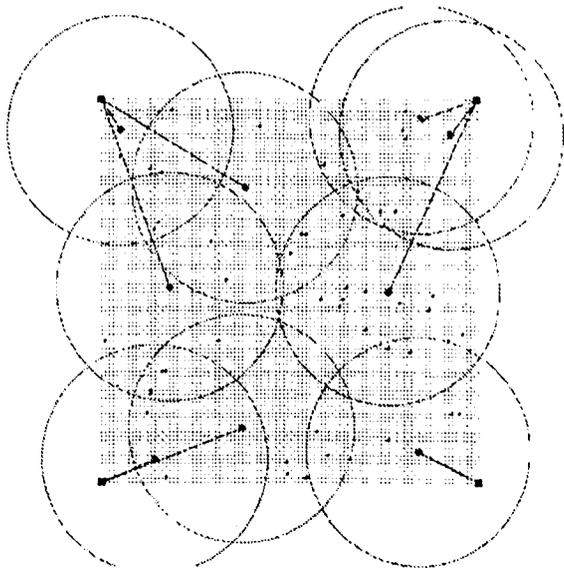


Figura 5.14: 60 nós sem obstáculo 60X30.

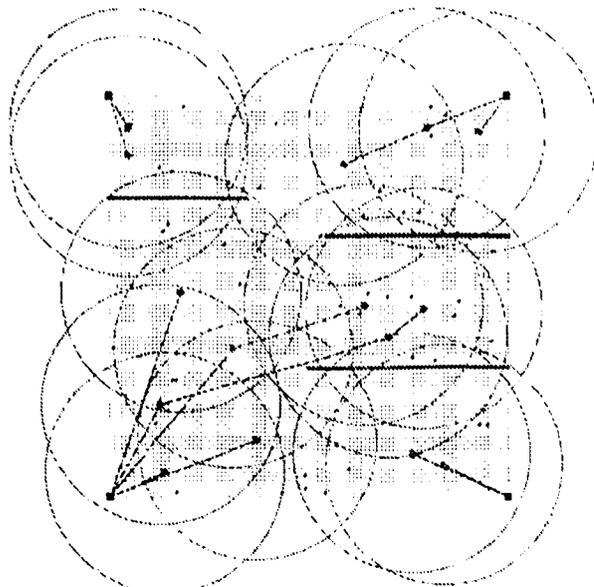


Figura 5.15: 60 nós com obstáculo 60X30.

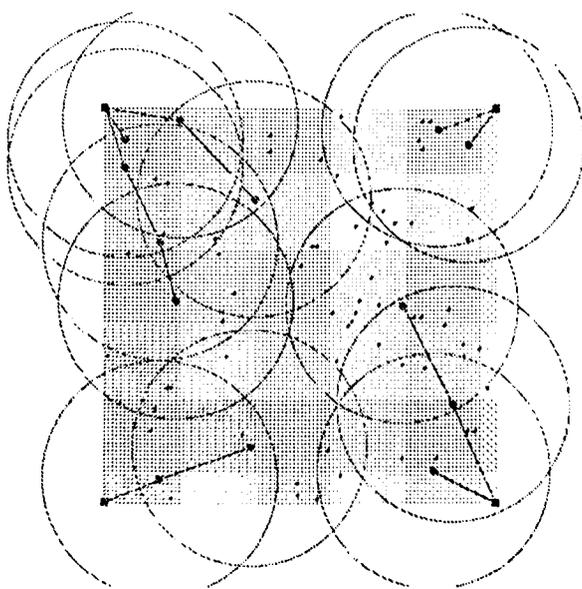


Figura 5.16: 80 nós sem obstáculo 30X30.

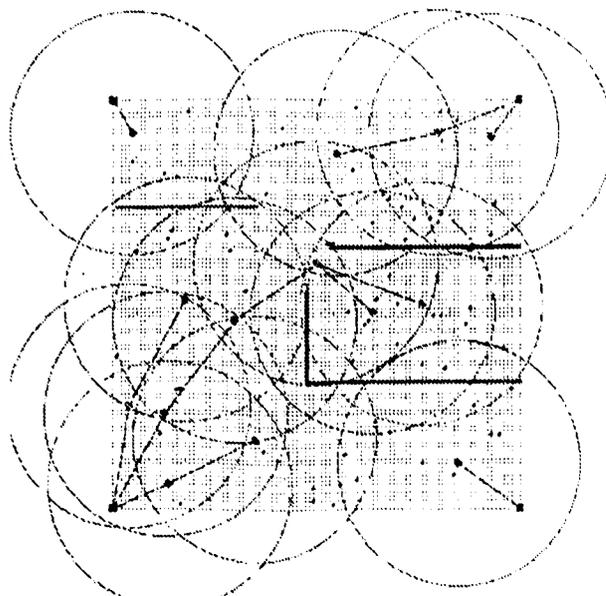


Figura 5.17: 80 nós com obstáculo 30X30.

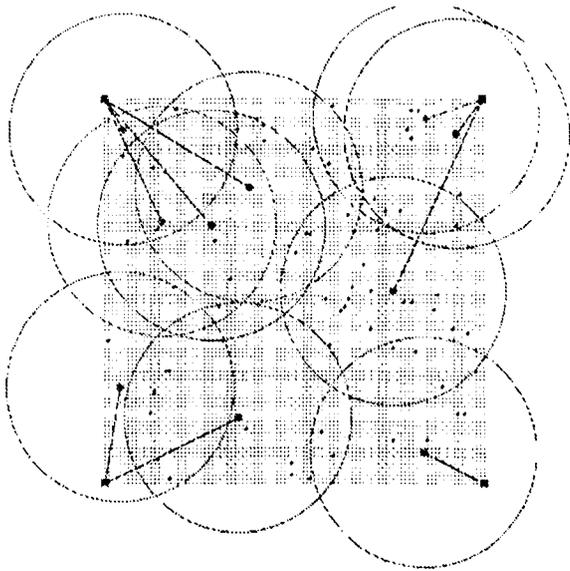


Figura 5.18: 80 nós sem obstáculo 60X30.

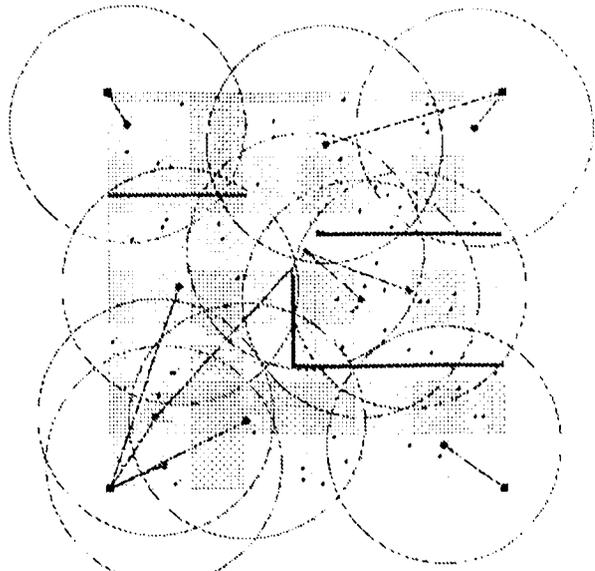


Figura 5.19: 80 nós com obstáculo 60X30.

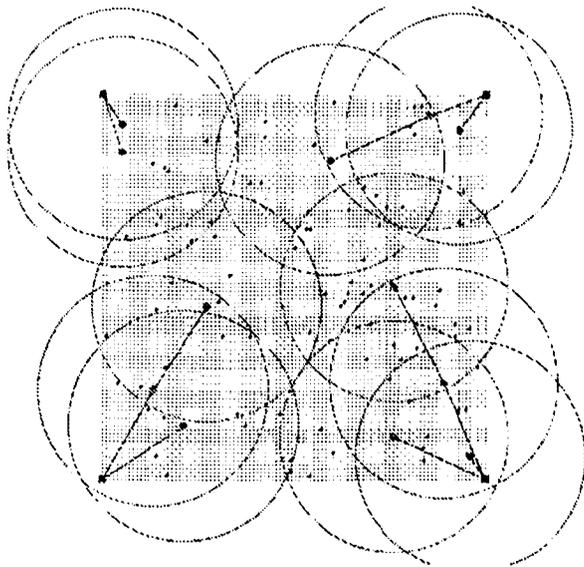


Figura 5.20: 100 nós sem obstáculo 30X30.

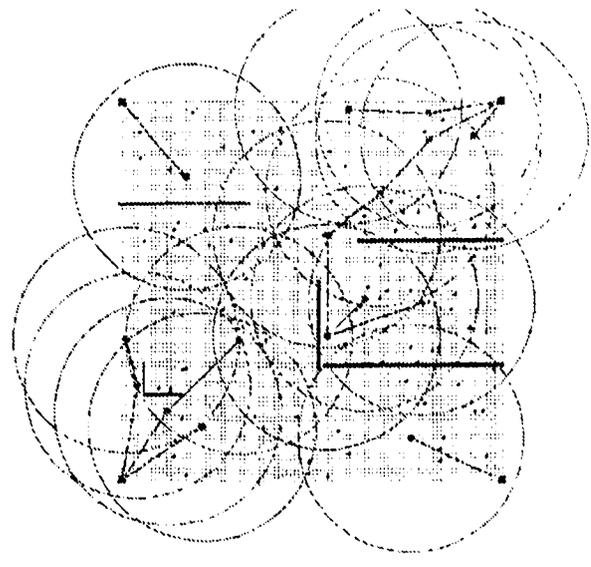


Figura 5.21: 100 nós com obstáculo 30X30.

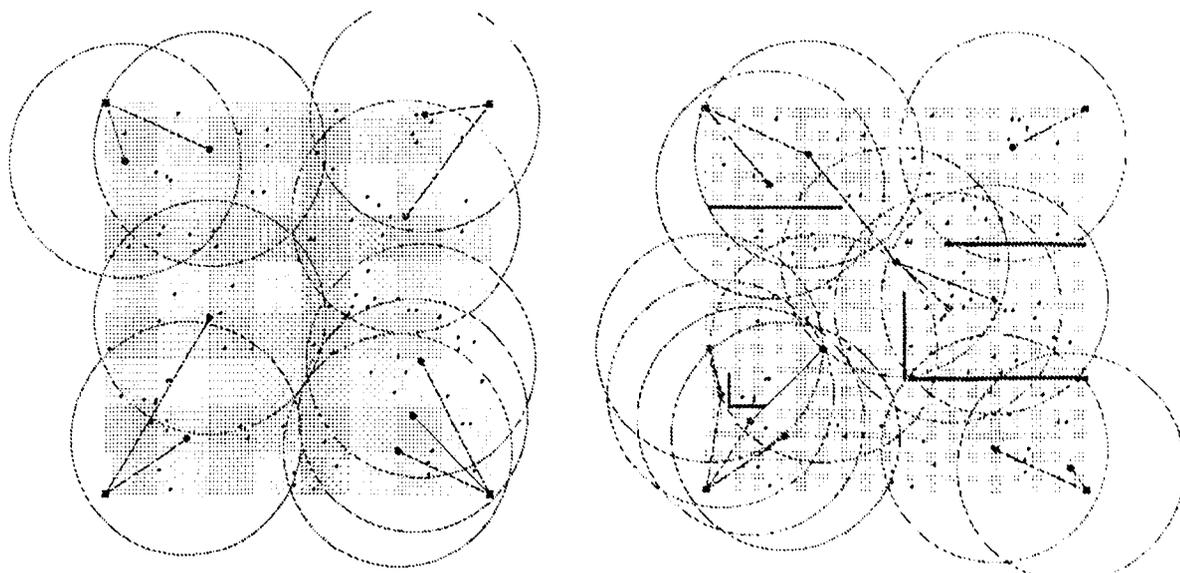


Figura 5.22: 100 nós sem obstáculo 60X30. Figura 5.23: 100 nós com obstáculo 60X30.

Com relação as soluções obtidas com e sem obstáculos e exibidas através das figuras acima, pode-se perceber que com obstáculos, há um aumento no número de nós sensores ativos e de rotas. Pois, para escapar dos obstáculos, passa a ser necessário mais nós sensores ativos para a cobertura e mais caminhos para garantir a conectividade da rede.

5.3 Considerações Finais

Neste capítulo foram apresentados o gerador de instâncias e os experimentos computacionais realizados para avaliar a qualidade da solução heurística proposta para resolver o PDCC-RSSF. A heurística baseada na relaxação Lagrangeana e no método de sub-gradientes conseguiu obter bons resultados. Para determinadas instâncias, a heurística obteve a solução ótima ou forneceu soluções que estão muito próximas do ótimo. Por outro lado, o pacote de otimização CPLEX não obteve resultados satisfatórios. Para um grande número de instâncias o resolvidor não foi capaz de gerar nenhuma solução viável. E mesmo para as instâncias em que foi possível obter o ótimo, o tempo de solução mostrou-se muito mais elevado que o obtido pela heurística. Embora a heurística tenha apresentado resultados satisfatórios, alguns ajustes podem ser realizados para procurar reduzir o *gap* entre os limites inferior e superior. Podem ser implementadas novas heurísticas para o limite superior, bem como podem ser estudados cortes na estrutura poliédrica do problema.

Capítulo 6

Considerações Finais

6.1 Conclusão

Neste trabalho foi apresentado um modelo de programação linear inteira mista (PLIM) para o problema de Densidade, Cobertura e Conectividade em Redes de Sensores sem Fio (PDCC-RSSF). A solução do modelo fornece um conjunto de nós sensores que devem estar ativos para garantir a cobertura da área de monitoramento bem como uma rota de cada nó sensor ativo até um dos nós sorvedouros minimizando o consumo de energia. Considerando as características combinatórias do problema, foi desenvolvido um algoritmo baseado na relaxação Lagrangeana e no método de sub-gradientes. Através dos experimentos computacionais realizados, foi possível constatar que o algoritmo proposto foi capaz de obter a solução ótima para diversas instâncias do problema, além de fornecer soluções que se não são ótimas, acredita-se estarem bem próximas do ótimo e com um esforço computacional muito menor do que o utilizado pelo pacote de otimização CPLEX.

6.2 Trabalhos Futuros

Os trabalhos futuros podem estar concentrados em dois aspectos: no primeiro, considerando o modelo de otimização proposto, podem ser feitas extensões na formulação do PDCC-RSSF de forma a inserir o planejamento da densidade, cobertura e conectividade de forma dinâmica, conforme proposto por Nakamura [26] bem como uma nova extensão da formulação de forma a considerar uma rede hierárquica, conforme Oliveira [29]. Para solucionar estes novos modelos podem ser aplicados, entre outros, a relaxação Lagrangeana e métodos como Branch and Bound [5].

O segundo aspecto, consiste em procurar melhorar as soluções obtidas pela heurística. Embora a relaxação Lagrangeana tem demonstrado ser eficaz para a obtenção de bons limites superiores e inferiores, cabe ressaltar que nem sempre é possível encontrar soluções ótimas para o problema (conforme os experimentos computacionais do capítulo 5 mostraram). No caso em que o interesse seja sempre encontrar a solução ótima para o problema, ou *gaps* de dualidade muito pequenos, faz-se necessário recorrer a outros métodos e a procedimentos de pesquisa em árvore tais como o Branch and Bound [5]. Nesta técnica, os valores dos limites superior e inferior são utilizados para reduzir o espaço de busca através da poda de ramos da árvore. Quando o procedimento de pesquisa em árvore varre todo o espaço de busca, o procedimento garante a otimalidade da solução encontrada. Este procedimento pode também ser utilizado para melhorar a

qualidade da solução aproximada. Neste caso, a busca na árvore pode ser interrompida antes que se tenha garantia que no espaço de busca restante não haja nenhuma solução melhor do que a melhor solução encontrada.

Métodos alternativos podem ser utilizados em conjunto com o Branch and Bound ou de forma independente. Uma alternativa para obtenção de melhores limites inferiores são os métodos de plano de corte [4] visando os algoritmos de Branch and Cut e os métodos de feixe (bundle) [11] que podem ser utilizados no lugar do método de sub-gradiente. Independente ou em conjunto, podem ser utilizadas outras heurísticas para procurar melhorar o limite superior como: algoritmos genéticos [23], [10], grasp [31], entre outros.

Referências Bibliográficas

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, 2002.
- [2] Franz Aurenhammer. Voronoi diagrams a survey of a fundamental geometric data structure. *ACM Comput. Surv.*, 23(3):345–405, 1991.
- [3] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho. Grid coverage for surveillance and target location in distributed sensor networks. In *IEEE Transactions on Computers*, pages 51(12):1448–1453, 2002.
- [4] E.W. Cheney and A.A. Goldstein. Newton’s method for convex programming and Tchebycheff approximation. *Numerische Mathematik*, 1:253–268, 1959.
- [5] F.R.B. Cruz, G.R. Mateus, and J.M. Smith. A branch-and-bound algorithm to solve a multi-level network optimization problem. *Journal of Mathematical Modelling and Algorithms*, 2(1):37–56, 2003.
- [6] G. R. Mateus F. F. Mazzini and J. M. Smith. Lagrangean based methods for solving large-scale cellular network design problems. *Wireless Networks*, 9(6):659–672, 2003.
- [7] M. L. Fisher. The lagrangian relaxation method for solving integer programming problems. *Management Science*, 27:1–18, 1981.
- [8] J.L. Goffin and J.P. Vial. Convex nondifferentiable optimization: an interior point perspective. In *2nd School on Optimization*, Universidad de Chile, Santiago, Chile, 1998.
- [9] M. C. Goldberg. O problema de alocação ótima de radares de vigilância: Um estudo por técnicas de cobertura. Master’s thesis, Instituto Militar de Engenharia, 1987.
- [10] R. Haupt and S. Haupt. *Practical Genetic Algorithms*. Wiley-Interscience, 1 edition, January 1998.
- [11] J.B. Hiriart and C. Lemaréchal. *Convex Analysis and Minimization Algorithms*. Springer-Verlag, 1991.
- [12] C. Huang and Y. Tseng. The coverage problem in a wireless sensor network. In *Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications*, pages 115–121. ACM Press, 2003.

- [13] K. Langendoen and N. Reijers. Distributed localization in wireless sensor networks: a quantitative comparison. *Computer Networks*, 43:499–518, 2003.
- [14] C. Lemaréchal. Nondifferentiable optimization. In G.L. Nemhauser, A.H.G. Rinnooy Kan, and M.J. Todd, editors, *Handbooks in Operations Research and Management Science*, volume 1, pages 529–572. Elsevier Science Publishers B.V., 1989.
- [15] Xiang-Yang Li, Peng-Jun Wan, and Ophir Frieder. Coverage in wireless ad-hoc sensor networks. In *Proc.*, 2001.
- [16] A. Loureiro, L. Ruiz, R. Mini, and J. Nogueira. Rede de sensores sem fio. In *Simpósio Brasileiro de Computação - Jornada de Atualização de Informática*, 2002.
- [17] H. P. L. Luna M. C. Goldberg. *Otimização Combinatória e Programação Linear: Modelos e Algoritmos*. Campus, 2000.
- [18] D.S Johnson M. R. Garey. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., 1990.
- [19] G.R. Mateus and Z.K.G. Patrocínio Jr. An algorithm for large scale capacitated location problem in networks. In *IFAC/IFORS/IMACS Symposium Large Scale Systems: Theory and Applications*, pages 65–72, Beijing, P.R. China, August 1992.
- [20] S. Megerian, F. Koushanfar, G. Qu, G. Veltri, and M. Potkonjak. Exposure in wireless sensor networks: Theory and practical solutions. *Wireless Networks*, 8(5):443–454, 2002.
- [21] S. Megerian and M. Potkonjak. Low power 0/1 coverage and scheduling techniques in sensor networks. Technical Reports 030001, University of California, Los Angeles, January 2003.
- [22] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava. Coverage problems in wireless ad hoc sensor networks. In *INFOCOM*, pages 1380–1387, 2001.
- [23] M. Michell. *An Introduction to Genetic Algorithms (Complex Adaptive Systems)*. MIT Press, 1 edition, February 1998.
- [24] D. Estrin N. Bulusu, J. Heidemann. Adaptive beacon placement. In *Twenty-first International Conference on Distributed Computing Systems (ICDCS-21)*, pages 489–498. University of California, Los Angeles, IEEE Computer Society, April 2001.
- [25] L. Girod N. Bulusu, D. Estrin and J. Heidemann. Scalable coordination for wireless sensor networks: Self-configuring localization systems. In *Proceedings of the Sixth International Symposium on Communication Theory and Applications (ISCTA '01)*. University of California, Los Angeles, July 15-20th, 2001.
- [26] F.G. Nakamura. Planejamento dinâmico para controle de cobertura e conectividade em redes de sensores sem fio planas. Master's thesis, Universidade Federal de Minas Gerais, 2003.

- [27] G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. Wiley-Interscience, 1988.
- [28] D. Niculescu and B. Nath. Ad hoc positioning systems. In *IEEE GlobeCom*, pages 2926–2931, 2001.
- [29] A.L. Oliveira. Modelo e algoritmos para organização de RSSFs hierárquicas. Master's thesis, Universidade Federal de Minas Gerais, 2004.
- [30] Z.K.G. Patrocínio Jr. Um algoritmo exato para o problema de localização capacitado baseado no modelo de fluxo de custo mínimo. Master's thesis, Universidade Federal de Minas Gerais, 1993.
- [31] M.G. Ravetti. Problemas de sequenciamento com máquinas paralelas e tempos de preparação de máquinas dependentes da sequência. Master's thesis, Universidade Federal de Minas Gerais, 2003.
- [32] L. Ruiz, J. Nogueira, and A. Loureiro. MANNA: A management architecture for wireless sensor networks. *IEEE Communications Magazine*, 14(2), February 2003.
- [33] C. Savarese, K. Langendoen, and J. Rabaey. Robust positioning algorithms for distributed ad hoc wireless sensor networks. In *USENIX Technical Annual Conference*, pages 317–328, 2002.
- [34] A. Savvides, H. Park, and M. Srivastava. The bits and flops of the n-hop multilateration primitive for node localization problems. In *First ACM International Workshop on Wireless Sensor Networks and Application*, pages 112–121, 2002.
- [35] A. Savvides, S. Park, and M. B. Srivastava. On modeling networks of wireless microsensors. *ACM SIGMETRICS Performance Evaluation Review*, 29(1):318–319, 2001.
- [36] N. Shor. *Minimization Methods for Nondifferentiable Functions*. Springer-Verlag, Berlin, 1985.
- [37] D. Tian and N. D. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications*, pages 32–41. ACM Press, 2002.
- [38] S. Tilak, N. B. Abu-Ghazaleh, and W. Heinzelman. Infrastructure tradeoffs for sensor networks. In *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications*, pages 49–58. ACM Press, 2002.
- [39] S. Tilak, N. B. Abu-Ghazaleh, and W. Heinzelman. A taxonomy of wireless micro-sensor network models. *ACM SIGMOBILE Mobile Computing and Communications Review*, 6(2):28–36, 2002.
- [40] Marcos A. M. Vieira, Luiz Felipe M. Vieira, Linnyer B. Ruiz, Antonio A. F. Loureiro, Antonio O. Fernandes, and José Marcos S. Nogueira. Scheduling nodes in wireless sensor network: A voronoi approach. In *Proceedings of the 28th Annual IEEE Local Computer Networks (LCN'03)*, pages 423–429, Bonn/Königswinter, Germany, October 2003.

-
- [41] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill. Integrated coverage and connectivity configuration in wireless sensor networks. In *Proceedings of the First International Conference on Embedded Networked Sensor Systems*, pages 28–39. ACM Press, 2003.
- [42] Y. Xu, S. Bne, U. Mori, J. Heidemann, and D. Estrin. Topology control protocols to conserve energy in wireless ad hoc networks. Technical Report 6, University of California, Los Angeles, Center for Embedded Networked Computing, January 2003.