

MAX DO VAL. MACHADO

COMUNICAÇÃO DE DADOS BASEADA NO  
ESCALONAMENTO DE TRABALHO DE  
NÓS VIZINHOS EM  
REDES DE SENSORES SEM FIO

Tese apresentada ao Programa de Pós-  
-Graduação em Ciência da Computação do  
Instituto de Ciências Exatas da Universi-  
dade Federal de Minas Gerais como requi-  
sito parcial para a obtenção do grau de Dou-  
tor em Ciência da Computação.

ORIENTADOR: ANTONIO A. F. LOUREIRO (DCC/UFMG)  
CO-ORIENTADORA: RAQUEL A. F. MINI (DCC/PUC MINAS)

Belo Horizonte  
Dezembro de 2013

© 2013, Max do Val. Machado.  
Todos os direitos reservados.

Machado, Max do Val.

M149c      Comunicação de dados baseada no escalonamento de  
trabalho de nós vizinhos em redes de sensores sem fio /  
Max do Val. Machado. — Belo Horizonte, 2013  
xxi, 127 f. : il. ; 29cm

Tese (doutorado) — Universidade Federal de  
Minas Gerais. Departamento de Ciência da  
Computação.

Orientador: Antonio Alfredo Ferreira Loureiro.  
Coorientadora: Raquel Aparecida de Freitas Mini.

1. Computação - Teses. 2. Comunicação de dados -  
Teses. 3. Redes de sensores sem fio - Teses.  
I. Orientador. II. Coorientadora. III. Título.

CDU 519.6\*22(043)



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

### FOLHA DE APROVAÇÃO

Comunicação de dados baseada no escalonamento de trabalho de nós vizinhos  
em redes de sensores sem fio

**MAX DO VAL MACHADO**

Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. ANTONIO ALFREDO FERREIRA LOUREIRO - Orientador  
Departamento de Ciência da Computação – UFMG

PROFA. RAQUEL APARECIDA DE FREITAS MINI - COORIENTADORA  
Departamento de Ciência da Computação - PUC-MG

PROF. DANIEL FERNANDES MACEDO  
Departamento de Ciência da Computação - UFMG

PROFA. FÁTIMA DE LIMA PROCÓPIO D. FIGUEIREDO  
Departamento de Ciência da Computação - PUC-MG

PROF. JOSÉ MARCOS SILVA NOGUEIRA  
Departamento de Ciência da Computação - UFMG

PROF. LEANDRO APARECIDO VILLAS  
Instituto de Computação - UNICAMP

PROF. LUIZ HENRIQUE ANDRADE CORREIA  
Departamento de Ciência da Computação - UFLA

Belo Horizonte, 18 de dezembro de 2013.

*Aos meus pais Max e Fátima;  
minha esposa Ana Carolina;  
e, meu filho Cauã.*

# Agradecimentos

Em primeiro lugar, eu gostaria de agradecer aos professores que participaram da defesa da proposta de tese pelas críticas e sugestões. Ao José Marcos e ao Robson pelas cartas de recomendação para o doutorado. Aos colegas da UFMG pelas sugestões e frases de apoio. Aos colegas e amigos da PUC Minas, pelos incentivos, conversas e discussões durante a longa caminhada. Às diversas barreiras encontradas durante os anos de doutorado pelo amadurecimento profissional e pessoal.

Um agradecimento especial ao Loureiro e à Raquel. Muito obrigado pelos ensinamentos e discussões.

Aos amigos porque após anos de faltas ainda se lembram de mim: voltei! Valeu Estevan, Hugo, Jacaré, Nivaldão e Vinícius. Afinal, *Be honest!* Valeu Kotots, Gilbertão e Amigos da Jupenga. Ao Galo por me fazer esquecer da tese em diversos momentos.

Agradeço à minha família: avôs (ós), tios (as), primos (as), sogro (a), cunhados (as) e sobrinhos (as). Vovô Paulo, como eu queria que você estivesse fisicamente ao meu lado neste momento! Em especial, ao Tio Chico pelos conselhos e orientações desde o segundo grau. Ao Tom Vêi pela companhia durante meus períodos de isolamento no sítio para trabalhar na tese. À minha irmã Lela, ao meu irmão João Guilherme e a minha “mãe” Cacá por tudo que vocês representam em minha vida: eu amo vocês. Ao papai e à mamãe pelo amor incondicional e por serem a minha base. Pai e mãe, obrigado! Este título é para vocês.

À Carol por todo o amor, carinho, compreensão e paciência. Meu amor, obrigado pela nossa defesa. Ao Cauã, pela oportunidade de sentir amor incondicional por uma pessoa. Filho, obrigado por aceitar que, às vezes, o papai não podia assistir Backyardigans, Barney, Amigãozão, nem ver os patos, nem jogar futebol, nem nadar, e nem brincar com você. O papai te ama. A Deus pela minha defesa, amigos e família.

*“Todos os animais são iguais,  
mas alguns animais são mais iguais do que os outros.”*  
(George Orwell - A Revolução dos Bichos)

# Resumo

A melhor forma de economizar energia em Redes de Sensores Sem Fio (RSSFs) é fazer com que os nós sensores adormeçam sempre que possível, fazendo com que eles fiquem indisponíveis periodicamente. No entanto, essa indisponibilidade pode afetar tarefas de rede como a comunicação de dados. Protocolos para controle de acesso ao meio para essas redes tipicamente tratam o problema de escalonar a comunicação de dados quando o próximo nó estiver dormindo. Uma das abordagens usadas por esses protocolos (política pró-ativa) é compartilhar seu escalonamento de trabalho com seus vizinhos e cada nó emissor agenda sua transmissão para o momento em que o próximo nó estiver acordado. Esse processo de compartilhamento é tipicamente baseado em troca de pacotes, o que aumenta o consumo de energia da rede. Outra abordagem (política reativa) é descobrir quando o próximo nó estará acordado e, depois, enviar o pacote nesse momento. Apesar dessa abordagem eliminar o custo de compartilhamento, suas descobertas também efetuam transmissões adicionais. Dadas as limitações das duas abordagens, esta tese propõe três soluções para minimizar os custos de compartilhamento e descoberta. A primeira solução é uma abordagem híbrida entre as duas políticas existentes. Nesse caso, cada nó emissor insere seu escalonamento de trabalho em sua mensagem e cada receptor guarda essa informação para usá-la em transmissões futuras. A segunda solução é um projeto combinado entre a política reativa e o roteamento baseado no receptor. Essa abordagem de roteamento aumenta o número de candidatos a próximo nó, o que minimiza o custo de energia da descoberta reativa. Nesse projeto combinado, o nó emissor tem que descobrir o escalonamento de trabalho de algum vizinho em vez do escalonamento de um nó específico. A última solução é o protocolo *Gossiping Using the Energy Map* (GEM), um projeto integrado envolvendo as camadas de rede e MAC para efetuar difusão de dados quando os nós sensores ficam quase sempre dormindo (por exemplo, durante 99% de seus tempos de vida). Nesse caso, quando um nó emissor envia um pacote, nem todos seus vizinhos estarão acordados. A maioria das soluções propostas para difusão de dados em RSSFs desconsidera que os nós sensores podem ficar frequentemente dormindo. A ideia

principal do GEM é criar uma árvore de difusão baseada no mapa de energia da rede (informação sobre a energia restante em cada parte da rede) e fazer com que cada nó sensor compartilhe informações apenas com um nó vizinho que participa dessa árvore disseminando dados. Além disso, o GEM usa o mapa de energia para adaptar o funcionamento básico dos nós localizados dentro de regiões de baixa energia, prolongando o tempo de vida da rede. Resultados de simulação mostram que as soluções propostas nesta tese reduzem a quantidade de transmissões e consomem menos energia quando comparadas com outras soluções propostas na literatura e avaliadas neste trabalho.

# Abstract

The best way to save energy in Wireless Sensor Networks (WSNs) is to have sensor nodes go to sleep whenever possible, making them periodically unavailable. On the other hand, this unavailability may affect network tasks such as data communication. Media Access Control protocols for these networks typically deal with the problem of scheduling the data communication when the next hop is sleeping. One of the approaches used by those protocols (proactive policy) is to share each node's working schedule with its neighbors and have sender nodes to schedule their transmissions to their next hop awaking moment. This sharing is typically based on message exchanges, which increases the network energy consumption. Another approach (reactive policy) is to discover when the next hop is going to be awake, and, have a node to send a message at that moment. Although this approach eliminates the sharing cost, its discoveries introduce additional packet transmissions as well. Given the limitations of both policies, this thesis proposes three solutions to minimize the sharing and discovery costs. The first solution is a hybrid policy that combines both existing approaches. In this case, each sender node inserts its working schedule into its packet and each receiver node saves this information to use in future transmissions. The second solution is a combined design involving the reactive policy and the receiver-based routing. This routing technique increases the number of next hop candidates and can minimize the cost of the reactive discovery. In this combined design, a sender node has to discover the working schedule of some neighbor node instead of a specific neighbor schedule. Our last solution is the *Gossiping Using the Energy Map* (GEM) protocol, a cross-layer design solution involving the network and MAC layers to perform broadcasting when sensor nodes frequently sleep (e.g., during 99% of their lifetime). In this case, when a sender node sends a packet, not all neighbors will be awake. Most of the solutions proposed for broadcasting in WSNs ignore that sensor nodes can frequently sleep. The main idea of GEM is to create a broadcasting tree based on the network energy map (the information about the amount of energy available at each part of the network) and make each sensor node share information with only one neighbor node

that participate in this tree forwarding data. Furthermore, GEM uses the energy map to adapt the basic operation of the nodes localized inside low-energy regions, prolonging the network lifetime. Simulation results reveal that our solutions reduce the number of transmissions and energy consumption when compared with other solutions proposed in the literature and evaluated in this work.

# Lista de Figuras

1.1	Exemplo de nós sensores . . . . .	2
1.2	Exemplo do ciclo operacional de três nós sensores . . . . .	2
2.1	Formas de comunicação de dados em redes de sensores sem fio . . . . .	12
2.2	Ação correspondente ao evento atualizar informação sobre vizinhos da política pró-ativa . . . . .	14
2.3	Ação correspondente ao evento enviar quadro da política pró-ativa . . . . .	15
2.4	Ação correspondente ao evento receber quadro da política pró-ativa . . . . .	15
2.5	Ação correspondente ao evento enviar quadro da política reativa . . . . .	16
2.6	Ação correspondente ao evento receber quadro da política reativa . . . . .	17
2.7	Funcionamento básico do roteamento baseado no emissor . . . . .	18
2.8	Funcionamento básico do roteamento baseado no receptor . . . . .	18
2.9	A classificação do termo <i>Gossip</i> em redes de sensores sem fio . . . . .	20
2.10	Métricas de decisão para os protocolos de roteamento da família <i>gossip</i> em redes de sensores sem fio . . . . .	22
4.1	Ação correspondente ao evento enviar quadro da política híbrida . . . . .	46
4.2	Ação correspondente ao evento descoberta de escalonamento da política híbrida . . . . .	47
4.3	Ação correspondente ao evento receber quadro da política híbrida . . . . .	48
4.4	Exemplo do ciclo operacional de dois nós sensores . . . . .	49
4.5	Protocolos usados para avaliar o efeito da política híbrida em um protocolo MAC pró-ativo . . . . .	53
4.6	Protocolos usados para avaliar o efeito da política híbrida em um protocolo MAC reativo . . . . .	54
4.7	Regiões em que os destinos estão localizados . . . . .	54
4.8	Número esperado de disseminações e ciclo de trabalho esperado para o segundo cenário de simulação durante o tempo de vida da rede . . . . .	55

4.9	Métricas avaliadas para os protocolos híbrido e pró-ativo com ciclo estático de 1%	56
4.10	Métricas avaliadas para os protocolos híbrido e pró-ativo com ciclo estático de 80%	59
4.11	Métricas avaliadas para os protocolos híbrido e pró-ativo com ciclo dinâmico	60
4.12	Métricas avaliadas para os protocolos híbrido e reativo com ciclo estático de 1%	61
4.13	Métricas avaliadas para os protocolos híbrido e reativo com ciclo estático de 80%	62
4.14	Métricas avaliadas para os protocolos híbrido e reativo com ciclo dinâmico	63
5.1	Exemplo da abordagem de roteamento baseada no emissor	66
5.2	Exemplo da abordagem de roteamento baseada no receptor	67
5.3	Funcionamento básico do RbMAC	70
5.4	Regiões nas quais os nós destinos estão localizados	76
5.5	Projetos avaliados para a comunicação de dados	77
5.6	Resultados teóricos e de simulação do projeto combinado RbMAC-TEDD para o número de transmissões por nó sensor	79
5.7	Desempenho do projeto combinado RbMAC-TEDD com 1, 4, 8 e 12 candidatos a próximo nó no cenário com ciclo de trabalho reduzido	81
5.8	Desempenho do projeto combinado RbMAC-TEDD com 1, 4, 8 e 12 candidatos a próximo nó no cenário com ciclo de trabalho elevado	82
5.9	Métricas avaliadas para os projetos de comunicação de dados no cenário com ciclo de trabalho reduzido	84
5.10	Métricas avaliadas para os projetos de comunicação de dados no cenário com ciclo de trabalho elevado	85
6.1	Exemplo de uma árvore de difusão do GEM	91
6.2	Exemplo intermediário da criação da árvore do GEM	92
6.3	Exemplo de criação da árvore do GEM	93
6.4	Ação receber pacote do GEM	94
6.5	Ação receber pacote de dados do GEM	95
6.6	Ação receber pacote de criação do GEM	96
6.7	Ação receber pacote de filiação do GEM	96
6.8	Ação receber pacote de reconhecimento do GEM	96
6.9	Ação de filiação do GEM	97
6.10	Ação de decisão do GEM	98

6.11 Exemplo de mapas de energia contendo regiões de baixa energia . . . . .	99
6.12 Exemplo de como o GEM calcula a energia de corte . . . . .	100
6.13 Métricas avaliadas no cenário sem regiões de baixa energia . . . . .	103
6.14 Métricas avaliadas para os nós sensores localizados dentro de trinta e duas regiões de baixa energia . . . . .	106
6.15 Métricas avaliadas para os nós sensores localizados dentro de uma região de baixa energia . . . . .	107

# Lista de Tabelas

3.1	Classificação de alguns protocolos MAC reativos e pró-ativos . . . . .	25
3.2	Classificação de alguns protocolos de roteamento baseados no emissor / receptor e em pertencentes ou não à família <i>gossip</i> . . . . .	33
5.1	Latência de entrega e consumo médio de energia por nó sensor . . . . .	87

# Lista de Algoritmos

1	Início do ciclo de trabalho dinâmico . . . . .	86
2	Término do ciclo de trabalho dinâmico . . . . .	86

# Glossário

**ACK** *Acknowledgement*. 69, 72

**AGSP** *Adaptive Gossip-based Sleep Protocol*. 29

**AMAC** *Asynchronous Medium Access Control*. 25, 28, 52, 60, 63, 76, 81–83

**BAMAC** *Battery Aware Medium Access Control*. 25, 30, 31

**BPS** *Broadcast Protocol for Sensor Networks*. 33, 37, 38

**CSMA** *Carrier Sense Multiple Access*. 26, 27, 29, 70, 71

**CSMA-MPS** *Carrier Sense Multiple Access Minimum Preamble Sampling*. 25, 30, 32

**CSMA/CA** *Carrier Sense Multiple Access with Collision Avoidance*. 26, 27

**CTS** *Clear to Send*. 27, 28, 30–32, 53, 57–59, 61, 62, 69, 72

**DCF** *Distributed Coordinated Function*. 29, 30

**DDB** *Dynamic Delayed Broadcasting*. 22, 33, 37–39, 41

**DPRD** *Distributed Passive Routing Decisions*. 25, 28

**GEM** *Gossiping using the Energy Map*. 5, 6, 8, 21, 22, 89–94, 98–108, 110–113

**GeRaF** *Geographic Random Forwarding*. 25, 28

**GSP** *Gossip-based Sleep Protocol*. 25, 28, 29

**HESMAC** *High Efficient Sensor Medium Access Control*. 25, 28

**IGF** *Implicit Geographic Forwarding*. 25, 28

**INPE** Instituto Nacional de Pesquisa Espacial. 10

**IP** *Internet Protocol*. 12, 68

**L<sup>2</sup>** *Lazy Forwarding in Low Duty Cycle Wireless Sensor Networks*. 25, 30, 32

**MAC** *Medium Access Control*. 2–6, 8, 11, 13, 15, 16, 19, 20, 23, 24, 26–28, 30–32, 37, 42, 44–47, 50, 52–54, 64–68, 70, 72, 74–77, 81, 87, 90, 102, 109–112

**MACAM** *Multiple Access Collision Avoidance protocol for Multicast service*. 25, 30–32

**MANET** *Mobile Ad Hoc Network*. 31, 39

**NAT** *Network Address Translation*. 12

**OF** *Opportunistic Flooding*. 35, 37, 101–104, 106, 107

**OMAC** *Opportunistic Medium Access Control*. 25, 30, 31, 52, 55, 57, 76, 77, 81–83

**RbMAC** *Receiver-based Medium Access Control*. 6, 25, 65, 68–78, 80–88, 110

**RI-MAC** *Receiver-Initiated MAC*. 25, 30

**RSSF** *Rede de Sensor Sem Fio*. 1, 2, 5, 9–11, 13, 17, 19–21, 23, 24, 26–29, 32, 34–39, 43, 44, 50, 52, 53, 60, 63, 65–68, 72, 75–77, 84, 87, 89, 90, 98, 100, 101, 104, 108, 109, 111

**RTS** *Request to Send*. 27, 28, 30–32, 53, 57–59, 61, 62, 69, 72

**SeMAC** *Sequential Medium Access Control*. 25, 30, 31

**SMAC** *Sensor-MAC*. 25, 30, 31

**SPIN** *Sensor Protocols for Information via Negotiation*. 33, 37, 38

**STEM** *Sparse Topology and Energy Management*. 25, 28, 29, 32

**TBF** *Trajectory Based Forwarding*. 33–35, 76, 77, 81–83

**TCP** *Transport Control Protocol*. 12, 67

**TEDD** *Trajectory and Energy-based Data Dissemination*. 21, 22, 33, 39, 41, 42, 76–78, 81–84, 86, 87

**TGSP** *Traffic-aware Gossip-based Sleep Protocol.* 29

**UDP** *User Datagram Protocol.* 12, 67

**WLAN** *Wireless Local Area Network.* 29

**XLM** *Unified Cross-layer Module.* 25, 28

**ZRP** *Zone Routing Protocol.* 40

# Sumário

Agradecimentos	v
Resumo	vii
Abstract	ix
Lista de Figuras	xi
Lista de Tabelas	xiv
Lista de Algoritmos	xv
Glossário	xvi
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	1
1.2 Objetivos e Contribuições . . . . .	3
1.3 Publicações . . . . .	6
1.4 Organização do Trabalho . . . . .	8
<b>2 Fundamentos</b>	<b>9</b>
2.1 Redes de Sensores Sem Fio . . . . .	9
2.2 Comunicação de Dados em Redes de Sensores Sem Fio . . . . .	11
2.2.1 Formas de Comunicação de Dados . . . . .	11
2.2.2 Projetos Combinado e Integrado entre Camadas . . . . .	11
2.2.3 Políticas MAC para Determinar o Escalonamento de Trabalho dos Nós Vizinhos . . . . .	13
2.2.4 Políticas de Roteamento para Escolha do Próximo Nó . . . . .	17
2.2.5 Escolha do Próximo Nó a partir do Escalonamento de Trabalho	19
2.2.6 Família de Protocolos <i>Gossip</i> . . . . .	20

2.3	Conclusões . . . . .	22
<b>3</b>	<b>Trabalhos Relacionados</b>	<b>24</b>
3.1	Protocolos para a Camada MAC . . . . .	24
3.1.1	Introdução . . . . .	24
3.1.2	Protocolos MAC Reativos . . . . .	28
3.1.3	Protocolos MAC Pró-ativos . . . . .	30
3.2	Protocolos de Roteamento . . . . .	32
3.2.1	Protocolos baseados no Emissor e Não <i>Gossip</i> . . . . .	32
3.2.2	Protocolos baseados no Emissor e <i>Gossip</i> . . . . .	35
3.2.3	Protocolos baseados no Receptor e Não <i>Gossip</i> . . . . .	37
3.2.4	Protocolos baseados no Receptor <i>Gossip</i> . . . . .	39
3.3	Conclusões . . . . .	42
<b>4</b>	<b>Política Híbrida para Determinar o Escalonamento de Trabalho dos Nós Vizinhos</b>	<b>43</b>
4.1	Introdução . . . . .	43
4.2	Política Híbrida . . . . .	45
4.2.1	Funcionamento Básico . . . . .	45
4.2.2	Parâmetros sobre Escalonamento de Trabalho Inseridos nos Quadros . . . . .	48
4.2.3	Retransmissão dos Quadros de Descoberta . . . . .	50
4.3	Resultados de Simulação . . . . .	52
4.3.1	Parâmetros de Simulação . . . . .	52
4.3.2	Política Híbrida com Protocolo MAC Pró-ativo . . . . .	55
4.3.3	Política Híbrida com Protocolo MAC Reativo . . . . .	60
4.4	Conclusões . . . . .	63
<b>5</b>	<b>Projeto Combinado entre a Política MAC Reativa e o Roteamento baseado no Receptor</b>	<b>65</b>
5.1	Introdução . . . . .	65
5.2	Protocolo Receiver-based MAC . . . . .	69
5.2.1	Funcionamento Básico . . . . .	69
5.2.2	Mecanismo de Descoberta . . . . .	72
5.2.3	Verificação Formal . . . . .	74
5.3	Resultados de Simulação . . . . .	75
5.3.1	Parâmetros de Simulação . . . . .	75
5.3.2	Número de Candidatos na Descoberta Reativa . . . . .	78

5.3.3	Projetos para Comunicação de Dados . . . . .	81
5.4	Projeto Integrado para Redução da Latência e do Consumo de Energia	84
5.4.1	Funcionamento Básico . . . . .	86
5.4.2	Resultados de Simulação . . . . .	86
5.5	Conclusões . . . . .	87
<b>6</b>	<b><i>Gossiping Using the Energy Map</i></b>	<b>89</b>
6.1	Introdução . . . . .	89
6.2	Protocolo <i>Gossiping using the Energy Map</i> . . . . .	90
6.2.1	Funcionamento Básico . . . . .	91
6.2.2	Eventos e Ações . . . . .	93
6.2.3	Regiões de Baixa Energia . . . . .	98
6.2.4	Probabilidade de Decisão . . . . .	100
6.3	Resultados de Simulação . . . . .	101
6.3.1	Parâmetros de Simulação . . . . .	101
6.3.2	Difusão de Dados sem Regiões de Baixa Energia . . . . .	103
6.3.3	Difusão de Dados com Regiões de Baixa Energia . . . . .	105
6.4	Conclusões . . . . .	108
<b>7</b>	<b>Conclusões e Trabalhos Futuros</b>	<b>109</b>
7.1	Conclusões . . . . .	109
7.2	Trabalhos Futuros . . . . .	111
	<b>Referências Bibliográficas</b>	<b>114</b>

# Capítulo 1

## Introdução

*Tudo o que existe na física é impossível ou trivial.  
É impossível até que o compreendamos, e então se torna trivial.  
Ernest Rutherford*

Este capítulo apresenta a motivação, objetivos, contribuições, publicações e a organização desta tese intitulada de “Comunicação de Dados baseada no Escalonamento de Trabalho de Nós Vizinhos em Redes de Sensores Sem Fio”. Esses tópicos são abordados nas seções 1.1 até 1.4, respectivamente.

### 1.1 Motivação

As Redes de Sensores Sem Fio (RSSFs) são compostas tipicamente por nós sensores de baixo custo (figura 1.1) que trabalham de forma cooperativa para monitorar condições físicas ou ambientais [4, 17, 130]. Por exemplo, eles podem monitorar condições ambientais em florestas, oceanos e planetas [85]. O principal desafio para transformar as RSSFs na tecnologia de sensoriamento do futuro é projetar soluções eficientes em termos de energia. A energia é o recurso mais escasso em RSSFs, pois a bateria dos nós sensores apresenta capacidade finita e sua recarga nem sempre é possível. A melhor forma de economizar energia nessas redes é fazer com que os nós sensores adormeçam sempre que possível [56, 139].

A maioria das soluções para RSSFs considera algum mecanismo para fazer com que os nós adormeçam periodicamente. Esse mecanismo divide o tempo de vida dos nós em ciclos operacionais de tal forma que cada ciclo é composto por um período no qual o nó permanece acordado seguido por outro em que ele dorme. A figura 1.2 ilustra o ciclo operacional de três nós sensores A, B e C em que o período acordado (cor branca) equivale a 10%, 30% e 70% dos respectivos ciclos operacionais. O período

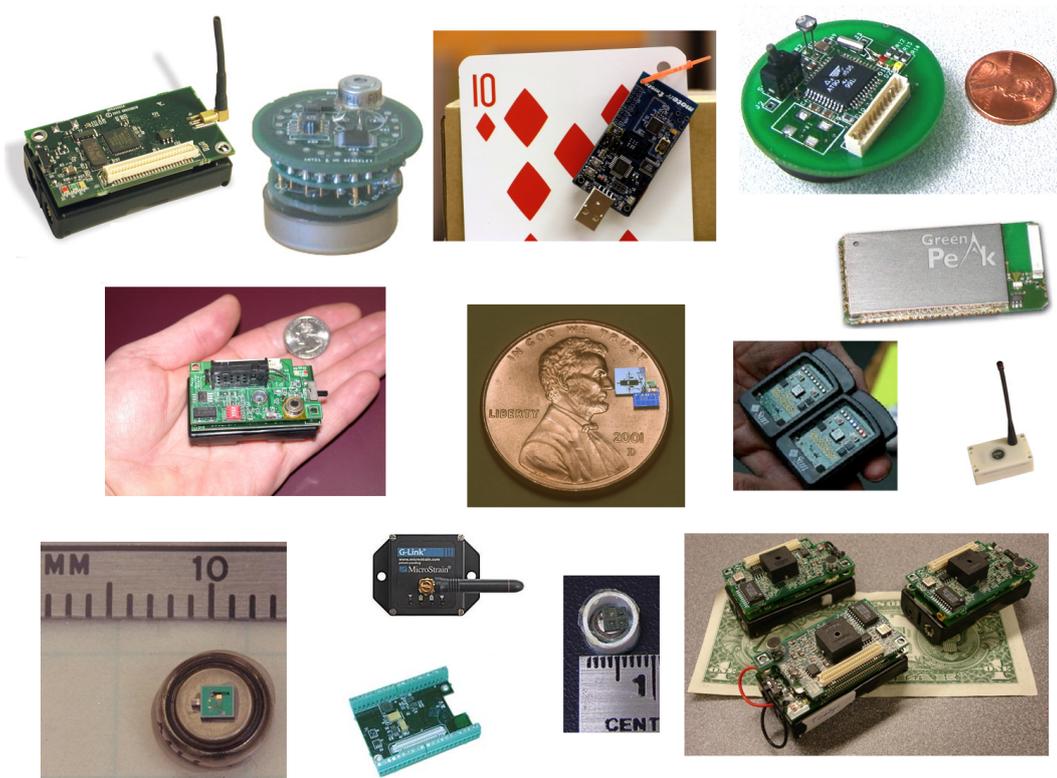


Figura 1.1: Exemplo de nós sensores

em que o nó dorme é ilustrado pela cor cinza. A porcentagem de tempo em que um nó permanece acordado em um ciclo operacional é chamada de ciclo de trabalho (*duty cycle*) [23, 29, 40, 51]. O projetista da rede define esse valor a partir de parâmetros como, por exemplo, o tráfego de dados, densidade da rede e tempo de vida esperado. A agenda de quando um nó estará acordado ou dormindo é chamada de escalonamento de trabalho. Dado que os nós sensores devem operar com um ciclo de trabalho reduzido para economizar energia, a indisponibilidade periódica dos nós adormecidos pode afetar as tarefas da rede [52, 127].

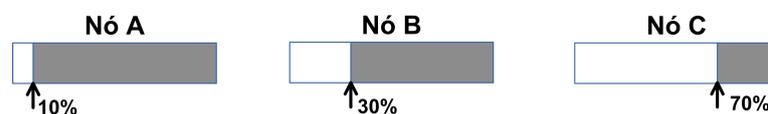


Figura 1.2: Exemplo do ciclo operacional de três nós sensores

A comunicação de dados é uma das tarefas afetadas por essa indisponibilidade. Quando um nó envia uma mensagem, a comunicação pode falhar se o próximo nó estiver dormindo. A camada de Controle de Acesso ao Meio (MAC, *Medium Access Control*) das RSSFs é responsável por tratar desse problema. Para isso, ela controla

o escalonamento de trabalho dos nós sensores. Essa informação é fundamental para que cada nó emissor efetue sua transmissão quando supostamente o próximo nó estiver acordado.

As políticas MAC para escalonamento de trabalho de nós vizinhos existentes na literatura [67, 75, 83, 87] podem ser divididas em pró-ativa e reativa. A primeira faz com que cada nó sensor compartilhe seu escalonamento de trabalho com todos seus vizinhos, permitindo que cada nó emissor agende sua transmissão para o momento em que o próximo nó estiver acordado. Contudo, esse compartilhamento é tipicamente baseado em troca de mensagens, o que aumenta o consumo de energia da rede. A segunda faz com que cada nó sensor descubra o escalonamento do próximo nó somente quando essa informação for necessária. Apesar dessa abordagem eliminar o custo de compartilhamento, sua descoberta também efetua transmissões adicionais. A escolha da política a ser usada em um protocolo MAC deve considerar a relação custo-benefício entre manter a informação sobre o escalonamento de trabalho dos vizinhos atualizada, obtê-la quando necessária e a frequência com que ela será utilizada. Dadas as limitações das duas políticas, novas soluções devem ser investigadas.

## 1.2 Objetivos e Contribuições

Este trabalho propõe e avalia soluções para a comunicação de dados baseadas no custo-benefício da informação sobre o escalonamento de trabalho de nós vizinhos em redes de sensores sem fio. Suas principais contribuições consistem em soluções relacionadas à obtenção do escalonamento de trabalho dos nós vizinhos, em especial, para cenários nos quais os nós sensores operam com um ciclo de trabalho reduzido. As principais contribuições desta tese são:

- **Política Híbrida para Determinar o Escalonamento de Trabalho dos Nós Vizinhos** – Esta solução combina as políticas MAC pró-ativa e reativa. A ideia dessa política é que cada nó emissor insira seu escalonamento de trabalho em seu quadro e cada receptor use essa informação para criar e atualizar uma tabela contendo o escalonamento de trabalho de seus vizinhos. Diferente da política pró-ativa, os nós sensores obtêm a informação sobre o escalonamento de trabalho dos vizinhos sem qualquer troca extra de mensagens, o que elimina o compartilhamento pró-ativo. Quando um nó envia um quadro, ele verifica se conhece o escalonamento de trabalho próximo nó. Em caso afirmativo, como na política pró-ativa, ele agenda sua transmissão para o momento em que o próximo nó estiver acordado. Caso contrário, ele inicia a descoberta do escalonamento

de trabalho do próximo nó. Quando o nó emissor descobre essa informação, ele atualiza sua tabela e efetua sua transmissão. Ao contrário da política reativa, as transmissões futuras usam tal informação para evitar o custo reativo de novas descobertas. Todavia, uma desvantagem da solução híbrida é que cada nó emissor insere um campo extra no cabeçalho dos quadros. Resultados de simulação revelam o efeito da política híbrida em protocolos MAC pró-ativo e reativo. A solução proposta reduz o número de transmissões e o consumo de energia das políticas existentes na maioria dos cenários avaliados. Em particular, esses resultados são mais expressivos nos cenários em que a informação sobre os vizinhos desatualiza durante o tempo de vida da rede ou os nós sensores operam com um ciclo de trabalho reduzido (1% ou menos) ou dinâmico. O capítulo 4 apresenta e analisa os detalhes da política híbrida.

- **Projeto Combinado entre a Política MAC Reativa e o Roteamento baseado no Receptor** – O roteamento baseado no receptor faz com que quando um nó recebe um pacote, esse nó decida localmente se deve ou não propagar o pacote. Ao contrário do roteamento baseado no emissor no qual o nó anterior escolhe quem será o próximo nó e insere essa informação no cabeçalho do pacote. Assim, o roteamento baseado no receptor aumenta o número de candidatos a próximo nó quando comparado com a abordagem baseada no emissor. A ideia do projeto combinado proposto neste trabalho é usar a diversidade de candidatos a próximo nó do roteamento baseado no receptor para reduzir o custo da descoberta reativa. Um nó emissor faz menos transmissões de descoberta para encontrar algum vizinho acordado que para encontrar um vizinho específico. Os protocolos MAC reativos propostos na literatura são combinados com o roteamento baseado no emissor. Resultados teóricos e de simulação revelam que o projeto combinado envolvendo a política reativa e o roteamento baseado no receptor reduz o número de transmissões e a latência na comunicação de dados quando comparado com outras soluções avaliadas neste trabalho. O capítulo 5 apresenta e analisa os detalhes do projeto combinado entre a política MAC reativa e o roteamento baseado no receptor. Além disso, esse capítulo também propõe um protocolo reativo MAC para o projeto proposto, faz a verificação formal de algumas propriedades desse protocolo e apresenta uma técnica de ciclo de trabalho dinâmica para o mesmo.
- ***Gossiping Using the Energy Map*** – A difusão de dados (*broadcasting*) é uma tarefa de comunicação fundamental em redes de sensores sem fio. Ela acontece quando o nó monitor deseja enviar uma informação para todos os nós sensores.

Essa tarefa é desafiadora em redes com ciclo de trabalho reduzido (1% ou menos) porque quando um nó envia um pacote, nem todos seus vizinhos estão acordados para recebê-lo. Eventualmente, todos os vizinhos de um nó emissor podem estar adormecidos quando ele transmitir. A maioria das soluções propostas na literatura para difusão de dados em RSSFs são protocolos de roteamento que ignoram o fato dos nós ficarem quase sempre adormecidos como, por exemplo, os protocolos propostos em [34, 55]. Outras soluções (e.g., [44, 124]) consideram um projeto integrado [116] entre as camadas de rede e MAC em que cada nó emissor efetua sua difusão através de várias transmissões *unicasting*, uma para cada vizinho quando esse estiver acordado. Essas soluções são baseadas em uma política MAC pró-ativa em que cada nó sensor compartilha seu escalonamento de trabalho com todos seus vizinhos para permitir que cada nó emissor saiba quando cada um de seus vizinhos estará acordado. O *Gossiping Using the Energy Map* (GEM) cria uma árvore de difusão baseada na energia residual da rede e faz com que cada nó sensor compartilhe informações com apenas um nó vizinho que pertence à árvore e é responsável por propagar pacotes. Resultados de simulação mostram que o GEM reduz o número de transmissões e o consumo de energia quando comparado com outras soluções propostas na literatura e avaliadas neste trabalho. Além disso, o GEM aborda outro desafio que é a difusão de dados quando a rede contém regiões de baixa energia. O capítulo 6 apresenta e analisa em detalhes o GEM.

As duas primeiras contribuições podem ser consideradas em projetos de comunicação de dados envolvendo a disseminação ou a coleta de dados. A disseminação de dados acontece quando o nó monitor envia dados para um ou mais nós sensores na rede e a coleta acontece quando os nós sensores enviam dados para o nó monitor. A disseminação de dados é crucial em redes de sensores sem fio porque o nó monitor realiza tarefas específicas tais como alterar o modo de funcionamento de nós sensores, ativar e desativar um ou mais nós e enviar requisições ou interesses para eles. A coleta de dados é a principal atividade de comunicação nessas redes e acontece quando os nós sensores enviam dados de sensoriamento ou de controle para o nó monitor. A terceira contribuição é específica para a difusão de dados em RSSFs com ciclo de trabalho reduzido.

## 1.3 Publicações

Durante o doutorado, o aluno Max do Val Machado publicou 13 trabalhos: 2 em revistas internacionais, 3 em conferências internacionais, 1 em revista nacional e 7 em conferências nacionais. Ele recebeu duas menções honrosas, uma no XXIV Simpósio Brasileiro de Redes de Computadores (SBRC'06) e outra no XXVIII Simpósio Brasileiro de Redes de Computadores (SBRC'10). As publicações foram resultados do RbMAC, Política Híbrida, GEM e de outros trabalhos. Elas estão listadas abaixo:

- Publicação sobre a Política Híbrida:
  - MACHADO, Max do Val; MINI, Raquel A. F.; LOUREIRO, Antonio A. F.. *Política Híbrida para Determinar o Momento Acordado do Próximo Nó na Comunicação em Redes de Sensores Sem Fio. XXX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC'12), Ouro Preto-MG, 2012*
- Publicações sobre o Projeto Combinado:
  - MACHADO, Max do Val; MINI, Raquel A. F.; LOUREIRO, Antonio A. F.. *Comunicação de Dados baseada no Receptor para Redes de Sensores Sem Fio. Revista Brasileira de Redes de Computadores e Sistemas Distribuídos, v. 3, p. 21-31, 2010*
  - MACHADO, Max do Val; Mini, Raquel A.F.; LOUREIRO, Antonio A. F.. *Comunicação de Dados Baseada no Receptor para Redes de Sensores Sem Fio. XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC'10), p. 61-74, Gramado-RS, 2010*
  - MACHADO, Max do Val; MINI, Raquel A. F.; LOUREIRO, Antonio A. F.. *A Combined Approach for Receiver-based MAC and Network Layers in Wireless Sensor Networks. XLIV IEEE International Conference on Communications (ICC'09), p. 1-6, Dresden, Germany, 2009*
- Publicações sobre o GEM:
  - MACHADO, Max do Val; MINI, Raquel A. F.; LOUREIRO, Antonio A. F.; GUIDONI, Daniel Ludovico; MELO, Pedro O. S. Vaz de. *Gossiping using the Energy Map in Wireless Sensor Networks. X ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM'07), Chania, Greece, 2007*

- GUIDONI, Daniel Ludovico; MACHADO, Max do Val; MINI, Raquel A. F.; LOUREIRO, Antonio A. F.. *Difusão de Dados Baseada em Atraso e Energia para Redes de Sensores Sem Fio. XXIV Simpósio Brasileiro de Redes de Computadores (SBRC'06), Curitiba-PR, 2006*
- Publicações sobre outros trabalhos:
  - MELO, Pedro O. S. Vaz de; LOUREIRO, Antonio A. F.; MACHADO, Max do Val; MINI, Raquel A. F.. *Um Protocolo Baseado em Teoria dos Jogos para a Cooperação entre Diferentes Redes de Sensores Sem Fio. XXVII Simpósio Brasileiro de Redes Computadores (SBRC'09), p. 1–14, Recife-PE, 2009*
  - MELO, Pedro O. S. Vaz de; MINI, Raquel A. F.; MACHADO, Max do Val; LOUREIRO, Antonio A. F.. *Teoria dos Jogos na Comunicação de Dados em Redes de Sensores Sem Fio. XXV Simpósio Brasileiro de Redes de Computadores (SBRC'07), Belém-PA, 2007*
  - MACHADO, Max do Val; GOUSSEVSKAIA, O.; Mini, R.A.F.; Rezende, C.G.; Loureiro, A.A.F.; Mateus, G.R.; Nogueira, J.M.S. *Data Dissemination in Autonomic Wireless Sensor Networks. IEEE Journal on Selected Areas in Communications, v. 23, p. 2305–2319, 2005*
  - GOUSSEVSKAIA, Olga; MACHADO, Max do Val; MINI, Raquel A. F.; LOUREIRO, Antonio A. F.; MATEUS, Geraldo Robson; NOGUEIRA, José Marcos Silva. *Data Dissemination based on the Energy Map. IEEE Communications Magazine, v. 43, n. 7, p. 134–143, 2005*
  - MACHADO, Max do Val; GOUSSEVSKAIA, Olga; MINI, Raquel A. F.; LOUREIRO, Antonio A. F.; MATEUS, Geraldo Robson; NOGUEIRA, José Marcos Silva. *Data Dissemination Using the Energy Map. II IEEE Wireless On demand Network Systems and Services (WONS'05), p. 139–148, St. Moritz, Switzerland, 2005*
  - GOUSSEVSKAIA, Olga; MACHADO, Max do Val; MINI, Raquel A. F.; LOUREIRO, Antonio A. F.; MATEUS, Geraldo Robson; NOGUEIRA, José Marcos Silva. *Avoiding Low-Energy Areas in Wireless Sensor Network Data Dissemination. XXIII Simpósio Brasileiro de Redes de Computadores (SBRC'05), p. 565–578, Fortaleza-CE, 2005*
  - CHAVES, R. D.; SILVA, J.E.C.; MACHADO, Max do Val; MEDEIROS, T.H.. *Aplicação de Modelos Neurais para Predição do Mapa de Energia de*

*uma Rede de Sensores Sem Fio. Encontro Nacional de Inteligência Artificial (ENIA '05), São Leopoldo-RS, 2005*

Além disso, os resultados desta tese também estão submetidos em:

- *MACHADO, Max do Val; Mini, Raquel A.F.; LOUREIRO, Antonio A. F.. Hybrid Policy to Determine the Working Schedule of Neighbor Nodes in Data Communication for Wireless Sensor Networks. Computer Networks, 2014*
- *MACHADO, Max do Val; Mini, Raquel A.F.; LOUREIRO, Antonio A. F.. Difusão de Dados em Redes de Sensores Sem Fio com Ciclo de Trabalho Reduzido. XXXII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC'14), 2014*

## 1.4 Organização do Trabalho

O restante desta tese está organizada da seguinte forma. Os capítulos 2 e 3 apresentam os fundamentos e os trabalhos relacionados desta tese. Os capítulos 4, 5 e 6 propõem e avaliam as três contribuições deste trabalho, respectivamente, a política híbrida para determinar o escalonamento de trabalho dos nós vizinhos, o projeto combinado entre a política MAC reativa e o roteamento baseado no receptor e o protocolo GEM. O capítulo 7 descreve as conclusões e as direções futuras desta tese.

# Capítulo 2

## Fundamentos

*Poderosos rios podem ser facilmente dominados na sua nascente.*

*Públio Siro*

Este capítulo apresenta os fundamentos desta tese e está organizado como descrito a seguir. A seção 2.1 apresenta um breve resumo sobre redes de sensores sem fio. A seção 2.2 aborda a comunicação de dados nessas redes e mostra algumas definições e classificações utilizadas nesta tese. A seção 2.3 conclui este capítulo.

### 2.1 Redes de Sensores Sem Fio

Desde os tempos remotos, o homem se preocupa em acompanhar as oscilações e situações de diversos fenômenos nos mais variados ambientes. A humanidade sempre tentou prever o futuro com a maior antecedência possível para facilitar as tomadas de decisão. Por exemplo, o homem criou dispositivos como os de monitoração de precipitação pluviométrica, temperatura e umidade para aprimorar os trabalhos na agricultura. A evolução da eletrônica, computação e comunicação de dados sem fio tem permitido a criação de pequenos dispositivos capazes de monitorar fenômenos e ambientes. A interconexão desses dispositivos através de comunicação sem fio é denominada de Redes de Sensores Sem Fio (RSSFs) [4, 17, 9, 37, 63, 71, 130]. Nessas redes, os nós sensores coletam dados do ambiente onde estão inseridos, processam e propagam esses dados para um nó monitor. Esse nó normalmente não tem restrições de recursos e é responsável por coletar dados de sensoriamento da rede e enviar dados de controle para os nós sensores. As RSSFs vão revolucionar a forma como o homem coleta e processa dados.

As RSSFs podem ser aplicadas no monitoramento, rastreamento, coordenação e processamento de diversas aplicações. Na área ambiental, as redes de sensores serão importantes para a prevenção e o salvamento de vítimas de catástrofes naturais. Por

exemplo, terremotos, vulcões, tsunamis, furações, secas, queimadas ou enchentes. Além disso, elas poderão melhorar a previsão do tempo. Segundo o Instituto Nacional de Pesquisa Espacial (INPE), existe uma média de 243.000 focos de incêndio no período de seca no Brasil (de maio a setembro). Esses focos são uma das maiores ameaças para as florestas, plantações, parques e áreas de proteção ambiental. Sistemas de detecção de incêndio podem ajudar na redução dos estragos causados por queimadas. Na área médica, as RSSFs podem ser utilizadas para monitorar o funcionamento de órgãos como o coração, detectar a presença de substâncias que indicam a existência ou o surgimento de um problema biológico. Na área industrial, elas podem prover mecanismos de controle industrial. Por exemplo, micro sensores sem fio podem ser embutidos em “peças” com o objetivo de testar o processo de manufatura das linhas de montagem. A produção industrial será otimizada a partir do monitoramento de indústrias petroquímicas, fábricas, refinarias e siderúrgicas. Além disso, as redes de sensores sem fio serão capazes de garantir o controle de dados em áreas de difícil acesso ou perigosas. Por exemplo, na indústria de petróleo e gás. Na área urbana, as redes de sensores podem melhorar as condições de tráfego e de segurança. Além disso, as RSSFs podem monitorar locais internos como prédios e residências.

Como consequência da gama de aplicações, as RSSFs vêm recebendo atenção especial por parte dos pesquisadores e seus desafios estão sendo estudados. Ao contrário das redes *ad hoc* convencionais, as RSSFs possuem potencialmente centenas de milhares de nós [113]. Os nós sensores devem operar em ambientes inóspitos e as redes devem ser densas para alcançarem uma boa resolução de sensoriamento. Assim, a escalabilidade em relação ao número de nós é um fator essencial nas RSSFs. Diferente dos nós de uma rede *ad hoc* convencional, os nós sensores geralmente permanecem parados no local de sua deposição. Entretanto, as RSSFs possuem uma topologia dinâmica porque os nós sensores adormecem periodicamente para economizar energia. Além disso, os nós podem ficar fora de serviço quando a energia de suas baterias acabar ou um evento destrutivo acontecer. Outra característica que distingue essas redes das demais é que os nós sensores possuem restrição de recursos, tais como capacidade computacional, de memória e de energia. Os nós sensores são equipados com bateria e, em algumas aplicações, eles serão colocados em áreas remotas, o que impossibilita a manutenção dos mesmos. Nesse cenário, o tempo de vida da rede depende da quantidade de energia disponível nos nós sensores. Por isso, o projetista da rede deve balancear os recursos dos nós sensores com o objetivo de aumentar o tempo de vida da rede [7, 17, 125]. A conservação de energia é um dos aspectos mais importantes a ser considerado no projeto das redes de sensores sem fio.

## 2.2 Comunicação de Dados em Redes de Sensores Sem Fio

A comunicação de dados representa um dos principais tópicos de pesquisa em RSSFs porque essa tarefa corresponde a um dos maiores consumos de energia nessas redes. O consumo de energia da comunicação de dados é aproximadamente três ordens de magnitude maior que o de processamento. Segundo [101], um nó sensor pode executar 3000 instruções consumindo a mesma quantidade de energia necessária para enviar um bit a 100 m. Tanto a camada de acesso ao meio quanto a de rede são responsáveis pela comunicação de dados em RSSFs. O restante desta seção descreve as formas de comunicação de dados em RSSFs, os projetos combinados e os integrados, as políticas MAC pró-ativa e reativa, as abordagens de roteamento baseadas no emissor e no receptor e a família de protocolos *gossip* em RSSFs.

### 2.2.1 Formas de Comunicação de Dados

A atividade de comunicação de dados em redes de sensores sem fio, sob o ponto de vista das entidades de comunicação, em geral, pode ser dividida em três casos como mostrado na figura 2.1: dos sensores para o nó monitor, entre os nós sensores vizinhos, e do nó monitor para os nós sensores. A primeira, chamada de coleta de dados, é usada pelos nós sensores para enviar dados de sensoriamento ou de controle para o nó monitor [21, 69]. A segunda está normalmente relacionada a alguma tarefa de cooperação [6, 79]. Por exemplo, os nós sensores podem trocar informações sobre um objeto em movimento com objetivo de enviarem para o nó monitor a melhor informação sobre a posição do objeto. A terceira, chamada de disseminação de dados, é usada para disseminar um ou mais dados para um subconjunto de nós sensores [50, 70]. Por exemplo, o nó monitor pode disseminar um novo interesse para rede ou alterar o modo de operação da rede toda ou de partes da mesma. Um tipo especial de disseminação é a difusão de dados (*broadcasting*) que acontece quando o nó monitor deseja enviar um dado para todos os nós da rede [25, 140].

### 2.2.2 Projetos Combinado e Integrado entre Camadas

A pilha de protocolos e todas as aplicações para RSSFs devem ser otimizadas para usar a energia da melhor forma possível. Diferente de outras redes que são normalmente projetadas para suportarem várias aplicações, as RSSFs são tipicamente destinadas a

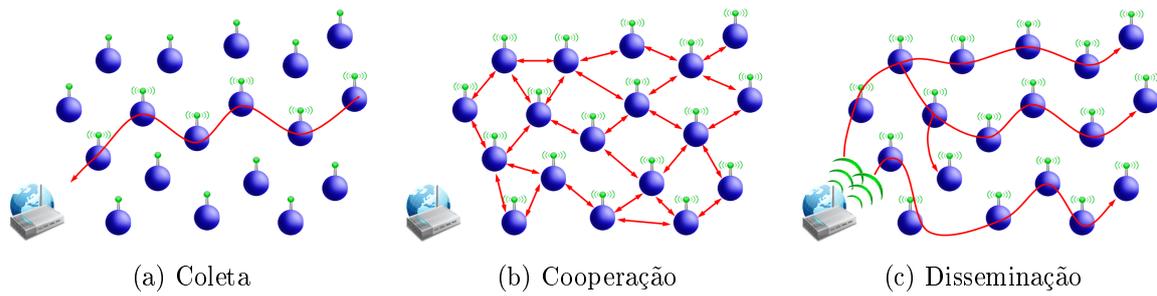


Figura 2.1: Formas de comunicação de dados em redes de sensores sem fio

uma aplicação específica. Assim, o desenvolvimento de duas ou mais camadas da pilha de protocolos pode acontecer de forma combinada ou integrada para explorar as vantagens e evitar as desvantagens de outras camadas. A consequência desses projetos serão os ganhos de desempenho ou a redução do consumo de energia. Um projeto combinado mantém a arquitetura da pilha de protocolos, contudo, escolhe e ajusta cada um dos protocolos da pilha para favorecer o funcionamento básico dos protocolos das outras camadas. Um exemplo de projeto combinado na área de redes de computadores acontece quando os protocolos de transporte *User Datagram Protocol* (UDP) e *Transport Control Protocol* (TCP) limitam o tamanho de seus segmentos para minimizar as fragmentações do *Internet Protocol* (IP) [118].

O projeto integrado [5, 13, 39, 90, 91, 110, 116, 126] consiste em uma camada violar a arquitetura da pilha de protocolos para acessar ou modificar informações disponíveis em outra camada. As técnicas de projeto integrado podem ser divididas em dois grupos [116]. O primeiro mantém a arquitetura da pilha de protocolos e permite que uma camada acesse ou atualize informações disponíveis em outra. O segundo faz a fusão de camadas e cria uma “super camada”. O primeiro grupo faz com que cada camada mantenha seus objetivos e tarefas e as técnicas de projeto integrado realizem as trocas verticais de informações entre as camadas [27]. Por outro lado, o segundo grupo distorce a arquitetura de camadas e embola os objetivos e as tarefas de cada camada. Um exemplo de projeto integrado na área de redes de computadores é o *Network Address Translation* (NAT) [118] no qual a camada de rede de um roteador acessa/atualiza informações sobre a porta (disponível no cabeçalho do segmento e inserida pela camada de transporte) e o IP da origem para permitir que um computador de uma rede interna acesse outra externa.

### 2.2.3 Políticas MAC para Determinar o Escalonamento de Trabalho dos Nós Vizinhos

Os protocolos MAC para RSSFs determinam o escalonamento de trabalho dos vizinhos a partir de políticas MAC pró-ativa e reativa. A seção 2.2.3.1 descreve a ideia básica dos algoritmos de comunicação pró-ativos e reativos. As seções 2.2.3.2 e 2.2.3.3 discutem o funcionamento básico das políticas MAC pró-ativa e reativa para determinar o escalonamento de trabalho dos nós vizinhos, respectivamente.

#### 2.2.3.1 Algoritmos de Comunicação Pró-ativo e Reativo

O funcionamento básico de alguns algoritmos de comunicação depende da existência de alguma informação sobre o processo de encaminhamento de mensagens. Por exemplo, alguns algoritmos de roteamento dependem de uma tabela de rotas. Os algoritmos que dependem dessa informação podem ser classificados em pró-ativos e reativos a partir da forma como criam e atualizam sua informação. No caso da comunicação na camada MAC, essa informação é de escalonamento de trabalho dos nós sensores. Os pró-ativos fazem essas tarefas a priori prevendo que a informação de escalonamento será necessária em um futuro próximo. Os reativos a criam somente quando a mesma se torna necessária e eles não a atualizam. A principal vantagem dos pró-ativos é a disponibilidade da informação sobre escalonamento quando ela é necessária. Isso permite a execução imediata do algoritmo, pode garantir a qualidade desejada devido à existência da informação de escalonamento e fazer o balanceamento da carga de trabalho uma vez que as etapas de criação e atualização da informação normalmente acontecem em momentos distintos ao de sua utilização. A principal vantagem dos reativos é a economia de recursos dado que a obtenção da informação sobre escalonamento acontece somente se ela for necessária. Isso evita o desperdício de recursos quando essa informação não for necessária.

Cada uma dessas estratégias apresenta um cenário mais indicado para seu funcionamento. Os algoritmos pró-ativos normalmente são aplicados a cenários nos quais as atualizações são fundamentais e não frequentes (em relação à utilização da informação de escalonamento) ou sua existência garante algum requisito de qualidade necessário à aplicação. Por exemplo, em aplicações dependentes do tempo, a existência da informação de escalonamento pode minimizar a latência da comunicação. Os algoritmos reativos costumam ser aplicados a cenários nos quais as desatualizações são frequentes e a obtenção dessa informação atende aos requisitos de qualidade da aplicação.

### 2.2.3.2 Política MAC Pró-ativa

A política pró-ativa faz com que cada nó sensor mantenha e atualize uma tabela contendo informações sobre o escalonamento de trabalho de seus vizinhos. Seu funcionamento básico pode ser modelado a partir de eventos e ações de tal forma que, quando um evento acontece, tem-se uma ação correspondente. Alguns exemplos de eventos são a expiração de temporizadores, o recebimento de pacotes ou de requisições. Alguns exemplos de ações são as execuções de comandos tais como ler um dado ou a inicialização de um temporizador. Os eventos da política pró-ativa são “atualizar informação sobre vizinhos”, “enviar quadro” e “receber quadro”. As figuras 2.2 a 2.4 mostram as ações correspondentes para cada um desses eventos.

O início do funcionamento de um nó sensor gera o evento “atualizar informação sobre vizinhos” para o nó compartilhar seu escalonamento de trabalho com seus vizinhos. Ação correspondente a esse evento faz com que o nó corrente envie um quadro de atualização contendo seu escalonamento de trabalho (figura 2.2, ponto *A*). Contudo, como os nós vizinhos quase nunca ficam acordados ao mesmo tempo, o nó corrente normalmente envia o mesmo quadro várias vezes. A principal limitação da política pró-ativa é o custo desse compartilhamento de informações. Guo et al. [43, 44] propõem uma solução para minimizar esse custo na qual cada nó emissor envia o quadro somente para uma fração de vizinhos. No próximo passo da ação, a política pró-ativa faz com que o nó reinicialize o temporizador de atualização, reescalando o evento (figura 2.2, ponto *B*). A expiração e reinicialização desse temporizador ocasiona uma repetição periódica do evento de atualização. O projetista da rede determina essa periodicidade baseada em parâmetros tais como a taxa de desatualização das informações sobre vizinhos, risco da informação estar desatualizada em cada camada da pilha de protocolos e fração de vizinhos que deve receber a informação. O ponto *C* da figura 2.2 indica o término da ação.

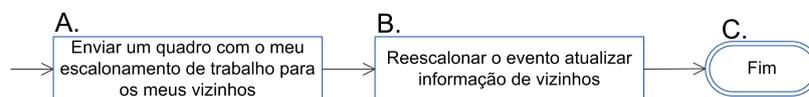


Figura 2.2: Ação correspondente ao evento atualizar informação sobre vizinhos da política pró-ativa

A geração do evento “enviar quadro” acontece através de requisições da camada de rede. A ação correspondente a esse evento faz com que o nó corrente verifique se o próximo nó está acordado (figura 2.3, ponto *A*). Se verdadeiro, o nó corrente envia o quadro (figura 2.3, ponto *B*), senão, ele reescala a transmissão do quadro para

o momento em que o próximo nó estiver acordado (figura 2.3, ponto *D*). A política pró-ativa considera que sua informação sobre escalonamento de trabalho dos vizinhos está atualizada.

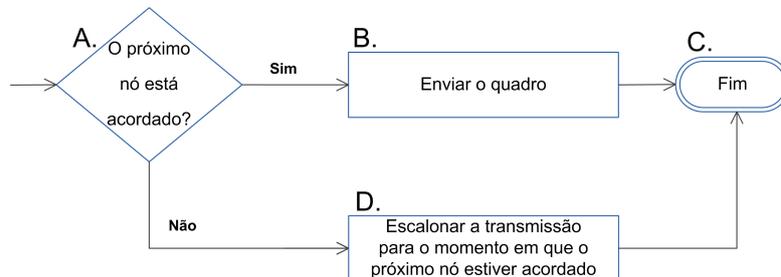


Figura 2.3: Ação correspondente ao evento enviar quadro da política pró-ativa

Quando o evento “receber quadro” acontece, a política pró-ativa faz com que o nó corrente verifique se ele recebeu um quadro de atualização (figura 2.4, ponto *A*). Se verdadeiro, o nó corrente atualiza sua informação sobre o escalonamento de trabalho do nó emissor (figura 2.4, ponto *B*). Caso contrário, ele permite que as demais rotinas da camada MAC processem o quadro (figura 2.4, ponto *D*).

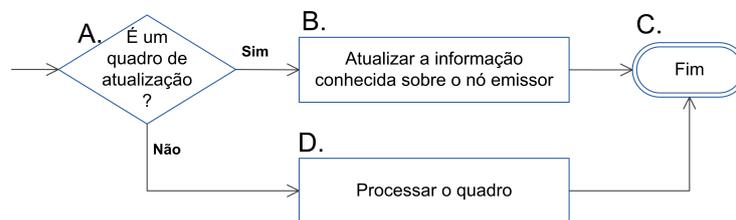


Figura 2.4: Ação correspondente ao evento receber quadro da política pró-ativa

### 2.2.3.3 Política MAC Reativa

A política reativa não armazena qualquer informação sobre o escalonamento de trabalho dos nós vizinhos e sempre que um nó precisa transmitir uma mensagem, ele executa uma nova descoberta para saber o escalonamento de trabalho do próximo nó. O nó emissor faz essa descoberta através de transmissões periódicas de um quadro de descoberta até que ele receba uma resposta do próximo nó ou alcance o número máximo de retransmissões. A principal diferença entre os protocolos MAC reativos está na forma como eles determinam o número máximo de retransmissões e o intervalo entre elas. O funcionamento básico da política reativa também pode ser modelado a partir de eventos e ações. Seus eventos são “enviar quadro” e “receber quadro”. As figuras 2.5 e 2.6 mostram as ações correspondentes a cada um desses eventos, respectivamente.

A ação correspondente ao evento “enviar quadro” faz com que o nó corrente inicie uma nova descoberta através da criação e do envio de um quadro de descoberta (figura 2.5, pontos *A* e *B*). Em seguida, o nó corrente incrementa um contador de retransmissões para a descoberta corrente (figura 2.5, ponto *C*) e verifica se terá outra chance para descobrir o escalonamento do próximo nó (figura 2.5, ponto *D*). O número de chances corresponde ao limite máximo de retransmissões do quadro de descoberta. Em caso positivo, o nó reescala o evento de enviar quadro, o que permite uma próxima retransmissão do quadro de descoberta (figura 2.5, ponto *E*). Em caso negativo, o nó simplesmente termina a descoberta sem sucesso (figura 2.5, ponto *F*). Por outro lado, quando o nó corrente recebe uma resposta do próximo nó, a execução da ação correspondente ao evento “receber quadro” termina a descoberta com sucesso.

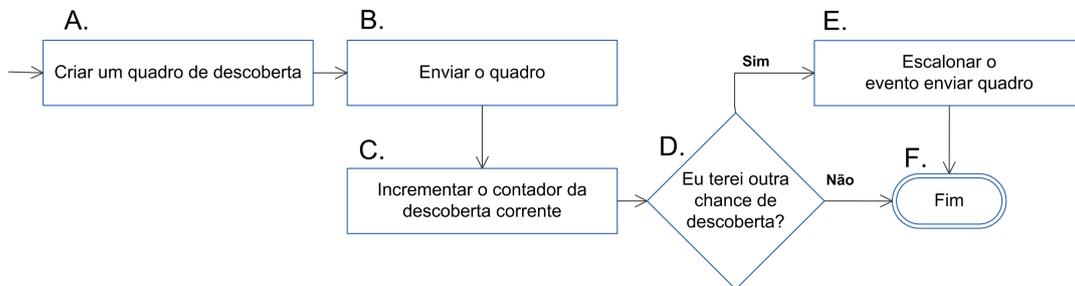


Figura 2.5: Ação correspondente ao evento enviar quadro da política reativa

A ação correspondente ao evento “receber quadro” faz com que o nó corrente verifique se ele é o destino do quadro recebido (figura 2.6, ponto *A*). Se verdadeiro, ele verifica se o quadro recebido é de descoberta (figura 2.6, ponto *B*). Se o quadro for de descoberta, o nó corrente é destino de uma descoberta enviada pelo emissor do quadro recebido. Logo, a política reativa faz com que o nó corrente envie um quadro de resposta para informar ao emissor da descoberta que o nó corrente está acordado (figura 2.6, ponto *C*). Se o quadro não for de descoberta, o nó corrente verifica se o quadro recebido é de resposta (figura 2.6, ponto *E*). Se verdadeiro, isso significa que o nó corrente iniciou uma descoberta cujo destino está acordado e o nó corrente pode enviar os dados desejados (figura 2.6, ponto *F*). Além disso, como o nó corrente descobriu a informação desejada e a descoberta reativa é baseada no reescalonamento do evento “enviar quadro”, a política reativa cancela o escalonamento desse evento uma vez que ele não é mais necessário (figura 2.6, ponto *F*). Quando o quadro não é de descoberta nem de resposta, o nó corrente apenas recebe o quadro a ser processado por outros procedimentos da camada MAC (figura 2.6, ponto *G*). Quando o nó corrente não é o destino do quadro recebido (figura 2.6, ponto *A*), a ação termina (figura 2.6, ponto *D*).

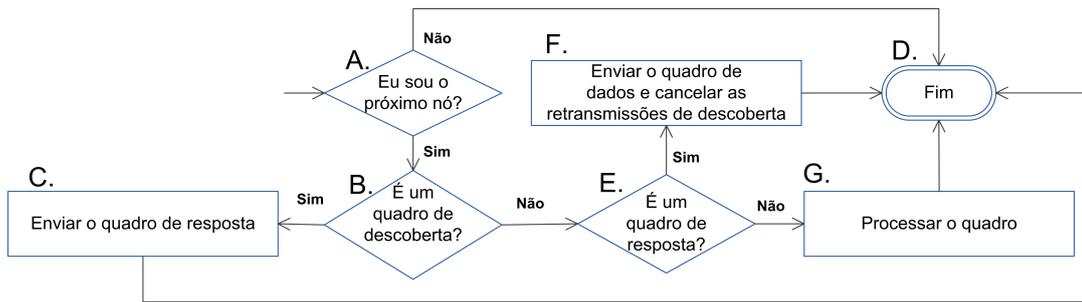


Figura 2.6: Ação correspondente ao evento receber quadro da política reativa

### 2.2.4 Políticas de Roteamento para Escolha do Próximo Nó

A camada de rede é responsável por definir as rotas através da escolha sistemática do próximo nó durante o roteamento. Essa escolha é chamada de decisão de propagação e envolve duas entidades: o nó emissor e os receptores. Os protocolos de roteamento são classificados como baseados no emissor quando o nó emissor escolhe o próximo nó. Caso contrário, eles são protocolos baseados no receptor uma vez que os nós receptores serão responsáveis pela escolha. A maioria dos protocolos de roteamento para RSSFs utiliza a abordagem baseada no emissor. Isso é uma consequência das redes cabeadas em que cada nó possui um enlace para cada um de seus vizinhos e a natureza dessa comunicação é um-para-um (*unicasting*). Por outro lado, a abordagem baseada no receptor é mais recente, aparecendo com as redes sem fio em que o meio é compartilhado. A natureza da comunicação no roteamento baseado no receptor é um-para-todos (*broadcasting*) e quando um nó envia um pacote, todos seus vizinhos podem recebê-lo sem um custo extra de energia.

A escolha baseada no emissor faz com que cada nó emissor insira no pacote uma referência que seja suficiente para cada nó receptor saber se foi escolhido para continuar o roteamento. Essa referência normalmente é o identificador do próximo nó. A figura 2.7 ilustra o funcionamento de uma abordagem baseada no emissor quando um nó recebe um pacote. O primeiro passo dessa abordagem faz com que o nó receptor verifique se ele é o próximo nó do pacote recebido (figura 2.7, ponto A). Se verdadeiro, o nó escolhe o próximo nó do pacote (figura 2.7, ponto B) e insere uma referência para esse nó no cabeçalho do pacote (figura 2.7, ponto C). Essa escolha normalmente é baseada em alguma informação sobre vizinhos anteriormente compartilhada. Em seguida, o nó propaga o pacote (figura 2.7, ponto D). Quando um nó recebe um pacote e ele não é o próximo nó (figura 2.7, ponto A), ele apenas descarta o pacote (figura 2.7, ponto F).

A escolha baseada no receptor faz com que os nós receptores usem informações

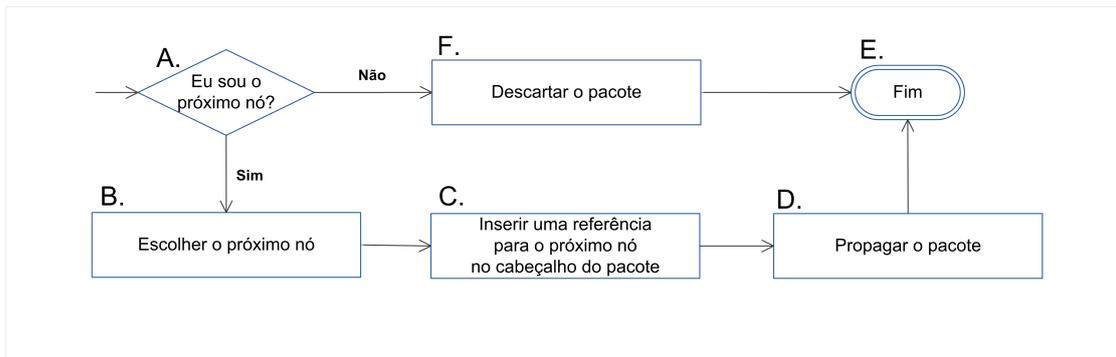


Figura 2.7: Funcionamento básico do roteamento baseado no emissor

loais (e.g., energia residual ou localização) para decidirem se devem continuar o roteamento. Os nós emissores também podem inserir no cabeçalho do pacote alguma referência para o próximo nó, contudo, nesse caso, os receptores combinam essa referência com informações locais para tomarem a decisão de propagação. Se na escolha baseada no emissor, a referência é suficiente para garantir a decisão do emissor, na baseada no receptor, a insuficiência ou inexistência da referência garante que a escolha será dos receptores. A figura 2.8 ilustra o funcionamento básico de uma abordagem baseada no receptor quando um nó recebe um pacote. Nessa abordagem, inicialmente, o nó receptor lê suas informações locais e as combina com a referência contida no pacote (se essa existir) (figura 2.8, ponto *A*). Em seguida, o nó utiliza a informação combinada para decidir se propaga o pacote (figura 2.8, ponto *B*). Se verdadeiro, o nó calcula sua referência para o próximo nó, a insere no cabeçalho do pacote e o propaga (figura 2.8, pontos *C*, *D* e *E*). Se o nó decidir não propagar o pacote (figura 2.8, ponto *B*), ele descarta o pacote (figura 2.8, ponto *G*).

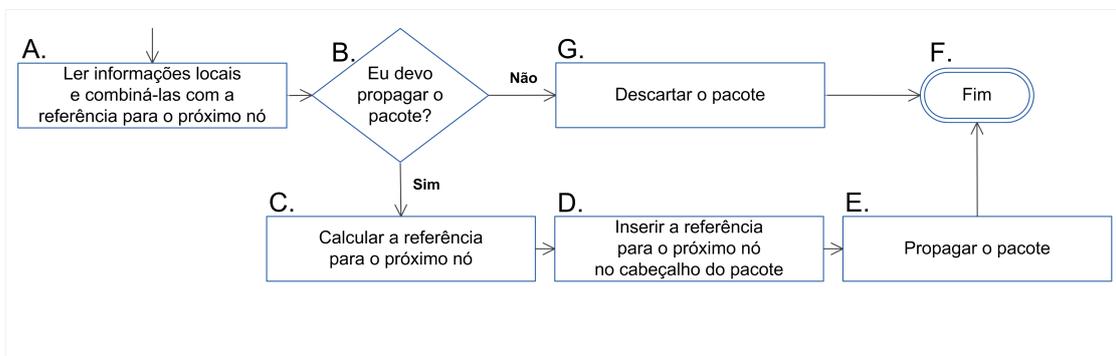


Figura 2.8: Funcionamento básico do roteamento baseado no receptor

A escolha baseada no receptor (figura 2.8, ponto *B*) acontece de forma imediata ou com atraso. Na primeira, quando um nó recebe um pacote, ele toma a decisão

imediatamente (e.g., usando probabilidade). Na abordagem com atraso, quando um nó recebe um pacote pela primeira vez, ele espera um tempo. Após esse tempo, se nenhum vizinho tiver propagado, o nó propaga. O projetista da rede define os parâmetros da decisão imediata ou o tempo de atraso a partir de informações locais do nó sensor. Essa definição considera a relação custo-benefício entre o consumo de energia, robustez e latência. O aumento das chances de cada nó propagar um pacote normalmente aumenta o número de transmissões e o consumo de energia da rede, reduz a latência porque os pacotes podem ser roteados por mais rotas e aumenta a robustez devido às novas transmissões. Uma variação da abordagem com atraso consiste em cada nó propagar um pacote recebido com uma probabilidade  $p$ . Se o nó não propagar, ele aguarda um tempo mínimo de espera e tenta novamente. O número de tentativas depende de cada protocolo, o valor de  $p$  é obtido a partir das informações locais e o tempo de espera é suficiente para que um nó transmita um pacote e esse seja recebido e processado pelos vizinhos.

Um parâmetro especial que diferencia as abordagens de roteamento baseadas no emissor e no receptor é o número de candidatos a próximo nó. O roteamento baseado no emissor tem normalmente um único candidato. Na abordagem baseada no receptor, qualquer vizinho que receber o pacote pode ser um candidato.

### 2.2.5 Escolha do Próximo Nó a partir do Escalonamento de Trabalho

Uma diferença entre as abordagens de roteamento baseadas no emissor e no receptor é que a primeira necessita que o emissor tenha informações sobre seus vizinhos para efetuar a escolha do próximo nó. Caso essa informação não exista ou esteja desatualizada, o protocolo de roteamento pode escolher nós menos indicados ou indisponíveis. A abordagem baseada no receptor não tem essa dependência porque sua escolha acontece *a posteriori*, nos nós receptores. Em redes com topologia dinâmica, a abordagem baseada no emissor normalmente precisa que as informações compartilhadas estejam atualizadas. Os projetos combinados ou integrados de comunicação de dados envolvendo as camadas MAC e de rede podem fazer com que a de rede use a disponibilidade de informações sobre vizinhos na camada MAC para escolher o próximo nó.

A maioria das soluções para a comunicação de dados em RSSFs utiliza uma abordagem de roteamento baseada no emissor e uma política MAC pró-ativa. Isso é uma herança das redes cabeadas em que cada nó possui um enlace com cada um de seus vizinhos e a topologia quase nunca se altera. Um projeto combinado ou integrado entre essas políticas pode fazer com que a camada de rede use as informações sobre

vizinhos da política MAC pró-ativa para escolher o próximo nó. Por exemplo, o vizinho que acordar primeiro será o próximo nó [116]. Além disso, a política MAC pró-ativa pode definir a taxa de atualização da informação compartilhada a partir de parâmetros da camada de rede.

O projeto combinado entre o roteamento baseado no receptor e a política MAC reativa é o foco da segunda contribuição desta tese (capítulo 5) em que a diversidade de candidatos a próximo nó pode minimizar o custo da descoberta reativa. O autor desta tese desconhece outros trabalhos que combinem o roteamento baseado no receptor e a política MAC reativa.

As outras combinações são o roteamento baseado no emissor com a política MAC reativa e o roteamento baseado no receptor com a política pró-ativa. Todos os trabalhos que o autor desta tese conhece sobre o protocolos MAC reativos consideram o roteamento baseado no emissor como, por exemplo, [24, 67, 83, 103]. A limitação dessa abordagem é que a camada de rede não tem informações para fazer sua escolha do próximo nó. A abordagem baseada no receptor com a política MAC pró-ativa faz com que cada nó tenha informações sobre vizinhos, mas não a utilize na decisão de roteamento.

## 2.2.6 Família de Protocolos *Gossip*

O termo *gossip* tem duas definições na área de redes de computadores e ambas têm publicações em RSSFs. A figura 2.9 mostra essas duas definições e o restante desta seção discute e exemplifica cada uma delas.



Figura 2.9: A classificação do termo *Gossip* em redes de sensores sem fio

A primeira definição é chamada de “todos para todos” e está relacionada a um paradigma de comunicação de dados no qual cada nó da rede tem uma informação que deve ser compartilhada com todos os demais. Esse *gossip* também é conhecido como “problema do telefone”. Baker et al. [11] e Tjeldeman [119] descrevem o problema do telefone como: “Tem-se  $n$  senhoras e cada uma conhece um segredo desconhecido pelas demais. Elas se comunicam através do sistema de telefonia tradicional e quando duas senhoras conversam, uma passa para a outra todos os segredos que conhece”. Segundo [49, 94], o problema principal consiste em determinar o número de ligações

necessárias para que todas as senhoras conheçam todos os segredos. Alguns estudos estão interessados no número de *rounds* necessários para que todas as senhoras conheçam todos os segredos. Um *round* é um conjunto de ligações simultâneas que acontecem em um mesmo instante. Outros trabalhos abordam o tamanho mínimo da rede de *gossip* [60]. Hedetniemi et al. [53] mostram um ótimo *survey* sobre essa versão do *gossip*. Boyd et al. [18] e Aoyama et al. [95] apresentam uma análise do tempo de difusão da informação dessa versão do *gossip* em RSSFs.

A segunda definição de *gossip* considera uma família de protocolos de roteamento. Cada passo desses protocolos seleciona um conjunto aleatório de nós para propagar o pacote. A maioria dos trabalhos sobre *gossip* em RSSFs considera essa definição. Os protocolos de roteamento da família *gossip* podem ser diferenciados em quatro métricas: escolha do próximo nó, comunicação de dados, tipo de decisão e parâmetro de decisão. A figura 2.10 ilustra o valor de cada uma dessas métricas. A família de protocolos *gossip* escolhe o próximo nó usando as abordagens baseadas no emissor, receptor ou híbrida. Na primeira abordagem, também chamada de roteamento epidêmico, quando um nó recebe um pacote, ele escolhe aleatoriamente um subconjunto de seus vizinhos para propagar o pacote. Existem trabalhos anteriores a 1955 que abordam esse tipo de protocolo de roteamento como, por exemplo, [72, 77, 102]. Na abordagem baseada no receptor, quando um nó recebe um pacote, ele decide aleatoriamente se deve propagá-lo. Khelil et al. [74] apresentam uma proposta sobre o *gossip* epidêmico em RSSF. [41, 47, 48, 82, 86, 88, 98, 135, 136] propõem soluções de roteamento pertencentes à família *gossip* e baseadas no receptor para RSSFs. Zhang et al. [137] descreve uma solução híbrida combinando as abordagens baseadas no emissor e no receptor.

Os protocolos da família *gossip* são usados em RSSFs nas seguintes formas de comunicação: *broadcasting*, *multicasting* e *unicasting*. O *Gossiping using the Energy Map* (GEM) (proposto no capítulo 6) e o *Gossip* [48] são protocolos para *broadcasting*. O *Epidemic routing* [120], *Trajectory and Energy-based Data Dissemination* (TEDD) [41, 86] e o *Regional Gossiping* [82] são protocolos para aplicações *unicasting* e *multicasting*.

O tipo de decisão dos protocolos da família *gossip* é o baseado em probabilidade ou em tempo. A maioria dos protocolos dessa família são baseados em probabilidade como em [32, 48] no qual cada nó receptor propaga o pacote com probabilidade  $p$  e o descarta com probabilidade  $1 - p$ . Os protocolos *gossip* baseados no receptor normalmente usam a técnica baseada em tempo na qual quando um nó recebe um pacote, ele espera um intervalo de tempo para propagá-lo. Os nós sensores calculam esse tempo usando alguma variável aleatória e, depois do mesmo, se outro vizinho não tiver propagado o pacote, o nó corrente o propaga. Alguns exemplos de protocolos *gossip* baseados em

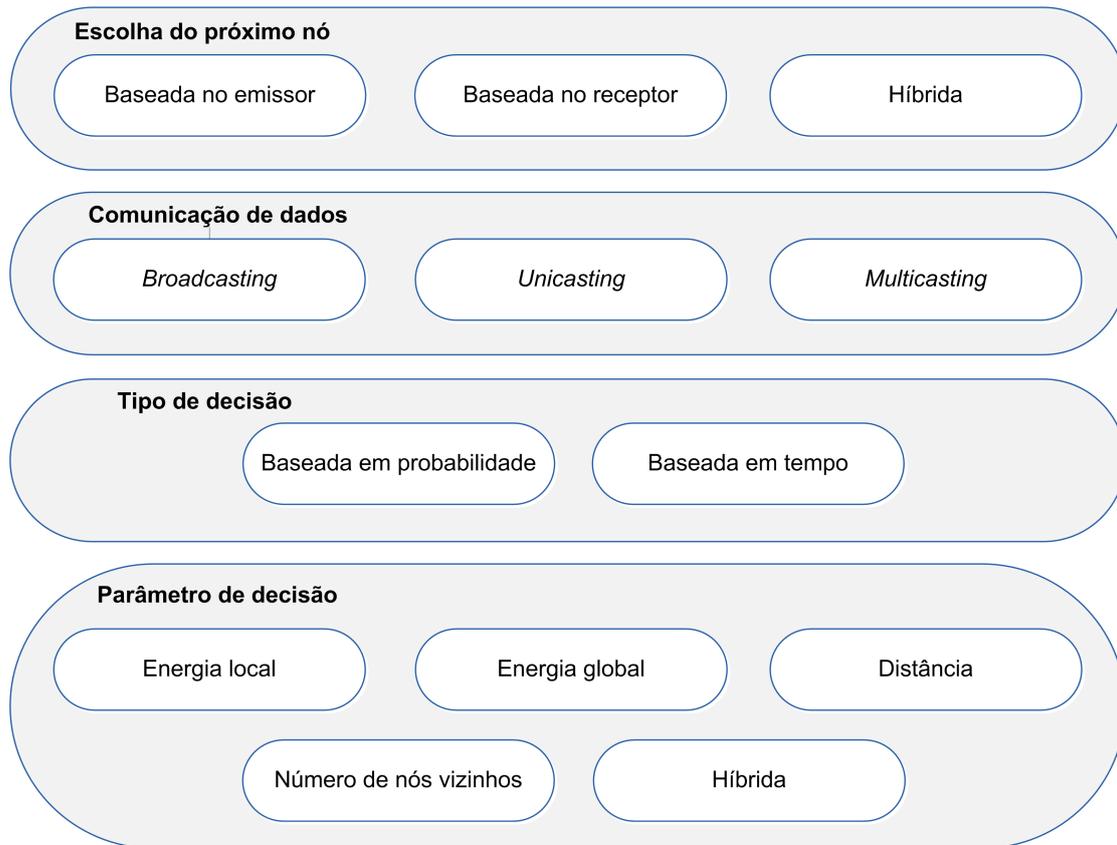


Figura 2.10: Métricas de decisão para os protocolos de roteamento da família *gossip* em redes de sensores sem fio

tempo são o TEDD e o *Dynamic Delayed Broadcasting 2* (DDB2) [55].

O parâmetro de decisão normalmente consiste em energia local ou global, distância, densidade (número de nós vizinhos) ou híbrido. O TEDD, por exemplo, é um protocolo *gossip* que usa uma informação sobre a energia global da rede para disseminar dados em redes de sensores sem fio. O DDB2 é um exemplo de protocolo *gossip* que usa uma informação local sobre energia para efetuar a difusão de dados. O *Gossip 2* considera o número de vizinhos de um nó para calcular sua probabilidade de transmissão. O GEM combina uma informação global de energia com a distância entre nós vizinhos como seu parâmetro de decisão.

## 2.3 Conclusões

A comunicação de dados é uma tarefa crucial em redes de sensores sem fio uma vez que nós sensores adormecem periodicamente para economizar energia. Quando um nó envia uma mensagem, a comunicação pode falhar se o próximo nó estiver adormecido.

Este capítulo discute a comunicação de dados em RSSFs. Os protocolos MAC para RSSFs normalmente abordam esse desafio da comunicação de dados compartilhando o escalonamento de trabalhos dos nós vizinhos ou descobrindo tal informação quando a mesma se faz necessária. As contribuições deste trabalho estão relacionadas com essas abordagens e formalizadas neste capítulo. Este capítulo discute os conceitos de roteamento baseado no emissor e no receptor necessários para o capítulo 5. Os conceitos de projetos combinados e integrados são apresentados neste capítulo e necessários nos capítulos 5 e 6, respectivamente. Outro tópico abordado neste capítulo é a família de protocolos *gossip* que contém o protocolo proposto no capítulo 6.

# Capítulo 3

## Trabalhos Relacionados

*Our knowledge is the amassed thought and experience of innumerable minds.  
Ralph Waldo Emerson*

Este capítulo apresenta os trabalhos relacionados desta tese. As seções 3.1 e 3.2 descrevem protocolos para as camadas MAC e de rede, respectivamente. A seção 3.3 finaliza esse capítulo apresentando suas conclusões.

### 3.1 Protocolos para a Camada MAC

Esta seção descreve alguns protocolos MAC propostos na literatura para RSSFs. A seção 3.1.1 apresenta uma breve introdução sobre esses protocolos. As seções 3.1.2 e 3.1.3 apresentam protocolos MAC reativos e pró-ativos, respectivamente. A tabela 3.1 elenca os protocolos descritos nas duas últimas seções.

#### 3.1.1 Introdução

Apesar da camada MAC ser um assunto bem estudado nas redes cabeadas e sem fio, o cenário desafiador das RSSFs cria uma mistura complexa de questões que demanda o desenvolvimento de soluções especiais para essas redes. Os protocolos MAC para outras redes sem fio normalmente maximizam a vazão, minimizam o atraso da rede e mantêm o consumo de energia reduzido como um requisito secundário. Por outro lado, o principal desafio dos protocolos MAC em RSSFs é o consumo de energia reduzido. Em seguida, eles priorizam a alocação de banda e o atraso reduzido.

Como mostrado em [10, 33, 78, 84], os protocolos MAC normalmente fazem com que os nós sensores adormeçam e evitem as principais formas de consumo de energia em RSSFs [128]:

<b>Reativos</b>	<ul style="list-style-type: none"> <li>-<i>Asynchronous MAC</i> (AMAC)</li> <li>-<i>High Efficient Sensor MAC</i> (HESMAC)</li> <li>-<i>Geographic Random Forwarding</i> (GeRaF)</li> <li>-<i>Implicit Geographic Forwarding</i> (IGF)</li> <li>-<i>Distributed Passive Routing Decisions</i> (DPRD)</li> <li>-<i>Unified Cross-layer Module</i> (XLM)</li> <li>-<i>Receiver-based MAC</i> (RbMAC)</li> <li>-<i>Gossip-based Sleep Protocol</i> (GSP)</li> <li>-<i>Preamble Sampling</i></li> <li>-<i>Sparse Topology and Energy Management</i> (STEM)</li> <li>-<i>Receiver-Initiated MAC</i> (RI-MAC)</li> </ul>
<b>Pró-ativos</b>	<ul style="list-style-type: none"> <li>-<i>Sensor-MAC</i> (SMAC)</li> <li>-<i>Battery Aware MAC</i> (BAMAC)</li> <li>-<i>Sequential MAC</i> (SeMAC)</li> <li>-<i>Opportunistic MAC</i> (OMAC)</li> <li>-<i>Multiple Access Collision Avoidance protocol for Multicast</i> (MACAM)</li> <li>-<i>WiseMAC</i></li> <li>-<i>Carrier Sense Multiple Access Minimum Preamble Sampling</i> (CSMA-MPS)</li> <li>-<i>X-MAC</i></li> <li>-<i>Rate Estimation MAC</i></li> <li>-<i>Lazy Forwarding in Low Duty Cycle Wireless Sensor Networks</i> (L<sup>2</sup>)</li> </ul>

Tabela 3.1: Classificação de alguns protocolos MAC reativos e pró-ativos

- **Escuta ociosa:** acontece quando um nó mantém seu rádio ligado para receber eventuais quadros e não existe qualquer tráfego de dados na área coberta pelo nó. Essa é a principal forma de consumo de energia em RSSFs.
- **Overhearing:** acontece quando um nó mantém seu rádio ligado e todos os quadros recebidos não são destinados a ele. Esse é outro efeito de um nó manter seu rádio ligado. Isso é problemático, principalmente, em redes densas.
- **Colisão:** acontece quando um nó tem dois ou mais vizinhos transmitindo quadros simultaneamente. Isso resulta em um sinal truncado e faz com que o receptor descarte os quadros.
- **Overhead (cabeçalhos e mensagens de controle):** bytes não úteis para a aplicação cujas transmissões consomem a mesma quantidade de energia que a dos úteis. Em outras redes, o número de bytes dos cabeçalhos e das mensagens de controle normalmente é menor que o dos dados úteis, amortizando o custo do *overhead*. Contudo, em RSSFs, como os dados produzidos pela aplicação normalmente se resumem a alguns bytes, o custo de energia com a transmissão de cabeçalhos e mensagens de controle merece uma atenção especial do projetista da rede.

Os protocolos MAC para RSSF normalmente consideram os esquemas tradicionais para controle de acesso ao meio como o *Aloha*, *Carrier Sense Multiple Access*(CSMA) e *Carrier Sense Multiple Access with Collision Avoidance*(CSMA/CA). No *Aloha*, quando um nó tem dados a serem enviados, ele sempre os transmite. Em seguida, o nó verifica se aconteceu alguma colisão. Se verdadeiro, o nó espera um tempo aleatório e transmite o dado novamente. No CSMA, quando um nó tiver que enviar um quadro, ele escuta o meio e verifica se esse está ocupado. Se verdadeiro, o nó espera até que o meio fique livre para efetuar sua transmissão. As versões do CSMA se diferenciam pela forma como verificam novamente se o meio continua ocupado. Na versão 1-persistente, o nó escuta o meio e quando esse fica livre, o nó inicia imediatamente sua transmissão. Na versão  $p$ -persistente, o nó escuta o meio e quando esse fica livre, o nó inicia sua transmissão com probabilidade  $p$  e, com probabilidade  $1 - p$ , ele aguarda um tempo para começá-la. Na versão não persistente, antes de transmitir, o nó analisa o meio. Se esse estiver livre, ele transmite, senão, ele pára de escutar o meio e aguarda um tempo aleatório para tentar transmitir novamente. Jamieson et al. [65] apresentam uma variação interessante do CSMA não persistente na qual os nós sensores escolhem o tempo aleatório da janela de contenção através de uma distribuição de probabilidade que priorize valores próximos ao término da janela. Os protocolos CSMA não

persistente tradicionalmente consideram uma distribuição de probabilidade uniforme. O principal problema do CSMA é o terminal escondido no qual tem-se três nós A, B e C sendo que B é vizinho de A e C e esses não são vizinhos entre si. Como A e C não são vizinhos, eles podem enviar quadros simultaneamente, contudo, B os receberá de forma truncada porque o sinal de um quadro deturpa o recebimento do outro. O CSMA/CA resolve o problema do terminal escondido através de uma troca de mensagens de controle que reserva o canal para a transmissão de dados. Quando um nó deseja enviar dados e o canal está livre, ele envia um quadro *Request to Send* (RTS) para o próximo nó da rota. Se esse nó puder receber os dados, ele responde com um quadro *Clear to Send* (CTS). Os vizinhos que recebem os quadros RTS ou CTS sabem que uma transmissão de dados acontecerá em breve e que eles devem esperar para efetuar qualquer transmissão. A troca de RTS/CTS é indicada para a comunicação *unicasting*. Na comunicação *broadcasting*, essa técnica provocaria uma implosão de mensagens de CTS. Algumas soluções para essa questão na comunicação *broadcasting* são apresentadas em [115, 122].

Apesar do funcionamento básico dos protocolos MAC para RSSFs ser baseado nas técnicas tradicionais, observa-se alguns pontos devido ao ambiente dinâmico dessas redes:

- Manter o rádio ligado significa consumir energia;
- O tamanho (em número de bytes) dos quadros de dados é similar ao dos de controle (e.g., RTS e CTS);
- A vantagem do *handshaking* RTS/CTS depende das condições de tráfego, características do canal sem fio, topologia da rede e tamanho das mensagens;
- As mensagens de controle também podem sofrer colisões, o que aumenta o consumo de energia da rede.

A literatura sobre RSSFs apresenta diversas classificações para os protocolos para controle e acesso ao meio. Por exemplo, a dependência do tráfego [10], definição do ciclo operacional [78], acesso ao meio [84] e sincronização do escalonamento de trabalho [24, 67, 75, 83, 103]. A classificação de dependência do tráfego divide os protocolos MAC segundo seus respectivos comportamentos em cenários com tráfego cuja carga de mensagens a serem roteadas é alta, média, baixa ou mista. A definição do ciclo operacional considera os parâmetros internos do mecanismo de dormir dos protocolos MAC (e. g, ciclo de trabalho, tamanho do *slot* e número de *slots*). A classificação de acesso ao meio é a mais tradicional e divide os protocolos de acordo com suas respectivas

técnicas para acesso ao meio. A classificação sobre a sincronização do escalonamento de trabalho os divide considerando se os nós vizinhos sincronizam ou não seus respectivos escalonamentos para dormirem e acordarem ao mesmo tempo. A classificação em protocolos MAC pró-ativos e reativos proposta nesta tese se diferencia das existentes por dividir esses protocolos considerando se os nós sensores compartilham ou não seus escalonamentos de trabalho com os nós vizinhos.

### 3.1.2 Protocolos MAC Reativos

A principal diferença entre os protocolos reativos está em seus respectivos processos de descoberta. O restante desta seção apresenta os protocolos *Asynchronous MAC* (AMAC) [83], *High Efficient Sensor MAC* (HESMAC) [103], *Geographic Random Forwarding* (GeRaF) [142], *Implicit Geographic Forwarding* (IGF) [16], *Distributed Passive Routing Decisions* (DPRD) [112], *Unified Cross-layer Module* (XLM) [5], *Gossip-based Sleep Protocol* (GSP) [58, 59], *Preamble Sampling* [35], *Sparse Topology and Energy Management* (STEM) [109] e *Receiver-Initiated MAC* [117].

O AMAC e HESMAC exploram os quadros RTS e CTS como os quadros de descoberta e resposta, respectivamente. As definições do número máximo de retransmissões do quadro RTS e do intervalo entre elas é baseada no escalonamento de trabalho dos nós sensores. Os dois protocolos evitam colisões usando técnicas tradicionais para redes sem fio. [24, 66, 67] propõem trabalhos similares a esses protocolos.

O GeRaF controla o acesso ao meio através de uma técnica de detecção de portadora baseada em sinais de ocupado e em um *handshaking* de quadros RTS e CTS. O GeRaF considera que os nós sensores são equipados com dois rádios e evita colisões nos receptores definindo uma frequência para a troca de dados e outra para a dos sinais de ocupado. Quando um nó deseja enviar um quadro, ele escuta as duas frequências. Se as duas estiverem livres, ele transmite. O *handshaking* de quadros RTS e CTS do GeRaF é baseado no receptor. Assim, os quadros RTS não possuem um destino específico e qualquer vizinho que recebê-lo pode enviar um CTS com uma prioridade baseada no receptor. Por exemplo, os nós mais próximos de uma determinada coordenada respondem primeiro. Em [141], os autores propõem uma extensão do GeRaF na qual os nós sensores operam com um rádio.

O IGF, DPRD e XLM são protocolos MAC que também controlam o acesso ao meio através de um *handshaking* de quadros RTS e CTS baseado no receptor. A principal diferença entre eles (e entre o GeRaF) está na atribuição da prioridade baseada no receptor. Entretanto, como observado em [106], o uso de quadros RTS/CTS não é uma solução apropriada para as RSSFs em que os quadros de dados têm praticamente

o mesmo tamanho dos de controle.

O GSP é proposto para fazer com que os nós sensores adormeçam periodicamente, reduzindo o consumo de energia em RSSFs. O GSP evita colisões através do *Distributed Coordination Function* (DCF) do IEEE 802.11 para *Wireless Local Area Networks* (WLANs). A ideia desse protocolo é que se o *Gossip* pode entregar uma mensagem para todos os nós da rede, então, os nós estão conectados no mínimo pela árvore de difusão. Logo, uma porcentagem  $p$  de nós pode dormir sem que a conectividade da rede seja perdida. O GSP apresenta duas versões. Na síncrona, os nós dormem e acordam ao mesmo tempo e, na assíncrona, isso não é sincronizado. Na primeira versão, inicialmente, cada nó sensor decide dormir com probabilidade  $p$  e ficar acordado com probabilidade  $1-p$ . Após um período específico de tempo, os nós que estavam dormindo acordam e os que estavam acordados dormem. Esse processo repete sistematicamente após o mesmo período de tempo. Por outro lado, na versão assíncrona, cada nó decide por conta própria e de forma aleatória se deve dormir ou permanecer acordado e o tempo que deve permanecer em cada um desses estados. Após esse tempo, cada nó repete esse processo independente do demais. [20] apresenta uma análise do consumo de energia do GSP em RSSFs. Em [57], os autores apresentam o protocolo *Adaptive Gossip-based Sleep* (AGSP), uma extensão do GSP para redes em que a densidade da rede não é uniforme. A ideia do AGSP é que cada nó sensor identifique sua densidade local (número de vizinhos) e ajuste dinamicamente sua *gossip sleep probability*, evitando desconexões em áreas esparsas e nós desnecessariamente acordados em áreas densas. O AGSP também permite que cada nó sensor determine sua probabilidade em conjunto com seus vizinhos. Em [59], os autores propõem o *Traffic-aware Gossip-based Sleep Protocol* (TGSP), outra extensão do GSP. O TGSP reduz o atraso e a taxa de mensagens perdidas quando o GSP coloca os nós para dormirem, interrompendo o tráfego de dados. O TGSP aumenta a probabilidade dos nós ficarem acordados quando estiverem participando regularmente do roteamento.

Elhoiydi et al. [35] apresenta o *Preamble sampling*, uma abordagem para aplicações com tráfego de dados esporádico em RSSFs. A ideia desse protocolo é fazer com que cada nó emissor envie um preâmbulo durante  $t$  unidades de tempo antes da transmissão de qualquer quadro de dados. Os nós sensores acordam periodicamente a cada  $t$  segundos e verificam se existe alguma atividade no meio. Se esse estiver livre, eles adormecem novamente. Senão, tem-se um preâmbulo fazendo com que o nó sensor permaneça acordado para aguardar o quadro de dados que será enviado. O *Preamble sampling* foi avaliado com o Aloha e o CSMA não persistente. Schurgers et al. [109] apresentam uma técnica similar ao *Preamble sampling* chamada de STEM na qual os nós executam as fases de despertar e de transferência de dados usando dois rádios

distintos para reduzir a interferência. [42] considera que os nós sensores possuem dois rádios sendo que o secundário opera com uma potência reduzida. Assim, quando um nó desliga o rádio principal, ele liga o secundário que consome menos energia, contudo, pode receber dados de nós vizinhos próximos.

O *Receiver-Initiated MAC* (RI-MAC) é uma adaptação do *Preamble sampling* na qual o nó emissor fica em silêncio e não envia o preâmbulo. Quando um nó sensor acorda, ele envia um quadro de controle indicando que está acordado caso algum vizinho deseje enviar dados. O objetivo do RI-MAC é reduzir o tempo que os protocolos baseados em preâmbulo deixam o meio ocupado. Yoo et al. [132] apresentam duas adaptações do RI-MAC para que os nós sensores operem com ciclo de trabalho dinâmico.

### 3.1.3 Protocolos MAC Pró-ativos

A diferença entre os protocolos MAC pró-ativos propostos na literatura está em como eles fazem com que os nós sensores compartilhem seus respectivos escalonamentos de trabalho. O restante desta seção mostra os protocolos *Sensor-MAC* (SMAC) [129], *Battery Aware Medium Access Control* (BAMAC) [68], *Opportunistic MAC* (OMAC) [43, 44], *Sequential Medium Access Control* (SeMAC) [133], *Multiple Access Collision Avoidance protocol for Multicast service* (MACAM) [80], *WiseMAC* [36], *Carrier Sense Multiple Access Minimum Preamble Sampling* (CSMA-MPS) [89], X-MAC [19], *Lazy Forwarding in Low Duty Cycle Wireless Sensor Networks* ( $L^2$ ) [22] e *Rate Estimation MAC* [92].

Ye et al. [128] propuseram o Sensor-MAC (SMAC) para reduzir o consumo de energia causado pela escuta ociosa, colisões, *overhearing* e *overhead*. Ele reduz o tempo de escuta ociosa colocando os nós para dormirem periodicamente. A comunicação de dados acontece porque os nós vizinhos sincronizam seus escalonamentos de trabalho para dormirem e acordarem simultaneamente. Alguns nós permanecem acordados por um tempo maior para permitir a comunicação entre os grupos com escalonamentos de trabalho distintos. Como os escalonamentos podem desatualizar durante o tempo de vida da rede, os nós vizinhos trocam periodicamente quadros de sincronização com seus vizinhos. O SMAC evita colisões usando a técnica de *Distributed Coordinated Function* (DCF) do IEEE 802.11 [14] o que inclui a detecção de portadora e o *handshaking* com RTS/CTS para evitar o problema do terminal escondido. Os nós sensores escutam o meio antes de transmitirem. Se ele estiver ocupado, o nó adormece e acorda quando o meio estiver livre e o próximo nó acordado. Os nós transmitem os pacotes de *broadcasting* sem RTS/CTS e os de *unicasting* seguindo a sequência RTS/CTS/DATA/ACK

entre a origem e o destino. Após a troca de RTS/CTS, os nós de origem e destino permanecem acordados até o término da comunicação com o pacote de ACK. Em seguida, eles retornam aos seus respectivos escalonamentos de trabalho. O SMAC considera a ideia do PAMAS [111] para evitar o *overhearing*. Assim, ele faz com que os nós que não participam da comunicação e escutaram os quadros de RTS ou CTS adormeçam. O SMAC também considera uma técnica de *message passing* para reduzir o *overhearing* e o *overhead* de controle. Nessa técnica, o SMAC fragmenta as mensagens longas em vários quadros que são enviados em uma rajada contendo apenas uma troca de mensagens de RTS/CTS para reservar o meio para a mensagem toda.

O BAMAC é um protocolo baseado em contenção que explora o efeito de recarga das baterias para controlar o acesso ao meio sem fio em MANETs. No BAMAC, cada nó mantém uma tabela com informações sobre a energia residual de seus vizinhos e quando um nó tem que transmitir um quadro, ele insere no cabeçalho o valor corrente de sua energia. Quando um nó recebe um quadro, ele atualiza sua tabela com as informações do emissor. A principal ideia do BAMAC é calcular o período de *backoff* para os nós em contenção a partir das informações de energia disponíveis na tabela. O quão maior a energia residual de um nó em relação à de seus vizinhos, menor o período de *backoff*. O BAMAC controla o acesso ao meio usando o esquema de *handshaking* com quadros de RTS/CTS.

O OMAC faz com que os nós sensores vizinhos compartilhem seus escalonamentos de trabalho e cada emissor escalone sua transmissão para o momento em que o próximo nó estiver acordado. Se o nó emissor estiver adormecido quando o próximo nó acordar, o OMAC faz uma comunicação oportunista. Nesse tipo de comunicação, o emissor acorda, transmite e retorna ao seu escalonamento de trabalho. O OMAC faz o acesso ao meio através de uma abordagem baseada no *handshaking* de RTS e CTS. O protocolo faz com que as transmissões *broadcasting* sejam tratadas como *unicasting*. Como esse protocolo faz o compartilhamento de informações sobre vizinhos e tal compartilhamento apresenta um custo não desprezível, Guo et al. [43, 44] propõem uma solução para minimizar esse custo na qual cada nó emissor deve enviar o quadro de atualização somente para uma fração de vizinhos.

O protocolo SeMAC tem como objetivo controlar o acesso ao meio sem fio na difusão de dados em redes sem fio com topologia dinâmica. A ideia básica desse protocolo é reservar periodicamente alguns *slots* de tempo nos quais os nós podem enviar mensagens de topologia. Os nós sensores evitam colisões trocando periodicamente mensagens de *hello*.

O protocolo MACAM é proposto para a comunicação *multicast* em redes sem fio. A ideia básica desse protocolo é que o quadro RTS tenha uma lista de próximos nós

especificada em seu cabeçalho. Assim, quando um nó recebe um RTS e ele está na  $k$ -ésima posição da lista, ele responde com um CTS em um tempo proporcional à sua posição na lista, evitando a colisão entre os CTSs. A desvantagem dessa abordagem é seu *overhead* que torna o MACAM impraticável a medida que o número de nós aumenta no campo de sensoriamento. O número de respostas CTS também faz com que o MACAM seja proibitivo para a comunicação *broadcasting*.

O *WiseMAC* é uma adaptação do protocolo MAC reativo *Preamble Sampling* para o roteamento *unicasting*. A ideia do *WiseMAC* é que cada nó sensor compartilhe informações sobre seu escalonamento de trabalho com seus vizinhos para que cada nó emissor possa reduzir a duração de seu preâmbulo. Em [89], Mahlhecht e Bock apresentam o CSMA-MPS que usa o mecanismo de transmissão e recepção do STEM para aumentar a eficiência do mecanismo de dormir/acordar do *WiseMAC*. Em [19], os autores apresentam o X-MAC, um protocolo similar ao CSMA-MPS. O  $L^2$  é uma adaptação do X-MAC na qual nós sensores vizinhos fazem uma sincronização local de seus escalonamentos de trabalho e uma estimativa da qualidade dos *links* que permitem a redução e a divisão dos preâmbulos.

O protocolo *Rate Estimation MAC* é proposto para o roteamento *unicasting* e aborda o problema de acordar todos os vizinhos. A ideia do *Rate Estimation MAC* é fazer com que cada emissor combine com seus receptores quando esses devem acordar novamente. Se os vizinhos estiverem acordados, o emissor pode efetuar sua próxima transmissão.

## 3.2 Protocolos de Roteamento

Esta seção descreve alguns protocolos da camada de rede propostos na literatura para RSSFs. As seções 3.2.1 e 3.2.2 apresentam protocolos de roteamento baseados no emissor. A diferença entre essas seções é que a primeira mostra protocolos não pertencentes à família de protocolos *gossip* e a outra apresenta protocolos dessa família. As seções 3.2.3 e 3.2.4 descrevem protocolos de roteamento baseados no receptor e se diferenciam da mesma forma que as duas anteriores. A tabela 3.2 classifica os protocolos descritos no restante desta seção.

### 3.2.1 Protocolos baseados no Emissor e Não *Gossip*

Esta seção apresenta alguns protocolos de roteamento baseados no emissor e não pertencentes à família *gossip* para as redes de sensores sem fio. O restante desta seção mostra

	Baseados no emissor	Baseados no receptor
<b>Não Gossip</b>	<ul style="list-style-type: none"> <li>-<i>Directed Diffusion</i></li> <li>-<i>Trajectory Based Forwarding (TBF)</i></li> <li>-Multi</li> </ul>	<ul style="list-style-type: none"> <li>-<i>Flooding</i></li> <li>-<i>Sensor Protocols for Information via Negotiation (SPIN)</i></li> <li>-<i>Dynamic Delayed Broadcasting 1 (DDB1)</i></li> <li>-<i>Broadcast Protocol for Sensor (BPS)</i></li> </ul>
<b>Gossip</b>	<ul style="list-style-type: none"> <li>-<i>Epidemic gossip</i></li> <li>-<i>Epidemic Routing</i></li> <li>-<i>Epidemic Gossiping Considering Density</i></li> <li>-<i>Opportunistic Flooding</i></li> <li>-<i>Duty-Cycle-Aware Broadcast</i></li> </ul>	<ul style="list-style-type: none"> <li>-<i>Gossip1, Gossip2, Gossip3 e Gossip4</i></li> <li>-<i>Probabilistic scheme</i></li> <li>-<i>Trajectory and Energy-based Data Dissemination (TEDD)</i></li> <li>-<i>Dynamic Probabilistic Broadcasting</i></li> <li>-<i>Regional Gossiping</i></li> <li>-<i>Leveled Gossiping</i></li> <li>-<i>Dynamic Probabilistic Broadcasting 2 (DDB2)</i></li> </ul>
	<ul style="list-style-type: none"> <li>-<i>Flossiping</i></li> </ul>	

Tabela 3.2: Classificação de alguns protocolos de roteamento baseados no emissor / receptor e em pertencentes ou não à família *gossip*

os protocolos *Directed Diffusion* [62, 63], *Trajectory Based Forwarding* (TBF) [97, 99] e Multi [38].

O *Directed Diffusion* é um algoritmo de roteamento para RSSFs que visa estabelecer uma comunicação eficiente entre os sensores e o nó de monitor. Quando esse nó deseja que os sensores efetuem alguma tarefa de sensoriamento, ele envia para rede uma mensagem de interesse especificando o que ela deve sensoriar e reportar. Essa mensagem contém diversos atributos do evento a ser monitorado tais como o tipo do evento, área alvo e taxa de envio de dados. Alguns exemplos de interesses que podem ser desejados pelo nó monitor são “Qual é o número de pedestres transitando na região geográfica X nos próximos Z minutos?” ou “Qual é o sentido de movimento de um veículo A localizado em uma região B nos próximos C minutos?”. Em seguida, os nós sensores efetuam a disseminação do interesse utilizando o *flooding*. A rota de disseminação do interesse cria uma árvore que será utilizada para o caminho reverso dos dados sensorizados. Nesse caso, através de interações baseadas no emissor, quando um nó recebe um pacote em que ele é o escolhido para propagar, ele escolhe seu “pai” na árvore para ser o próximo da rota. Além disso, durante a coleta, o *Direct Diffusion* faz com que os nós emissores efetuem a agregação de dados para reduzir o tráfego na rede. O *Direct Diffusion* é aprimorado em [61, 63] em que os autores propõem uma abordagem mais eficiente em termos de energia para a agregação de dados. Na literatura, existem vários outros protocolos baseados no *Direct Diffusion* tais como os propostos em [96, 121].

O TBF é uma técnica de “roteamento em curva” para disseminar pacotes em redes sem fio densas. A inovação dessa abordagem é a definição e manipulação de rotas (trajetórias) como uma função contínua e não como um conjunto discreto de pontos. A ideia principal do TBF é inserir uma equação de curva (trajetória) no pacote e fazer com que cada nó intermediário decida qual será o próximo nó a propagar o pacote. Um nó intermediário toma essa decisão com base na distância de seus vizinhos em relação à trajetória contida no pacote. O TBF é um algoritmo baseado no emissor porque, sistematicamente, o nó corrente escolhe o próximo nó da rota. Essa escolha acontece de forma determinística, baseada na equação da trajetória, em uma tabela de vizinhos e em uma política pré-definida de propagação. Essa política pode ser, por exemplo, o vizinho mais próximo do destino ou o mais próximo da curva. O TBF faz com que os nós sensores mantenham suas tabelas de vizinhos atualizadas fazendo com que os nós vizinhos troquem periodicamente entre si um pacote especial denominado *beacon*. Quando um nó recebe um pacote de dados, ele verifica se é o escolhido para propagar o pacote. Se verdadeiro, ele escolhe o próximo nó, insere uma identificação sobre o próximo nó no cabeçalho do pacote e faz sua transmissão. Outra característica

do TBF é que ele é um algoritmo do tipo *source routing* uma vez que o nó origem determina todo o caminho pelo qual o pacote será roteado até o seu destino. Duas vantagens do TBF são a representação compacta da rota porque as curvas podem ser descritas com alguns parâmetros e a independência dos nós porque a trajetória não contém endereços específicos de nós.

O *Multi* é um protocolo de roteamento híbrido entre o roteamento pró-ativo e o reativo em RSSFs. A ideia principal desse protocolo é prever a necessidade de se criar e manter uma infraestrutura de roteamento como, por exemplo, a árvore de coleta. Quando acontecem muitos eventos de sensoriamento na rede, o *Multi* apresenta um comportamento pró-ativo no qual ele cria e atualiza a infraestrutura de roteamento. Por outro lado, quando existem poucos eventos de sensoriamento, o *Multi* apresenta um comportamento reativo em que a infraestrutura é criada sobre demanda. A principal vantagem desse protocolo é o compromisso com o custo de energia para manter a infraestrutura na abordagem pró-ativa e o tempo necessário para criá-la na abordagem reativa. O *Multi* é um algoritmo baseado no emissor uma vez que o roteamento é realizado sistematicamente de tal forma que cada nó emissor escolhe seu “pai” na árvore de coleta para continuar o roteamento.

### 3.2.2 Protocolos baseados no Emissor e *Gossip*

Os protocolos do tipo *gossip* baseados no emissor também são conhecidos como protocolos epidêmicos porque eles fazem com que os nós disseminem dados na rede de forma similar à disseminação de uma doença contagiosa. Nesses protocolos, quando um nó recebe um pacote de dados pela primeira vez, o nó escolhe aleatoriamente um subconjunto de seus vizinhos para enviar o pacote recebido. Uma das diferenças entre os protocolos propostos na literatura está na distribuição de probabilidade que cada um considera para efetuar essa escolha. Várias aplicações dentro da Ciência da Computação usam uma abordagem do tipo *gossip* baseado no emissor para a disseminação de dados. Por exemplo, para a replicação de banco de dados em [32], detecção de falhas [105], e o monitoramento de recursos [104]. A literatura sobre redes cabeadas e sem fio também usa essa técnica [74, 137]. O restante dessa seção descreve os seguintes protocolos: *Epidemic Gossip* [31, 32], *Epidemic Routing* [120], *Epidemic Gossiping Considering Density* [74], *Flossiping* [137], *Opportunistic Flooding(OF)* [43, 44] e *Duty-Cycle-Aware Broadcast* [123, 124].

Demers et al. [31, 32] apresentam o protocolo *Epidemic Gossip*, um *gossip* baseado no emissor, para resolver inconsistências entre os servidores de banco de dados da Xerox. O objetivo desse protocolo é distribuir de forma consistente atualizações e

*drives* para as réplicas do banco de dados. Um dos principais desafios na replicação de um banco de dados para vários nós é manter a consistência entre os nós. Os autores apresentam e avaliam duas versões do protocolo proposto: *Anti-entropy* e *Rumor mongering*. A primeira, baseada em [15], faz com que cada nó escolha aleatoriamente e sistematicamente outro nó para trocar conteúdos desatualizados entre suas bases de dados. Essa troca acontece de três formas: *push*, *pull* e *push-pull*. Na troca do tipo *push*, apenas o nó corrente envia atualizações para o outro, na *pull*, a atualização acontece no sentido inverso e, na *push-pull*, em ambos os sentidos. Nas três situações, somente conteúdos desatualizados são enviados. A vantagem da *Anti-entropy* é sua chance de infecção total da população. [100] mostra que se apenas um nó estiver “infectado”, o tempo esperado para que toda a população também esteja é proporcional ao logaritmo do tamanho da população. A constante de proporcionalidade depende do tipo de troca. Por exemplo, para a abordagem do tipo *push* considerando uma população com  $n > 1000$  elementos, o tempo esperado para infecção total da população é  $\log_2(n) + \ln(n) + O(1)$ . Na versão *Rumor mongering*, inicialmente, os nós são “ignorantes” e quando um nó recebe uma nova atualização, ele se torna um “hot rumor”. Enquanto ele tiver essa condição, ele escolhe aleatoriamente e periodicamente outro nó para enviar sua nova atualização. Quando no mínimo  $n$  vizinhos conhecerem a atualização, o nó considera que ela é antiga e pára de propagá-la. Os ciclos dessa técnica normalmente são mais frequentes que os da *Anti-entropy* porque exigem menos recursos em cada nó, contudo, normalmente, a chance de infecção total da população é menor.

O *Epidemic Routing* é um protocolo epidêmico para efetuar disseminações de dados do tipo *unicasting* em redes sem fio com topologia dinâmica tais como as redes móveis *ad hoc* e RSSFs. A ideia desse protocolo é baseada na troca *push-pull* da *Anti-entropy* e faz com que cada nó emissor escolha sistematicamente um conjunto aleatório de vizinhos para continuar o roteamento. Sua principal vantagem é que ele é capaz de entregar pacotes para um dado destino até mesmo quando não existe uma rota entre a origem e o destino. Os autores mostram resultados interessantes para a taxa de entrega, latência e consumo total de recursos. Contudo, em [48], quando a partição da rede se torna um evento raro, o *Epidemic Routing* não se apresenta como uma solução apropriada.

O *Epidemic Gossiping Considering Density* é um modelo analítico epidêmico para disseminação de dados em redes *ad hoc* e RSSFs. O modelo proposto considera a densidade da rede para calcular a taxa e o tempo total de infecção da população.

Zhang et al. [137] apresentam e avaliam o *Flossiping*, um protocolo de roteamento para RSSFs que combina as abordagens baseadas no emissor e no receptor do *gossip*.

Quando um nó tem que enviar um pacote, ele atua como no *gossip* baseado no emissor. O nó corrente escolhe aleatoriamente um subconjunto de vizinhos para continuar a disseminação, insere uma referência para cada um dos vizinhos escolhidos no cabeçalho do pacote e faz sua transmissão. Quando um nó recebe um pacote, ele verifica se é o escolhido como próximo nó. Se verdadeiro, o nó corrente atua como no *gossip* baseado no emissor e continua o roteamento. Caso contrário, ele atua como no *gossip* baseado no receptor e continua o roteamento com probabilidade  $p$ . Com probabilidade  $1 - p$ , o nó corrente descarta o pacote.

Guo et al. [43, 44] apresentam o *Opportunistic Flooding*(OF), um protocolo para difusão de dados em redes de sensores sem fio com ciclo de trabalho reduzido. A difusão de dados em RSSFs é uma tarefa desafiadora quando os nós operam com ciclo de trabalho reduzido (1% ou menos) porque quando um nó envia um pacote, nem todos os seus vizinhos estão acordados. A ideia do OF é criar uma árvore de difusão baseada na energia e na latência dos nós sensores e fazer com que cada nó emissor transforme cada uma de suas transmissões *broadcasting* em várias do tipo *unicasting*. Nesse caso, cada nó emissor agenda uma transmissão *unicasting* para o momento em que cada um de seus vizinhos estiver acordado. Contudo, para isso, o nó emissor tem que conhecer o escalonamento de trabalho dos seus vizinhos. Como o OF usa uma política MAC pró-ativa para compartilhar o escalonamento de trabalho dos nós vizinhos, este trabalho seguiu a proposta de seus autores para minimizar o custo de compartilhamento na qual cada nó sensor envia seu escalonamento somente para uma fração de vizinhos. Wang et al. [123, 124] apresentam o *Duty-Cycle-Aware Broadcast*, uma solução similar ao *Opportunistic Flooding*. Os dois protocolos são baseados no emissor porque cada nó interno escolhe o próximo nó. Além disso, eles pertencem a família *gossip* porque a construção de suas árvores considera variáveis aleatórias.

### 3.2.3 Protocolos baseados no Receptor e Não *Gossip*

Esta seção apresenta alguns protocolos de roteamento baseados no receptor e não pertencentes à família *gossip* para as redes de sensores sem fio. Nesse caso, quando um nó recebe um pacote pela primeira vez, ele decide deterministicamente se propaga o pacote. O restante desta seção mostra os protocolos *flooding*, *Sensor Protocols for Information via Negotiation* (SPIN) [54, 76], *Dynamic Delayed Broadcasting 1* (DDB1) [55] e *Broadcast Protocol for Sensor Networks* (BPS) [34].

O *flooding* ou inundação é a técnica ingênua de roteamento em redes cabeadas ou sem fio. O funcionamento básico desse algoritmo faz com que quando um nó recebe um pacote pela primeira vez, ele o propague para todos seus vizinhos. O *flooding* é um

protocolo baseado no receptor e não *gossip* porque cada nó decide por conta própria e de forma determinista se deve propagar cada pacote recebido. A desvantagem desse protocolo é seu número de pacotes duplicados [118] que o torna ineficiente em termos de consumo de energia.

O SPIN é uma família de protocolos adaptativos para disseminar dados em RSSFs. Ele considera que todos os nós sensores são monitores potenciais e, por isso, faz com que cada nó sensor dissemine suas observações para todos os sensores da rede. Essas observações são denominadas de metadados. A família de protocolos SPIN se baseia em uma negociação de dados e uma adaptação de recursos. Antes de um nó enviar sua observação para os demais, ele negocia com seus vizinhos para evitar a disseminação de observações redundantes na rede. O funcionamento básico do SPIN consiste em um *handshake* entre nós vizinhos dividido em três etapas. A primeira é uma etapa de divulgação que acontece quando um nó tem dados a serem enviados. Nesse caso, o nó envia uma mensagem de divulgação para informar aos seus vizinhos sobre seus dados. A segunda acontece quando um nó recebe essa mensagem e decide de forma determinística e por conta própria se deve propagá-la. Se verdadeiro, o nó receptor da mensagem de divulgação envia uma requisição ao emissor da mesma solicitando que esse envie os dados. A terceira acontece quando o emissor da divulgação recebe a requisição de seu vizinho e envia seus dados.

O DDB1 é um protocolo para a difusão de dados baseado na técnica de propagação com atraso. Nessa técnica, quando um nó recebe um pacote, ele aguarda um tempo. Após esse tempo, se outro vizinho não propagou o pacote, o nó receptor o propaga. O DDB1 faz com que o tempo do atraso seja inversamente proporcional a área adicional. Essa corresponde à região que será coberta pela transmissão do nó menos sua interseção com área anteriormente coberta pela disseminação. O objetivo do DDB1 é reduzir o número de transmissões da disseminação. Quando um nó recebe um pacote pela primeira vez, ele calcula de forma determinista sua área adicional. Se o tamanho dessa área for menor que um limiar predefinido, o nó descarta o pacote. Caso contrário, ele calcula e aguarda o tempo de atraso. Se o nó receber o mesmo pacote durante o tempo de atraso, ele recalcula sua área adicional. Se a nova área for menor que o limiar, o nó cancela a espera e descarta o pacote. Senão, o nó reajusta o atraso.

O BPS é outro protocolo de roteamento baseado na propagação com atraso. Esse protocolo usa a métrica de distância para calcular seu atraso. O BPS também minimiza o número de pacotes transmitidos em cada difusão. Quando um nó recebe um pacote, o nó calcula seu tempo de atraso a partir da coordenada geográfica do emissor do pacote. Nesse caso, o nó corrente considera que o nó anterior é o centro de um hexágono inscrito no raio de comunicação do anterior. Em seguida, o nó corrente

calcula sua distância em relação ao vértice mais próximo do hexágono. O tempo de atraso é inversamente proporcional a essa distância. Assim, um nó que se encontra mais próximo a algum vértice do hexágono terá um tempo de espera menor e será o primeiro a retransmitir o pacote. Observa-se que a decisão é baseada no receptor e acontece de forma determinística.

### 3.2.4 Protocolos baseados no Receptor *Gossip*

O *gossip* baseado no receptor é uma família de protocolos de roteamento para a difusão de dados em redes sem fio no qual quando um nó recebe um pacote, esse nó usa uma abordagem não determinística para decidir se deve propagar o pacote. A maioria dos trabalhos sobre *gossip* em MANETs e RSSFs consideram essa versão do *gossip*. Em [47, 48], os autores observam que a versão baseada no emissor não é uma abordagem interessante para rotear dados em redes com topologia dinâmica porque ela exige que cada nó intermediário escolha um subconjunto de seus vizinhos para propagar o pacote e isso nem sempre é possível devido à dinamicidade da rede. O restante desta seção mostra os seguintes protocolos: *Gossip1*, *Gossip2*, *Gossip3* e *Gossip4* [47, 48], *Probabilistic scheme and its variations* [98], *Dynamic Probabilistic Broadcasting* [135, 134], *Regional Gossiping* [81, 82], *Leveled Gossiping* [136], *Dynamic Delayed Broadcasting 2* (DDB2) [55] and *Trajectory and Energy Data Dissemination* (TEDD) [41, 86]. Destaca-se que alguns protocolos como, por exemplo, o DDB2 e TEDD, não abordam o termo *gossip*, mas na classificação desta tese podem ser considerados protocolos da família *gossip*.

O *Gossip* é um protocolo para a difusão de dados em redes sem fio. Nesse protocolo, quando um nó recebe um pacote, o nó decide de forma aleatória se propaga o pacote recebido. Essa decisão é baseada em probabilidade e, quando um nó recebe um pacote, o nó o propaga com probabilidade  $p$  e o descarta com probabilidade  $1 - p$ . Se um nó receber um mesmo pacote mais de uma vez, o nó apenas descarta o pacote repetido. O protocolo apresenta um comportamento bimodal em função da densidade da rede e da probabilidade utilizada. Se a rede for esparsa ou a probabilidade for pequena, as rotas são quebradas com facilidade e quase nenhum nó recebe o pacote. Por outro lado, em redes densas ou quando a probabilidade for suficiente, o protocolo apresenta um desempenho satisfatório em relação ao número de nós que recebe o pacote. Essa versão do *gossip* é chamada de *Gossip1(p)*. [47, 48] apresentam mais quatro abordagens para melhorar o desempenho do *gossip*. A primeira, *Gossip1(p, k)*, considera que quando a origem da difusão tem poucos vizinhos, existe uma chance não desprezível de quase nenhum nó receber o pacote. Assim, o *Gossip1(p, k)* faz com que

os primeiros  $k$  hops propaguem o pacote com probabilidade 1 e os demais,  $p$ . Outra abordagem (*Gossip2*) assume que a rede é irregular e os nós possuem números distintos de vizinhos. A ideia do *Gossip2* é fazer com que a probabilidade seja uma função do grau do nó de tal forma que nós com menos vizinhos possuem maior probabilidade. A abordagem seguinte (*Gossip3*) usa retransmissões para evitar o término prematuro da difusão. Essa versão funciona da mesma forma que o *Gossip1* exceto quando um nó não transmite o pacote. Nesse caso, em vez de descartar o pacote, o nó espera um tempo e se ele não tiver recebido o mesmo pacote de pelo menos  $m$  vizinhos, o nó propaga o pacote novamente. A última abordagem (*Gossip4*) organiza a rede em zonas como proposto em [45, 46] para o protocolo *Zone Routing Protocol* (ZRP). Nesse caso, cada nó  $u$  mantém uma zona que consiste em todos os nós localizados a no máximo  $\rho$  hops do nó  $u$ . Cada nó mantém de forma pró-ativa uma tabela de rotas para todos os nós de sua zona. Quando um nó recebe um pacote, ele verifica se o destino pertence a sua zona. Se verdadeiro, o nó propaga o pacote considerando a tabela de rotas, senão, ele trabalha como o *Gossip1*( $p, k$ ).

Ni et al. [98] apresentam cinco técnicas para reduzir a carga de trabalho do *flooding* em redes móveis *ad hoc*. A primeira, chamada de Probabilística é igual ao *Gossip1*. Na abordagem *Counter-based*, quando um nó recebe um pacote, ele espera um tempo aleatório para continuar a propagação. Após esse tempo, se o nó receber até  $c$  cópias do mesmo pacote, o nó propaga o pacote, senão, ele descarta. As abordagens *Distance-based* e *Location-based* são similares à anterior, contudo, elas consideram respectivamente a distância entre nós e a área adicional para um nó decidir se propaga ou descarta um pacote. A abordagem *Cluster-based* clusteriza a rede de tal forma que o *cluster head* seja vizinho de todos os demais nós de seu *cluster*. O roteamento entre os *clusters* acontece através nós *gateway* e usando uma das abordagens anteriores. As duas primeiras abordagens são naturalmente protocolos *gossip*; as abordagens *Distance-based* e *Location-based* também são dessa família porque a distância entre os nós e a área adicional são variáveis aleatórias que dependem da mobilidade dos nós. A abordagem *Cluster-based* também pertence porque usa as abordagens anteriores.

Zhang et al. [134, 135] observaram que o mecanismo de dois valores fixos de probabilidade existente no *Gossip2* [47, 48] nem sempre reflete a importância de um nó no processo de disseminação. Por exemplo, quando um nó que possui um único vizinho recebe um pacote, esse não deve propagá-lo porque seu vizinho o recebeu anteriormente. Por isso, os autores propõem o *Dynamic Probabilistic Gossiping*, um protocolo para difusão de dados em redes sem fio em que cada nó ajusta dinamicamente o valor de sua probabilidade de *gossip* de acordo com o número de pacotes recebidos. Apesar desse protocolo ser uma melhora da abordagem com duas probabilidades fixas, o *Dynamic*

*Probabilistic Gossiping* se limita ao pouco conhecimento sobre a “importância” de um nó no processo de difusão.

Li et al. [81, 82] propõem o *Regional Gossiping* para a disseminação de dados *unicasting* em redes sem fio. A ideia principal desse protocolo consiste em reduzir o número de transmissões do *gossip* no cenário de disseminação *unicasting*. O *Regional Gossiping* elimina as transmissões realizadas em regiões distantes da linha conectando os nós origem e destino. O protocolo considera que existe uma região entre os nós origem e destino na qual o pacote deve ser transmitido. Essa região pode ser, por exemplo, uma elipse em que os focos correspondem aos nós origem e destino. O funcionamento básico do *Regional Gossiping* é descrito a seguir. Quando um nó deseja enviar um pacote, ele insere as informações da região de transmissão dentro do pacote. Por exemplo, quando a região é uma elipse, o nó insere os focos dessa elipse no cabeçalho do pacote. Quando um nó recebe um pacote, ele verifica se sua coordenada  $(x, y)$  está localizada dentro da região de transmissão do pacote. Se verdadeiro, o nó propaga o pacote com probabilidade  $p$  e o descarta com probabilidade  $1 - p$ . Caso contrário, o nó também descarta o pacote.

O *Leveled Gossiping* é uma variação do *Gossip1(p)* que classifica os nós da rede em quatro grupos de acordo com o grau dos mesmos e atribui valores distintos de probabilidade para cada um dos grupos. Os nós com maior grau possuem menor probabilidade.

O DDB2 é uma versão especial do DDB1, apresentado na seção anterior como um protocolo não *gossip*. O funcionamento básico deles é similar e a diferença entre eles está no cálculo do tempo de atraso. No DDB1, esse valor é inversamente proporcional à área adicional – região que será coberta pela transmissão do nó menos sua interseção com área anteriormente coberta pela disseminação. No DDB2, o atraso é proporcional à subtração da energia inicial do nó por seu valor corrente. Consequentemente, os nós com mais energia normalmente propagam primeiro, balanceando o consumo de energia da rede e prolongando o tempo de vida da mesma quando comparado ao DDB1. Nas duas versões, quando um nó recebe um pacote pela primeira vez, ele calcula sua área adicional. Se o tamanho dessa área for menor que um limiar predefinido, o nó descarta o pacote. Caso contrário, ele calcula e aguarda o tempo de atraso. Se o nó receber o mesmo pacote durante o tempo de atraso, ele recalcula sua área adicional. Se a nova área for menor que o limiar, a espera será cancelada e o pacote descartado. Senão, o tempo de atraso será reajustado. O DDB2 é um protocolo da família *gossip* porque a energia de cada nó sensor é uma variável aleatória.

O TEDD é a dissertação de mestrado do autor desta tese e corresponde a um algoritmo que combina os conceitos de “roteamento em curva” com o mapa de energia [93]

(quantidade de energia disponível nas diversas partes da rede) para rotear dados de forma ciente da energia em RSSFs. O TEDD é composto de duas partes. A primeira é uma política de geração de curvas que passam por regiões com mais energia e evitam o roteamento dentro de regiões com menos energia. A ideia principal dessa política é escolher um subconjunto de nós com maior reserva de energia e fazer com que as curvas de roteamento passem por esses nós. A outra parte consiste em um mecanismo baseado na técnica de propagação com atraso para que cada nó decida localmente se participa do roteamento. O processo de decisão considera a distância entre o nó corrente e um ponto de referência sobre a curva para efetuar a decisão. O TEDD é um protocolo da família *gossip* porque sua geração de curvas depende da energia dos nós sensores que é uma variável aleatória.

### 3.3 Conclusões

Este capítulo apresenta protocolos para a comunicação de dados em redes de sensores sem fio. Como as camadas de rede e de acesso ao meio (MAC) são responsáveis pela comunicação nesse tipo de rede, dividiu-se esses protocolos por suas camadas. Os protocolos da camada MAC estão divididos em pró-ativos e reativos. A principal diferença entre os protocolos pró-ativos apresentados é a forma como eles compartilham informações sobre vizinhos e a entre os reativos está em suas técnicas de descoberta. Os protocolos da camada de rede foram divididos em baseados no emissor e no receptor e, também, entre os pertencentes ou não à família de protocolos *Gossip*.

Esta tese apresenta três soluções para minimizar os custos de compartilhamento e descoberta das políticas MAC pró-ativa e reativa. A primeira solução é uma abordagem híbrida entre as duas políticas existentes e o autor desconhece outros trabalhos que combinem as duas políticas. A segunda solução consiste em um projeto combinado entre a política reativa e o roteamento baseado no receptor no qual a diversidade de candidatos a próximo nó desse tipo de roteamento minimiza o custo da descoberta reativa. Todas as soluções MAC reativas conhecidas pelo autor desta tese consideram o roteamento baseado no emissor no qual normalmente tem-se um único candidato a próximo nó. A última solução é um projeto integrado pertence à família *Gossip* que envolve as camadas de rede e MAC para efetuar difusão de dados quando os nós sensores ficam quase sempre dormindo. A ideia dessa solução é minimizar o custo de compartilhamento MAC pró-ativo existente nas soluções para esse tipo de difusão de dados. As soluções existentes para esse tipo de comunicação normalmente fazem com que cada nó compartilhe seu escalonamento de trabalho com todos seus vizinhos.

## Capítulo 4

# Política Híbrida para Determinar o Escalonamento de Trabalho dos Nós Vizinhos

*Se apenas houvesse uma única verdade, não poderiam pintar-se cem telas sobre o mesmo tema.*  
Pablo Picasso

Este capítulo propõe e avalia uma política híbrida para determinar o escalonamento de trabalho dos nós vizinhos e ele está organizado como descrito a seguir. A seção 4.1 introduz a solução proposta e mostra sua motivação e objetivos. A seção 4.2 descreve a política híbrida. A seção 4.3 apresenta os principais resultados de simulação da solução proposta. A seção 4.4 apresenta as conclusões deste capítulo.

### 4.1 Introdução

As tarefas de comunicação de dados em RSSFs podem fazer com que dois nós sensores troquem seguidas mensagens em um intervalo de tempo reduzido (e.g., em menos de um segundo). As chances de dois nós se comunicarem mais de uma vez em um intervalo reduzido depende da aplicação. Isso porque algumas aplicações são baseadas na repetição de atividades recentes de comunicação. Por exemplo, em uma coleta de dados na qual os nós sensores reportam dados de forma contínua para o nó monitor como a temperatura de uma região ou um evento com baixa mobilidade em relação ao raio de sensoriamento. Outra aplicação acontece quando o nó monitor tem que enviar requisições periódicas para os nós sensores localizados em uma mesma região. Nesse caso, o nó monitor pode solicitar periodicamente a alguns nós para alterarem

seus respectivos modos de operação. Um terceiro exemplo acontece nas aplicações em que o nó monitor envia requisições e os sensores as respondem.

Uma informação recentemente compartilhada sobre o escalonamento de trabalho pode ser necessária quando os nós sensores envolvidos no compartilhamento se comunicam frequentemente em um intervalo reduzido. Essa informação é fundamental em qualquer tarefa de comunicação em RSSFs porque cada nó emissor deve efetuar sua transmissão quando supostamente o próximo nó estiver acordado. Para isso, o emissor deve conhecer o escalonamento de trabalho do próximo nó.

As políticas MAC para determinar o escalonamento de trabalho dos nós vizinhos existentes na literatura [75, 83, 87] podem ser divididas em pró-ativa e reativa. A primeira faz com que cada nó sensor compartilhe seu escalonamento de trabalho com todos seus vizinhos, permitindo que cada nó emissor agende sua transmissão para o momento em que o próximo nó estiver acordado. Contudo, esse compartilhamento é tipicamente baseado em troca de mensagens, o que aumenta o consumo de energia da rede. A segunda faz com que cada nó sensor descubra o escalonamento do próximo nó somente quando essa informação for necessária. Nesse caso, o nó emissor normalmente efetua retransmissões de um quadro de descoberta até que o próximo nó envie uma resposta informando que está acordado. Apesar da abordagem reativa eliminar o custo de compartilhamento, as retransmissões da descoberta representam um consumo extra de energia. A escolha da política a ser usada em um protocolo MAC deve considerar a relação custo-benefício entre manter a informação sobre o escalonamento de trabalho dos vizinhos atualizada, obtê-la quando necessária e a frequência com que ela será utilizada. Dadas as limitações das duas políticas, novas soluções devem ser investigadas.

Uma técnica usada para acessar dados recentemente usados é a memória *cache* cujo uso pode ser investigado nas políticas para determinar o escalonamento de trabalho de nós vizinhos. O uso da técnica de *cache* na obtenção dessa informação sobre vizinhos cria uma solução híbrida entre as políticas MAC pró-ativa e reativa. A ideia é que cada nó emissor insira seu escalonamento no quadro e cada receptor use essa informação para criar e atualizar uma tabela contendo o escalonamento de trabalhos de seus vizinhos. Na solução híbrida, diferente da pró-ativa, os nós sensores determinam a informação sobre o escalonamento dos seus vizinhos sem qualquer troca extra de mensagens, o que elimina o custo de compartilhamento pró-ativo. Quando um nó envia um quadro, ele verifica se conhece o escalonamento do próximo nó. Se conhecer, como na política MAC pró-ativa, o nó emissor agenda sua transmissão para o momento no qual o próximo nó estiver acordado. Caso contrário, a política híbrida atua como a reativa e faz com que o nó emissor inicie uma descoberta pelo escalonamento do próximo nó. Quando o nó emissor descobre essa informação, ele atualiza sua tabela e efetua sua transmissão. Na

política híbrida, ao contrário da reativa, as transmissões futuras usam tal informação para evitar o custo reativo de novas descobertas. Todavia, uma desvantagem da solução híbrida é que cada nó emissor insere um campo extra no cabeçalho dos quadros.

Este capítulo apresenta e avalia a política híbrida para determinar o escalonamento de trabalho dos nós vizinhos. O objetivo dessa política é minimizar o custo de compartilhamento das abordagens pró-ativas e o de descoberta das reativas. Resultados de simulação revelam seu efeito em protocolos MAC pró-ativo e reativo. A solução proposta reduz o número de transmissões e o consumo de energia das políticas existentes na maioria dos cenários avaliados. Em particular, esses resultados são mais expressivos nos cenários nos quais os nós sensores operavam com um ciclo de trabalho reduzido (1% ou menos) ou dinâmico.

## 4.2 Política Híbrida

Esta seção propõe a política híbrida e está organizada como descrito a seguir. A seção 4.2.1 apresenta o funcionamento básico da política proposta e a seção 4.2.2 descreve os parâmetros de escalonamento que cada emissor insere no quadro. A descoberta da política híbrida é baseada no mecanismo da política reativa que acontece através de retransmissões periódicas de quadros de descoberta. Como tais retransmissões apresentam um custo de energia não desprezível, a seção 4.2.3 define o número máximo e o esperado de retransmissões desses quadros.

### 4.2.1 Funcionamento Básico

O funcionamento básico da política híbrida pode ser modelado a partir de eventos e ações de tal forma que, quando um evento acontece, ocorre uma ou mais ações correspondentes. Os eventos considerados nesta política são “enviar quadro”, “receber quadro” e “descobrir escalonamento de trabalho”. As figuras 4.1 a 4.3 mostram as ações da política híbrida para cada um desses eventos.

A geração do evento “enviar quadro” acontece através de requisições da camada de rede quando um nó tiver que enviar uma mensagem. A ação correspondente a esse evento faz com que o nó corrente verifique se ele conhece o escalonamento do próximo nó (figura 4.1, ponto A). Como a política pró-ativa, a híbrida considera que o protocolo MAC é responsável por identificar se a informação sobre o escalonamento de trabalho de um vizinho está desatualizada e, conseqüentemente, excluir essa informação da tabela de escalonamentos. O protocolo MAC pode fazer essa identificação quando, por exemplo, enviar quadros para um nó que não responde. Assim, quando o nó corrente

conhece o escalonamento de trabalho do próximo nó (figura 4.1, ponto *A*), ele verifica se o próximo nó está acordado (figura 4.1, ponto *B*). Se verdadeiro, o nó corrente insere seu escalonamento de trabalho no quadro e o envia (figura 4.1, pontos *C* e *D*), senão, ele reescalona o evento “enviar quadro” para o momento em que o próximo nó estiver acordado (figura 4.1, ponto *F*). Esse conjunto de ações termina após as etapas representadas pelos pontos *D* e *F* (figura 4.1, ponto *E*). Por outro lado, quando o nó corrente não conhece o escalonamento de trabalho do próximo nó (figura 4.1, ponto *A*), a política híbrida funciona como a política MAC reativa que executa a ação de descoberta de escalonamento (figura 4.1, ponto *G*) e, em seguida, termina a ação corrente (figura 4.1, ponto *E*).

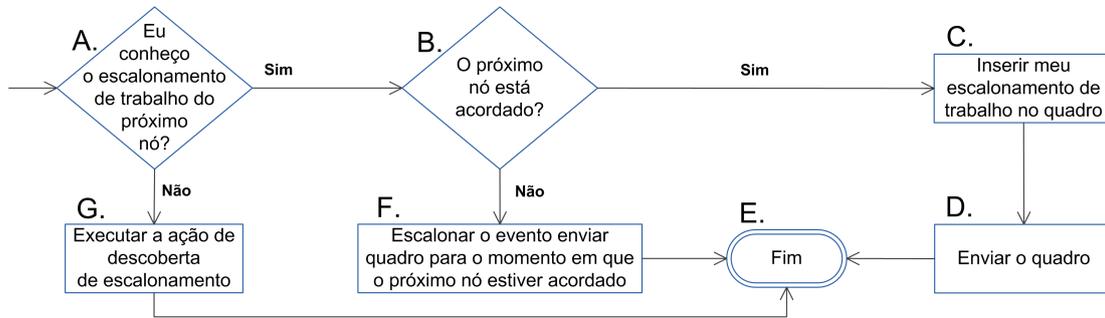


Figura 4.1: Ação correspondente ao evento enviar quadro da política híbrida

A ação de descoberta de escalonamento faz com que o nó corrente retransmita periodicamente um quadro de descoberta. O nó retransmite esse quadro até receber uma resposta do vizinho cujo escalonamento é desconhecido ou alcançar o número máximo de retransmissões. O primeiro passo da descoberta faz com que o nó corrente crie e envie um quadro de descoberta contendo seu escalonamento de trabalho (figura 4.2, pontos *A* e *B*). Em seguida, a política híbrida funciona como a política MAC reativa e faz com que o nó corrente incremente um contador de retransmissões para a descoberta em questão (figura 4.2, ponto *C*) e verifique se ele terá outra chance de descobrir o escalonamento do próximo nó (figura 4.2, ponto *D*). O número de chances corresponde ao número máximo de retransmissões. Se o contador de retransmissões for menor que o máximo, o nó escala a descoberta de escalonamento (figura 4.2, ponto *E*), caso contrário, ele termina a descoberta sem sucesso (figura 4.2, ponto *F*). O sucesso da descoberta acontece quando o nó corrente recebe uma resposta do próximo nó. Quando isso acontecer, a ação correspondente do evento “receber quadro” cancela o escalonamento do evento “descobrir escalonamento de trabalho” para terminar a descoberta.

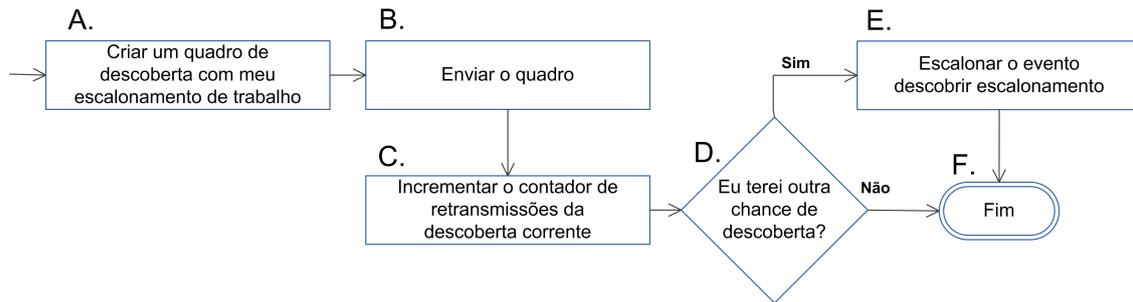


Figura 4.2: Ação correspondente ao evento descoberta de escalonamento da política híbrida

A ação correspondente ao evento “receber quadro” faz com que o nó corrente atualize sua informação sobre o escalonamento de trabalho do nó emissor (figura 4.3, ponto *A*). Em seguida, a política híbrida trabalha exatamente como a política reativa e faz com que o nó corrente verifique se ele é o próximo nó do quadro recebido (figura 4.3, ponto *B*). Observa-se que o fato do nó corrente atualizar sua tabela de escalonamentos antes de verificar se ele é o próximo nó do quadro permite que todos os nós receptores de um quadro atualizem suas tabelas, evitando eventuais descobertas futuras. Se o nó corrente for o próximo nó, ele verifica se o quadro recebido é de descoberta (figura 4.3, ponto *C*). Nesse caso, o emissor deseja saber o escalonamento de trabalho do nó corrente e, por isso, esse nó envia um quadro de resposta com a informação desejada (figura 4.3, ponto *D*). Caso o quadro recebido não seja de descoberta (figura 4.3, ponto *C*), o nó corrente verifica se é de resposta (figura 4.3, ponto *F*). Se for o caso, significa que o nó corrente iniciou uma descoberta e recebeu o escalonamento de trabalho desejado. Assim, o nó corrente escalonar a transmissão do quadro desejado, de acordo com o escalonamento descoberto (figura 4.3, ponto *G*) e cancela a descoberta pendente (figura 4.3, ponto *H*) porque a mesma terminou com sucesso. Quando o quadro não é de descoberta nem de resposta (figura 4.3, ponto *F*), o nó corrente apenas recebe o quadro a ser processado por outras rotinas da camada MAC (figura 4.3, ponto *I*). Quando o nó corrente não é o próximo nó do quadro recebido (figura 4.3, ponto *B*), a ação termina (figura 4.3, ponto *E*).

As ações da política híbrida efetuam um chaveamento dinâmico entre as políticas MAC pró-ativa e reativa. O comportamento inicial é similar ao da política reativa em que não existe troca extra de quadros. Outra característica da política reativa é a existência de um mecanismo de descoberta. Contudo, a política híbrida utiliza esse mecanismo apenas quando o nó corrente desconhece o escalonamento de trabalho do próximo nó. O fato da política híbrida criar e atualizar uma tabela com o escalonamento

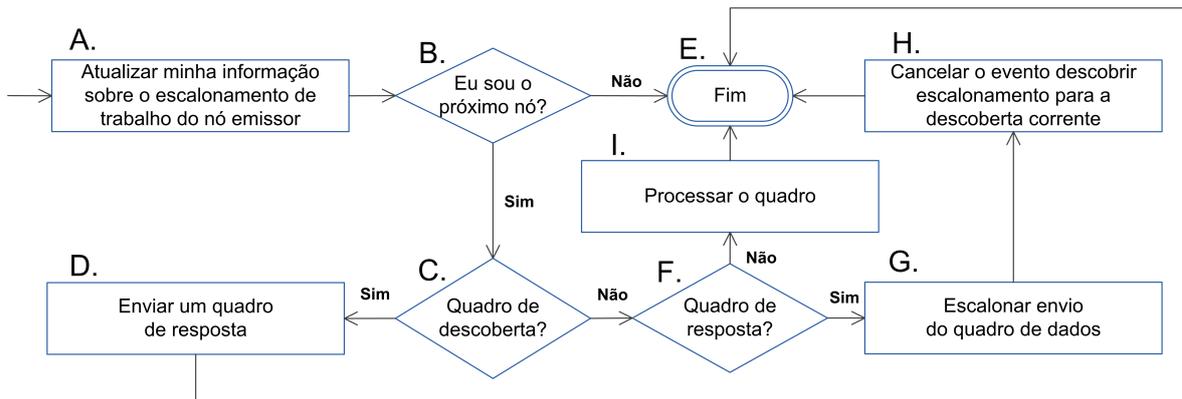


Figura 4.3: Ação correspondente ao evento receber quadro da política híbrida

dos nós vizinhos mostra um comportamento pró-ativo, contudo, sem a troca extra de quadros.

Da mesma forma que as políticas pró-ativa e reativa, a política híbrida considera que os atrasos na comunicação devem ser tratados pela política de acesso ao meio. Por exemplo, o projetista da rede deve ter uma atenção especial ao fato de que quando um nó sensor tiver que enviar uma mensagem, o nó aguarda pelo momento em que o próximo nó estiver acordado para efetuar sua transmissão. Contudo, quando isso acontecer, o nó emissor também deverá aguardar pelo acesso ao meio sem fio que eventualmente pode ser obtido quando o próximo nó estiver adormecido novamente.

#### 4.2.2 Parâmetros sobre Escalonamento de Trabalho Inseridos nos Quadros

A definição dos parâmetros inseridos no quadro é baseada em três suposições. A primeira é que o tempo de vida dos nós deve ser dividido em ciclos operacionais contínuos, de tamanho fixo e com o período acordado sempre antecedendo o de dormir. Consequentemente, um nó conhece seu escalonamento de trabalho (ou o de um vizinho) tendo apenas o ciclo de trabalho e a posição no ciclo operacional. A posição de um nó em seu ciclo operacional corresponde à situação atual de um nó (em porcentagem) no seu ciclo operacional. Por exemplo, a figura 4.4 ilustra o ciclo operacional de dois nós sensores A e B em que o período acordado (cor branca) equivale a 10% e 70% dos respectivos ciclos operacionais. O período em que o nó dorme é ilustrado pela cor cinza. Nesse exemplo, o nó A está acordado se sua posição estiver entre 0% e 10%. Da mesma forma, B está acordado se sua posição estiver entre 0% e 70%. A segunda suposição é que a posição dos nós sensores em seus ciclos operacionais é assíncrona porque a aplicação pode ativá-los em momentos distintos. A terceira suposição é que um

nó adormecido acorda apenas conforme seu escalonamento de trabalho ou para efetuar uma transmissão oportunista [44]. Esse tipo de comunicação normalmente acontece quando um nó sensor tem que se comunicar com outro que fica acordado somente quando o nó em questão está adormecido. Na comunicação oportunista, o nó emissor acorda, transmite e retorna ao seu escalonamento de trabalho. Um exemplo em que a comunicação oportunista será necessária acontece quando o nó A da figura 4.4 tiver que enviar um quadro para B.

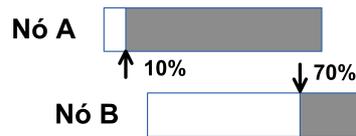


Figura 4.4: Exemplo do ciclo operacional de dois nós sensores

Dadas as suposições apresentadas acima, os parâmetros inseridos por cada nó emissor no quadro são seu ciclo de trabalho e sua posição no ciclo operacional. Essas informações permitem aos receptores conhecerem o escalonamento de trabalho do nó emissor, que pode ser usado em transmissões futuras. Contudo, como a posição de um nó em seu ciclo operacional muda ao longo do tempo, os receptores usam a posição do emissor para calcular e armazenar sua diferença de ciclo para esse nó. Essa diferença não muda durante o tempo de vida da rede, exceto com desatualizações. Um nó receptor calcula sua diferença de ciclo para o emissor a partir da equação (4.1), onde  $Receptor_{pos}$  e  $Emissor_{pos}$  são, respectivamente, as posições do receptor e do emissor em seus ciclos operacionais no fim da transmissão. Como a subtração das posições pode ser negativa e a posição é cíclica, a equação (4.1) adiciona um ciclo operacional à subtração se seu resultado for negativo. Por exemplo, se os nós A e B da figura 4.4 estiverem respectivamente nas posições 28% e 5% quando B receber um quadro de A, a diferença de B para A será  $5\% - 28\% + 100\% = 77\%$ , pois  $Receptor_{pos} < Emissor_{pos}$ . Se eles estiverem nas mesmas posições e A receber um quadro de B, a diferença de A para B será  $28\% - 5\% = 23\%$ , pois  $Receptor_{pos} \geq Emissor_{pos}$ . Assim, se A souber que sua diferença para B é de 23% e que o ciclo de trabalho de B é de 70%, quando A tiver que enviar um quadro para B, A envia quando sua posição estiver entre 23% e 93%. Da mesma forma, se B souber que sua diferença para A é de 77% e que o ciclo de trabalho de A é de 10%, B envia quando sua posição estiver entre 77% e 87%. Observa-se que, quando um nó recebe um quadro, esse nó ajusta o valor da posição atual do nó emissor considerando a diferença de tempo entre o momento em que o emissor inseriu essa informação no quadro e o momento no qual o receptor recebeu e

processou o quadro.

$$diferença = \begin{cases} Receptor_{pos} - Emissor_{pos}, & \text{se } Receptor_{pos} \geq Emissor_{pos} \\ Receptor_{pos} - Emissor_{pos} + 100\%, & \text{caso contrário.} \end{cases} \quad (4.1)$$

A desvantagem da inserção de parâmetros em um quadro é o aumento do seu tamanho que minimiza os ganhos de energia da política híbrida. O tamanho de um quadro em RSSFs depende do protocolo MAC, que normalmente está entre 20 e 60 bytes [28, 30, 107, 108, 114, 131]. Os dois parâmetros inseridos no quadro podem ser representados por três bytes: um byte para o ciclo de trabalho e dois para a posição do nó em seu ciclo operacional. Uma forma de minimizar essa desvantagem é explorar técnicas de codificação ou reduzir o número de quadros contendo esse *overhead*.

### 4.2.3 Retransmissão dos Quadros de Descoberta

A retransmissão de quadros de descoberta apresenta um custo de energia não desprezível. Esta seção define o número máximo e o intervalo entre elas e mostra uma definição analítica para o número esperado de retransmissões desses quadros.

A política híbrida define o número máximo e o intervalo entre as retransmissões dos quadros de descoberta com os objetivos de garantir que o próximo nó estará acordado em pelo menos uma das retransmissões e de minimizar o número de retransmissões na descoberta. Para isso, a política proposta escalona as retransmissões obedecendo a duas condições. Primeiro, o tempo total da descoberta (o intervalo entre a primeira e a última retransmissão) deve ser maior ou igual ao período adormecido, garantindo que o próximo nó sempre acorda durante a descoberta. Segundo, o intervalo entre as retransmissões deve ser menor ou igual ao período acordado para garantir que quando o próximo nó acordar, o emissor fará pelo menos uma retransmissão antes que o próximo nó adormeça novamente. Dado que o próximo nó sempre acorda durante o processo de descoberta e que quando ele está acordado o emissor sempre faz pelo menos uma retransmissão, a política híbrida garante que o próximo nó estará acordado em pelo menos uma das retransmissões da descoberta. Entretanto, eventualmente, algum vizinho pode não receber um quadro devido a colisões ou alguma alteração em seu escalonamento de trabalho. Observa-se que se o ciclo de trabalho da rede for homogêneo e estático, o nó corrente considera seus próprios valores de períodos acordado e adormecido para efetuar as definições anteriores. Caso contrário, ele considera o menor valor do ciclo de trabalho que o projetista da rede definiu para os nós sensores. Nesse caso, assume-se que todos os nós sensores conhecem a priori esse valor.

A política híbrida minimiza o número de retransmissões, maximiza o intervalo entre elas e minimiza o tempo total da descoberta. Para isso, ela faz com que o tempo total da descoberta e o intervalo entre as retransmissões sejam iguais aos períodos adormecido e acordado, respectivamente. O número total de retransmissões é o teto da divisão do período adormecido pelo acordado.

A definição analítica do número esperado de retransmissões na descoberta deve considerar os motivos pelos quais um nó envia um quadro de descoberta e não recebe resposta do próximo nó. Um motivo é o fato do próximo nó estar adormecido durante a transmissão. Outros são as colisões e os erros físicos de transmissão. Este trabalho modela esses outros motivos como uma variável aleatória com probabilidade constante  $p_E$  e com distribuição uniforme durante cada ciclo operacional. Por outro lado, a probabilidade do próximo nó estar adormecido reduz à medida que o nó emissor efetua suas retransmissões. Isso porque, como mostrado anteriormente, o próximo nó acorda pelo menos uma vez durante a descoberta. Assim, à medida que o nó emissor efetua as retransmissões e o próximo nó continua adormecido, a última retransmissão se aproxima, o que aumenta a probabilidade desse nó acordar. A equação (4.2) calcula a probabilidade do próximo nó estar dormindo na  $i$ -ésima retransmissão em que  $a$  é o tamanho do período acordado,  $d$  é o tamanho do período adormecido,  $(d + a)$  é o tamanho de um ciclo operacional e  $(i \times a)$  é o tempo que o próximo nó dormiu entre a primeira e a  $i$ -ésima transmissão.

$$dormir(i) = \frac{d - (i \times a)}{(d + a) - (i \times a)}. \quad (4.2)$$

O número esperado de retransmissões depende da probabilidade do nó emissor receber uma resposta nas retransmissões anteriores. A equação (4.3) calcula a probabilidade de um nó não receber resposta na  $i$ -ésima retransmissão dada a probabilidade do próximo nó estar adormecido nessa retransmissão e a de outros motivos acontecerem em um ciclo operacional. A equação (4.4) calcula a probabilidade do nó corrente receber a resposta na  $i$ -ésima retransmissão dada a probabilidade do próximo nó estar acordado  $(1 - dormir(i))$ , a de outros motivos não acontecerem em um ciclo operacional  $(1 - p_E)$  e a do nó emissor não ter recebido resposta nos  $i - 1$  envios anteriores. O número esperado de retransmissões é obtido através do somatório apresentado na equação (4.5) onde  $\lceil d/a \rceil$  é o número máximo de retransmissões.

$$semResposta(i) = dormir(i) + (1 - dormir(i)) \times p_E. \quad (4.3)$$

$$resposta(i) = (1 - dormir(i)) \times (1 - p_E) \times \prod_{k=0}^{i-1} semResposta(k). \quad (4.4)$$

$$E[R] = \sum_{i=0}^{\lceil d/a \rceil} i \times resposta(i). \quad (4.5)$$

### 4.3 Resultados de Simulação

Esta seção avalia a política híbrida em versões adaptadas de protocolos MAC pró-ativo e reativo, tendo como aplicação a disseminação de dados em que o nó monitor envia dados periodicamente para nós sensores localizados nas mesmas regiões. Esse tipo de comunicação é fundamental em RSSFs, pois o nó monitor realiza tarefas específicas tais como alterar o modo de funcionamento de nós sensores, ativar e desativar um ou mais nós e enviar requisições ou interesses para eles. A seção 4.3.1 mostra os parâmetros de simulação e as seções 4.3.2 e 4.3.3 avaliam os efeitos da política proposta em versões adaptadas de protocolos MAC pró-ativo e reativo, respectivamente.

#### 4.3.1 Parâmetros de Simulação

Todos os protocolos foram implementados no *Network Simulator 2* [2] e os resultados de simulação correspondem à média aritmética de  $n$  simulações, onde  $n$  é o menor tamanho de amostra que proporciona o intervalo de confiança desejado. O nível de confiança é de 95% e utilizou-se o *teste-T* [64] com 0,05 de significância para afirmar que um protocolo é melhor ou pior que outro para uma dada métrica.

Em todas as simulações, utilizou-se uma rede com 500 nós estáticos, homogêneos e com capacidade de energia finita e não renovável. O consumo de energia de cada nó é baseado no Mica2 [1], sua energia inicial foi suficiente para que nenhum nó morresse por falta de energia durante a simulação e seu raio de comunicação é de 100 m. Os nós foram depositados de forma aleatória em uma área  $1000 \times 1000 \text{ m}^2$  e considerou-se que cada nó conhece sua localização [8]. Além disso, a topologia da rede pode ser representada por um grafo conexo com graus mínimo, médio e máximo iguais a 5, 13 e 25, respectivamente. A rede também contém um nó monitor sem restrição de recursos, localizado no canto inferior esquerdo da rede e cujo ciclo de trabalho é de 100%.

Este trabalho avalia os efeitos da política híbrida no protocolo pró-ativo *Opportunistic MAC* (OMAC) [43, 44] e no reativo *Asynchronous MAC* (AMAC) [83]. A principal alteração para a versão híbrida do OMAC é a remoção do compartilhamento e a inserção das descobertas. Para a versão híbrida do AMAC, são as inserções da

comunicação oportunista e da tabela contendo informações sobre o escalonamento de trabalho dos vizinhos. Nos dois protocolos híbridos, cada nó emissor insere seus parâmetros de escalonamento somente nos quadros RTS e CTS. Isso porque seus receptores são praticamente os mesmos dos demais quadros (dados e ACK). A inserção de parâmetros nos quadros de dados e ACK aumentaria o consumo de energia e seria redundante para os receptores. O tamanho dos quadros RTS e CTS não híbridos é de 20 bytes. Como o aumento do tamanho dos quadros influencia o consumo de energia, as simulações deste trabalho consideram que o tamanho dos quadros RTS e CTS híbridos é 1,15 vezes maior que o dos não híbridos. Esse aumento corresponde aos três bytes necessários para o *overhead* híbrido. Além disso, o tamanho dos quadros RTS e CTS não híbridos é igual ao dos quadros ACK e de compartilhamento (usado no protocolo pró-ativo) e o tamanho dos quadros de dados é o dobro desse valor. Todos os protocolos foram configurados com o mesmo tamanho de ciclo operacional e a posição inicial de cada nó em seu ciclo operacional é uma variável aleatória com distribuição uniforme.

Utilizou-se o protocolo *Trajectory Based Forwarding* [99] na camada de rede porque ele é um protocolo de roteamento eficiente em termos de energia e concebido para a disseminação de dados em RSSFs. As figuras 4.5 e 4.6 mostram as combinações de protocolos das camadas de rede (superior) e MAC (inferior) usados nas duas seções de resultados descritas abaixo.

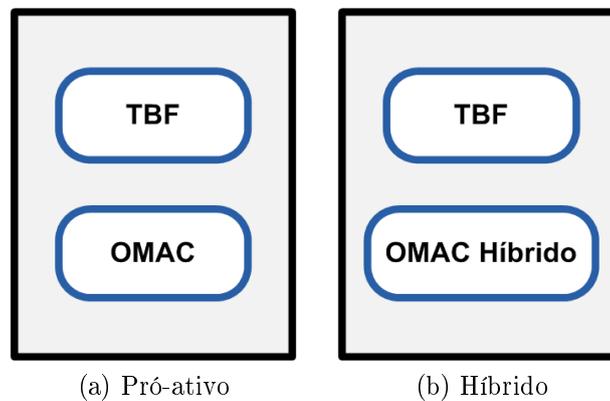


Figura 4.5: Protocolos usados para avaliar o efeito da política híbrida em um protocolo MAC pró-ativo

Nos cenários avaliados, o nó monitor executa uma série de disseminações *unicasting*. O nó destino de cada disseminação é escolhido aleatoriamente em três regiões circulares com centros equidistantes do nó monitor e raio igual ao de comunicação. A figura 4.7 ilustra essas três regiões representadas em cinza. O tempo de simulação é de 200 s. Porém, a primeira disseminação acontece após 10 s, para que os protocolos

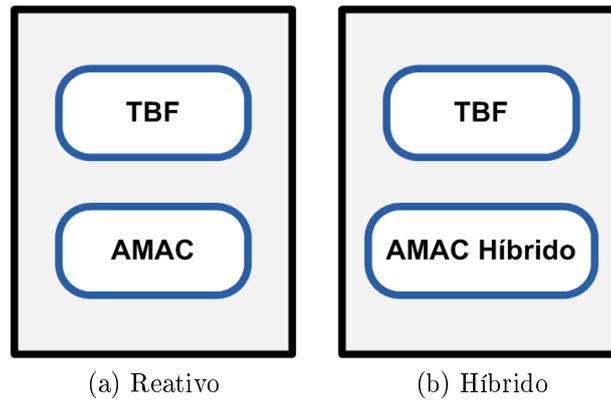


Figura 4.6: Protocolos usados para avaliar o efeito da política híbrida em um protocolo MAC reativo

sejam inicializados, e a última acontece antes de 150 s, para que todas as mensagens disseminadas pelo nó monitor sejam roteadas.

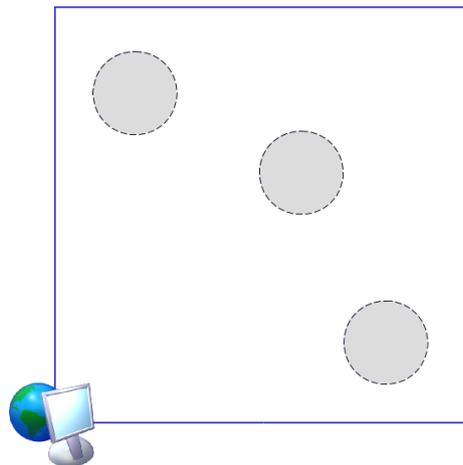


Figura 4.7: Regiões em que os destinos estão localizados

Este trabalho considera dois cenários de simulação: um em que os nós sensores operam com um ciclo de trabalho estático e outro em que esse parâmetro é dinâmico. No primeiro, o ciclo de trabalho dos nós não se modifica ao longo do tempo de vida da rede, o que favorece a política MAC pró-ativa que compartilha informações. Todos os protocolos foram avaliados com os valores estáticos de 1% e 80% para o ciclo de trabalho. Além disso, nesse cenário, o nó monitor disseminou 500 mensagens de dados gerados por uma distribuição de probabilidade uniforme.

No cenário com ciclo dinâmico, os nós sensores possuem um comportamento adaptativo no qual modificam seus respectivos ciclos de trabalho ao longo do tempo de vida da rede. Isso favorece a política MAC reativa porque ela não compartilha informações

sobre vizinhos que, no cenário com ciclo dinâmico, desatualizam frequentemente. O cenário proposto considera um fluxo de dados que varia ao longo da simulação e faz com que os nós sensores que participarem da disseminação adaptem seu ciclo de trabalho para prover suporte às variações no tráfego de dados. O nó monitor dissemina 1000 mensagens de dados dos quais ele gera metade a partir da distribuição de probabilidade uniforme e a outra metade a partir da distribuição Gaussiana, com média 100 e desvio padrão 20. O ciclo de trabalho inicial dos nós sensores é de 1% e, quando um nó envia uma mensagem de dados, ele atualiza seu ciclo de trabalho de acordo com um valor definido pelo nó monitor em cada mensagem de dados. Como o nó monitor gera o tráfego de dados, este trabalho assume que ele pode solicitar que os nós sensores atualizem o ciclo de trabalho, provendo suporte às flutuações do tráfego de dados. A figura 4.8 mostra o número esperado de disseminações por segundo que serão iniciadas pelo nó monitor e o ciclo de trabalho esperado para os nós sensores que propagarem essas disseminações no cenário com ciclo de trabalho dinâmico. O eixo  $y$  à esquerda da figura indica o primeiro valor e o da direita, o outro. As disseminações acontecem entre 10 s e 150 s, que correspondem aos tempos da primeira e da última disseminações. O número de disseminações e o ciclo de trabalho aumentam após 40 s e reduzem após 100 s de acordo com a geração Gaussiana de transmissões.

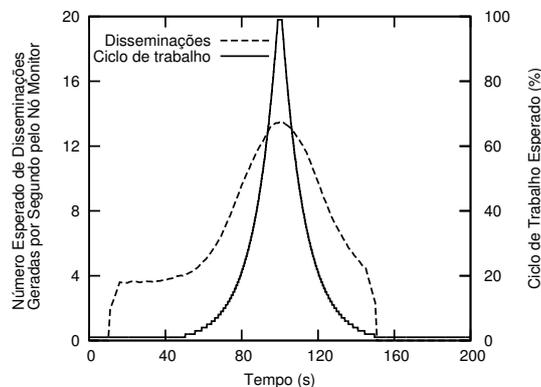


Figura 4.8: Número esperado de disseminações e ciclo de trabalho esperado para o segundo cenário de simulação durante o tempo de vida da rede

### 4.3.2 Política Híbrida com Protocolo MAC Pró-ativo

Esta seção avalia a influência da política híbrida no protocolo OMAC em cenários de disseminação de dados nos quais os nós sensores operam com ciclos de trabalho estático e dinâmico. As métricas avaliadas são a taxa média de entrega, os números de transmissões de quadros e bytes e a latência. A taxa média de entrega significa a

porcentagem de mensagens entregues aos seus destinos e ela é fundamental porque as mensagens podem ser perdidas durante a disseminação. Essa porcentagem esta relacionada ao número total de mensagens enviadas até o final da simulação. Os números de transmissões de quadros e de bytes indicam o custo dos protocolos. Em especial, o número de bytes transmitidos mostra o efeito da política híbrida inserir bytes extra no cabeçalho dos quadros. O aumento do número de bytes transmitidos implica no aumento do consumo de energia com transmissões. A latência é um parâmetro crucial porque o atraso pode causar congestionamentos e perdas, principalmente quando os nós operam com um ciclo de trabalho reduzido no qual os emissores esperaram mais tempo até que o próximo nó acorde.

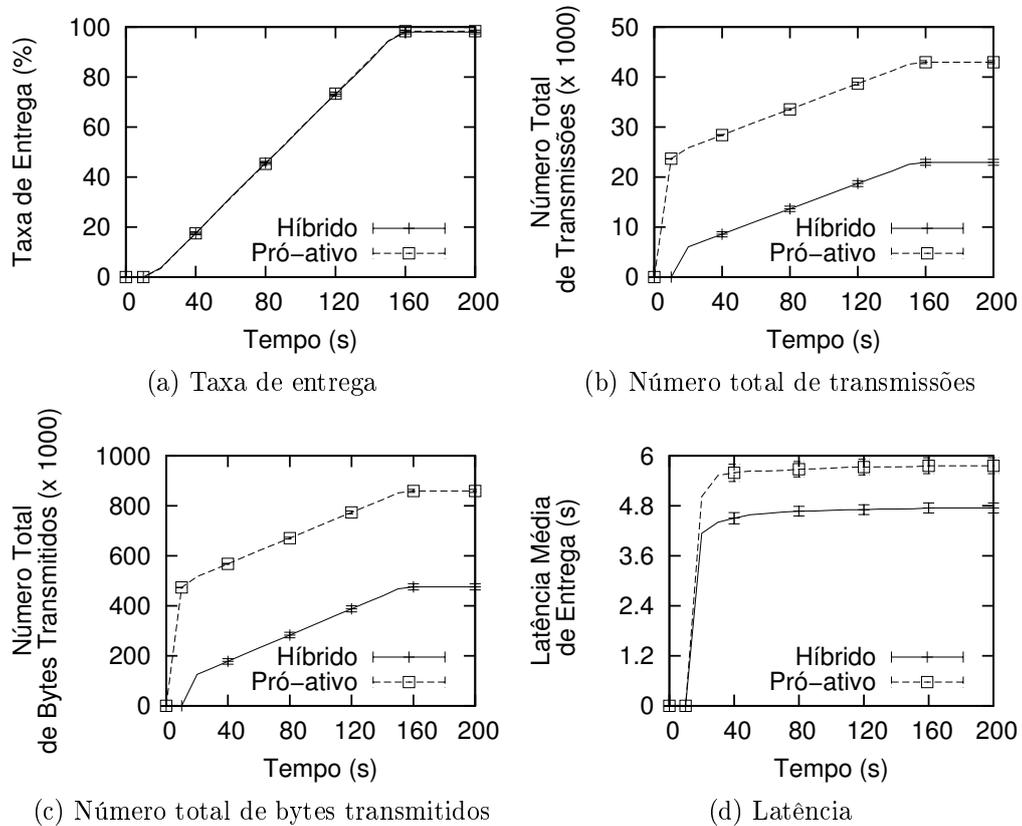


Figura 4.9: Métricas avaliadas para os protocolos híbrido e pró-ativo com ciclo estático de 1%

A figura 4.9 apresenta os resultados para as métricas avaliadas quando os nós sensores operam com um ciclo de trabalho estático de 1%. A figura 4.9-a mostra a taxa média de entrega na qual os protocolos híbrido e pró-ativo entregam 98% das mensagens disseminadas pelo nó monitor. O teste-T com 0,05 de significância mostra que o resultado dos dois protocolos não apresenta qualquer diferença. As perdas de

mensagens nos dois protocolos acontecem por causa do fluxo de dados e do ciclo de trabalho avaliados. Quando os nós sensores operam com um ciclo de trabalho reduzido, eles ficam menos tempo disponíveis para a comunicação, o que causa a concorrência entre as mensagens disseminadas e, conseqüentemente, colisões e atrasos na comunicação. As colisões aconteceram entre os quadros RTS e CTS.

A figura 4.9-b mostra o número total de transmissões durante o tempo de vida da rede para o ciclo de trabalho estático de 1%. O pró-ativo efetua duas vezes mais transmissões que o híbrido porque no pró-ativo, todos os nós compartilham seus escalonamentos e no híbrido, somente os emissores efetuam a descoberta. Como a principal limitação dos protocolos pró-ativos no cenário com ciclo de trabalho reduzido é o custo de compartilhamento, este trabalho seguiu a proposta de Guo et al. [43, 44] para minimizar esse custo fazendo com que cada nó sensor enviasse seu escalonamento somente para uma fração de vizinhos. Assim, os nós sensores do OMAC compartilharam seu escalonamento com 50% de seus vizinhos para reduzir o custo de compartilhamento. Essa fração foi escolhida empiricamente como o menor valor que não reduzisse a taxa de entrega do protocolo pró-ativo. Se ela fosse maior, o protocolo pró-ativo faria mais transmissões. O compartilhamento pró-ativo acontece no início da simulação até 10 s, antes da primeira disseminação, quando o número total de transmissões do protocolo pró-ativo corresponde a 55% do seu número final de transmissões. O custo da descoberta híbrida está concentrado nas primeiras disseminações. Isso porque o destino de cada disseminação está localizado em uma das três áreas de destino ilustradas na figura 4.7, fazendo com que as rotas de disseminação sejam praticamente as mesmas e permitindo que os nós emissores usem as informações descobertas em disseminações subsequentes. Nas primeiras disseminações (entre 10 e 20 s), o protocolo híbrido efetua 4.8 mais transmissões que nas suas outras disseminações.

A principal forma de economizar energia em redes de sensores sem fio é fazer com que os nós sensores adormeçam periodicamente para economizar energia. Dessa forma, quanto menor o ciclo de trabalho dos nós sensores, maior a economia de energia da rede. Outra fonte relevante de consumo de energia é a comunicação de dados. Uma forma de mensurar o consumo de energia com transmissões é a partir do número de bytes transmitidos. A figura 4.9-c mostra o número total de bytes transmitidos durante o tempo de vida da rede para o ciclo de trabalho estático de 1%. O protocolo pró-ativo transmitiu 1,7 vezes mais bytes que o do híbrido. Isso mostra que a economia de energia obtida pelo fato do protocolo híbrido eliminar o compartilhamento pró-ativo é maior que o consumo extra dos quadros híbridos RTS e CTS no cenário avaliado.

A figura 4.9-d apresenta a latência média de entrega durante o tempo de vida da rede para o ciclo de trabalho estático de 1%. A latência do pró-ativo é 1,2 maior que a

do híbrido porque, no pró-ativo, cada nó conhece somente 50% de seus vizinhos, o que reduz as opções de próximo nó. Se o protocolo pró-ativo tivesse feito com que cada nó conhecesse o escalonamento de trabalho de todos seus vizinhos, sua latência seria menor, contudo, isso exige mais transmissões no compartilhamento.

A figura 4.10 apresenta as métricas avaliadas quando os nós sensores operam com um ciclo de trabalho estático de 80%. As figuras 4.10-a à 4.10-d mostram a taxa média de entrega, número total de transmissões, a energia consumida com transmissões e a latência média de entrega durante o tempo de vida da rede, respectivamente. Na figura 4.10-a, os protocolos híbrido e pró-ativo entregam todas as mensagens disseminadas pelo nó monitor. Isso acontece porque os nós sensores ficam acordados e disponíveis para a comunicação durante quase todo o tempo de vida da rede. Assim, diferente do cenário com ciclo de trabalho estático com 1%, a concorrência entre as mensagens disseminadas pelo nó monitor não afeta a entrega das mensagens. A figura 4.10-b mostra que o protocolo pró-ativo efetua 1,1 vezes mais transmissões que o híbrido. Esse resultado é consequência do protocolo pró-ativo fazer com que todos os nós sensores compartilhem seus escalonamentos. Até a primeira disseminação, no intervalo de 10 s, o número de transmissões do protocolo pró-ativo corresponde a 5% do seu número final de transmissões. Após esse tempo, a diferença entre o número de transmissões dos dois protocolos é praticamente a mesma. Comparando as figuras 4.9-b e 4.10-b, observa-se que o aumento do ciclo de trabalho implica em uma redução do número de transmissões. Isso porque quando os nós ficam mais tempo acordados, aumenta-se a chance do próximo nó estar acordado e isso faz com que o compartilhamento pró-ativo e a descoberta híbrida demandem menos transmissões. A figura 4.10-c mostra que o protocolo híbrido transmitiu 1,05 vezes mais bytes que o pró-ativo. Isso mostra que neste cenário, a economia obtida pelo fato do protocolo híbrido eliminar o compartilhamento pró-ativo é menor que o custo extra dos quadros híbridos RTS e CTS. Esse resultado aconteceu porque o aumento do ciclo de trabalho minimiza o número de transmissões necessárias para o compartilhamento pró-ativo e não afeta o *overhead* da política híbrida. A figura 4.10-d mostra que os protocolos pró-ativo e híbrido apresentaram a mesma latência. Isso acontece porque nos dois protocolos, todos os nós ficam acordados durante quase todo o tempo de vida da rede. Nesse caso, o *teste-T* com 0,05 de significância mostra que os dois protocolos não apresentam diferenças na latência.

A figura 4.11 apresenta o desempenho dos dois protocolos avaliados quando os nós operam com um ciclo de trabalho dinâmico. As figuras 4.11-a à 4.11-d apresentam as métricas avaliadas no eixo *y* à esquerda e no da direita, o ciclo de trabalho esperado para os nós que participam das disseminações. A figura 4.11-a mostra que os dois

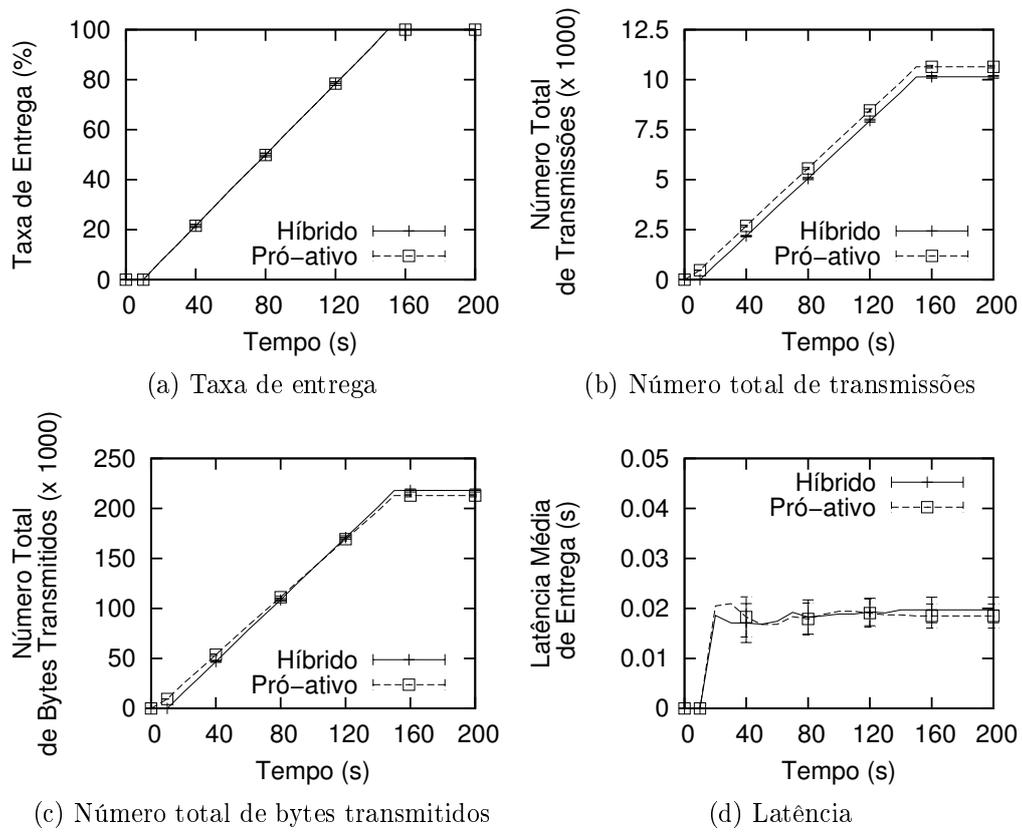


Figura 4.10: Métricas avaliadas para os protocolos híbrido e pró-ativo com ciclo estático de 80%

protocolos entregam quase todos os quadros. A inclinação das curvas é maior no meio do tempo de vida da rede quando o cenário proposto faz com que o nó monitor dissemine mais mensagens. A figura 4.11-b mostra o número total de transmissões em que o protocolo pró-ativo efetuou 1,8 vezes mais transmissões que o protocolo híbrido devido às transmissões iniciais do compartilhamento pró-ativo. A figura 4.11-c mostra que o protocolo pró-ativo transmite 1,43 vezes mais bytes que o protocolo híbrido. Isso porque, para o cenário com ciclo de trabalho dinâmico, a economia de energia obtida pelo fato do protocolo híbrido eliminar o compartilhamento pró-ativo é maior que o custo extra dos quadros híbridos RTS e CTS. A figura 4.11-d mostra a latência média de entrega em que o atraso pró-ativo é 1,3 vezes maior que o do protocolo híbrido. Isso porque no protocolo pró-ativo, cada nó conhece o escalonamento de 50% de seus vizinhos. A latência inicial dos dois protocolos é maior que no restante da simulação porque os nós operam com um ciclo de trabalho reduzido. No final da simulação, eles também operam com um ciclo de trabalho reduzido, mas esse atraso é minimizado pelo fato do valor mostrado ser a latência média durante todo o tempo de vida da rede.

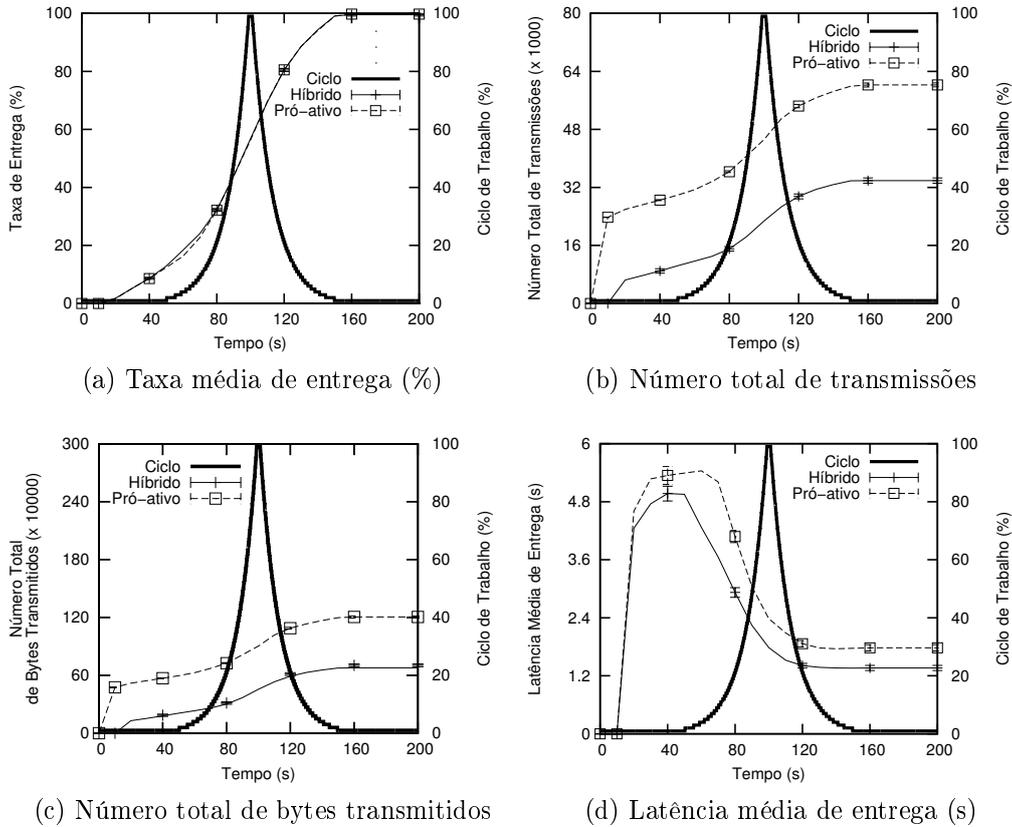


Figura 4.11: Métricas avaliadas para os protocolos híbrido e pró-ativo com ciclo dinâmico

Os resultados de simulação revelam que para o cenário avaliado, a política híbrida é mais indicada que a pró-ativa quando os nós operam com um ciclo estático reduzido. Como a melhor forma de economizar energia em RSSFs é reduzir o ciclo de trabalho, esse resultado é positivo. Além disso, a política proposta também apresenta resultados expressivos para o ciclo dinâmico.

### 4.3.3 Política Híbrida com Protocolo MAC Reativo

Esta seção avalia a influência da política híbrida no protocolo AMAC para cenários de disseminação de dados em que os nós operam com ciclos de trabalho estático e dinâmico. As métricas avaliadas são as mesmas da seção anterior.

A figura 4.12 apresenta as métricas avaliadas quando os nós sensores operam com um ciclo de trabalho estático de 1%. A figura 4.12-a apresenta a taxa média de entrega na qual o protocolo híbrido entregou 1,03 vezes mais quadros que o protocolo reativo e a figura 4.12-b mostra o número total de transmissões em que o reativo efetua 7,8

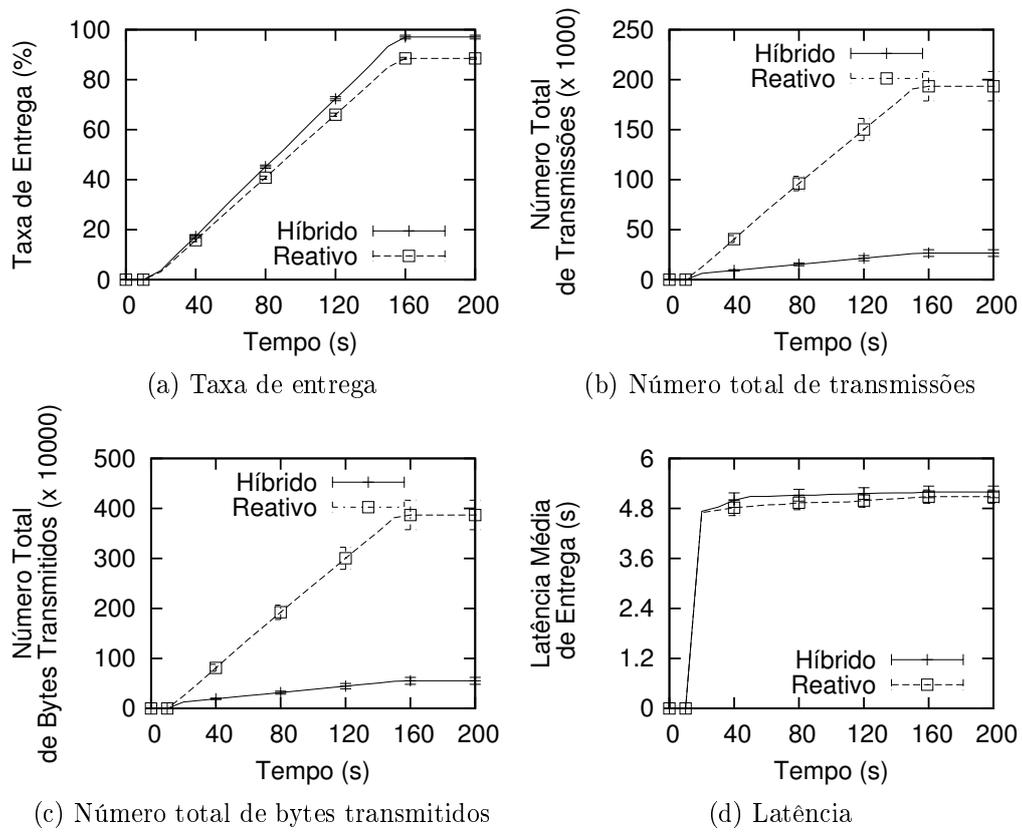


Figura 4.12: Métricas avaliadas para os protocolos híbrido e reativo com ciclo estático de 1%

vezes mais transmissões que o protocolo híbrido. Os dois resultados acontecem porque quando os nós sensores operam com um ciclo de trabalho reduzido, a descoberta reativa demanda que cada nó emissor retransmita mais quadros para descobrir o momento em que próximo nó estará acordado. O protocolo reativo faz mais transmissões que o protocolo híbrido porque no primeiro, para cada propagação do quadro de dados, o nó emissor efetua descobertas enquanto no protocolo híbrido somente quando desconhece o escalonamento de trabalho do próximo nó. Como os sensores do cenário em questão ficam quase sempre adormecidos e, por isso, indisponíveis para a comunicação, o aumento do número de retransmissões do protocolo reativo aumenta a concorrência entre as mensagens disseminadas. Consequentemente, o protocolo reativo faz mais colisões e tem um atraso maior que o do protocolo híbrido. Tais colisões aconteceram entre os quadros RTS e CTS usados na descoberta e justificam as perdas de mensagens e a diferença entre os dois protocolos. A figura 4.12-c mostra que o protocolo reativo transmite 7,2 vezes mais bytes que o protocolo híbrido e isso significa que, para o cenário avaliado, os ganhos de energia obtidos pelo fato do protocolo híbrido evitar descobertas são

maiores que o custo extra dos quadros híbridos RTS e CTS. A figura 4.12-d mostra a latência média de entrega em que os dois protocolos apresentaram o mesmo resultado. O *teste-T* com 0,05 de significância mostra que os dois protocolos não apresentam qualquer diferença de latência. Se por um lado a latência do protocolo reativo deveria ser menor que a do híbrido porque no reativo, cada nó emissor sempre escolhe como próximo nó o primeiro vizinho que acordar; por outro lado, o protocolo reativo faz mais transmissões que o híbrido. Essa diferença, em termos de transmissões, normalmente aumenta o número de colisões e, conseqüentemente, o atrasa a comunicação de dados.

A figura 4.13 apresenta as métricas avaliadas quando os nós sensores operam com um ciclo de trabalho estático de 80%. Na figura 4.13-a, os dois protocolos são capazes de entregar todas as mensagens. Nas figuras 4.13-b e 4.13-c, o protocolo reativo efetua 1,1 e 1,05 vezes mais transmissões de quadros e bytes que o protocolo híbrido, respectivamente. Como explicado anteriormente, tais resultados acontecem porque o protocolo híbrido evita descobertas. A figura 4.13-d mostra a latência média de entrega em que os dois protocolos apresentaram o mesmo resultado.

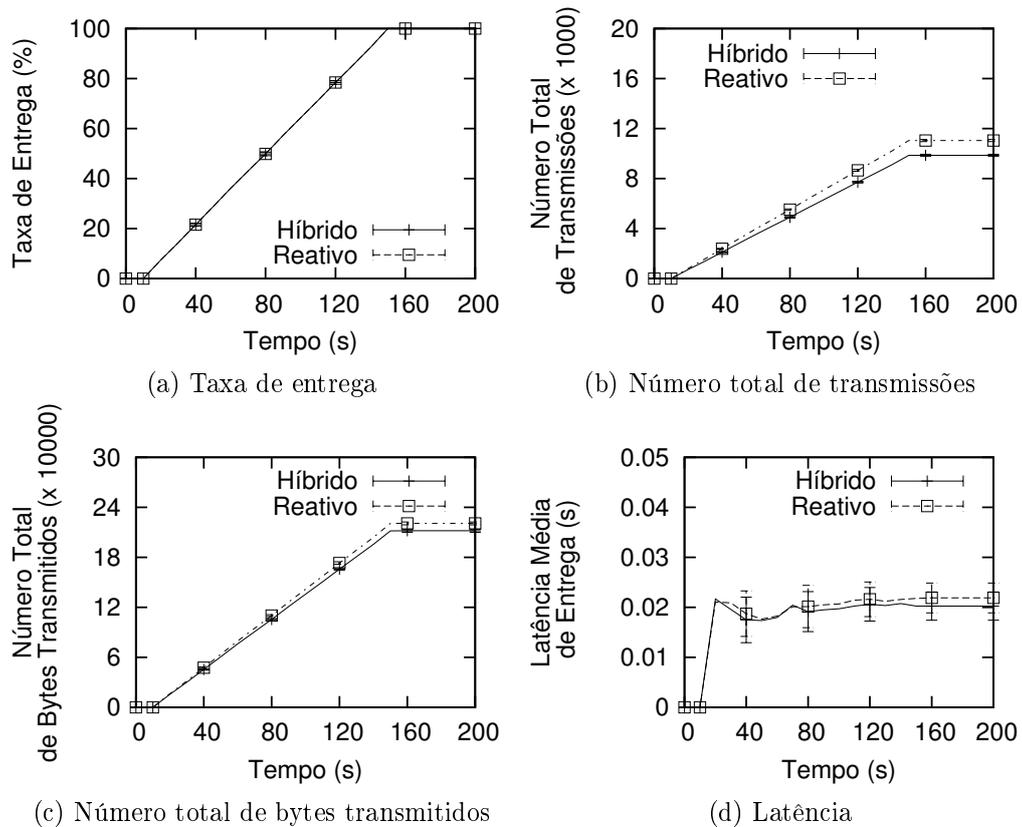


Figura 4.13: Métricas avaliadas para os protocolos híbrido e reativo com ciclo estático de 80%

Na figura 4.14, os nós operam com um ciclo de trabalho dinâmico. Os dois protocolos entregam praticamente todas as mensagens e têm a mesma latência. O protocolo reativo transmite 3,2 e 2,6 vezes mais quadros e bytes que o protocolo híbrido, respectivamente. Isso porque o protocolo híbrido executa suas descobertas somente para as primeiras disseminações. Além disso, a diferença entre os protocolos nas duas figuras é maior no início e no final da simulação quando os nós operam com um ciclo de trabalho reduzido.

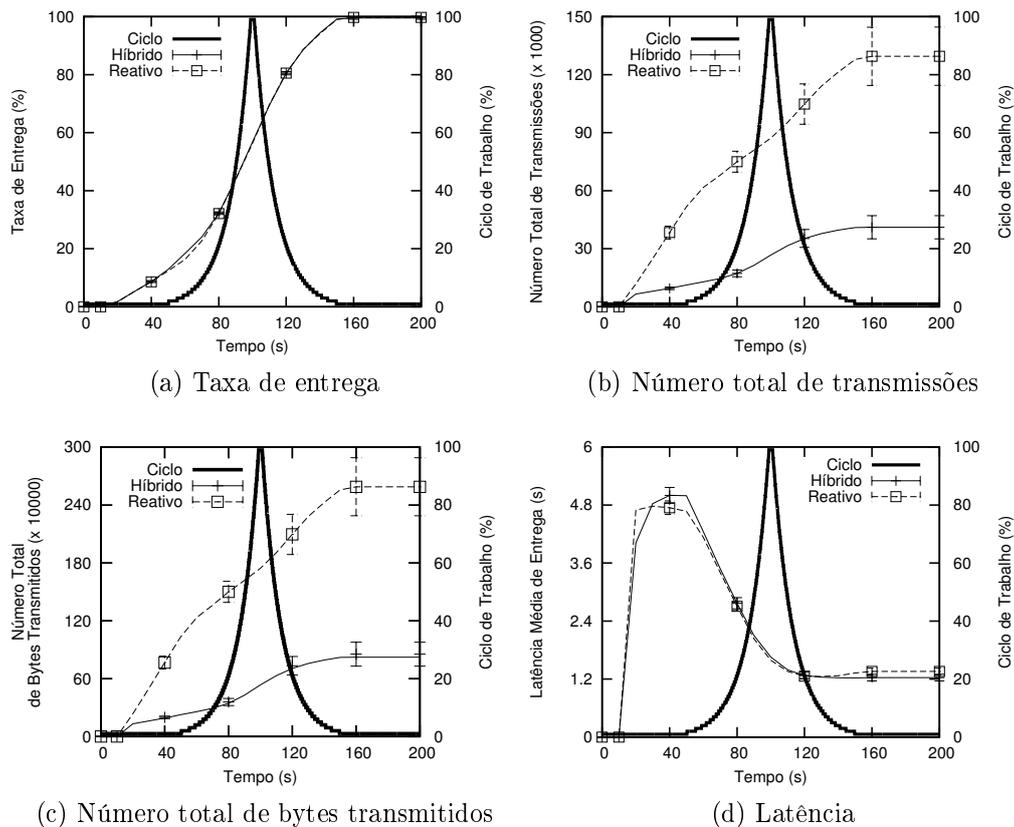


Figura 4.14: Métricas avaliadas para os protocolos híbrido e reativo com ciclo dinâmico

Os resultados de simulação avaliaram os efeitos da inserção da política híbrida no protocolo AMAC reativo. Eles mostram que para o cenário avaliado, a política híbrida é mais indicada que a reativa quando os nós operam com ciclo estático reduzido ou dinâmico.

## 4.4 Conclusões

Este capítulo propôs uma política híbrida para determinar o escalonamento de trabalho de nós vizinhos na comunicação de dados em RSSFs. A ideia básica dessa política é

que cada nó emissor insere seu escalonamento de trabalho no quadro e cada receptor mantém uma tabela com essa informação para ser usada em transmissões futuras. A informação inserida nos quadros elimina o custo de compartilhamento dos protocolos MAC pró-ativos e minimiza o de descoberta dos protocolos reativos. Resultados de simulação revelam que a política híbrida reduz o número de transmissões de quadros e de bytes dos protocolos MAC pró-ativos e reativos na maioria dos cenários avaliados. O único cenário avaliado em que isso não acontece é com o ciclo de trabalho estático de 80% em que o protocolo híbrido teve um custo com transmissões de bytes maior que o protocolo pró-ativo, apesar desse transmitir mais quadros. Em especial, a política proposta é mais indicada que as políticas pró-ativa e reativa quando os nós operam com ciclo estático reduzido ou dinâmico.

## Capítulo 5

# Projeto Combinado entre a Política MAC Reativa e o Roteamento baseado no Receptor

*Projetos conjuntos têm mais chance de sucesso quando se beneficiam de ambos os lados.*  
*Eurípedes*

Este capítulo apresenta e avalia um projeto combinado envolvendo a política MAC reativa e uma abordagem de roteamento baseada no receptor e está organizado como descrito a seguir. A seção 5.1 discute a motivação e objetivos desse projeto. A seção 5.2 propõe o *Receiver-based Medium Access Control* (RbMAC), o protocolo MAC reativo usado no projeto combinado proposto neste capítulo. Essa seção também verifica formalmente algumas propriedades do RbMAC. A seção 5.3 apresenta um projeto combinado entre o RbMAC e um algoritmo de roteamento baseado no receptor e o compara com outras soluções para a comunicação de dados em redes de sensores sem fio. A seção 5.4 propõe e avalia uma técnica de ciclo de trabalho dinâmico para o projeto proposto. A seção 5.5 conclui este capítulo.

### 5.1 Introdução

O principal desafio para transformar as Redes de Sensores Sem Fio (RSSFs) [4] na tecnologia de sensoriamento do futuro é projetar soluções eficientes em termos de energia. Outro desafio crucial é a topologia dinâmica dessas redes, que geralmente é consequência dos nós adormecerem para economizar energia. Nesse contexto, a comunicação de dados vem recebendo atenção especial dos pesquisadores porque, normalmente, corresponde à tarefa com o maior custo de energia nas RSSFs e trata das mudanças topo-

lógicas. As camadas de rede e de controle de acesso ao meio (MAC) são responsáveis pela comunicação em RSSFs.

A camada de rede é responsável pela escolha sistemática do próximo nó durante o roteamento, o que leva à definição de rotas. Essa escolha de roteamento é chamada de decisão de propagação e é baseada no emissor ou no receptor. Na primeira forma de roteamento, quando um nó recebe um pacote, o nó verifica se ele foi o escolhido para continuar o roteamento. Se verdadeiro, o nó escolhe um subconjunto de seus vizinhos para continuar o roteamento e, depois, propaga o pacote. Na escolha baseada no receptor, quando um nó recebe um pacote, o nó decide localmente se deve ou não propagar o pacote. As figuras 5.1 e 5.2 ilustram respectivamente um exemplo de cada uma dessas abordagens quando um nó A recebe um pacote. No roteamento baseado no emissor, quando o nó A recebe um pacote e ele é o próximo nó, A toma sua decisão de propagação (figura 5.1-a). Supondo que A escolhe C como o próximo nó (figura 5.1-b), A envia o pacote para C (figura 5.1-c) que repete esse processo (figura 5.1-d). No roteamento baseado no receptor, quando o nó A recebe um pacote, ele também toma a decisão de propagação (figura 5.2-a), contudo, nesse caso, essa decisão será se ele propaga ou não o pacote recebido. A figura 5.2-b considera que essa decisão é pela propagação e, por isso, na figura 5.2-c, A propaga o pacote. Na figura 5.2-d, todos os nós que receberam o pacote tomam a decisão de propagação, continuando o roteamento.

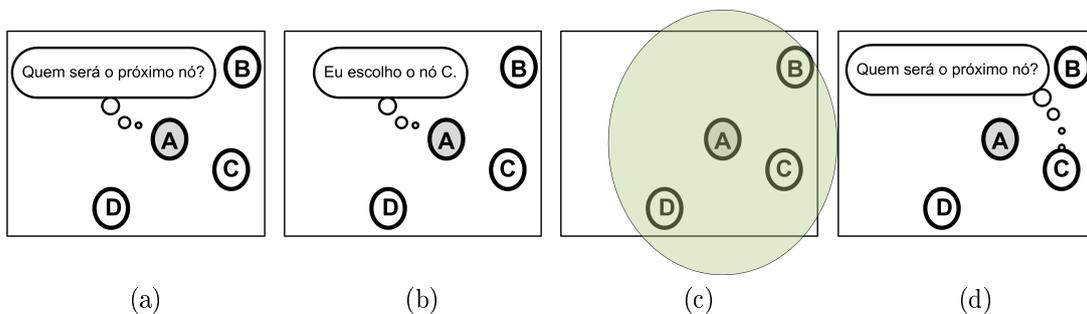


Figura 5.1: Exemplo da abordagem de roteamento baseada no emissor

A camada MAC é responsável por entregar o pacote para o próximo nó. Por isso, ela controla o uso do canal para evitar colisões e gerencia o escalonamento de trabalho dos nós para tratar da comunicação quando o próximo nó estiver adormecido. O gerenciamento do escalonamento de trabalho é crucial em RSSFs, pois a melhor forma de economizar energia nessas redes é desativar os nós sempre que possível [56]. Dada importância da energia em RSSFs, o principal desafio da camada MAC nessas redes é determinar o escalonamento de trabalho dos nós para que o roteamento não seja prejudicado quando o próximo nó estiver adormecido. A maioria dos protocolos

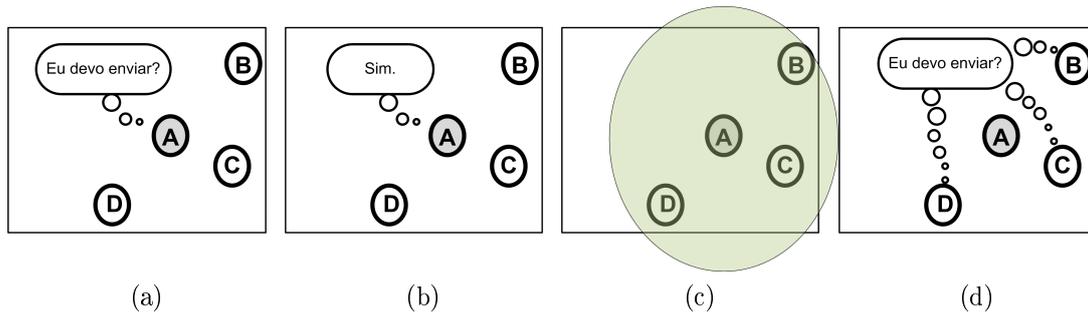


Figura 5.2: Exemplo da abordagem de roteamento baseada no receptor

MAC considera a política pró-ativa na qual os nós vizinhos compartilham seus respectivos escalonamentos de trabalho para que quando um nó tiver que enviar um quadro, ele saiba quando o vizinho escolhido estará acordado. A limitação da política MAC pró-ativa é que seu compartilhamento acontece através da troca de mensagens, o que aumenta o consumo de energia da rede. Além disso, como o ambiente das RSSFs é dinâmico e os nós possuem restrições de hardware, as informações compartilhadas se desatualizam durante o tempo de vida da rede fazendo com que os nós sensores troquem mais mensagens para atualizar as informações compartilhadas. Esses compartilhamentos acontecem até mesmo durante os períodos de inatividade em que nenhum dado de sensoriamento é roteado. Por outro lado, alguns protocolos MAC usam a política reativa na qual os nós sensores não compartilham informações sobre escalonamento e quando um nó tem que enviar uma mensagem, primeiro, ele descobre quando o próximo nó está acordado. Essa descoberta acontece através da retransmissão de quadros de controle. A limitação da política reativa é o custo de energia com o envio dos quadros de descoberta. Dadas as limitações das técnicas existentes, novas soluções para a comunicação quando o próximo nó estiver adormecido devem ser investigadas.

Diferente das redes tradicionais que são normalmente projetadas para acomodar uma variedade de aplicações, as RSSFs são projetadas tipicamente para uma aplicação específica. Assim, o desenvolvimento combinado de duas ou mais camadas da pilha de protocolos pode melhorar o desempenho e o consumo de energia dessas redes. O projeto combinado da pilha de protocolos consiste em se escolher e ajustar o protocolo de uma camada para favorecer o funcionamento básico dos protocolos das outras camadas, evitando seus piores casos e estimulando os melhores. Na arquitetura das redes de propósito geral, um protocolo pode desconsiderar o funcionamento básico dos protocolos existentes em outras camadas. Um exemplo de projeto combinado na área de redes de computadores acontece quando os protocolos de transporte *User Datagram Protocol* (UDP) e *Transport Control Protocol* (TCP) limitam o tamanho de seus segmentos para

minimizar as fragmentações do *Internet Protocol* (IP).

O projeto combinado entre o roteamento baseado no receptor e uma política MAC reativa poderia ser investigado como outra solução para a comunicação de dados quando o próximo nó estiver adormecido. Isso porque a diversidade de candidatos a próximo nó do roteamento baseado no receptor pode reduzir o custo da descoberta reativa. O roteamento baseado no receptor é mais robusto às falhas no próximo nó que o baseado no emissor, pois qualquer vizinho que recebe um pacote pode se escolher como o próximo nó. No roteamento baseado no emissor, apenas o vizinho escolhido pode ser o próximo nó. O roteamento baseado no receptor normalmente aumenta o número de candidatos a próximo nó quando comparado com a abordagem baseada no emissor. Conseqüentemente, ele minimiza o custo-benefício da informação sobre o escalonamento de trabalho dos nós vizinhos para a descoberta reativa e pode tornar a estratégia reativa de retransmissões eficiente em termos de energia. Uma característica dos protocolos propostos para a camada MAC das RSSFs é que eles são focados no roteamento baseado no emissor de tal forma que a camada de rede escolhe o próximo nó e o insere no cabeçalho do pacote e a camada MAC entrega essa mensagem quando o nó escolhido para continuar o roteamento estiver acordado.

O objetivo deste capítulo é apresentar e avaliar um projeto combinado envolvendo a política MAC reativa e o roteamento baseado no receptor e compará-lo com outros projetos para a comunicação de dados envolvendo as políticas MAC pró-ativa e reativa e abordagens de roteamento baseadas no emissor e no receptor. A ideia do projeto combinado proposto neste capítulo é usar a diversidade de candidatos a próximo nó do roteamento baseado no receptor para reduzir o custo de descoberta da abordagem reativa. Um nó emissor faz menos transmissões de descoberta para encontrar algum vizinho acordado que para encontrar um vizinho específico. O autor desta tese desconhece outro trabalho que reduz o custo de descoberta reativo a partir da diversidade de candidatos a próximo nó do roteamento baseado no receptor. Este capítulo também propõe o protocolo *Receiver-based Medium Access Control* (RbMAC), o protocolo MAC usado no projeto combinado proposto. O RbMAC não tem o custo de compartilhamento da política pró-ativa e se diferencia dos demais protocolos reativos propostos na literatura porque sua descoberta busca algum vizinho acordado e faz confirmações baseadas em *piggyback* de que cada próximo nó continuará o roteamento. Quando um nó decide ser o próximo nó, ele propaga o quadro de dados imediatamente, começando a próxima descoberta. Essa propagação também permite que o nó anterior pegue carona na nova descoberta para confirmar que um vizinho continuou o roteamento e, conseqüentemente, parar a descoberta anterior. Os demais protocolos reativos fazem a descoberta de um vizinho específico e consideram o *handshaking*

RTS/CTS/dados/ACK do IEEE 802.11 para o nó anterior terminar sua descoberta e confirmar que o próximo nó continuará o roteamento. Resultados teóricos e de simulação revelam que o projeto combinado proposto reduz o número de transmissões e a latência na comunicação de dados quando comparado com outras soluções avaliadas neste trabalho. Este capítulo também verifica formalmente algumas propriedades do RbMAC e apresenta uma técnica de ciclo dinâmico para o RbMAC reduzir a latência e o consumo de energia do projeto combinado proposto em cenários específicos nos quais a mesma rota é utilizada sistematicamente.

## 5.2 Protocolo Receiver-based MAC

Esta seção apresenta o RbMAC. A seção 5.2.1 descreve seu funcionamento básico. A seção 5.2.2 mostra o mecanismo de descobertas do RbMAC e a seção 5.2.3 verifica formalmente algumas de suas propriedades.

### 5.2.1 Funcionamento Básico

O funcionamento básico do RbMAC é modelado a partir de eventos e ações de tal forma que a ocorrência de um evento gera uma ação como resposta. A figura 5.3 descreve a máquina de estados finita do RbMAC em que as transições acontecem devido a eventos gerados por outras camadas ou pela expiração de temporizadores do RbMAC. Os estados dessa máquina são: *escutar*, *receber*, *esperar*, *enviar* e *dormir*. Seus eventos são *enviar quadro*, *receber quadro*, *escutar*, *CSMA p-persistente*, *dormir* e *acordar*. Suas ações são:  $\lambda$ , *processar quadro*, *acessar meio*, *reescalonar*, *dormir*, *acordar* e *continuar acordado*. A ação  $\lambda$  é nula e permite que o RbMAC mude de estado sem executar qualquer atividade. Quando um nó começa a operar, ele liga o rádio para escutar o meio e o RbMAC vai para o estado *escutar*.

A fase de recepção do RbMAC começa quando a camada física identifica alguma alteração no canal. As alterações no canal significam que algum vizinho começará uma transmissão. Assim, a camada física gera o evento *receber quadro* (figura 5.3, ponto A). Esse evento faz com que o RbMAC vá para o estado *receber* e execute a ação  $\lambda$ . Nesse estado, o nó corrente recebe o quadro a ser enviado e o RbMAC permanece nele até que o meio fique livre. Quando isso acontecer, a camada física do nó corrente gera o evento *escutar* cuja ocorrência faz com que o RbMAC retorne ao estado *escutar* e execute a ação *processar quadro* (figura 5.3, ponto B). Essa ação efetua as tarefas de enlace tais como enquadramento, verificação e correção de erros. Se o nó tiver recebido

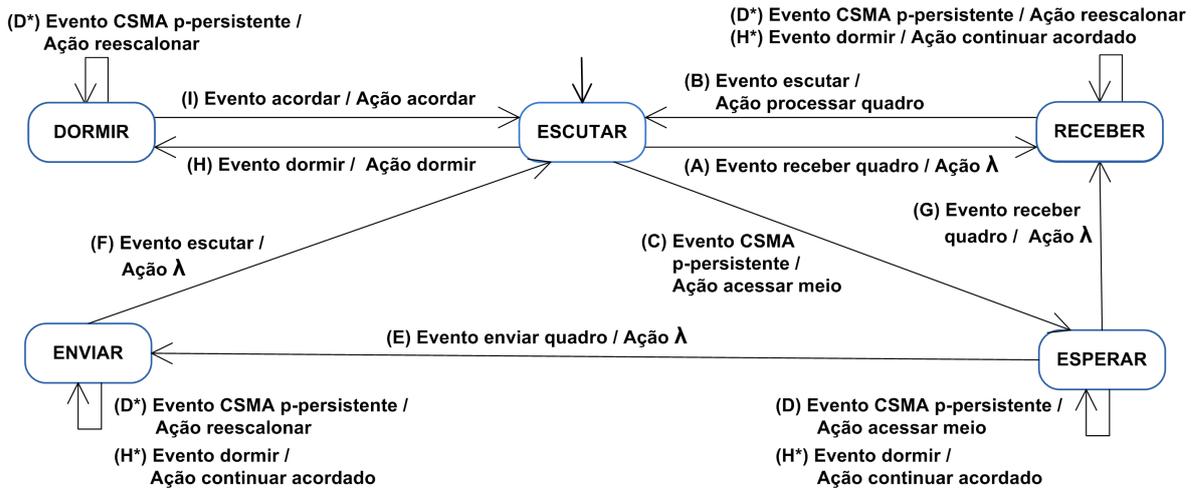


Figura 5.3: Funcionamento básico do RbMAC

o quadro sem erros (e.g., devido a colisões), o RbMAC o encaminha para a camada de rede e, caso contrário, ele descarta o quadro recebido.

O RbMAC controla o acesso ao meio através de uma abordagem baseada em *Carrier Sense Multiple Access* (CSMA) *p*-persistente. Escolheu-se essa abordagem porque ela não compartilha informações sobre vizinhos (como, por exemplo, o *Frequency and Time Division Multiplexing*), não envia quadros de controle (como, por exemplo, o *handshaking* do IEEE 802.11) e efetua normalmente menos colisões que outras abordagens tradicionais tais como o Aloha, CSMA 1-persistente e CSMA não-persistente. Entretanto, essa abordagem não é livre de colisões. As colisões são um dos principais desafios na camada MAC e o RbMAC apresenta duas alternativas para reduzir os efeitos das colisões. Primeiro, como o protocolo proposto considera o roteamento baseado no receptor, outro candidato a próximo nó que não detectou a colisão pode continuar o roteamento. Segundo, a descoberta reativa do RbMAC faz com que os receptores tenham várias oportunidades (cada uma das retransmissões) para receber o quadro de dados.

A fase de envio começa quando a camada de rede envia um pacote para a camada MAC, gerando o evento *CSMA p-persistente*. Quando esse evento acontece e o RbMAC está no estado *escutar*, o protocolo proposto nesta seção vai para o estado *aguardar* e executa a ação *acessar meio* (figura 5.3, ponto *C*). Nessa ação, o RbMAC permite que a camada física comece uma transmissão com probabilidade  $p$  e, com a probabilidade  $1-p$ , o RbMAC reescalona o evento *CSMA p-persistente* para continuar suas tentativas de acesso ao meio. Se o evento *CSMA p-persistente* acontecer quando o RbMAC estiver no estado *aguardar*, ele permanece nesse estado e executa a ação *acessar meio* (figura 5.3, ponto *D*). Se o evento *CSMA p-persistente* acontecer quando o RbMAC

estiver nos estados *dormir*, *receber* ou *enviar*, o RbMAC permanece no mesmo estado (figura 5.3, ponto  $D^*$ ) e executa a ação reescalonar no qual ele reescalona o evento ocorrido. Observa-se que o tempo de reescalonamento do evento CSMA p-persistente depende do estado corrente do RbMAC. Assim, por exemplo, no estado *aguardar*, esse valor é baseado no tempo de transmissão de um quadro e, no estado *dormir*, ele depende do tempo em que o nó ainda permanecerá adormecido no ciclo operacional corrente. Quando a camada física começa uma transmissão, ela muda o rádio para o modo de transmissão, o que gera o evento *enviar quadro*. Esse evento faz com que o RbMAC vá para o estado *enviar* e execute a ação  $\lambda$  (figura 5.3, ponto  $E$ ). O RbMAC permanece nesse estado até que a camada física termine sua transmissão. Quando isso acontecer, tem-se o evento *escutar*, fazendo com que o RbMAC retorne ao estado *escutar* e execute a ação  $\lambda$  (figura 5.3, ponto  $F$ ). Como o rádio de um nó sensor permanece no modo de recepção enquanto o RbMAC estiver no estado *aguardar*, tem-se que o evento *receber quadro* pode acontecer e, nesse caso, o RbMAC vai para o estado *receber* e executa a ação  $\lambda$  (figura 5.3, ponto  $G$ ). Entretanto, quando o RbMAC estiver no estado *enviar*, o nó sensor não será capaz de receber quadros porque o rádio normalmente é *half-duplex*.

O RbMAC também faz com que os nós sensores adormeçam periodicamente para economizarem energia. Ele divide o tempo de vida dos nós sensores em ciclos operacionais sendo que cada ciclo é composto por um período em que o nó permanece acordado seguido por outro em que o nó dorme. A posição de cada nó em seu ciclo operacional é independente dos demais porque a solução proposta nesta seção não efetua sincronizações. Quando um nó começa a operar, o RbMAC liga o rádio e inicia um temporizador para controlar seus ciclos operacionais. Se esse temporizador expirar e o RbMAC estiver no estado *escutar*, tem-se o evento *dormir*, fazendo com que o RbMAC vá para o estado *dormir* e execute a ação *dormir* (figura 5.3, ponto  $H$ ). Nessa ação, o RbMAC desliga o rádio e reinicia seu temporizador para expirar no final do ciclo operacional corrente. O término desse temporizador no estado *dormir* gera o evento *acordar* e faz com que o RbMAC execute a ação *acordar*. Nessa ação, o RbMAC liga o rádio e reinicia o temporizador para começar um novo ciclo operacional (figura 5.3, ponto  $I$ ). Quando o evento *dormir* acontecer e o RbMAC estiver nos estados *receber*, *aguardar* ou *enviar*, o RbMAC permanece no estado corrente e executa a ação *continuar acordado* (figura 5.3, ponto  $H^*$ ). Essa ação faz com que o nó permaneça acordado por mais algumas unidades de tempo para terminar sua atividade corrente de comunicação (enviar ou receber). Quando um nó sensor ficar acordado por um tempo extra, o projetista da rede pode permitir que o nó aumente provisoriamente seu período de dormir nos próximos ciclos operacionais para recompensar seu consumo de energia extra.

### 5.2.2 Mecanismo de Descoberta

A principal diferença do RbMAC para os protocolos MAC reativos propostos na literatura está no mecanismo de descoberta. Esses protocolos consideram o roteamento baseado no emissor, que possui um único candidato a próximo nó e, o RbMAC, o roteamento baseado no receptor que possui múltiplos candidatos a próximo nó. Outra diferença é que como os quadros de controle e de dados em RSSFs possuem praticamente o mesmo número de bytes, o RbMAC não envia quadros de controle e usa os de dados para as descobertas e respostas. Assim, quando um nó emissor inicia sua descoberta, ele envia o próprio quadro de dados como o de descoberta. Da mesma forma, quando o próximo nó, estiver acordado e receber esse quadro, ele envia o próprio quadro de dados como o de resposta. Como mostrado na seção 2.2.3.3, a política MAC reativa faz com que cada nó emissor envie quadros de descoberta para identificar se o próximo nó está acordado e, quando isso acontecer, o próximo nó envia um quadro de resposta para informar que está acordado. O fato do RbMAC usar o quadro de dados para descobertas e respostas também proporciona o uso de uma técnica baseada em *piggyback* na qual cada nó intermediário pode confirmar que algum vizinho continuará o roteamento. Quando um vizinho recebe um quadro de dados e decide ser o próximo nó, esse vizinho propaga o quadro imediatamente, começando a próxima descoberta. Quando o nó anterior recebe a propagação do quadro, ele pega carona nesse envio e confirma o sucesso de sua descoberta e que o roteamento continuou. O efeito dessa confirmação é similar ao quadro ACK do protocolo IEEE 802.11 no qual o próximo nó confirma o recebimento dos quadros dados. Os protocolos reativos normalmente são baseados no *handshaking* do protocolo IEEE 802.11 e trocam quadros RTS/CTS/DADOS/ACK.

Os protocolos MAC reativos se diferenciam pela forma como definem o número máximo e o intervalo entre as retransmissões. O RbMAC define esses parâmetros para garantir que todos os vizinhos de um nó emissor estarão acordados em pelo menos uma das possíveis retransmissões da descoberta e minimizar o número máximo de retransmissões. Assim, o RbMAC escalona as retransmissões baseado em duas condições:

- O tempo total da descoberta (o intervalo entre a primeira e última retransmissão) deve ser maior ou igual ao período adormecido para garantir que todos os vizinhos sempre acordarão durante a descoberta.
- O intervalo entre as retransmissões deve ser menor ou igual ao período acordado para garantir que quando um vizinho acordar, o nó emissor sempre efetuará pelo menos uma retransmissão antes que esse vizinho adormeça novamente.

Dado que todos os vizinhos sempre acordam durante a descoberta e que quando

um vizinho acorda, o nó emissor sempre efetua uma retransmissão antes que esse vizinho adormeça novamente, o RbMAC garante que cada vizinho do nó emissor estará acordado em pelo menos uma das possíveis retransmissões do quadro de dados. Além disso, o RbMAC minimiza o número de retransmissões ao maximizar o intervalo entre elas e minimizar o tempo total da descoberta. Logo, o protocolo proposto nesta seção faz com que o tempo total da descoberta e o intervalo entre as retransmissões sejam iguais aos períodos adormecido e acordado, respectivamente. Como mostrado na equação (5.1), onde o tamanho desses períodos são respectivamente  $d$  e  $a$ , o número total de retransmissões será o teto da divisão do período adormecido pelo acordado.

$$\text{Número de retransmissões} = \left\lceil \frac{d+a}{a} \right\rceil \quad (5.1)$$

A equação (5.2) mostra o número esperado de transmissões do RbMAC que corresponde ao somatório de cada uma das  $\left\lceil \frac{d+a}{a} \right\rceil$  possíveis retransmissões multiplicado por sua probabilidade, onde  $c$  é número de candidatos a próximo nó.

$$E[TX_c] = \sum_{i=0}^{\left\lceil \frac{d+a}{a} \right\rceil} [i \times P(TX_{i,c})] \quad (5.2)$$

A equação (5.3) mostra a probabilidade de cada uma das possíveis  $i$  retransmissões acontecerem quando o nó emissor tem  $c$  candidatos a próximo nó, onde  $P(S_k)$  e  $P(A_k)$  são as probabilidades de um candidato estar dormindo ou acordado na  $k$ -ésima retransmissão, respectivamente. Nessa equação, todas as  $i - 1$  retransmissões anteriores têm que acontecer e pelo menos um dos  $c$  candidatos tem que estar acordado na retransmissão corrente. Assim, a equação multiplica a probabilidade de cada candidato estar dormindo nas  $i - 1$  retransmissões anteriores pela probabilidade de pelo menos um candidato estar acordado na  $i$ -ésima retransmissão.

$$P(TX_{i,c}) = \prod_{j=0}^{i-1} [P(S_j)^c] \times \sum_{j=1}^c [P(S_i)^{j-1} \times P(A_i)] \quad (5.3)$$

As equações (5.4) e (5.5) apresentam as probabilidades de um candidato a próximo nó estar dormindo ou acordado na  $k$ -ésima retransmissão, respectivamente. As duas equações são dadas pela definição clássica de probabilidade onde a probabilidade de um evento é o número de pontos amostrais favoráveis a ele dividido pelo espaço amostral. Em ambas as equações, o espaço amostral que corresponde ao tempo restante na descoberta cujo valor diminui em  $a$  unidades de tempo por retransmissão. Dado

que o nó emissor faz uma retransmissão somente quando ele considera que todos seus candidatos estavam dormindo em todas as retransmissões anteriores, o número de pontos amostrais favoráveis ao evento dormir diminui em  $a$  unidades e o de favoráveis ao evento acordado se mantém constante.

$$P(S_k) = \frac{d - (k \times a)}{(d + a) - (k \times a)} \quad (5.4)$$

$$P(A_k) = \frac{a}{(d + a) - (k \times a)} \quad (5.5)$$

### 5.2.3 Verificação Formal

Esta seção verifica formalmente algumas propriedades do RbMAC. A técnica de verificação considerada é o *model checking* [12, 26, 73], uma técnica de verificação formal na qual um sistema pode ser modelado a partir de um grafo contendo estados e transições. A estrutura desse grafo consiste em uma árvore cuja raiz é o estado inicial e os caminhos na árvore representam todas as computações do sistema modelado. A verificação consiste em procurar em todos os estados do grafo se o sistema satisfaz um conjunto de propriedades expressadas em uma lógica temporal. O modelo de verificação é o NuSMV [3] e o conjunto  $\varphi$  de propriedades verificadas é listado abaixo.

- $\varphi_1$ : Cada estado  $s$  é alcançável a partir do estado inicial  $s_0$  e  $s_0$  é alcançável a partir de cada estado  $s$ . Essa propriedade garante a inexistência de estados de erro ou inalcançáveis.
- $\varphi_2$ : Se o RbMAC estiver no estado *dormir*, o nó sensor não recebe nem envia mensagens. Essa propriedade garante que o rádio está desligado no estado *dormir*.
- $\varphi_3$ : Os protocolos MAC e de rede podem receber/enviar pacotes/quadros simultaneamente. Essa propriedade,  $\varphi_4$  e  $\varphi_5$  garantem (ou limitam) a independência das ações de receber/enviar.
- $\varphi_4$ : O RbMAC não pode receber e enviar mensagens simultaneamente. Essa propriedade garante a limitação física dos nós sensores em que o rádio é geralmente *half-duplex*.
- $\varphi_5$ : A camada de rede pode receber e enviar pacotes simultaneamente, pois a camada MAC armazena os pacotes ainda não processados.

- $\varphi_6$ : A camada de rede recebe um pacote se e, somente se, o RbMAC o receber anteriormente. Essa propriedade e  $\varphi_7$  garantem a ordem em que os dados são processados na pilha de protocolos.
- $\varphi_7$ : O RbMAC envia um quadro se e, somente se, a camada de rede o enviou anteriormente.

## 5.3 Resultados de Simulação

Esta seção apresenta e avalia um projeto combinado envolvendo o RbMAC e um protocolo de roteamento baseado no receptor e compara esse projeto combinado com outras soluções para a comunicação de dados envolvendo protocolos MAC pró-ativos e reativos e algoritmos de roteamento baseados no emissor e no receptor. As avaliações e comparações acontecem em um cenário de disseminação de dados para RSSFs no qual um nó monitor deseja enviar dados para os nós sensores. A seção 5.3.1 mostra os parâmetros de simulação. A seção 5.3.2 avalia o efeito do número de candidatos na descoberta reativa. A seção 5.3.3 compara o projeto combinado envolvendo o RbMAC e o roteamento baseado no receptor com outras soluções para a comunicação de dados.

### 5.3.1 Parâmetros de Simulação

Todos os modelos foram implementados no *Network Simulator 2* [2] e os resultados de simulação correspondem à média aritmética de  $n$  simulações, onde  $n$  é o menor tamanho de amostra que proporciona o intervalo de confiança desejado. O nível de confiança é de 95% e utilizou-se o *teste-T* [64] com 0,05 de significância para afirmar que um protocolo é melhor ou pior que outro para uma dada métrica.

Em todas as simulações, utilizou-se uma rede com 250, 500, 750 ou 1000 nós sensores. Esse parâmetro é constante em cada simulação, mas variado entre elas. Os nós sensores são estáticos, homogêneos e com uma capacidade de energia finita e não renovável. O consumo de energia de cada nó é baseado no Mica2 [1] e sua energia inicial foi suficiente para que nenhum nó morresse por falta de energia durante a simulação. O ciclo de trabalho é estático e homogêneo durante todo o tempo de vida da rede. Este trabalho considerou dois cenários de simulação, um no qual os nós operam com um ciclo de trabalho reduzido de 1% e outro em que esse valor é de 80%. O primeiro cenário é crucial para RSSFs dado que a melhor forma de se economizar energia nessas redes é fazer com que os nós sensores operem com um ciclo de trabalho reduzido. O segundo também é importante porque alguns requisitos de qualidade

como, por exemplo, latência reduzida e vazão elevada normalmente são obtidos quando a rede opera com ciclo de trabalho elevado. O raio de comunicação dos nós sensores é de 100 m, eles foram depositados de forma aleatória em uma área  $1000 \times 1000 \text{ m}^2$  e cada nó conhece a sua localização. O desafio da localização é um tópico que vem recebendo atenção especial dos pesquisadores em RSSFs e suas soluções encontram-se em desenvolvimento [8]. Representou-se a topologia da rede como um grafo conexo.

A rede também contém um nó monitor sem restrição de recursos, localizado no canto inferior esquerdo da rede e cujo ciclo de trabalho é de 100%. O nó monitor efetua 500 disseminações *unicasting* geradas através de uma distribuição uniforme de probabilidade. O destino de cada disseminação é um nó escolhido aleatoriamente em três regiões circulares. A figura 5.4 mostra essas regiões (ilustradas em cinza) cujos centros são equidistantes do nó monitor e o raio é igual ao de comunicação dos nós sensores. Além disso, o tempo de simulação é de 200 s. Contudo, a primeira disseminação acontece somente após 10 s para que os protocolos avaliados façam suas inicializações e a última acontece até 150 s para que todo o roteamento seja realizado.

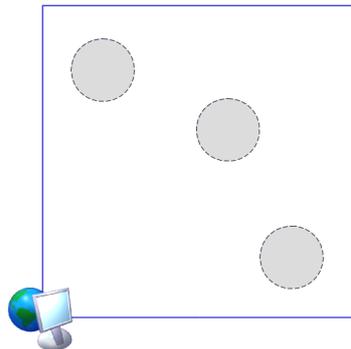


Figura 5.4: Regiões nas quais os nós destinos estão localizados

Esta seção avalia projetos para comunicação de dados combinando as políticas MAC pró-ativas e reativas e as abordagens de roteamento baseadas no emissor e no receptor. O objetivo dessas avaliações é verificar o desempenho de um projeto combinado entre a política MAC reativa e o roteamento baseado no receptor em relação às demais combinações. Os protocolos considerados para a avaliação do RbMAC são o OMAC [43, 44], AMAC [83], TEDD [41, 86] e TBF [99]. Escolheu-se o pró-ativo OMAC e o reativo AMAC porque eles são protocolos eficientes que usam a mesma técnica para tratar de colisões. Este capítulo considera que todos os quadros têm o mesmo tamanho. A principal diferença entre o OMAC e AMAC está em suas políticas para determinar o escalonamento de trabalho do próximo nó. Como o OMAC compartilha o escalonamento de trabalho dos nós vizinhos, este trabalho seguiu a proposta de Guo et al. [43, 44] para minimizar o custo de compartilhamento na qual cada nó sensor envia

seu escalonamento somente para uma fração de vizinhos. Entretanto, como essa fração depende da densidade de nós no campo de sensoriamento, para as simulações com 250, 500, 750 e 1000 nós, as frações de compartilhamento do OMAC são 100%, 50%, 40% e 30%, respectivamente. Escolheu-se empiricamente essas frações porque elas são os menores valores que não reduziram a taxa de entrega do OMAC. A fração de compartilhamento desse protocolo reduz a taxa de entrega se os nós sensores descartarem pacotes porque não conhecem o escalonamento dos seus vizinhos. O compartilhamento de informações do OMAC acontece no início de cada simulação. Na camada de rede, considerou-se o TBF para o roteamento baseado no emissor e o TEDD para o baseado no receptor. Esses protocolos foram escolhidos por serem eficientes em termos de energia e projetados para a disseminação de dados em RSSFs. A diferença entre eles está na forma de escolha do próximo nó. A figura 5.5 ilustra as soluções para comunicação de dados avaliadas neste trabalho envolvendo protocolos MAC pró-ativo e reativo e de roteamento baseados no emissor e no receptor.

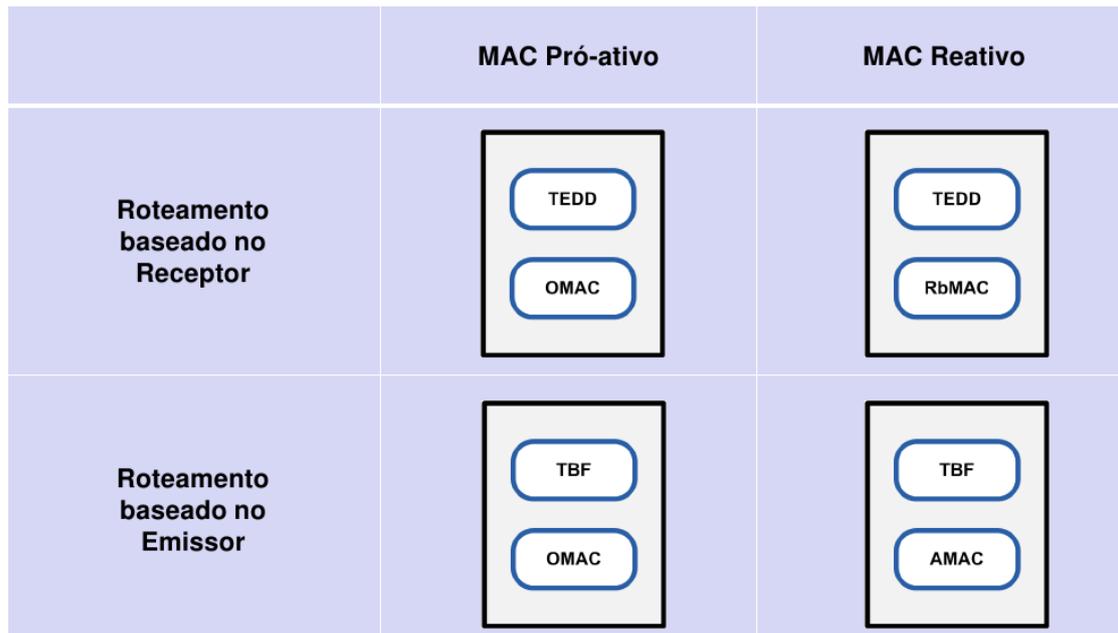


Figura 5.5: Projetos avaliados para a comunicação de dados

As soluções avaliadas neste capítulo consideram algumas interações de projeto integrado entre as camadas de rede e de acesso ao meio. No projeto RbMAC-TEDD, como a decisão do TEDD é baseada em atraso, o RbMAC não permite que os nós sensores adormeçam durante a decisão de propagação. O projeto OMAC-TBF usa um esquema no qual o TBF acessa as informações sobre escalonamento de trabalho disponíveis no OMAC para escolher o primeiro nó que acordar como o próximo nó. O projeto OMAC-TEDD usa uma técnica na qual quando o OMAC tem que enviar um

quadro, ele acessa a referência sobre próximo nó inserida pelo TEDD e envia o quadro somente quando um dos possíveis candidatos estiver acordado.

### 5.3.2 Número de Candidatos na Descoberta Reativa

Esta seção avalia a influência do número de candidatos a próximo nó na descoberta reativa do RbMAC. As figuras 5.6 a 5.8 mostram o desempenho do projeto combinado entre o RbMAC-TEDD à medida que os números de candidatos a próximo nó e de nós sensores aumentam para os cenários com ciclo de trabalho elevado e reduzido. Os dois parâmetros afetam a descoberta do RbMAC. As métricas avaliadas são a taxa de entrega, número de transmissões, consumo de energia e latência de entrega. A taxa de entrega significa a porcentagem de mensagens entregues aos seus destinos e ela é fundamental porque as mensagens podem ser perdidas durante a disseminação. Essa porcentagem está relacionada ao número total de mensagens enviadas até o final da simulação. O número de transmissões e o consumo de energia indicam o custo dos protocolos. A latência é um parâmetro crucial porque o atraso pode causar congestionamentos e perdas, principalmente quando os nós operam com um ciclo de trabalho reduzido no qual os emissores esperaram mais tempo até que o próximo nó acorde.

A figura 5.6 compara os resultados teórico (equação (5.2)) e de simulação para o número de transmissões realizadas por cada nó emissor à medida que se aumenta o número de candidatos a próximo nó. As figuras 5.6-a e 5.6-b apresentam os resultados para o cenário com ciclo de trabalho reduzido quando a rede tem 1000 e 500 nós sensores, respectivamente. As duas figuras comprovam a hipótese deste capítulo na qual o aumento do número de candidatos a próximo nó reduz o número de transmissões na descoberta reativa. Quando a rede tem 1000 nós sensores, as instâncias do projeto combinado com 1, 4 e 8 candidatos fazem respectivamente 5,6, 2,3 e 1,4 vezes mais transmissões que a com 12 candidatos. Os resultados de simulação quando a rede tem 1000 nós sensores são praticamente os mesmos que os teóricos, contudo, quando a rede tem 500 nós, a simulação transmite mais pacotes que o resultado esperado pela teoria. Isso porque o número de vizinhos de cada nó emissor normalmente é menor que o número de candidatos definido na instância do projeto combinado, fazendo com que o número de nós sensores na rede limite o número de candidatos. Na rede com 500 nós, devido à limitação da densidade da rede, quanto maior o número de candidatos, maior a diferença entre os números de transmissões teóricos e de simulação. Por exemplo, com 500 nós, as instâncias com 8 e 12 candidatos fazem o mesmo número de transmissões. Outro motivo para o número de transmissões nas simulações ser maior que o teórico são as retransmissões causadas devido às colisões. O modelo teórico da equação (5.2)

não considera colisões. As figuras 5.6-c e 5.6-d apresentam os resultados para o cenário com ciclo de trabalho elevado no qual o número de candidatos não influencia o número de transmissões. Isso porque os nós sensores ficam quase sempre acordados, o que permite ao protocolo de roteamento fazer normalmente uma transmissão para cada escolha de próximo nó. Aplicando o *teste-T* com 0,05 de significância não é possível determinar qual é o número de candidatos que efetua menos transmissões, contudo, os resultados de simulação fazem mais transmissões que o teórico. Isso aconteceu por causa das colisões existentes nas simulações.

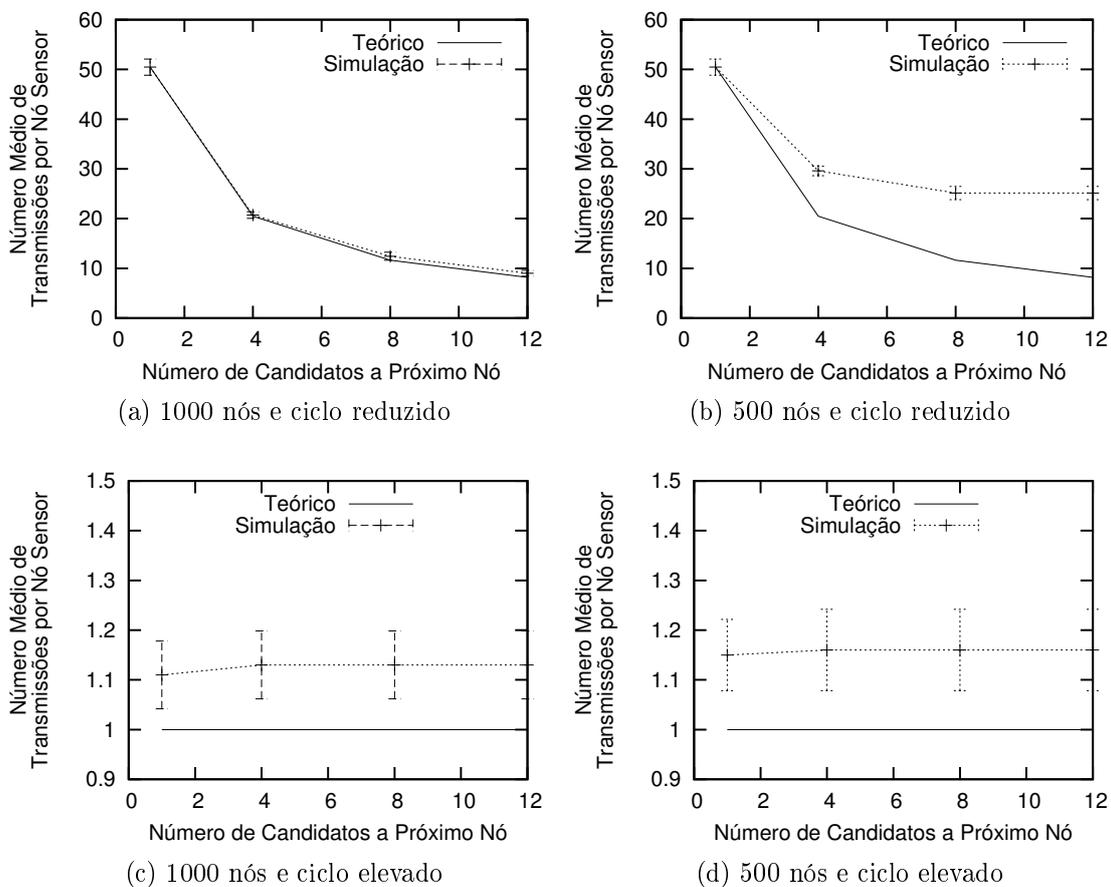


Figura 5.6: Resultados teóricos e de simulação do projeto combinado RbMAC-TEDD para o número de transmissões por nó sensor

A figura 5.7 apresenta as métricas avaliadas para as instâncias do projeto combinado proposto neste capítulo com 1, 4, 8 e 12 candidatos a próximo nó no cenário com ciclo de trabalho reduzido à medida que o número de nós na rede aumenta. A instância com um candidato sempre apresenta a menor taxa de entrega, o maior número de transmissões, consumo de energia e latência. As demais instâncias entregaram 1,1 vezes mais pacotes que a com um candidato. O número de transmissões, o consumo de

energia e a latência da instância com um candidato são 2,17, 1,12 e 1,35 vezes maiores que os resultados das demais, respectivamente. Esses resultados acontecem porque o custo da descoberta de um escalonamento de trabalho específico é maior que o de descobrir essa informação para algum vizinho, aumentando o número de transmissões, consumo de energia e latência. Os aumentos do número de transmissões e latência implicam no aumento de colisões, o que reduz a taxa de entrega. Quando a rede tem 1000 nós, a instância com um candidato faz 5,6 vezes mais transmissões, consome 1,12 vezes mais energia e tem uma latência 2,43 vezes maior que a instância com doze candidatos. Além disso, essa apresenta uma taxa de entrega 1,03 vezes maior que a instância com um candidato. Quando a rede possui 250 nós, todas as instâncias avaliadas apresentam as menores taxas de entrega e os maiores números de transmissões, consumo de energia e latência. O resultado para a taxa de entrega acontece porque a rede é esparsa e, por isso, o protocolo de roteamento nem sempre encontra uma rota entre o nó monitor e o destino das disseminações. Os resultados para o número de transmissões, consumo de energia e a latência são consequência do limite ao número de candidatos imposto pela densidade da rede esparsa. Essa limitação também explica o motivo das instâncias com 8 e 12 candidatos apresentarem os mesmos resultados para todas as métricas avaliadas quando a rede tem 500 nós sensores. O aumento do número de nós na rede minimiza essa limitação.

A figura 5.8 mostra as métricas avaliadas para o cenário com ciclo de trabalho elevado no qual se observa uma redução do número de transmissões e da latência e um aumento do consumo de energia e da taxa de entrega em relação ao cenário anterior. A latência no cenário com ciclo de trabalho reduzido é quase cem vezes maior que a deste cenário e o consumo de energia deste cenário é quase cem vezes maior que a do cenário anterior. Esses resultados são porque os nós sensores ficam acordados por mais tempo, o que reduz o número de transmissões necessárias para a descoberta reativa e faz com que cada nó intermediário espere menos tempo até que o próximo nó acorde. Aplicando o *teste-T* com 0,05 de significância, o RbMAC não apresenta qualquer diferença entre as instâncias e métricas avaliadas. Isso porque como todos os nós ficam quase sempre acordados, o primeiro candidato é quase sempre a opção escolhida.

Os resultados de simulação desta seção revelam que, para o cenário avaliado, o aumento do número de candidatos minimiza o número de transmissões, o consumo de energia e a latência da descoberta reativa do RbMAC quando os nós operam com ciclo de trabalho reduzido. Contudo, o aumento do número de candidatos é indiferente para o ciclo de trabalho elevado. A principal vantagem do cenário com ciclo de trabalho elevado em relação ao reduzido é a latência. Esta seção também mostra que quando a rede é esparsa, o número de vizinhos limita o de candidatos.

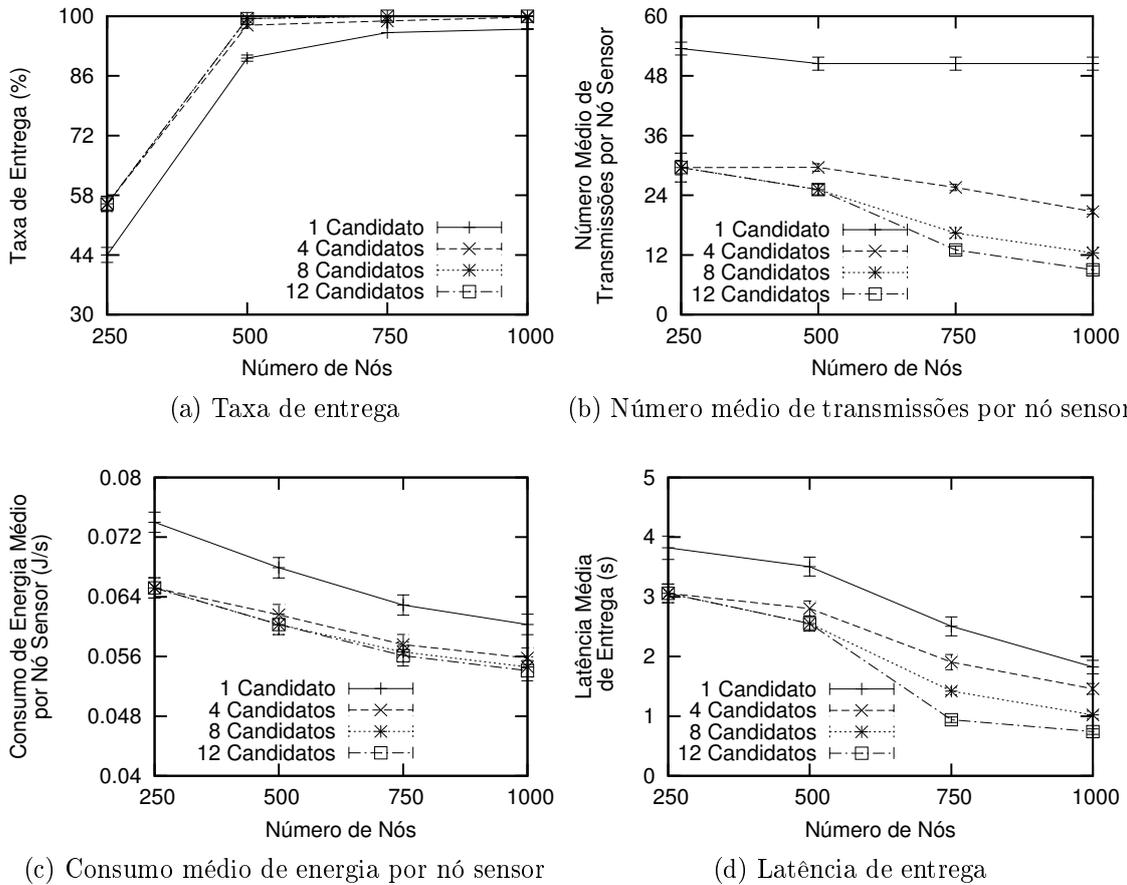


Figura 5.7: Desempenho do projeto combinado RbMAC-TEDD com 1, 4, 8 e 12 candidatos a próximo nó no cenário com ciclo de trabalho reduzido

### 5.3.3 Projetos para Comunicação de Dados

Esta seção compara projetos para a comunicação de dados envolvendo as abordagens MAC pró-ativa e reativa com as de roteamento baseadas no emissor e no receptor. O projeto combinado entre a política reativa e o roteamento baseado no receptor é o RbMAC-TEDD com 12 candidatos. Os projetos que possuem a política pró-ativa são o OMAC-TBF e o OMAC-TEDD e o combinando a política reativa e o roteamento baseado no emissor é o AMAC-TBF. A figura 5.5 ilustra os quatro projetos para comunicação de dados avaliados nesta seção. As métricas avaliadas são as mesmas da seção anterior, contudo, como o protocolo pró-ativo faz com que cada nó sensor compartilhe seu escalonamento de trabalho, a métrica de transmissões desta seção é o número total em vez do médio por nó emissor.

A figura 5.9 mostra os resultados de simulação para o cenário com ciclo de trabalho reduzido à medida que o número de nós na rede aumenta. Todos os pro-

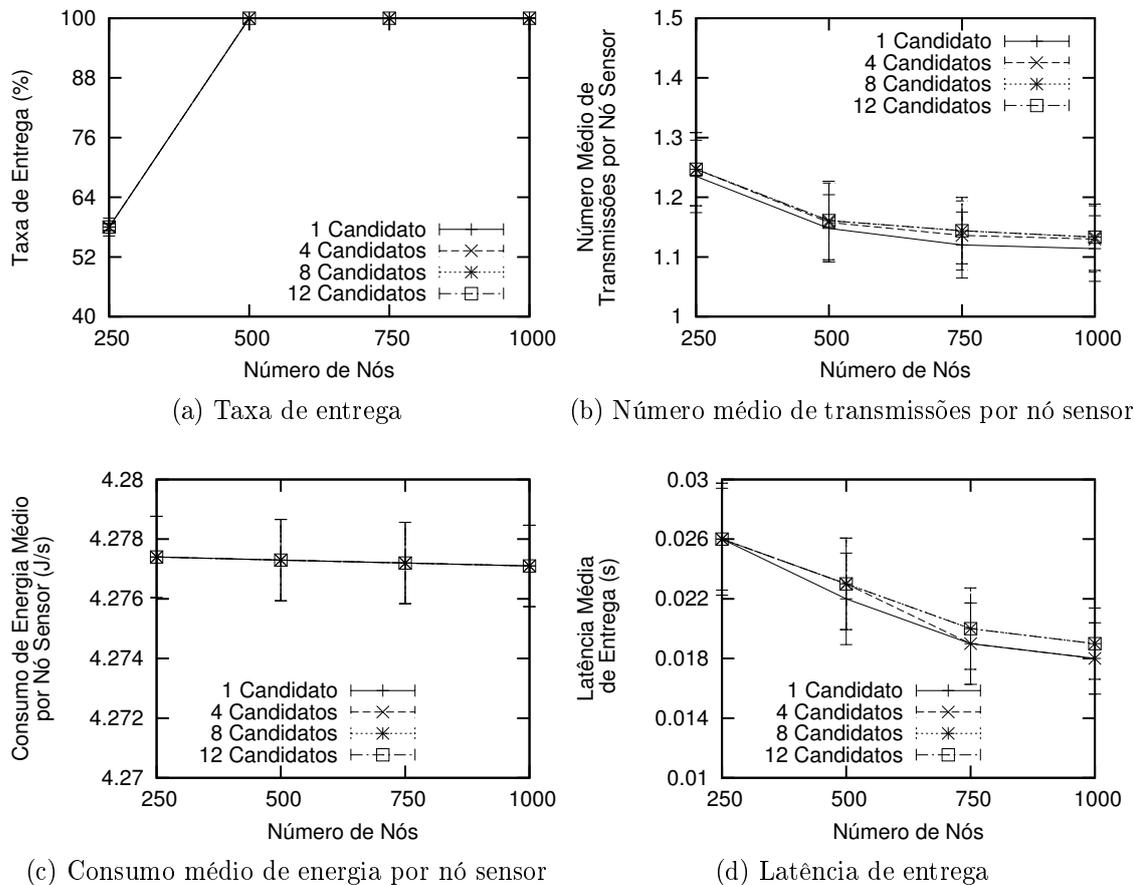


Figura 5.8: Desempenho do projeto combinado RbMAC-TEDD com 1, 4, 8 e 12 candidatos a próximo nó no cenário com ciclo de trabalho elevado

jetos avaliados entregam praticamente todas as mensagens, exceto quando a rede tem 250 nós ou no projeto AMAC-TBF. O resultado da rede com 250 nós aconteceu porque a mesma é esparsa, fazendo com que nem sempre exista uma rota entre os nós monitor e destino. O resultado do projeto AMAC-TBF é consequência do número de transmissões necessárias para a descoberta reativa encontrar o escalonamento de trabalho de um vizinho específico. O aumento do número de transmissões implica em mais colisões, o que reduz a taxa de entrega. O projeto AMAC-TBF efetua 3,83 vezes mais transmissões que os demais projetos. Na figura 5.9-b, o número de transmissões dos projetos contendo o protocolo pró-ativo OMAC aumenta à medida que o número de nós aumenta por causa do maior número de nós para compartilhar informações. Os projetos RbMAC-TEDD e AMAC-TBF fizeram menos transmissões quando a rede tem 250 nós devido ao fato dela ser esparsa, o que interrompe frequentemente o roteamento de pacotes. O RbMAC-TEDD faz mais transmissões quando a rede tem 500 nós que quando ela tem 1000 nós porque no primeiro caso, a densidade da rede limita o número

de candidatos a próximo nó de tal forma que o número efetivo de candidatos quase sempre é menor que os doze definidos nesse projeto. Com 500 nós, o RbMAC-TEDD faz 2,19 vezes mais transmissões que os projetos envolvendo o OMAC. Com 1000 nós, esses fazem 1,42 vezes mais transmissões que o RbMAC-TEDD. Isso mostra que o custo de compartilhamento dos projetos contendo o OMAC é menor para as redes esparsas e o de descoberta reativa combinado com o roteamento baseado no receptor é menor para as redes mais densas. Na figura 5.9-c, o consumo de energia por nó sensor de todos os projetos avaliados diminuiu com o aumento do número de nós porque são mais nós para dividir o custo de energia com as transmissões. Na rede com 1000 nós, todos os protocolos apresentam o mesmo consumo de energia (aplicando o *teste-T* com 0,05 de significância). Na figura 5.9-d, a latência dos projetos envolvendo o protocolo pró-ativo OMAC é 3,1 vezes maior que a do RbMAC. Isso porque no OMAC, cada nó conhece o escalonamento de trabalho de apenas uma fração de seus vizinhos, o que reduz a disponibilidade de candidatos para próximo nó. A latência do projeto AMAC-TBF é 1,93 vezes maior que a do RbMAC-TEDD porque o primeiro faz com que cada nó emissor espere por um próximo nó específico e o segundo, por algum vizinho que possa ser o próximo nó.

A figura 5.10 mostra os resultados de simulação para o cenário com ciclo de trabalho elevado quando o número de nós na rede aumenta. Os resultados da taxa de entrega são similares aos do cenário anterior, exceto para o projeto AMAC-TBF no qual o aumento do ciclo de trabalho reduz o custo de descoberta com transmissões e, conseqüentemente, minimiza a ocorrência de colisões e faz com que o AMAC-TBF entregue praticamente todos os pacotes. A figura 5.10-b mostra que todos os projetos avaliados fazem menos transmissões que o cenário anterior. Esse resultado acontece porque como os nós ficam acordados por mais tempo, as descobertas reativas (RbMAC e AMAC) e o compartilhamento (OMAC) necessitam de menos transmissões. Neste cenário, o protocolo pró-ativo OMAC faz com que cada nó sensor efetue somente uma transmissão de compartilhamento e essa é suficiente para que quase todos os vizinhos de um nó recebam seu escalonamento de trabalho. Os projetos OMAC-TBF, OMAC-TEDD e AMAC-TBF transmitem 2,86 vezes mais pacotes que o RbMAC-TEDD. Isso acontece principalmente por causa do mecanismo de confirmação baseado em *piggyback* do RbMAC que não envia quadros de controle como nos demais protocolos. Aplicando o *teste-T* com 0,05 de significância não é possível identificar diferenças de latência nem de consumo de energia dos projetos avaliados, exceto que a latência do AMAC-TBF é 1,61 vezes maior que a dos demais projetos. Como mostrado na seção 5.3.2, com ciclo de trabalho elevado, tem-se um aumento no consumo de energia e uma redução na latência em relação aos resultados do cenário anterior.

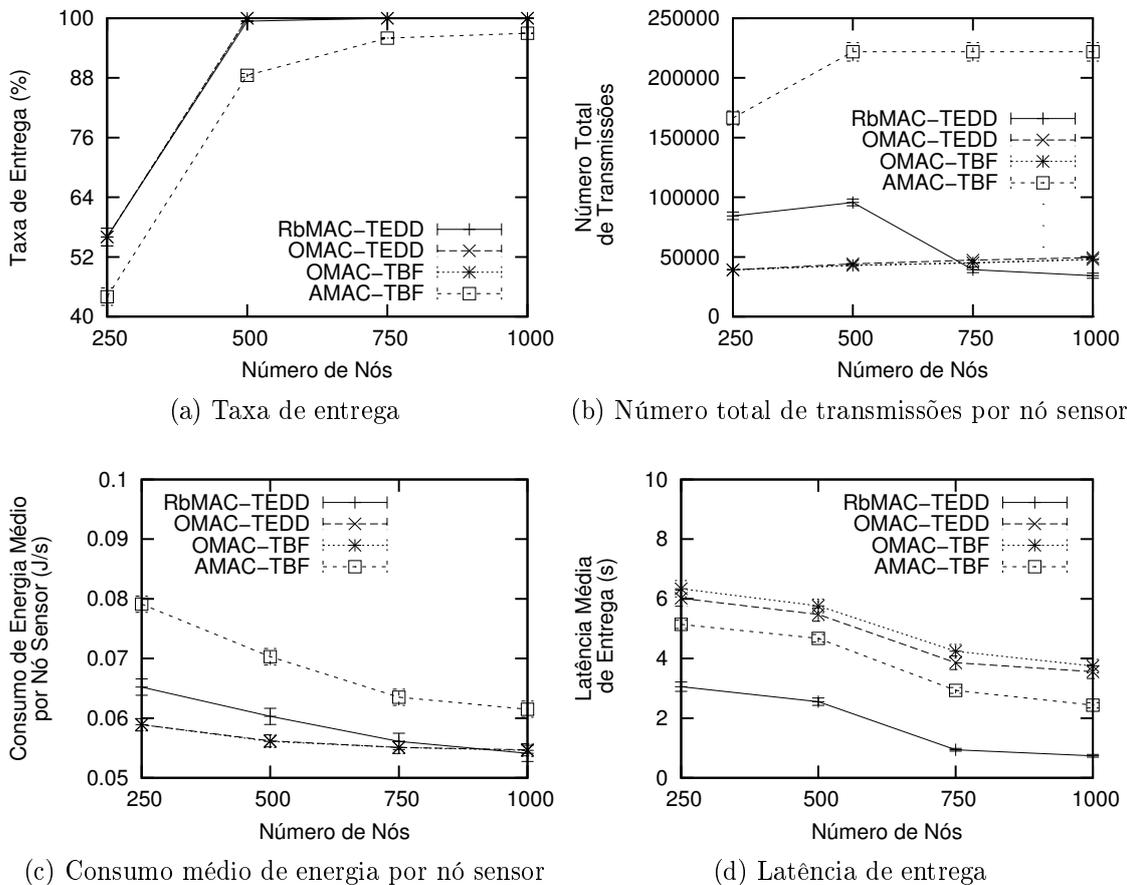


Figura 5.9: Métricas avaliadas para os projetos de comunicação de dados no cenário com ciclo de trabalho reduzido

Os resultados de simulação desta seção revelam que o projeto combinado RbMAC-TEDD transmite menos pacotes que os outros projetos avaliados quando a rede é densa ou os nós sensores operam com um ciclo de trabalho elevado. Novamente, a melhor forma de economizar energia é quando os nós operam com um ciclo de trabalho reduzido, contudo, a latência desse cenário é maior que no cenário com ciclo elevado.

## 5.4 Projeto Integrado para Redução da Latência e do Consumo de Energia

O ambiente dinâmico das RSSFs exige que as soluções para essas redes sejam adaptativas e trabalhem de acordo com os requisitos correntes da aplicação. Quando a rede tiver atividades reduzidas de sensoriamento ou comunicação, os nós sensores devem operar com um ciclo de trabalho reduzido para economizar energia. Contudo, quando

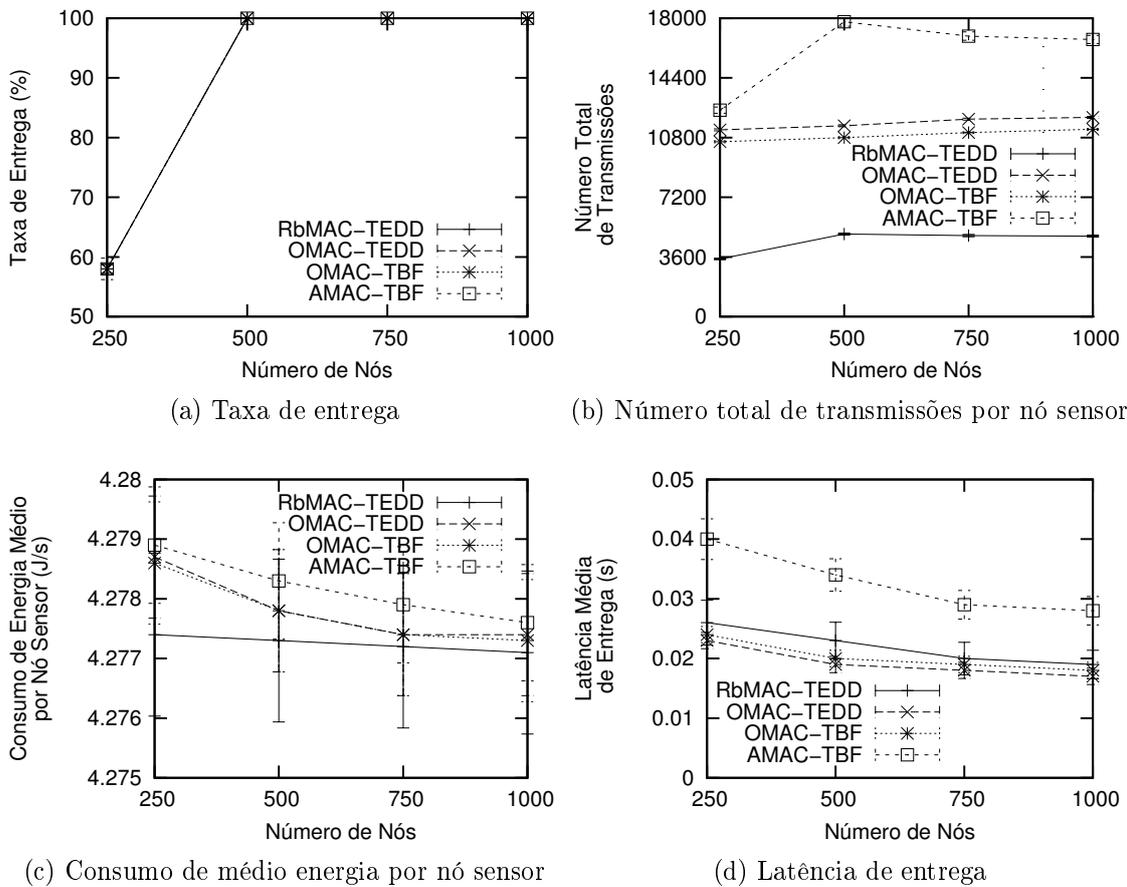


Figura 5.10: Métricas avaliadas para os projetos de comunicação de dados no cenário com ciclo de trabalho elevado

a aplicação exigir latência reduzida ou vazão elevada, os nós sensores devem operar com um ciclo elevado para prover os requisitos desejados pela aplicação. As camadas superiores da pilha de protocolos podem alterar o ciclo de trabalho dos nós sensores através de interações entre camadas.

Esta seção propõe e avalia um projeto integrado entre o RbMAC e a camada de rede para disseminar dados com a latência e consumo de energia reduzidos. A ideia da solução proposta é que todos os nós da rede sejam configurados com um ciclo de trabalho reduzido e quando a aplicação exigir o requisito de latência reduzida, aumentase o ciclo de trabalho de alguns nós para criar um caminho especial de roteamento. A limitação dessa solução é que o primeiro pacote roteado em cada novo caminho terá uma latência proporcional ao ciclo de trabalho padrão da rede.

### 5.4.1 Funcionamento Básico

A solução proposta consiste nos algoritmos 1 e 2. A camada de rede executa o primeiro algoritmo sempre que um nó propagar um pacote de dados. O objetivo desse algoritmo é criar um caminho de roteamento composto por nós com ciclo de trabalho elevado. No primeiro passo (linha 1), a camada de rede verifica se o nó está fora dos caminhos especiais. Se ele estiver, a camada de rede efetua uma interação entre camadas para aumentar o ciclo de trabalho do nó corrente disponível no RbMAC (linha 2). Em seguida, ela elimina o atraso da decisão de roteamento para qualquer pacote do novo caminho (linha 3). Como explicado na seção 2.2.4, alguns protocolos de roteamento baseados no receptor (e.g., TEDD) fazem com que cada nó intermediário insira atrasos em suas respectivas decisões. No passo seguinte (linha 4), a camada de rede determina quando o caminho expira, escalonando um temporizador de retorno. Por outro lado, quando um nó não pertence a algum caminho (linha 5), a camada de rede apenas atrasa o término do temporizador de retorno (linha 6). Quando esse temporizador expirar, a camada de rede executa o algoritmo 2, que restaura os valores originais da decisão de propagação (linha 1) e o ciclo de trabalho (linha 2).

---

#### Algoritmo 1 Início do ciclo de trabalho dinâmico

---

- 1: **se** eu não pertencço a um caminho especial de roteamento **então**
  - 2:   Aumentar o meu ciclo de trabalho no RbMAC
  - 3:   Eliminar o meu atraso para a propagação de pacotes do caminho
  - 4:   Escalar o tempo de retorno
  - 5: **senão**
  - 6:   Reescalonar o tempo de retorno
  - 7: **fim se**
- 

---

#### Algoritmo 2 Término do ciclo de trabalho dinâmico

---

- 1: Retornar o meu atraso original para a propagação de pacotes do caminho
  - 2: Retornar o meu ciclo de trabalho original no RbMAC
- 

### 5.4.2 Resultados de Simulação

Esta seção avalia o projeto combinado RbMAC-TEDD com e sem a técnica de interação entre camadas proposta nesta seção. O cenário de simulação é igual ao descrito na seção 5.3.1, contudo, o nó monitor envia dados sempre para os nós localizados em uma das três regiões da figura 5.4. A escolha dessa região acontece aleatoriamente antes de cada simulação. As disseminações periódicas para uma mesma região acontecem, por exemplo, quando o nó monitor deseja enviar sistematicamente dados para nós

localizados em posições estratégicas ou para nós responsáveis por tarefas estáticas. Os nós sensores operam com um ciclo de trabalho de 1%, contudo, quando um nó decide participar de um caminho especial de roteamento, ele aumenta seu ciclo de trabalho para 80%. O projeto com ciclo de trabalho dinâmico será comparado com as versões usando ciclos estáticos com 1% e 80%.

A tabela 5.1 apresenta o consumo de energia e a latência média de entrega para o projeto RbMAC-TEDD. A versão com ciclo de trabalho dinâmico apresenta uma latência próxima à versão estática com ciclo de trabalho elevado e um consumo de energia próximo ao da versão estática com ciclo de trabalho reduzido. A latência reduzida da versão dinâmica acontece devido aos nós com ciclo de trabalho elevado e, também, porque o TEDD não insere atrasos. O consumo reduzido da técnica proposta nesta seção acontece porque a maioria dos nós sensores opera com um ciclo de trabalho reduzido. Esta seção não apresenta a taxa de entrega e número de transmissões porque esses valores são similares aos mostrados na seção 5.3.

	<b>Latência de Entrega (s)</b>	<b>Consumo de Energia Médio por Nó Sensor(J/s)</b>
Ciclo de trabalho dinâmico	0,202	0,156
Ciclo de trabalho estático de 1%	0,745	0,061
Ciclo de trabalho estático de 80%	0,198	4,277

Tabela 5.1: Latência de entrega e consumo médio de energia por nó sensor

## 5.5 Conclusões

Este capítulo apresenta e avalia um projeto combinado envolvendo a política MAC reativa e o roteamento baseado no receptor. A ideia desse projeto combinado é usar a diversidade de candidatos a próximo nó do roteamento baseado no receptor para reduzir o custo de descoberta. O projeto proposto considera o RbMAC (também proposto neste capítulo) e o TEDD. O RbMAC não tem o custo de compartilhamento da política pró-ativa e se diferencia dos demais protocolos reativos propostos na literatura porque sua descoberta busca algum vizinho acordado e faz confirmações baseadas em *piggyback* de que cada próximo nó continuará o roteamento. O TEDD é um algoritmo eficiente para a disseminação de dados em RSSFs. Resultados teóricos e de simulação revelam que o projeto combinado proposto reduz o número de transmissões e a latência na comunicação de dados quando comparado com outras soluções avaliadas neste trabalho. Este capítulo também verifica formalmente algumas propriedades do

RbMAC e apresenta uma técnica de ciclo dinâmico para o RbMAC reduzir sua latência e consumo de energia em cenários específicos nos quais a mesma rota é utilizada sistematicamente.

# Capítulo 6

## *Gossiping Using the Energy Map*

*Devemos pagar caro pelos nossos erros se quisermos ver-nos  
livres deles, e depois podemos até dizer que temos sorte.  
Johann Wolfgang von Goethe*

Este capítulo propõe e avalia o protocolo *Gossiping Using the Energy Map* (GEM) para a difusão de dados em redes de sensores sem fio com ciclo de trabalho reduzido. O restante deste capítulo está organizado como mostrado a seguir. A seção 6.1 introduz o protocolo proposto e mostra sua motivação e objetivos. As seções 6.2 e 6.3 descrevem e avaliam o GEM, respectivamente. A seção 6.4 conclui este capítulo.

### 6.1 Introdução

A difusão de dados (*broadcasting*) é uma tarefa de comunicação crucial em RSSFs. Ela acontece quando o nó monitor deseja enviar uma informação para todos os nós sensores. A partir de algoritmos de difusão eficientes, o nó monitor realiza algumas atividades tais como ativar/desativar um ou mais sensores, enviar requisições/interesses para a rede e alterar o modo de operação dos nós sensores. Por exemplo, o nó monitor pode reduzir o ciclo de trabalho de nós sensores localizados dentro de regiões de baixa energia, preservando a energia dos mesmos. A difusão de dados também é fundamental para o funcionamento básico de vários protocolos para RSSFs tais como os de descoberta de localização, identificação de múltiplos caminhos entre nós e estabelecimento/manutenção de rotas. Os protocolos para a coleta de dados (e.g., *Directed Diffusion* [61]) normalmente também dependem de um protocolo de difusão de dados.

A difusão de dados em RSSFs é uma tarefa desafiadora quando os nós operam com ciclo de trabalho reduzido (1% ou menos) porque quando um nó envia um pacote, nem todos os seus vizinhos estão acordados. Eventualmente, todos os vizinhos de um

nó emissor podem estar adormecidos quando ele transmitir. A maioria das soluções propostas para difusão de dados nessas redes ignoram o fato que os nós podem ficar frequentemente adormecidos. Por outro lado, as soluções que consideram (e.g., [44, 123, 124]) transformam cada transmissão da difusão em várias do tipo *unicasting*. Nesse caso, cada nó emissor agenda uma transmissão *unicasting* para o momento em que cada um de seus vizinhos estiver acordado. Contudo, para isso, o nó emissor tem que conhecer o escalonamento de trabalho dos seus vizinhos. Essas soluções usam uma política MAC pró-ativa em que cada nó sensor compartilha seu escalonamento de trabalho com todos seus vizinhos. A principal limitação dessa política é o custo de compartilhamento gerado pelas trocas de mensagens que aumentam o consumo de energia da rede. Novas soluções para a difusão em RSSFs com ciclo de trabalho reduzido deveriam ser investigadas para minimizar o custo de compartilhamento.

Este capítulo propõe e avalia o protocolo *Gossiping using the Energy Map* (GEM) para difusão de dados em RSSFs com ciclo de trabalho reduzido. O mapa de energia é a informação sobre a energia restante nas diversas partes da rede [93]. A principal diferença entre o GEM e as demais soluções conhecidas para difusão de dados em redes de sensores sem fio com ciclo de trabalho reduzido é que o protocolo proposto neste capítulo faz com que cada nó sensor compartilhe seu escalonamento de trabalho com apenas um nó sensor. As demais soluções fazem com que cada nó sensor compartilhe seu escalonamento com todos seus vizinhos. O GEM cria uma árvore de difusão baseada no mapa de energia da rede e faz com que cada nó sensor compartilhe seu escalonamento de trabalho com apenas um nó vizinho que participa da árvore disseminando pacotes. O GEM também preserva a energia dos nós localizados dentro de regiões de baixa energia. O projetista de uma RSSF normalmente tem um cuidado especial com essas regiões porque o término de sua energia pode tornar a rede desconexa ou criar áreas sem sensoriamento. Resultados de simulação revelam que o GEM reduz o número de transmissões, consumo de energia e latência quando comparado com outras soluções propostas na literatura e avaliadas neste capítulo. Além disso, ele pode reduzir o fluxo de dados e o consumo de energia de nós sensores localizados dentro de regiões de baixa energia para prolongar o tempo de vida da rede.

## 6.2 Protocolo *Gossiping using the Energy Map*

Esta seção propõe o protocolo *Gossiping using the Energy Map* (GEM) e está organizada como descrito a seguir. A seção 6.2.1 descreve o funcionamento básico do GEM. Esse funcionamento pode ser modelado a partir de eventos e ações conforme descrito

na seção 6.2.2. A seção 6.2.3 apresenta como o protocolo proposto identifica regiões de baixa energia. Cada nó sensor decide probabilisticamente como será sua forma de participação na árvore de difusão e a seção 6.2.4 detalha o cálculo dessa probabilidade de decisão.

### 6.2.1 Funcionamento Básico

O funcionamento básico do GEM é baseado em uma árvore de difusão. Ele constrói essa árvore de forma probabilística a partir do mapa de energia da rede e faz com que cada nó sensor compartilhe seu escalonamento de trabalho com somente um nó vizinho que participa da árvore como nó interno. Cada nó interno armazena o escalonamento de trabalho de todos seus filhos e é responsável por enviar os pacotes de difusão quando cada um deles estiver acordado. Os nós folhas não armazenam informações sobre vizinhos. O nó monitor é a raiz da árvore e ele pode criá-la, recriá-la quando julgar necessário e utilizá-la em outras difusões. Outras aplicações também podem se beneficiar da árvore ou das informações sobre escalonamento de vizinhos compartilhadas. A figura 6.1 ilustra a árvore de difusão do GEM na qual os nós internos têm a cor cinza, as folhas são brancas e o nó monitor (raiz da árvore) está localizado no canto inferior esquerdo.

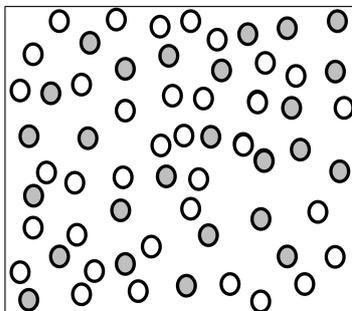


Figura 6.1: Exemplo de uma árvore de difusão do GEM

O nó monitor cria a árvore enviando um pacote de criação para seus vizinhos. Quando um nó sensor recebe esse pacote, ele decide se será um nó interno. Se verdadeiro, ele também envia o pacote de criação para seus vizinhos, aumentando o tamanho da árvore. Cada nó interno deve descobrir o escalonamento de trabalho dos seus vizinhos e agendar um momento em que esses acordarão simultaneamente para decidir quem também será nó interno. Nesse caso, cada nó vizinho acorda oportunisticamente no momento combinado, aguarda pela decisão conjunta e, depois, retoma normalmente seu escalonamento de trabalho na posição do ciclo operacional em que ele estaria se não tivesse participado da decisão. Essas descobertas e agendamentos são baseados na

retransmissão de um pacote para a criação da árvore. O projetista da rede define o número total de retransmissões e o intervalo entre elas a partir do ciclo de trabalho dos nós. O GEM constrói sua árvore com nós distantes entre si com o objetivo de reduzir o número de nós internos na árvore e, conseqüentemente, as descobertas e os agendamentos.

As figuras 6.2 e 6.3 ilustram a criação da árvore do GEM. A figura 6.2 mostra um exemplo intermediário da criação da árvore no qual os nós representados com bordas grossas participam da árvore sendo que os cinzas são nós internos e os brancos, folhas. Os nós brancos com borda fina ainda não receberam o pacote de criação e, por isso, não participam da árvore.

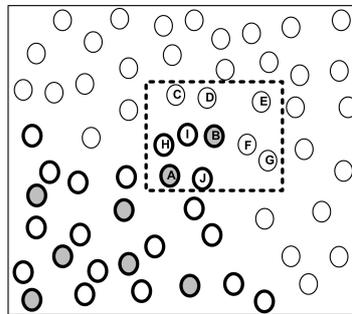


Figura 6.2: Exemplo intermediário da criação da árvore do GEM

A figura 6.3 apresenta um passo a passo da criação da árvore do GEM para os nós localizados dentro da área pontilhada na figura 6.2. A figura 6.3-a corresponde a um zoom dessa área pontilhada em que os nós internos A e B e as folhas H, I e J receberam anteriormente o pacote de criação. Na figura 6.2-b, C está acordado quando B envia o pacote de criação. Um dos parâmetros contidos no cabeçalho desse pacote é uma informação sobre agendamento indicando em quantas unidades de tempo o receptor deve acordar oportunisticamente para decidir em conjunto com os demais vizinhos do emissor quem também será nó interno. Assim, C agenda o momento em que deve acordar para decidir se será um nó interno ou uma folha. Na figura 6.2-c, D e E estão acordados quando B envia o pacote de criação novamente. Quando os dois recebem o pacote, eles agendam o momento em que devem acordar oportunisticamente. Na figura 6.2-d, F e G estão acordados, receberem o pacote de B e agendam o momento em que devem acordar. Na figura 6.2-e, os filhos de B acordam oportunisticamente no momento combinado. Como os nós A, H, I e J pertenciam a árvore antes das transmissões de B e não são filhos desse nó, eles não acordam com os demais. Nas figuras 6.2-f e 6.2-g, C decide ser um nó interno e envia o pacote de criação. Como D é vizinho de C, ele recebe o pacote e decide ser uma folha. Nas figuras 6.2-h e 6.2-i, G

decide participar da árvore e envia o pacote de criação. Como E e F são vizinhos de G, eles recebem o pacote e decidem que serão folhas.

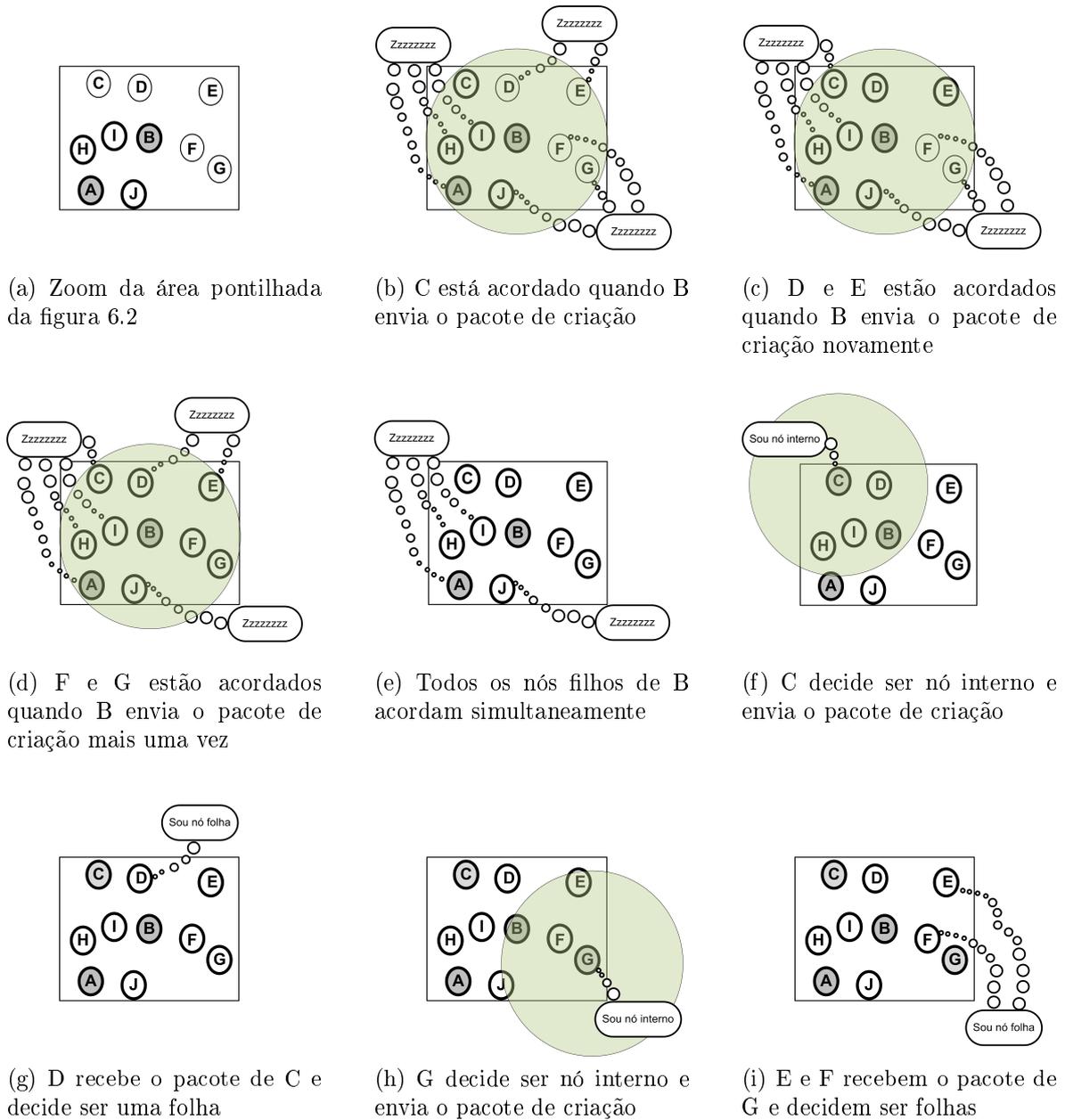


Figura 6.3: Exemplo de criação da árvore do GEM

## 6.2.2 Eventos e Ações

Esta seção modela o funcionamento do GEM baseado em eventos e ações de tal forma que, quando um evento acontece, ocorre uma ou mais ações correspondentes. Os

eventos considerados no GEM são “receber pacote”, “filiação” e “decisão”. As figuras 6.4 à 6.10 ilustram as ações correspondentes do GEM para cada um desses eventos.

O evento “receber pacote” acontece quando um nó recebe um pacote. O primeiro passo da ação de receber pacote é verificar se o nó corrente é o próximo nó do pacote recebido (figura 6.4, ponto *A*). Se verdadeiro, o nó analisa o tipo do pacote (figura 6.4, pontos *B*, *E* e *G*) e executa uma ação específica para receber cada um dos tipos de pacote (figura 6.4, pontos *C*, *F*, *H* e *I*). Em seguida, o nó termina a ação (figura 6.4, ponto *D*). Um pacote pode ser dos tipos dados, criação da árvore, filiação e reconhecimento. O primeiro contém dados de difusão. O nó monitor usa o segundo tipo para criar a árvore. O pacote de filiação permite que cada nó filho se apresente para seu respectivo pai que reconhecerá seu novo filho enviando um pacote de reconhecimento como resposta. Caso o nó corrente não seja o próximo nó do pacote recebido (figura 6.4, ponto *A*), esse nó apenas descarta o pacote (figura 6.4, ponto *J*). Observa-se que todos os nós da rede serão o próximo nó dos pacotes de dados, pois o GEM é um protocolo para a difusão de dados.

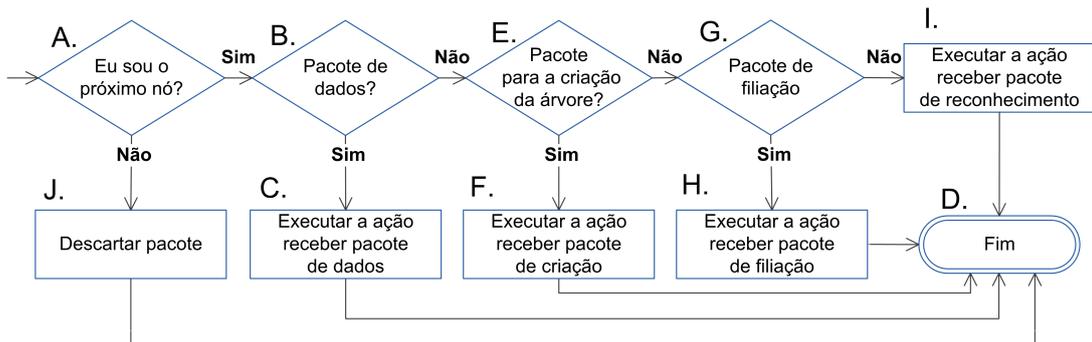


Figura 6.4: Ação receber pacote do GEM

A ação para receber um pacote de dados verifica se o pacote está sendo recebido pela primeira vez (figura 6.5, ponto *A*). Se verdadeiro, o GEM encaminha o pacote para a camada de aplicação (figura 6.5, ponto *B*). Em seguida, se o nó corrente é um nó interno (figura 6.5, ponto *C*), ele escalona o envio do pacote recebido para o momento em que cada um dos seus filhos estiver acordado (figura 6.5, ponto *D*) e termina ação (figura 6.5, ponto *E*). A ação para receber pacote de dados também termina quando o nó não é interno (figura 6.5, pontos *C* e *E*). Se o nó recebeu anteriormente o pacote (figura 6.5, ponto *A*), ele descarta o pacote recebido e termina a ação (figura 6.5, pontos *F* e *E*).

A ação para um receber pacote de criação também verifica se o pacote está sendo recebido pela primeira vez (figura 6.6, ponto *A*). Se verdadeiro, o nó corrente assume que o nó anterior do pacote é seu pai (figura 6.6, ponto *B*). Em seguida, ele executa

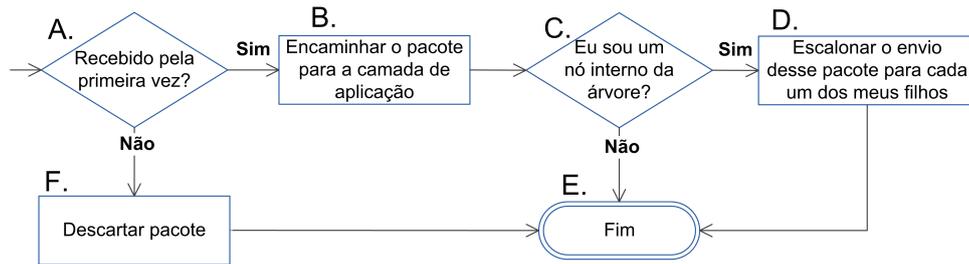


Figura 6.5: Ação receber pacote de dados do GEM

a ação “filiação” (figura 6.6, ponto *C*) na qual o nó envia um pacote de filiação para seu pai saber sobre o novo filho. O passo seguinte faz com que o nó corrente verifique se ele está localizado dentro de uma região de baixa energia (figura 6.6, ponto *D*). A seção 6.2.3 descreve como um nó faz essa verificação. Se o nó estiver localizado dentro de uma dessas regiões, ele adapta seu comportamento para economizar energia e prolongar o tempo de vida da rede (figura 6.6, ponto *E*). Por exemplo, o nó pode reduzir seu ciclo de trabalho para dormir por mais tempo ou decidir que não ser um nó interno para eliminar descobertas e agendamentos. Em seguida, o nó corrente verifica se sua distância em relação ao nó anterior do pacote é maior que um limiar de distância (figura 6.6, ponto *F*). Se verdadeiro, ele escalona o evento “decisão” para decidir se será um nó interno (figura 6.6, ponto *G*). Caso contrário, ele será uma folha (figura 6.6, ponto *I*) porque está localizado próximo ao anterior e suas descobertas seriam redundantes com as desse nó. O fato dos nós “próximos” ao anterior se tornarem folhas aumenta a distância média entre os nós internos, reduz o número desses nós na árvore e minimiza as descobertas e os agendamentos. Somente os nós internos realizam essas tarefas. Quando um nó decide ser folha, ele cancela eventuais escalonamentos do evento “decisão” (figura 6.6, ponto *J*) e termina a ação corrente (figura 6.6, ponto *H*). O nó deve cancelar esse evento porque sua decisão está tomada e ele pode ter escalonado tal evento quando recebeu o pacote de criação pela primeira vez.

As ações para receber os pacotes de filiação e reconhecimento e a ação correspondente ao evento “filiação” fazem com que cada nó sensor tenha um nó pai. Todo nó pai é um nó interno e é responsável por enviar os pacotes de difusão quando cada um de seus filhos estiver acordado. As figuras 6.7, 6.8 e 6.9 ilustram as três ações, respectivamente. A ação para receber uma pacote de filiação acontece porque o nó corrente enviou um pacote de criação e recebeu como resposta outro de filiação, significando que algum vizinho assumiu o nó corrente como pai. O primeiro passo dessa ação faz com que o nó corrente atualize sua lista de filhos (figura 6.7, ponto *A*) e escalone o envio de um pacote de reconhecimento destinado ao seu novo filho (figura 6.7, ponto *B*).

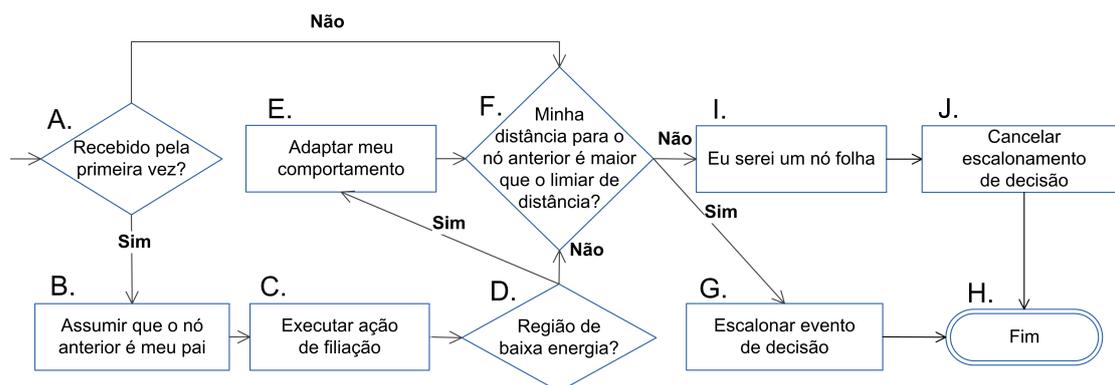


Figura 6.6: Ação receber pacote de criação do GEM

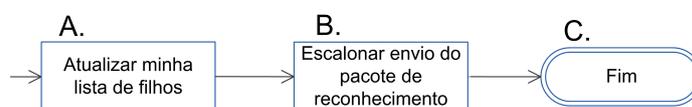


Figura 6.7: Ação receber pacote de filiação do GEM

A ação para receber um pacote de reconhecimento faz com que o nó corrente saiba que foi reconhecido por seu pai. Assim, ele marca esse reconhecimento (figura 6.8, ponto A) e cancela o escalonamento do evento “filiação”.

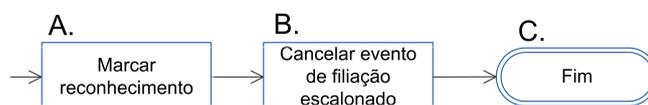


Figura 6.8: Ação receber pacote de reconhecimento do GEM

Um nó sensor executa a ação “filiação” quando recebe um pacote de criação pela primeira vez. Nessa ação, o nó corrente escalona o envio de um pacote de filiação para que o pai saiba sobre esse novo filho (figura 6.9, ponto A). Esse evento acontece no momento em que o pai estiver acordado. Isso é possível porque o cabeçalho do pacote de criação contém o escalonamento de trabalho do nó pai. O pacote de filiação também tem esse campo para que o pai saiba quando enviar pacotes para seu novo filho. Como mostrado no capítulo 4, as informações sobre escalonamento de trabalho podem ser descritas em alguns bytes. O próximo passo da ação “filiação” faz com que o nó corrente reescalone esse evento (figura 6.9, ponto B). Isso porque se o nó corrente não receber seu reconhecimento, ele precisa enviar o pacote de filiação novamente.

O evento “decisão” acontece simultaneamente em todos os vizinhos de um nó interno. Essa simultaneidade acontece porque quando um nó envia um pacote de criação, ele especifica no cabeçalho em quanto tempo seus vizinhos devem escalonar o evento

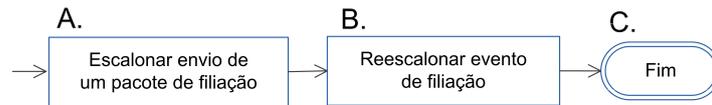


Figura 6.9: Ação de filiação do GEM

“decisão”. Além disso, quando esse evento acontece, o nó corrente acorda e permanece assim até tomar sua decisão. O primeiro passo da ação de decisão faz com que o nó corrente verifique se ele participa da árvore (figura 6.10, ponto *A*). Se verdadeiro, ele calcula sua probabilidade de decisão, escolhe um número aleatório e verifica se esse número é menor que sua probabilidade de decisão (figura 6.10, pontos *B*, *C* e *D*). A seção 6.2.4 mostra como cada nó sensor calcula essa probabilidade. Se o número aleatório for menor que a probabilidade de decisão (figura 6.10, ponto *D*), o nó corrente será um nó interno (figura 6.10, ponto *E*). Assim, ele escalona suas transmissões do pacote de criação e termina a ação corrente (figura 6.10, pontos *F* e *G*). Essas transmissões possibilitam que o novo nó interno faça suas descobertas e agendamentos. Quando o número aleatório for maior ou igual à probabilidade de decisão (figura 6.10, ponto *D*), o nó corrente reescalona o evento de decisão para acontecer em um *time step* (figura 6.10, ponto *H*) – tempo necessário para que um nó envie um pacote de criação e todos seus vizinhos que estiverem acordados recebam e tratem o pacote. Durante o tempo de espera, se um nó receber um pacote de criação e a distância entre esse nó e o emissor do pacote for menor que o limiar de distância, a ação “receber pacote de criação” faz com que o nó seja uma folha, terminando a decisão. Após o tempo de espera, se o nó não tomou sua decisão, ele executa novamente a ação corrente, ganhando uma nova oportunidade de decisão. O nó sensor reescalona o evento de decisão até que ele ou um vizinho “próximo” decida ser um nó interno. O projetista da rede pode definir um número máximo de reescalamentos, eliminando a possibilidade de uma repetição infinita. Quando o evento de decisão acontece em um nó interno, esse envia um pacote de criação (figura 6.10, pontos *A* e *I*), fazendo com que vizinhos próximos sejam folhas, o que minimiza o número de nós internos na árvore.

Como no capítulo 4, para que todos os vizinhos de um nó interno tenham a oportunidade de receber o pacote de criação com o menor número possível de retransmissões, cada nó interno escalona suas retransmissões baseadas em dois pressupostos. Primeiro, o tempo total do processo de retransmissões deve ser igual ao período em que um nó permanece adormecido em seu ciclo operacional, garantindo que todos os vizinhos do nó interno sempre acordam durante o processo. Segundo, o intervalo entre as retransmissões deve ser igual ao período em que os nós permanecem acordados em

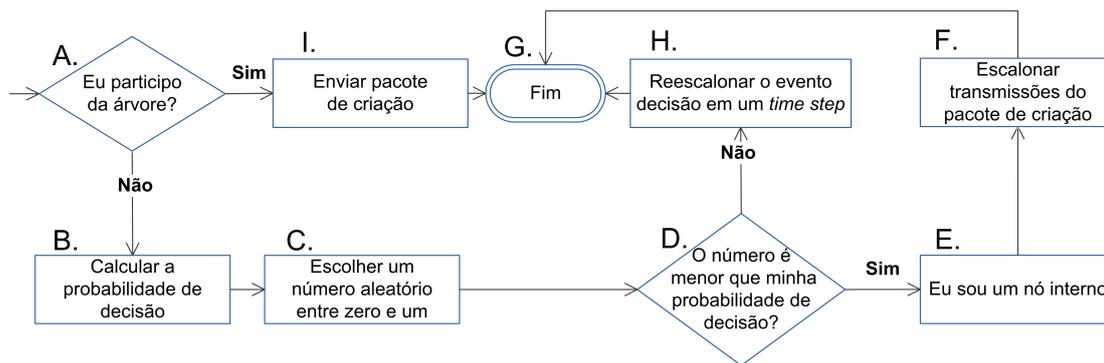


Figura 6.10: Ação de decisão do GEM

um ciclo operacional, garantindo que quando um vizinho acorda, o nó interno corrente sempre efetua pelo menos uma retransmissão antes desse vizinho voltar a dormir.

### 6.2.3 Regiões de Baixa Energia

Um dos principais desafios em redes de sensores sem fio são regiões de baixa energia porque o término da energia dos nós localizados dentro dessas regiões pode tornar a rede desconexa ou criar áreas sem sensoriamento. O protocolo proposto identifica tais regiões através de um parâmetro chamado de energia de corte. O nó monitor calcula esse valor a partir do mapa de energia da rede e o insere no cabeçalho dos pacotes de criação. Um nó sensor está localizado dentro de uma dessas regiões se sua energia for menor que a de corte. O restante desta seção mostra como o GEM constrói o mapa de energia e calcula a energia de corte.

#### 6.2.3.1 Construção do Mapa de Energia

O mapa de energia pode ser útil para preservar a energia das RSSFs. Como ilustrado na figura 6.11, ele pode ser representado em tons de cinza de forma que as regiões mais claras representam áreas com mais energia e as mais escuras, as com menos. Pode-se construí-lo a partir de uma abordagem ingênua em que cada nó envia periodicamente o valor de sua energia para o nó monitor. Contudo, o custo de comunicação dessa abordagem provavelmente a torna inviável para RSSFs. Zhao et al. [138] propõem uma solução baseada em agregação de dados. Mini et al. [93] apresentam outra solução baseada na previsão do consumo de energia dos nós sensores.

Este trabalho constrói o mapa de energia a partir da abordagem baseada em previsão cuja ideia é que cada nó sensor use um mecanismo baseado em Cadeia de Markov para prever de forma eficiente a quantidade de energia que ele consumirá no

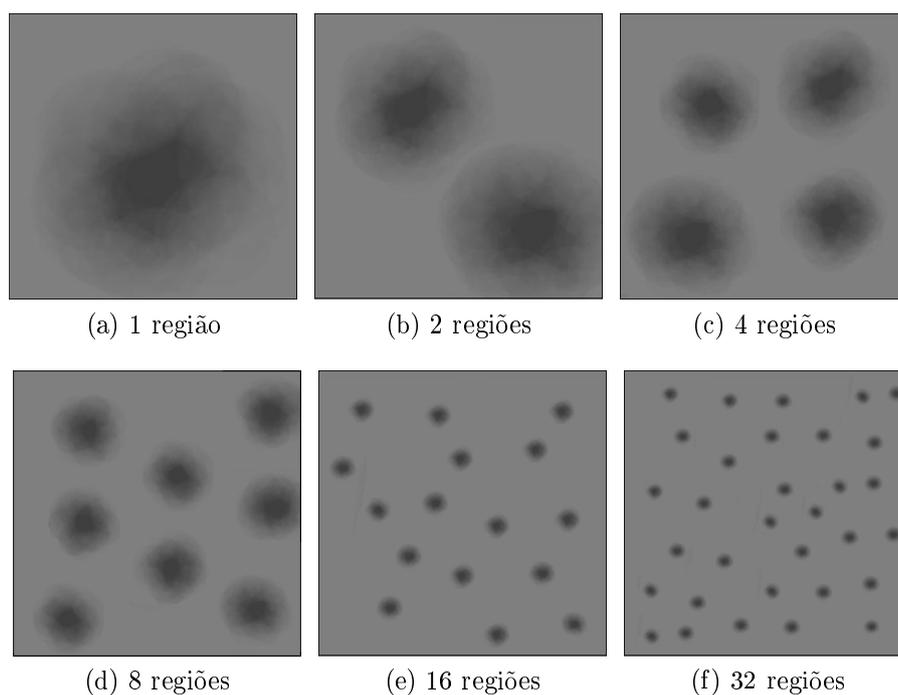


Figura 6.11: Exemplo de mapas de energia contendo regiões de baixa energia

futuro. Em seguida, cada nó sensor envia para o nó monitor os parâmetros descrevendo seu consumo de energia. Com essa informação, o nó monitor pode atualizar localmente sua informação sobre a energia corrente de cada nó sensor.

### 6.2.3.2 Cálculo da Energia de Corte

Antes de enviar um pacote de criação, o nó monitor calcula a energia de corte da rede e a insere no cabeçalho do pacote. A energia de corte é um limiar usado por cada nó sensor para determinar se ele está localizado dentro de uma região de baixa energia. Este trabalho considera que uma região é de baixa energia quando a energia média de todos os nós que a cobrem é menor que a energia média da rede ( $\bar{e}$ ) menos o desvio padrão ( $s$ ).

O nó monitor utiliza o mapa de energia para calcular  $\bar{e}$  e  $s$ . Em seguida, ele calcula um coeficiente de energia  $e_i$  para cada nó  $i$  que corresponde à soma da energia do nó  $i$  com a de todos os  $n$  vizinhos de  $i$  dividido por  $n + 1$ . Dado  $\bar{e}$ ,  $s$  e cada  $e_i$ , a energia de corte será igual ao maior coeficiente que seja menor que  $\bar{e} - s$ . Quando esse coeficiente não existe, a rede não tem regiões de baixa energia e a energia de corte é zero. A figura 6.12 ilustra um exemplo de como o GEM calcula a energia de corte. Nessa figura, o eixo x lista alguns nós sensores e o eixo y mostra o coeficiente de energia desses nós. O coeficiente de energia de cada nó está representado por uma bola preta.

As linhas pontilhada, tracejada e contínua representam  $\bar{e}$ ,  $\bar{e} + s$  e  $\bar{e} - s$ , respectivamente. Os nós localizados dentro de alguma região de baixa energia são o 1, 5, 7, 9 e 14. A energia de corte corresponde ao coeficiente do nono nó sensor.

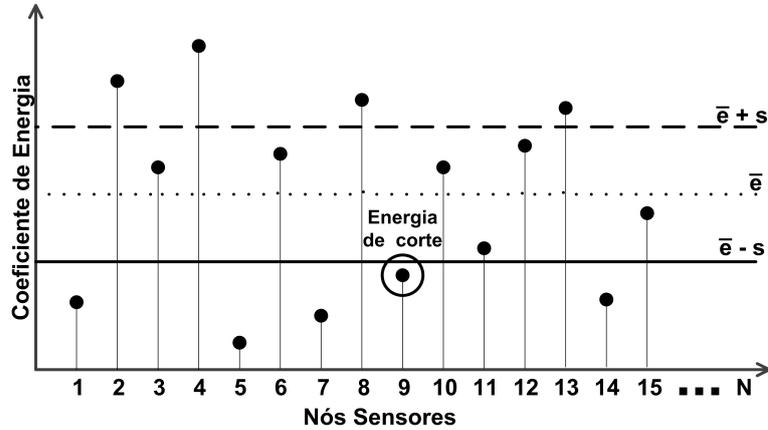


Figura 6.12: Exemplo de como o GEM calcula a energia de corte

#### 6.2.4 Probabilidade de Decisão

O GEM calcula a probabilidade de decisão de forma que os nós mais distantes do emissor do pacote e com as maiores reservas de energia tenham mais chances de pertencer à árvore de difusão. Aumentando a distância entre os nós, reduz-se o número de nós internos na árvore e, conseqüentemente, as descobertas e agendamentos. Além disso, o fato dos nós sensores com as maiores reservas de energia terem mais chances de serem nós internos balanceando a energia residual da rede.

A equação (6.1) mostra como calcular a probabilidade de decisão. O *raio de comunicação* corresponde à distância máxima que uma transmissão pode entregar um pacote, *distância* é a distância entre o nó corrente e o anterior, *minha energia* é a energia corrente do nó, *energia max* é o maior valor de energia que um nó operando com o ciclo de trabalho padrão pode ter dado o tempo de vida corrente da rede e  $p_{\max}$  é o maior valor possível da probabilidade de decisão.

$$probabilidade\ de\ decisão = \frac{distância}{raio\ comunicação} \times \frac{minha\ energia}{energia\ max} \times p_{\max} \quad (6.1)$$

## 6.3 Resultados de Simulação

Esta seção avalia o comportamento do GEM em cenários de difusão de dados em RSSFs com ciclo de trabalho reduzido e o compara com outras soluções para difusão propostas na literatura. Considerou-se cenários com e sem regiões de baixa energia. A seção 6.3.1 mostra os parâmetros de simulação. A seção 6.3.2 discute os resultados de simulação obtidos pelos protocolos avaliados quando a rede não contém regiões de baixa energia. A seção 6.3.3 possui regiões de baixa energia geradas no início da simulação.

### 6.3.1 Parâmetros de Simulação

Todos os protocolos foram implementados no *Network Simulator 2* [2] e os resultados de simulação correspondem à média aritmética de  $n$  simulações, onde  $n$  é o menor tamanho de amostra que proporciona o intervalo de confiança desejado. O nível de confiança é de 95% e utilizou-se o *teste-T* [64] com 0,05 de significância para afirmar que um protocolo é melhor ou pior que outro para uma dada métrica.

Em todas as simulações, utilizou-se uma rede com nós estáticos, homogêneos e com capacidade de energia finita e não renovável. O consumo de energia de cada nó é baseado no Mica2 [1], sua energia inicial é suficiente para que nenhum nó morra por falta de energia durante a simulação e seu raio de comunicação é de 100 m. Os nós operam com um ciclo de trabalho reduzido de 1% e foram depositados de forma aleatória em uma área  $1000 \times 1000$  m<sup>2</sup>. Considerou-se também que cada nó conhece sua localização [8]. O número de nós sensores no campo de sensoriamento é um parâmetro alterado em cada simulação. Esta seção avalia a rede com 250, 500, 750 ou 1000 nós sensores. Como a área de sensoriamento é constante, o aumento do número de nós sensores implica no aumento da densidade da rede (número médio de vizinhos) e do número de destinos para cada difusão.

As regiões de baixa energia são círculos cujos centros são escolhidos aleatoriamente e o comprimento do raio depende do número de regiões no campo do sensoriamento. Em todas as simulações, a soma das áreas dessas regiões é constante e igual a 10% do campo do sensoriamento. Como ilustrado na figura 6.11, aumentando-se o número de regiões no campo de sensoriamento, reduz-se o raio e a área das mesmas. A criação dessas regiões acontece no início da simulação fazendo com que a energia dos nós localizados dentro delas seja metade do valor inicial dos demais nós.

A rede contém um nó monitor sem restrição de recursos, localizado no canto inferior esquerdo da rede e cujo ciclo de trabalho é 100%. Considera-se que ele conhece as coordenadas de todos os nós sensores. Durante cada simulação, ele envia quinze

mensagens de difusão distribuídas uniformemente ao longo do tempo de simulação. O tempo total de simulação é de 1000 s. Contudo, a primeira difusão acontece somente após 50 s, permitindo a inicialização dos protocolos avaliados. A última acontece antes de 950 s para que a rede faça o roteamento de todos os pacotes.

Esta seção compara o GEM com o *Opportunistic Flooding* (OF) [43, 44] e o *Gossip* [48]. Avaliou-se o OF porque ele é um protocolo desenvolvido especialmente para a difusão de dados em RSSFs com ciclo de trabalho reduzido. Como o OF usa uma política MAC pró-ativa para compartilhar o escalonamento de trabalho dos nós vizinhos, este trabalho seguiu a proposta de Guo et al. [43, 44] para minimizar o custo de compartilhamento na qual cada nó sensor envia seu escalonamento somente para uma fração de vizinhos. Entretanto, como essa fração depende da densidade de nós no campo de sensoriamento, para as simulações com 250, 500, 750 e 1000 nós, as frações de compartilhamento do OF são 100%, 50%, 40% e 30%, respectivamente. Escolheu-se empiricamente essas frações porque elas são os menores valores que não reduzem a taxa de entrega do OF. A fração de compartilhamento desse protocolo reduz a taxa de entrega se os nós sensores descartarem pacotes porque não conhecem o escalonamento dos seus vizinhos. O compartilhamento de informações do OF acontece no início de cada simulação. Esta seção avaliou o *Gossip* devido à sua eficácia, simplicidade e robustez para difusão de dados. Contudo, como esse protocolo apresenta uma taxa de entrega reduzida quando os nós sensores operam com ciclo de trabalho reduzido, inseriu-se os agendamentos do GEM no *Gossip*. Quando um vizinho acorda no momento agendado, ele propaga o pacote com probabilidade  $p$  e, com probabilidade  $1 - p$ , ele aguarda um *time step* e tenta novamente. O nó repete esse processo até que ele ou outro vizinho propague o pacote. A probabilidade de decisão do *Gossip* é calculada como no GEM (equação 6.1). Em ambos os protocolos, o limiar de distância é de 70% do raio de comunicação, o valor máximo da probabilidade de decisão é 0,001 e a energia máxima é calculada a partir do comportamento do Mica2 [1]. A adaptação do comportamento dos nós localizados dentro das regiões de baixa energia pelo GEM consistiu em reduzir o ciclo de trabalho desses nós pela metade e proibir que esses nós sejam nós internos, eliminando descobertas e agendamentos dentro das regiões de baixa energia. Além disso, esta seção considerou o custo integral da construção do mapa de energia no GEM, contudo, outras aplicações também podem se beneficiar dessa informação. O GEM cria a árvore de difusão no início de cada simulação.

### 6.3.2 Difusão de Dados sem Regiões de Baixa Energia

Esta seção avalia o GEM em um cenário de difusão de dados em que os nós sensores operam com um ciclo de trabalho reduzido e a rede não contém regiões de baixa energia. A figura 6.13 mostra o desempenho do GEM quando se aumenta o número de nós sensores no campo sensorial. As métricas avaliadas são a taxa média de entrega, número total de pacotes transmitidos, consumo de energia e latência. A taxa média de entrega significa a porcentagem de pacotes entregues aos seus destinos e ela é fundamental porque os pacotes podem ser perdidos durante a difusão. Essa porcentagem está relacionada ao número total de mensagens enviadas até o final da simulação. O número total de pacotes transmitidos e o consumo de energia indicam o custo dos protocolos. A latência é um parâmetro crucial porque o atraso pode causar congestionamentos e perdas, principalmente quando os nós operam com um ciclo de trabalho reduzido no qual os emissores esperam mais tempo até que o próximo nó acorde.

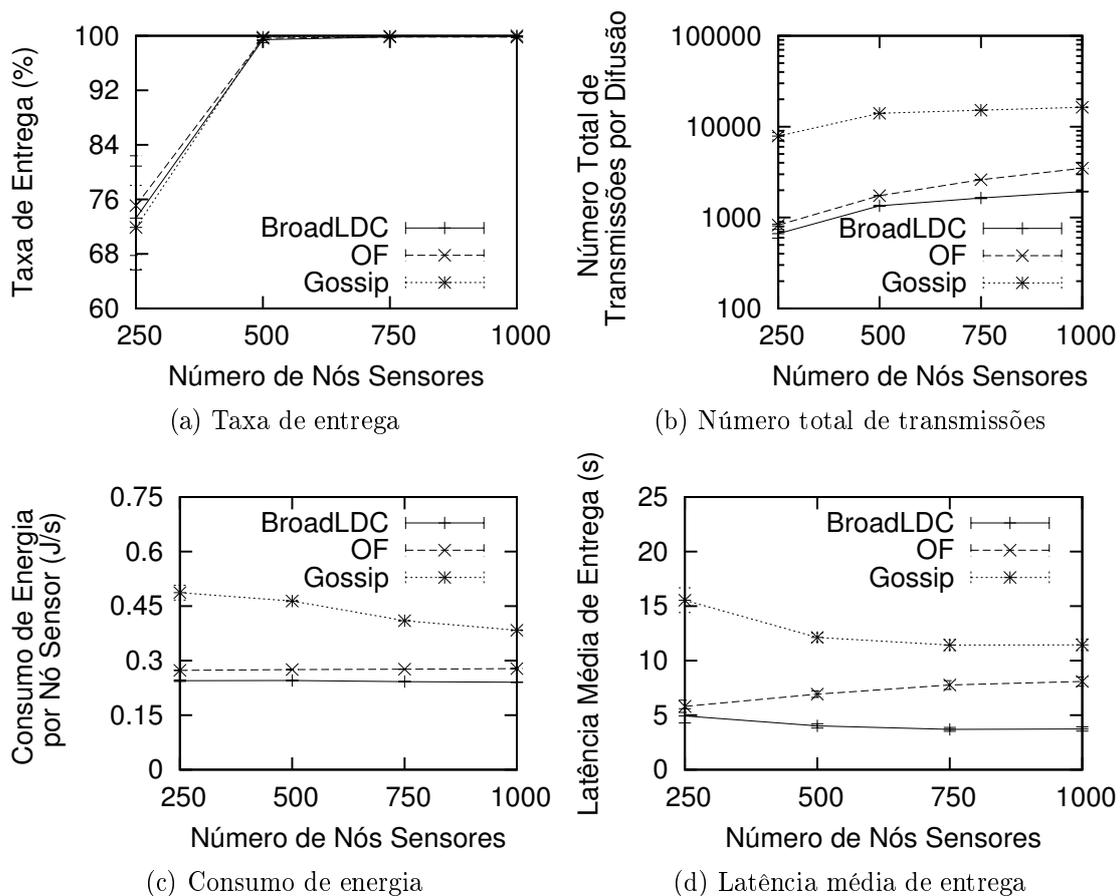


Figura 6.13: Métricas avaliadas no cenário sem regiões de baixa energia

A figura 6.13-a mostra a taxa de entrega do GEM, OF e *Gossip* à medida que se aumenta o número de nós no campo sensorial. Quando a rede tem 250 nós, os protocolos avaliados apresentam a menor taxa de entrega porque a rede é menos densa fazendo com que os três protocolos nem sempre encontrem uma rota para todos os destinos. A partir de 500 nós, os três protocolos entregam os pacotes para quase todos os nós da rede. Aplicando o teste-*T* com 0,05 de significância, não existe diferença entre os resultados obtidos pelos protocolos avaliados à medida que o número de nós sensores aumenta.

A figura 6.13-b mostra o número total de transmissões com escala logarítmica no eixo y. Na figura, o aumento do número de nós sensores no campo de sensoriamento faz com que os três protocolos avaliados façam mais transmissões. Isso acontece porque cada difusão tem mais destinos. O *Gossip* faz em média 6,17 e 9,58 vezes mais transmissões que o OF e o GEM, respectivamente. Isso porque no *Gossip*, em cada difusão, todos os nós emissores fazem agendamentos. Quando a rede tem 250 nós, o OF faz 1,24 vezes mais transmissões que o GEM e quando ela tem 1000 nós, essa diferença aumenta para 1,81 vezes. O aumento dessa diferença acontece porque no OF, todos os nós sensores compartilham informações sobre escalonamento e o aumento do número de nós na rede representa mais compartilhamentos. Por outro lado, no GEM, o número de nós internos na árvore de difusão quando a rede tem 500, 750 ou 1000 nós é praticamente o mesmo fazendo com que o custo com descobertas e agendamentos seja similar para os três casos. No GEM, o aumento do número de nós sensores resulta no aumento do número de folhas.

A figura 6.13-c mostra o consumo de energia dos nós sensores. O *Gossip* faz com que eles consumam em média 1,54 e 1,74 vezes mais energia que o OF e GEM, respectivamente. O consumo de energia do OF é 1,13 vezes maior que o do GEM. Esses resultados são consequência do número de transmissões obtido pelos protocolos avaliados. Além disso, o consumo do *Gossip* quando a rede tem 250 nós é 1,32 vezes maior que quando ela tem 1000 nós. Isso porque o aumento do número de nós aumenta a porcentagem de folhas cujo consumo de energia é menor que o dos nós internos. Pelo mesmo motivo, o consumo do GEM quando a rede tem 250 nós é 1,03 vezes maior que quando ela tem 1000 nós.

A figura 6.13-d mostra a latência média de entrega. A latência do *Gossip* é em média 1,77 vezes maior que a do OF e essa é 1,74 vezes maior que a do GEM. Isso porque todas as propagações do *Gossip* são baseadas em agendamentos, fazendo com que cada nó emissor aguarde um tempo para decidir se propaga um pacote de difusão recebido. O resultado do OF é porque cada nó sensor conhece apenas o escalonamento de uma fração de seus vizinhos, aumentando o caminho médio entre a origem e os nós

sensores. A latência do *Gossip* na rede com 250 nós é 1,36 vezes maior que na com 1000 nós. No GEM, essa relação é de 1,32 vezes. Esses resultados acontecem porque o aumento o número de nós faz com que os nós emissores fiquem normalmente mais distantes entre si, aumentando a velocidade da difusão. O OF apresenta um resultado oposto em que a latência da rede com 1000 nós é 1,39 vezes maior que a com 250 nós porque o aumento do número de nós implica na redução da fração de compartilhamento do OF.

Resultados de simulação revelam que no cenário avaliado, o GEM faz menos transmissões, consome menos energia e tem uma latência menor que a dos demais protocolos avaliados. Esse resultado mostra que soluções para difusão de dados em RSSFs com ciclo de trabalho reduzido não precisam que todos os nós sensores compartilhem informações sobre escalonamento.

### 6.3.3 Difusão de Dados com Regiões de Baixa Energia

Esta seção avalia o GEM em um cenário de difusão de dados contendo regiões de baixa energia. As figuras 6.14 e 6.15 apresentam o desempenho dos protocolos avaliados para os nós localizados dentro das regiões de baixa energia à medida que se aumenta o número de nós sensores no campo de sensoriamento. A diferença entre as duas figuras é que a primeira considera trinta e duas regiões de baixa energia e a segunda, somente uma região. Esta seção não apresenta os resultados de simulação para os nós localizados fora dessas regiões porque eles são similares aos do cenário sem regiões de baixa energia.

As figuras 6.14-a e 6.15-a apresentam a taxa de entrega para os nós localizados dentro das regiões de baixa energia. Como no cenário anterior, todos os protocolos têm a menor taxa de entrega quando a rede tem 250 nós. Na figura 6.14-a, as taxas de entregas do OF e a do *Gossip* são em média 1,04 vezes maiores que a do GEM quando a rede tem 250 nós. A partir de 500 nós, os três protocolos apresentam praticamente as mesmas taxas de entrega, o que pode ser confirmado aplicando o *teste-T* com 0,05 de significância. Na figura 6.15-a, as taxas de entrega do OF e *Gossip* são em média 1,93 vezes maiores que a do GEM. Esses resultados são consequência do GEM proibir que os nós localizados dentro das regiões de baixa energia sejam nós internos. A taxa de entrega média do GEM quando a rede tem trinta e duas regiões é 1,85 vezes maior que quando ela tem uma região porque no primeiro cenário, as regiões têm raios menores permitindo que mais nós localizados dentro das regiões de baixa energia sejam folhas na árvore de difusão.

As figuras 6.14-b e 6.15-b mostram o número total de transmissões para os nós

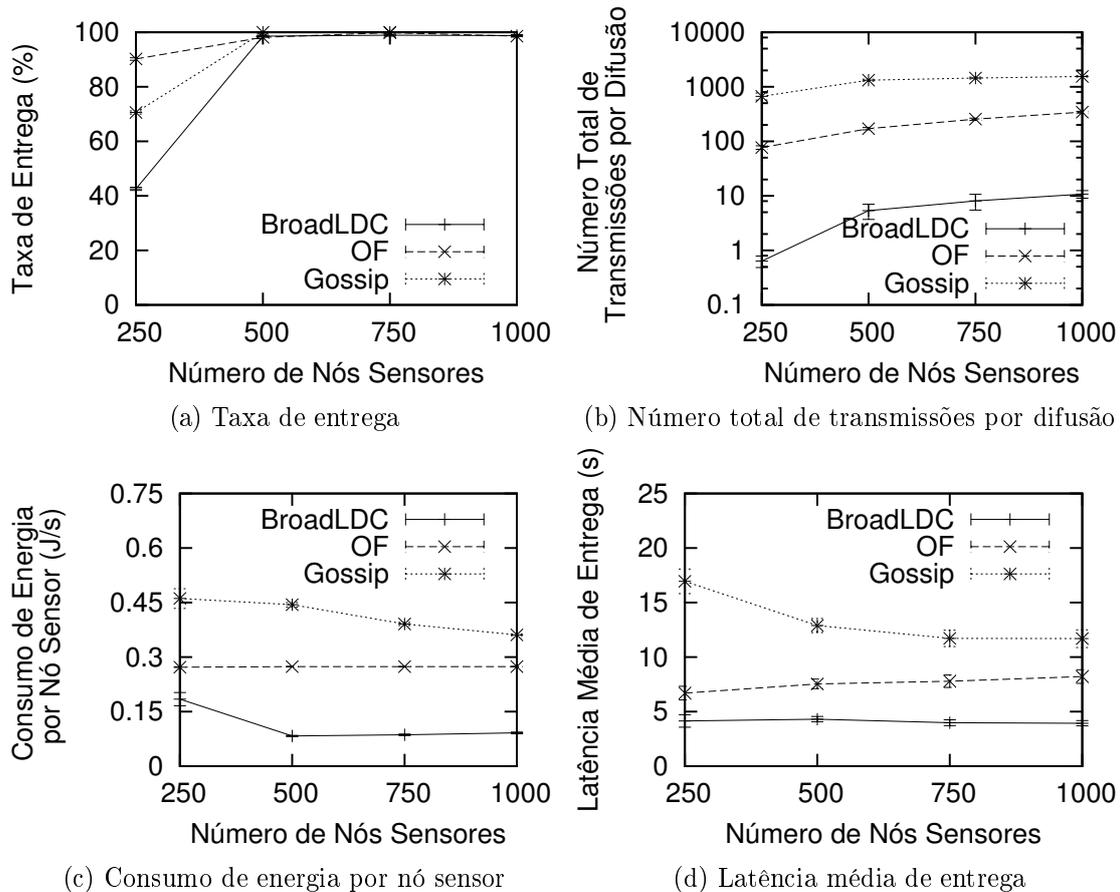


Figura 6.14: Métricas avaliadas para os nós sensores localizados dentro de trinta e duas regiões de baixa energia

localizados dentro das regiões de baixa energia. As figuras têm escala logarítmica no eixo y e o aumento do número de nós sensores no campo de sensoriamento faz com que os três protocolos avaliados façam mais transmissões. Isso acontece porque o aumento do número de nós sensores no campo de sensoriamento aumenta a densidade da rede, fazendo com que cada região de baixa energia tenha mais nós sensores (destinos das difusões). O GEM faz menos transmissões que os demais protocolos avaliados porque ele proíbe que os nós localizados dentro das regiões de baixa energia sejam nós internos. As transmissões realizadas por esse protocolo são de pacotes de filiação. Quando a rede tem trinta e duas regiões de baixa energia, o OF e *Gossip* fazem em média 34 e 201 vezes mais transmissões que o GEM, respectivamente. Quando ela tem uma região, eles fazem em média 90 e 555 vezes mais transmissões que o GEM, respectivamente. Na figura 6.14-b, o *Gossip* faz em média 6.20 vezes mais transmissões que o OF e, na figura 6.15-b, essa relação é de 5.89 vezes. Esse resultado é consequência dos agendamentos do *Gossip*. Comparando as figuras 6.14-b e 6.15-b, o GEM faz 3,06

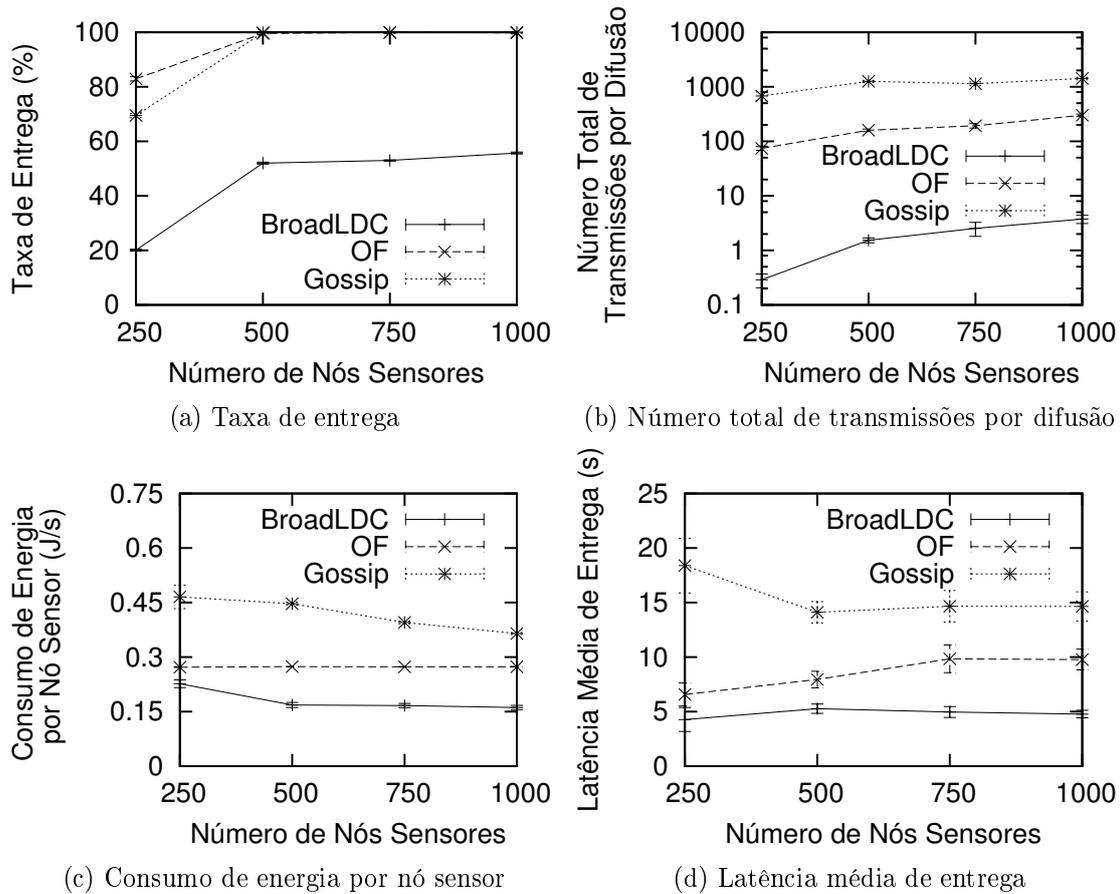


Figura 6.15: Métricas avaliadas para os nós sensores localizados dentro de uma região de baixa energia

vezes mais transmissões no cenário com trinta e duas regiões que no com uma região. Esse resultado é porque o raio das trinta e duas regiões é menor que o da região única fazendo com que no primeiro caso, os nós localizados dentro das regiões tenham mais vizinhos externos às mesmas. Como esses vizinhos podem ser nós internos, aumenta-se o número de nós folhas e de transmissões de pacotes de filiação dentro das regiões com menos energia.

As figuras 6.14-c e 6.15-c apresentam o consumo de energia dos nós sensores para os três protocolos avaliados. O consumo de energia do OF e do *Gossip* é em média 1,99 e 3,02 vezes maior que o do GEM, respectivamente. Isso acontece porque o GEM reduz o ciclo de trabalho dos nós localizados dentro das regiões com menos energia e efetua menos transmissões que os demais protocolos. Comparando as figuras 6.14-c e 6.15-c, o GEM consome 1,62 vezes mais energia no cenário com uma região que no com trinta e duas. Esse resultado é porque com trinta e duas regiões, o GEM faz com que mais nós localizados dentro dessas regiões participem da árvore (como folhas), reduzindo

seus respectivos ciclos de trabalho.

As figuras 6.14-d e 6.15-d mostram a latência média de entrega para os nós localizados dentro das regiões de baixa energia. Como no cenário anterior, o *Gossip* apresenta a maior latência seguido do OF. As latências do *Gossip* e do OF são em média 3,23 e 1,81 vezes maiores que a do GEM. Esses resultados são devido aos agendamentos do *Gossip* e à fração de compartilhamento do OF.

Resultados de simulação revelam que para o cenário avaliado, o GEM reduz o número de transmissões, consumo de energia e latência dentro das regiões de baixa energia em relação aos demais protocolos avaliados. Ele também consegue manter a taxa de entrega quando essas regiões têm um raio menor.

## 6.4 Conclusões

Este capítulo propõe e avalia o protocolo GEM para a difusão de dados em RSSFs com ciclo de trabalho reduzido. A difusão de dados é uma tarefa desafiadora nesse tipo de cenário porque quando um nó sensor envia um pacote, nem todos seus vizinhos estão acordados. A ideia principal do GEM é criar uma árvore de difusão baseada no mapa de energia da rede e fazer com que cada nó sensor compartilhe informações com apenas um nó vizinho que participa da árvore disseminando pacotes. O GEM também altera o funcionamento básico dos nós localizados dentro das regiões de baixa energia para prolongar o tempo de vida da rede. Resultados de simulação mostram que o GEM reduz o número de transmissões, consumo de energia e latência quando comparado com outras soluções propostas na literatura e avaliadas neste trabalho. Além disso, ele tem a habilidade de evitar regiões de baixa energia, reduzindo o fluxo de dados e o consumo de energia dentro das mesmas.

# Capítulo 7

## Conclusões e Trabalhos Futuros

*O fim é a oportunidade do recomeço.  
Autor desconhecido*

Este capítulo apresenta as conclusões e as direções futuras desta tese. As seções 7.1 e 7.2 abordam respectivamente esses tópicos.

### 7.1 Conclusões

A comunicação de dados em redes de sensores sem fio falha quando um nó envia um pacote para um vizinho adormecido. Protocolos MAC para essas redes tipicamente tratam o problema de escalonar a comunicação de dados quando o próximo nó estiver dormindo. Uma das abordagens usadas por eles é a política pró-ativa, na qual cada nó sensor compartilha seu escalonamento de trabalho com seus vizinhos e cada nó emissor agenda sua transmissão para o momento em que o próximo nó estiver acordado. Esse compartilhamento acontece através da troca de mensagens, aumentando o consumo de energia da rede. Outra abordagem utilizada pelos protocolos MAC para RSSFs é a política reativa, na qual cada nó emissor descobre o escalonamento de trabalho do próximo nó e, em seguida, envia sua mensagem. Apesar da política reativa eliminar o custo de compartilhamento, sua descoberta também faz transmissões adicionais. Dadas as limitações das duas políticas, esta tese apresenta três soluções para minimizar os custos de compartilhamento e descoberta.

A primeira solução é uma política híbrida para determinar o escalonamento de trabalho dos nós vizinhos, combinando as políticas MAC pró-ativa e reativa. A ideia da política híbrida é que cada nó emissor insira seu escalonamento de trabalho em seu quadro e cada receptor use essa informação para criar e atualizar uma tabela contendo o escalonamento de trabalho de seus vizinhos. Diferente da política pró-

ativa, os nós sensores obtêm o escalonamento de trabalho de seus vizinhos sem qualquer troca extra de mensagens. Quando um nó precisa enviar um quadro para um vizinho cujo escalonamento de trabalho é conhecido, como na política pró-ativa, o nó corrente agenda sua transmissão para o momento em que o próximo nó estiver acordado. Se o nó corrente desconhece essa informação, como na política reativa, ele inicia a descoberta da mesma. Quando ele a descobre, ele atualiza sua tabela e efetua sua transmissão. Ao contrário da política reativa, as transmissões futuras usam tal informação para evitar o custo reativo de novas descobertas. Todavia, uma desvantagem da solução híbrida é que cada nó emissor insere um campo extra no cabeçalho dos quadros. Resultados de simulação revelam o efeito da política híbrida em protocolos MAC pró-ativo e reativo. A solução proposta reduz o número de transmissões e o consumo de energia das políticas existentes na maioria dos cenários avaliados. Em particular, esses resultados são mais expressivos nos cenários em que a informação sobre os vizinhos desatualiza durante o tempo de vida da rede ou os nós sensores operam com um ciclo de trabalho reduzido (1% ou menos) ou dinâmico.

A segunda solução é um projeto combinado entre a política MAC reativa e o roteamento baseado no receptor. O roteamento baseado no receptor faz com que quando um nó recebe um pacote, esse nó decida localmente se deve ou não propagar o pacote. Ao contrário do roteamento baseado no emissor no qual o nó anterior escolhe quem será o próximo nó e insere essa informação no cabeçalho do pacote. O roteamento baseado no receptor aumenta o número de candidatos a próximo nó quando comparado com a abordagem baseada no emissor. O projeto combinado proposto nesta tese usa a diversidade de candidatos a próximo nó do roteamento baseado no receptor para reduzir o custo da descoberta reativa. Um nó emissor faz menos transmissões de descoberta para encontrar algum vizinho acordado que para um vizinho específico. Os protocolos MAC reativos propostos na literatura são combinados com o roteamento baseado no emissor. Resultados teóricos e de simulação revelam que o projeto combinado envolvendo a política reativa e o roteamento baseado no receptor reduz o número de transmissões e a latência na comunicação de dados quando comparado com outras soluções avaliadas neste trabalho. Esta tese também apresentou o protocolo *Receiver-based MAC* (RbMAC), um protocolo reativo para o projeto proposto. Além disso, fez-se a verificação formal de algumas propriedades do RbMAC e avaliou-se uma técnica de ciclo de trabalho dinâmica para ele.

A terceira solução é o protocolo *Gossiping Using the Energy Map* (GEM) para a difusão de dados em redes de sensores sem fio. A difusão de dados é uma tarefa de comunicação fundamental nessas redes. Ela acontece quando o nó monitor deseja enviar uma informação para todos os nós sensores. Essa tarefa é desafiadora em redes

com ciclo de trabalho reduzido porque quando um nó envia um pacote, nem todos seus vizinhos estão acordados para recebê-lo. A maioria das soluções propostas na literatura para difusão de dados em RSSFs são protocolos de roteamento que ignoram o fato dos nós ficarem quase sempre adormecidos. Outras soluções consideram um projeto integrado entre as camadas de rede e MAC em que cada nó emissor efetua sua difusão através de várias transmissões *unicasting*. Uma para cada vizinho quando esse estiver acordado. Essas soluções são baseadas em uma política MAC pró-ativa em que cada nó sensor compartilha seu escalonamento de trabalho com todos seus vizinhos, permitindo que cada nó emissor saiba quando cada um de seus vizinhos estará acordado. O GEM cria uma árvore de difusão baseada na energia residual da rede e faz com que cada nó sensor compartilhe informações apenas com um nó vizinho que pertence à árvore disseminando pacotes. Resultados de simulação mostram que o GEM reduz o número de transmissões, consumo de energia e latência quando comparado com outras soluções propostas na literatura e avaliadas nesta tese. O GEM aborda outro desafio que é a difusão de dados quando a rede contém regiões de baixa energia.

## 7.2 Trabalhos Futuros

Esta seção descreve brevemente as direções futuras desta tese.

- **Overhead da política híbrida** – A política híbrida proposta no capítulo 4 faz com que todos os nós emissores insiram um *overhead* em seus pacotes. Trabalhos futuros podem usar técnicas de codificação ou de compactação ou avaliar o efeito de inserir essa informação somente em uma porcentagem dos pacotes. Além disso, trabalhos futuros podem avaliar o efeito do aumento/redução do número de bits para representar o escalonamento de trabalho na política híbrida.
- **Protocolo híbrido entre as abordagens de roteamento baseadas no emissor e no receptor** – O capítulo 4 apresentou uma política híbrida envolvendo as políticas MAC pró-ativa e reativa para obter o escalonamento de trabalho de nós vizinhos e trabalhos futuros podem considerar uma abordagem híbrida entre as técnicas de roteamento baseadas no emissor e no receptor. Os protocolos de roteamento baseados no emissor normalmente efetuam menos transmissões que os baseados no receptor quando a rede opera com ciclo de trabalho reduzido ou a rede apresenta menos nós sensores dentro do campo de sensoria-mento. Caso contrário, a abordagem baseada no receptor normalmente efetua menos transmissões. Outra métrica para determinar a escolha do próximo nó é a ocorrência de falhas.

- **Ciclo de trabalho dinâmico** – Os capítulos 4, 5 e 6 avaliam suas respectivas conclusões em cenários em que os nós sensores operam com ciclo de trabalho dinâmico. As soluções propostas fazem um chaveamento entre dois valores fixos (capítulos 5 e 6) ou consideram um valor pré-definido pelo nó monitor (capítulo 4). Trabalhos futuros podem considerar outros cenários com ciclo dinâmico em que os nós atualizam esse parâmetro de acordo com informações locais como, por exemplo, o que o GEM fez com os nós localizados dentro de regiões de baixa energia. Trabalhos futuros podem modificar o ciclo de trabalho dos nós sensores a partir do tráfego percebido por eles.
- **Projeto integrado envolvendo a comunicação de dados baseada no receptor** – O projeto integrado entre as camadas MAC e de rede pode otimizar o consumo de energia da rede, uma vez que uma camada tem informações que podem ser úteis para a outra. Por exemplo, a camada MAC pode usar informações sobre a direção das rotas (e.g., as equações de curva utilizada nos protocolos de roteamento em curva) para definir ou atualizar o escalonamento de trabalho dos nós sensores. O projeto integrado entre essas camadas também pode reduzir o consumo de energia da rede se reduzir o número de colisões. Por exemplo, os protocolos da família *gossip* (e. g., *Gossip 1* e GEM) podem modificar suas respectivas probabilidades de transmissão a partir do número de pacotes recebidos com erro. Outras funções de rede (e.g., fusão de dados) podem ser integradas com soluções para a coleta de dados.
- **Projeto integrado envolvendo outras camadas da pilha de protocolos** – As camadas superiores normalmente possuem informações adicionais que podem otimizar o funcionamento de camadas inferiores como as camadas MAC e de rede. Por exemplo, quando a camada de aplicação precisa de dados somente durante o dia, a camada MAC pode fazer com que os nós sensores adormeçam no início da noite e acordem no início do dia.
- **Tolerância a falhas na árvore do GEM** – Um ponto importante no projeto de uma árvore de difusão é a ocorrência de falhas. Trabalhos futuros podem abordar técnicas de reconstrução local da árvore do GEM.
- **GEM sem mapa de energia** – A difusão de dados em redes de sensores sem fio com ciclo de trabalho reduzido é um tópico importante no projeto desse tipo de rede. Contudo, nem sempre, o nó monitor terá o mapa de energia da rede. Trabalhos futuros podem construir a árvore de difusão do GEM considerando outras métricas como, por exemplo, a energia local dos nós sensores. Outro exemplo

acontece quando a aplicação estiver temporariamente interessada em dados de sensoriamento de apenas uma parte da rede. Nesse caso, o nó monitor pode fazer com que os nós sensores localizados nessa região tenham um comportamento distinto das demais regiões.

- **Coleta de dados com a árvore do GEM** – Trabalho futuros podem usar a árvore do GEM para a coleta de dados. Nesse caso, quando um nó sensor tiver algum dado para o nó monitor, esse nó sensor propaga para seu pai, que repete o processo até o nó monitor. Esses nós também podem explorar alguma técnica de fusão de dados. Além disso, a árvore também pode ser explorada por protocolos de clusterização em que os líderes dos *clusters* são os nós da árvore.

## Referências Bibliográficas

- [1] *Mts/mda sensor and data acquisition boards user's manual, document 7430-0020-03*. [www.xbow.com](http://www.xbow.com), Dez. 2013.
- [2] *Network simulator*. [www.isi.edu/nsnam/ns](http://www.isi.edu/nsnam/ns), Dez. 2013.
- [3] *The nusmv system*. [nusmv.irst.itc.it](http://nusmvIRST.ITC.IT), Dez. 2013.
- [4] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, *Wireless sensor networks: A survey*, *Computer Networks*, 38 (2002), pp. 393–422.
- [5] I. Akyildiz, M. Vuran, and O. Akan, *A cross-layer protocol for wireless sensor networks*, in *Proceedings of the IEEE Conference on Information Sciences and Systems (CISS)*, 2006, pp. 1102–1107.
- [6] M. Alaei and J. Barcelo-Ordinas, *A collaborative node management scheme for energy-efficient monitoring in wireless multimedia sensor networks*, *Springer Wireless Networks*, 19 (2013), pp. 639–659.
- [7] M. Ali, U. Saif, A. Dunkels, T. Voigt, K. Römer, K. Langendoen, J. Polastre, and Z. Uzmi, *Medium access control issues in sensor networks*, *ACM SIGCOMM - Computer Communication Review*, 36 (2006), pp. 33–36.
- [8] U. N. M. Arshad, N. Shahid, and S. Raza, *Classification of localization algorithms for wireless sensor network: A survey*, in *Open Source Systems and Technologies (ICOSST)*, 2012 International Conference on, 2012, pp. 1–5.
- [9] G. Asada, T. Dong, F. Lin, G. Pottie, W. Kaiser, and H. Marcy, *Wireless integrated network sensors: Low power systems on a chip*, in *Proceedings of the European Solid State Circuits Conference (ESSCIRC)*, 1998, pp. 9–16.
- [10] A. Bachir, M. Dohler, T. Watteyne, and K. Leung, *Mac essentials for wireless sensor networks*, *IEEE Communications Surveys and Tutorials*, 12 (2010), pp. 222–248.

- [11] B. Baker and R. Shostak, *Gossips and telephones*, Discrete Mathematics, 2 (1972), pp. 191–193.
- [12] H. Barros, S. Campos, M. Song, and L. Zarate, *Exploring clause symmetry in a distributed bounded model checking algorithm*, in Proceedings of the IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS), 2007, pp. 531–538.
- [13] T. Beluch, D. Dragomirescu, and R. Plana, *A sub-nanosecond synchronized mac - phy cross-layer design for wireless sensor networks*, Journal Ad Hoc Networks, 11 (2013), pp. 833–845.
- [14] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, *Macaw: a media access protocol for wireless lan's*, in Proceedings of the ACM International Conference on Communications architectures, protocols and applications (SIGCOMM), 1994, pp. 212–225.
- [15] A. Birrell, R. Levin, M. Schroeder, and R. Needham, *Grapevine: an exercise in distributed computing*, Communications of the ACM, 25 (1982), pp. 260–274.
- [16] B. Blum, T. He, S. Son, and J. Stankovic, *Igf: A state-free robust communication protocol for wireless sensor networks*, tech. rep., Technical Report CS-2003-11, Univ. of Virginia, 2003.
- [17] A. Boonsongsrikul, S. Kocijancic, and S. Suppharangsarn, *Effective energy consumption on wireless sensor networks: Survey and challenges*, in Proceedings of the IEEE International Convention on Information Communication Technology Electronics Microelectronics (MIPRO), 2013, pp. 469–473.
- [18] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, *Randomized gossip algorithms*, IEEE/ACM Transactions on Networking, 14 (2006), pp. 2508–2530.
- [19] M. Buettner, G. Yee, E. Anderson, and R. Han, *X-mac: a short preamble mac protocol for duty-cycled wireless sensor networks*, in Proceedings of the IEEE International Conference on Embedded networked sensor systems (SENSYS), 2006, pp. 307–320.
- [20] M. Calle and J. Kabara, *Measuring energy consumption in wireless sensor networks using gsp*, in Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), 2006, pp. 1–5.

- [21] Y. Cao, S. Guo, and T. He, *Robust multi-pipeline scheduling in low-duty-cycle wireless sensor networks*, in Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), 2012, pp. 361–369.
- [22] Z. Cao, Y. He, and Y. Liu,  *$l^2$ : Lazy forwarding in low duty cycle wireless sensor networks*, in Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), 2012, pp. 1323–1331.
- [23] R. Carrano, D. Passos, L. Magalhaes, and C. Albuquerque, *Survey and taxonomy of duty cycling mechanisms in wireless sensor networks*, IEEE Communications Surveys Tutorials, PP (2013), pp. 1–14.
- [24] Y. Chang, J. Jiang, J. Sheu, and H. Shih, *Adca: An asynchronous duty cycle adjustment mac protocol for wireless sensor networks*, in Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), 2008, pp. 1–5.
- [25] H. Cheng, N. Xiong, X. Huang, and L. Yang, *An efficient scheduling model for broadcasting in wireless sensor networks*, in Proceedings of the IEEE International Parallel and Distributed Processing Symposium Workshops PhD Forum (IPDPSW), 2013, pp. 1417–1428.
- [26] E. Clarke, O. Grumberg, and D. Long, *Model checking and abstraction*, ACM Transactions on Programming Languages and Systems, 16 (1994), pp. 1512–1542.
- [27] M. Conti, G. Maselli, G. Turi, and S. Giordano, *Cross-layering in mobile ad hoc network design*, IEEE Computer, 37 (4), pp. 48–51.
- [28] D. Costa, L. Guedes, F. Vasques, and P. Portugal, *Effect of frame size on energy consumption in wireless image sensor networks*, in Proceedings of the IEEE International Conference on Imaging Systems and Techniques (IST), 2012, pp. 239–244.
- [29] F. Cunha, R. Mini, and a. Loureiro, *Sensor-mac with dynamic duty cycle in wireless sensor networks*, in Proceedings of the IEEE Global Communications Conference (GLOBECOM), 2012, pp. 530–536.
- [30] U. Datta and S. Kundu, *Packet size optimization for multi hop cdma wireless sensor networks with nearest neighbors based routing*, in Proceedings of the IEEE International Conference on Emerging Applications of Information Technology (EAIT), 2012, pp. 408–412.

- [31] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry, *Epidemic algorithms for replicated database maintenance*, in Proceedings of the ACM International Symposium on Principles of distributed computing (PODC), 1987, pp. 1–12.
- [32] A. Demers, D. Greene, C. Houser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry, *Epidemic algorithms for replicated database maintenance*, ACM SIGOPS Operating Systems Review, 22 (1988), pp. 8–32.
- [33] I. Demirkol, C. Ersoy, and F. Alagoz, *Mac protocols for wireless sensor networks: a survey*, IEEE Communications Magazine, 44 (2006), pp. 115–121.
- [34] A. Durresi, V. Paruchuri, S. Iyengar, and R. Kannan, *Optimized broadcast protocol for sensor networks*, IEEE Transactions on Computers, 54 (2005), pp. 1013–1024.
- [35] A. El-Hoiydi, *Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks*, in Proceedings of the IEEE International Conference on Communications (ICC), 2002, pp. 3418–3423.
- [36] A. El-Hoiydi and J. Decotignie, *Wisemac: an ultra low power mac protocol for the downlink of infrastructure wireless sensor networks*, in Proceedings of the IEEE Symposium on Computers and Communications (ISCC), 2004, pp. 244–251.
- [37] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, *Next century challenges: scalable coordination in sensor networks*, in Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM), 1999, pp. 263–270.
- [38] C. Figueiredo, E. Nakamura, A. Loureiro, and L. Ruiz, *An event-detection estimation model for hybrid adaptive routing in wsns*, in Proceedings of the IEEE International Conference on Communications (ICC), 2007, pp. 3887–3894.
- [39] S. Gajjar, S. Pradhan, and K. Dasgupta, *Performance analysis of cross layer protocols for wireless sensor networks*, in Proceedings of the ACM International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2012, pp. 348–354.
- [40] G. Ghidini and S. Das, *Energy-efficient markov chain-based duty cycling schemes for greener wireless sensor networks*, ACM Journal on Emerging Technologies in Computing Systems (JETC), 8 (2012), pp. 29:1–29:32.

- [41] O. Goussevskaia, M. Machado, R. Mini, A. Loureiro, G. Mateus, and J. Nogueira, *Data dissemination based on the energy map*, IEEE Communications Magazine, 43 (2005), pp. 134–143.
- [42] C. Guo, L. Zhong, and J. Rabaey, *Low power distributed MAC for ad hoc sensor radio networks*, in Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), 2001, pp. 2944–2948.
- [43] S. Guo, Y. Gu, B. Jiang, and T. He, *Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links*, in Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM), 2009, pp. 133–144.
- [44] S. Guo, L. He, Y. Gu, B. Jiang, and T. He, *Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links*, IEEE Transactions on Computers, PP (2013), pp. 1–14.
- [45] Z. Haas, *A new routing protocol for the reconfigurable wireless networks*, in Proceedings of the IEEE International Conference on Universal Personal Communications Record (ICUPC), 1997, pp. 562–566.
- [46] —, *On some challenges and design choices in ad-hoc communications*, in Proceedings of the IEEE Military Communications Conference (MILCOM), 1998, pp. 187–192.
- [47] Z. Haas, J. Halpern, and L. Li, *Gossip-based ad hoc routing*, in Proceedings of the IEEE Joint Conference of the IEEE Computer and Communications Societies, (INFOCOM), 2002, pp. 1707–1716.
- [48] —, *Gossip-based ad hoc routing*, IEEE/ACM Transactions on Networking, 14 (2006), pp. 479–491.
- [49] A. Hajnal, E. C. Milner, and E. Szemerédi, *A cure for the telephone disease*, Canadian Mathematical Bulletin, 15 (1972), pp. 447–450.
- [50] K. Han, Y. Liu, and J. Luo, *Duty-cycle-aware minimum-energy multicasting in wireless sensor networks*, IEEE/ACM Transactions on Networking, 21 (2013), pp. 910–923.
- [51] J. Hao, B. Zhang, and H. Mouftah, *Routing protocols for duty cycled wireless sensor networks: A survey*, IEEE Communications Magazine, 50 (2012), pp. 116–123.

- [52] S. He, J. Chen, and Y. Sun, *Coverage and connectivity in duty-cycled wireless sensor networks for event monitoring*, IEEE Transactions on Parallel and Distributed Systems, 23 (2012), pp. 475–482.
- [53] S. Hedetniemi, S. Hedetniemi, and A. Liestman, *A survey of gossiping and broadcasting in communication networks*, Networks, 18 (1988), pp. 319–349.
- [54] W. Heinzelman, J. Kulik, and H. Balakrishnan, *Adaptive protocols for information dissemination in wireless sensor networks*, in Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM), 1999, pp. 174–185.
- [55] M. Heissenbuttel, T. Braun, M. Walchli, and T. Bernoulli, *Optimized stateless broadcasting in wireless multi-hop networks*, in Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), 2006, pp. 1–12.
- [56] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, *System architecture directions for networked sensors*, ACM SIGARCH Computer Architecture News, 28 (2000), pp. 93–104.
- [57] X. Hou and D. Tipper, *Adaptive gossip-based energy conservation for wireless ad hoc routing*, in Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), 2004, pp. 2099–2102.
- [58] ———, *Gossip-based sleep protocol (gsp) for energy efficient routing in wireless ad hoc networks*, (2004), pp. 1305–1310.
- [59] X. Hou, D. Tipper, and S. Wu, *A gossip-based energy conservation protocol for wireless ad hoc and sensor networks*, Journal of Network and Systems Management, 14 (2006), pp. 381–414.
- [60] C. Hurkens, *Spreading gossip efficiently*, Nieuw Archief voor Wiskunde, 5 (2000), pp. 208–210.
- [61] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, *Impact of network density on data aggregation in wireless sensor networks*, in Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS), 2002, pp. 457–463.
- [62] C. Intanagonwiwat, R. Govindan, and D. Estrin, *Directed diffusion: a scalable and robust communication paradigm for sensor networks*, in Proceedings of the

- ACM International Conference on Mobile Computing and Networking (MOBICOM), 2000, pp. 56–67.
- [63] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, *Directed diffusion for wireless sensor networking*, IEEE/ACM Transactions on Networking, 11 (2003), pp. 2–16.
- [64] R. Jain, *The Art of Computer Systems Performance Analysis: techniques for experimental design, measurement, simulation and modeling*, 1991.
- [65] K. Jamieson, H. Balakrishnan, and Y. Tay, *Sift: a mac protocol for event-driven wireless sensor networks*, in Proceedings of the European Workshop on Wireless Sensor Networks (EWSN), 2006, pp. 260–275.
- [66] B. Jang, J. Lim, and M. Sichitiu, *As-mac: An asynchronous scheduled mac protocol for wireless sensor networks*, in Proceedings of the IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS), 2008, pp. 434–441.
- [67] ———, *An asynchronous scheduled mac protocol for wireless sensor networks*, Computer Networks, 57 (2013), pp. 85–98.
- [68] S. Jayashree, B. Manoj, and C. Murthy, *On using battery state for medium access control in ad hoc wireless networks*, in Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM), 2004, pp. 360–373.
- [69] S. Ji and Z. Cai, *Distributed data collection in large-scale asynchronous wireless sensor networks under the generalized physical interference model*, IEEE/ACM Transactions on Networking, 21 (2013), pp. 1270–1283.
- [70] Q. Jia, L. Shi, Y. Mo, and B. Sinopoli, *On optimal partial broadcasting of wireless sensor networks for kalman filtering*, IEEE Transactions on Automatic Control, 57 (2012), pp. 715–721.
- [71] J. M. Kahn, R. H. Katz, and K. S. J. Pister, *Next century challenges: Mobile networking for smart dust*, in Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM), 1999, pp. 271–278.
- [72] W. Kermack and A. McKendrick, *A contribution to the mathematical theory of epidemics*, Royal Society, 115 (1927), pp. 700–721.

- [73] C. Kern and M. Greenstreet, *Formal verification in hardware design: a survey*, ACM Transactions on Design Automation of Electronic Systems, 4 (1999), pp. 123–193.
- [74] A. Khelil, C. Becker, J. Tian, and K. Rothermel, *An epidemic model for information diffusion in manets*, in Proceedings of the ACM International Workshop on Modeling analysis and simulation of wireless and mobile systems (MSWiM), 2002, pp. 54–60.
- [75] J. Kim, J. On, S. Kim, and J. Lee, *Performance evaluation of synchronous and asynchronous mac protocols for wireless sensor networks*, in Proceedings of the IEEE International Conference on Sensor Technologies and Applications (SENSORCOMM), 2008, pp. 500–506.
- [76] J. Kulik, W. Heinzelman, and H. Balakrishnan, *Negotiation-based protocols for disseminating information in wireless sensor networks*, in Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBI-COM), 1999, pp. 169–185.
- [77] H. Landau and A. Rapoport, *Contribution to the mathematical theory of contagion and spread of information: I. spread through a thoroughly mixed population*, Journal Bulletin of Mathematical Biology, 15 (1953), pp. 173–183.
- [78] K. Langendoen and A. Meier, *Analyzing mac protocols for low data-rate applications*, ACM Transactions on Sensor Networks, 7 (2010), pp. 1–40.
- [79] J. Lee, D. Lee, J. Kim, W. Cho, and J. Pajak, *A dynamic sensing cycle decision scheme for energy efficiency and data reliability in wireless sensor networks*, in Proceedings of the International Conference on Advanced Intelligent Computing Theories and Applications, 2007, pp. 218–229.
- [80] K. Lee and D. Cho, *A multiple access collision avoidance protocol for multicast service in mobile ad hoc networks*, in Proceedings of the IEEE Vehicular Technology Conference (VTC), 2003, pp. 1793–1797.
- [81] X. Li, K. Moaveninejad, and O. Frieder, *Regional gossip routing for wireless ad hoc networks*, in Proceedings of the IEEE International Conference on Local Computer Networks (LCN), 2003, pp. 274–275.
- [82] —, *Regional gossip routing for wireless ad hoc networks*, Mobile Networks and Applications, 10 (2005), pp. 61–77.

- [83] W. Liao and H. Wang, *An asynchronous mac protocol for wireless sensor networks*, Journal of Network and Computer Applications, 31 (2008), pp. 807–820.
- [84] R. Lin, Z. Wang, and Y. Sun, *Energy efficient medium access control protocols for wireless sensor networks and its state-of-art*, in Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE), 2004, pp. 669–674.
- [85] M. Machado, *Disseminação de dados baseada em trajetória e energia para redes de sensores sem fio*, Master’s thesis, Federal University of Minas Gerais, 2005.
- [86] M. Machado, O. Goussevskaia, R. Mini, C. Rezende, A. Loureiro, G. Mateus, and J. Nogueira, *Data dissemination in autonomic wireless sensor networks*, IEEE Journal on Selected Areas in Communications, 23 (2005), pp. 2305–2319.
- [87] M. Machado, R. Mini, and A. Loureiro, *A combined approach for receiver-based mac and network layers in wireless sensor networks*, in Proceedings of the IEEE International Conference on Communications (ICC), 2009, pp. 102–107.
- [88] M. Machado, R. Mini, A. Loureiro, D. Guidoni, and P. Melo, *Gossiping using the energy map in wireless sensor networks*, in Proceedings of the ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), 2007, pp. 368–372.
- [89] S. Mahlhecht and M. Bock, *Csma-mps: A minimum preamble sampling mac protocol for low power wireless sensor networks*, in Proceedings of the IEEE International Workshop on Factory Communication Systems (WFCS), 2004, pp. 73–80.
- [90] T. Melodia, M. Vuran, and D. Pompili, *The state of the art in cross-layer design for wireless sensor networks*, in Proceedings of the Springer International Conference on Wireless Systems and Network Architectures in Next Generation Internet (EURONGI), 2005, pp. 78–92.
- [91] L. Mendes and J. Rodrigues, *Review: A survey on cross-layer solutions for wireless sensor networks*, Journal of Network and Computer Applications - Elsevier, 34 (2011), pp. 523–534.
- [92] M. Miller and N. Vaidya, *A mac protocol to reduce sensor network energy consumption using a wakeup radio*, IEEE Transactions on Mobile Computing, 4 (2005), pp. 228–242.

- [93] R. Mini, M. Machado, A. Loureiro, and B. Nath, *Prediction-based energy map for wireless sensor networks*, Ad Hoc Networks Journal, 3 (2005), pp. 235–253.
- [94] J. Moon, *Random exchanges of information*, Nieuw Arch., 20 (1972), pp. 246–249.
- [95] D. Mosk-Aoyama and D. Shah, *Computing separable functions via gossip*, in Proceedings of the ACM International Symposium on Principles of distributed computing (PODC), 2006, pp. 113–122.
- [96] E. Nakamura, H. Oliveira, L. Pontello, and L. A. Antonio, *On demand role assignment for event-detection in sensor networks*, in Proceedings of the IEEE Symposium on Computers and Communications (ISCC), 2006, pp. 941–947.
- [97] B. Nath and D. Niculescu, *Routing on a curve*, ACM SIGCOMM - Computer Communication Review, 33 (2003), pp. 155–160.
- [98] S. Ni, Y. Tseng, Y. Chen, and J. Sheu, *The broadcast storm problem in a mobile ad hoc network*, in Proceedings of the ACM/IEEE International Conference on Mobile computing and networking (MOBICOM), 1999, pp. 151–162.
- [99] D. Niculescu and B. Nath, *Trajectory-based forwarding and its applications*, in Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM), 2003, pp. 260–272.
- [100] B. Pittel, *On spreading a rumor*, Journal on Applied Mathematics (SIAP), 47 (1987), pp. 213–223.
- [101] G. Pottie and W. Kaiser, *Embedding the internet wireless integrated network sensors*, Communications of the ACM, 43 (2000), pp. 51–58.
- [102] A. Rapoport and L. Rebhun<sup>1</sup>, *On the mathematical theory of rumor spread*, Journal Bulletin of Mathematical Biology, 14 (1952).
- [103] S. Rashwand, J. Misic, V. Misic, S. Biswas, and M. Haque, *A novel asynchronous, energy efficient, low transmission delay mac protocol for wireless sensor networks*, in Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS), 2009, pp. 186–193.
- [104] R. Renesse, K. Birman, and W. Vogels, *Astrolabe: A robust and scalable technology for distributed system monitoring, management, and data mining*, ACM Transactions on Computer Systems, 21 (2003), pp. 164–206.

- [105] R. Renesse, Y. Minsky, and M. Hayden, *A gossip-style failure detection service*, tech. rep., 1998. Cornell University.
- [106] M. Salajegheh, H. Soroush, and A. Kalis, *Hymac: Hybrid tdma/fdma medium access control protocol for wireless sensor networks*, in Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), 2007, pp. 1–5.
- [107] Y. Sankarasubramaniam, I. Akyildiz, and S. McLaughlin, *Energy efficiency based packet size optimization in wireless sensor networks*, in Proceedings of the IEEE International Workshop on Sensor Network Protocols and Applications (SNPA), 2003, pp. 1–8.
- [108] A. Sayadi, S. Mahfoudh, and A. Laouiti, *Sensor-ostr: Novel energy-efficient dynamic tdma frame size-based mac protocol for wireless multi-hop sensor networks*, in Proceedings of the IFIP/IEEE Wireless Days (WD), 2012, pp. 1–3.
- [109] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, *Optimizing sensor networks in the energy-latency-density design space*, IEEE Transactions on Mobile Computing, 1 (2002), pp. 70–80.
- [110] L. Shi and A. Fapojuwo, *Cross-layer optimization with cooperative communication for minimum power cost in packet error rate constrained wireless sensor networks*, Journal Ad Hoc Networks, 10 (2012), pp. 1457–1468.
- [111] S. Singh and C. Raghavendra, *Pamas—power aware multi-access protocol with signalling for ad hoc networks*, SIGCOMM Computer Communication Review, 28 (1998), pp. 5–26.
- [112] P. Skraba, H. Aghajan, and A. Bahai, *Distributed passive routing decisions in mobile ad-hoc networks*, in Proceedings of the IEEE Vehicular Technology Conference (VTC), 2004, pp. 2814–2818.
- [113] K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie, *Protocols for self-organization of a wireless sensor network*, IEEE Personal Communications, 7 (2000), pp. 16–27.
- [114] C. Song, H. Sharif, and K. Nuli, *Study of an adaptive frame size predictor to enhance energy conservation in wireless sensor networks*, IEEE Journal on Selected Areas in Communications, 23 (2005), pp. 283–292.

- [115] F. Stann and J. Heidemann, *Rmst: Reliable data transport in sensor networks*, in Proceedings of the IEEE International Workshop on Sensor Net Protocols and Applications (SNPA), 2003, pp. 102–112.
- [116] W. Su and T. L. Lim, *Cross-layer design and optimization for wireless sensor networks*, in Proceedings of the ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD), 2006, pp. 278–284.
- [117] Y. Sun, O. Gurewitz, and D. Johnson, *Ri-mac: A receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks*, in Proceedings of the ACM Conference on Embedded Networked Sensor Systems (SENSYS), 2008, pp. 1–14.
- [118] A. Tanenbaum, *Computer Networks*, Prentice Hall Professional Technical Reference, 2011.
- [119] R. Tijdeman, *On a telephone problem*, Nieuw Archief voor Wiskunde, 3 (1971).
- [120] A. Vahdat and D. Becker, *Epidemic routing for partially connected ad hoc networks*, tech. rep., 2000. Technical Report CS-200006, Duke University.
- [121] L. Villas, A. Boukerche, R. Araujo, and A. Loureiro, *A reliable and data aggregation aware routing protocol for wireless sensor networks*, in Proceedings of the ACM International Conference on Modeling, analysis and simulation of wireless and mobile systems (MSWiM), 2009, pp. 245–252.
- [122] C. Wan, A. Campbell, and L. Krishnamurthy, *Psfq: a reliable transport protocol for wireless sensor networks*, in Proceedings of the ACM International Workshop on Wireless sensor networks and applications (WSNA), 2002, pp. 1–11.
- [123] F. Wang and J. Liu, *Duty-cycle-aware broadcast in wireless sensor networks*, in Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), 2009, pp. 468–476.
- [124] —, *On reliable broadcast in low duty-cycle wireless sensor networks*, IEEE Transactions on Mobile Computing, 11 (2012), pp. 767–779.
- [125] L. Wang and Y. Xiao, *A survey of energy-efficient scheduling mechanisms in sensor networks*, Mobile Networks and Applications, 11 (2006), pp. 723–740.

- [126] Y. Wang, M. Vuran, and S. Goddard, *Cross-layer analysis of the end-to-end delay distribution in wireless sensor networks*, IEEE/ACM Transactions on Networking, 20 (2012), pp. 305–318.
- [127] S. Xiong, J. Li, Z. Li, J. Wang, and Y. Liu, *Multiple task scheduling for low-duty-cycled wireless sensor networks*, in Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), 2011, pp. 1323–1331.
- [128] W. Ye, J. Heidemann, and D. Estrin, *An energy-efficient mac protocol for wireless sensor networks*, in Proceedings of the IEEE International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), 2002, pp. 1567–1576.
- [129] ———, *Medium access control with coordinated adaptive sleeping for wireless sensor networks*, IEEE/ACM Transactions on Networking, 12 (2004), pp. 493–506.
- [130] J. Yick, B. Mukherjee, and D. Ghosal, *Wireless sensor network survey*, Computer Networks, 52 (2008), pp. 2292–2330.
- [131] D. Yoo and S. Choi, *Medium access control with dynamic frame length in wireless sensor networks*, Journal of Information Processing Systems, 6 (2010), pp. 501–510.
- [132] H. Yoo, M. Shim, and D. Kim, *Dynamic duty-cycle scheduling schemes for energy-harvesting wireless sensor networks*, IEEE Communications Letters, 16 (2012), pp. 202–204.
- [133] T. You, H. Hassanein, and C. Yeh, *Semac: robust broadcast mac protocol for multi-hop wireless ad hoc networks*, in Proceedings of the IEEE International Performance, Computing, and Communications Conference (IPCCC), 2006, pp. 78–85.
- [134] Q. Zhang and D. Agrawal, *Dynamic probabilistic broadcasting in mobile ad hoc networks*, in Proceedings of the IEEE Vehicular Technology Conference (VTC), 2003, pp. 2860–2864.
- [135] ———, *Dynamic probabilistic broadcasting in manets*, Special issue of the Journal of Parallel and Distributed Computing (JPDC), 65 (2005), pp. 220–233.
- [136] ———, *Performance evaluation of leveled probabilistic broadcasting in manets and wireless sensor networks*, Simulation, 81 (2005), pp. 533–545.

- [137] Y. Zhang and L. Cheng, *Flossiping: A new routing protocol for wireless sensor networks*, in Proceedings of the IEEE International Conference on Networking, Sensing & Control (ICNSC), 2004, pp. 1218–1223.
- [138] Y. Zhao, R. Govindan, and D. Estrin, *Residual energy scans for monitoring wireless sensor networks*, in Proceedings of the IEEE International Wireless Communications and Networking Conference (WCNC), 2002, pp. 17–21.
- [139] Y. Zhao, C. Miao, M. Ma, J. Zhang, and C. Leung, *A survey and projection on medium access control protocols for wireless sensor networks*, ACM Computing Surveys, 45 (2012), pp. 7:1–7:37.
- [140] T. Zhu, Z. Zhong, T. He, and Z. Zhang, *Achieving efficient flooding by utilizing link correlation in wireless sensor networks*, IEEE/ACM Transactions on Networking, 21 (2013), pp. 121–134.
- [141] M. Zorzi, *A new contention-based mac protocol for geographic forwarding in ad hoc and sensor networks*, in Proceedings of the IEEE International Conference on Communications (ICC), 2004, pp. 3481–3485.
- [142] M. Zorzi and R. Rao, *Geographic random forwarding (geraf) for ad hoc and sensor networks: Energy and latency performance*, IEEE Transactions on Mobile Computing, 2 (2003), pp. 349–365.