

UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

**ROTEAMENTO E AGREGAÇÃO DE DADOS BASEADO
NO RSSI EM REDES DE SENSORES SEM FIO**

MOYSÉS MENDES DE LIMA

Manaus
Março de 2013

MOYSÉS MENDES DE LIMA

**ROTEAMENTO E AGREGAÇÃO DE DADOS BASEADO
NO RSSI EM REDES DE SENSORES SEM FIO**

Dissertação apresentada ao Programa de
Pós-Graduação em Informática do Instituto
de Computação da Universidade Federal do
Amazonas como requisito parcial para a ob-
tenção do grau de Mestre em Informática.

**ORIENTADOR: HORÁCIO ANTÔNIO BRAGA FERNANDES DE
OLIVEIRA**

Manaus

Março de 2013

© 2013, Moysés Mendes de Lima.
Todos os direitos reservados.

A524p Lima, Moysés Mendes de
Roteamento e Agregação de Dados baseado no RSSI
em Redes de Sensores Sem Fio / Moysés Mendes de
Lima. — Manaus, 2013
x, 61 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal do
Amazonas

Orientador(a): Horácio Antônio Braga Fernandes de
Oliveira

Redes de Sensores sem Fio, Roteamento Guloso,
Agregação de Dados

CDU 1234

[Folha de Aprovação]

Quando a secretaria do Curso fornecer esta folha,
ela deve ser digitalizada e armazenada no disco em formato gráfico.

Se você estiver usando o `pdflatex`,
armazene o arquivo preferencialmente em formato PNG
(o formato JPEG é pior neste caso).

Se você estiver usando o `latex` (não o `pdflatex`),
terá que converter o arquivo gráfico para o formato EPS.

Em seguida, acrescente a opção `approval={nome do arquivo}`
ao comando `\ppgccufmg`.

Se a imagem da folha de aprovação precisar ser ajustada, use:
`approval=[ajuste] [escala] {nome do arquivo}`

onde *ajuste* é uma distância para deslocar a imagem para baixo
e *escala* é um fator de escala para a imagem. Por exemplo:

`approval=[-2cm] [0.9] {nome do arquivo}`
desloca a imagem 2cm para cima e a escala em 90%.

Agradecimentos

Agradeço em primeiro lugar ao meu Senhor e Salvador Jesus Cristo, que esteve ao meu lado durante a caminhada, que renovava as minhas forças quando eu pensava em desistir e me surpreendia a cada manhã com sua bondade e o seu amor infinito. A Ele a honra, a glória, a força e o poder.

Ao meu orientador, Professor Horácio Fernandes, por primeiramente acreditar em mim e me receber como aluno, por me proporcionar conforto emocional e oferecer seus préstimos com paciência e honestidade. Seus esclarecimentos, sua constante ajuda e principalmente o apoio em horas de crise tornaram possível a realização deste trabalho.

Ao meu co-orientador, Professor Eduardo Nakamura, pelos apoios e pelas ideias desafiadoras que contribuíram para a melhoria do trabalho realizado.

Ao Professor Loureiro da UFMG pelo apoio e por fazer parte deste trabalho.

Aos Professores Edjard, Edjair e Professora Rosiane de Freitas, pelas aulas inspirativas, carregadas de muito conhecimento e ministradas com autoridade.

Aos colegas do IComp na UFAM, pela hospitalidade, amizade e pelos cafés que tomamos juntos na Faculdade de Direito, cheios de companheirismo e boas conversas. Certamente estes momentos serão lembrados com muita alegria.

Aos meus pais, Moisés Xavier e Maria das Graças, pelo apoio, carinho e pelo incentivo aos estudos. Ao meu irmão Rildo, pelo companheirismo e palavras de força.

Ao meu filhinho querido Pedro, que por muitas noites teve que brincar sozinho e mesmo assim, com um sorriso no rosto dizia: “*Está tudo bem, pai. Pode fazer a tua tarefa!*”. Pelos abraços que recebia dele enquanto me debruçava em livros e esquemas gráficos em nossa sala de jantar. Amo você meu filho.

Finalmente, à minha esposa Etyanne, fiel companheira, auxiliadora idônea e parceira, por seu apoio incondicional, por seu correto proceder diante das crises e por não se deixar entristecer pelos momentos difíceis. Agradeço pela paciência e pelas incontáveis demonstrações de amor e cuidado. Amo você querida.

Resumo

Uma Rede de Sensores Sem Fio (RSSF) consiste em um conjunto de sensores, dispostos em um campo de sensoreamento em contato ou próximos a um evento ou fenômeno a ser sensoreado. Tais redes têm um grande impacto em aplicações que envolvem o monitoramento de condições ambientais tais como temperatura, luminosidade, movimento e presença de certos tipos de objetos. Nestas e na maioria das outras aplicações, os nós sensores possuem limitações de energia e largura de banda. Desta forma, técnicas de redução do consumo de energia e comunicação são necessárias em muitos cenários. Um dos objetivos principais no projeto de uma RSSF é prover a comunicação de dados entre os nós sensores e o nó *sink*. O desafio é prolongar a vida útil da rede mantendo uma qualidade na comunicação e na entrega dos dados. Um dos pontos-chaves observados em diversos estudos é que os protocolos de roteamento em RSSFs irão diferir de acordo com a aplicação e a arquitetura da rede. Recentemente, algoritmos que exploram informações geográficas, chamados Algoritmos de Roteamento Geográfico, têm sido propostos para o transporte de dados nas RSSFs por serem escaláveis, dinâmicos e possuírem uma alta taxa de entrega de dados. Estes algoritmos se baseiam na posição física dos nós da rede ao invés de seus endereços para criar rotas em direção ao nó *sink*. A principal desvantagem destes algoritmos é a necessidade de localização, que pode ser custoso em diversos aspectos. Neste trabalho, vamos além e propomos dois novos algoritmos de roteamento geográfico que não necessitam das informações de posição dos nós enquanto ainda proveem a funcionalidade de agregação de dados na rede. Nossa proposta tira vantagem da possibilidade de maior alcance de comunicação do nó *sink*, bem como do indicador de potência do sinal recebido (RSSI) dos nós sensores para configurar rotas e tempos de agregação de dados em direção ao nó *sink*. Os resultados obtidos mostram claramente que os algoritmos propostos reduzem a quantidade de dados redundantes e o número de transmissões na rede enquanto mantém todas as vantagens deste tipo de algoritmo.

Palavras-chave: Redes de Sensores Sem Fio, Roteamento, Agregação de Dados, RSSI.

Abstract

A Wireless Sensor Network (WSN) consists of a set of sensors arranged in a sensor field in contact or near an event or phenomenon to be monitored. Such networks have a major impact on applications involving the monitoring of environmental conditions such as temperature, light, motion and presence of certain types of objects. Accordingly in most other applications, the sensor nodes have limited power and bandwidth. Thus, techniques for reducing power consumption and communication are required in many settings. One of the main objectives in the design of a WSN is to provide data communication between the sensor nodes and the node textit sink. The challenge is to extend the network lifetime while maintaining a quality communication and data delivery. One of the key points observed in several studies is that routing protocols for WSNs will differ depending on the application and network architecture. Recently, algorithms that exploit information geographic, called Geographic Routing Algorithms, been considered as one of the most viable solution for routing in Wireless Sensor Networks (WSNs) due to their high scalability, dynamism, and high data delivery rate. These algorithms refer to nodes by position rather than address and use these coordinates to discover routes to the sink node. The main drawback of these algorithms is the need for position information of the nodes, which can be expensive in several ways. In this paper, we go further and propose two new routing algorithms that does not require position information while also adds data aggregation functionality to the network. Our proposal takes advantage of the grater communication range of the sink node as well as the Received Signal Strength Indicator (RSSI) of the sensor nodes to configure routing paths and aggregation times back to the sink node. Our results indicate clearly that the proposed algorithm reduces the amount of redundant data and the number of transmissions in the network while maintaining all advantages of these kind of algorithms.

Keywords: Wireless Sendor Networks, Routing, Data Aggregation, RSSI.

Lista de Figuras

2.1	Exemplo de agregação de dados. Nós A , B , C e D , repassando informações agregadas.	11
2.2	Encaminhamento Guloso. Nó S seleciona o nó A , que está mais próximo do destino D . O procedimento é repetido até a entrega do pacote ao destino D , normalmente um <i>Sink</i> . Baseado em (Park et al., 2010).	18
2.3	Exemplo de buraco de cobertura entre o nó x e o destino D . Baseado em (Karp, 2000).	19
2.4	Exemplo de rota no algoritmo BVGF. Baseado em (Xing et al., 2004). . .	20
2.5	No GeRaF, os nós enviam pacotes usando mensagens <i>broadcast</i> . Todos os nós dentro da área de cobertura irão receber o pacote e avaliar suas prioridades na tentativa de atuar como <i>relay</i> . Baseado em (Zorzi & Rao, 2004).	21
2.6	(a) Encaminhamento Guloso avançado. (b) LEAR em clusters. Baseado em (Alasem et al., 2011).	22
2.7	Desvio de buraco usando posicionamento virtual. (a) Nó 12 simulando um buraco tanto para Greedy quanto para MFR. (b) Caminho que pode ser encontrado através do algoritmo MFR-ViP. (c) Caminhos em volta do buraco. Baseado em (You et al., 2009).	26
2.8	Propagação do sinal RSSI. Nós mais distantes recebem consulta do <i>sink</i> com sinal baixo. Baseado (Boukerche et al., 2008).	28
3.1	(a) Potência do sinal decrescendo à medida que a mensagem de consulta do <i>sink</i> se propaga na rede. (b) Nós sensores localizados mais distantes do <i>sink</i> receberão a mensagem com RSSI menor que os nós mais próximos . .	32
3.2	Gráfico de Fluxo dos Algoritmos GRDA	33

3.3	Exemplo de funcionamento do algoritmo <i>GRDA Selection</i> . Após disseminação da consulta pelo <i>sink</i> (a), nós regulares iniciam processo de escolha do próximo salto, com base na distância dos vizinhos em relação ao <i>sink</i> (b). Em seguida, nó encaminha pacote ao vizinho vizinho mais próximo do <i>sink</i> (c).	35
3.4	Exemplo de funcionamento do algoritmo <i>GRDA Election</i> . Após disseminação da consulta pelo <i>sink</i> e cálculo da distância, nós regulares iniciam processo de transmissão de pacotes via mensagem única de <i>broadcast</i> (a), auto-eleição do próximo salto (b) e repasse de pacotes (c).	37
4.1	Avaliação de performance em relação à escala da rede.	43
4.2	Avaliação de performance em relação à densidade da rede.	45
4.3	Avaliação de performance em relação à imprecisão do RSSI a curta distância.	47
4.4	Avaliação de performance em relação à imprecisão do RSSI a curta distância.	48
4.5	Avaliação de performance em relação à imprecisão do RSSI a curta distância.	50

Lista de Tabelas

2.1	Classificação dos Protocolos de Roteamento	15
2.2	Classificação dos algoritmos estudados	29
4.1	Cenário padrão de simulação.	42

Sumário

Agradecimentos	v
Resumo	vi
Abstract	vii
Lista de Figuras	viii
Lista de Tabelas	ix
1 Introdução	1
1.1 Motivação	1
1.2 Objetivos	3
1.3 Principais Contribuições	4
1.4 Organização da Dissertação	4
2 Fundamentação Teórica	6
2.1 Fusão de Dados em RSSF	9
2.1.1 Importância da Fusão de Dados	10
2.1.2 Agregação de Dados	10
2.2 Flooding	11
2.3 Estimativa de Distância Baseada no RSSI	12
2.4 Visão Geral sobre Protocolos de Roteamento	14
2.5 Protocolos de Roteamento Geográfico	16
2.6 Classificação dos Algoritmos de Roteamento	18
2.6.1 Algoritmos Baseados em Coordenadas Geográficas	18
2.6.2 Algoritmos Baseados em Coordenadas Virtuais	23
2.6.3 Algoritmos Livres de Posicionamento	27
2.7 Considerações Finais	29

3	Roteamento e Agregação de Dados baseado no RSSI	32
3.1	Definição do Problema	34
3.1.1	GRDA Selection	35
3.1.2	GRDA Election	37
3.2	Considerações Finais	39
4	Metodologia e Avaliação de Desempenho	41
4.1	Metodologia	41
4.2	Avaliação de Desempenho	43
4.2.1	Impacto da Escala da Rede	43
4.2.2	Impacto da Densidade da Rede	44
4.2.3	Impacto da imprecisão do RSSI a curta distância	45
4.2.4	Impacto no Raio de Comunicação	46
4.2.5	Erros na Escolha do Próximo Salto	48
4.3	Considerações Finais	49
5	Conclusões	52
5.1	Aplicabilidade da Proposta	53
5.2	Desafios e Trabalhos Futuros	54
	Referências Bibliográficas	55

Capítulo 1

Introdução

1.1 Motivação

Uma Rede de Sensores sem Fio (RSSF) consiste em um grande conjunto de sensores, dispostos estrategicamente em um campo de sensoreamento em contato ou próximos a um evento ou fenômeno a ser sensoreado. O alto número de sensores neste tipo de rede permite a cobertura e sensoreamento de grandes regiões geográficas, aumentando a precisão da coleta.

As RSSFs têm grande impacto no campo das aplicações militares como reconhecimento de imagens em um campo de batalha, detecção de intrusão, monitoramento de condições ambientais tais como temperatura, luminosidade, movimento e presença de certos tipos de objetos. Na maioria das aplicações, os nós possuem limitações de energia e largura de banda. Desta forma, técnicas de otimização do consumo energético e comunicação são necessárias em muitos cenários (Al-karaki & Kamal, 2004), (Zheng & Jamalipour, 2009).

Redes de sensores podem sensorear diferentes tipos de informações como, térmicas, acústicas, infravermelho, monitoramento de condições ambientais diversas incluindo temperatura, umidade, movimento veicular, variações de luminosidade, pressão, composição do solo e características como velocidade, direção e tamanho de objetos (Akyildiz et al., 2002b). Por questões de economia de energia, o conjunto de informações coletado no campo de sensoreamento precisa ser enviado através de um esquema multi-salto em direção ao *sink*, ou nó sorvedouro. Este, é equipado com mecanismos de processamento e tratamento dos dados sensoreados. O transporte dos dados coletados pelo nó sorvedouro é uma das mais importantes tarefas em RSSFs.

Um dos objetivos principais em um projeto de rede de sensores é prover a comunicação de dados entre os nós sensores. O desafio é prolongar a vida útil da rede

além de prever e tratar a degradação na conectividade através de técnicas de gerenciamento de energia. Ao longo dos anos, pesquisadores têm observado fatores que influenciam diretamente no desenvolvimento de protocolos de roteamento como a implantação do nó sensor, consumo de energia sem perda de precisão de coleta, comunicação estável/heterogênea de dados e tolerância a falhas (Al-karaki & Kamal, 2004), (Akkaya & Younis, 2005). Esse conjunto de fatores deve ser superado ainda na fase de concepção do projeto, a fim de permitir comunicação eficaz entre os nós da rede.

O roteamento em RSSFs difere do roteamento convencional em que uma rede utiliza um sistema de endereçamento definido através do Protocolo de Internet (IP). Em RSSFs não há uma infra-estrutura definida e normalmente os links sem fio não são confiáveis. Além disso, os nós podem falhar e neste caso, os protocolos de roteamento precisam encontrar meios de atender requisitos mínimos que possibilitem um correto funcionamento da rede. Segundo (Al-karaki & Kamal, 2004), o desenvolvimento de protocolos de roteamento é influenciado por diversos fatores desafiadores que devem ser superados antes do seu objetivo maior, que é a comunicação eficiente entre os nós.

Considerado com uma das mais importantes abordagens para o roteamento de pacotes em redes sem fio, o roteamento geográfico considera o destino final de um pacote como os dados serão encaminhados a esse destino (Kim et al., 2005), (Nati, 2008). Diferente do modelo tradicional, baseado no próximo salto ou no repasse nó-a-nó de um pacote, no roteamento geográfico os pacotes são encaminhados com base nas coordenadas geográficas do nó de origem, de destino e os nós intermediários, que irão participar da rota criada até o destino. De forma resumida, os pacotes são encaminhados ao vizinho que está geograficamente mais próximo do destino, normalmente um *sink*.

A utilização roteamento geográfico é especialmente atrativa em redes sem fio alocadas em grandes áreas e que utilizem o esquema multi-salto. Esse protocolo permite que cada nó determine um encaminhamento de pacotes no próximo salto usando técnicas de roteamento guloso. Estas, são baseadas na manutenção de rotas até o destino. Um dos elementos mais importantes na maioria das técnicas de roteamento geográfico é o mecanismo de *Greedy Forwarding*, ou encaminhamento Guloso, que pode garantir a entrega dos pacotes e alcançar o mesmo desempenho dos algoritmos de caminho mínimo. Segundo (Seada et al., 2004), o encaminhamento guloso pode ser um método eficaz para reduzir o custo de entrega de pacotes quando assumimos que a densidade da rede é suficiente, a localização dos nós é precisa e a comunicação entre os nós é confiável, independente do alcance do sinal.

Um dos pontos fracos do roteamento geográfico é a necessidade de localização dos nós, que pode ser custoso e susceptível a grandes erros em RSSFs (Souza et al., 2010).

Além disso, tais algoritmos em geral não possuem agregação de dados, o que os tornam inviáveis para RSSFs em que grande parte dos nós possuem dados a serem enviados ao *sink*. Para sobrepor tal limitação, propomos um novo algoritmo de roteamento geográfico com agregação de dados que não requer informações de posicionamento dos nós: *GRDA* (*Greedy Routing and Data Aggregation*). A ideia principal do algoritmo proposto é tirar proveito da possibilidade de equipar o nó *sink* com um dispositivo de comunicação de maior potência para alcançar todos os nós da rede em um único salto. Entretanto, a resposta/retorno dos nós sensores para o nó *sink* é realizada através de múltiplos saltos utilizando nossa abordagem de roteamento guloso que tira proveito do RSSI (*Received Signal Strength Indicator*) para calcular o próximo salto do roteamento em direção ao nó *sink* bem como para calcular o tempo de espera para possibilitar a agregação de dados.

A ideia incentivadora para esta arquitetura de comunicação surgiu a partir do projeto ATTO (*Amazonian Tall Tower Observatory*), que já está implementado na Amazônia e consiste em uma torre no meio da floresta com uma estrutura física de 320 m de altura e alta capacidade de comunicação. O foco do projeto é mapear eventos na região e fornecer medidas confiáveis de fontes e sumidouros de gases de efeito estufa como CO_2 , CH_4 e N_2O (Tollefson, 2010).

1.2 Objetivos

O principal objetivo deste trabalho é propor e demonstrar uma solução eficiente para roteamento geográfico guloso e fusão de dados em redes de sensores sem fio que tire proveito da grande capacidade de comunicação do nó *sink* em alcançar todos os nós regulares da rede em um único salto.

Para atingir este objetivo, os seguintes objetivos específicos devem ser cumpridos:

1. *Ampliar a utilização da abordagem de encaminhamento guloso em roteamento geográfico para utilização com agregação de dados;*
2. *Demonstrar que a utilização de nossa abordagem de encaminhamento guloso para o envio de pacotes, reduz a quantidade de pacotes da rede e, consequentemente, o consumo de energia;*
3. *Aprimorar uma das abordagens atuais de roteamento geográfico guloso através da implementação de um tempo de espera para o envio dos pacotes.*

1.3 Principais Contribuições

O desenvolvimento e avaliação dos dois novos algoritmos de roteamento geográfico que não necessitam de informações de posição dos nós enquanto ainda proveem a funcionalidade de agregação de dados na rede, traz as seguintes contribuições:

1. **Survey sobre técnicas de encaminhamento guloso para RSSFs.** Além de prover uma ampla visão do estado-da-arte em roteamento geográfico, nos permitiu identificar questões em aberto serviu como ponto de partida para pesquisa.
2. **Roteamento e Agregação de Dados baseado no RSSI:** O objetivo desta contribuição é mostrar como implementar agregação de dados em algoritmos de roteamento guloso, utilizando o RSSI do *Sink* como métrica de encaminhamento para o próximo salto. A presente proposta tira proveito da possibilidade de se equipar o nó *sink* com um dispositivo de comunicação de maior potência, de forma que, em um único salto, o nó *sink* pode enviar uma consulta para toda a rede

1.4 Organização da Dissertação

O Capítulo 2, apresenta uma visão geral sobre RSSFs, mostrando aspectos gerais e alguns desafios de projeto. Além disso, identificamos e apresentamos pontos importantes sobre o roteamento geográfico e as diversas técnicas de encaminhamento geográfico guloso. Um breve comentário sobre fusão de dados e sua importância para as RSSFs também é apresentado. Neste contexto, damos ênfase à técnica de agregação de dados, utilizada para reduzir o número de informações redundantes, além de reduzir o consumo de energia da rede. Apresentamos um breve comentário sobre o protocolo *Flooding*, além de uma explanação breve sobre a técnica de estimativa de distância baseada no RSSI, técnica utilizada neste trabalho para estimar a distância entre dois nós baseado na força do sinal recebido pelo outro nó.

No Capítulo 3, nós propomos um novo algoritmo de Roteamento e Agregação de Dados baseado no RSSI, com suas duas variações: o *GRDA Selection* e o *GRDA Election*. A proposta do algoritmo é possuir as vantagens de um algoritmo de roteamento geográfico tais como robustez, escalabilidade, dinamicidade e entrega confiável dos dados, mas sem precisar das informações de posicionamento dos nós ao mesmo tempo que provê o recurso de agregação de dados na rede. A idéia sobre os algoritmos surgiu a partir projeto ATTO (*Amazonian Tall Tower Observatory*), que já está implementado na Amazônia e consiste em uma torre no meio da floresta com uma estrutura física

de 320 m de altura e alta capacidade de comunicação, que irá realizar diversas medições no cenário amazônico.

O Capítulo 4, apresenta a metodologia adotada neste trabalho assim como a avaliação de desempenho do algoritmo proposto. O foco de nossa avaliação é a aplicação da nova técnica de encaminhamento guloso bem como na técnica de agregação de dados, baseada na métrica para o cálculo do tempo de espera para agregação. Também comparamos a performance dos algoritmos *GRDA* com o algoritmo *Flooding+Aggregation*, que usa o *flooding* clássico para divulgar a consulta do *sink* em múltiplos saltos. O retorno dos nós é feito usando a árvore inversa do *flooding* e ainda realiza agregação de dados nos nós que possuem filhos.

Finalmente, no Capítulo 5 apresentamos as conclusões, aplicabilidade da solução apresentada e as propostas para a continuidade do presente trabalho.

Capítulo 2

Fundamentação Teórica

As Redes de Sensores sem Fio (RSSFs), proporcionaram um novo paradigma para o sensoreamento e disseminação de informações, a partir de ambientes variados, com potencial para servir a muitas aplicações. Este tipo especial de rede *ad hoc*, se tornou bastante popular devido à sua aplicabilidade em diversas áreas como, meio ambiente, saúde, industrial, agrícola, espacial e militar. Além destas, várias propriedades físicas podem ser monitoradas como, temperatura, umidade, pressão, luz ambiente e movimento. Normalmente, as RSSFs possuem um grande número de nós sensores distribuídos, possuem restrições de energia, e devem possuir mecanismos de para auto-configuração e adaptação, devido a problemas como falha na comunicação e perda de nós (Loureiro et al., 2003), (Yoneki & Bacon, 2005), (Akyildiz et al., 2002a).

A principal tarefa de um nó sensor é sensorear e coletar dados a partir de um certo domínio ou região de sensoriamento. Os nós ainda são responsáveis pelo processamento e encaminhamento das informações até o nó *sink*, onde a aplicação reside. A maioria das RSSFs possuem diferentes requisitos de aplicação. Neste sentido, é praticamente impossível considerar todos os aspectos que podem ser explorados em uma única rede. Diante deste fato, apenas uma parte destes é considerada na fase de projeto e implementação. Um projeto de RSSFs é influenciado por muitos fatores, que incluem (Akyildiz et al., 2002b), (Zheng & Jamalipour, 2009):

- **Tolerância a falhas:** Alguns sensores podem falhar ou serem impedidos de funcionar durante uma eventual perda de energia, dano físico ou interferências do ambiente de sensoreamento. Contudo, uma eventual falha em um sensor não pode afetar o funcionamento global de uma RSSF. Esta característica é chamada de tolerância a falhas, que é a capacidade de manter as funcionalidades da rede, mesmo em eventuais falhas de nós sensores.

- **Escalabilidade:** A quantidade de sensores monitorando um fenômeno, pode chegar à casa das centenas e em alguns casos dos milhares. Dependendo da aplicação, esse número pode chegar facilmente à casa dos milhões. Neste cenário, a densidade (quantidade de nós por metro quadrado) pode variar de poucos nós sensores para algumas centenas em uma dada região. Desta forma, as RSSFs devem considerar densidades altas, permitindo a operação da aplicação e utilização de esquemas de encaminhamento.
- **Custo de operação:** Considerando que uma RSSFs pode ser composta por um grande número de nós sensores, o custo de um único nó é importante para justificar o custo global de toda a rede. Se o custo da rede é mais alto que a implantação de sensores tradicionais, esta rede é economicamente justificável. Desta forma, o custo de cada nó sensor tem de ser mantido baixo. Um nó sensor pode ser implementado com algumas unidades adicionais, como sensores diversos e unidades de processamento. Além disso, pode haver necessidade de equipá-lo com um sistema de localização, como um GPS, e até mesmo módulo de energia, dependendo da aplicação utilizada.
- **Topologia da rede:** Centenas de nós sensores são implantados por todo o campo de sensoriamento, normalmente a dezenas de metros um do outro. A densidade de rede pode ser tão elevada quanto o número $20 \text{ nós}/m^3$. Nós sensores podem ser lançados de aviões, via foguetes ou mísseis, ou ainda colocados no campo de sensoriamento por um humano, em locais pré-determinados. Uma vez presentes no ambiente, as RSSFs podem sofrer alterações de topologia devido a fatores como: destruição dos sensores pelo meio ambiente, falhas de funcionamento devido ao escoamento de bateria, sensores podem ser desligados e ligados para fins de economia de energia, novos sensores podem ser acrescentados à rede e pode-se ter intermitência na comunicação devido a ruídos ou obstáculos no ambiente. Por este motivo, a implantação de um elevado número de nós um exige tratamento cuidadoso da topologia.
- **Consumo de energia:** Nós sensores são dispositivos autônomos, equipados com capacidades de sensoriamento, processamento e comunicação. São normalmente formados por cinco componentes básicos: transceptor, memória, processador, sensor e bateria. Um dos componentes mais importantes é a bateria, pois é o responsável pela manutenção do funcionamento de um nó sensor na maior parte do tempo. Neste contexto, o tempo de vida de um nó sensor depende da quantidade de energia disponível e, conseqüentemente, a quantidade de energia

consumida por um nó sensor. O funcionamento incorreto de alguns nós pode causar significativas alterações topológicas e exigir redirecionamento de pacotes, além da reorganização da rede. Assim, a conservação e o gerenciamento de energia assumem uma importância crucial para o projeto de RSSFs.

Do ponto de vista das estratégias de roteamento, a energia é apontada como o recurso mais escasso e que demanda a maior parte das atenções em RSSFs. Além disso, é considerada como fator essencial para o tempo de vida na grande maioria das soluções atuais. Neste sentido, um dos objetivos mais importantes do projeto de RSSFs é minimizar o consumo de energia do nó e maximizar a vida útil da rede. A natureza da energia limitada de nós sem fio requer o uso de estratégias eficientes de energia para maximizar a vida útil da rede (Mahfoudh & Minet, 2008), tais como:

- **Escoamento eficiente de energia:** Utilizado para minimizar a energia consumida através da transmissão fim-a-fim do pacote, para evitar os nós com um baixo consumo de energia residual e reduzir o número de transmissões sem sucesso.
- **Agendamento dos nós para *sleeping state* (modo dormindo):** Permite a mudança do estado de um nó de ativo para dormindo, com o objetivo de poupar energia, desde que a execução das aplicações e a funcionalidade de rede não sejam afetadas.
- **Topologia de controle por meio do ajuste da potência de transmissão:** A estratégia é utilizada para encontrar a potência de transmissão ideal que um nó necessita minimizar o consumo de energia, além de melhorar a conectividade de rede durante o ciclo de vida da rede.
- **Redução do volume de informações transmitidas:** Essa redução pode ser alcançada através da fusão de dados com a formação ou não de *clusters*, redução na frequência da atualização de informações da rede, bem como o não envio (ou diminuição) de informações para nós não participantes do encaminhamento.

Esses fatores são importantes porque servem como um guia para projetar protocolos e um algoritmos para RSSFs. Devido a tantas características, muitos algoritmos têm sido propostos para resolver o problema de como encaminhar dados em RSSFs. Os algoritmos de roteamento, como são chamados, além das características apresentadas, precisam considerar os requisitos de arquitetura e aplicação.

No restante do capítulo, será apresentada uma revisão da literatura compreendendo os conceitos sobre roteamento e seu relacionamento com técnicas de encaminhamento geográfico e agregação de dados em RSSFs. A seção 2.1 apresenta uma visão

geral sobre fusão de dados em RSSFs, sua importância, além de um tópico sobre o tema agregação de dados. Na seção 2.2, apresentamos um breve comentário sobre o protocolo *flooding* e novas técnicas de utilização deste protocolo em RSSFs. A seção 2.3 apresenta conceitos relacionados à estimativa de distância baseada no RSSI, bem como o modelo utilizado neste trabalho. Uma visão geral sobre protocolos de roteamento é apresentada na seção 2.4, enquanto na seção 2.5, os conceitos sobre protocolos de roteamento geográfico são destacados. Na seção 2.6, classificamos os algoritmos de roteamento em três categorias, para melhor entendimento, e finalmente na seção 2.7, apresentamos nossas considerações sobre o capítulo.

2.1 Fusão de Dados em RSSF

Fusão de Dados é uma importante tecnologia em RSSFs. A abordagem tem sido apontada como alternativa para o pré-processamento os dados de forma distribuída, aproveitando a capacidade de processamento dos nós sensores. Algoritmos de fusão de dados são projetados para lidar com um grande número de dados, de forma a remover informações redundantes e minimizar a quantidade de dados a serem transferidos (Loureiro et al., 2003), (Li & Wei-jia, 2011).

As RSSFs são consideradas redes centradas em dados, ou seja, o interesse nos dados sensoreados não se restringe à aplicação sendo comum a todas as atividades que possam tirar proveito da correlação existente entre estes dados. Assim, as atividades como roteamento devem permitir que os dados sejam analisados no nível da aplicação para decidir de estes serão retransmitidos, fundidos ou suprimidos.

Os métodos mais comuns de fusão de dados (Nakamura, 2007), (Abdelgawad & Bayoumi, 2012) são: (1) agregação, (2) inferência, (3) estimação. Os métodos de agregação são os mais simples e produzem como resultado um dado de menor representatividade do que o conjunto dos dados utilizados na fusão. A vantagem destes métodos reside na redução do volume de dados que trafegam pela rede e inclui operações de agregação como média, máximo, e mínimo. Os métodos de inferência têm como objetivo processar dados e tirar conclusões a respeito dos mesmos. A decisão é tomada com base no conhecimento da situação percebida. Exemplos destes métodos são: Fusão Bayesiana e Filtros Dempster-Shafer. Já os métodos de estimação têm como objetivo estimar o vetor de estado de um processo a partir de um vetor ou sequência de vetores de medições de sensores. Exemplos destes métodos são: *quadrados mínimos*, *quadrados mínimos ponderados* e *Filtros de Kalman*.

2.1.1 Importância da Fusão de Dados

Um dos pontos fundamentais no desenvolvimento de algoritmos de fusão de dados, está no tratamento adequado de informações redundantes. Este requisito precisa ser atendido, afim de se obter dados relevantes e ainda: reduzir a quantidade de transmissão de dados, reduzir a colisão de dados durante a transmissão, reduzir o congestionamento da rede, economizar energia de um nó e além disso, prolongar o tempo de vida da rede. Em (Akkaya & Younis, 2005), a fusão de dados é apresentada como estratégia onde um nó é capaz de produzir um sinal de saída mais preciso, utilizando técnicas de combinação dos sinais de entrada e redução do ruído nos sinais recebidos. A acurácia alcançada na entrega dos dados ao *sink* é apontada como benefício da fusão de dados em (Lin et al., 2010). Neste sentido, a acurácia é medida pelo total de dados enviados por todos os nós durante o ciclo de vida da rede.

A fusão de dados possui pelo menos dois fatores que tornam importante a sua utilização em RSSFs. O primeiro consiste na leitura de maior precisão, tornando a rede mais robusta e menos vulnerável a falhas e imprecisões de um único nó sensor. O segundo, é a economia de energia através da redução da quantidade mensagens e dados que são transmitidos pelos nós sensores (Loureiro et al., 2003).

2.1.2 Agregação de Dados

Uma forma especial de fusão de dados é chamada de agregação, onde dados originados de diferentes fontes são combinados através da eliminação de duplicidades. Algumas destas estratégias podem ser originadas a partir de cada nó ou serem conduzidas por nós específicos na rede. Os dados coletados a partir dos nós sensores, são derivadas de fenômenos comuns normalmente sensoreados em uma área específica. Neste contexto, a agregação de dados consiste em suprimir a redundância nas mensagens de dados diferentes, além de reduzir o consumo de energia da rede. Em (Al-karaki & Kamal, 2004), a agregação de dados é apresentada como técnica de roteamento essencial para a estratégias envolvendo formação de clusters e diminuição da dados redundantes, pois determinadas aplicações geram grande quantidade de tráfego de dados. Já em (Yoneki & Bacon, 2005), agregação em RSSFs é apontada como estratégia de combinação ou sumarização de informações de alguns ou todos os sensores.

Em (Cohen et al., 2001), agregação de dados é apresentada como um processo que inclui uma coleção de dados, a composição flexível e programada destes dados em um novo dado refinado de menor volume e a entrega destes dados agregados ao destino final. A técnica, é usada para resolver problemas de implosão e sobreposição no roteamento centrado em dados. Nesta técnica, uma rede de sensores é geralmente entendida como

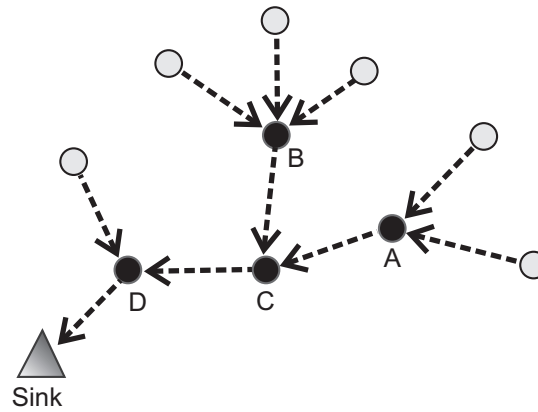


Figura 2.1. Exemplo de agregação de dados. Nós *A*, *B*, *C* e *D*, repassando informações agregadas.

uma árvore *multicast* inversa. Neste contexto, o nó *sink* inicia o roteamento através de uma consulta, divulgada em múltiplos saltos, solicitando as informações sensoreadas. Então, os dados provenientes dos múltiplos nós sensores são agregados e enviados ao destino final. Em cenários onde vários nós sensores possuem informações semelhantes, agregação de dados é utilizada para diferenciar dados simplesmente coletados de dados ou informações úteis (Abbasi & Younis, 2007). Esta característica é fundamental para o processamento das informações e o repasse dos dados agregados.

Neste trabalho, utilizamos a agregação de dados com forma de reduzir o número de informações redundantes e aumentar o tempo de vida da rede, também proporcionado pela economia de energia que é um benefício direto da agregação de dados.

2.2 Flooding

Flooding é um protocolo de roteamento de simples implementação em que cada nó precisa repassar via mensagens de *broadcast* cada mensagem recebida dos seus vizinhos, não necessitando para isso de informações sobre a topologia ou análise apurada das informações da rede. O *flooding* é um protocolo confiável, pois independentemente de quaisquer alterações na rede, os dados podem ser transmitidos ao destino final, neste caso um *sink*, desde que a rota entre o nó de origem e o nó *sink* exista. Apesar do elevado consumo de energia, gerado pelo re-envio de mensagens de *broadcast* a todos os vizinhos e geração grande quantidade de informações redundantes durante encaminhamento das mensagens, o protocolo normalmente apresenta alto desempenho na entrega de pacotes, pela mesma razão (Heinzelman et al., 1999), (Shin et al., 2007). A quantidade de mensagens de enviadas a rede, causada pela utilização de um *flooding*,

pode causar um problema *chamado de broadcast storm* (tempestade de mensagens de broadcast). Em (Agarwal et al., 2011), são apresentadas três deficiências do protocolo de roteamento *flooding*:

- **Implosão:** Uma vez que no esquema de *flooding* o repasse de pacotes é realizado através de mensagens de *broadcast*, o mesmo pacote pode chegar ao mesmo nó, através de diferentes caminhos. Quando um nó sensor recebe um pacote, não irá checar se o mesmo pacote já recebido antes. Esta característica permite o envio de pacotes duplicados para o mesmo nó.
- **Sobreposição:** Quando dois nós sensores detectam o mesmo evento, ambos podem encaminhar o evento até o *sink*, o que também permite a ocorrência de duplicidade de informações sendo enviadas ao *sink*.
- **Cegueira de recursos (resource blindness):** Em um *flooding* clássico, os nós não modificam seus estados de atividades com base na quantidade de energia disponível, mesmo se não possuem energia suficiente para a operar.

Recentemente, novas formas de utilização do protocolo *flooding* têm sido implementadas como forma de minimizar o número de transmissões cegas (*blind transmission*) de dados e o número de nós envolvidos no processo de encaminhamento das mensagens. Em (Hassanzadeh et al., 2011), a redução do número de mensagens é tratada através da utilização de chaves, definidas a partir de uma estação base, configuradas em um arquivo de mapeamento de chaves. Durante o *flooding*, cada nó recebe um pacote contendo a informação *chave/nó* e utiliza essa informação para definir que chave utilizar para o re-envio da informação via *broadcast*.

Em (Zhao et al., 2009), um mecanismo de *flooding* direcionado, de acordo com o número de saltos é apresentado. Nesta abordagem um algoritmo distribuído de contagem de saltos é utilizado para encaminhar pacotes de dados de forma controlada, mantendo a confiabilidade no roteamento e oferecendo economia de energia.

Apesar da alta taxa de entrega alcançada neste protocolo, o alto consumo de energia necessário para a correta implementação do *flooding* diminui o tempo de vida da rede, dificultando sua utilização em redes densas, onde há um maior número de vizinhos que irão re-enviar o pacote recebido.

2.3 Estimativa de Distância Baseada no RSSI

Em RSSFs, diferentes métodos podem ser utilizados para estimar distâncias e ângulos entre dois nós. Neste contexto, os sistemas de localização utilizam métodos que incluem

o tempo de chegada e diferença do tempo de chegada (ToA/TDoA), ângulo de chegada do sinal (AoA) e o RSSI (*Received Signal Strength Indicator*), ou Indicador de Potência do Sinal Recebido (Boukerche et al., 2007), (Zhu et al., 2009).

O RSSI é uma das métricas mais utilizadas em estudos referentes às RSSFs. O RSSI pode ser usado para estimar a distância entre dois nós baseado na força do sinal recebido pelo outro nó. Um nó que está transmitindo informação envia um sinal com uma certa força que irá decrescer conforme a propagação. Quanto maior a distância do nó que recebe a informação, menor será a força do sinal. Esta estratégia tem sido amplamente utilizada como métrica para aferição da potência de sinal, recebida pelos nós.

Existem diversos modelos de propagação de ondas de rádio para redes sem fio. Estes, podem prever a intensidade média do sinal recebido a determinada distância do transmissor, além da variabilidade da intensidade do sinal em áreas muito próximas a um local específico. Segundo (Benkic et al., 2008), três são os modelos mais utilizados em RSSFs:

- **Modelo de propagação no espaço livre:** a potência recebida por uma antena, colocada em uma distância de separação d de uma antena transmissora, é dada pela conhecida equação do espaço livre Friis (Rappaport, 2002),

$$P_r(d) = C_f \frac{P_t}{d^2}$$

onde P_t é a potência transmitida, $P_r(d)$ a potência recebida, e C_f Uma constante de transmissão. Atualmente, muitos dispositivos de sensoriamento são equipados com circuitos para medição de valores de RSSI, como *Smartphones* e *Tablets*.

- **Modelo de reflexão no solo (modelo de dois raios):** considera o caminho direto e um caminho de propagação refletido no solo entre o transmissor e o receptor,

$$P_r(d) = C_t \frac{P_t}{d^4}$$

onde C_t representa as características da transmissão para o modelo.

- **Modelo Log-Distância:** derivado de métodos empíricos e analíticos, o modelo considera que a potência média recebida irá decrescer, de forma logarítmica, em relação à distância do transmissor. O modelo é bastante utilizado em simulações

envolvendo RSSFs. A simplificação do modelo é dada pela fórmula,

$$P_r(d) \propto \frac{P_t}{d^\alpha}$$

onde α é chamado de gradiente distância-potência. O modelo é amplamente discutido em (Santi, 2005) e (Jacinto, 2012).

Em condições ideais, o valor do RSSI pode ser usado para o cálculo da distância estimada entre um transmissor e um receptor. Em (Jacinto, 2012), o RSSI é apresentado como uma das melhores métricas de qualidade do canal em zonas de recepção com baixa ou alta taxa de recepção e acima do limiar de sensibilidade da antena. A confiabilidade do RSSI, utilizado como métrica também é apresentada em (Holland et al., 2006). Nesta abordagem, o RSSI, que é comparado à técnica LQI (*Link Quality Indication*), apresenta uma margem de erro muito inferior a este último.

Neste trabalho, o RSSI foi utilizado para estimar a distância entre os nós regulares e o *sink*. A aferição é baseada na potência do sinal recebido, de forma que nós mais distantes do *sink* recebam menor valor de RSSI (devido à propagação do sinal).

2.4 Visão Geral sobre Protocolos de Roteamento

O roteamento é o processo de escolha de um caminho válido, entre a origem e o destino, conforme a solicitação de transmissão recebida. Em RSSFs, o roteamento difere das redes convencionais (que utilizam endereços IP) em vários aspectos. Não há infraestrutura definida, os nós estão sujeitos a falhas e restrições de energia. Por esta razão, os protocolos de roteamento para RSSFs precisam atender um conjunto de requisitos como: autonomia, eficiência energética, escalabilidade, resiliência, heterogeneidade (em relação aos sensores) e mobilidade (Ahmed et al., 2003), (Villalba et al., 2009) e (Agarwal et al., 2011).

Muitos protocolos de roteamento têm sido desenvolvidos para RSSFs. Em (Goyal & Tripathy, 2012), os protocolos de roteamento são classificados em cinco (05) categorias: Protocolos baseados em Localização, Centrados em Dados, Hierárquicos, baseados em Caminhos Múltiplos e baseados em QoS. Um resumo com as informações sobre cada protocolo é apresentado na Tabela 2.1.

Quando o nó *sink* encontra-se distante ou fora da faixa de comunicação de um nó de origem, a estratégia de encaminhamento do pacote através de múltiplos saltos deve ser utilizada. Desta forma, nós intermediários irão repassar pacotes aos vizinhos a salto de distância (dentro da faixa de comunicação deste nó). Em muitas aplicações de

Tabela 2.1. Classificação dos Protocolos de Roteamento

Protocolo	Descrição
Baseados em Localização	Nós sensores são endereçados através de suas informações de localização, necessárias para o cálculo da distância entre dois nós.
Centrados em Dados	Difere dos protocolos tradicionais na maneira em que os nós enviam suas informações ao <i>sink</i> . Quando um nó envia seu dado ao <i>sink</i> , sensores intermediários podem realizar agregação, nos dados originados de múltiplos sensores, para então encaminhar os dados agregados até o <i>sink</i> .
Hierárquicos	Protocolos organizados em uma hierarquia, em que a rede é composta por agrupamentos (<i>clusters</i>) de sensores. Cada grupo é gerenciado por um nó específico, chamado <i>cluster head</i> , responsável pelas atividades de transmissão de dados de dos os membros do grupo.
Baseados em Múltiplos Caminhos	Considera que durante uma transmissão de dados até o <i>sink</i> , dois paradigmas são seguidos: roteamento <i>single-path</i> e <i>multipath</i> . No primeiro, cada nó envia seu dado ao <i>sink</i> através do menor caminho. Já no segundo, cada sensor (origem) encontra do primeiro k menor caminhor at o <i>sink</i> , dividindo sua carga entra os caminhos encontrados.
Baseados em QoS	Em RSSFs, muitas soluções consideram a qualidade de serviço (QoS) e seus requisitos, em termos de atraso, confiabilidade e tolerância a falhas durante o roteamento.

RSSFs, não é possível atribuir identificadores globais para cada nó, devido ao grande número de nós implantados. Essa ausência de identificação global, juntamente com a implantação aleatória de nós sensores, torna difícil a escolha de um próximo salto.

Recentemente, algoritmos que exploram informações geográficas, chamados Algoritmos de Roteamento Geográfico tem sido propostos para facilitar o transporte de dados nas RSSFs. Estes algoritmos se baseiam na posição dos nós da rede ao invés de seus endereços para criar rotas em direção ao destino, ou um *sink* (Rao et al., 2003). Cada nó só precisa conhecer a posição dos seus vizinhos, normalmente a um salto de distância, para a criação das rotas. Uma forma simplificada de utilização dos algoritmos de roteamento geográfico é utilizar um algoritmo de roteamento geográfico guloso

(*Greedy Geographic Routing*). A técnica realiza o roteamento dos pacotes através do Encaminhamento do pacote ao vizinho mais próximo do nó sorvedouro (Xing et al., 2004).

Em muitos protocolos de roteamento, a informação de localização é necessária para calcular a distância entre dois nós específicos, chamados protocolos de roteamento baseados em localização. Nestes protocolos, os nós são endereçados através de suas informações de posicionamento e deste modo, sistemas de localização são usados para fornecer essas informações. Tais informações também são utilizadas em outras tarefas como a identificação e correlação dos dados sensoreados, endereçamento dos nós, gerenciamento e localização das consultas realizadas pelos nós de acordo com a região sensoreada, avaliação da densidade e cobertura de nós, construção do mapa de energia, rastreamento de objetos e roteamento geográfico. A distância entre o nó e seus vizinhos pode ser estimada com base na informação do RSSI recebido e desta forma, coordenadas relativas podem ser obtidas através da troca de informações entre os vizinhos (Savvides et al., 2001), (Al-karaki & Kamal, 2004).

2.5 Protocolos de Roteamento Geográfico

Os protocolos de roteamento geográficos têm sido vistos como uma das principais soluções de roteamento em Redes de Sensores Sem Fio (RSSFs) por serem escaláveis, dinâmicos e possuírem uma alta taxa de entrega de dados. Estes algoritmos tiram vantagem da informação de localização dos nós para implementar técnicas de encaminhamento mais eficiente. Nesta abordagem, os vizinhos trocam informações sobre sua localização e desta forma, quando um nó precisa encaminhar um pacote, ele envia para o vizinho que se supõe ser mais próximo do destino final, ou um *sink*, através do paradigma guloso, também chamado de roteamento guloso (*greedy forwarding*) (Tan & Kermarrec, 2011). Desta forma, o nó de origem insere as coordenadas de destino nos pacotes a serem enviados. A informação de localização utilizada nos algoritmos geográficos pode ser derivada de dispositivos como um GPS ou ainda modelada a partir de coordenadas virtuais (Villalba et al., 2009).

O roteamento geográfico possui muitas características que vão ao encontro dos requisitos singulares das RSSFs, como por exemplo, não possuir monitoramento de estado, uma vez que os nós só precisam armazenar informações sobre seu posicionamento atual de um número limitado de destinos ou *sinks*. Esta é uma característica importante, pois nós sensores possuem limitação de memória. Além disso, sendo baseado em informações locais, o roteamento geográfico é escalável e portanto, aplicável a RSSFs

densas (Nati, 2008). Um dos trabalhos mais relevantes da literatura sobre protocolos de roteamento geográfico é o GPSR (Karp & Kung, 2000), que utiliza a posição dos nós e o destino do pacote para tomar decisões de encaminhamento guloso, através das informações sobre os vizinhos a um salto de distância.

Para Karl e AWillig (Karl & Willig, 2005), a idéia por trás de grande parte dos protocolos de roteamento geográfico possui duas vertentes:

1. Para a maioria das aplicações é necessário endereçar localizações físicas, como por exemplo, fazer referência a “qualquer nó em uma dada região” ou ainda “um nó em/próximo a um determinado ponto”. Quando tais requisitos existem, estes devem ser atendidos por um sistema de encaminhamento de pacotes adequado.
2. Quando a posição do destino final, da origem da mensagem e dos nós intermediários são conhecidas, esse conhecimento pode ser usado para auxiliar no processo de encaminhamento. Para isto, o destino deve ser especificado tanto geograficamente quanto através de um sistema de localização. Sua posição atual deve ser considerada caso necessário.

O primeiro aspecto abordado é conhecido como *geocasting*. Nesta abordagem de roteamento os dados são enviados a nós arbitrários em uma região especificada. O *geocast* é considerado como uma subclasse do protocolo *multicast* e pode ser implementado como o *multicast* convencional através a simples definição de um grupo de nós em uma determinada região geográfica, chamado *multicast group* (Maihofer, 2004). O segundo aspecto é conhecido como roteamento baseado em posição, que normalmente é combinado a um serviço de localização. A ajuda de um serviço de localização é necessária para o conhecimento da posição atual de um determinado nó, pois suas posições são registradas através do serviço. Quando um nó não conhece a posição de um vizinho, o serviço de localização é acionado com a requisição da informação (Mauve et al., 2001).

A capacidade de auto-descrição referente ao posicionamento dos vizinhos a um salto de distância é a chave da utilização da informação geográfica em protocolos de roteamento. Assim, quando falamos de roteamento geográfico, fazemos referência à forma de encaminhamento em que o nó de origem direciona o envio do pacote. Um nó toma uma decisão de transmissão de pacotes baseado na posição dos seus. Neste contexto, o nó de origem compara a localização do destino com as coordenadas que já possui dos vizinhos. Só então, a mensagem é encaminhada ao vizinho que está geograficamente mais próximo do destino final. Esta técnica é chamada de encaminhamento guloso. O procedimento é repetido até os pacotes alcançarem o destino pretendido, na

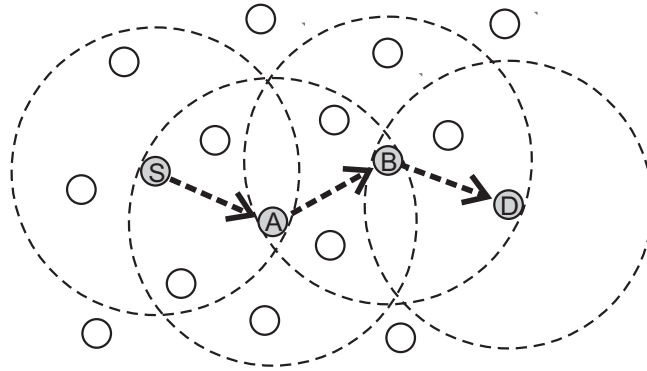


Figura 2.2. Encaminhamento Guloso. Nó S seleciona o nó A , que está mais próximo do destino D . O procedimento é repetido até a entrega do pacote ao destino D , normalmente um *Sink*. Baseado em (Park et al., 2010).

maioria das vezes um *sink*. A estratégia é baseada no progresso positivo dos pacotes, ou seja, o repasse em múltiplos saltos em direção ao *sink*.

A Figura 2.2, mostra o funcionamento da estratégia de encaminhamento geográfico guloso. O nó fonte S seleciona o nó A que está geograficamente mais próximo do destino D . O mesmo ocorre em relação ao nó A , que seleciona o nó B , geograficamente mais próximo do destino D . Finalmente, o nó B encaminha o pacote replicado ao destino D .

O roteamento geográfico se baseia no conhecimento da localização física dos nós sensores, que quando equipado com dispositivos GPS deixam de ser economicamente viáveis. A alternativa é a utilização algoritmos de localização para o cálculo das coordenadas físicas.

Soluções de roteamento baseadas em informações geográficas têm se mostrado atraentes no que se refere à escalabilidade e eficiência energética. Contudo, em algumas situações os métodos envolvendo informações geográficas não podem ser utilizados, especialmente em cenários onde a informação de localização não está disponível ou não existe. Nestes casos, métodos alternativos de roteamento que tirem proveito da escolha do próximo em direção ao destino final são necessárias. Desta forma, um dos pontos fracos do roteamento geográfico é a necessidade de localização dos nós, que pode ser custoso e susceptível a grandes erros em RSSFs (Souza et al., 2010).

2.6 Classificação dos Algoritmos de Roteamento

Buscando facilitar a entendimento sobre o conhecimento que os nós sensores possuem sobre suas informações de posicionamento, classificamos os algoritmos estudados em

três categorias distintas: (1) baseados em coordenadas geográficas; (2) baseados em coordenadas virtuais; e (3) livres de posicionamento.

2.6.1 Algoritmos Baseados em Coordenadas Geográficas

Nesta categoria, os protocolos requerem uma estimativa da localização global dos nós (latitude/longitude). Tal estimativa deve ser obtida (1) equipando-se todos os nós com receptores GPS (*Global Positioning System*); (2) ou através da execução de um algoritmo de localização; (3) ou mesmo através do posicionamento manual dos nós (Boukerche et al., 2007). Nesta seção apresentamos alguns exemplos de algoritmos que utilizam coordenadas geográficas.

2.6.1.1 Greedy Perimeter Stateless Routing for Wireless Networks (GPSR)

O *Greedy Perimeter Stateless Routing for Wireless Networks (GPSR)* (Karp & Kung, 2000), (Karp, 2000), utiliza a posição dos nós e o destino do pacote para tomar decisões de encaminhamento guloso, baseado nas informações dos vizinhos a um salto de distância. O algoritmo consiste de dois métodos de encaminhamento de pacotes: encaminhamento guloso, usado quando aplicável, e encaminhamento por perímetro, utilizado em regiões onde o primeiro falha. Nesta abordagem, quando um pacote atinge uma região onde o encaminhamento guloso não é possível, o algoritmo passa a operar em modo de perímetro. Uma vez que o pacote encontra uma borda, ou um caminho ótimo, o pacote é roteado ao longo da superfície do perímetro, alternando entre as bordas no sentido anti-horário, estratégia conhecida como regra da mão direita. Na Figura 2.3, um exemplo do funcionamento do algoritmo é mostrado. Neste exemplo, x está mais próximo a D do que seus vizinhos w e y . O arco pontilhado, que é a faixa de comunicação de D , possui raio igual a distância entre x e D . Embora as opções de caminhos ($x \rightarrow y \rightarrow z \rightarrow D$) e ($x \rightarrow w \rightarrow v \rightarrow D$), existam para D , x não irá optar pelo encaminhamento a w ou y utilizando encaminhamento guloso, pois ambos estão fora da regra de encaminhamento. Neste caso, o encaminhamento por perímetro com a aplicação da regra da mão direita, será necessário.

2.6.1.2 Bounded Voronoi Greedy Forwarding (BVGF)

Similar à abordagem mais conhecida de Encaminhamento Guloso, o *Bounded Voronoi Greedy Forwarding (BVGF)* (Xing et al., 2004), é um algoritmo de localização que realiza decisões de Encaminhamento Guloso baseadas na localização dos vizinhos a um salto de distância. Uma abordagem similar chamada DV-hop baseada no diagrama de

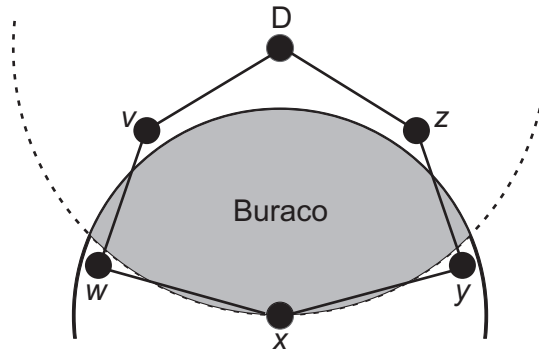


Figura 2.3. Exemplo de buraco de cobertura entre o nó x e o destino D . Baseado em (Karp, 2000).

Voronoi foi proposta por (Boukerche et al., 2009). Um exemplo de rota do algoritmo BVGF é apresentada na Figura 2.4. Neste algoritmo uma rede é modelada através de um diagrama de Voronoi onde as regiões representam as localizações dos sensores. Cada nó conhece sua posição geográfica e mantém uma tabela com os endereços dos nós vizinhos. Para manter esta tabela, cada nós divulga periodicamente, através de *broadcast*, uma mensagem que inclui sua localização, bem como a localizações dos vértices das regiões de Voronoi. A diferença fundamenta entre o Encaminhamento Guloso tradicional e o BVGF é que este sempre encaminha um pacote ao nó onde a região de Voronoi é intersectada pela reta que une a origem do destino. O algoritmo escolhe como próximo salto o vizinho que tem a menor distância euclidiana do destino entre os nós elegíveis para o Encaminhamento. Os nós elegíveis sempre irão atuar no próximo salto. Quando existem muitos vizinhos que também são elegíveis e estão próximos ao destino, o nó que irá criar a rota escolhe randomicamente o vizinho do próximo salto. Apesar do exposto, o algoritmo BVGF não considera energia como métrica (Xing et al., 2004), (Xing, 2006).

2.6.1.3 Geographic Random Forwarding (GeRaF)

No *Geographic Random Forwarding (GeRaF)* (Zorzi & Rao, 2003), quando um nó possui um pacote para ser passado adiante, este realiza o Encaminhamento através de mensagens *broadcast* que contém sua informação de localização ao destino pretendido. Neste algoritmo, o sensor não precisa salvar a informação de localização sobre seus vizinhos. Como apresentado na Figura 2.5, o GeRaF adota o Encaminhamento através do menor esforço, em que o nó atual que irá atuar como nó de retransmissão, ou *relay*, não é conhecido pelo nó que irá enviar o pacote. O nó que vai transmitir o pacote, realiza um *broadcast* e todos os nós ativos, dentro de sua área de cobertura, irão escutar

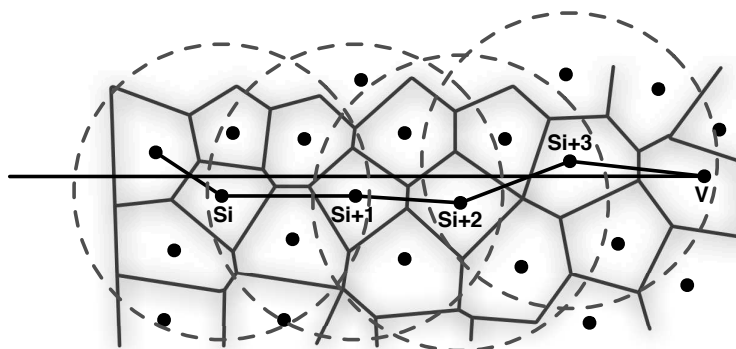


Figura 2.4. Exemplo de rota no algoritmo BVGF. Baseado em (Xing et al., 2004).

a mensagem. Cada nó irá determinar sua distância do destino final e avaliar se pode atuar como *relay*. Nós que se encontram na região de não-retransmissão nunca serão irão atuar como *relays*. Não há garantias de que o nó que envia o pacote sempre estará habilitado para encaminhar a mensagem até o destino. A abordagem integra algoritmo geográfico e programação a atividade e inatividade dos nós através de um algoritmo *awake-sleep*. Quando um nó realiza um sensoriamento de dados para enviá-lo ao *sink*, ele primeiro checa se o canal de transmissão está livre para evitar colisões. Caso o canal tiver que permanecer inativo por um período de tempo, o sensor transmite a mensagem *Request To Send (RTS)* original para todos os vizinhos ativos. Essa mensagem inclui a localização do sensor de origem e do *sink*. Quando sensores ativos recebem a mensagem RTS, eles avaliam suas prioridades de acordo com suas informações locais e as do *sink*. No GeRaF não há tabela de manutenção de rotas e as coordenadas entre os nós da rede não são necessárias (?).

2.6.1.4 Location based Energy-Aware Reliable routing protocol (LEAR)

Proposto para as RSSFs baseadas no posicionamento dos sensores e em clusterização, ou agrupamento, *Location based Energy-Aware Reliable routing protocol (LEAR)* (Alasem et al., 2011), ou protocolo de roteamento confiável e ciente de energia baseado em localização, aproveita a vantagem sobre o conhecimento da informação de posicionamento dos nós para criar rotas eficientes na rede. A informação de localização pode ser extraída através de dispositivos como um GPS. Cada nó na área geográfica LEAR constrói sua tabela de roteamento baseada nas distâncias para os seus vizinhos. Como na maioria dos algoritmos que utilizam o Encaminhamento Guloso, um nó decide sobre um caminho de transmissão. O principal componente no LEAR, é o

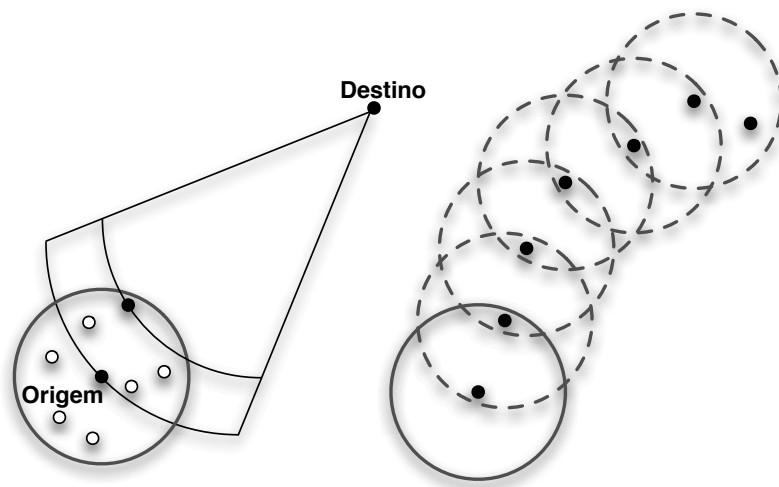


Figura 2.5. No GeRaF, os nós enviam pacotes usando mensagens *broadcast*. Todos os nós dentro da área de cobertura irão receber o pacote e avaliar suas prioridades na tentativa de atuar como *relay*. Baseado em (Zorzi & Rao, 2004).

algoritmo chamado *Enhanced Greedy Forwarding (EGF)*, ou Encaminhamento Guloso avançado, mostrado na Figura 2.6, considerado como cerne do protocolo. O algoritmo seleciona o nó mais próximo do nó ativo baseado em sua distância. Esta escolha da menor distância geográfica para o próximo salto tenta minimizar o consumo de energia por mensagem transmitida. O Algoritmo EGF seleciona somente os nós que estão no caminho do nó de destino, podando os nós que foram deixados para trás ou não eleitos como os participantes da rota. O LEAR é baseado no posicionamento dos nós e na clusterização. Cada cluster é ocupado aleatoriamente com nós sensores. A clusterização tem sido utilizada em muitas abordagens para produzir algoritmos eficientes e econômicos de energia para redes de sensores.

2.6.1.5 Energy Aware Greedy Routing (EAGR)

O *Energy Aware Greedy Routing (EAGR)* (Haider et al., 2007) é um algoritmo de localização para redes de sensores que tem como objetivo principal estender o tempo de vida da rede, recebendo a maior taxa de entrega de dados para equilibrar o consumo de energia dos nós. Nesta abordagem, cada nó faz uma decisão local para escolher seu próximo salto. Além disso, cada nó conhece sua localização geográfica, o seu nível de energia e as localizações e níveis de energia dos vizinhos. A técnica utiliza a informação baseada na localização geográfica dos nós e seus níveis de energia para selecionar o vizinho que irá encaminhar o pacote em direção ao destino. No EAGR é assumido que os nós sensores são estáticos e existe um banco de dados de localização, o que permite a

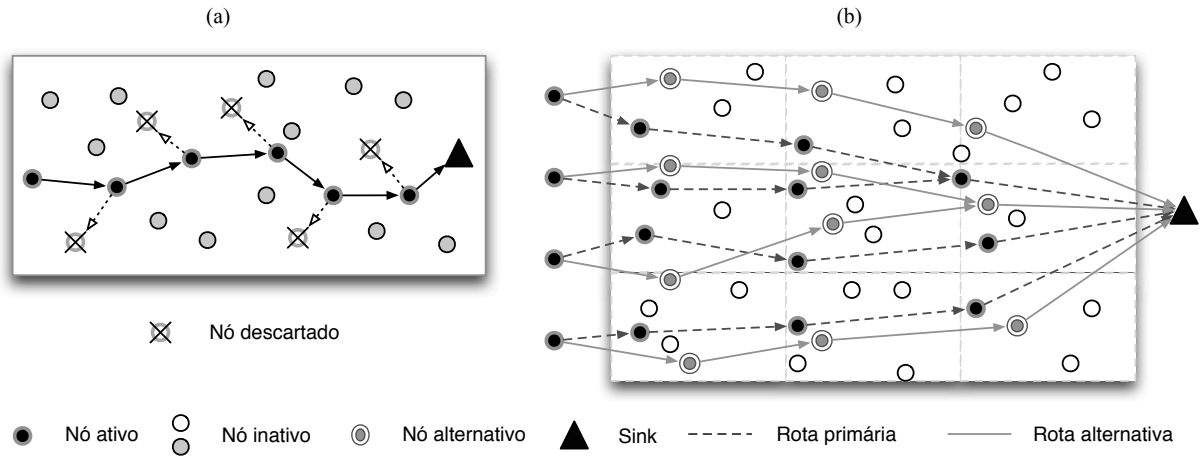


Figura 2.6. (a) Encaminhamento Guloso avançado. (b) LEAR em clusters. Baseado em (Alasem et al., 2011).

cada um dos nós descobrir sua posição, ou seja, cada nó pode aprender sua localização através de um sistema de localização, como um GPS. Outro importante pressuposto é que a topologia irregular pode ser considerada para redes de sensores. Em comparação com a abordagem de Encaminhamento Guloso simples, que não considera a energia como métrica, o algoritmo provê maior tempo de vida da rede.

2.6.1.6 Greedy with Path Optimization Routing (GPOR)

Greedy with Path Optimization Routing (GPOR) (Dinh & Nguyen, 2010) é um algoritmo de roteamento geográfico híbrido, que utiliza a técnica de encaminhamento guiado, baseada em um cache de roteamento aplicada antes do encaminhamento guloso, para tomar decisões de roteamento. No *GPOR*, independente da topologia do roteamento, cada tomada de decisão por encaminhamento não é baseada em um destino em particular e sim em um grupo de destinos que estão geograficamente próximos a uma posição definida. Esta característica é chamada área de aplicação. Desta forma, o encaminhamento guiado ocorre quando um nó procura em seu cache de roteamento por entradas que são aplicáveis ao destino escolhido. Então, este nó seleciona uma entrada identificada como área aplicável (nó próximo a um destino escolhido), e encaminha o pacote para uma região próxima da entrada selecionada. O algoritmo realiza ainda otimização de rotas para melhorar a eficiência durante o processo de encaminhamento, onde cada nó grava pelo menos uma entrada relacionada a área aplicável (posição para o envio do pacote), em seu cache de roteamento.

2.6.2 Algoritmos Baseados em Coordenadas Virtuais

Esses algoritmos tentam criar um outro tipo de sistema de coordenadas, que não está relacionado ao sistema de posicionamento global. Nesta seção apresentamos alguns algoritmos que utilizam sistemas de coordenadas virtuais.

2.6.2.1 Greedy Minimum energy consumption Forwarding Protocol (GMFP)

O *Greedy Minimum energy consumption Forwarding Protocol* (GMFP) (Panigrahi et al., 2009) visa aumentar o tempo de vida da rede através da escolha ótima para um nó elegível em cada salto de Encaminhamento. Nesta abordagem, o Encaminhamento geográfico Guloso é combinado com a informação do consumo de energia dos nós, considerando conexões de baixa qualidade nas decisões de Encaminhamento salto-a-salto. A seleção do nó elegível é feita como base no menos custo em relação ao consumo de energia e transmissão de pacotes. A implementação do protocolo obedece alguns princípios e entre eles destacamos a distribuição aleatória dos nós, conhecimento de sua informação de localização, canais de comunicação não correlacionados entre os nós vizinhos, pacotes perdidos em decorrência de falhas em qualquer estágio do roteamento e o Encaminhamento de pacotes ao próximo salto baseado somente na informação e processamento local. No GMFP os nós possuem a informação de posicionamento da vizinhança local baseada avisos periódicos. É assumido que todos os nós possuem algum tipo de conhecimento de sua localização, que pode ser geográfica ou virtual.

2.6.2.2 Virtual Coordinate assignment protocol (VCap)

O *Virtual Coordinate assignment protocol* (VCap) (Caruso et al., 2005), (Tsai et al., 2009) define um sistema de coordenadas para e distâncias em saltos. O protocolo assume que cada nó é estático e tem o mesmo alcance de transmissão. Além disso, cada nó tem um identificador único (ID) utilizado para evitar laços, que podem ser facilmente removidos, permitindo a cada nó escolher uma faixa de transmissão. A implementação do VCap está dividida em quatro fases, baseadas em mensagens broadcast, com o objetivo de identificar três nós âncora (X , Y e Z), dinamicamente definidos durante a inicialização. O algoritmo atribui a cada nó um trio de coordenadas dadas pela distância em saltos dos nós âncoras. Quando a última fase é completada, cada nó irá utilizar o trio de coordenadas atribuídas. No VCap, o sistema de coordenadas utiliza um protocolo de roteamento geográfico para entregar os pacotes às zonas de destino. A sobrecarga de armazenamento para cada nó é limitada pelo armazenamento

do trio de coordenadas atribuído. Na presença de buracos na rede, o pacote que está sendo enviado irá seguir uma regra chamada *local detour*, implementada para melhorar a técnica de Encaminhamento Guloso nesta abordagem, permitindo que aja uma mudança na rota, de forma que o progresso do pacote de revertido e o algoritmo encaminhe a mensagem de volta, ou para trás, caso necessário.

2.6.2.3 Greedy Forwarding with Virtual Position (GF-ViP)

O *Greedy Forwarding with Virtual Position (GF-ViP)* (You et al., 2009) foi desenvolvido para melhorar a taxa de Encaminhamento Guloso em RSSFs esparsas, com pequenos buracos encontrados ao longo do Encaminhamento. O algoritmo foi proposto com dois complementos, o *Virtual Multi-Level Position (MVP)* e o *Greedy Forwarding with Virtual Hierarchical Position (HVP)*. Cada algoritmo possui duas variações que utilizam algoritmos Gulosos baseado em progresso *Most Forwarding Within Radius (MFR)* (Takagi & Kleinrock, 1984). A principal vantagem nestas abordagens é a aplicação do Encaminhamento Guloso durante todo o processo de roteamento, gerando resultados altamente eficientes na geração de rotas. O método explora o posicionamento virtual através da criação de um ponto médio em relação a todos os vizinhos diretos. Desta forma, a informação sobre o posicionamento é armazenada nos próprios nós e em seus vizinhos diretos. Cada nó possui informações sobre sua posição virtual e a de seus vizinhos. O processo de desvio de buraco (*bypass hole*) foi ilustrado na Figura 2.7. Ao invés de comparar as localizações geográficas atuais de cada nó, o algoritmo utiliza posicionamento virtual para selecionar o próximo salto. Cada só calcula sua posição e transmite ao seu vizinho virtual direto. Desta forma o nó virtual conhece sua posição e a posição dos vizinhos virtuais. Esse posicionamento virtual de um nó fornece uma indicação de como os vizinhos diretos são localizados ao redor do nó em média, utilizado para mostrar a tendência de Encaminhamento geográfico durante um processo de Encaminhamento. Os nós sensores precisam interagir com os seus vizinhos diretos para obterem as informações do posicionamento virtual.

2.6.2.4 Greedy Embedding Spring Coordinates (GSpring)

Desenvolvido para gerar coordenadas virtuais úteis, o *Greedy Embedding Spring Coordinates (GSpring)* (Leong et al., 2007), tem sido utilizado para melhorar o desempenho na geração de rotas de algoritmos espaciais, de forma que seja possível detectar situações de beco sem saída (*dead ends*) durante o processo de Encaminhamento Guloso. O algoritmo pode ajustar suas coordenadas a fim de aumentar a convexidade de espaços vazios na topologia da rede. Isto reduz a probabilidade de pacotes estejam sendo

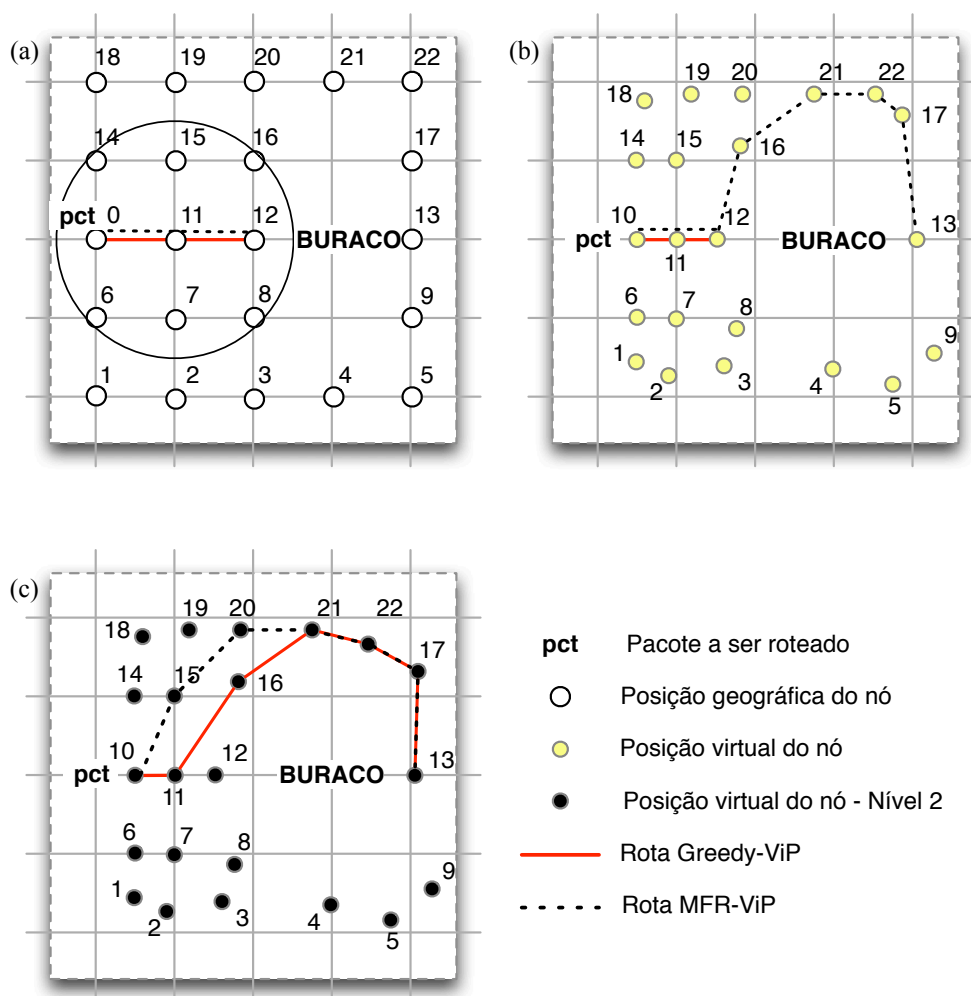


Figura 2.7. Desvio de buraco usando posicionamento virtual. (a) Nó 12 simulando um buraco tanto para Greedy quanto para MFR. (b) Caminho que pode ser encontrado através do algoritmo MFR-ViP. (c) Caminhos em volta do buraco. Baseado em (You et al., 2009).

enviados para bcos sem saída durante o Encaminhamento, melhorando o desempenho do roteamento. O GSpring usa um algoritmo de detecção de perímetro para identificar nós no perímetro da rede e usa essa informação para atribuir coordenadas iniciais. O algoritmo calcula, para cada nó, uma área de propriedade que pode conter os nós conflitantes que podem atrair tráfego para espaços vazios ou buracos, quando os nós são eleitos como destino. Um nó pode então empurrar o nó conflitante para fora da região de propriedade. Para uma rede em U , por exemplo, pacotes são encaminhados através do método Guloso entre nós em cada ponta final da estrutura em U (Kermarrec & Tan, 2010). A abordagem foi projetada para redes que utilizam nós não-fixos. O GSpring funciona em duas etapas. Na primeira, cada nó atribui sua própria coordenada ini-

cial. Na segunda etapa, o Encaminhamento Guloso é implementado para melhorar o desempenho da rede, depois de centenas de milhares de iterações e ajustes para as coordenadas, de acordo com a topologia da rede.

2.6.2.5 Geo-Logical Routing in Wireless Sensor Networks (GLR)

O *Geo-Logical Routing in Wireless Sensor Networks (GLR)* (Dhanapala & Jayasumana, 2011), utiliza informações de posicionamento derivadas a partir coordenadas virtuais (CV), como alternativa para informações de localização. A técnica traz as vantagens do roteamento geográfico sem a necessidade de posicionamento físico de cada nó. O algoritmo alterna entre três modos de operação durante o processo de encaminhamento: (1) CT, que utiliza coordenadas baseadas na topologia da rede (2) CV, que utiliza coordenadas virtuais e uma função de distância, e finalmente (3) MA, ou Modo âncora, que encaminha o pacote até o âncora que está mais próximo do destino. O posicionamento dos nós baseado em informações físicas pode ser custoso em relação à localização do nó e/ou utilização de GPS. Desta forma, nesta abordagem um nó é caracterizado como um vetor de CV e consiste em um número mínimo saltos de um conjunto de nós âncora. CVs contém informações derivadas da conectividade da rede, mas não possuem informações do domínio físico, como buracos.

2.6.3 Algoritmos Livres de Posicionamento

Nesta abordagem, os nós não precisam saber informações sobre seu posicionamento nem o posicionamento de qualquer outro nó vizinho da rede.

2.6.3.1 Greedy Forward with Received Signal Strength Indicator (GF-RSSI)

Desenvolvido para superar uma limitação em algoritmos de Encaminhamento Guloso convencionais, o *Greedy Forward with Received Signal Strength Indicator (GF-RSSI)* (Pham et al., 2006) usa a força do sinal como filtro para gerar caminhos ótimos na rede. A técnica de Encaminhamento Guloso sempre escolhe o próximo salto baseado no menor caminho até o destino. Em redes densas ou em ambientes internos, os algoritmos de Encaminhamento Guloso não produzem o efeito esperado, uma vez que a comunicação entre os nós torna-se muitas vezes instável devido a interferências de comunicação entre os vizinhos. O GF-RSSI, usa a força do sinal para estimar a qualidade do link de comunicação. Desta forma, se o nó que está enviando o pacote encontrar um vizinho próximo ao destino, e o sinal de origem deste vizinho está acima de um

certo limitar, o dado é redirecionado a um outro nó vizinho. Uma visão geral sobre o funcionamento do algoritmo foi apresentada na Figura 2.2.

2.6.3.2 Received Signal Strength Routing (RSSR)

A idéia chave do algoritmo *Received Signal Strength Routing (RSSR)* (Boukerche et al., 2008), é aproveitar a vantagem da grande capacidade do *sink* e equipar este nó com um dispositivo de comunicação potente, de forma que o pacote seja enviado a todos os nós da rede em um único salto. O nós regulares da rede pode replicar mensagem do *sink* usado comunicação multi-sato. A mesma mensagem, atinge dos os nós com diferentes indicações de força de sinal (RSSI) de modo os nós mais distantes tenham indicação da força do sinal baixo durante a propagação. O princípio é redirecionar o pacote ao vizinho que recebe a mensagem com o sinal mais forte, na teoria, o vizinho mais próximo do *sink*. Baseado nesta informação, duas variações do algoritmo são propostas: *RSSR-Election* e *RSSR-Selection*. Na primeira abordagem, um algoritmo de eleição de líder, que não requer troca de pacotes foi proposto para a escolha do próximo salto. O objetivo é que os nós próximos do *sink*, que receberam RSSI alto, irão esperar por um curto período de tempo antes de serem eleitos como líder, e o nós mais distantes do *sink* irão eleger a si como o próximo salto após um período tempo mais longo. Quando um nó elege a si como o próximo salto, ele imediatamente encaminha o pacote replicado ao *sink*. Na segunda abordagem, nós vizinhos trocam informações e o vizinho com RSSI alto é selecionado em cada etapa. Quando o *sink* envia a primeira mensagem, todos os nós trocam informações com seus vizinhos, estimando suas distâncias de transmissão através de anúncios. Um nó recebe um anúncio de cada vizinho e grava essa informação em uma tabela de roteamento, Assim, o nó checa se a mensagem é para ele e, se for o caso, espera um período curso de tempo, envia uma réplica para o vizinho próximo do *sink* e salva o procedimento em uma tabela de Encaminhamento de pacotes. O algoritmo permite a execução da estratégia de Encaminhamento e não requer nenhuma informação de localização ou coordenadas virtuais.

2.6.3.3 Síntese

Os algoritmos discutidos neste trabalho foram categorizados de acordo com nossa proposta de classificação. Também adicionamos algumas características compartilhadas por alguns dos algoritmos estudados como ciência de energia, desvio de buracos e transmissão de informações sobre o nó. A classificação é mostrada na Tabela 2.2.

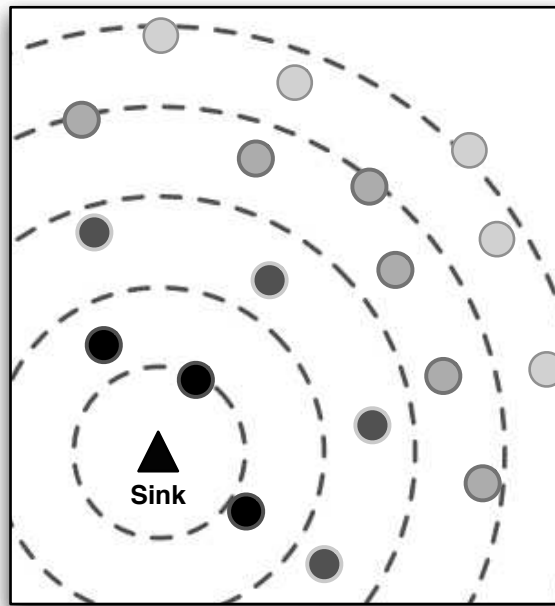


Figura 2.8. Propagação do sinal RSSI. Nós mais distantes recebem consulta do *sink* com sinal baixo. Baseado (Boukerche et al., 2008).

2.7 Considerações Finais

Neste capítulo, foram apresentados conceitos fundamentais sobre RSSFs que serviram como base para o desenvolvimento de nossa proposta. Entre os aspectos abordados destaque:

- A energia é apontada como o recurso mais escasso e que demanda a maior parte das atenções em RSSFs. É também considerada como fator essencial para o tempo de vida na grande maioria das soluções atuais.
- A importância da fusão de dados em RSSFs, como forma de reduzir a quantidade de transmissão de dados, reduzir a colisão de dados durante a transmissão, reduzir o congestionamento da rede, economizar energia de um nó e além disso, prolongar o tempo de vida da rede.
- A agregação de dados como forma especial de fusão de dados, utilizada para suprimir a redundância de informações e reduzir o consumo de energia. Além disso, é considerada essencial para as estratégias de formação de *clusters*, combinação e/ou sumarização de informações de alguns ou todos os nós sensores.
- Novas formas de utilização do protocolo *flooding* têm sido implementadas como forma de minimizar o número de transmissões cegas (*blind transmission*) de da-

	A	B	C	D	E	F
BVGF (Xing et al., 2004), (Xing, 2006)		✓				✓
EAGR (Haider et al., 2007)		✓		✓		
LEAR(Alasem et al., 2011)		✓		✓		
GeRaF (Zorzi & Rao, 2003)		✓				✓
GF-ViP (You et al., 2009)	✓				✓	✓
GF-RSSI (Pham et al., 2006)			✓			
GLR (Dhanapala & Jayasumana, 2011)	✓					
GMFP (Panigrahi et al., 2009)	✓	✓		✓		
GPOR (Dinh & Nguyen, 2010)	✓					
GPSR (Karp & Kung, 2000)		✓			✓	
GSpring (Kermarrec & Tan, 2010)	✓				✓	
RSSR (Boukerche et al., 2008)			✓			
VCap (Caruso et al., 2005)	✓				✓	✓

A) Algoritmos que utilizam coordenadas virtuais.

B) Algoritmos que utilizam algum sistema de localização ou GPS.

C) Algoritmos livres de posição.

D) Algoritmos cientes de energia.

E) Algoritmos que tratam o desvio de buracos na rede.

F) Algoritmos que informam suas posições via mensagens *broadcast*.

Tabela 2.2. Classificação dos algoritmos estudados

dos e o número de nós envolvidos no processo de encaminhamento das mensagens. Entre as novas abordagens, destacamos a estratégia de flooding controlado, baseada no número de saltos até o *sink* e a utilização de chaves, definidas a partir de uma estação base, configuradas em um arquivo de mapeamento de chaves. Apesar da alta taxa de entrega alcançada pelo protocolo *flooding*, o alto consumo de energia necessário para sua execução pode diminuir o tempo de vida da rede, dificultando sua utilização em redes densas.

- O RSSI pode ser usado para estimar a distância entre dois nós baseado na força do sinal recebido pelo outro nó. Um nó que está transmitindo uma informação envia um sinal com uma certa força que irá decrescer conforme a propagação. Quanto maior a distância do nó que recebe a informação, menor será a força do sinal. Esta estratégia tem sido amplamente utilizada como métrica para aferição da potência de sinal, recebida pelos nós.
- Os protocolos de roteamento geográficos têm sido vistos como uma das principais soluções de roteamento em Redes de Sensores Sem Fio (RSSFs) por serem escalá-

veis, dinâmicos e possuem uma alta taxa de entrega de dados. Estes algoritmos tiram vantagem da informação de localização dos nós para implementar técnicas de encaminhamento mais eficiente. As informações de localização utilizadas nos algoritmos geográficos podem ser derivadas de dispositivos como um GPS ou ainda modeladas a partir de coordenadas virtuais.

- Neste trabalho, os algoritmos estudados foram classificados em três categorias distintas: (1) baseados em coordenadas geográficas; (2) baseados em coordenadas virtuais; e (3) livres de posicionamento. Esta classificação, tem o objetivo de facilitar o entendimento sobre a forma de anúncio dos nós, informações de localização e encaminhamento dos pacotes até o *sink*.

No capítulo 3, apresentamos um novo algoritmo de roteamento geográfico que não requer informações de posicionamento dos nós. Nesta nova abordagem, detalhada nas seções 3.1.1 e 3.1.2, aproveitamos algumas vantagens de um algoritmo de roteamento geográfico tais como robustez, escalabilidade, dinamicidade e entrega confiável dos dados, mas sem precisar das informações de posicionamento dos nós ao mesmo tempo que provê o recurso de agregação de dados na rede. Com base apenas no RSSI *sink*, enviado em uma única consulta que alcança todos os nós da rede, um novo e inovador esquema de roteamento e agregação de dados foi proposto. Nossa técnica de agregação de dados utiliza um temporizador para esperar pelos pacotes a serem agregados que também é feito com base na mesma informação de RSSI.

Capítulo 3

Roteamento e Agregação de Dados baseado no RSSI

Neste capítulo, propomos um novo algoritmo de roteamento geográfico com agregação de dados para redes de sensores sem fio que não requer informações de posicionamento dos nós: o *GRDA* (*Greedy Routing and Data Aggregation*). A proposta do algoritmo é possuir todas as vantagens de um algoritmo de roteamento geográfico tais como robustez, escalabilidade, dinamicidade e entrega confiável dos dados, mas sem precisar das informações de posicionamento dos nós ao mesmo tempo que provê o recurso de agregação de dados na rede.

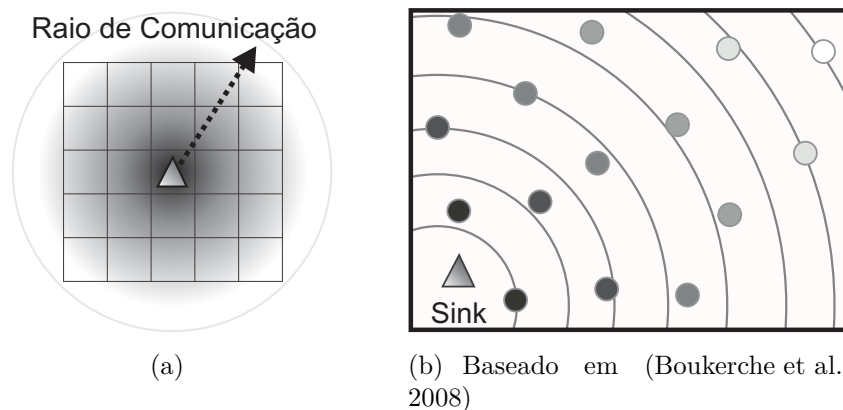


Figura 3.1. (a) Potência do sinal decrescendo à medida que a mensagem de consulta do *sink* se propaga na rede. (b) Nós sensores localizados mais distantes do *sink* receberão a mensagem com RSSI menor que os nós mais próximos

Assim como no RSSR (Boukerche et al., 2008), nosso algoritmo tira proveito da possibilidade de se equipar o nó *sink* com um dispositivo de comunicação de maior potência, de forma que, em um único salto, o nó *sink* pode enviar uma consulta para toda

a rede. Então, os nós regulares pode responder à consulta do *sink* em múltiplos saltos, através da abordagem de encaminhamento guloso proposta neste trabalho. A ideia desta arquitetura de comunicação surgiu a partir do projeto ATTO (*Amazonian Tall Tower Observatory*), que já está implementado na Amazônia e consiste em uma torre no meio da floresta com uma estrutura física de 320 m de altura e alta capacidade de comunicação. O foco do projeto é mapear eventos na região e fornecer medidas confiáveis de fontes e sumidouros de gases de efeito estufa como CO_2 , CH_4 e N_2O (Tollefson, 2010).

Em nossa proposta, a mesma mensagem de consulta do nó *sink* alcançará os nós da rede com diferentes potências (RSSIs), sendo que os nós mais distantes possuirão um sinal mais baixo, ao contrário dos nós mais próximos, que irão receber a mensagem com um sinal mais forte, conforme ilustrado nas Figuras 3.1(a) e 3.1(b).

Com o objetivo de visualizar e comparar as rotas geradas pelos algoritmos, a

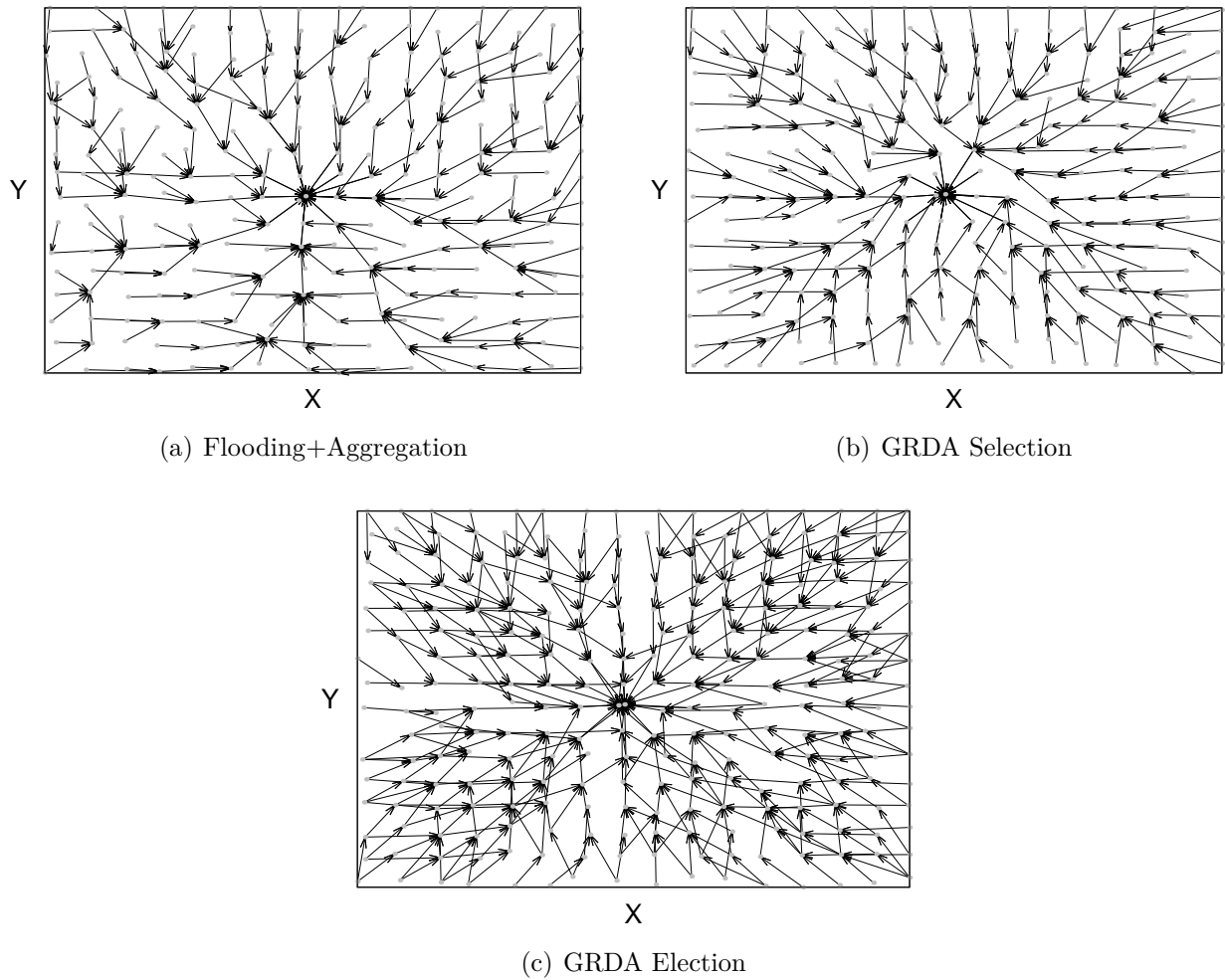


Figura 3.2. Gráfico de Fluxo dos Algoritmos GRDA

Figura 3.2(a) apresenta um gráfico de rotas gerado pela árvore inversa, gerada pelo algoritmo *Flooding+Aggregation*, enquanto as Figuras 3.2(b) e 3.2(c), apresentamos os gráficos de rota gerados pelos algoritmos *GRDA Selection* e *GRDA Election*, usando somente as informações do RSSI.

O restante deste capítulo é organizado como segue. Na próxima seção, descrevemos as definições utilizadas ao longo deste trabalho e nossa definição do problema. Apresentaremos as duas variações do algoritmo GRDA: o *GRDA Selection*, que será mostrado na seção 3.1.1 e o *GRDA Election*, detalhado na seção 3.1.2.

3.1 Definição do Problema

Nesta seção, serão apresentadas algumas definições para um melhor entendimento deste trabalho.

Definição 1 - Escopo do problema: Considere uma Rede de Sensores sem Fio (RSSF) como sendo um grafo euclidiano $G = (V, E, r)$, com as seguintes propriedades:

- $n = |V|$ é o número de nós e r , o raio de comunicação;
- $V = v_0, v_1, \dots, v_{n-1}$, onde v_0 é o único nó sorvedouro, e v_i é um nó fonte para todo $0 < i < n$;
- $\langle i, j \rangle \in E$, se e somente se v_j é o pai de v_i , ou seja, v_i envia seus dados para o sorvedouro através de v_j ;

Definição 2 - Algoritmo de Consulta de Dados: Diferentemente de tradicionais esquemas de roteamento fim-a-fim que tentam manter rotas entre nós de origem e destino, este trabalho está focado em algoritmos de roteamento para consulta de dados (Boukerche et al., 2005), objetivando dar suporte à agregação de dados *in-network*. Neste algoritmo, um *sink* equipado com um dispositivo de comunicação mais potente, envia uma consulta a todos os nós da rede em um único salto. Em seguida, os nós sensores com dados relacionados montam um pacote de resposta que é enviado para o nó *sink*, em múltiplos saltos. Em cada salto, os nós sensores podem olhar o conteúdo dos pacotes e efetuar agregação de dados em múltiplos pacotes de entrada para encaminhar um único pacote agregado, de acordo com a função de agregação especificada na consulta. Neste trabalho, consideramos funções simples de agregação de dados (Krishnamachari et al., 2002) como: *máximo*, *mínimo* e *média*.

Definição 3 - Mensagem de Broadcast: Uma mensagem de *broadcast* é uma forma de comunicação em que um nó (origem) $s \in V(G)$ dissemina pacotes a todos os vizinhos.

3.1.1 GRDA Selection

O algoritmo proposto (definido no Algoritmo 1) inicia quando o nó *sink* envia uma consulta para toda a rede, em um único salto (Linha 8), também ilustrado na Figura 3.3(a). Cada nó, que recebe a consulta do *sink*, estima sua distância para o *sink* usando a técnica do RSSI (Linhas 9–10), descrita em (Faheem et al., 2010) e (Benkic et al., 2008). Em seguida, cada nó envia um pacote de anúncio aos seus vizinhos a um salto de distância, contendo sua distância estimada até o *sink* (Linha 11), ilustrado na Figura 3.3(b). Se um nó tem qualquer dado, relacionado à consulta do *sink*, este agrega esse dado à lista de agregados (Linhas 12-14).

Para realizar a agregação de dados de todos os pacotes recebidos, um temporizador é iniciado (Linhas 15–16) e um tempo de espera é definido, de acordo com o valor do RSSI do *sink* recebido por cada nó. Neste caso, quanto menor for o valor do RSSI recebido, menor será o tempo que o um nó terá para agregar as informações. Esse tempo, é proporcional à distância do nó em relação *sink*. Quando um nó recebe um pacote de anúncio de seus vizinhos, estas informações são armazenadas em uma tabela de roteamento (Linhas 17–18) além de serem usadas para o encaminhamento dos dados agregados no próximo salto (Linha 19). Desta forma, com as distâncias de cada vizinho em sua tabela de roteamento, o nó pode escolher o vizinho mais próximo do *sink*.

Quando o temporizador de um nó expira (Linha 20), um pacote de resposta com os dados agregados é enviado e enviado ao vizinho mais próximo do *sink* (Linha 21), também ilustrado na Figura 3.3(c). Este pacote é recebido pelo próximo vizinho (Linha

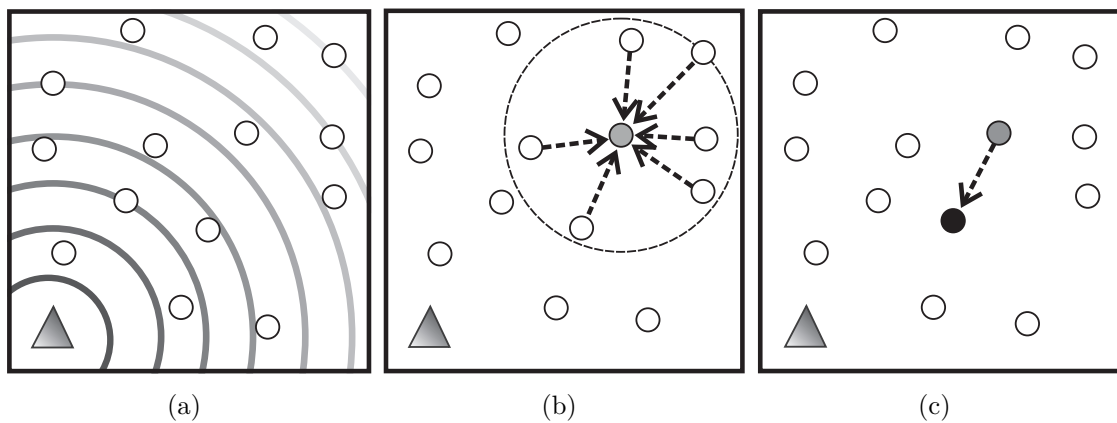


Figura 3.3. Exemplo de funcionamento do algoritmo *GRDA Selection*. Após disseminação da consulta pelo *sink* (a), nós regulares iniciam processo de escolha do próximo salto, com base na distância dos vizinhos em relação ao *sink* (b). Em seguida, o nó encaminha o pacote de dados agregados ao vizinho mais próximo do *sink* (c).

Algoritmo 1 GRDA Selection

Variáveis:

- 1: $neighborTable_i = \emptyset$; {Distâncias dos vizinhos ao *sink*}
- 2: $aggregData_i = \emptyset$; {Dados agregados a serem enviados}
- 3: $replyData_i$; {Dado coletado}
- 4: $sendTimer_i$; {Tempo de espera para encaminhar o dado agregado}
- 5: $timer_i$; {Tempo de espera para a agregação de dados}
- 6: $nextHop_i$; {Próximo salto para enviar os dados agregados}

Evento:

- 7: *Sink* recebe da central de monitoramento uma requisição: $QueryRequest(queryId_k, query_k)$;

Ação:

- 8: Envia $query(queryId_k, query_k)$; {a todos os nós regulares}

Evento:

- 9: $msg_i = query(queryId_k, query_k)$; tal que
- 10: $dist_i = distanceEstimation(msg_i)$;

Ação:

- 11: Envia $adv(n_i, dist_i) \forall n_j \in neig_i$. {Nó envia pacote de anúncio com sua distância}
- 12: **Se** $replyData_i = getReplyForQuery(query_k)$ **Então**
- 13: $aggregData_i := aggreData_i \cup replyData_i$; {Agrega dado}
- 14: **Fim Se**
- 15: $sendTimer_i = aggregationTime(dist_i)$; {Tempo de agregação}
- 16: $timer_i.start(sendTimer_i)$; {Início do temporizador}

Evento:

- 17: $msg_i = adv(neighbor_k, dist_k)$;

Ação:

- 18: $neighborTable_i = neighborTable_i \cup (neighbor_k, dist_k)$;
- 19: $nextHop_i = neighborClosestToSink()$;

Evento:

- 20: $timer_i.timeOut()$; {Temporizador expirou}

Ação:

- 21: Envia $reply(aggregData_i)$ para $nextHop_i$; {Encaminha dado agregado}
- 22: $aggregData_i = \emptyset$;

Evento:

- 23: $msg_i = reply(aggregData_k)$; {Recebe pacote com dados agregados de vizinho}

Ação:

- 24: **Se** $n_i \neq 0$ **Então** {Sou um nó regular?}
 - 25: $aggregData_i := aggreData_i \cup aggreData_k$;
 - 26: **Se** $timer_i.isActive()$ **Então** {Dado já foi encaminhado?}
 - 27: Envia $reply(aggregData)$ to $nextHop_i$;
 - 28: $aggregData_i = \emptyset$;
 - 29: **Fim Se**
 - 30: **Senão** {Eu sou o nó *sink*}
 - 31: Envia dado à central de monitoramento.
 - 32: **Fim Se**
-

23), que provavelmente não teve o seu temporizador expirado ainda, desde que este seja o mais próximo do *sink*. O dado recebido é então agregado à lista de dados agregados deste último salto (Linha 25) que irá encaminhar o dado quando seu timer também expirar (Linhas 20-22). Este processo é repetido até que o pacote seja entregue ao

nó *sink*, que neste caso recebe o dado e envia à central de monitoramento (Linha 31). Entretanto, em raros casos é possível que o temporizador deste nó já tenha expirado (e ele já tenha enviado seus dados agregados) provavelmente devido a erros de RSSI que resultaram em erros no cálculo do tempo de espera. Nestes casos, o pacote recebido será encaminhado imediatamente para o vizinho mais próximo do *sink* (Lines 26-29).

3.1.2 GRDA Election

O algoritmo *GRDA Election*, é uma variação da primeira abordagem. Nesta solução, os nós que irão participar do encaminhamento dos pacotes elegem a si próprios como próximo salto, sem a necessidade de troca de informações entre os vizinhos a um salto de distância.

O algoritmo (definido no Algoritmo 2), inicia quando o nó *sink* envia uma consulta para toda a rede (Linha 6), também em um único salto. Cada nó que recebe a consulta, estima sua distância até o *sink* através da técnica do RSSI (Linhas 7–8). Logo em seguida se um nó possui qualquer dado relacionado a esta consulta do nó *sink*, este agrega esse dado à sua lista de dados agregados (Linhas 9–11). Como no algoritmo *GRDA Selection*, um temporizador é iniciado em cada nó, afim de realizar a agregação de dados envolvendo os pacotes recebidos (Linhas 12–13) e ainda, um tempo de espera para o envio das informações é definido de acordo do o valor do RSSI recebido do *sink*.

A grande diferença entre as duas variações ocorre quando o temporizador expira. No *GRDA Election*, após o temporizador expirar (Linha 14), um pacote de resposta é gerado contendo tando o dado agregado quando a distância do nó até o *sink*. O pacote

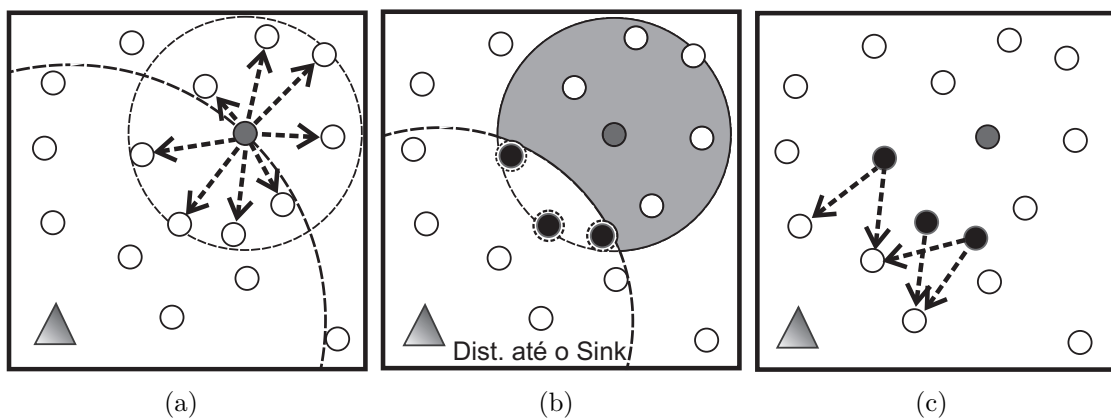


Figura 3.4. Exemplo de funcionamento do algoritmo *GRDA Election*. Após disseminação da consulta pelo *sink* e cálculo da distância, nós regulares iniciam processo de transmissão de pacotes via mensagem única de *broadcast* (a), auto-eleição do próximo salto (b) e repasse de pacotes (c).

Algoritmo 2 GRDA Election

Variáveis:

- 1: $aggregData_i = \emptyset$; {Dados agregados a serem enviados}
- 2: $replyData_i$; {Dado coletado}
- 3: $sendTimer_i$; {Tempo de espera para encaminhar o dado agregado}
- 4: $timer_i$; {Tempo de espera para a agregação de dados}

Evento:

- 5: *Sink* recebe da central de monitoramento uma requisição: $request(queryId_k, query_k)$;

Ação:

- 6: Envia $query(queryId_k, query_k)$; {a todos os nós regulares}

Evento:

- 7: $msg_i = query(queryId_k, query_k)$; tal que
- 8: $dist_i = distanceEstimation(msg_i)$;
- 9: **Se** $replyData_i = getReplyForQuery(query_k)$ **Então**
- 10: $aggregData_i := aggreData_i \cup replyData_i$; {Agrega dado}
- 11: **Fim Se**
- 12: $sendTimer_i = aggregationTime(dist_i)$; {Tempo de agregação}
- 13: $timer_i.start(sendTimer_i)$; {Início do temporizador}

Evento:

- 14: $timer_i.timeOut()$; {Temporizador expirou}

Ação:

- 15: Mensagem $broadcast\ reply(aggregData_i, dist_i)$; {Envia dado agregado e distância em relação ao *sink*}
- 16: $aggregData_i = \emptyset$;

Evento:

- 17: $msg_i = reply(aggregData_k, dist_k)$; {Recebe pacote com dados agregados de vizinho}
- 18: **Se** $n_i \neq 0$ **Então** {Sou um nó regular?}

Ação:

- 19: **Se** $timer_i.isActive()$ **Então** {Ainda tenho tempo para agregar dados?}
 - 20: **Se** $dist_i < dist_k$ **Então** {Estou próximo ao sink}
 - 21: $aggregData_i = aggreData_i \cup aggreData_k$;
 - 22: **Fim Se**
 - 23: **Fim Se**
 - 24: **Senão** {Eu sou o nó *sink*}
 - 25: Envia dado à central de monitoramento.
 - 26: **Fim Se**
-

é enviado a todos os vizinhos via mensagem *broadcast* (Linha 15, também ilustrado na Figura 3.4(a)). O nó que recebe o pacote, elege a si próprio como o salto mais próximo do *sink*, no lugar do nó que envia o pacote (Linha 20, também ilustrado na Figura 3.4(b)). O nó então, irá agregar o dado recebido (Linha 21) e aguardar o tempo de agregação para só então enviar o pacote com os dados agregados, via mensagem de broadcast, aos vizinhos a um salto de distância. (Linhas 14-16, também ilustrado na Figura 3.4(c)). O processo é repetido até que o *sink* receba as informações.

No *GRDA Election*, mas de um nó pode eleger a si próprio como o próximo salto. Neste caso, informações duplicadas podem ser enviados, conforme ilustrado na Fi-

gura 3.4(c). Contudo, uma vez que o algoritmo realiza agregação dados em cada salto, as informações duplicadas podem ser facilmente removidas antes do encaminhamento ao próximo salto. Além disso, afim de reduzir a quantidade de informações enviadas, o faixa de comunicação do *sink* (RSSI), em relação ao nó que possui o pacote a ser encaminhado, o foi reduzido, conforme ilustrado na Figura 3.4(b). Desta forma, o número de vizinhos que irão participar da eleição no próximo salto é reduzido e, somente os nós que estão dentro da nova faixa de comunicação irão encaminhar pacotes ao próximo salto. Este procedimento, evita que vizinhos não tão próximos ao *sink*, ou que possuem distância maior que a do nó que enviou o pacote via mensagem de *broadcast*, sejam elegíveis. Como exemplo, na Figura 3.4(a), o nó que realiza o anúncio via mensagem de *broadcast* tem quatro vizinhos com distância menor que a do nó que enviou o pacote agregado mas, como já visto na Figura 3.4(b), somente três estavam na faixa de distância que permitiu a auto-eleição como próximo salto para o encaminhamento dos dados.

Em todos os casos, apenas os dados recebidos pelos vizinhos que estão mais próximos do *sink* são agregados no nó auto-elegido, e que consequentemente, irá encaminhar os pacote com os dados agregados. (Linhas 17–23). Diferente do *GRDA Selection*, no *GRDA Election* não há troca de informações entre vizinhos. Ao invés disso, os dados são transmitidos apenas uma vez, quando temporizador do nó expira e, desta forma, os pacotes recebidos depois serão descartados.

3.2 Considerações Finais

Nesta capítulo apresentamos um novo algoritmo de roteamento geográfico com agregação de dados para redes de sensores sem fio que não requer informações de posicionamento dos nós: o *GRDA (Greedy Routing and Data Aggregation)*. Nosso algoritmo tira proveito da possibilidade de se equipar o nó *sink* com um dispositivo de comunicação de maior potência. Desta forma, em um único salto, o nó *sink* pode enviar uma consulta para toda a rede. O algoritmo foi proposto com duas variações: o *GRDA Selection* e o *GRDA Election*.

No *GRDA Selection*, o nó *sink*, equipado com um dispositivo de comunicação potente, envia uma consulta para toda a rede, em um único salto. Cada nó, que recebe a consulta e estima sua distância para o *sink* através da técnica do RSSI. Cada nó, envia um pacote de anúncio aos seus vizinhos a um salto de distância, contendo sua distância estimada até o *sink*. Em seguida, um temporizador é iniciado com base na distância estimada até o *sink*. Quando o temporizador de um nó expira, seu pacote é

então encaminhado ao vizinho que está geograficamente mais próximo do *sink*.

No algoritmo *GRDA Election*, os nós que irão participar do encaminhamento dos pacotes elegem a si próprios como próximo salto, sem a necessidade de troca de informações entre os vizinhos a um salto de distância. Como no algoritmo *GRDA Selection*, o nó *sink* envia uma consulta que alcança a todos os nós da, também em um único salto. Cada nó que recebe a consulta, estima sua distância até o *sink* através da técnica do RSSI e configura um temporizador para o envio de pacotes. Contudo, para o envio das informações é definido de acordo do o valor do RSSI recebido do *sink*. Após o temporizador expirar, um pacote de resposta é gerado contendo tanto o dado agregado quando a distância do nó até o *sink*. Contudo, ao contrário do primeiro algoritmo, no *GRDA Election* o pacote é enviado a todos os vizinhos via mensagem *broadcast*. O nó então, agrega o dado recebido e aguarda o tempo de agregação para só então enviar o pacote com os dados agregados, via mensagem de *brodcast*, aos vizinhos a um salto de distância. O processo é repetido até que o *sink* receba as informações.

No próximo capítulo, os algoritmos propostos são avaliados e comentados. Além disso, comparamos sua performance com o algoritmo *Flooding+Aggregation*, algoritmo básico de agregação de dados que usa o *flooding* clássico para divulgar a consulta do *sink* em múltiplos saltos e o retorno dos nós é feito usando a árvore inversa do *flooding* e realizando agregação de dados nos nós que possuem filhos. A metodologia adotada no trabalho é apresentada na seção 4.1. Na seção 4.2, apresentamos a avaliação de desempenho, baseada nas métricas: *Pacotes transmitidos*, *Pacotes entregues*, *Economia de transmissões* e *Acertos na escolha do próximo salto*.

Capítulo 4

Metodologia e Avaliação de Desempenho

Na presente seção, avaliamos o desempenho dos algoritmo *GRDA* propostos. Também comparamos sua performance com o algoritmo *Flooding+Aggregation*, algoritmo básico de agregação de dados que usa o *flooding* clássico para divulgar a consulta do *sink* em múltiplos saltos e o retorno dos nós é feito usando a árvore inversa do *flooding* e realizando agregação de dados nos nós que possuem filhos. O algoritmo *Flooding+Aggregation* permitirá avaliar o ganho do algoritmo proposto em relação a outro que já faz agregação.

O foco de nossa avaliação está na aplicação da nova técnica de encaminhamento guloso bem como na técnica de agregação de dados, baseada na métrica para o cálculo do tempo de espera para agregação. Espera-se também entender o funcionamento do algoritmo proposto nos mais diversos cenários. Na seção 4.1 apresentamos a metodologia utilizada e a na seção 4.2 a avaliação de desempenho realizada neste trabalho.

4.1 Metodologia

A avaliação de performance foi realizada através de simulações, realizadas com o simulador Sinalgo (ETH-Zurich, 2012), que provê um ambiente completo para simulação de algoritmos distribuídos. As simulações foram realizadas em um campo de sensoramento de $130\text{ m} \times 130\text{ m}^2$. Em todos os resultados, as curvas representam valores médios, enquanto as barras de erro representam intervalos de confiança para 95% de confiança a partir de 33 instâncias independentes (sementes aleatórias). Quanto à topologia da rede, assumimos que a implantação dos nós obedece a uma grade perturbada, na qual a localização de cada nó é perturbada por um erro aleatório Gaussiano de

média zero, de forma a evitar buracos na rede e mantendo a densidade de 0.03 nós/m^2 . Desta forma, os nós tendem a ocupar uniformemente o campo de sensores sem formar uma grade regular. Os parâmetros de simulação foram baseados no sensor MicaZ e os valores utilizados são mostrados na Tabela 4.1.

Tabela 4.1. Cenário padrão de simulação.

Parâmetro	Valor padrão
Área monitorada	$130 \text{ m} \times 130 \text{ m}^2$
Quantidade de nós	512 nós (grade perturbada)
Densidade	0.03 nós/m^2
Alcance da comunicação	20 m
Imprecisão do RSSI a curta distância	5 % da dist. real
Posição do Sink	Centro da área monitorada

Durante as simulações, cada nó sensor gera um pacote de dados com eventos detectados. Neste aspecto, a economia de transmissões foi alcançada através da função de agregação. Esta função recebe um conjunto pacotes de entrada e retorna apenas um pacote com os dados agregados do conjunto de pacotes. Esta função é efetuada em cada salto de rede. Além disso, o número de pacotes encaminhados sem a utilização da agregação de dados também foi registrado.

Para a avaliação de desempenho utilizamos quatro métricas, descritas a seguir:

- **Pacotes transmitidos:** número total de pacotes de dados transmitidos pelos nós sensores;
- **Pacotes entregues:** razão entre ao número total de pacotes de dados gerados pelos nós sensores e a quantidade de pacotes de dados recebidos pelo nó *sink*;
- **Economia de transmissões:** razão entre o número de transmissões que não foram necessárias devido à agregação de dados e o número de transmissões que seriam efetuadas sem agregar dados;
- **Acertos na escolha do próximo salto:** um acerto ocorre quando um pacote sai de um nó em direção ao seu vizinho, geograficamente mais próximo do nó *sink*, enquanto que um erro se dá quando este pacote foi enviado para um vizinho que não é o mais próximo do *sink*, ou seja, ele poderia ter sido enviado para um vizinho melhor.

4.2 Avaliação de Desempenho

No presente capítulo, apresentamos a avaliação de desempenho dos algoritmos propostos neste trabalho. Avaliamos o impacto da escala da rede, o impacto da densidade, o impacto da imprecisão do RSSI a curta distância, o impacto do raio de comunicação, e finalmente, os erros na escolha do próximo salto. Em todos os casos, a taxa de pacotes entregues foi superior a 90 %.

4.2.1 Impacto da Escala da Rede

A escalabilidade é avaliada através do aumento da quantidade de nós da rede de 256 até 1536, mantendo a densidade em 0.03 nós/m^2 . Desta forma, a área monitorada é redimensionada de acordo com o número de nós. Quando aumentamos o tamanho da rede, a sobrecarga na rede aumenta na mesma proporção, pois todos os nós sensores geram um pacote de dados para transmitir. Na Figura 4.1(a), é possível verificar que no algoritmo *GRDA Election* o número de transmissões é menor que nos dois outros algoritmos, neste caso, o *Flooding+Aggregation* e o *GRDA Selection*.

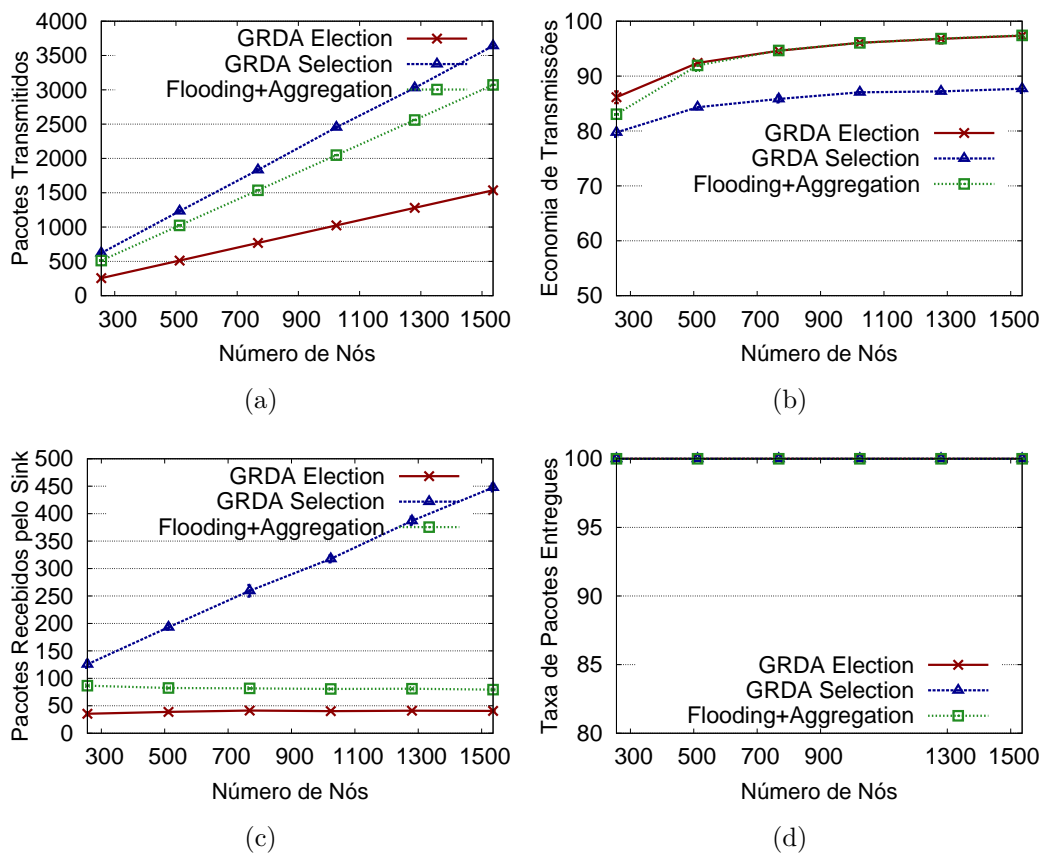


Figura 4.1. Avaliação de performance em relação à escala da rede.

A Figura 4.1(b) mostra claramente que o número de transmissões no algoritmo *GRDA Election* aumenta mais lentamente do que o clássico *Flooding+Aggregation* quando o número de nós na rede aumenta. Isto mostra que esta nova técnica de agregação de dados é mais eficiente no que se refere ao consumo de energia. Como é possível observar, o algoritmo *GRDA Election* possui uma taxa de economia superior, mesmo com poucos nós na rede, superando o algoritmo *Flooding+Aggregation*. Neste sentido, a economia de transmissões, característica da utilização de técnicas de agregação de dados, contribuiu para a redução da sobrecarga na rede, reduzindo quantidade de pacotes enviados.

O número de pacotes recebidos pelo *sink* também foi avaliado. Na Figura 4.1(c), é possível observar que o algoritmo *GRDA Election* apresenta um número menor de pacotes chegando no *sink*, uma vez que nesta abordagem cada nó só transmite um único pacote, diferente do *GRDA Selection* e do *Flooding+Aggregation*, que devido a fatores com processo de cálculo da distância dos em direção ao *sink* baseado no RSSI (no caso do *GRDA Selection*), possíveis erros de localização e montagem da árvore reversa (no caso do flooding), irão enviar pacotes mais de uma vez.

Finalmente, na Figura 4.1(d) é possível observar os algoritmos não foram afetados pelo aumento do número de nós, apresentando taxas de pacotes entregues a partir 99%.

4.2.2 Impacto da Densidade da Rede

O impacto da densidade da rede em nosso algoritmo foi avaliado começando com uma densidade relativamente baixa de 0.01 nós/m^2 e aumentando tal densidade até 0.05 nós/m^2 . Em densidades baixas, um nó terá poucos vizinhos, dificultando a agregação, enquanto que ao se aumentar a densidade, o número de vizinhos de cada nó aumentará. O gráfico da Figura 4.2(a) mostra a quantidade de pacotes transmitidos na rede ao se aumentar a densidade. Neste gráfico, é possível observar que mesmo a densidades baixas o algoritmo *GRDA Election* continua funcionando e que ele é capaz de funcionar de forma bem mais eficiente que o *Flooding+Aggregation*. Neste contexto, o algoritmo *GRDA Selection* apresenta uma pequena desvantagem em relação ao outros dois algoritmos, pois é capaz de enviar pacotes em direção ao *sink* após o tempo de espera para o envio de pacotes expirar. Esta característica aumenta o número de transmissões. Apesar desta desvantagem, o algoritmo *GRDA Selection* é confiável, pois aumenta a probabilidade de envio do pacote com as informações sensoreadas.

A economia de transmissões também foi avaliada. Na Figura 4.2(b), é possível observar que os algoritmos GRDA superam o algoritmo *Flooding+Aggregation*, mesmo em densidades baixas, confirmando a eficiência desta nova técnica de encaminhando

gulo, usando agregação de dados. Contudo, na Figura 4.2(c), é possível observar o aumento do número de pacotes chegando no *sink* com uma agregação menor, uma vez que o *sink* terá mais vizinhos que enviarão os pacotes diretamente para ele. Neste sentido, o algoritmo *GRDA Election* apresenta resultados semelhantes ao clássico *Flooding+Aggregation*, pois o número de vizinhos do nó *sink* que irão enviar um pacote com informações agregadas é o mesmo. Todos os algoritmos apresentaram taxas de pacotes entregues superiores a 98 %, como é possível observar na Figura 4.2(d).

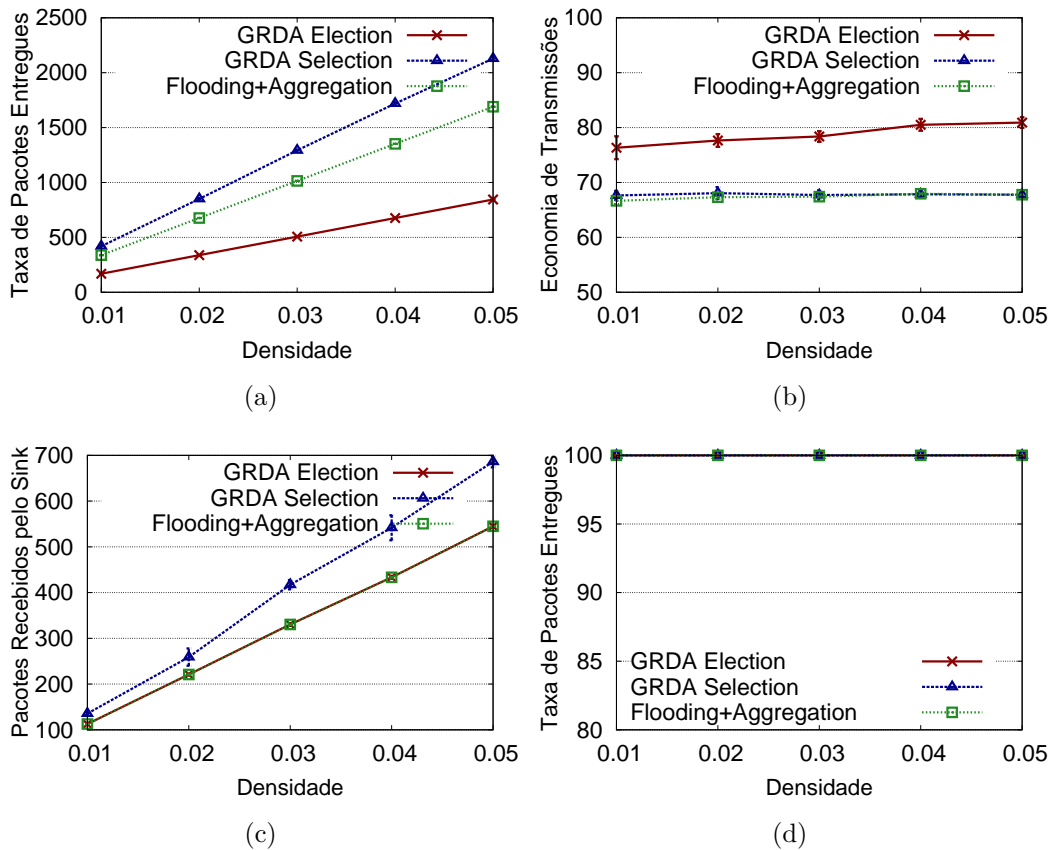


Figura 4.2. Avaliação de performance em relação à densidade da rede.

4.2.3 Impacto da imprecisão do RSSI a curta distância

O impacto da imprecisão do RSSI a curta distância (distância estimada entre os nós regulares) foi avaliada através do aumento desta imprecisão de 0 até 50 % da distância real entre os nós. Na Figura 4.3(a), é possível observar que tanto o algoritmo *GRDA Election* quanto o *Flooding+Aggregation*, não são afetados pelo aumento na imprecisão do RSSI. Também observamos que o número de transmissões no *GRDA Se-*

lection aumenta à medida que se aumenta o erro de RSSI, em todos os casos, ficando acima dos algoritmos *GRDA Election* e *Flooding+Aggregation*.

Na Figura 4.3(b), é possível observar que o algoritmo *GRDA Election* possui uma leve vantagem em relação ao algoritmo *Flooding+Aggregation*, no que se refere à economia de transmissões. Já o algoritmo *GRDA Selection*, apresenta uma desvantagem neste aspecto, pois o algoritmo escolhe o próximo salto baseado na distância estimada entre os nós vizinhos. Este comportamento já era esperado neste algoritmo, uma vez que a imprecisão no valor do RSSI influencia diretamente no cálculo da distância, feita em cada saldo.

O gráfico da Figura 4.3(c) mostra a quantidade de pacotes recebidos pelo nó *sink*. Neste resultado, fica evidente que o número de pacotes recebidos pelo *sink*, nos algoritmos *GRDA Election* e *Flooding+Aggregation* permanece praticamente constante mesmo ao aumentar bastante o número de nós. Contudo, no primeiro algoritmo, o número de pacotes é praticamente a metade do segundo. Neste contexto, o número de pacotes entregues se limita ao número de vizinhos do nó *sink* que possuem o mesmo raio de comunicação e ao número de transmissões, já mostrado anteriormente. Estes resultados, mostram novamente a viabilidade da técnica proposta de agregação de dados.

Nesta avaliação, todos os algoritmos apresentaram taxas de pacotes entregues superiores a 98 %, como é possível observar na Figura 4.3(d).

4.2.4 Impacto no Raio de Comunicação

Para avaliarmos o impacto do raio de comunicação dos nós em nosso algoritmo, começamos com um raio relativamente pequeno, de 15 m, e o aumentamos para 40 m. Raios de comunicação pequenos reduzem o número de nós vizinhos, o que consequentemente reduz o número de possibilidades para o próximo salto (Boukerche et al., 2008).

Na Figura 4.4(a), é possível observar que o *GRDA Election* consegue manter o número de pacotes transmitidos na rede baixo, mesmo em baixas faixas de comunicação. Podemos observar ainda a economia no número de transmissões em relação ao *Flooding+Aggregation* que, ao se aumentar o raio de comunicação, começa a transmitir mais pacotes, uma vez que o número de vizinhos próximos ao *sink* irá aumentar. Nos algoritmos GRDA esta característica é compensada pelo roteamento guloso, que sempre encaminha os pacotes aos vizinhos mais próximos do *sink*.

No gráfico da Figura 4.4(c), é possível perceber que mesmo quando aumentamos o raio de comunicação entre os nós, o número de pacotes recebidos pelo *sink* ainda é expressivamente menor no algoritmo *GRDA Election* que, neste caso, tem o número

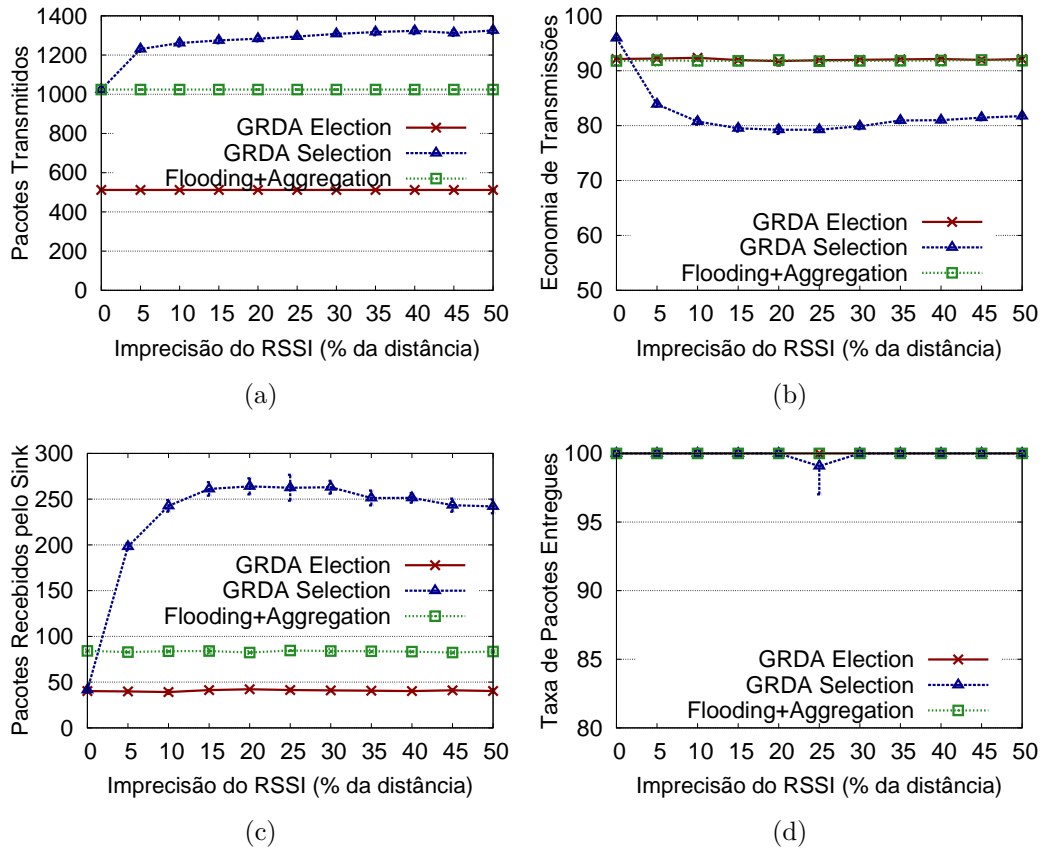


Figura 4.3. Avaliação de performance em relação à imprecisão do RSSI a curta distância.

de vizinhos diretos do *sink* aumentado e, portanto, maior possibilidade de envio direto dos pacotes (mas com menos pacotes agregados). Contudo, apesar de escolher o próximo salto mais próximo do *sink*, o *GRDA Selection* é afetado pelo erro padrão RSSI, enviando pacotes após o temporizador, configurado quando o *sink* envia a primeira consulta, expirar.

Já no caso do *Flooding+Aggregation*, não há escolha do próximo salto, pois os pacotes são enviados de acordo com a árvore inversa, criada com o primeiro *flooding* a partir do *sink*. Neste caso, com o raio de comunicação aumentado, temos um maior número de vizinhos em cada salto, o que aumentando o número de transmissões, uma vez que o *Flooding+Aggregation* não escolhe o próximo salto baseado na distância para o *sink*.

Finalmente, o gráfico da Figura 4.4(d) mostra que o número de pacotes entregues pelo *GRDA Selection* é de cerca de 95% em 15 m. Neste caso, o número de vizinhos é relativamente pequeno, aumentando as chances de falhas no envio das mensagens. O mesmo não ocorre no algoritmo *GRDA Election*, pois na maioria dos casos, este envia

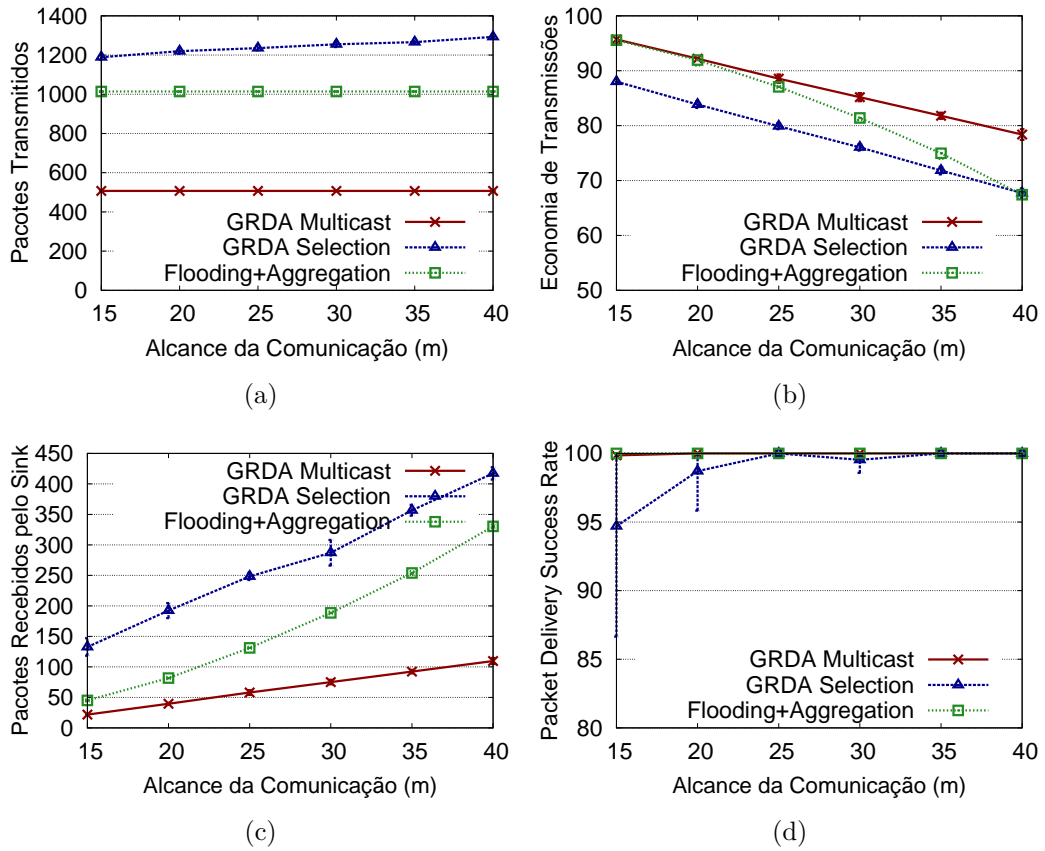


Figura 4.4. Avaliação de performance em relação à imprecisão do RSSI a curta distância.

mensagens a mais de um vizinho por salto, o que aumenta a confiabilidade na entrega das informações. Tal característica, pode ser considerada um fator positivo, pois os mesmos resultados são alcançados pelo *Flooding+Aggregation*, algoritmo clássico que apresenta altas taxas de entrega de pacotes.

4.2.5 Erros na Escolha do Próximo Salto

Os erros na escolha do próximo salto também foram avaliados. Até o momento, o algoritmo *GRDA Election* obteve praticamente o mesmo desempenho que o algoritmo básico de agregação de dados *Flooding+Aggregation*, e em alguns casos alcançando resultados superiores. Tal característica é um ponto positivo para a nossa abordagem, uma vez que o *flooding* é um dos algoritmos mais confiáveis em RSSFs. Entretanto, apesar de sua confiabilidade, o *flooding* não garante os menores nem os melhores caminhos na rede e muito menos a criação de árvores de escoamento de qualidade. Os algoritmos *GRDA*, por outro lado, são capazes de gerar árvores de escoamento muito mais próximas da ótima, uma vez que se baseiam em distâncias ao *sink*.

Para avaliarmos esse aspecto, os gráficos das Figuras 4.5(a), 4.5(b), 4.5(c) e 4.5(d), mostram a quantidade de erros na escolha do próximo salto de um pacote. Um acerto se dá quando um pacote sai de um nó em direção ao seu vizinho que está fisicamente mais próximo do nó *sink*, enquanto que um erro se dá quando este pacote foi enviado para um vizinho que não é o mais próximo do *sink*, ou seja, ele poderia ter sido enviado para um vizinho melhor. Quanto menor a taxa de erro, melhor será a agregação e o caminho dos pacotes em direção ao *sink*.

No gráfico da Figura 4.5(d), é possível observar o comportamento da escolha do próximo vizinho ao se aumentar o erro de RSSI, uma vez que este é um dos fatores mais diretamente relacionados à escolha do vizinho em nossa abordagem. Como pode-se observar, a taxa de erro obtida pelo algoritmo *GRDA Election* é muito inferior à do algoritmo *Flooding+Aggregation*, mostrando que a abordagem possui todas as vantagens deste último, mas obtendo uma árvore de agregação muito mais eficiente. Contudo, observamos uma pequena desvantagem do algoritmo *GRDA Election*, pois neste caso o algoritmo possui mais de um vizinho elegível no próximo salto, o que aumenta o número de possibilidades para o cálculo do próximo salto. Apesar desta característica, os resultados observados nas Figuras 4.5(a), 4.5(b) e 4.5(d), mostram que mesmo com um número maior de vizinhos elegíveis o *GRDA Election*, em relação ao *Flooding+Aggregation*, apresenta melhor desempenho no que se refere às escolhas de encaminhamento.

Finalmente, vê-se que a qualidade do algoritmo *GRDA Selection* cai ao se aumentar o erro de RSSI. Neste caso, a taxa de erros na escolha do próximo salto é afetada. Contudo, em todos os casos, esta taxa fica bem abaixo da apresentada pelo *Flooding+Aggregation*.

4.3 Considerações Finais

No presente capítulo, as duas variações do algoritmo *GRDA* foram avaliadas. Sua performance foi comparada com o algoritmo *Flooding+Aggregation*, que utiliza um *flooding* clássico para divulgar a consulta do *sink* em múltiplos saltos. Avaliamos a aplicação da nova técnica de encaminhamento guloso bem como a agregação de dados, baseada na no RSSI do *sink* para o cálculo do tempo de espera para agregação. Nossos experimentos foram simulados com o *framework* Sinalgo, que provê um ambiente completo para simulação de algoritmos distribuídos. Em nenhum momento o *flooding* clássico superou os dois algoritmos juntos. Contudo, em alguns cenários um deles era mais afetado do que o outro, o que já era esperado pela característica singular de cada abordagem.

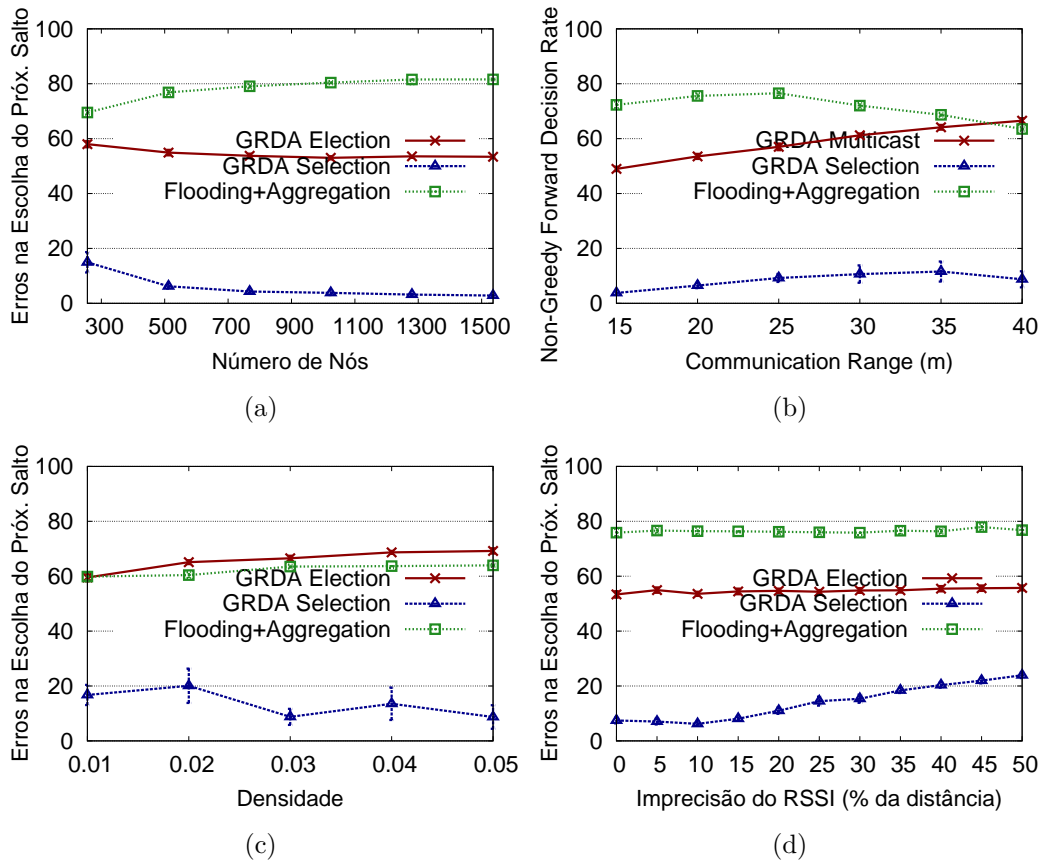


Figura 4.5. Avaliação de performance em relação à imprecisão do RSSI a curta distância.

Entre os resultados encontrados destaco:

- **Impacto na Escala da Rede:** O algoritmo *GRDA Election* foi capaz de transmitir menos pacotes na rede, mesmo com um número pequeno de nós. Consequentemente, o número de pacotes de chegou ao *sink* também foi menor, o que contribuiu para a alta taxa de economia de transmissões. Já o algoritmo *GRDA Selection*, esteve em desvantagem, pois é afetado pelo erro padrão de RSSI e além disso, pode enviar pacotes após o temporizador expirar.
- **Impacto na Densidade da Rede:** Em densidades baixas, um nó terá poucos vizinhos, dificultando a agregação, enquanto que ao se aumentar a densidade, o número de vizinhos de cada nó aumentará. Neste contexto, mesmo a densidades baixas o algoritmo *GRDA Election* continua funcionando e que ele é capaz de funcionar de forma bem mais eficiente que o *Flooding+Aggregation*. O algoritmo *GRDA Selection* apresenta uma pequena desvantagem em relação aos outros dois algoritmos, pois é capaz de enviar pacotes em direção ao *sink* após o tempo

de espera para o envio de pacotes expirar. Esta característica aumenta o número de transmissões mas torna o algoritmo confiável, pois aumenta a probabilidade de envio do pacote com as informações sensoreadas.

- **Impacto na Imprecisão do RSSI a curta Distância:** Na maioria dos experimentos, tanto o algoritmo *GRDA Election* quanto o *Flooding+Aggregation*, não são afetados pelo aumento na impressão do RSSI, pois o número de vizinhos elegíveis a próximo salto é bem maior que no *GRDA Selection*.
- **Impacto no Raio de Comunicação:** Raios de comunicação pequenos reduzem o número de nós vizinhos, o que consequentemente reduz o número de possibilidades para o próximo salto. Neste sentido, o *GRDA Election* consegue manter o número de pacotes transmitidos na rede baixo, mesmo em baixas faixas de comunicação. Podemos observar ainda a economia no número de transmissões, ao contrário do algoritmo *Flooding+Aggregation* que, ao se aumentar o raio de comunicação, começa a transmitir mais pacotes, uma vez que o número de vizinhos próximos ao *sink* irá aumentar. Nos algoritmos *GRDA* esta característica é compensada pelo roteamento guloso, que sempre encaminha os pacotes aos vizinhos mais próximos do *sink*.
- **Erros na Escolha do Próximo Salto:** O algoritmo *GRDA Election* obteve praticamente o mesmo desempenho que o algoritmo básico de agregação de dados *Flooding+Aggregation*, e em alguns casos alcançando resultados superiores. Tal característica é um ponto positivo para a nossa abordagem, uma vez que o *flooding* é um dos algoritmos mais confiáveis em RSSFs. A taxa de erro obtida pelo algoritmo *GRDA Election* é muito inferior à do algoritmo *Flooding+Aggregation*, mostrando que a abordagem possui todas as vantagens deste último, mas obtendo uma árvore de agregação muito mais eficiente. A qualidade do algoritmo *GRDA Selection* cai ao se aumentar o erro de RSSI. Neste caso, a taxa de erros na escolha do próximo salto é afetada. Contudo, em todos os casos, esta taxa fica bem abaixo da apresentada pelo *Flooding+Aggregation*.

Capítulo 5

Conclusões

Neste trabalho, propomos um novo algoritmo de roteamento geográfico com agregação de dados para redes de sensores sem fio que não requer informações de posicionamento dos nós: o *GRDA* (*Greedy Routing and Data Aggregation*). O algoritmo, possui todas as vantagens de um algoritmo de roteamento geográfico tais como robustez, escalabilidade, dinamicidade e entrega confiável dos dados, sem a necessidade do posicionamento dos nós ao mesmo tempo que provê o recurso de agregação de dados na rede. Em nossa proposta, uma única mensagem, a partir do nó *sink* alcança todos os nós da rede com diferentes potências (valores de RSSIs), sendo que os nós mais distantes possuem sinal mais baixo, ao contrário dos nós mais próximos, que recebem a mensagem com um sinal mais forte.

O algoritmo foi proposto com duas variações: o *GRDA Selection* e o *GRDA Election*. No *GRDA Selection*, o nó *sink*, equipado com um dispositivo de comunicação potente, envia uma consulta para toda a rede, em um único salto. Cada nó, que recebe a consulta e estima sua distância para o *sink* através da técnica do RSSI. Cada nó, envia um pacote de anúncio aos seus vizinhos a um salto de distância, contendo sua distância estimada até o *sink*. Em seguida, um temporizador é iniciado com base na distância estimada até o *sink*. Este temporizador será o responsável pela realização da agregação de dados. Quanto menor for o valor do RSSI recebido, menor será o tempo que o um nó terá para agregar as informações. Quando o temporizador de um nó expira, seu pacote é então encaminhado ao vizinho que está geograficamente mais próximo do *sink*. Em casos raros, é possível que o temporizador deste nó já tenha expirado (e ele já tenha enviado seus dados agregados) provavelmente devido a erros de RSSI que resultaram em erros no cálculo do tempo de espera. Nestes casos, o pacote recebido será encaminhado imediatamente para o vizinho mais próximo do *sink*.

No algoritmo *GRDA Election*, os nós que irão participar do encaminhamento

dos pacotes elegem a si próprios como próximo salto, sem a necessidade de troca de informações entre os vizinhos a um salto de distância. Como no algoritmo *GRDA Selection*, o nó *sink* envia uma consulta que alcança a todos os nós da, também em um único salto. Cada nó que recebe a consulta, estima sua distância até o *sink* através da técnica do RSSI e configura um temporizador para o envio de pacotes. Contudo, para o envio das informações é definido de acordo do o valor do RSSI recebido do *sink*. Após o temporizador expirar, um pacote de resposta é gerado contendo tanto o dado agregado quando a distância do nó até o *sink*. Diferentemente do primeiro algoritmo, no *GRDA Election* o pacote é enviado a todos os vizinhos via mensagem *broadcast*. Então, o nó que recebe o pacote, elege a si próprio como o salto mais próximo do *sink*, no lugar do nó que enviou o pacote. O nó então, agrega o dado recebido e aguarda o tempo de agregação para só então enviar o pacote com os dados agregados, via mensagem de *broadcast*, aos vizinhos a um salto de distância. O processo é repetido até que o *sink* receba as informações.

Um extenso conjunto de experimentos mostra claramente o bom desempenho dos algoritmos propostos. Além disso, as abordagens apresentaram economias significativas de energia para os cenários analisados. Em todos os cenários, a taxa de entrega nos algoritmos *GRDA*, foi superior a 95 %. O algoritmo *GRDA Election*, apresentou uma taxa de economia de transmissões superior, mesmo em cenários com poucos nós na rede, superando o algoritmo *Flooding+Aggregation*. Neste sentido, a economia de transmissões é característica da utilização de técnicas de agregação de dados, confirmando a eficiência de nossa abordagem. Apesar do grande número de transmissões do algoritmo *GRDA Selection*, com o aumento da densidade, este é capaz de enviar pacotes em direção ao *sink* após o temporizador de envio de pacotes expirar. Esta característica torna a abordagem confiável e aumenta as chances de um pacote chegar ao *sink*. Desta forma, concluímos que todas as técnicas foram capazes de alcançar resultados significativos, em relação à economia de energia e entrega de dados.

5.1 Aplicabilidade da Proposta

Neste trabalho consideramos um nó *sink* equipado com um dispositivo de comunicação potente, de forma que em um único salto seja possível alcançar todos os nós regulares na rede. Nossa abordagem é aplicável em diversos cenários amazônicos, dentre eles, o projeto ATTO (*Amazonian Tall Tower Observatory*), para o qual foi proposto. Tal projeto consiste em uma torre no meio da floresta amazônica, com uma estrutura física de 320 metros de altura, comparável à torre *Eiffel*, cujo foco é mapear eventos da

região e fornecer medidas confiáveis de fontes e sumidouros de gases de efeito estufa como CO_2 , CH_4 e N_2O (Tollefson, 2010). A torre, a primeira desta natureza no cenário amazônico, está instalada na Reserva de Desenvolvimento Sustentável do Uatumã, no interior do Amazonas. Neste contexto, o algoritmo *GRDA* foi proposto como uma solução de encaminhamento guloso utilizando agregação de dados para os nós sensores em volta da torre, que servirão como *sink*. Ao se aproveitar a comunicação de alto alcance do *sink*, nossa solução evita a dependência de um *flooding* inicial, que tem comportamento ruim em ambientes com baixa qualidade de comunicação, como é o caso na região amazônica.

5.2 Desafios e Trabalhos Futuros

Neste trabalho, propomos um novo protocolo de roteamento geográfico com agregação de dados para RSSFs: o *GRDA* (*Greedy Routing and Data Aggregation*). Tal protocolo parte do princípio de que o nó *sink* possui uma capacidade alta de comunicação de modo a mandar uma única mensagem alcançando todos os nós da rede. Com base apenas no RSSI da mensagem do *sink* chegando em cada um dos nós, propomos um novo e inovador esquema de roteamento e agregação de dados. Nosso algoritmo de agregação de dados utiliza um temporizador para esperar pelos pacotes a serem agregados que também é feito com base na mesma informação de RSSI.

Uma série de experimentos foi realizada para avaliar o protocolo proposto e compará-lo com um dos protocolos mais conhecidos da literatura: o *Flooding+Aggregation*, que realiza agregação de dados, baseada na árvore de escoamento obtida a partir de um *flooding*. Os resultados obtidos mostram claramente os benefícios introduzidos pelo esquema de agregação de dados proposto bem como a qualidade superior da árvore de agregação obtida em relação ao *flooding* clássico, enquanto implementa um esquema simples de roteamento guloso durante o encaminhamento das mensagens até o *sink*.

Apesar de obtermos resultados satisfatórios com vantagens importantes, algumas limitações devem ser exploradas em trabalhos futuros, como, por exemplo, a combinação de nossa solução com algoritmos de reconhecimento de perímetro e desvio de buracos, que sejam também baseadas apenas nas informações disponíveis de RSSI, algo ainda não explorado na literatura.

Referências Bibliográficas

- Abbasi, A. A. & Younis, M. (2007). A survey on clustering algorithms for wireless sensor networks. *Comput. Commun.*, 30(14-15):2826--2841.
- Abdelgawad, A. & Bayoumi, M. (2012). *Resource-Aware Data Fusion Algorithms for Wireless Sensor Networks*, volume 118 of *Lecture Notes in Electrical Engineering*. Springer.
- Agarwal, S.; Satpathy, S. & Sharma, L. (2011). Routing techniques for reliable wireless sensor networks. In *Communication Systems and Network Technologies (CSNT), 2011 International Conference on*, pp. 54 –57.
- Ahmed, A.; Shi, H. & Shang, Y. (2003). A survey on network protocols for wireless sensor networks. In *Information Technology: Research and Education, 2003. Proceedings. ITRE2003. International Conference on*, pp. 301 – 305.
- Akkaya, K. & Younis, M. (2005). A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3:325--349.
- Akyildiz, I.; Su, W.; Sankarasubramaniam, Y. & Cayirci, E. (2002a). A survey on sensor networks. *Communications Magazine, IEEE*, 40(8):102–114.
- Akyildiz, I. F.; Su, W.; Sankarasubramaniam, Y. & Cayirci, E. (2002b). Wireless sensor networks: a survey. *Computer Networks*, 38:393--422.
- Al-karaki, J. N. & Kamal, A. E. (2004). Routing techniques in wireless sensor networks: A survey. *IEEE Wireless Communications*, 11:6--28.
- Alasem, R.; Reda, A. & Mansour, M. (2011). Location based energy-efficient reliable routing protocol for wireless sensor networks. In *Proceeding of 10th WSEAS international conference on electronics, hardware, wireless and optical communications, NEHIPISIC'11*, pp. 180--185, Stevens Point, Wisconsin, USA. World Scientific and Engineering Academy and Society (WSEAS).

- Benkic, K.; Malajner, M.; Planinsic, P. & Cucej, Z. (2008). Using rssi value for distance estimation in wireless sensor networks based on zigbee. In *Systems, Signals and Image Processing, 2008. IWSSIP 2008. 15th International Conference on*, pp. 303–306.
- Boukerche, A.; Chatzigiannakis, I. & Nikolettseas, S. (2005). Power-efficient data propagation protocols for wireless sensor networks. *Simulation*, 81(6):399–411.
- Boukerche, A.; Oliveira, H.; Nakamura, E. & Loureiro, A. (2007). Localization systems for wireless sensor networks. *Wireless Communications, IEEE*, 14(6):6–12.
- Boukerche, A.; Oliveira, H.; Nakamura, E. & Loureiro, A. (2008). A novel location-free greedy forward algorithm for wireless sensor networks. In *Communications, 2008. ICC '08. IEEE International Conference on*.
- Boukerche, A.; Oliveira, H.; Nakamura, E. & Loureiro, A. (2009). Dv-loc: a scalable localization protocol using voronoi diagrams for wireless sensor networks. *Wireless Communications, IEEE*, 16(2):50–55.
- Caruso, A.; Chessa, S.; De, S. & Urpi, R. (2005). Gps free coordinate assignment and routing in wireless sensor networks. In *In IEEE INFOCOM*, pp. 150–160.
- Cohen, N. H.; Cohen, N. H.; Purakayastha, A.; Purakayastha, A.; Turek, J.; Turek, J.; Wong, L.; Wong, L.; Yeh, D. & Yeh, D. (2001). Challenges in flexible aggregation of pervasive data. Relatório técnico, IBM Research Division, Thomas J. Watson Research Center, P.O.Box 704, Yorktown Heights, NY.
- Dhanapala, D. C. & Jayasumana, A. P. (2011). Geo-logical routing in wireless sensor networks. In *SECON*, pp. 305–313. IEEE.
- Dinh, T. L. & Nguyen, D. T. (2010). Greedy geographic routing with path optimization in wireless sensor networks. In *RIVF*, pp. 1–6.
- ETH-Zurich, D. C. G. a. (2012). Sinalgo - simulator for network algorithms. In <http://dgc.ethz.ch/projects/sinalgo/>.
- Faheem, A.; Virrankoski, R. & Elmusrati, M. (2010). Improving rssi based distance estimation for 802.15.4 wireless sensor networks. In *Wireless Information Technology and Systems (ICWITS), 2010 IEEE International Conference on*, pp. 1–4.
- Goyal, D. & Tripathy, M. (2012). Routing protocols in wireless sensor networks: A survey. In *Advanced Computing Communication Technologies (ACCT), 2012 Second International Conference on*, pp. 474–480.

- Haider, R.; Javed, M. & Khattak, N. (2007). Eagr: Energy aware greedy routing in sensor networks. In *Future Generation Communication and Networking (FGCN 2007)*, volume 2, p. 344349.
- Hassanzadeh, A.; Stoleru, R. & Chen, J. (2011). Efficient flooding in Wireless Sensor Networks secured with neighborhood keys. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2011 IEEE 7th International Conference on*, pp. 119 –126.
- Heinzelman, W. R.; Kulik, J. & Balakrishnan, H. (1999). Adaptive protocols for information dissemination in wireless sensor networks. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, MobiCom '99, pp. 174--185, New York, NY, USA. ACM.
- Holland, M.; Aures, R. & Heinzelman, W. (2006). Experimental investigation of radio performance in wireless sensor networks. In *Wireless Mesh Networks, 2006. WiMesh 2006. 2nd IEEE Workshop on*, pp. 140 –150.
- Jacinto, R. M. P. (2012). Modelação da Propagação numa Rede de Sensores sem Fios. Dissertação de mestrado, Universidade Nova de Lisboa.
- Karl, H. & Willig, A. (2005). *Protocols and Architectures for Wireless Sensor Networks*. John Wiley & Sons.
- Karp, B. & Kung, H. T. (2000). Gpsr: Greedy perimeter stateless routing for wireless networks. In *MOBICOM*, pp. 243--254.
- Karp, B. N. (2000). Geographic routing for wireless networks. Relatório técnico, Harvard University.
- Kermarrec, A.-M. & Tan, G. (2010). Greedy geographic routing in large-scale sensor networks: a minimum network decomposition approach. In *Proceedings of the eleventh ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '10, pp. 161--170, New York, NY, USA. ACM.
- Kim, Y.-J.; Govindan, R.; Karp, B. & Shenker, S. (2005). Geographic routing made practical. In *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation - Volume 2*, NSDI'05, pp. 217--230, Berkeley, CA, USA. USENIX Association.
- Krishnamachari, B.; Estrin, D. & Wicker, S. B. (2002). The impact of data aggregation in wireless sensor networks. In *Proceedings of the 22nd International Conference on*

- Distributed Computing Systems*, ICDCSW '02, pp. 575--578, Washington, DC, USA. IEEE Computer Society.
- Leong, B.; Liskov, B. & Morris, R. (2007). Greedy virtual coordinates for geographic routing. In *Network Protocols, 2007. ICNP 2007. IEEE International Conference on*, pp. 71–80.
- Li, L. & Wei-jia, L. (2011). The analysis of data fusion energy consumption in wsn. In *System Science, Engineering Design and Manufacturing Informatization (ICSEM), 2011 International Conference on*, volume 1, pp. 310–313.
- Lin, S.; Liqin, W. & Zhengwei, Z. (2010). A clustering algorithm based geographic location information for wireless sensor networks. In *Proceedings of the 2010 International Conference on Electrical and Control Engineering*, ICECE '10, pp. 2588--2592, Washington, DC, USA. IEEE Computer Society.
- Loureiro, A. A. F.; Nogueira, M. S.; Nakamura, E. F. & Maur, C. (2003). Redes de sensores sem fio. *Brasileiro de Redes*, 21:179--226.
- Mahfoudh, S. & Minet, P. (2008). Survey of energy efficient strategies in wireless ad hoc and sensor networks. In *Networking, 2008. ICN 2008. Seventh International Conference on*, pp. 1–7.
- Maihofer, C. (2004). A survey of geocast routing protocols. *Communications Surveys Tutorials, IEEE*, 6.
- Mauve, M.; Widmer, J. & Hartenstein, H. (2001). A survey on position-based routing in mobile ad hoc networks. *Network, IEEE*, 15(6).
- Nakamura, E. F. (2007). *Information Fusion in Wireless Sensor Networks*. Tese de doutorado, Federal University of Minas Gerais, BH, MG, Brazil.
- Nati, M. (2008). *Geographic Routing: Mission Possible*. Tese de doutorado, University of Rome.
- Panigrahi, B.; De, S. & Sun Luk, J.-D. (2009). A greedy minimum energy consumption forwarding protocol for wireless sensor networks. In *Communication Systems and Networks and Workshops, 2009. COMSNETS 2009. First International*, pp. 1–6.
- Park, E.; Bae, D. & Choo, H. (2010). Energy efficient geographic routing for prolonging network lifetime in wireless sensor networks. In *Proceedings of the 2010 International Conference on Computational Science and Its Applications*, ICCSA '10, pp. 285--288, Washington, DC, USA. IEEE Computer Society.

- Pham, N.; Youn, J. & Won, C. (2006). A comparison of wireless sensor network routing protocols on an experimental testbed. In *Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006. IEEE International Conference on*, volume 2, pp. 276–281.
- Rao, A.; Ratnasamy, S.; Papadimitriou, C.; Shenker, S. & Stoica, I. (2003). Geographic routing without location information. In *MOBICOM'03*, pp. 96--108. ACM Press.
- Rappaport, T. (2002). *Wireless communications: principles and practice*. Prentice Hall communications engineering and emerging technologies series. Prentice Hall PTR.
- Santi, P. (2005). Topology control in wireless ad hoc and sensor networks. *ACM Comput. Surv.*, 37(2):164--194.
- Savvides, A.; Han, C.-C. & Strivastava, M. B. (2001). Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, MobiCom '01, pp. 166--179, New York, NY, USA. ACM.
- Seada, K.; Zuniga, M.; Helmy, A. & Krishnamachari, B. (2004). Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, SenSys '04, pp. 108--121, New York, NY, USA. ACM.
- Shin, J.; Ramachandran, U. & Ammar, M. (2007). On improving the reliability of packet delivery in dense wireless sensor networks. In *Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on*, pp. 718–723.
- Souza, E. L. d.; Nakamura, E. F. & Oliveira, H. A. B. F. d. (2010). Uma abordagem de fusao de dados em redes de sensores para reduzir o impacto de erros de localizacao em algoritmos de rastreamento. In *28o Simposio Brasileiro de Redes de Computadores e Sistemas Distribuidos (SBRC'2010)*, pp. 291–304.
- Takagi, H. & Kleinrock, L. (1984). Optimal transmission ranges for randomly distributed packet radio terminals. *Communications, IEEE Transactions on*, 32(3):246 – 257.
- Tan, G. & Kermarrec, A.-M. (2011). Greedy geographic routing in large-scale sensor networks: A minimum network decomposition approach. *Networking, IEEE/ACM Transactions on*, PP(99):1.

- Tollefson, J. (2010). A towering experiment an ambitious project to track greenhouse gases from a perch high above the amazon forest will provide crucial data - but only if scientists can get it built. *Nature*, 467(386).
- Tsai, M.-J.; Yang, H.-Y.; Liu, B.-H. & Huang, W.-Q. (2009). Virtual-coordinate-based delivery-guaranteed routing protocol in wireless sensor networks. *IEEE/ACM Trans. Netw.*, 17:1228--1241.
- Villalba, L. J.; Orozco, A. L.; Cabrera, A. & Barenco, C. J. (2009). Routing protocols in wireless sensor networks. *Sensors*, 9(11):8399--8421.
- Xing, G. (2006). *Unified power management in wireless sensor networks*. Tese de doutorado, St. Louis, MO, USA. AAI3238702.
- Xing, G.; Lu, C.; Pless, R. & Huang, Q. (2004). On greedy geographic routing algorithms in sensing-covered networks. In *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '04, pp. 31--42, New York, NY, USA. ACM.
- Yoneki, E. & Bacon, J. (2005). A survey of Wireless Sensor Network technologies: research trends and middleware's role. Relatório técnico UCAM-CL-TR-646, University of Cambridge, Computer Laboratory, 15 JJ Thomson Avenue, Cambridge CB3 0FD, United Kingdom, phone +44 1223 763500.
- You, J.; Lieckfeldt, D.; Han, Q.; Salzmann, J. & Timmermann, D. (2009). Look-ahead geographic routing for sensor networks. In *Pervasive Computing and Communications, 2009. PerCom 2009. IEEE International Conference on*, pp. 1 --6.
- Zhao, L.; Liu, G.; Chen, J. & Zhang, Z. (2009). Flooding and directed diffusion routing algorithm in wireless sensor networks. In *Hybrid Intelligent Systems, 2009. HIS '09. Ninth International Conference on*, volume 2, pp. 235 --239.
- Zheng, J. & Jamalipour, A. (2009). *Wireless Sensor Networks: A Networking Perspective*, volume 2008 of *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*. Wiley-IEEE Press.
- Zhu, Y.; Zhang, B.; Yu, F. & Ning, S. (2009). A rssi based localization algorithm using a mobile anchor node for wireless sensor networks. In *Computational Sciences and Optimization, 2009. CSO 2009. International Joint Conference on*, volume 1, pp. 123 --126.

- Zorzi, M. & Rao, R. (2003). Geographic random forwarding (geraf) for ad hoc and sensor networks: multihop performance. *Mobile Computing, IEEE Transactions on*, 2(4):337 – 348.
- Zorzi, M. & Rao, R. (2004). Energy-efficient forwarding for ad hoc and sensor networks in the presence of fading. In *Communications, 2004 IEEE International Conference on*, volume 7, pp. 3784 – 3789 Vol.7.