

ALEXANDRE MASSAYUKI OKAZAKI

**ALGORITMO DE ROTEAMENTO BASEADO EM
COLÔNIA DE FORMIGAS COM HEURÍSTICAS
CONFIGURÁVEIS PARA REDES SENSORES SEM
FIO DE TOPOLOGIA DINÂMICA**

FLORIANÓPOLIS

2012

UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO
EM CIÊNCIA DA COMPUTAÇÃO

**ALGORITMO DE ROTEAMENTO BASEADO EM
COLÔNIA DE FORMIGAS COM HEURÍSTICAS
CONFIGURÁVEIS PARA REDES SENSORES SEM
FIO DE TOPOLOGIA DINÂMICA**

Dissertação submetida à
Universidade Federal de Santa Catarina
como parte dos requisitos para a
obtenção do grau de Mestre em Ciência da Computação

ALEXANDRE MASSAYUKI OKAZAKI

Florianópolis, Fevereiro de 2012

Catálogo na fonte pela Biblioteca Universitária
da
Universidade Federal de Santa Catarina

O41a Okazaki, Alexandre Massayuki
Algoritmo de roteamento baseado em colônia de formigas com heurísticas configuráveis para redes sensores sem fio de topologia dinâmica [dissertação] / Alexandre Massayuki Okazaki ; orientador, Antônio Augusto Fröhlich. - Florianópolis, SC, 2012.

1 v.: il., grafs., tabs.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Ciência da Computação.

Inclui referências

1. Informática. 2. Ciência da computação. 3. Sistemas de comunicação sem fio. 4. Algoritmos de computador. 5. Redes de computadores - Protocolos. I. Frohlich, Antonio Augusto Medeiros. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Ciência da Computação. III. Título.

CDU 681

**ALGORITMO DE ROTEAMENTO BASEADO EM COLÔNIA
DE FORMIGAS COM HEURÍSTICAS CONFIGURÁVEIS
PARA REDES SENSORES SEM FIO DE TOPOLOGIA
DINÂMICA**

ALEXANDRE MASSAYUKI OKAZAKI

Esta Dissertação foi julgada adequada para obtenção do Título de Mestre em Ciência da Computação, Área de concentração Sistemas de Computação, e aprovada em sua forma final pelo Programa de Pós Graduação em Ciência da Computação da Universidade Federal de Santa Catarina.

Antônio Augusto Fröhlich, Dr.
Orientador

Ronaldo dos Santos Mello, Dr.
Coordenador do Programa de Pós-Graduação em Ciência da
Computação

Banca Examinadora:

Lisandro Zambenedetti Granville, Dr.

Mario Dantas, Dr.

Mauro Roisenberg, Dr.

*Dedico esta dissertação à minha família,
aos meus pais, ao meu irmão e à minha
querida sobrinha Ariela.*

AGRADECIMENTOS

Agradecimento é a parte mais importante da nossa formação. Apenas sendo gratos é que aprendemos a reconhecer a ajuda do outro e o valor do nosso próprio trabalho. Agradecer a todos que ajudaram a construir esta dissertação não é tarefa fácil. O maior problema no agradecimento seletivo não é decidir quem incluir, mas decidir quem não mencionar.

Se serei seletivo, então é melhor começar por quem nunca poderei deixar de mencionar. O meu maior agradecimento é dirigido aos meus familiares, a quem dedico esta dissertação, por terem sido o contínuo apoio em todos estes anos, ensinando-me, principalmente, a importância da construção e coerência de meus próprios valores. Agradeço ao meu pai, especialmente, pelo apoio incondicional em tudo que diz respeito ao meu futuro e à minha vida. Agradeço à minha mãe pela preocupação intermitente com a minha felicidade. Agradeço ao meu irmão pela minha linda sobrinha.

Agradeço ao Laboratório de Integração *Software e Hardware*, o qual foi palco de diversos momentos de trabalho exaustivo, suor, debates, discussões, alegria e diversão. Agradeço aos amigos que fiz e às pessoas que conheci dentro desse palco. Agradeço em especial ao Mateus Krepsky Ludwich pela amizade, parceria e ajuda durante todo o mestrado. Agradeço ao Prof. Antônio Augusto Fröhlich por ter me ensinado a arte de pensar o trabalho acadêmico com rigor e disciplina, propiciando-me a fundamentação básica, sem a qual este trabalho não teria sido escrito. Suas sugestões levaram a sucessivas revisões do meu modo de pensar, cujas eventuais falhas, inteiramente responsabilidade do autor, teriam sido mais numerosas caso não fosse por sua crítica constante e incisiva.

Enfim, com um carinho especial, agradeço os meus amigos de

infância e os amigos que conquistei durante o mestrado. Dentre eles, Daniel Menegon, Tiago dos Santos Peres, Cassio Conti, Jéssica Bertol e Rayani Mariano são amigos que, de uma forma ou de outra, perto ou longe, contribuíram com sua amizade para a realização desta dissertação, gostaria de expressar minha profunda gratidão.

Resumo da Dissertação apresentada à UFSC como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciência da Computação.

**ALGORITMO DE ROTEAMENTO BASEADO EM COLÔNIA
DE FORMIGAS COM HEURÍSTICAS CONFIGURÁVEIS
PARA REDES SENSORES SEM FIO DE TOPOLOGIA
DINÂMICA**

Alexandre Massayuki Okazaki

Fevereiro / 2012

Orientador: Antônio Augusto Fröhlich, Dr..

Área de Concentração: Sistemas de Computação.

Palavras-chave: Redes Sensores Sem Fio, Otimização por Colônia de Formigas, Topologias Dinâmicas, Redes Ad Hoc Móveis.

Número de Páginas: 102

Há muitos desafios na concepção de protocolos de roteamento para redes sensores móveis, pois mudanças de topologia é um fator que afeta a taxa de entrega dos dados, o *overhead* de roteamento e a vida útil das redes sensores sem fio. No entanto, com a robustez dos protocolos de roteamento baseados em formigas para redes *ad hoc* móveis, lidar com topologias dinâmicas torna-se uma tarefa menos árdua para redes sensores. Esta dissertação apresenta o *Protocolo de Otimização de Saltos Dinâmicos baseado em Formiga* (ADHOP), um protocolo de roteamento reativo e autoconfigurável para redes sensores sem fio móveis. O ADHOP visa lidar com as restrições das redes sensores sem fio e ainda melhorar os processos de descoberta e manutenção de rotas através das formigas. No ADHOP, as formigas tomam decisões de roteamento baseando-se em heurísticas que variam conforme o contexto da rede. De acordo com o contexto em que a rede é aplicada, o roteamento deve priorizar taxa de entrega, latência, distância, força do sinal, memória utilizada, ou uma combinação destes e/ou outros fatores. O

algoritmo de roteamento proposto foi avaliado através das redes IEEE 802.11 e IEEE 802.15.4, comparando-o com protocolos de roteamento bem conhecidos. O ADHOP obtém melhores resultados em termos de taxa de entrega de dados, *overhead* de roteamento, e na prevenção de congestionamento para ambientes de topologia dinâmica. Os resultados também demonstram que o ADHOP, através da heurística de balanceamento de energia, é capaz de manter um melhor equilíbrio de energia a fim de prolongar o tempo de vida da rede enquanto aumenta a taxa de entrega de dados.

Abstract of Dissertation presented to UFSC as a partial fulfillment of the requirements for the degree of Master in Computer Science.

**ROUTING ALGORITHM BASED ON ANT COLONY WITH
CONFIGURABLE HEURISTICS FOR DYNAMIC
TOPOLOGY WIRELESS SENSOR NETWORKS**

Alexandre Massayuki Okazaki

February / 2012

Advisor: Antônio Augusto Fröhlich, Dr..

Area of Concentration: Computing Systems.

Keywords: Wireless Sensor Networks, Ant Colony Optimization, Dynamic Topologies, Mobile Ad Hoc Networks.

Number of pages: 102

There are many challenges in designing routing protocols for mobile sensor networks, and topology change is a factor that affects the data deliver ratio, routing overhead thus affecting the network lifetime of wireless sensor networks. However, with the robustness of ant-based routing protocols for mobile ad hoc networks, dealing with dynamic topologies becomes a less arduous task for sensor networks. In this work, we introduce the *Ant-based Dynamic Hop Optimization Protocol* (ADHOP), a self-configuring reactive routing protocol for mobile wireless sensor networks. ADHOP aims at dealing with the restrictions of sensor networks and yet improve the route discovery and the route maintenance through ants. In ADHOP, the ants make routing decisions based on the heuristic which varies depending on the context of the network. According to the context in which the network is applied, the routing should prioritize delivery rate, latency, distance, signal strength, memory used, or a combination thereof and/or other factors. The proposed routing algorithm has been evaluated through IEEE 802.11 and IEEE 802.15.4 networks, comparing it with well-

known routing protocols. ADHOP have obtained better results in terms of data delivery ratio, routing overhead, and congestion avoidance for environments of dynamic topology. Experimental results also show that ADHOP with the energy balancing heuristics is able to maintain better balancing energy to prolong the network lifetime while it increases the delivery ratio.

SUMÁRIO

LISTA DE FIGURAS	xix
LISTA DE TABELAS	xxiii
LISTA DE ABREVIATURAS E SIGLAS	xxv
1 INTRODUÇÃO	1
1.1 Motivação	3
1.2 Objetivos	7
1.3 Organização deste Documento	8
2 ALGORITMOS DE ROTEAMENTO PARA REDES AD HOC	11
2.1 Protocolos Pró-ativos	12
2.2 Protocolos Reativos	14
2.3 Protocolos Híbridos	16
2.4 Protocolos para Redes Sensores Sem Fio	17
2.4.1 Protocolos Centrados em Dados	18
2.4.2 Protocolos Hierárquicos	20

2.4.3	Protocolos Geográficos	22
2.4.4	Protocolos de Fluxo de Rede	24
2.5	Comparação	26
3	ALGORITMO PROPOSTO I: AD-ZRP	31
3.1	O Algoritmo HOPNET	32
3.2	Do HOPNET ao AD-ZRP	39
3.3	Avaliação: HOPNET x AD-ZRP	47
3.4	Discussão	52
4	ALGORITMO PROPOSTO II: ADHOP	55
4.1	Método de Roteamento de Saltos Dinâmicos	56
4.2	Estruturas de Tabela e Formiga	60
4.3	O roteamento ADHOP	63
4.4	Heurísticas	70
5	EXPERIMENTAÇÃO E AVALIAÇÃO	73
5.1	Primeiro Estágio: <i>IEEE 802.11</i>	73
5.1.1	Avaliação - Mobilidade	74
5.1.2	Avaliação - Heurísticas	76
5.2	Segundo Estágio: <i>IEEE 802.15.4</i>	79
5.2.1	Avaliação - Mobilidade	82
5.2.2	Avaliação - Heurísticas	87

6 CONCLUSÃO	93
Referências	97

LISTA DE FIGURAS

1	Classificação dos Algoritmos de Roteamento <i>Ad Hoc</i> . . .	11
2	Roteamento pró-ativo.	13
3	Roteamento reativo (AODV) - Requisição de Rota. . .	15
4	Roteamento reativo (AODV) - Descoberta de Rota. . .	16
5	Roteamento híbrido (ZRP/HOPNET).	17
6	Roteamento centrado em dados (AACS + MST) - Propagação das mensagens de interesse pela rede. . . .	19
7	Roteamento centrado em dados (AACS + MST) - Construção da MST.	20
8	Roteamento hierárquico.	21
9	Roteamento geográfico (GRLI).	23
10	Roteamento de fluxo de rede (congestionamento - ARCC).	25
11	Protocolos híbridos: zonas - raio igual a 2.	33
12	HOPNET - Estrutura da formiga.	33
13	HOPNET - FEIs periódicas e recepção de pacotes. . . .	34
14	HOPNET - Recepção de FEIs.	35
15	HOPNET - Transmissão de Dados.	36
16	HOPNET - Recepção de FEEs.	37
17	HOPNET - Recepção de FRs.	38

18	AD-ZRP - Zonas dinâmicas.	40
19	AD-ZRP - Mudança de forma da zona.	41
20	AD-ZRP - Estruturas da Formiga.	41
21	AD-ZRP - Transmissão de Dados.	43
22	AD-ZRP - Recepção de FTEs.	44
23	AD-ZRP - Recepção de FTIs.	45
24	Taxa de entrega de dados.	49
25	Taxa de falha de <i>links</i>	50
26	<i>Overhead</i> de roteamento.	51
27	HOPNET - Roteamento reativo.	53
28	Trilha de feromônio.	58
29	Escolhendo uma trilha.	58
30	Saltos dinâmicos.	59
31	ADHOP - Estrutura da tabela de roteamento.	61
32	ADHOP - Tabela de feromônio (<i>MAX_HASH</i> == 3 e <i>MAX_LIST</i> == 4).	61
33	ADHOP - Tipos de formigas.	62
34	ADHOP - Estrutura da formiga.	62
35	Roteamento ADHOP.	64
36	Falhas, mobilidade e caminhos alternativos.	65
37	ADHOP - Transmissões de dados.	66
38	ADHOP - Recepção da FTE.	68
39	ADHOP - Recepção da FT.	69
40	ADHOP - Heurísticas.	71

41	Taxa de entrega de dados.	75
42	<i>Overhead</i> de roteamento.	76
43	Desvio padrão do tempo de vida dos nodos.	78
44	Tempo de vida esperado.	79
45	Taxa de entrega de pacotes de dados.	80
46	<i>Overhead</i> de roteamento.	83
47	Taxa de falha de <i>links</i>	85
48	Taxa de quebra de rotas.	86
49	Taxa de entrega de pacotes de dados.	88
50	Desvio padrão do consumo de energia.	89
51	Consumo médio de energia.	91
52	Taxa de entrega de pacotes de dados.	91

LISTA DE TABELAS

1	Comparação dos protocolos <i>ad hoc</i>	27
2	Configuração do GloMoSim.	48
3	Configuração do GloMoSim.	74
4	Configuração energética do GloMoSim.	77
5	Configuração do OMNeT++.	81
6	Configuração energética do OMNeT++.	88

LISTA DE ABREVIATURAS E SIGLAS

RSSF, Rede Sensor Sem Fio

ACO, Otimização por Colônia de Formigas - *Ant Colony Optimization*

MANET, Rede Ad Hoc Móvel - *Mobile Ad hoc Network*

OLSR, Roteamento por Estado de Conectividade Otimizado - *Optimized Link State Routing*

DSDV, Vetor de Distâncias de Destinos Sequenciados - *Destination-Sequenced Distance-Vector*

AODV, Vetor de Distâncias Sob Demanda Ad hoc - *Ad hoc On Demand Distance Vector*

RREQ, Requisição de Rota - *Route Requests*

RREP, Resposta de Rota - *Route Replies*

RERR, Erros de Rota - *Route Errors*

DYMO, Protocolo Dinâmico e Sob Demanda para MANETs - *Dynamic On-demand MANET routing protocol*

ZRP, Protocolo de Roteamento por Zonas - *Zone Routing Protocol*

AACS, Sistema Adaptativo de Colônia de Formigas - *Adaptive Ant Colony System*

MST, Árvore Mínima de Steiner - *Minimum Steiner Tree*

NDN, Rede de Dados Nomeados - *Named Data Networking*

LEACH, Agrupamento Hierárquico e Adaptativo de Baixa Energia - *Low-Energy Adaptive Clustering Hierarchy*

EE-VAR, Roteamento Energeticamente Eficiente de Área Variável - *Energy Efficient Variable Area Routing*

GEAR, Roteamento Geográfico e Consciente de Energia - *Geographic and Energy Aware Routing*

GRLI, Roteamento Geográfico sem Informação de Localização - *Geographic Routing without Location Information*

RACE, Protocolo de encaminhamento geográfico consciente das condições de rede para aplicações de tempo real em redes móveis de sensores sem fio - *network conditions Aware geographical forwarding protocol for real-time applications in mobile wireless sensor network*

AOER, Rotas de Energia Sob Demanda baseado em Formigas - *Ant-based On-demand Energy Route*

ARCC, Roteamento baseado em Formiga com Controle de Congestionamento - *Ant based Routing with Congestion Control*

FROMS, Roteamento de Retorno de Informações para Otimização de Múltiplas Estações Base - *Feedback Routing for Optimization Multiple Sinks*

AD-ZRP, Protocolo para Roteamento em Zonas Dinâmicas baseado em Formiga - *Ant-based Dynamic Zone Routing Protocol*

FEI, Formiga de Exploração Interna

FEE, Formiga de Exploração Externa

FR, Formiga de Retorno

FN, Formiga de Notificação

FE, Formiga de Erro

FTI, Formiga Transportadora Interna

FTE, Formiga Transportadora Exploratória

GloMoSim, Simulador de Sistemas de Informação Móvel Global -

Global Mobile Information System Simulator

RWP, Coordenadas Intermediárias Aleatórias - *Random Waypoint*

MAC, Controle de Acesso ao Meio - *Medium Access Control*

CTS, Pronto para enviar - *Clear to send*

ACK, Confirmação de recebimento - *Acknowledgment*

FT, Formiga Transportadora

FTE, Formiga Transportadora de Exploração

MM, Mobilidade em Massa - *Mass Mobility*

MAC, Controle de Acesso ao Meio - *Medium Access Control*

EPOS, Sistema Operacional Paralelo Embarcado - *Embedded Parallel Operating System*

C-MAC, MAC-Configurável - *Configurable-MAC*

1 INTRODUÇÃO

Protocolos da camada de rede estão basicamente relacionados à transferência de pacotes da origem ao destino, os quais geralmente exigem diversos saltos entre os roteadores intermediários do percurso. Portanto, trata-se da camada mais baixa que visa transferências fim-a-fim. No entanto, para cumprir tal função, a camada de rede deve geralmente conhecer o conjunto de todos os roteadores para escolher, através de algoritmos de roteamento, os caminhos mais apropriados (TANENBAUM, 2002).

Algoritmos de roteamento são responsáveis pela tomada de decisão sobre o próximo salto a ser utilizado na transmissão do pacote na direção de um destino específico. O algoritmo de roteamento age nos processos de preenchimento e atualização das tabelas de roteamento, ou seja, nos processos de descoberta e manutenção de rotas na rede. No entanto, para realizar tais processos, certas propriedades são desejáveis nos algoritmos de roteamento (TANENBAUM, 2002):

- **Correção:** esta propriedade consiste na preservação intermitente do estado funcional da rede através de procedimentos de reparo de rotas;
- **Simplicidade:** consiste em obter um menor grau de complexidade dos procedimentos de roteamento;
- **Robustez:** o algoritmo de roteamento deve ser capaz de *aceitar* as alterações na topologia e no fluxo de dados evitando que os demais dispositivos (nodos) sejam interrompidos ou que a rede seja reiniciada;
- **Estabilidade:** consiste em resistir às mudanças de topologia e fluxo de dados, um algoritmo estável alcança o estado de equilíbrio e tende a permanecer em equilíbrio;
- **Otimização:** consiste em encontrar “rotas ótimas” na rede, isto é, que resulte no melhor desempenho possível do que se

busca otimizar, como por exemplo, minimização da latência e maximização do tempo de vida da rede.

Porém, em redes sem infraestrutura predefinida ou redes *ad hoc* sem fio, as usuais regras sobre topologia estática/fixa, vizinhos conhecidos, endereçamento de rede vinculado à localização e entre outras não são aplicáveis. *Ad hoc* é uma expressão latina que literalmente significa “para esta finalidade” ou “com este objetivo”. Geralmente se refere a soluções destinadas a atender a uma necessidade específica ou resolver um problema imediato. Portanto, tem um caráter temporário. Em redes *ad hoc*, o processo nunca é planejado ou preparado antecipadamente. Os terminais funcionam como roteadores encaminhando de forma comunitária as comunicações advindas dos terminais vizinhos. Nestas redes não há topologia predeterminada, nem controle centralizado. Ou seja, elas não requerem uma infraestrutura. Os nodos se comunicam através de conexões físicas entre eles, na qual os dispositivos fazem parte dela apenas durante a sessão de comunicação, enquanto estão a certa proximidade do restante da rede. Redes *ad hoc* incluem um modo de operação distribuído ponto-a-ponto, roteamento de múltiplos saltos e mudanças relativamente constantes na distribuição dos nodos da rede. Além disso, a responsabilidade pela organização e controle da rede tende a ser distribuída entre os próprios nodos. Portanto, alcançar propriedades como correção, simplicidade, robustez, estabilidade e otimização em redes *ad hoc* torna-se um grande desafio, especialmente em redes móveis. Redes móveis são redes cuja topologia tende a se alterar constantemente devido à mobilidade dos nodos (WANG; WU; TSENG, 2010). No entanto, em redes sem fio, as causas de uma topologia dinâmica podem ser definidas pelo movimento físico dos nodos, mudanças no canal do rádio, desempenho da rede e falhas de nodos (SHELBY; BORMANN, 2010):

- **Movimento físico:** a razão mais evidente para mobilidade em uma rede é quando os nodos se movem fisicamente em relação aos demais;
- **Canal do rádio:** mudanças no ambiente afetam a propagação das transmissões de rádio (enfraquecimento do sinal ou *fading*), causando a perda de conectividade entre os nodos;

- **Desempenho da rede:** perda de dados e latência da entrega de dados em redes sem fio são geralmente causadas pelo fraco sinal do rádio, colisões, sobrecarga da capacidade do canal e congestionamento;
- **Falhas de nodos:** nodos autônomos sem fio são propensos a erros, principalmente devido ao esgotamento das baterias dos nodos, causando repentinas mudanças de topologia na rede.

Deste modo, o roteamento em redes *ad hoc* e/ou em redes móveis torna-se bem distinto do roteamento em redes estruturadas, exigindo algoritmos adaptáveis com a finalidade de lidar com a imprevisibilidade deste tipo de rede (GARCIA-MACIAS; GOMEZ, 2007). Algoritmos de roteamento adaptativos são algoritmos que mudam as decisões de roteamento a fim de refletir as mudanças na topologia e no tráfego de dados da rede (TANENBAUM, 2002). Em consequência disso, tais algoritmos são soluções adequadas para lidar com redes *ad hoc* e/ou móveis.

1.1 Motivação

Rede Sensor Sem Fio (RSSF) é um tipo de rede *ad hoc* que consiste basicamente de nodos sensores espacialmente distribuídos para a monitoração de um determinado fenômeno em estruturas, máquinas, ambientes ou pessoas. Este tipo de rede possui aplicações na automação e controle de sistemas, prédios e cidades inteligentes, logística e transporte, e apresenta soluções em vigilância e segurança de ambientes. Nodos sensores geralmente apresentam limitações no poder de processamento, memória, largura de banda e capacidade de energia (MATROUK; LANDFELDT, 2009). Tais dispositivos são tipicamente utilizados em ambientes de difícil acesso ou em regiões perigosas. Como consequência, falhas e danos aos nodos tornam-se eventos comuns. Lidar com tais falhas sem afetar a taxa de entrega, o consumo de energia e outros aspectos da qualidade de rede é um grande desafio. Deste modo, técnicas de roteamento que lidam eficientemente e dinamicamente com mudanças de topologia tornam-se

uma necessidade para RSSFs. O desafio tende a aumentar quando tais mudanças são causadas pela mobilidade dos nodos (GARCIA-MACIAS; GOMEZ, 2007). Protocolos de roteamento para RSSFs são geralmente projetados para nodos sensores fixos/estáticos e homogêneos com apenas um nodo central ou estação base receptora (MARRÓN; MINDER, 2006). Porém, redes heterogêneas e cooperativas, que coletam informações para diversas estações bases, podem utilizar nodos e/ou estações bases móveis. Portanto, algoritmos de roteamento auto-organizáveis para RSSFs que lidam com o *overhead* das mudanças de topologia e/ou mobilidade têm despertado significativo interesse na comunidade científica (AKKAYA; YOUNIS, 2005; WATTEYNE et al., 2010). Mobilidade é uma importante característica da rede e está presente em diversos cenários de aplicações, permitindo procedimentos de implantação automática dos nodos, ajuste flexível da topologia e reação rápida a eventos (WANG; WU; TSENG, 2010). Em RSSFs, sensores podem ser anexados a pessoas, animais, veículos ou robôs autônomos a fim de coletar, armazenar ou transmitir informações de um determinado ambiente (MARRÓN; MINDER, 2006). Suas aplicações possuem a finalidade de exploração de ambientes de difícil acesso humano, reforçando a necessidade de algoritmos de roteamento que podem se adaptar às variações de topologia. Um projeto desenvolvido em Esslingen (FESTO, 2012), no interior da Alemanha, visa captar a eficiência física dos animais para nadar, voar, correr e manipular objetos a fim de utilizar essa habilidade natural em robôs. Quanto mais realísticos esses animais biônicos são, mais úteis eles se tornam para a indústria naval, automobilística e aeroespacial, além do sensoriamento das águas e do céu que tal projeto pode proporcionar.

Todos os protocolos de roteamento são capazes de lidar com falhas de nodos e mobilidade. Porém, a reação a esta variação de topologia pode requerer algum tempo, implicando em perda de desempenho. Além disso, relacionar mobilidade a dispositivos que são usualmente alimentados por baterias impõe diversas restrições nos projetos de protocolos para RSSFs, da camada física a camada de aplicação (ZENG et al., 2009; ANASTASI et al., 2009).

A maioria dos protocolos de roteamento que lidam com mudanças de topologia tende a descartar a rota inválida e compu-

tar uma nova, implicando em perda de desempenho e desperdício de recursos. Portanto, soluções cujas decisões de roteamento são tomadas em relação a alguma informação coletada pelo próprio dispositivo tendem a lidar melhor com topologias dinâmicas. Muitas técnicas de roteamento visam obter *links* confiáveis para as transmissões entre nodos sensores a fim de melhorar a taxa de entrega de dados, visto que RSSFs são ruidosas e propensas a erros. Assim, soluções que monitoram a qualidade do *link* tipicamente se baseiam em métricas, tais como força do sinal do rádio, latência, taxa de entrega, localização, e até mesmo heurísticas a fim de obter e manter rotas confiáveis entre nodos (STANKOVIC, 2008). Protocolos de roteamento baseados na meta-heurística *Otimização por Colônia de Formigas* (do inglês *Ant Colony Optimization - ACO*) são exemplos de soluções que utilizam heurísticas na resolução de problemas de roteamento. ACO é uma meta-heurística utilizada na resolução de problemas discretos de otimização (DORIGO; BLUM, 2005; DORIGO; BIRATTARI; STUTZLE, 2006). Algoritmos de roteamento baseados em ACO exploram os mecanismos de otimização computacional para a construção e manutenção de rotas eficientes (CARO, 2004a). Estes protocolos inspiram-se na inteligência coletiva das formigas com a finalidade de trabalhar de modo distribuído, utilizando agentes móveis (formigas) para aprender rotas mais curtas e adaptar-se às mudanças na topologia e no tráfego da rede (LU; ZHAO; SU, 2004; IYENGAR et al., 2007). Formigas se comunicam umas com as outras utilizando feromônio. Elas produzem diversos tipos de feromônio a fim de atrair parceiros, sinalizar perigo para a colônia ou para dar instruções sobre a localização. As formigas geralmente usam a superfície do solo para deixar trilhas de feromônio para que outras formigas a sigam. Tal comportamento explica como as formigas exercem notável cooperação na construção de uma colônia, inspirando técnicas de otimização e protocolos de roteamento. Deste modo, protocolos de roteamento baseados em ACO são alternativas para diversas redes *ad hoc* e escaláveis (CARO, 2004b; CARO; DUCATELLE; GAMBARDELLA, 2005; WANG et al., 2009), incluindo RSSFs (OKDEM; KARABOGA, 2009; DHURANDHER et al., 2009).

Rede Ad Hoc Móvel (do inglês *Mobile Ad hoc Network - MANET*) é um sistema autônomo de nodos/roteadores móveis conectados

por *links* sem fio. Os nodos/roteadores são livres para se locomoverem e se auto-organizarem arbitrariamente, resultando em redes cujas topologias mudam de forma intermitente e imprevisível. RSSFs são definidas por uma aplicação específica a fim de alcançar um determinado objetivo, isto é, a monitoração do fenômeno o qual será analisado. De acordo com o fenômeno, a substituição ou recarga das baterias dos dispositivos torna-se uma atividade difícil ou até mesmo impossível. Lidar com energia é um dos maiores desafios no projeto de sistemas embarcados para RSSFs, especialmente em sistemas móveis em que, dependendo do cenário, a recarga das baterias é custosa ou impraticável. Consequentemente, algoritmos eficientes energeticamente são necessários para o uso apropriado dos recursos do sistema a fim de não drenar energia sem necessidade. Assim, o consumo de energia torna-se um fator crítico no projeto de protocolos de RSSFs, especialmente nos protocolos de roteamento. Nodos sensores tipicamente propagam pacotes de dados através de múltiplos saltos até um destino específico (estação base). Logo, difundir pacotes intermitentemente pela rede móvel a fim de sempre alcançar um nodo destino distante é uma opção energeticamente ruim. Por outro lado, a maioria dos protocolos de roteamento visa à construção de rotas ótimas, suscetíveis para serem utilizadas nas transmissões seguintes. No entanto, o uso excessivo de tais rotas tende a acelerar o consumo de energia dos nodos, principalmente em redes de baixa potência e propensas a erros como RSSFs (KUMAR; KUMAR; PATEL, 2010). No pior caso, a rede pode ser dividida em partições disjuntas pelo esgotamento de energia da bateria de um dos roteadores. Não obstante, o consumo de energia não é o único fator crítico para o roteamento em RSSFs. De acordo com o contexto em que a rede é aplicada, o roteamento deve priorizar taxa de entrega, latência, distância, força do sinal, memória utilizada, ou uma combinação destes e/ou outros fatores. Cabe ao projeto da aplicação e dos protocolos a identificação do contexto a fim de refletir os objetivos e as reais necessidades das RSSFs.

“Contexto é qualquer informação que pode ser usada para caracterizar a situação de uma entidade. Uma entidade pode ser uma pessoa, um local ou um objeto que é considerado relevante na interação entre um usuário e

uma aplicação, incluindo o usuário e a aplicação” (DEY; ABOWD, 1999).

Um sistema consciente de contexto pode ser definido como um sistema que pode extrair, interpretar e usar as informações do contexto, adaptando as funcionalidades de acordo com o uso do contexto atual (MARRÓN; MINDER, 2006). Tal concepção pode ser estendida para o uso de diversas informações incluindo dados e métricas de ambiente, e aplicada em protocolos de roteamento a fim de adaptar os parâmetros de acordo com o contexto do nodo ou da rede. No entanto, o desafio de tais protocolos é a complexidade de lidar com informações contextuais de forma eficiente, além da capacidade de deduzir o significado corretamente. Especialmente em computação móvel, onde o contexto tende a mudar rapidamente e frequentemente.

1.2 **Objetivos**

O objetivo principal desta dissertação é o desenvolvimento de um algoritmo de roteamento para redes móveis inspirado na meta-heurística ACO. Tal algoritmo visa se adaptar a frequentes mudanças de topologia e necessidades da rede através de heurísticas configuráveis. As heurísticas, nessa abordagem, introduzem um método de roteamento que auxilia o cálculo da melhor rota para um destino específico de acordo com a quantidade de feromônio no nodo e alguma informação contextual dos vizinhos ou da rede. O conceito de contexto desta abordagem inclui diversas informações que se referem ao estado ou condição dos nodos e da rede, tais como localização, cobertura e densidade dos nodos, mobilidade, limitações de *hardware*, energia, taxa de entrega entre outras informações que são identificadas como informações heurísticas a fim de auxiliar as decisões de roteamento. A heurística de um roteamento deve particularmente indicar os caminhos e possibilidades a serem aprofundadas na tentativa de tornar o roteamento adequado à aplicação. Por exemplo, uma heurística que visa o balanceamento do consumo de energia dos nodos deve disseminar feromônio pela rede de modo que as rotas sejam distribuídas

para balancear o consumo de energia na comunicação entre os nodos. Além disso, o algoritmo proposto deve visar uma eficiente taxa de entrega, bem como uma redução do *overhead* do tráfego de pacotes de controle na rede, melhorando a descoberta e a manutenção de rotas ainda consciente das usuais limitações das RSSFs.

Com a finalidade de alcançar tais objetivos, as seguintes atividades foram definidas:

- Estudar (revisão bibliográfica) algoritmos de roteamento existentes para redes *ad hoc*, redes móveis e redes sensores;
- Estudar técnicas de roteamento focadas em mobilidade;
- Estudar técnicas de roteamento focadas nas limitações de redes sensores;
- Estudar técnicas de roteamento focadas na meta-heurística ACO;
- Desenvolver o algoritmo proposto inspirando-se nas técnicas estudadas:
 - Implementar um algoritmo de roteamento estudado;
 - Propor e modelar um algoritmo;
 - Implementar o algoritmo proposto;
 - Testar implementações;
- Analisar os resultados obtidos.

1.3 Organização deste Documento

O restante da dissertação está organizado da seguinte forma:

- A fundamentação teórica consiste na apresentação dos principais conceitos teóricos necessários para o desenvolvimento deste trabalho.

- **Capítulo 2** descreve os principais conceitos e tecnologias de roteamento, inclusive em redes de sensores sem fios.
- O desenvolvimento consiste na exibição do assunto do trabalho e do método empregado de maneira ordenada e detalhada da forma mais ampla e relevante possível.
 - **Capítulo 3** descreve e analisa o algoritmo de roteamento proposto (AD-ZRP), comparando-o com o HOPNET e retratando as estruturas definidas e as funcionalidades atribuídas para lidar com topologias dinâmicas.
 - **Capítulo 4** descreve e analisa o algoritmo de roteamento proposto (ADHOP), retratando as estruturas definidas e as funcionalidades atribuídas para lidar com topologias dinâmicas.
 - **Capítulo 5** avalia e compara o algoritmo modelado e implementado com diversos algoritmos para redes *ad hoc*.
- As considerações finais consistem nas discussões e conclusões finais do trabalho a partir das teses iniciais e objetivos.
 - **Capítulo 6** conclui a dissertação e apresenta os trabalhos em andamento e futuros.

2 ALGORITMOS DE ROTEAMENTO PARA REDES AD HOC

Algoritmos de roteamento podem ser classificados através de diversos critérios a fim de exaltar as características e modos de operação. Como ilustrado na Figura 1, algoritmos de roteamento para redes *ad hoc* podem ser classificados pelos seguintes critérios: organização, composição, mobilidade, fluxo de informação e modo de construção/manutenção de rotas (LIU, 2005).

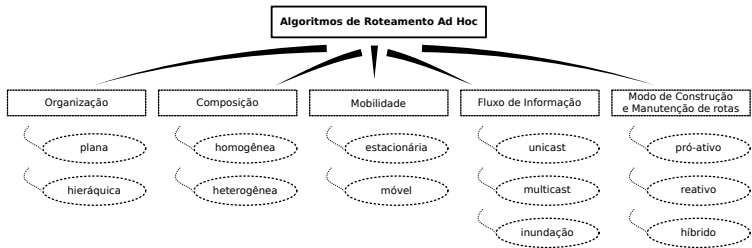


Figura 1: Classificação dos Algoritmos de Roteamento *Ad Hoc*.

No critério *organização*, o roteamento se divide entre plano e hierárquico. No roteamento plano, todos os nodos apresentam as mesmas responsabilidades na rede. Tais algoritmos tendem a trocar e compartilhar informações a respeito da topologia. No hierárquico, os nodos geralmente se organizam em grupos ou aglomerados estabelecendo hierarquias e responsabilidades distintas entre eles. No critério *composição*, o roteamento se divide entre homogêneo e heterogêneo. Os nodos que compõe redes homogêneas se apresentam com as mesmas funcionalidades e capacidades físicas, enquanto os nodos de redes heterogêneas tendem a apresentar funcionalidades ou capacidades físicas distintas. O critério de *mobilidade* classifica os algoritmos quanto à capacidade de suportar mobilidade durante o roteamento. No critério de *fluxo de informação*, o roteamento se divide entre *unicast*, *multicast* e inundação. Em *unicast*, os nodos tendem a se comunicar diretamente com o nodo destino utilizando protocolos de rotea-

mento de múltiplos saltos. Em *multicast*, os nodos formam grupos e utilizam o roteamento para a entrega simultânea de informação entre os membros de um mesmo grupo. Em inundação, os nodos difundem as informações para os nodos vizinhos que também difundem tais informações até alcançar o nodo destino.

Os protocolos de roteamento são algoritmos que constituem rotinas com o objetivo de mapear a topologia da rede. Esse mapeamento é realizado através das tabelas de roteamento, cuja construção, atualização e manutenção variam de acordo com o método de roteamento escolhido. Por isso, apesar dos diversos critérios de classificação apresentados, nesta dissertação os protocolos de roteamento serão classificados utilizando apenas o critério de *modo de construção/manutenção de rotas*: pró-ativo, reativo e híbrido. RSSF, compreendida como uma subclasse de rede *ad hoc*, também pode usar tal critério para classificar os algoritmos de roteamento. Porém, as RSSFs diferem das redes tradicionais *ad hoc* em diversos aspectos, tais como limitações de recursos e tempo de vida da rede. O principal objetivo deste tipo de algoritmo de roteamento é considerar tais limitações a fim de minimizar os problemas relativos às características das RSSFs, empregando técnicas que visam otimizar o roteamento. Desta forma, neste trabalho os algoritmos de roteamento para RSSFs são classificados separadamente.

2.1 Protocolos Pró-ativos

Protocolos de roteamento pró-ativos geralmente são soluções de baixa latência, isto é, nodos podem rapidamente obter rotas para qualquer outro nodo da rede, pois no roteamento pró-ativo as rotas são construídas antes da necessidade de utilizá-las. Cada nodo compartilha/difunde e captura informações de topologia e conectividade periodicamente a fim de manter as tabelas de roteamento atualizadas e precisas, como ilustrado na Figura 2.

Um exemplo de protocolo de roteamento pró-ativo é o *Roteamento por Estado de Conectividade Otimizado* (do inglês *Optimized*

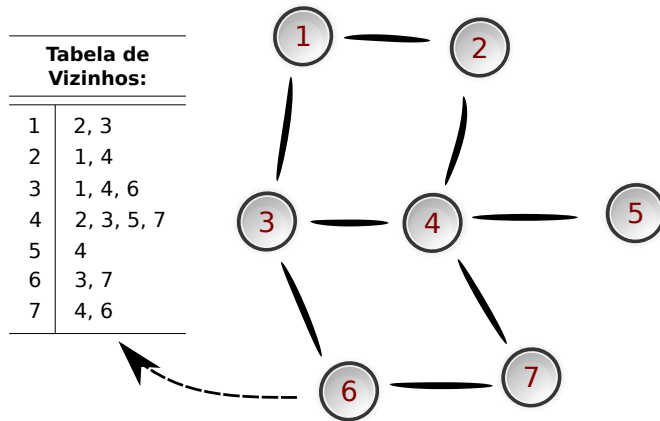


Figura 2: Roteamento pró-ativo.

Link State Routing - OLSR), um protocolo que mantém as rotas baseadas na qualidade de conectividade entre os nodos da rede (CLAUSEN; JACQUET, 2003). Neste protocolo, os nodos difundem mensagens de controle sobre informações de topologia periodicamente e intermitentemente a fim de disseminar as informações de conectividade entre os vizinhos pela rede. Tais informações são armazenadas nas tabelas de roteamento e utilizadas para computar os saltos que os pacotes deverão realizar a fim de alcançar o destino. Baseado no estado de conectividade da rede, cada nodo seleciona alguns nodos vizinhos denominados *retransmissores de múltiplos pontos*. Somente estes nodos são capazes de retransmitir pacotes, visando diminuir o *overhead* da rede.

Vetor de Distâncias de Destinos Sequenciados (do inglês *Destination-Sequenced Distance-Vector - DSDV*) é outro protocolo de roteamento pró-ativo, baseado no algoritmo de *Bellman-Ford*, que visa garantir rotas livres de ciclos na rede (PERKINS; BHAGWAT, 1994). DSDV modela nodos cooperativos que periodicamente anunciam a topologia de interconexão entre os nodos para prover informações precisas de conectividade a fim de lidar com mobilidade em redes

ad hoc. Um ponto importante do DSDV é a capacidade de manter informações de quando uma entrada na tabela de roteamento foi alterada e o ator desta ação.

2.2 Protocolos Reativos

Protocolos de roteamento reativos são soluções que, diferentemente dos pró-ativos, não computam rotas antes de serem realmente necessárias. O objetivo de tais protocolos é manter a rede “leve”, com baixo *overhead* de pacotes de controle. Ou seja, as rotas não são previamente computadas, mas sim computadas sob demanda. Tais protocolos são caracterizados pela simplicidade e flexibilidade em lidar com redes *ad hoc*, mas também são caracterizados pela latência gerada ao se definir as rotas.

Vetor de Distâncias Sob Demanda Ad hoc (do inglês *Ad hoc On Demand Distance Vector - AODV*) é o protocolo de roteamento referência em redes *ad hoc* (PERKINS; ROYER, 1999). AODV é um protocolo reativo projetado para MANETs, o qual computa as rotas através de um ciclo de requisições e respostas utilizando mensagens de controle. *Requisição de Rota* (do inglês *Route Requests - RREQ*), *Resposta de Rota* (do inglês *Route Replies - RREP*) e *Erros de Rota* (do inglês *Route Errors - RERR*) são os tipos de mensagens definidas no AODV. Quando um nodo particular precisa de uma rota até um destino específico, as mensagens RREQ são difundidas pela rede, como ilustrado na Figura 3. A resposta, através da mensagem RREP, indica uma rota pronta para ser utilizada, como ilustrado na Figura 4. O AODV tende a manter as rotas requisitadas na tabela de roteamento por um determinado tempo. Qualquer constatação de que a topologia da rede foi alterada, com possíveis rotas invalidadas, as mensagens RERRs são transmitidas pela rede a fim de notificar os devidos nodos e iniciar um novo processo de descoberta de rota, se necessário.

Protocolo Dinâmico e Sob Demanda para MANETs (do inglês *Dynamic On-demand MANET routing protocol - DYMO*) é outro pro-

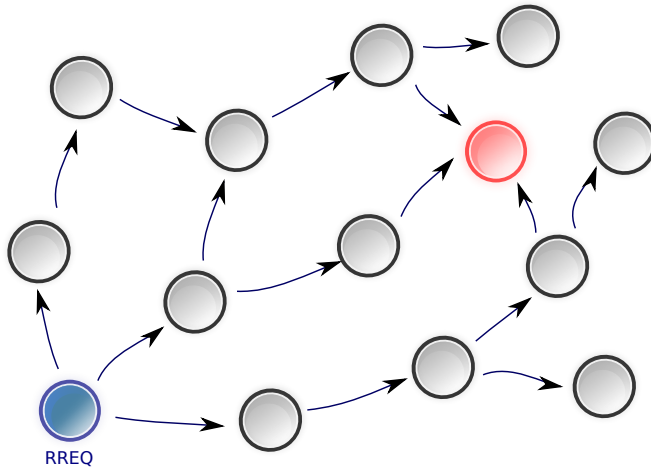


Figura 3: Roteamento reativo (AODV) - Requisição de Rota.

protocolo de roteamento reativo específico para MANETs (CHAKERES; PERKINS, 2009). Similar ao AODV, DYMO apresenta procedimentos, de descoberta e manutenção de rotas, mais básicos e simples. As operações de roteamento deste protocolo se dividem entre a operação de descoberta de rotas e a operação de manutenção de rotas. A descoberta de rotas se baseia nos procedimentos de requisição e resposta de rotas, semelhante ao AODV com RREQs e RREPs (Figuras 3 e 3). Durante tais procedimentos, os nodos intermediários que recebem RREQs gravam as rotas para o nodo origem enquanto os nodos intermediários que recebem RREPs gravam as rotas para o nodo destino. A manutenção de rotas ocorre de duas maneiras:

- a fim de preservar as rotas em uso, o algoritmo tende a aumentar o tempo de vida das mesmas ao retransmitir pacotes corretamente;
- o pacote RERR tende a ser enviado ao nodo origem com a finalidade de notificar a ausência ou invalidez do nodo destino.

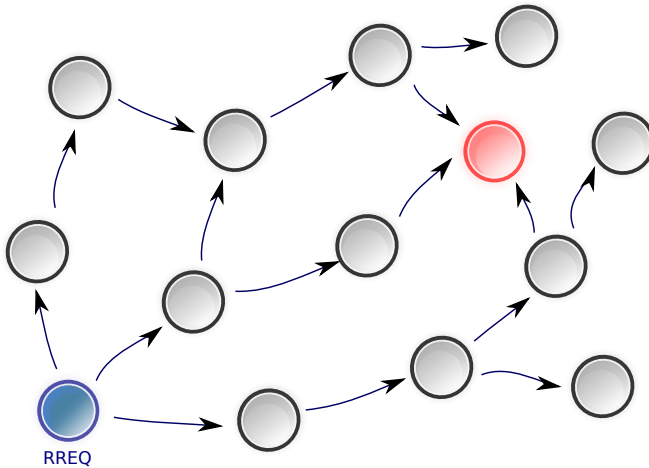


Figura 4: Roteamento reativo (AODV) - Descoberta de Rota.

2.3 Protocolos Híbridos

Protocolos de roteamento híbridos visam combinar os esquemas de roteamento pró-ativos e reativos a fim de unificar duas características complementares destes tipos de protocolos. Tais algoritmos visam um roteamento de baixa latência (pró-ativos) que consiga manter um baixo *overhead* de roteamento (reativos).

O primeiro protocolo a atender tal desafio foi o *Protocolo de Roteamento por Zonas* (do inglês *Zone Routing Protocol - ZRP*) (HAAS, 1997). O ZRP usa o conceito de zonas para o roteamento híbrido, como ilustrado na Figura 5. Uma zona é definida por um nó central que a partir de um número específico de saltos, denominado raio da zona, define uma “área” na qual ocorrerá o roteamento pró-ativo. Qualquer roteamento para fora das zonas é realizado de forma reativa. Diferentemente do conceito de grupos (*cluster*), cada nó na rede possui e mantém a própria zona. Deste modo, o ZRP mantém o roteamento pró-ativo em uma parte limitada da rede, reduzindo o *overhead* de roteamento, ao passo que o roteamento reativo tende a atender o

restante da rede, reduzindo a latência da rede ao computar rotas.

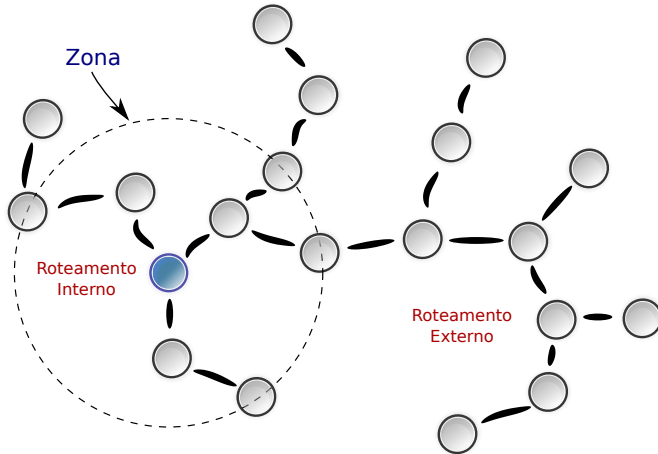


Figura 5: Roteamento híbrido (ZRP/HOPNET).

HOPNET é um protocolo de roteamento híbrido baseado no ZRP (WANG et al., 2009). Diferentemente, HOPNET envolve ACO no roteamento pró-ativo das zonas a fim de obter informações confiáveis de conectividade entre os nodos de forma mais eficiente, seja para manter e melhorar as rotas existentes ou explorar melhores opções.

2.4 Protocolos para Redes Sensores Sem Fio

Sensores são dispositivos que recebem e respondem a sinais ou estímulos, como temperatura e pressão, convertendo os dados obtidos em sinais eletrônicos. RSSFs podem ser caracterizadas pelo uso de uma grande quantidade de nodos sensores com capacidade de se comunicar. Esses nodos podem ser “colocados” dentro do fenômeno a ser analisado ou próximo a ele. As posições de cada nodo não são predeterminadas ou previamente calculadas. Elas são aleatórias, visto que a implantação de RSSFs em locais de difícil acesso pode ocorrer

pelo uso de veículos, “soltando” os nodos pela região alvo, ou através de mecanismos de mobilidade. Devido à escalabilidade característica de RSSFs, a comunicação geralmente é realizada com um nodo transmitindo a outro e repassando os valores do sensoriamento até o destino especificado. Porém, RSSF, apesar de ser considerada uma subclasse de rede *ad hoc*, não faz uso dos protocolos de gerenciamento de energia desenvolvidos para as MANETs, pois a natureza aleatória da conectividade sem fio de RSSFs impacta severamente a confiabilidade da comunicação e as limitadas capacidades de computação e de bateria dos nodos (WATTEYNE et al., 2010). Com o intuito de economizar recursos e aumentar o desempenho das redes, os protocolos de roteamento para RSSFs, em geral, aperfeiçoam o funcionamento de acordo com as características de tráfego de uma aplicação específica, seja na economia de energia ou no gerenciamento de congestionamento e/ou mobilidade. Deste modo, os algoritmos de roteamento para RSSFs não são apenas classificados de acordo com a forma de funcionamento na descoberta e manutenção de rotas. Nesta dissertação, os protocolos de roteamento para RSSFs são classificados de maneira distinta: *centrados em dados*, *hierárquicos*, *geográficos* e *fluxo de rede* (AKKAYA; YOUNIS, 2005).

2.4.1 Protocolos Centrados em Dados

Protocolos que utilizam o endereçamento como base no roteamento carecem de uma identificação global que somada à implantação aleatória dos nodos sensores dificultam a seleção de um grupo específico para solicitar informações (AKKAYA; YOUNIS, 2005). Geralmente tais grupos de nodos sensores são representados pelo tipo de informação coletada pelo sensor. Portanto, protocolos de roteamento centrados em dados que são capazes de selecionar um grupo de nodos durante as retransmissões visam reduzir a quantidade de informações redundantes transmitidas na rede, reduzindo o *overhead* de roteamento e economizando a energia consumida.

Sistema Adaptativo de Colônia de Formigas (do inglês *Adaptive Ant Colony System - AACS*) é um protocolo de roteamento centrado

em dados que utiliza *Difusão Direta* (do inglês *Direct Diffusion*) para distribuir mensagens de interesse e construir a *Árvore Mínima de Steiner* (do inglês *Minimum Steiner Tree - MST*) (LI; SHI, 2008). Como ilustrado na Figura 6, a estação base dissemina pacotes de interesse na RSSF. Como resultado, o AACS constrói a MST, uma árvore oti-

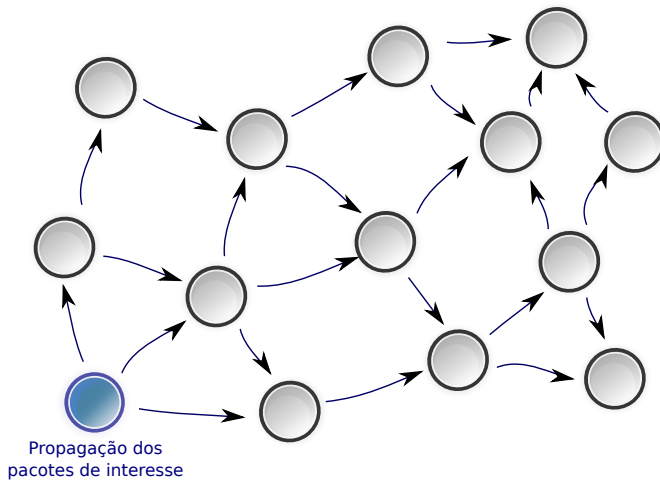


Figura 6: Roteamento centrado em dados (AACs + MST) - Propagação das mensagens de interesse pela rede.

mizada e formada pelos nodos da rede, tendo a estação base como a raiz da árvore, como ilustrado na Figura 7. O objetivo da MST é evitar inundação de pacotes na rede definindo rotas sem ciclos a fim de rotear os dados até a estação base. A agregação de dados baseada no AACs, juntamente com MST, tende a reduzir a quantidade de dados transmitidos na rede, implicando em economia de energia e prolongamento do tempo de vida da rede.

Rede de Dados Nomeados (do inglês *Named Data Networking - NDN*) é uma arquitetura de comunicação baseada em nomes (JACOBSON et al., 2009). Ao invés de endereçar os dados utilizando meios de identificação da origem e do destino, os dados são nomeados a fim de transformá-los em entidades reconhecidas pelos conteúdos e não pe-

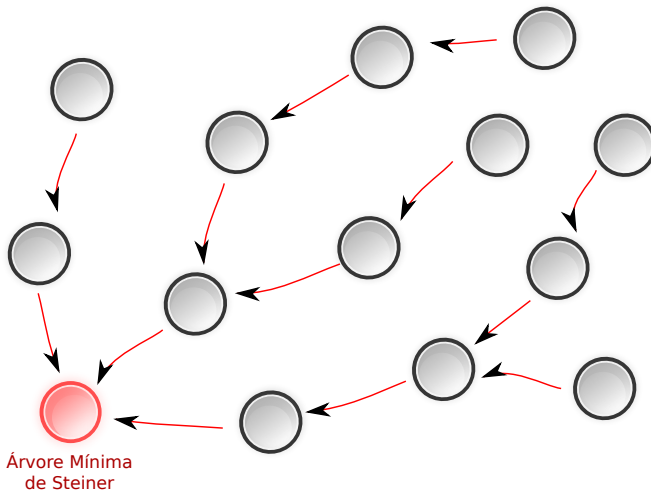


Figura 7: Roteamento centrado em dados (AACSS + MST) - Construção da MST.

las localizações. Deste modo, o NDN pode ser usado para equilibrar a demanda de tráfego entre os roteadores para que esta, através da rede, possa ser ajustada para satisfazer os requisitos de um tráfego consciente de energia. Como resultado, os dados não necessitam trafegar por toda a rede, pois a arquitetura NDN permite armazenar e recuperar o conteúdo nos roteadores a fim de facilitar a engenharia de tráfego consciente de energia (SONG; LIU; WANG, 2011).

2.4.2 Protocolos Hierárquicos

Protocolos de roteamento hierárquicos são soluções que visam o consumo eficiente de energia das RSSFs. Tais protocolos envolvem os nodos em uma comunicação de múltiplos saltos dentro de um particular grupo ou aglomerado (*cluster*) a fim de realizar agregação ou fusão de dados para reduzir o número de mensagens transmitidas para a estação base (AKKAYA; YOUNIS, 2005). Em protocolos hierárquicos

de roteamento para RSSFs, a topologia da rede é particionada logicamente em um número de domínios separados (geralmente geograficamente correlacionados), cada um executando a própria instância de sensoriamento, como exemplificado na Figura 8. Cada grupo mantém um líder responsável por coletar os dados do domínio e encaminhá-los à estação base. O protocolo de roteamento é utilizado para o roteamento entre domínios mantendo somente as informações sobre interconexões, possibilitando o roteamento de uma grande área topológica com uma pequena quantidade de informação mantida em cada roteador.

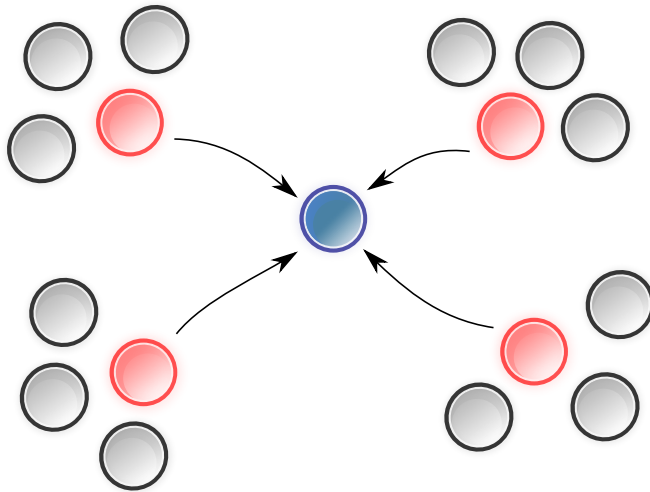


Figura 8: Roteamento hierárquico.

Agrupamento Hierárquico e Adaptativo de Baixa Energia (do inglês *Low-Energy Adaptive Clustering Hierarchy - LEACH*) é um protocolo hierárquico e baseado em grupos o qual visa minimizar o uso de energia global ao distribuir a carga de transmissões em diferentes pontos no decorrer do tempo (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2000). LEACH realiza computação local em cada grupo a fim de reduzir a quantidade de dados transmitidos para a estação base. O líder do grupo transmite as informações à estação base, através

de uma transmissão de único salto.

O protocolo *Roteamento Energeticamente Eficiente de Área Variável* (do inglês *Energy Efficient Variable Area Routing - EE-VAR*) é um algoritmo de formação de cluster que visa melhorar a eficiência energética (CHOI; MOH; CHUNG, 2008). Este estudo propõe uma abordagem diferente do LEACH, que representa um método básico de agrupamento e consome uma grande quantidade de energia na formação dos grupos. O método proposto minimiza o consumo de energia reduzindo o período de formação de grupos, que ocorrem frequentemente, melhorando a eficiência de comunicação em RSSFs comparado a métodos usuais de agrupamento, como o LEACH.

2.4.3 Protocolos Geográficos

Esquemas de agregação de dados (do inglês *data aggregation scheme*) geralmente necessitam levar em consideração o roteamento geográfico, pois os dados sensoriais em RSSFs são plausíveis de serem geograficamente correlacionados (GANESAN et al., 2004). Consequentemente, diversas soluções têm sido dirigidas a protocolos geográficos. Informações de posicionamento permitem calcular as distâncias entre nodos e deduzir o consumo de energia de cada transmissão possível. O protocolo *Roteamento Geográfico e Consciente de Energia* (do inglês *Geographic and Energy Aware Routing - GEAR*) utiliza heurísticas para seleção de nodos vizinhos para rotear pacotes de/para uma região alvo (YU; GOVINDAN; ESTRIN, 2001). Tal roteamento visa balancear o consumo de energia e, deste modo, aumentar o tempo de vida da rede. GEAR é baseado unicamente em informações geográficas e energéticas na construção de rotas, sendo capaz de encontrar rotas ótimas em termos de energia residual e distância entre os nodos.

Roteamento Geográfico sem Informação de Localização (do inglês *Geographic Routing without Location Information - GRLI*) é um protocolo de roteamento que utiliza de coordenadas virtuais para rotear os dados (RAO et al., 2003). Para GRLI, o conjunto de coordenadas reais dos nodos não é condição necessária para a utilização do protocolo de roteamento, apesar de este assumir o posicionamento de

tais nodos. O roteamento ocorre ao transmitir os pacotes sempre para o nodo mais próximo do destino estipulado. Como exemplificado na Figura 9, cada nodo transmissor analisa as opções de rotas usando as coordenadas virtuais a fim de alcançar um determinado nodo destino, escolhendo a opção que está mais próxima do destino.

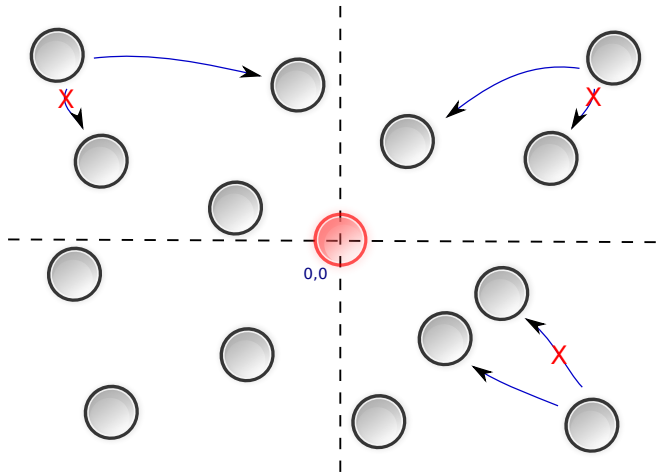


Figura 9: Roteamento geográfico (GRLI).

Protocolo de encaminhamento geográfico consciente das condições de rede para aplicações de tempo real em redes móveis de sensores sem fio (do inglês *netwoRk conditions Aware geographiCal forwarding protocol for rEal-time applications in mobile wireless sensor network - RACE*) é um protocolo de roteamento proposto para lidar com as questões de mobilidade e alta carga de mensagens (ARAÚJO; BECKER, 2011). O protocolo RACE foi projetado para suportar aplicações de nodos sensores móveis que se comuniquem com estações bases estacionárias. Além disso, o protocolo provê qualidade de serviço temporal para aplicações que necessitem de cumprimento de prazos na entrega de mensagens. Essencialmente, o protocolo RACE foi implementado sob a ótica de um algoritmo de roteamento geográfico, na qual os nodos são cientes de localizações geográficas, e que considera vários aspectos da rede para realizar as decisões de

roteamento.

2.4.4 Protocolos de Fluxo de Rede

Protocolos de fluxo de rede visam reduzir a redundância de dados ou distribuir o tráfego de dados a fim de reduzir e/ou balancear o consumo de energia da rede. Tais protocolos tendem a ser ótimas soluções para lidar com o congestionamento pelas capacidades de reduzir ou distribuir o tráfego de dados na rede.

Rotas de Energia Sob Demanda baseado em Formigas (do inglês *Ant-based On-demand Energy Route - AOER*) é um algoritmo de roteamento para redes IEEE 802.15.4 (SHUANG; LI; CHEN, 2009). Diferentemente de outros protocolos baseados em ACO, AOER requer menor quantidade de memória, menor processamento e utiliza estruturas simples de dados para as formigas e a tabela de roteamento. O algoritmo mantém as rotas inserindo feromônio de acordo com a quantidade de energia residual nos nodos a fim de equalizar o tráfego de dados na rede. AOER obtêm bons resultados em prolongar o tempo de vida da rede e balancear o consumo de energia entre os nodos.

Roteamento baseado em Formiga com Controle de Congestionamento (do inglês *Ant based Routing with Congestion Control - ARCC*) é um algoritmo baseado em ACO para RSSFs que foca no controle de congestionamento da rede (DHURANDHER et al., 2009). ARCC é eficiente em lidar com congestionamento e obtêm bons resultados em relação à taxa de transferência, perda de pacotes e desempenho. Tal algoritmo utiliza um mecanismo de detecção inteligente de congestionamento que mede o intervalo de tempo entre dois pacotes transmitidos sequencialmente. Com isso, ARCC tende a rotear os pacotes de modo que as conexões de congestionamento sejam evitadas, como ilustrado na Figura 10.

O Roteamento de Retorno de Informações para Otimização de Múltiplas Estações Bases (do inglês *Feedback Routing for Optimization Multiple Sinks - FROMS*) é um algoritmo de roteamento o qual suporta falhas de nodos e estações bases móveis (FÖRSTER;

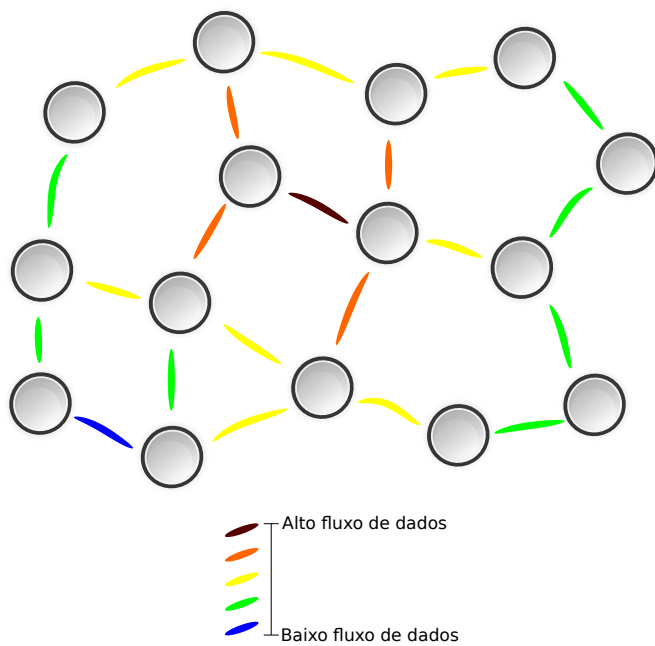


Figura 10: Roteamento de fluxo de rede (congestionamento - ARCC).

MURPHY, 2007). FROMS apresenta uma definição formal do problema de múltiplas estações bases como uma tarefa de aprendizagem por reforço. Tal algoritmo demonstra que o custo de roteamento da técnica de aprendizagem, combinado com o baixo *overhead* das informações trafegadas, diminui significativamente. Os autores também propuseram uma implementação real nos nodos sensores *ScatterWeb* (FÖRSTER et al., 2008). Eles demonstraram a viabilidade de implementação do protocolo baseado na máquina de aprendizagem em *hardware* real, bem como o ganho de desempenho em comparação a abordagens tradicionais. Eles também oferecem algumas observações gerais de propriedade e “ciladas” apresentadas em implementações reais de RSSFs, juntamente com potenciais soluções.

Vlajic e Stevanovic analisaram os prós e contras da implantação de estações bases móveis em redes IEEE 802.15.4 reais (VLAJIC; STEVANOVIC, 2009). Eles introduziram dois mecanismos simples para a redução do *overhead* relacionado à mobilidade em RSSFs. Eles também demonstraram analiticamente e através de simulações que em redes ideais, estações bases móveis podem resultar em melhor distribuição da carga de roteamento e um maior tempo de vida da rede. Infelizmente, em redes reais, incluindo IEEE 802.15.4, o *overhead* não é zero. Tais redes utilizam mecanismos que geram *overhead* adicional a fim de gerenciar congestionamento e mobilidade. Portanto, para contemplar o uso de RSSFs com mudanças de topologias contínuas, a redução do *overhead* de roteamento deve ser o primeiro curso de ação no projeto de protocolos de roteamento.

2.5 Comparação

Nodos sensores sem fio geralmente sofrem de limitações de recursos em termos de memória, energia e poder de processamento, as quais são particularmente acentuadas em RSSFs (WATTEYNE et al., 2010). Como dito anteriormente, RSSFs são consideradas uma subclasse de rede *ad hoc*, mas não fazem uso dos protocolos de gerenciamento de energia desenvolvidos para as MANETs, pois a natureza aleatória da conectividade sem fio de RSSFs impacta severa-

mente a confiabilidade da comunicação e as limitadas capacidades de computação e de bateria dos nodos. Com o intuito de economizar recursos e aumentar o desempenho das redes, os protocolos de roteamento para RSSFs, em geral, aperfeiçoam o funcionamento de acordo com as características de tráfego de uma aplicação específica, seja na economia de energia ou no gerenciamento de congestionamento e/ou mobilidade. Deste modo, protocolos para RSSFs tendem a ser classificados de acordo com tais aperfeiçoamentos, diferentemente dos protocolos para MANETs, como demonstrado na Tabela 1.

Tabela 1: Comparação dos protocolos *ad hoc*.

PROTOCOLOS	Pró-ativos	Reativos	Híbridos	Centrados em Dados	Hierárquicos	Geográficos	Fluxo de Redes
OLSR	X						
DSDV	X						
AODV		X					
DYMO		X					
ZRP			X				
HOPNET			X				
AACS				X			
NDN				X			
LEACH					X		
EE-VAR					X		
GRLI						X	
GEAR						X	
RACE						X	
AOER							X
ARCC							X
FROMS							X

Os protocolos pró-ativos, reativos e híbridos geralmente visam maneiras eficientes para lidar com a mobilidade ou falta de estrutura de uma rede *ad hoc*. No entanto, em redes mais propensas a erros que MANETs, como RSSFs, as transmissões de dados entre nodos são amplamente influenciadas pelo fato de que o meio comum de comunicação é tipicamente através de canais de transmissões sem fio com sérias limitações de largura de banda (WATTEYNE et al., 2010). Ou seja, maior probabilidade de dados corrompidos na camada física

e maior probabilidade de colisões na camada de controle de acesso ao meio, dificultando a aplicação de soluções para lidar com mobilidade e falta de estrutura das RSSFs. Protocolos pró-ativos, como OLSR e DSDV, são algoritmos eficientes em manter as tabelas de roteamento sempre atualizadas. Porém, tais protocolos são fortemente dependentes de um alto *overhead* de roteamento a fim de suportar as trocas de informações para a atualização das tabelas de roteamento, impactando negativamente a taxa de entrega de dados e o consumo de energia da rede. Em oposição aos protocolos pró-ativos, os protocolos reativos visam à redução da carga de pacotes na rede ao realizar procedimentos de descoberta de rotas sob demanda. Porém, o grande problema da maioria dos protocolos reativos, como AODV e DYMO, está na manutenção das rotas. Tais protocolos mantêm as rotas descobertas nas tabelas de roteamento por um tempo específico e ainda dispõem de mecanismos de reparação em casos de erros no roteamento. O mecanismo de reparo envolve notificações de erro para os nodos envolvidos com a rota e um novo procedimento de descoberta caso necessário, implicando em aumento do *overhead* de roteamento. No entanto, o *overhead* de roteamento tende a se intensificar em RSSFs móveis, cujos erros de roteamento são eventos frequentes. Em consequência disso, protocolos híbridos também tendem a sofrer com alto *overhead* em redes propensas a erros, no roteamento pró-ativo e reativo.

Protocolos de roteamento para RSSFs geralmente apresentam soluções eficientes para lidar com o consumo de energia e congestionamento. No entanto, tais protocolos não oferecem a **robustez** necessária para lidar com topologias dinâmicas. RSSFs são geralmente implantadas de modo estático ou com mobilidade “premeditada” (WANG; WU; TSENG, 2010). Porém, com o advento de aplicações que necessitam lidar com mobilidade “arbitrária” (FESTO, 2012), tal robustez torna-se uma necessidade para o roteamento em RSSFs. Protocolos centrados em dados, como AACS e NDN, focam nos dados ao invés do endereçamento para realizar o roteamento. No AACS, como resultado da propagação de pacotes de interesse, a rede cria uma árvore otimizada (MST) para realizar as transmissões dos dados (ver Figura 6). No entanto, em topologias dinâmicas, o algoritmo necessita reconstruir a árvore a cada mudança de topologia, causando *overhead*

adicional. Além disso, com a falta de endereçamento, algoritmos de roteamento, como o NDN, sempre recorrem a sucessivos *broadcasts* na RSSF para a transmissão e propagação da informação, principalmente em algoritmos que não utilizam recursos (MST) para limitar a inundação de pacotes na rede. Protocolos hierárquicos, como LEACH e EE-VAR, são algoritmos que focam na redução do consumo de energia da rede. Tais protocolos utilizam grupos ou aglomerados para realizar fusão ou agregação de dados a fim de reduzir a carga de dados transmitidos para a estação base, resultando na redução do tamanho das tabelas de roteamento e facilitando a gerência do tráfego. No entanto, protocolos hierárquicos não são soluções eficientes para redes móveis, pois a mobilidade dos nodos tende a dificultar a formação e a manutenção dos grupos, assim como a eleição dos líderes dos grupos. Protocolos geográficos, como GRLI, GEAR e RACE, são algoritmos que utilizam informações de posicionamento para auxiliar o roteamento e estimar o consumo de energia dos nodos. GRLI e GEAR são eficientes em determinar rotas ótimas geograficamente, mas carecem de mecanismos para lidar com congestionamento, especialmente em redes propensas a alta taxa de erros. Em contrapartida, o RACE é um protocolo proposto para lidar com mobilidade a uma alta carga de dados. Porém, os nodos precisam estar cientes dos posicionamentos, seja através de algoritmos de localização ou através do *Sistema de Posicionamento Global* (do inglês *Global Positioning System*). Dentre as classes de protocolos para RSSFs, os protocolos de fluxo de rede são os que apresentam soluções e técnicas para lidar com congestionamento. AOER e ARCC são protocolos baseados em ACO que visam, respectivamente, o balanceamento do consumo de energia dos nodos e o controle de fluxo de dados. O FROMS, diferentemente, utiliza uma técnica de aprendizagem por reforço para lidar com o congestionamento. Devido às técnicas de controle de fluxo de dados, tais protocolos tendem a ser ótimas opções para redes móveis. No entanto, antes de qualquer técnica de balanceamento do consumo de energia ou do tráfego de dados, a redução do *overhead* deve ser o primeiro curso de ação no desenvolvimento de um protocolo de roteamento para RSSFs móveis (VLAJIC; STEVANOVIC, 2009).

3 ALGORITMO PROPOSTO I: AD-ZRP

Protocolos de roteamento baseados em ACO geralmente dispõem da capacidade de aprender as rotas ótimas (LU; ZHAO; SU, 2004) e ainda se adaptar automaticamente às mudanças de topologia de rede através de heurísticas relacionadas ao feromônio (IYENGAR et al., 2007). RSSFs são frequentemente implantadas em ambientes hostis. Como resultado, danos e falhas em nodos sensores tornam-se acontecimentos comuns. Isso também contribui para as típicas mudanças de topologia devido à mobilidade dos nodos (GARCIA-MACIAS; GOMEZ, 2007).

Uma vez que RSSF são ruidosas, propensas a erros e tipicamente implantadas como redes *ad hoc*, muitos dos algoritmos de roteamento focam em obter rotas confiáveis e lidam com o *overhead* das mudanças de topologia e de mobilidade. Com tal objetivo, várias soluções monitoram a qualidade das transmissões de dados através de métricas de qualidade, tais como força do sinal, taxa de recepção de dados, localização e heurísticas (STANKOVIC, 2008). Além disso, algoritmos de roteamento inspirados pela inteligência coletiva das formigas possibilitam uma maneira eficaz de lidar com topologias dinâmicas devido à habilidade das formigas de perceber as mudanças nas redes através de feromônio.

HOPNET é um protocolo de roteamento híbrido para MANETs inspirado na meta-heurística ACO e baseado no protocolo ZRP (WANG et al., 2009). Este protocolo suporta um algoritmo de roteamento de múltiplos saltos e autoconfigurável o qual se baseia no roteamento por zonas e se inspira na inteligência coletiva das formigas a fim de obter rotas confiáveis entre nodos de redes *ad hoc*. Porém, tal algoritmo apresenta diversas limitações para a implantação em RSSFs, da estrutura das tabelas de roteamento aos procedimentos de descoberta e manutenção de rotas (OKAZAKI; FRÖHLICH, 2009). Tais limitações impulsionaram a proposta do *Protocolo para Roteamento em Zonas Dinâmicas baseado em Formiga* (do inglês *Ant-based Dynamic Zone Routing Protocol - AD-ZRP*), um algoritmo de roteamento reativo e autoconfigurável, que possui como base o algoritmo HOPNET (OKA-

ZAKI; FROHLICH, 2011a). Atuando em conjunto com ACO e com uma técnica denominada *zonas dinâmicas*, AD-ZRP permite melhorar os mecanismos de descoberta e manutenção de rotas, através de feromônio, visando à resolução de importantes problemas no roteamento em redes *ad hoc*. Tais contribuições permitem atingir um algoritmo de roteamento eficiente o suficiente para garantir rotas confiáveis entre nodos e lidar com congestionamento e habituais problemas de ambientes de redes de topologia dinâmica, como MANETs.

3.1 O Algoritmo HOPNET

Diferentemente do protocolo de roteamento híbrido denominado ZRP, o HOPNET envolve a inteligência coletiva de formigas na dinâmica do roteamento pró-ativo para manter e melhorar as rotas existentes ou explorar as melhores opções. ACO é uma técnica de inteligência coletiva a qual se baseia na utilização do comportamento coletivo das formigas para resolver problemas computacionais (DORIGO; BLUM, 2005; DORIGO; BIRATTARI; STUTZLE, 2006).

Protocolos híbridos geralmente visam um esquema de roteamento que une os benefícios dos protocolos reativos e pró-ativos. ZRP e HOPNET são algoritmos cujos nodos mantêm zonas definidas por um raio, número de saltos do nodo origem à borda da zona, como demonstrado pela Figura 11. Deste modo, cada nodo não necessita manter as rotas para toda a rede de forma pró-ativa, apenas para a própria zona. Por outro lado, todo roteamento para nodos fora da zona é realizado de forma reativa.

A ideia por trás dos protocolos de roteamento baseados em ACO é aplicá-los nos procedimentos de descoberta e manutenção das rotas através de um agente, a formiga. Deste modo, tais protocolos podem manter as tabelas de roteamento eficientemente atualizadas devido ao dinamismo proporcionado pelas formigas para detectar, por feromônio, mudanças na topologia da rede.

No HOPNET, as formigas são classificadas em 5 categorias: *Formiga de Exploração Interna* (FEI), *Formiga de Exploração Externa*

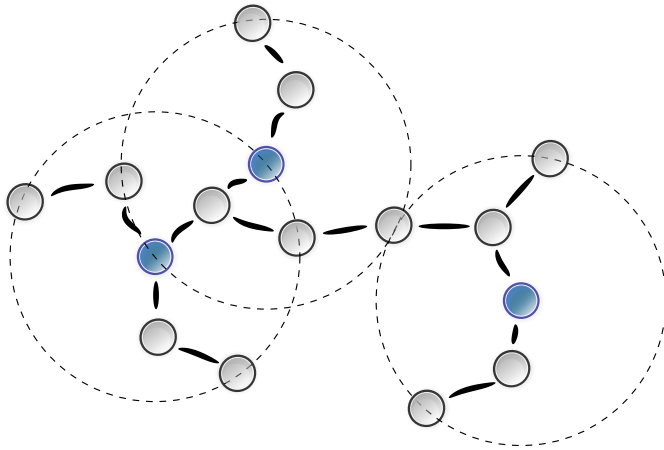


Figura 11: Protocolos híbridos: zonas - raio igual a 2.

(FEE), *Formiga de Retorno* (FR), *Formiga de Notificação* (FN), e *Formiga de Erro* (FE). Apesar de cada categoria de formiga ter uma função diferente, elas compartilham a mesma estrutura de dados. A Figura 12 apresenta a estrutura de dados das formigas do HOPNET. Esta estrutura inclui os campos de endereçamento, tais como *Origem* e *Destino*. O campo *Ordem* é utilizado para controle. O campo *Tipo* indica a categoria da formiga, e o campo *Saltos* indica o número de saltos que a formiga realizou. O campo *Caminho* indica a sequência de endereços dos nós que uma formiga percorre.

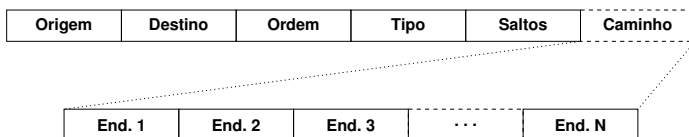


Figura 12: HOPNET - Estrutura da formiga.

As FEIs são periodicamente enviadas dentro dos limites da zona a fim de manter as melhores rotas através de feromônio. FEIs tem o objetivo de disseminar feromônio dentro da zona a fim manter o ro-

teamento híbrido enquanto o mesmo se esgota devido à evaporação, como o diagrama de sequência apresentado na Figura 13. Entretanto, cada nodo na rede espera por pacotes que serão tratados em conformidade com os diagramas de sequência a seguir.

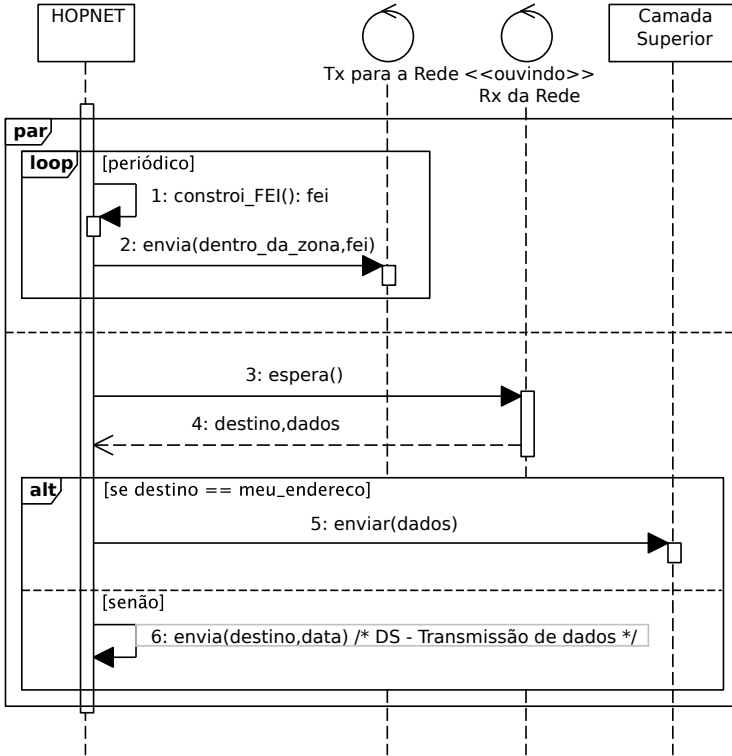


Figura 13: HOPNET - FEIs periódicas e recepção de pacotes.

Em cada nodo ao longo de uma rota, da origem até a fronteira da zona, a FEI deve verificar e atualizar as informações de feromônio na tabela de roteamento interna, como apresentado no diagrama da Figura 14. Assim, de acordo com a quantidade de feromônio na tabela interna, cada formiga pode detectar o melhor nodo vizinho para atingir um determinado nodo destino dentro da zona, aumentando assim,

a quantidade de feromônio sobre esta rota para auxiliar o tráfego de pacotes de dados na rede. Todas as formigas possuem a função de depositar feromônio e mapear a distribuição nesta tabela a fim de definir as rotas, mas a FEI é a responsável por mantê-la de forma pró-ativa dentro das zonas. A tabela interna é definida como uma matriz cujas linhas representam os nodos vizinhos e as colunas representam todos os nodos identificados na zona.

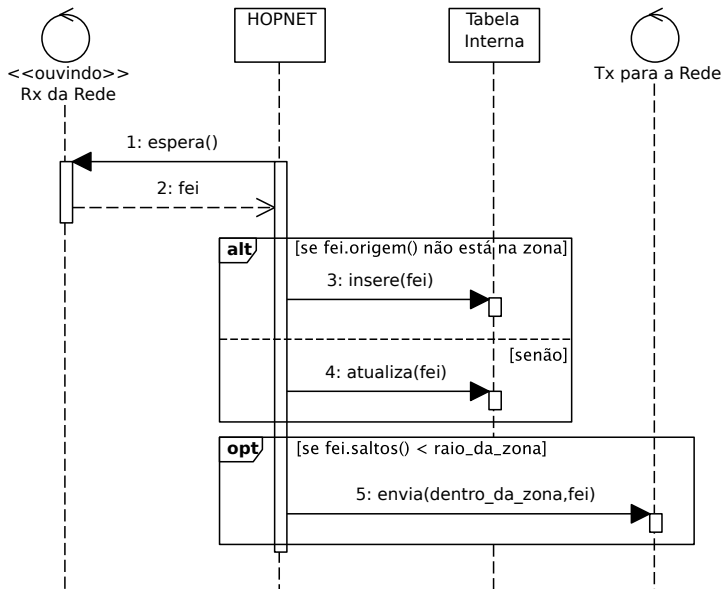


Figura 14: HOPNET - Recepção de FEIs.

A Figura 15 apresenta o diagrama de seqüência de transmissão de dados. Quando um nodo (origem) deseja enviar um pacote de dados, a tabela interna é verificada para descobrir se o nodo destino está dentro da zona. Caso o nodo destino esteja, o nodo origem define um nodo vizinho para transmitir os dados de acordo com a quantidade de feromônio. Caso contrário, o nodo origem precisa verificar a tabela externa de roteamento do HOPNET, uma tabela responsável por armazenar o campo *Caminho* descoberto pelas formigas (FEEs e FRs)

de forma reativa, desde a origem até o destino, quando o destino está fora da zona ou é um nodo desconhecido pela origem. As tabelas externas armazenam as rotas previamente descobertas pelas formigas, e cada rota permanece nela por um tempo determinado.

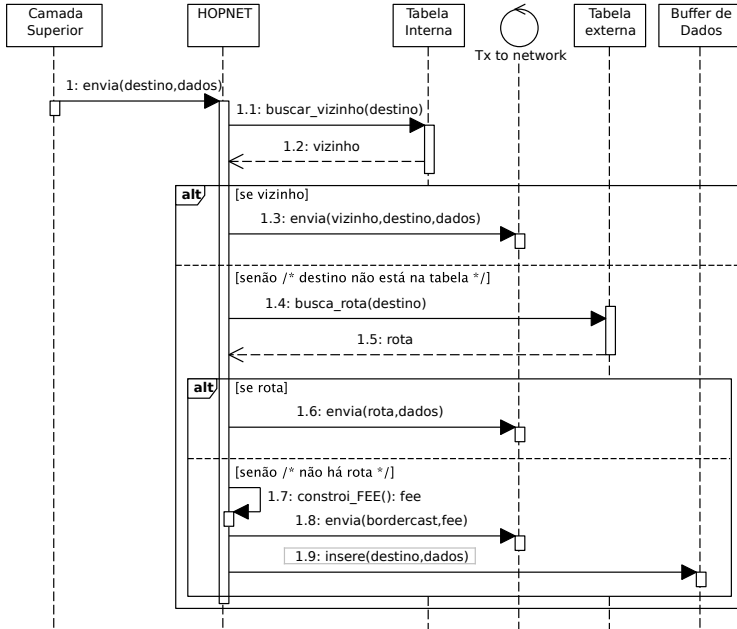


Figura 15: HOPNET - Transmissão de Dados.

No entanto, caso não exista uma rota para o destino na tabela externa, o pacote de dados é armazenado num *buffer*, e então o nodo recorre ao roteamento reativo e inicia um processo de busca para definir um novo caminho para o nodo destino, como apresentado nos diagramas das Figuras 15 e 16. Este processo de busca envolve FEEs e FRs, e consiste em transmitir os FEEs entre as fronteiras das zonas para encontrar a zona que contém o destino. Essa propagação de formigas através das fronteiras das zonas é chamada *bordercast* (WANG et al., 2009). Enquanto isso, o pacote de dados permanece no *buffer* até o retorno de uma FR do destino.

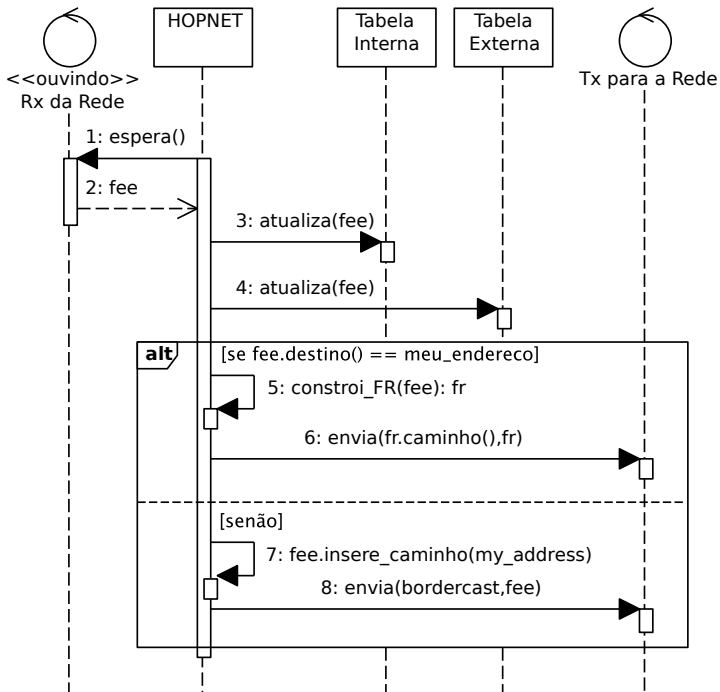


Figura 16: HOPNET - Recepção de FEEs.

Quando uma FEE é recebida no destino, ela é convertida para uma FR e enviada de volta para o nodo origem percorrendo a rota descoberta. Ao receber o FR, o nodo origem será capaz de enviar os pacotes de dados armazenados na *buffer* para os respectivos e conhecidos destinos, como apresentado nos diagramas das Figuras 16 e 17. Depois disso, o nodo de origem irá armazenar a rota descoberta na tabela externa.

Já as FNs são usadas para alertar a rede, durante a transmissão de dados, sobre qualquer mudança na topologia. Elas são enviadas para o nodo origem e para todos os demais nodos ao longo da rota para substituir as rotas inválidas nas tabelas externas. Em contraste, as FEs

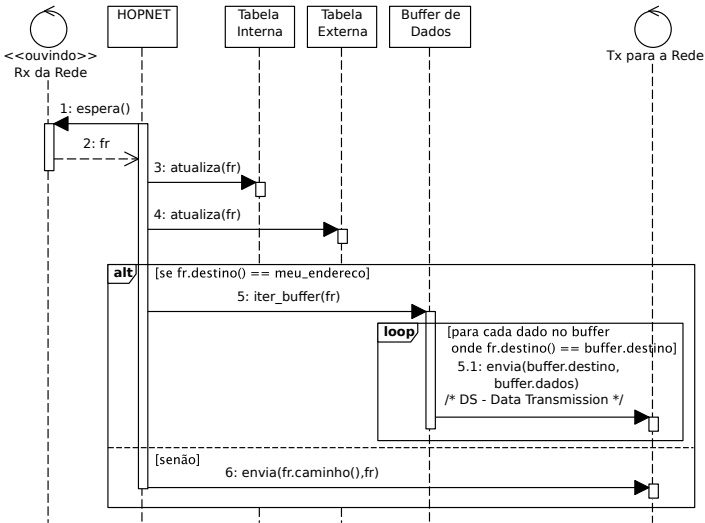


Figura 17: HOPNET - Recepção de FRs.

são usadas para reparar quebras de comunicação entre nodos. Elas são enviadas para o nodo origem a fim de iniciar uma nova pesquisa para o destino caso a origem ainda precise de uma rota para a transmissão de dados.

O gerenciamento de feromônio do HOPNET consiste na utilização de duas equações:

- *Equação de atualização de feromônio;*
- *Equação de evaporação.*

A equação de atualização de feromônio funciona da seguinte maneira: cada formiga, baseado na concentração de feromônio nas trilhas, seleciona o nodo v_j para o próximo salto a partir do nodo atual v_i . Na verdade, v_j é o nodo vizinho que possui a maior concentração de feromônio. A formiga seleciona o nodo v_j como o próximo salto

a partir do nodo atual v_i . No nodo v_j , a formiga atualiza o feromônio $\tau_{i,s}$ na entrada (v_i, v_s) na tabela interna, onde v_s é o nodo origem, como segue (WANG et al., 2009):

$$\tau_{i,s} = \tau_{i,s} + \Delta\tau, \quad (3.1)$$

onde $\Delta\tau$ representa a quantidade de feromônio considerando o tempo necessário para atravessar de v_s para v_j .

A evaporação ocorre no vínculo entre um nodo vizinho e o nodo origem, através da seguinte equação (WANG et al., 2009):

$$\tau_{l,s} = (1 - \rho) \cdot \tau_{l,s}, \quad \forall l \in N, \quad (3.2)$$

onde $\rho \in (0, 1]$ é a taxa de evaporação, e N é o conjunto de nodos vizinhos que têm entradas para o nodo v_s na tabela interna.

3.2 Do HOPNET ao AD-ZRP

AD-ZRP é baseado no algoritmo HOPNET (WANG et al., 2009), eficiente protocolo para MANETs que visa alta taxa de entrega de dados em redes escaláveis e de alta mobilidade. Com a robustez do HOPNET para MANETs e redes escaláveis, o AD-ZRP visa lidar com redes de topologia dinâmica com o objetivo de evitar falhas de comunicação, congestionamento e melhorar a descoberta e manutenção de rotas.

Diferentemente do HOPNET, que utiliza zonas de tamanho fixo definido pelo raio, AD-ZRP usa zonas dinâmicas que variam em tamanho e forma de acordo com as transmissões sob demanda a fim de reduzir o *overhead* da rede de forma mais eficiente. Uma zona dinâmica é uma região definida apenas pela presença de feromônio nas rotas entre os nodos da zona e o nodo de origem. Deste modo, AD-ZRP é um protocolo de roteamento reativo, pois se a presença de feromônio define uma zona dinâmica, então as formigas não precisam trafegar pela rede o tempo todo. O envio periódico de formigas em todas as zonas gera um *overhead* adicional e desnecessário, principalmente em

RSSFs. Portanto, o objetivo do AD-ZRP é minimizar o *overhead* da rede enquanto tenta-se reduzir a latência através do efeito destas zonas dinâmicas, como na maioria dos protocolos híbridos de roteamento.

Uma vez que as zonas dinâmicas variam em tamanho de acordo com as transmissões sob demanda, elas podem mudar a forma e dimensão quase que constantemente. Inicialmente, todas as zonas estão vazias, isto é, os nodos iniciam sem nenhuma informação de rotas. Após cada transmissão de pacote de dados para um destino desconhecido, uma nova rota é adicionada à zona, além das informações dos vizinhos, que são agregadas automaticamente à tabela de roteamento. Da mesma forma, algumas rotas podem, em algum momento, sair da zona devido à evaporação, se tais rotas não são mais utilizadas. A Figura 18 demonstra o comportamento de uma zona dinâmica quando uma nova rota é descoberta e introduzida como parte da zona, e quando outras rotas podem deixá-la devido à evaporação de feromônio. Dessa

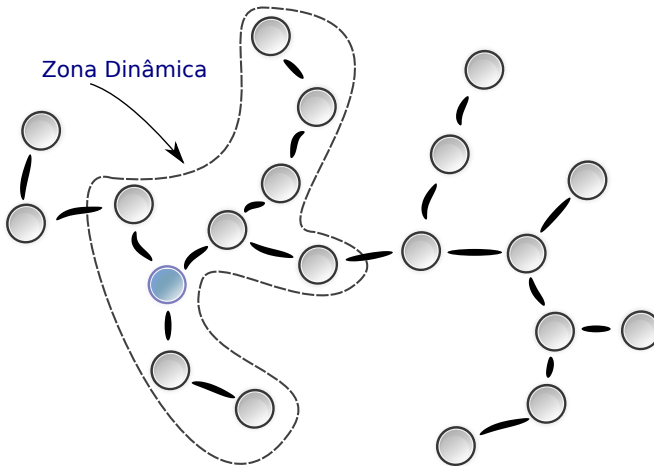


Figura 18: AD-ZRP - Zonas dinâmicas.

forma, novas rotas podem ser adicionadas à zona, enquanto outras rotas podem deixá-la, como demonstradas na Figura 19.

Enquanto HOPNET utiliza duas tabelas de roteamento para reali-

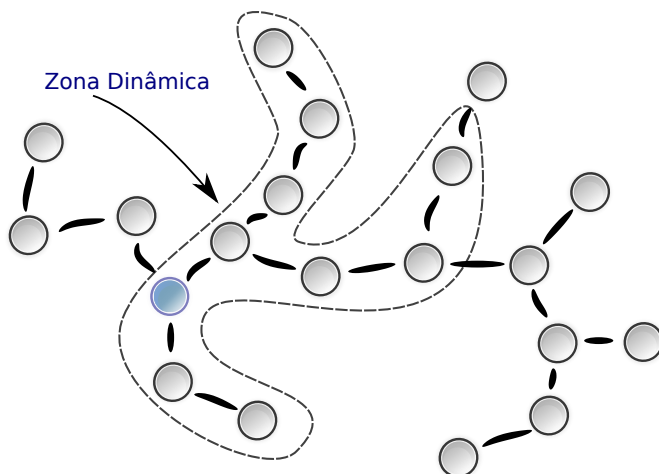


Figura 19: AD-ZRP - Mudança de forma da zona.

zar roteamento interno e externo separadamente, AD-ZRP utiliza uma única estrutura de tabela de roteamento. A técnica de zonas dinâmicas pode ser facilmente realizada por esta tabela semelhante à tabela interna do HOPNET. Diferentemente da tabela externa, as operações da tabela interna são mais simples e rápidas, e as rotas não precisam ser armazenadas inteiramente.

Desde que a tabela interna é a única tabela de roteamento do AD-ZRP, as operações de roteamento são realizadas somente por meio de feromônio. Para realizar tais operações de roteamento de forma eficiente, uma nova coleção de formigas é apresentada: *Formiga Transportadora Interna* (FTI) e *Formiga Transportadora Exploratória* (FTE). Ambas as formigas compartilham uma estrutura de dados comuns, como apresentada na Figura 20.

Comparando FTI e FTE com as formigas do HOPNET (Figura 12), observa-se que as formigas do AD-ZRP perdem o campo *Caminho*. Esse campo era responsável por armazenar a sequência de nodos descoberta pelas FEEs e FRs para a tabela externa, e responsável por

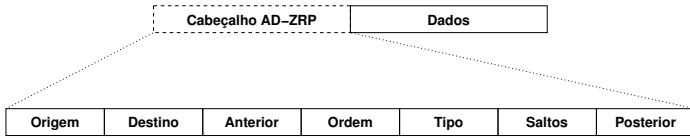


Figura 20: AD-ZRP - Estruturas da Formiga.

um *overhead* desnecessário devido ao acúmulo de dados sendo transmitidos a cada salto na rede. Por outro lado, dois outros campos foram adicionados a fim de ajudar na distribuição de feromônio e nos movimentos das formigas entre nodos. O campo *Anterior*, o qual é responsável por armazenar o endereço do nodo anterior, e o campo *Posterior*, o qual é responsável por armazenar o endereço do próximo nodo que receberá a formiga, a partir do nodo atual. Tais campos também reduzem a complexidade por oferecer melhores táticas para difundir e verificar feromônio entre os nodos, e reforçar os vínculos de feromônio entre nodos vizinhos a fim de manter as melhores rotas na zona. Além disso, tanto FTI quanto FTE realizam entrega de dados enquanto realizam o depósito de feromônio sobre as rotas que trafegam.

No HOPNET, se houver qualquer alteração nas rotas durante a transmissão de dados, FNs e FEs são enviadas para avisar outros nodos e iniciar a busca de uma nova rota, respectivamente. No AD-ZRP, tais procedimentos de reparo não ocorrem, pois as mudanças de topologia são notadas pelas zonas dinâmicas, e através delas, os nodos são capazes de realizar as transmissões dos dados de forma simplificada. A Figura 21 apresenta o diagrama de sequência da transmissão de dados no AD-ZRP. Diferentemente do HOPNET (Figura 15), o pacote de dados é enviado juntamente com a formiga a fim de garantir que mudanças bruscas na rede não interfiram com o transporte dos dados para o destino.

FTEs, diferente das FEEs (Figura 16), são responsáveis pela entrega de dados ao destino e pela descoberta de rotas para nodos desconhecidos. Estas formigas viajam através da rede para descobrir o nodo destino. No destino, a FTE apresenta o pacote de dados e retorna para

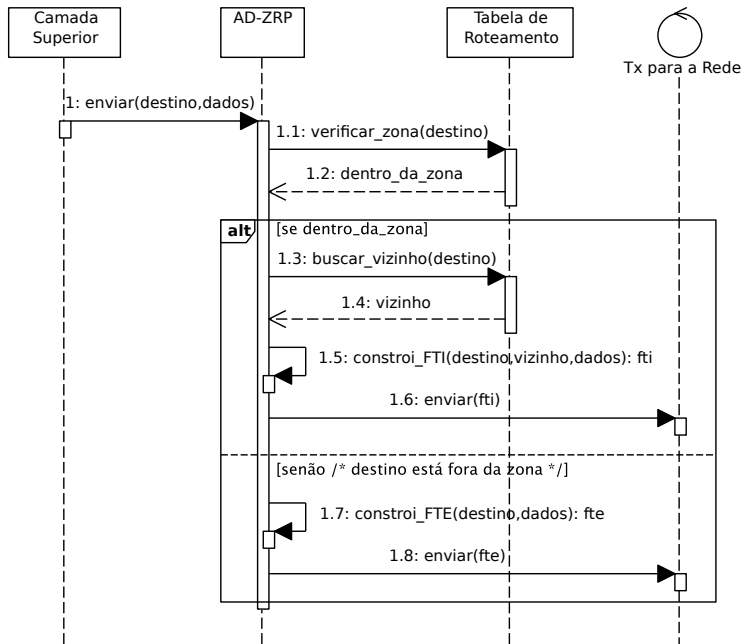


Figura 21: AD-ZRP - Transmissão de Dados.

o nodo origem. No caminho de volta, a formiga define a trilha de feromônio para as próximas transmissões, adicionando-a a zona, como apresentado na Figura 18 e no diagrama de sequência da Figura 22.

FTI funciona de forma semelhante à FEI (Figura 14), pois ambas as formigas são transmitidas apenas dentro da própria zona, no entanto, as FTIs também são responsáveis pela entrega dos pacotes de dados. Quando um nodo origem descobre uma nova rota para determinado destino através da FTE, as seguintes transmissões de pacotes de dados serão realizadas pelas FTIs até que a quantidade de feromônio na rota evapore completamente. No entanto, a qualquer momento, se uma rota qualquer para um determinado destino “quebrar”, qualquer nodo na rota pode usar uma FTE para recuperá-la ou descobrir um novo caminho, como no diagrama de sequência apresentado na Figura

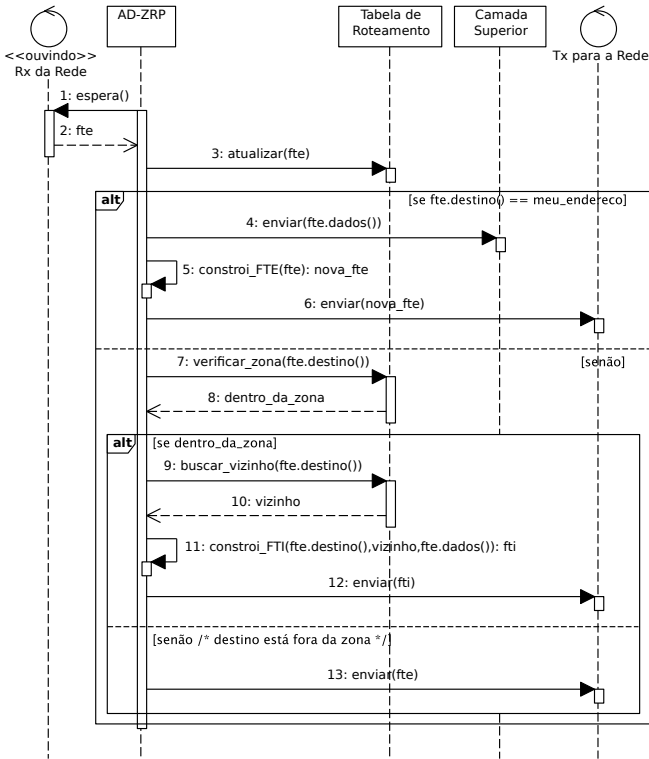


Figura 22: AD-ZRP - Recepção de FTEs.

23. Tal abordagem permite lidar com topologias de rede dinâmica e evitar, tanto quanto o possível, as quebras de comunicação entre nós.

Diferentemente do HOPNET, o AD-ZRP utiliza diferentes equações para o depósito e evaporação de feromônio. Cada formiga seleciona um nó v_j como o próximo salto a partir do nó atual v_i . Na verdade, v_j é o nó vizinho que possui a maior concentração de feromônio. No nó v_j , a formiga atualiza o feromônio $\tau_{i,s}$ na entrada (v_i, v_s) da tabela de roteamento interna, onde v_s é o nó origem,

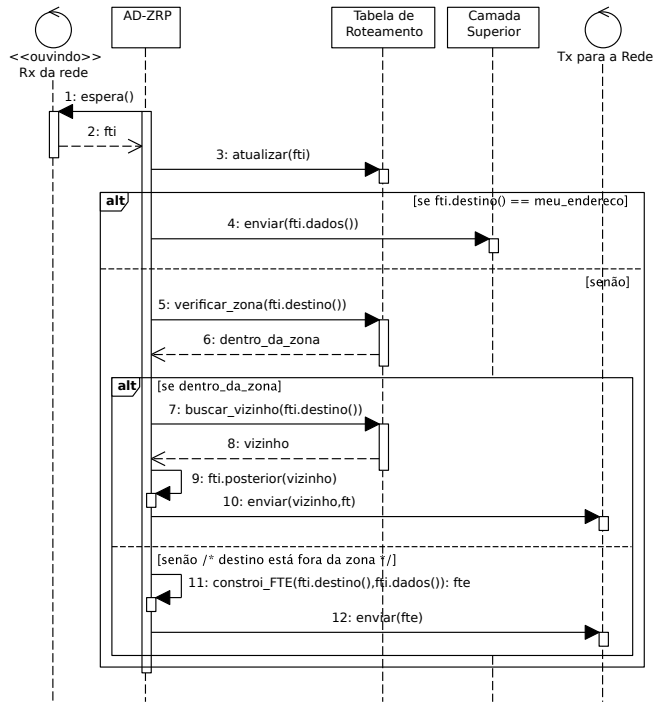


Figura 23: AD-ZRP - Recepção de FTIs.

como segue (DORIGO; BIRATTARI; STUTZLE, 2006):

$$\tau_{i,s} = (1 - \varphi) \cdot \tau_{i,s} + \varphi \cdot \tau_0 \quad (3.3)$$

onde $\varphi \in (0, 1]$ é o coeficiente de declínio do feromônio, e τ_0 é o valor inicial da quantidade de feromônio. Diferente da equação (3.1), a equação (3.3) permite diversificar o processo de pesquisa, diminuindo assim, a quantidade de feromônio nas rotas e permitindo que outras formigas descubram outras diferentes rotas, além de aumentar o efeito das zonas dinâmicas.

Diferentemente da equação (3.2), a evaporação ocorre periodicamente

mente em todos os nodos da rede através da seguinte equação (DORIGO; BIRATTARI; STUTZLE, 2006):

$$\tau_{i,j} = (1 - \rho) \cdot \tau_{i,j}, \quad \forall i \in N, \quad \forall j \in Z \quad (3.4)$$

onde $\rho \in (0, 1]$ é a taxa de evaporação, N é o conjunto de nodos vizinhos, e Z é o conjunto de nodos que, juntamente com os nodos vizinhos, definem as entradas (v_i, v_j) em tabela interna.

Enquanto as zonas do HOPNET são limitadas pelo raio, AD-ZRP possui zonas que frequentemente mudam ao longo do tempo com a finalidade de evitar a perda de importantes rotas devido às falhas ou mobilidade dos nodos. No entanto, por causa das zonas dinâmicas, há a possibilidade de que elas possam, ocasionalmente, incluir toda a rede. O AD-ZRP pode lidar com tais situações sem qualquer *overhead* adicional, pois cada nodo apenas precisa de informações de feromônio sobre os vizinhos para atingir um destino específico. Além de que, manter toda uma rede de alta escalabilidade em uma única zona dinâmica chega a ser um desafio, mesmo que por um breve momento, pois se trata de um protocolo de roteamento reativo e rotas, definidas por feromônio, tendem a se desfazerem devido à evaporação do feromônio. Deste modo, as equações (3.3) e (3.4) também ajudam a evitar tais situações através dos procedimentos de evaporação e controle de disseminação de feromônio.

AD-ZRP também visa abstrair problemas na rede, lidando com as mudanças de topologia através de feromônio a fim de minimizar o *overhead*, bem como possíveis congestionamentos. Congestionamento geralmente provoca perda de pacotes, atrasos de fila, e conseqüentemente, menor depósito de feromônio para as rotas congestionadas. Uma vez que a estrutura de tabela de roteamento mantém informações de feromônio entre nodos, ela não precisa ser atualizada a cada mudança na topologia da rede, tal como ocorre no HOPNET. As mudanças na topologia são observadas involuntariamente por formigas, e mensagens de advertência ou pacotes de controle, como FN e FEs, são desnecessários. Tal abordagem permite lidar com o congestionamento da maneira que operações complexas na rede possam ser evitadas.

3.3 Avaliação: HOPNET x AD-ZRP

As implementações foram baseadas no *Simulador de Sistemas de Informação Móvel Global* (do inglês *Global Mobile Information System Simulator* - GloMoSim). GloMoSim é um simulador de protocolos para sistemas de redes *ad hoc* sem fio. O HOPNET foi avaliado utilizando tal simulador (WANG et al., 2009), logo, para efeito de comparação o AD-ZRP também foi implementado para o GloMoSim.

A avaliação ocorre considerando diversos cenários de simulações sob a rede IEEE 802.11. A Tabela 2 apresenta os parâmetros de simulação utilizados no GloMoSim. Cada cenário de simulação é executado por um total de 900 segundos num ambiente propenso a alta taxa de perda de dados. Os nodos são implantados aleatoriamente em uma área retangular de 700x400 metros quadrados. Cada nodo move-se a uma velocidade máxima de 10 metros por segundo, de acordo com o algoritmo de mobilidade chamado *Coordenadas Intermediárias Aleatórias* (do inglês *Random Waypoint* - RWP). O tráfego de dados é gerado por 20 nodos móveis para outros 20 nodos móveis a uma taxa constante de *bits*. O poder de transmissão do rádio é estabelecido com 15dBm com uma taxa de *bits* de 2 megabits por segundo.

Este cenário base é o utilizado nos experimentos com o GloMoSim. Os cenários de testes são obtidos ao variar parâmetros específicos no cenário base, tal como o número de nodos e raio da zona. O número de nodos varia de 20 a 200, enquanto o raio de uma zona varia de 2 a 5 saltos.

A Figura 24 apresenta a taxa de entrega dos pacotes de dados neste ambiente de simulação. Conforme o número de nodos aumenta, a taxa de entrega tende a aumentar devido às formigas, que são capazes de direcionar o fluxo de dados pelas melhores trilhas de feromônio até determinado destino. Pela figura, nota-se que se a rede é muito grande e densa, então a taxa de entrega tende a ser alta devido à grande quantidade de opções que um grande número de nodos proporciona. Por outro lado, se a rede é pequena e esparsa, então a taxa de entrega tende a ser menor devido à falta de conectividade física entre os nodos. Nota-se também que o AD-ZRP retrata melhores resultados

Tabela 2: Configuração do GloMoSim.

Parâmetros	
Tempo de simulação	900s
Quantidade de nodos	20 ~ 200
Área	700mX400m
Algoritmo de mobilidade	RWP
Velocidade dos nodos	1 ~ 10m/s
Tamanho dos dados da aplicação	64B
Frequência de envio dos dados	3s
Protocolo de transporte	UDP
Protocolo de rede	IP
Máscara da rede	255.255.0.0
Frequência de canal	2.4GHz
Taxa de <i>bits</i>	2Mbit/s
Poder de transmissão do rádio	15dBm
Sensitividade do rádio	-71.067dBm

de taxa de entrega em redes de alta escalabilidade e alta mobilidade. Percebem-se nos resultados algumas irregularidades, principalmente nas simulações de 100 nodos do HOPNET, que são atribuídas a eventuais congestionamentos resultantes da mobilidade e da disposição dos nodos em determinados momentos da simulação, gerando baixa taxa entrega. Considera-se que a perda de pacotes na rede ocorre de duas maneiras:

- através de quebra de rotas, referentes aos problemas de encaminhamento de pacotes recebidos por rotas que não existem mais;
- através de falhas de conectividade (*links*) entre os nodos, referentes aos problemas que se originam na camada de *Controle de Acesso ao Meio* (do inglês *Medium Access Control* - MAC).

No primeiro caso, o protocolo de roteamento identifica e resolve os erros de roteamento. No segundo, os erros são identificados pela camada MAC e notificados ao protocolo de roteamento. A Figura 25 retrata os resultados referentes à taxa de perda de quadros em razão da

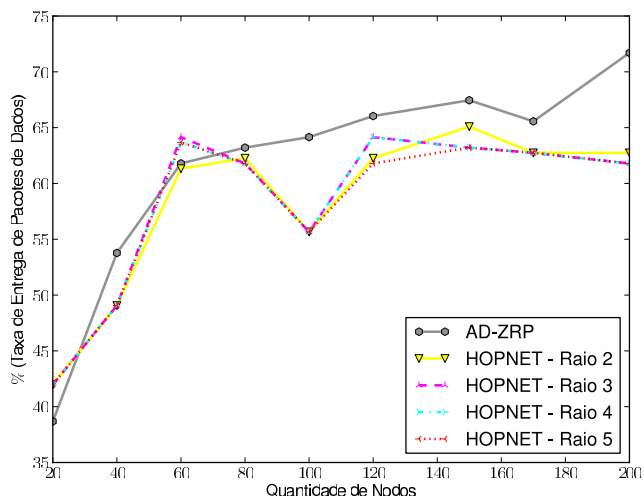


Figura 24: Taxa de entrega de dados.

taxa de falha de *links* do HOPNET a fim de justificar as irregularidades da Figura 24. Diferentemente da taxa de quebra de rotas, falhas de *links* se referem às mensagens de erros originadas na camada MAC. Nestes experimentos, caso um nodo não receba, na camada MAC, uma mensagem *Pronto para Enviar* (do inglês *Clear to send* - CTS) ou *Confirmação de recebimento* (do inglês *Acknowledgment* - ACK) após diversas tentativas, uma mensagem de falha de *links* é enviada ao protocolo de roteamento para conduzir possíveis procedimentos de reparo e/ou o uso de uma nova rota. Nota-se pela figura que a taxa de falha de *links* tende a ser superior em redes esparsas devido à baixa conectividade da rede. Devido à taxa de dados constante em todas as simulações, a taxa de falha de *links* tende a crescer levemente conforme a rede torna-se mais densa.

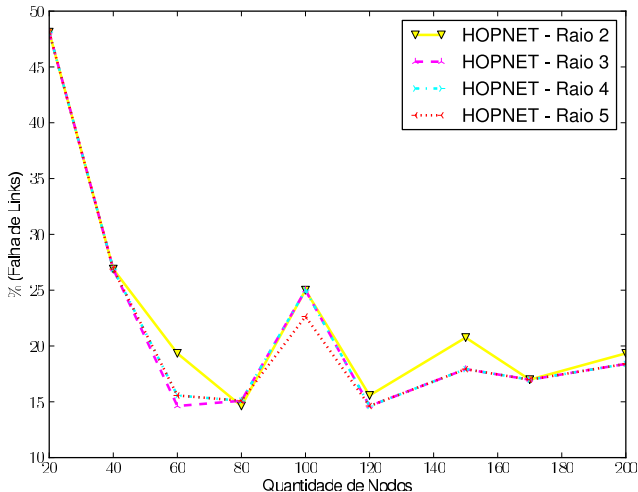
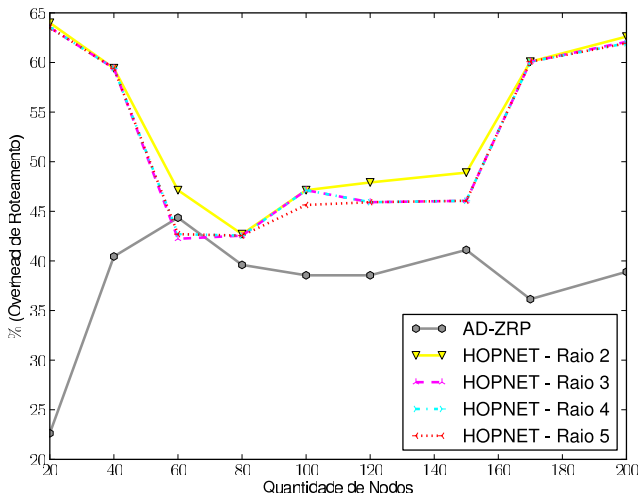


Figura 25: Taxa de falha de *links*.

A Figura 26 retrata a comparação dos protocolos de roteamento em razão do *overhead* de pacotes de controle na rede. No HOPNET, as formigas (pacotes de controle) são periodicamente difundidas pela

zona a fim de manter as rotas de forma pró-ativa, enquanto outras formigas são difundidas pela rede a fim de satisfazer o roteamento reativo do protocolo híbrido, além de auxiliar procedimentos de reparos de rotas. No AD-ZRP, os dados são transportados pelas formigas a fim de reduzir os pacotes de controle na rede. Além disso, não há procedimentos de reparo de rotas, pois as formigas se baseiam na distribuição de feromônio para formar o conhecimento da rede através de zonas dinâmicas. Em consequência disso, nota-se pela figura que o AD-ZRP obtém resultados melhores que o HOPNET em redes esparsas devido à parte do roteamento pró-ativo do HOPNET que tenta enviar as formigas mesmo sem conectividade. Nota-se também que o AD-ZRP tende a produzir resultados substancialmente melhores em redes escaláveis e densas. O HOPNET tende a aumentar o *overhead* conforme a rede torna-se mais densa devido à “soma” do *overhead* do roteamento pró-ativo e reativo em redes densas e móveis.

Figura 26: *Overhead* de roteamento.

3.4 Discussão

AD-ZRP é baseado no algoritmo HOPNET, um eficiente protocolo para MANETs que visa alta taxa de entrega de dados em redes escaláveis e de alta mobilidade. Mas HOPNET também possui características e funcionalidades complexas que dificultam o processo de descoberta de rotas, especialmente em redes suscetíveis a falhas e propensas a erros, tal como RSSFs. AD-ZRP visa lidar com as seguintes características através das zonas dinâmicas:

- *Complexa estrutura de tabelas de roteamento:* o HOPNET utiliza duas tabelas, uma para o roteamento interno e outra para o roteamento externo, implicando em complexas operações e desperdício de memória. A complexidade das operações pode ser notada nos diagramas de sequência apresentados na seção 3.1, e comparados aos diagramas de sequência do AD-ZRP na seção 3.2. Quanto ao desperdício de memória, nas implementações realizadas, as estruturas do HOPNET ocupam 2,03 vezes mais espaço do que as estruturas do AD-ZRP a princípio, pois o desperdício tende a aumentar conforme as transmissões sob demanda ocorrem, ao povoar a tabela externa de roteamento.
- *Complexas operações ao lidar com roteamento interno e externo de uma zona:* assim como as tabelas, o roteamento também é dividido em duas partes, sendo que apenas o roteamento interno tenta lidar com a distribuição eficiente do feromônio, mas ainda de forma pró-ativa. Além disso, o HOPNET utiliza procedimentos de reparo na rede através de FNs e FEs enquanto o AD-ZRP usa zonas dinâmicas para se autoadaptar às mudanças da rede. De forma contrária ao HOPNET, o AD-ZRP visa manter um roteamento de distribuição eficiente de feromônio de forma reativa, visando um baixo *overhead* na rede, como pode ser observado na Figura 26.
- *Roteamento reativo ineficiente/complexo e custoso para os padrões de RSSFs:* no roteamento reativo, muitas informações são transmitidas de forma ineficiente desperdiçando largura de

banda e energia. Considerando que a maioria das transmissões é destinada a nodos que estão dentro da zona do nodo origem, a rota é obtida rapidamente pelo roteamento pró-ativo. No entanto, se a maioria das transmissões é para nodos que estão fora da zona, então o roteamento tende a ser altamente custoso. O roteamento reativo do HOPNET consiste em usar formigas para construir rotas (sequências de endereços) e armazená-las na tabela externa de roteamento por um determinado tempo. Neste caso, a formiga precisa lembrar/gravar toda a rota estabelecida e retransmiti-la ao nodo origem, como ilustrado na Figura 27. Mas o maior problema consiste na identificação do destino. Se um nodo externo é armazenado como parte de uma sequência de endereços na tabela de roteamento externo, mas não como destino, então cada rota na tabela externa deve ser verificada minuciosamente a fim de definir o nodo externo como destino e obter uma rota para tal. No pior caso, o nodo origem deve iniciar um novo processo de descoberta de rota. De qualquer modo, ambos os procedimentos desperdiçam memória, processamento e largura de banda.

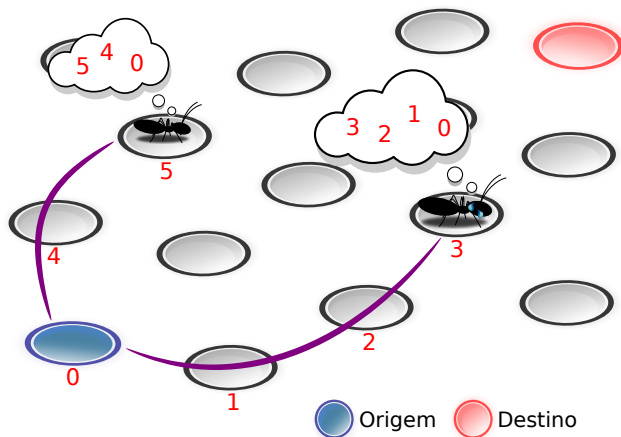


Figura 27: HOPNET - Roteamento reativo.

- *Formigas mais especializadas em armazenar e manter sequências de rotas do que usar e distribuir feromônio adequadamente/eficientemente pela rede:* o roteamento interno do HOPNET é pró-ativo e tende a gerar alto *overhead* na rede enquanto o externo é reativo e não utiliza totalmente os benefícios do feromônio nas decisões de roteamento. A solução mais simples para tal problema é balancear as transmissões sob demanda da rede calculando o raio das zonas para redes específicas. Porém, definir um valor de raio ótimo para cada rede é um grande desafio. Tal dificuldade de cálculo do raio é compreendida ao imaginar dois extremos em uma rede *densa* e *escalável*. Se o raio é muito pequeno, a probabilidade de transmissões sob demanda, para fora da zona, será maior. Caso o raio seja muito grande, o roteamento pró-ativo pesará muito na rede, visto que cada nodo mantém a própria zona. Em consequência disso, AD-ZRP utiliza zonas dinâmicas como método de roteamento, distribuindo o feromônio de forma eficiente e permitindo que as formigas utilizem tais informações para detectar rotas “seguras” em ambientes de alta mobilidade.

No entanto, apesar do AD-ZRP reduzir substancialmente o *overhead* de roteamento, o gerenciamento das zonas dinâmicas ainda tende a gerar uma baixa taxa de entrega na rede, como pode ser observado na Figura 24. Em redes mais propensas a erros que IEEE 802.11, tal taxa de entrega tende a diminuir ainda mais. Com isso, o algoritmo proposto não cumpre com todos os objetivos definidos para esta dissertação. Além disso, tal abordagem, juntamente com o método de zonas dinâmicas, não é projetada para utilizar heurísticas no roteamento. Portanto, o AD-ZRP é um algoritmo eficiente ao lidar com mobilidade e escalabilidade, mas carece de mecanismos ainda mais simples na manutenção de feromônio a fim de obter uma maior taxa de entrega de dados e uma melhor implantação em RSSFs. Com isso, a fim de lidar com tais limitações e cumprir com os objetivos estabelecidos nesta dissertação, um novo algoritmo é proposto e apresentado no Capítulo 4.

4 ALGORITMO PROPOSTO II: ADHOP

ADHOP é um protocolo de roteamento reativo e autoconfigurável que visa prover roteamento a redes sensores móveis (OKAZAKI; FROHLICH, 2011b). Este utiliza feromônio nas decisões de roteamento para lidar com topologias dinâmicas. O ADHOP é uma evolução do AD-ZRP, o qual também utiliza feromônio como métrica para tomadas de decisões no roteamento. O AD-ZRP é um algoritmo de roteamento eficiente em termos de *overhead* de roteamento. Porém, de acordo com as limitações do AD-ZRP apontadas na Seção 3.4, o ADHOP apresenta um foco diferente no roteamento. Este algoritmo apresenta um método de roteamento mais simples e robusto, e uma melhor integração e uso das informações heurísticas a fim de auxiliar as tomadas de decisões das formigas no roteamento em RSSFs.

ADHOP é classificado como um algoritmo de roteamento reativo. No entanto, este é projetado para **reagir** às mudanças de topologia sem influenciar o fluxo de dados e sem sobrecarregar a rede com procedimentos de reparo de rotas e pacotes de controle. Ou seja, minimizar o *overhead* do roteamento sem comprometer a taxa de entrega de dados. Inspirado na robustez dos protocolos para MANETs, os quais são capazes de lidar com mobilidade de forma eficiente, ADHOP também é capaz de lidar com topologias dinâmicas e outros importantes problemas de redes *ad hoc*. Porém, o ADHOP também necessita lidar com as usuais limitações de RSSFs. Para isso, a abordagem proposta consiste basicamente no uso do método chamado de *saltos dinâmicos* que, juntamente com estruturas simplificadas e o uso de heurísticas, possibilita ao ADHOP um roteamento eficiente em termos de taxa de entrega e um roteamento simples nas operações.

4.1 Método de Roteamento de Saltos Dinâmicos

A grande maioria dos protocolos de roteamento para redes *ad hoc* trabalha com o objetivo de construir rotas. Tais rotas podem ser exemplificadas como sequências de endereços que informam a ordem necessária que os dados devem trafegar entre os nodos para alcançar o destino. Outros protocolos, a fim de evitar a construção de tais sequências, trabalham na busca de informações sobre a topologia da rede de modo a informar a direção/sentido ou apenas o próximo salto que os dados precisam tomar para alcançar o destino. No entanto, tais métodos não são adequados para lidar com mobilidade visando uma utilização eficiente dos recursos, especialmente em RSSFs. Seja na construção de uma sequência de endereços ou na busca de informações de topologia a fim de informar a direção ou o sentido correto, o algoritmo necessita de uma grande quantidade de informação trafegada ou uma grande quantidade de informação atualizada e precisa. Ou seja, precisa-se de um algoritmo que gere um alto *overhead* de pacotes de controle e informações distribuídas ou de um roteamento pró-ativo para que as informações sejam precisas o suficiente no momento solicitado. Porém, obter informações de roteamento com um alto grau de precisão considerando as restrições de recursos de RSSFs não é uma tarefa fácil.

O método de *saltos dinâmicos* visa uma solução inspirada na natureza, especificamente na inteligência coletiva das formigas através da meta-heurística ACO. Muitos algoritmos de roteamento são inspirados em ACO, como o HOPNET, mas não pelo comportamento real das formigas. Protocolos baseados em ACO geralmente possuem a capacidade de se adaptar a mudanças de topologias que ocorrem na rede. Porém, em RSSFs móveis a adaptação não é unicamente suficiente. Também são necessários modos inteligentes de tráfego de dados, pouco uso de memória nas tabelas de roteamento, inteligência nos nodos para *sentir* mudanças de topologia e simplicidade nas operações.

Obter informações precisas em RSSFs móveis é um desafio muito grande, e o método de saltos dinâmicos abstrai tais informações na

forma de feromônio. A ideia é obter um algoritmo simples, mas robusto o suficiente para detectar mudanças de topologia sem esgotar os recursos dos nodos sensores. A maneira para alcançar tal objetivo é tratar topologias dinâmicas de forma que nenhum outro protocolo de roteamento trata, como se fosse “algo normal”. Protocolos de roteamento para redes móveis geralmente suportam métodos e técnicas para lidar com topologias dinâmicas. Diferentemente, o método de saltos dinâmicos não visa resolver os problemas referentes à mobilidade ou falhas de nodos como o HOPNET e o AD-ZRP fazem, mas apenas reagir a estas situações. Por este motivo ADHOP inspira-se no comportamento real de uma colônia de formigas.

No mundo real, as formigas se comunicam umas com as outras através de feromônio. Através desta sensibilidade, elas são capazes de seguir trilhas de feromônio na busca por alimentos. Inicialmente, as formigas usam caminhos aleatórios do formigueiro até o alimento esperado. No retorno para o formigueiro, elas depositam o feromônio pelo caminho a fim de indicar para as outras formigas o caminho tomado, como apresentado na Figura 28. Enquanto houver alimento e formigas trafegando pelo caminho tomado pela primeira formiga, a trilha utilizada terá a concentração de feromônio reforçada.

Formigas podem seguir diversos caminhos indiscriminadamente, mas elas tendem a seguir trilhas com maior concentração de feromônio. A cada passo, a formiga verifica o meio e toma a decisão baseado no que ela *sentiu* naquele momento e naquela posição, como pode ser apresentado na Figura 29. Por isso, caminhos mais curtos ou menos custosos tendem a ser mais atrativos. No entanto, no decorrer do tempo, a concentração na trilha reduz gradualmente, isto é, o feromônio tende a evaporar. Como consequência, quando o alimento esgotar, a trilha não será reforçada, e o feromônio dissipará lentamente. Tal comportamento ajuda as formigas a lidar com mudanças no ambiente.

A ideia é utilizar, literalmente, este comportamento coletivo das formigas na descoberta e manutenção das melhores rotas entre os nodos da rede. Deste modo, as formigas podem manter as tabelas de feromônio atualizadas eficientemente usando feromônio, além de

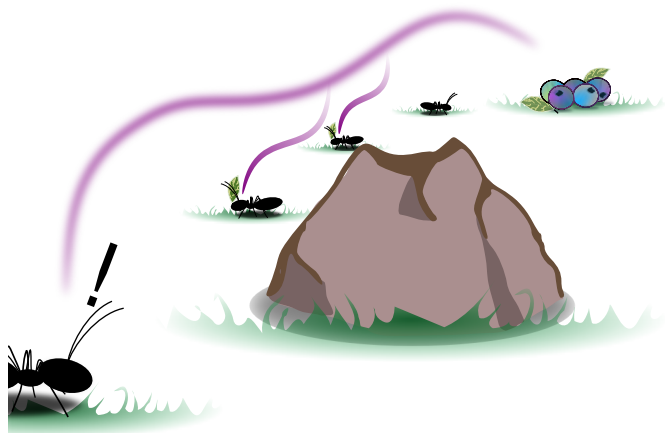


Figura 28: Trilha de feromônio.



Figura 29: Escolhendo uma trilha.

usufruir da capacidade delas em detectar mudanças na topologia. O método de saltos dinâmicos consiste basicamente na emulação do comportamento das formigas descrito acima. Toda formiga, a cada passo, verificará a concentração de feromônio em cada nodo trafegado até o destino esperado. Ou seja, a abordagem proposta abstrai qualquer tipo de informação e procedimento na forma de feromônio, e foca apenas no próximo salto que a formiga deve tomar a fim de alcançar o destino, como ilustrado na Figura 30.

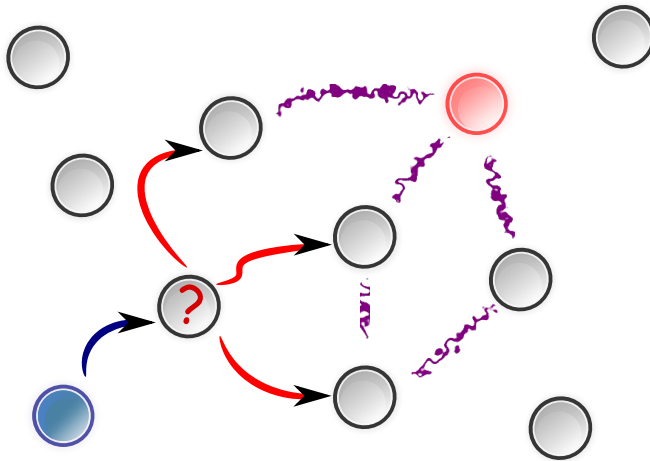


Figura 30: Saltos dinâmicos.

A diferença entre outros métodos semelhantes está na maneira como este método é executado. Saltos dinâmicos ocorrem apenas sob demanda, ou seja, este método é um esquema de roteamento reativo. Porém, este método apresenta algumas diferenças básicas com os demais protocolos reativos. Como dito anteriormente, a reação está focada principalmente nas mudanças de topologia. Qualquer variação de feromônio é sentida pelas formigas e alternativas são estabelecidas dinamicamente sem qualquer procedimento de reparo de rotas ou disseminação de notificações sobre a topologia. O motivo de não haver procedimentos de reparo ou notificações consiste na forma que o método trata tais erros. Como dito anteriormente, o método de sal-

tos dinâmicos considera erros de topologia algo normal na rede, que ocorrem cedo ou tarde. Logo, em vez de tentar corrigir ou reparar, o algoritmo simplesmente tenta uma alternativa previamente estabelecida.

4.2 Estruturas de Tabela e Formiga

O ADHOP possui uma estrutura única de tabela de roteamento denominada *tabela de feromônio*. Tal estrutura é responsável por mapear e armazenar a concentração de feromônio até um determinado destino. Semelhante à tabela de roteamento interno do HOPNET e à tabela inversa de feromônio do AOER, a estrutura do ADHOP prova ser um eficiente meio de lidar com feromônio (WANG et al., 2009; SHUANG; LI; CHEN, 2009). No entanto, a estrutura da tabela do ADHOP permite operações ainda mais simples e rápidas. A estrutura básica consiste de uma tabela de dispersão (*hash*) de listas ordenadas por feromônio, como retratado no diagrama de classe na Figura 31.

Como exemplificado pela Figura 32, as chaves das tabelas de dispersão equivalem aos endereços dos nodos destinos enquanto a lista ordenada mantém uma lista de nodos vizinhos, ambas limitadas pelas constantes *MAX_HASH* e *MAX_LIST*, respectivamente. Tal estrutura permite realizar operações de busca, atualização e inserção com complexidade $O(1)$, além de armazenar as informações de maneira compacta.

A fim de realizar tais operações, ADHOP apresenta a seguinte coleção de formiga:

- *Formiga Transportadora* (FT);
- *Formiga Transportadora de Exploração* (FTE).

Ambas as formigas têm a função de transportar os dados da origem ao destino. Porém, a FTE é utilizada para explorar novas opções de roteamento enquanto que a FT apenas segue e reforça a trilha de

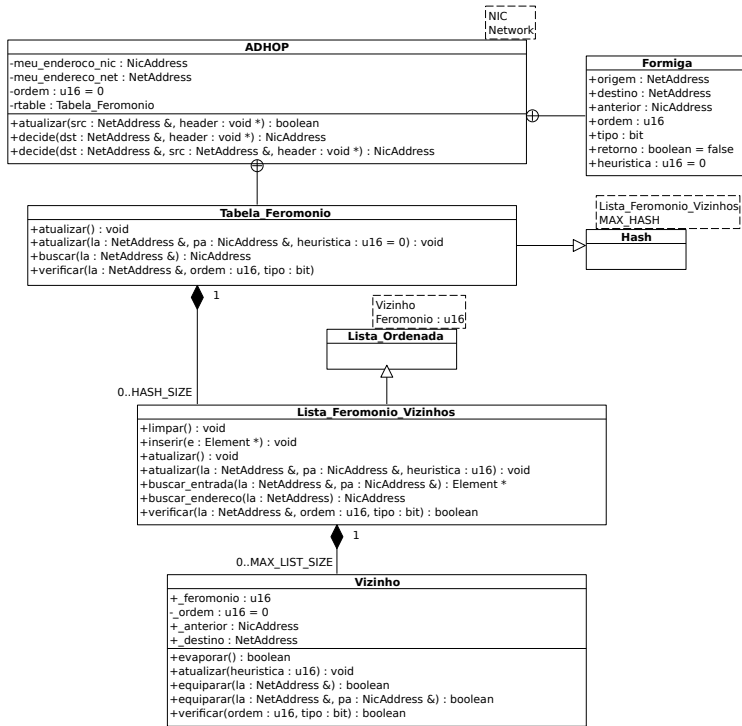


Figura 31: ADHOP - Estrutura da tabela de roteamento.

Tabela de Roteamento								
Destinos de chave == 0			Destinos de chave == 1			Destinos de chave == 2		
Dst. 03	Viz. 02	Ferom. 100	Dst. 04	Viz. 02	Ferom. 100	Dst. 14	Viz. 06	Ferom. 150
Dst. 00	Viz. 00	Ferom. 95	Dst. 04	Viz. 00	Ferom. 80	Dst. 11	Viz. 06	Ferom. 90
Dst. 06	Viz. 06	Ferom. 80	Dst. 10	Viz. 06	Ferom. 75	Dst. 14	Viz. 05	Ferom. 85
Dst. 09	Viz. 02	Ferom. 30				Dst. 14	Viz. 02	Ferom. 20

Figura 32: ADHOP - Tabela de feromônio ($MAX_HASH == 3$ e $MAX_LIST == 4$).

feromônio encontrada pela FTE, como ilustrado na Figura 33. As formigas transportam os dados com o objetivo de reforçar as trilhas de feromônio enquanto os dados trafegam, reduzindo o *overhead* de pacotes de controle na rede e ainda permitindo que os dados, juntamente com as formigas, sejam sensíveis às mudanças de topologia.

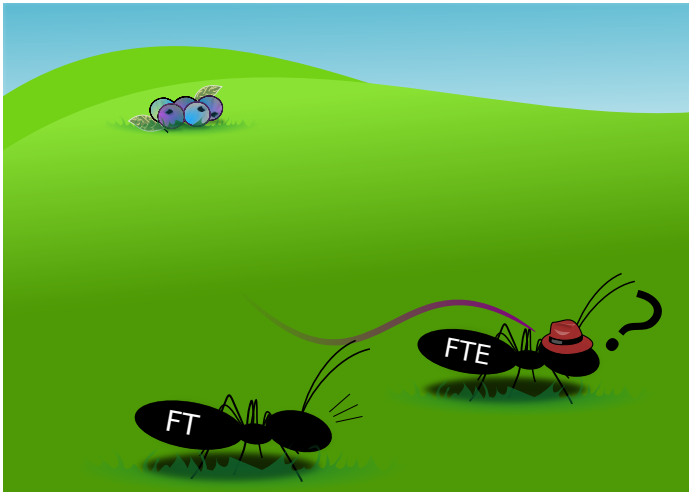


Figura 33: ADHOP - Tipos de formigas.

Apesar de cada uma exercer uma função diferente, ambas compartilham a mesma estrutura. A Figura 34 apresenta a estrutura da formiga no ADHOP.

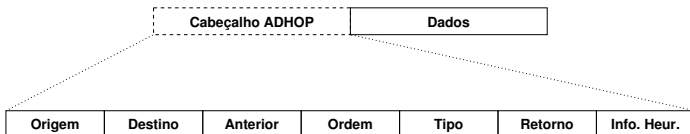


Figura 34: ADHOP - Estrutura da formiga.

A formiga inclui os campos de endereços: *Origem* e *Destino*. Além do campo *Anterior*, o qual representa o endereço do nodo do

salto anterior. O campo *Ordem* é utilizado para o controle de sequência das transmissões. O campo *Tipo* indica a tipo da formiga (FT ou FTE), e o campo *Retorno* indica o sinal de retorno da formiga. O campo *Info. Heur.* é responsável por armazenar a informação heurística necessária para o auxílio no cálculo das taxas de depósito e evaporação de feromônio. Tais informações ajudam as formigas a utilizarem melhores táticas para difundir conhecimento na rede e verificar a concentração de feromônio nos nodos, reduzindo a complexidade das operações de descoberta e manutenção de rotas. A ideia é utilizar tais informações para reforçar as melhores rotas de acordo com a heurística utilizada.

4.3 O roteamento ADHOP

A fim de realizar decisões de roteamento corretamente, as formigas são responsáveis por distribuir feromônio pela rede e indicar opções de caminhos para outras formigas, incluindo a melhor rota a ser tomada naquele instante. Os nodos não precisam se preocupar em descobrir ou manter rotas, pois ambos os procedimentos são realizados naturalmente pelas formigas usando o método de roteamento de saltos dinâmicos. O roteamento é realizado utilizando o método de saltos dinâmicos com as estruturas otimizadas descritas na seção anterior (Seção 4.2). Por este motivo as informações que uma formiga necessita para tomar as decisões de roteamento e disseminar conhecimento pela rede tende a ser mínima. Como descrito na Figura 34 e ilustrado na Figura 35, as formigas apenas precisam conhecer três informações para realizar roteamento com ADHOP: de onde ela veio (origem), para onde ela vai (destino) e onde ela estava no passo anterior (último nodo / nodo anterior).

Na maioria dos protocolos de roteamento para redes *ad hoc*, quando ocorre alguma mudança na topologia da rede, procedimentos de reparo ou pacotes de controle são disseminados na rede a fim de notificar e/ou reparar a quebra da rota. ADHOP tenta lidar com estas situações através das informações de feromônio distribuídas pela rede. Mas na realidade, o protocolo não fará nada para resolver a situação. Geralmente, as rotas do ADHOP podem ser descritas no se-

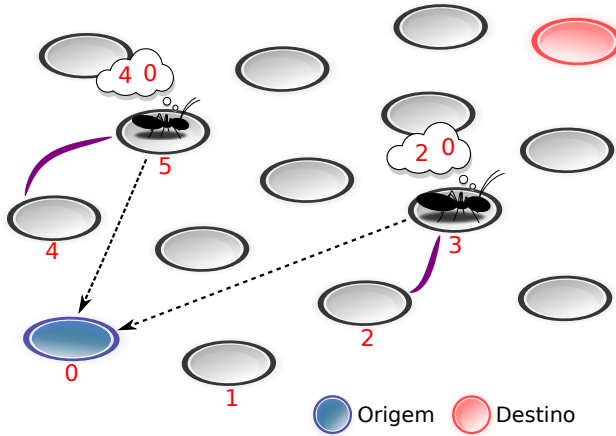


Figura 35: Roteamento ADHOP.

guinte formato: “para alcançar o nodo 0, a melhor opção é, conforme a concentração de feromônio no momento, passar pelo nodo 5”. No entanto, se um nodo se movimenta ou falha, algo comum em RSSFs, a trilha de feromônio que passa por tal nodo tende a evaporar. O algoritmo, então, irá redirecionar a formiga dinamicamente através de rota: “para alcançar o nodo 0, a melhor opção é, conforme a concentração de feromônio no momento, passar pelo nodo 6”, como ilustrado na Figura 36.

Portanto, as rotas no ADHOP não são predeterminadas e os saltos são dinamicamente decididos a cada instante, adaptando-se às mudanças de topologia e necessidades da rede de forma transparente. Assim, as tabelas de roteamento não precisam ser periodicamente atualizadas ou tratadas de forma pró-ativa a fim de obter informações precisas, pois elas tendem a se atualizarem sozinhas através da evaporação do feromônio enquanto as mudanças nas topologias serão observadas involuntariamente pelas formigas. Em consequência disso, procedimentos de reparo e pacotes de controles não são necessários no algoritmo do ADHOP.

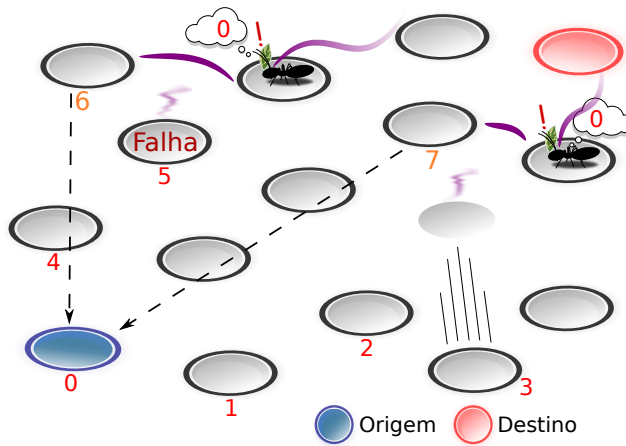


Figura 36: Falhas, mobilidade e caminhos alternativos.

Desta forma, ADHOP se especializa apenas na distribuição e gerenciamento eficiente de feromônio com a finalidade de realizar roteamento de forma robusta, simples e leve. Assim, o algoritmo proposto é apresentado através de diagramas de sequência, descrevendo as funcionalidades e comportamento.

A Figura 37 retrata o diagrama de sequência para transmissão de dados do ADHOP. Como dito anteriormente, os dados são transmitidos juntamente com a formiga para assegurar que mudanças repentinas na rede não interfiram no transporte dos dados. Deste modo, os dados podem ser redirecionados para rotas mais seguras. De qualquer maneira, a transmissão dos dados ocorre de modo simples: se o destino é conhecido, os dados serão transmitidos utilizando FTs, caso contrário, os dados serão transmitidos com FTEs. Nota-se que, independente de haver ou não um destino conhecido, a transmissão ocorre sem a necessidade de se conhecer uma rota. Como dito anteriormente, ADHOP considera mudanças de topologia algo frequente e normal nas redes, portanto, o protocolo sempre assume que uma rota previamente utilizada não exista mais.

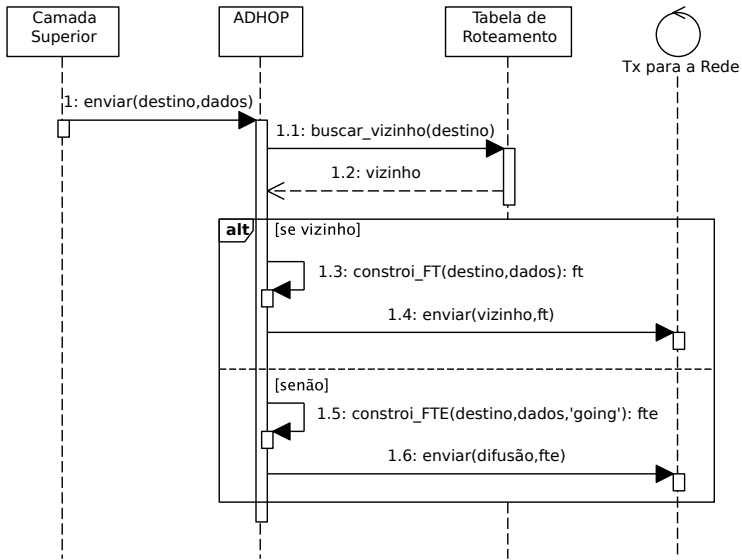


Figura 37: ADHOP - Transmissões de dados.

Basicamente, FTEs são formigas responsáveis por procurar um destino desconhecido na rede. Mas na verdade, elas fazem mais que isso, elas buscam por trilhas de feromônio que levam ao nodo destino. Ou seja, a função da formiga é descobrir e aumentar as opções de tráfego de dados até um determinado destino. Um efeito colateral disso é a capacidade de unir ou ligar duas ou mais trilhas. Inicialmente, o nodo origem difunde a FTE pela rede com o objetivo de encontrar o nodo destino ou algum nodo que tenha a trilha para o nodo destino. Caso a FTE encontre uma trilha para o destino, o protocolo enviará uma FT com os dados para o destino e a FTE retornará para a origem a fim de reforçar a trilha descoberta. Caso a FTE atinja o nodo destino, os dados serão entregues e a FTE retornará ao nodo origem a fim de reforçar a trilha descoberta. Tal comportamento pode ser descrito no diagrama de sequência na Figura 38.

FTs, diferentemente, têm como objetivo reforçar o feromônio da trilha utilizada. Elas são responsáveis por entregar os dados a qualquer destino conhecido na rede. Ou seja, quando uma rota é descoberta pelas FTEs, as seguintes transmissões serão realizadas pelas FTs até a trilha evaporar completamente. No entanto, a qualquer momento durante a transmissão de uma FT, se a trilha de feromônio de desfêz, qualquer nodo pode enviar uma FTE com os dados a fim de atingir o destino, como apresentado no diagrama de sequência na Figura 39.

Deste modo, através deste comportamento, de ambas as formigas, de delegar a entrega de dados uma a outra sempre que conveniente, o algoritmo ADHOP consegue não apenas lidar com situações de casualidades na rede, mas também consegue reagir a quaisquer mudanças de topologia de forma simples e sem gerar qualquer *overhead* na rede. No entanto, a responsabilidade não deve ser atribuída apenas ao comportamento do algoritmo. O ADHOP depende muito das taxas de depósito e evaporação do feromônio, assim como as heurísticas configuradas e atribuídas ao algoritmo.

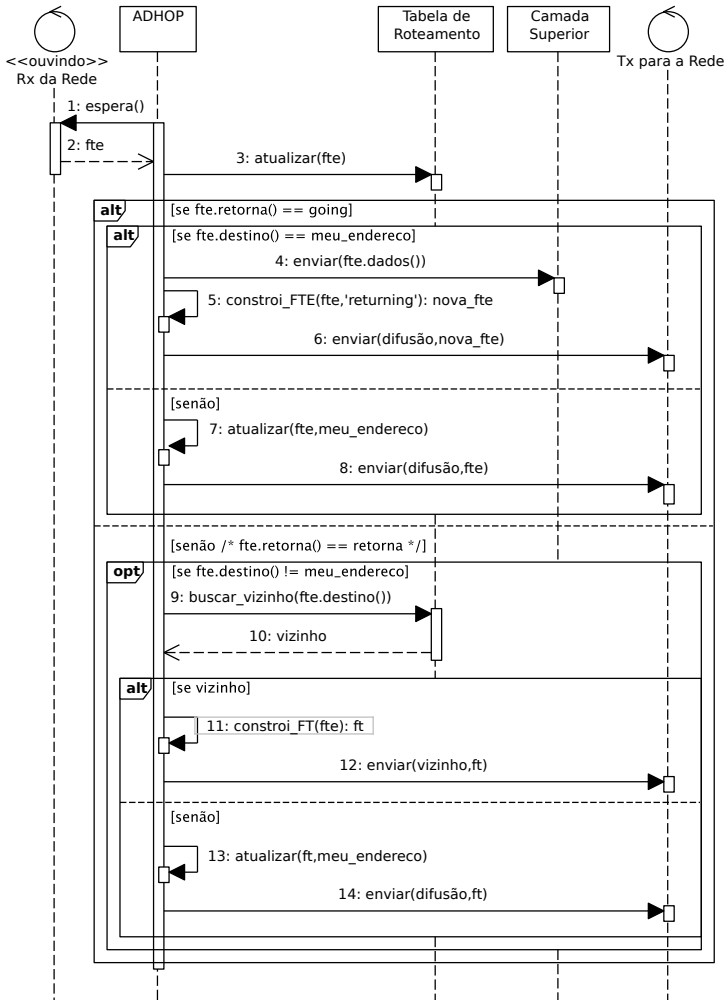


Figura 38: ADHOP - Recepção da FTE.

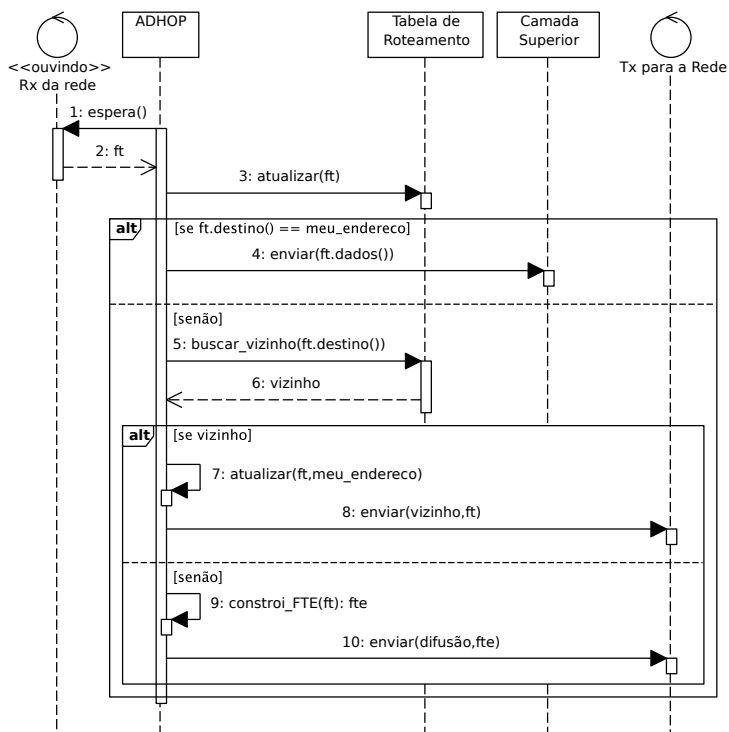


Figura 39: ADHOP - Recepção da FT.

4.4 Heurísticas

Heurística é a parte da pesquisa que visa à aproximação das soluções ótimas dos problemas através de descobertas empíricas. Trata-se de métodos ou algoritmos exploratórios e exaustivos para solução de problemas de grande complexidade computacional. As soluções são buscadas por aproximações sucessivas, avaliando-se os progressos alcançados, até que o problema seja resolvido. As técnicas heurísticas não asseguram as melhores soluções, mas soluções válidas.

O uso de heurísticas no algoritmo do ADHOP consiste na busca de um ou mais objetivos a fim de satisfazer as necessidades da rede, além de uma alta taxa de entrega de dados. Por exemplo, reduzir o consumo de energia dos nodos, balancear o consumo de energia na rede, favorecer trilhas de feromônio com menor latência, favorecer trilhas de feromônio com menor distância, etc. A ideia é utilizar as informações disponibilizadas pelos nodos/rede e apresentá-las como um problema cuja solução deva ser maximizada ou minimizada. As informações disponibilizadas, representadas numericamente, são utilizadas nas equações das taxas de depósito e evaporação de feromônio, influenciando a concentração de feromônio nas trilhas e por consequência, as tomadas de decisão do roteamento, como ilustrado pela Figura 40.

Como dito anteriormente, o gerenciamento de feromônio do ADHOP consiste na utilização de duas equações:

- *Equação de atualização de feromônio;*
- *Equação de evaporação.*

A equação de atualização de feromônio funciona da seguinte maneira: cada formiga, baseado na concentração de feromônio nas trilhas, seleciona o nodo v_j para o próximo salto a partir do nodo atual v_i . Na verdade, v_j é o nodo vizinho que possui a maior concentração de feromônio. No nodo v_j , a formiga atualiza a concentração de feromônio $\tau_{i,o}$ na entrada (v_i, v_o) da tabela de feromônio, onde v_o re-

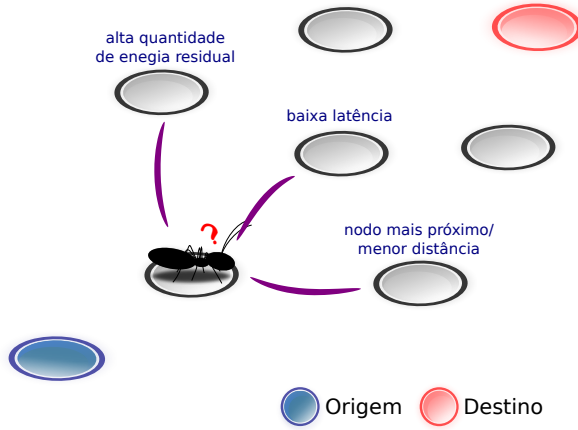


Figura 40: ADHOP - Heurísticas.

apresenta o nó origem da formiga, utilizando a seguinte equação (DORIGO; BIRATTARI; STUTZLE, 2006):

$$\tau_{i,o} = (1 - \varphi) \cdot \tau_{i,o} + \varphi \cdot \tau_0 \quad (4.1)$$

onde $\varphi \in (0, 1]$ representa o coeficiente de declínio de feromônio e τ_0 é o valor da quantidade de depósito inicial de feromônio. Tal equação permite diversificar o processo de busca ao decrementar a concentração de feromônio nas trilhas enquanto realiza o depósito de feromônio, permitindo que outras formigas tentem outras opções com maior facilidade. Ela também auxilia e aumenta os efeitos do método de saltos dinâmicos.

A equação de evaporação funciona da seguinte maneira: a evaporação ocorre periodicamente para todos os nós da rede, utilizando a seguinte equação (DORIGO; BIRATTARI; STUTZLE, 2006):

$$\tau_{i,j} = (1 - \rho) \cdot \tau_{i,j}, \quad \forall i \in N, \quad \forall j \in Z \quad (4.2)$$

onde $\rho \in (0, 1]$ é o coeficiente de evaporação, N representa o conjunto

de nodos vizinhos e Z representa o conjunto de nodos os quais, juntamente com N , definem entradas (v_i, v_j) na tabela de feromônio.

Ao utilizar as Equações (4.1) e (4.2), ADHOP permite o uso de diversas heurísticas para realizar roteamento. Diferentemente das Equações (3.3) e (3.4), o coeficiente de declínio de feromônio (φ) e o coeficiente de evaporação (ρ) são calculados usando o campo *Info. Heur.* da estrutura da formiga (Figura 34). Ou seja, o campo *Info. Heur.* mantém informações que afetam as taxas de depósito e evaporação de feromônio. O objetivo de tal campo é possibilitar que o algoritmo beneficie um determinado nodo mais do que outros, dado certo momento e situação. Por exemplo, o nodo vizinho A possui 90% de carga de bateria enquanto o nodo vizinho B possui 75%. Utilizar tais informações como heurísticas possibilitam uma infinidade de possibilidades para aplicar ao roteamento. Uma possível abordagem seria aumentar o depósito de feromônio e diminuir a taxa de evaporação quanto maior a carga da bateria do nodo vizinho. Com esta abordagem, o nodo A seria o escolhido para as próximas transmissões em vez do nodo B . Tais características garante um algoritmo flexível para os moldes de RSSFs, sempre visando uma adaptação ao contexto da rede.

5 EXPERIMENTAÇÃO E AVALIAÇÃO

Os experimentos foram conduzidos em dois estágios de simulações a fim de analisar e avaliar a abordagem proposta de forma mais completa. Em ambos os estágios, o cenário das simulações consiste em um ambiente propenso a erros e de mobilidade.

No primeiro estágio, considera-se um ambiente de MANETs, ao utilizar o padrão de redes sem fio *IEEE 802.11*. O objetivo deste estágio é avaliar o ADHOP quanto à capacidade de se adaptar em meios de alta mobilidade. O ADHOP é avaliado e comparado com o AD-ZRP através de análises de desempenho. Neste estágio, os algoritmos também são analisados utilizando as heurísticas de balanceamento de energia e de rotas de baixa latência.

No segundo estágio, o desempenho do ADHOP é avaliado em um ambiente mais próximo de RSSFs. Neste estágio, retrata-se um cenário de nodos móveis utilizando o padrão sem fio *IEEE 802.15.4*. A abordagem proposta é comparada a outros algoritmos bem conhecidos para redes *ad hoc* através de um ambiente configurado para simular, além da mobilidade, todas as restrições de RSSFs em termos de largura de banda. Da mesma forma, como no primeiro estágio, os algoritmos são analisados utilizando as heurísticas de balanceamento de energia e de rotas de baixa latência.

5.1 Primeiro Estágio: *IEEE 802.11*

Neste estágio, as implementações foram baseadas no GloMoSim. A avaliação ocorre considerando diversos cenários de simulações. A Tabela 3 apresenta os parâmetros de simulação utilizados no GloMoSim. Cada cenário de simulação é executado por um total de 900 segundos num ambiente propenso a alta taxa de perda de dados. Os nodos são implantados aleatoriamente em uma área retangular de 700x400 metros quadrados. Cada nodo move-se a uma velocidade

máxima de 10 metros por segundo, de acordo com o algoritmo de mobilidade RWP. O tráfego de dados é gerado por 20 nodos móveis para outros 20 nodos móveis a uma taxa constante de *bits*. O poder de transmissão do rádio é estabelecido com $15dBm$ com uma taxa de *bits* de 2 *megabits* por segundo.

Este cenário base é o utilizado nos experimentos com o GloMoSim. Os valores demonstrados abaixo representam a média dos resultados de um cenário que é replicado 10 vezes. Os cenários de testes são obtidos ao variar parâmetros específicos no cenário base, tal como o número de nodos, raio da zona e velocidade dos nodos. O número de nodos varia de 20 a 200, enquanto o raio de uma zona varia de 2 a 5 saltos.

Tabela 3: Configuração do GloMoSim.

Parâmetros	
Tempo de simulação	900s
Quantidade de nodos	20 ~ 200
Área	700mX400m
Algoritmo de mobilidade	RWP
Velocidade dos nodos	1 ~ 10m/s
Tamanho dos dados da aplicação	64B
Frequência de envio dos dados	3s
Protocolo de transporte	UDP
Protocolo de rede	IP
Máscara da rede	255.255.0.0
Frequência de canal	2.4GHz
Taxa de <i>bits</i>	2Mbit/s
Poder de transmissão do rádio	15dBm
Sensitividade do rádio	-71.067dBm

5.1.1 Avaliação - Mobilidade

A Figura 41 apresenta a taxa de entrega dos pacotes de dados neste ambiente de simulação. Conforme o número de nodos aumenta,

a taxa de entrega tende a aumentar devido às formigas, que são capazes de direcionar o fluxo de dados pelas melhores trilhas de feromônio até determinado destino. Pela figura, nota-se que se a rede é muito grande e densa, então a taxa de entrega tende a ser alta devido à grande quantidade de opções que um grande número de nodos proporciona. Por outro lado, se a rede é pequena e esparsa, então a taxa de entrega tende a ser menor devido à falta de *links* física entre os nodos. Nota-se também que o ADHOP retrata melhores resultados de taxa de entrega em redes de alta escalabilidade e alta mobilidade. ADHOP possui operações mais simples e robustas na manutenção das rotas, permitindo ao algoritmo identificar as mudanças de topologia através das variações de feromônio entre os nodos da rede.

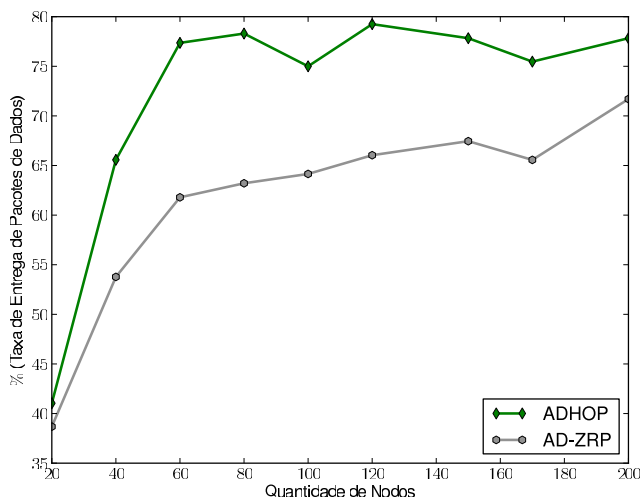


Figura 41: Taxa de entrega de dados.

A Figura 42 retrata a comparação dos protocolos de roteamento em razão ao *overhead* de pacotes de controle na rede. No ADHOP e no AD-ZRP, os dados são transportados pelas formigas a fim de reduzir os pacotes de controle na rede. Além disso, não há procedi-

mentos de reparo de rotas, pois as formigas se baseiam na distribuição de feromônio para formar o conhecimento da rede. Por este motivo, o ADHOP visa manter o *overhead* de roteamento da rede quase que constante em redes escaláveis e densas. O mesmo ocorre com o AD-ZRP, mas este apresenta uma menor taxa de entrega, como apresentado na Figura 41.

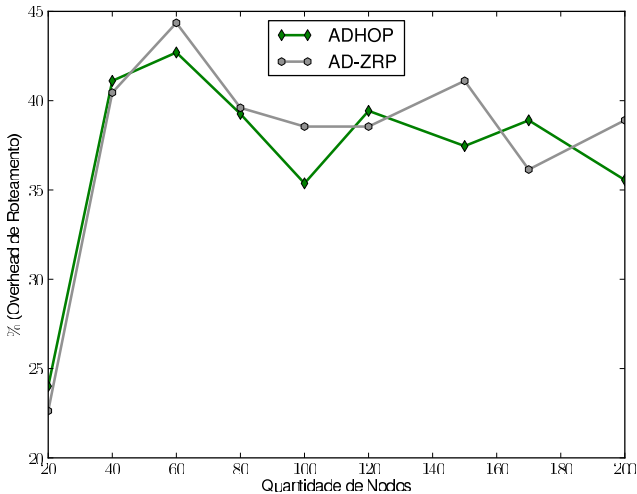


Figura 42: *Overhead* de roteamento.

5.1.2 Avaliação - Heurísticas

Como dito anteriormente, o ADHOP oferece a possibilidade de utilizar informações heurísticas para auxiliar o roteamento, na verdade, através das Equações (4.1) e (4.2), ADHOP permite usar informações heurísticas a fim de modificar o comportamento padrão do algoritmo. Tal abordagem geralmente tende a acelerar o consumo de energia dos nodos que fazem parte de uma rota “ótima” até de-

terminado destino. No entanto, ao utilizar a carga da bateria como informação heurística, o algoritmo possibilita diversas abordagens para lidar com o consumo de energia. Neste experimento, a abordagem utilizada será o balanceamento de energia entre os nodos da rede, em contraste com o comportamento padrão do ADHOP que visa à baixa latência das rotas. Deste modo, as formigas privilegiam as trilhas de feromônio com nodos que possuem carga de bateria superior aos demais nodos. Isto é, nodos vizinhos com maior quantidade de carga de energia recebem maior quantidade de feromônio enquanto a taxa de evaporação será menor. Tal abordagem visa maximizar a probabilidade de escolha de rotas com maior quantidade de energia residual, implicando em balancear o consumo de energia de toda a rede. Outra abordagem utilizada no experimento se refere à latência entre o nodo origem e o nodo destino. Tal abordagem visa o tráfego de mensagens fim-a-fim de forma mais rápida. Neste caso, as formigas privilegiam as trilhas de feromônio que realizam a entrega de dados mais rapidamente.

A ideia do experimento é analisar o comportamento do algoritmo e validar o uso de informações heurísticas de modo a tornar o ADHOP em um protocolo flexível o bastante para se adaptar aos diferentes contextos que uma rede sensor pode ter. Deste modo, na heurística de baixa latência, o ADHOP utiliza ϕ igual a percentagem da latência da rota atual em relação à rota mais rápida até determinado destino e ρ igual a 0,5. Na heurística de balanceamento de energia, o ADHOP utiliza ϕ igual a percentagem de energia residual do nodo vizinho e ρ igual a percentagem de energia residual do próprio nodo. A Tabela 4 apresenta a configuração energética do GloMoSim.

Tabela 4: Configuração energética do GloMoSim.

Parâmetros de energia	
Capacidade da bateria	850mAh
Voltagem da bateria	3V
Consumo do rádio - dormindo	0,8mA
Consumo do rádio - recebendo	16mA
Consumo do rádio - transmitindo	19.3mA

A figura 43 retrata o desvio padrão do tempo de vida dos nodos da rede. Conforme a quantidade de nodos aumenta, o desvio padrão tende a diminuir, pois a rede apresenta mais opções para o roteamento, e mais rotas são exploradas. Nota-se que o algoritmo consciente do consumo de energia obtém melhores resultados ao balancear o consumo na rede, prolongando assim, o tempo de vida.

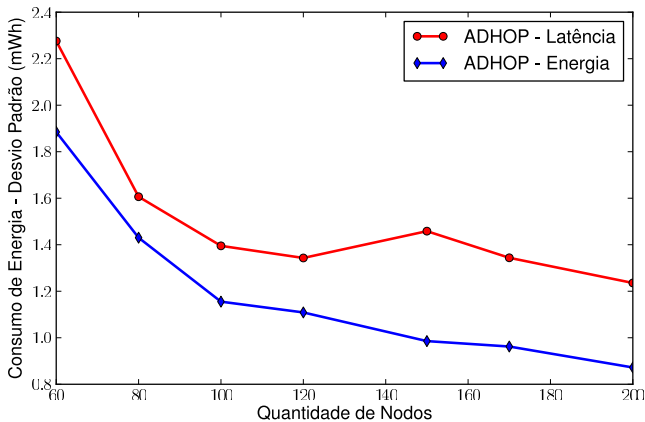


Figura 43: Desvio padrão do tempo de vida dos nodos.

A Figura 44 retrata o consumo médio de energia por nodo. Nota-se que apesar do algoritmo consciente do consumo de energia apresentar melhor balanceamento de energia na rede (ver Figura 43), o consumo médio é pior que na abordagem da baixa latência. O motivo é que a abordagem de balanceamento de energia tende a usar todos os nodos a fim de equalizar as cargas de baterias, diferentemente da abordagem da baixa latência que visa à rota mais rápida até o destino.

A figura 45 retrata taxa de entrega de dados comparando ADHOP em ambas as abordagens de uso das informações heurísticas. O ADHOP consciente de energia realiza as tomadas de decisão baseados na concentração de feromônio, o qual é influenciado pela energia residual dos nodos. A outra abordagem objetiva rotas de

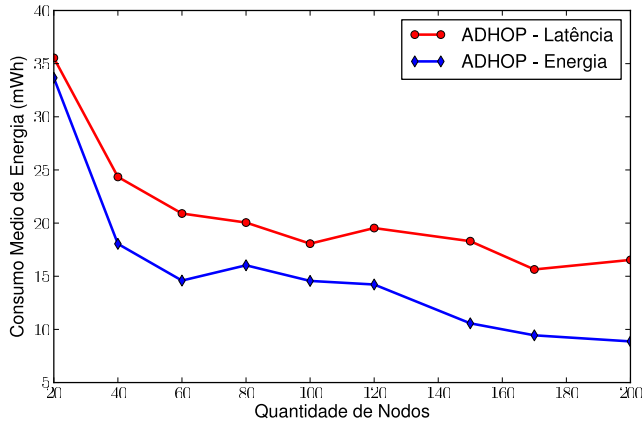


Figura 44: Tempo de vida esperado.

baixa latência. Conforme a quantidade de nós aumenta, a taxa de entrega também tende a aumentar, e a abordagem consciente de energia apresenta melhores resultados. Isto ocorre devido ao “efeito colateral” desta abordagem, em que, ao tentar balancear o consumo de energia, o algoritmo também balanceou a carga de dados trafegados pela rede.

5.2 Segundo Estágio: *IEEE 802.15.4*

Neste estágio, as implementações foram baseadas no OMNeT++, um simulador modular e extensível na modelagem de protocolos para comunicação sem fio. As simulações ocorrem através de diversos cenários de simulações utilizando protocolos bem conhecidos para redes *ad hoc*, tais como AODV, DYMO e OLSR.

A Tabela 5 apresenta os parâmetros de simulações do OMNeT++. De maneira similar ao estágio anterior, os cenários são executados por

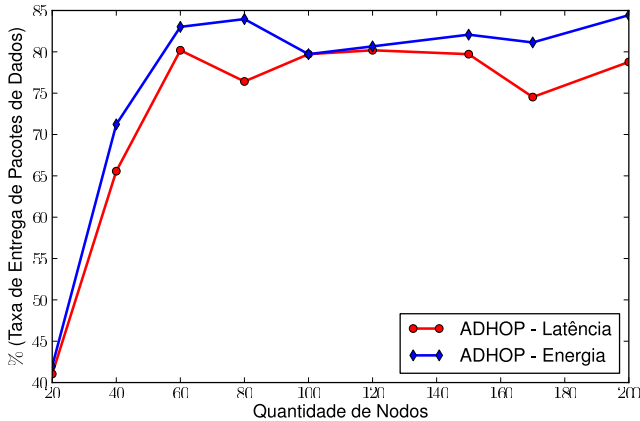


Figura 45: Taxa de entrega de pacotes de dados.

300 segundos em um ambiente de mobilidade e alta taxa de perda de pacotes. Neste estágio, tais cenários de simulações retratam com mais exatidão os ambientes de RSSFs, para uma análise mais adequada do algoritmo. Os nós são implantados aleatoriamente em uma área quadrática de 1000×1000 metros quadrados, com cada nó se movendo a velocidade máxima de 5 metros por segundo, através do algoritmo de *Mobilidade em Massa* (do inglês *Mass Mobility* - MM). O tráfego de dados é gerado por 20 nós móveis para outros 20 nós móveis. Na camada física, a sensibilidade dos nós é de $-85dBm$, o ruído térmico é de $-110dBm$ e o poder de transmissão é de $1mW$ com uma taxa de *bits* de 250 *kilobits* por segundo. Este cenário base é utilizado para os experimentos no OMNeT++. Os valores demonstrados abaixo representam a média dos resultados de um cenário que é replicado 10 vezes. Os cenários de testes foram obtidos ao variar parâmetros específicos, como a quantidade de nós que varia de 20 a 100.

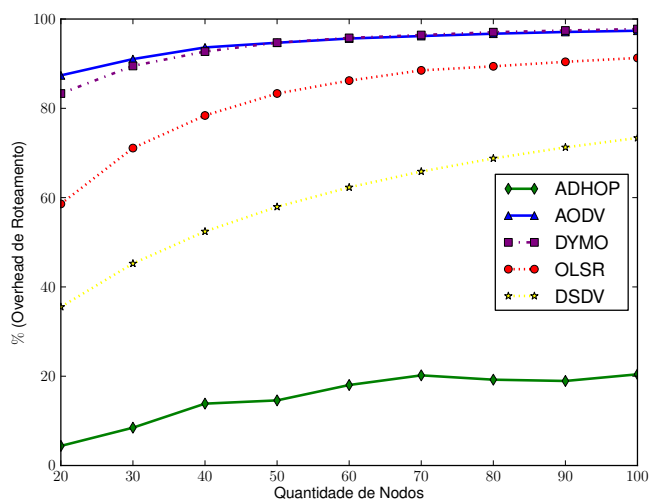
Tabela 5: Configuração do OMNeT++.

Parâmetros	
Tempo de simulação	300s
Quantidade de nodos	20 ~ 100
Área	1000mX1000m
Algoritmo de mobilidade	MM
Velocidade dos nodos	1 ~ 5m/s
Tamanho dos dados da Aplicação	32B
Frequência de envio dos dados	2s
Protocolo de transporte	UDP
Largura do cabeçalho do UDP	8B
Protocolo de rede	IP
Largura do cabeçalho do IP	20B
Máscara da rede	255.255.0.0
Largura do cabeçalho do ADHOP	20B
Poder de transmissão do rádio	1mW
Sensitividade do rádio	-85dBm
Ruído Térmico	-110dBm
Frequência de canal	2.4GHz
Taxa de bits	250kbit/s
IEEE 802.15.4 MTU	127B
Largura do cabeçalho IEEE 802.15.4	22B
IEEE 802.15.4 - Tamanho máximo do quadro	102B

5.2.1 Avaliação - Mobilidade

A Figura 46 retrata a comparação entre protocolos em razão ao *overhead* de pacotes de controle na rede. AODV e DYMO são protocolos reativos de roteamento, os quais visam um baixo *overhead* de roteamento através de mecanismos de roteamento sob demanda. No entanto, tais mecanismos não se aplicam em RSSFs móveis. Em ambientes em que os nodos estão constantemente se movendo, pacotes acabam se perdendo em algum momento. Porém, protocolos como AODV e DYMO, apesar de reativos, visam sempre manter rotas corretas até o destino desejado. Então, a cada momento que uma rota é alterada ou desfeita, o protocolo tenta repará-la ou obter uma nova rota. Como mencionado anteriormente, AODV utiliza RREQ, RREP e RERR como mensagens de controle a fim de construir e manter as rotas na rede. Em consequência disso, em cenários de topologias dinâmicas, as rotas tendem a se invalidarem a todo instante. Isto implica em uma grande quantidade de pacotes de controle trafegando pela rede. Por exemplo, cada momento em que uma rota é invalidada, o AODV inunda a rede com RERRs, RREQs e RREPs para notificar os demais nodos, e reparar ou construir novas rotas. DYMO recorre dos mesmos procedimentos. Porém, o maior problema é a alta taxa de perda de dados das redes *IEEE 802.15.4*, que implica no aumento da quantidade de pacotes de controle transmitidos na rede. Diferentemente, ADHOP é um protocolo reativo às mudanças de topologia, ou seja, trata-se de um protocolo que se preocupa com o fluxo de dados da rede. A fim de não quebrar tal fluxo, o ADHOP tenta lidar com tais casualidades da maneira mais simples e rápida possível, produzindo o mínimo de *overhead* na rede. Para isso, a abordagem proposta depende de uma distribuição eficiente de feromônio. Deste modo, ADHOP é capaz de obter baixo *overhead* de roteamento até mesmo para redes escaláveis e densas. OLSR e DSDV, como protocolos pró-ativos tendem a possuir um *overhead* esperado, pois dependem da configuração da periodicidade em que as informações são trocadas a fim de atualizar as tabelas de roteamento.

Do mesmo modo que no estágio de simulação anterior, a taxa de perda de dados é avaliada de duas maneiras distintas:

Figura 46: *Overhead* de roteamento.

- através de quebra de rotas, referentes aos problemas de retransmissões de pacotes recebidos por rotas que não existem mais;
- através de falhas de conectividade entre os nodos, referentes aos problemas que se originam na camada de *Controle de Acesso ao Meio* (do inglês *Medium Access Control* - MAC).

A Figura 47 retrata a taxa de perda de quadros em razão da taxa de falha de *links*. No entanto, diferentemente do estágio anterior que utiliza uma rede *IEEE 802.11*, manter a conectividade dos nodos em redes *IEEE 802.15.4* é muito mais difícil, pois o suporte para redes móveis ainda está para ser estabelecido. Ademais, a conectividade tende a piorar com altas velocidades e uma maior quantidade de nodos móveis (ZEN et al., 2008). Devido à carência de sincronização com os coordenadores da rede *IEEE 802.15.4*, a conectividade tende a ser baixa em redes móveis, degradando a taxa de entrega e aumentando a taxa de falha de *links* entre os nodos. Especialmente em protocolos como ADHOP e DSDV, que se adaptam às mudanças de topologia visando produzir baixo *overhead* na rede. Por outro lado, protocolos como OLSR que focam no estado da conectividade entre nodos, tendem a ter menor taxa de falhas. Por este motivo, nota-se que a maioria dos protocolos produz melhores resultados que o ADHOP. Porém, como dito anteriormente, a baixa conectividade entre os nodos, representada pela concentração de feromônio, tende a melhorar a adaptabilidade do ADHOP.

A Figura 48 retrata a taxa de perda de quadros em razão à taxa de quebra de rotas na rede. Em contraste com o gráfico anterior, ADHOP e DSDV produzem menores taxas de quebra de rotas, pois são protocolos que focam em construir rotas mais confiáveis do que manter melhor conectividade entre nodos. Diferentemente, OLSR possui dificuldades em manter e disseminar informações sobre os estados de *links* para toda a rede, implicando em rotas menos confiáveis no momento de transmissão dos dados.

A Figura 49 retrata a taxa de entrega de dados neste ambiente de simulação. Como um ambiente suscetível a falhas de *links* e com uma alta taxa de perda de pacotes, a taxa de entrega tende a diminuir

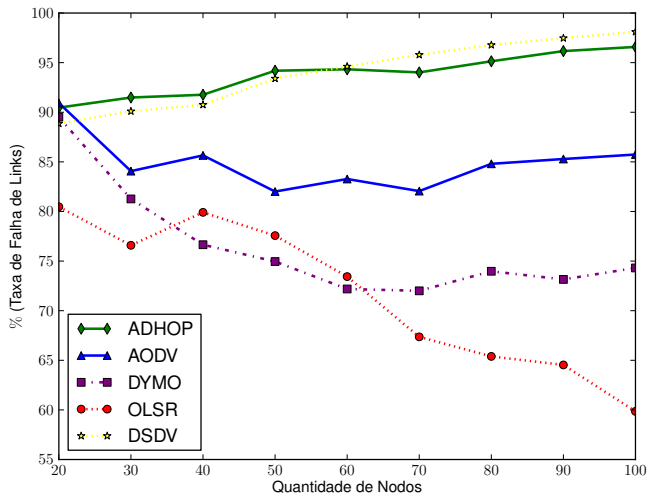


Figura 47: Taxa de falha de links.

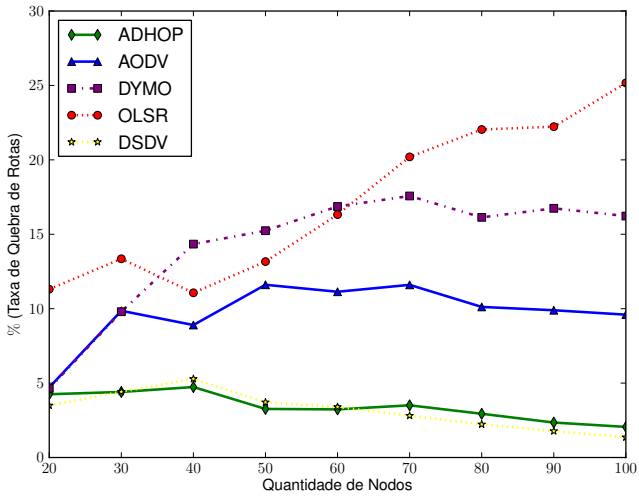


Figura 48: Taxa de quebra de rotas.

ao passo que a quantidade de nodos móveis e o *overhead* de roteamento aumentam na rede (ver Figuras 46 e 47). Como dito anteriormente, para RSSFs com contínuas mudanças de topologia na rede, a minimização do *overhead* do roteamento é a melhor maneira de lidar com o problema. ADHOP segue tal conceito, e os resultados podem ser observados na figura em questão. Conforme a quantidade de nodos aumenta, a taxa de entrega do ADHOP também aumenta. Se a rede é muito grande e densa, a taxa de entrega aumenta devido à grande quantidade de opções que a rede oferece e que as formigas descobrem, aumentando a chance dos dados serem redirecionados para rotas mais seguras. Por outro lado, em redes esparsas, a taxa de entrega diminui devido à baixa conectividade física entre os nodos. Nota-se a superioridade do ADHOP em redes escaláveis e densas. A figura também apresenta a taxa máxima de entrega do AODV, considerando um ambiente sem erros de roteamento, isto é, com a taxa de quebra de rotas igual a zero. Tal informação demonstra as limitações que redes *IEEE 802.15.4* possuem quando são aplicadas em ambientes de mobilidade.

5.2.2 Avaliação - Heurísticas

Diferentemente dos cenários bases estabelecidos para este estágio de avaliação, cada simulação é executada por 900 segundos, com a quantidade de nodos variando de 20 a 200. As heurísticas utilizadas neste estágio são as mesmas especificadas na Seção 5.1.2 do estágio anterior. Ou seja, as abordagens utilizadas são: balanceamento do consumo de energia e baixa latência fim-a-fim dos dados. A tabela 6 apresenta os parâmetros alterados e os parâmetros de energia utilizados no experimento com o OMNeT++. Nesta avaliação, o ADHOP também será comparado ao AOER, um algoritmo eficiente no consumo e no balanceamento de energia (SHUANG; LI; CHEN, 2009). Na heurística de baixa latência, o ADHOP utiliza ϕ igual a percentagem da latência da rota atual em relação à rota mais rápida até determinado destino e ρ igual a 0,5. Na heurística de balanceamento de energia, o ADHOP utiliza ϕ igual a percentagem de energia residual do nodo vizinho e ρ igual a percentagem de energia residual do próprio nodo.

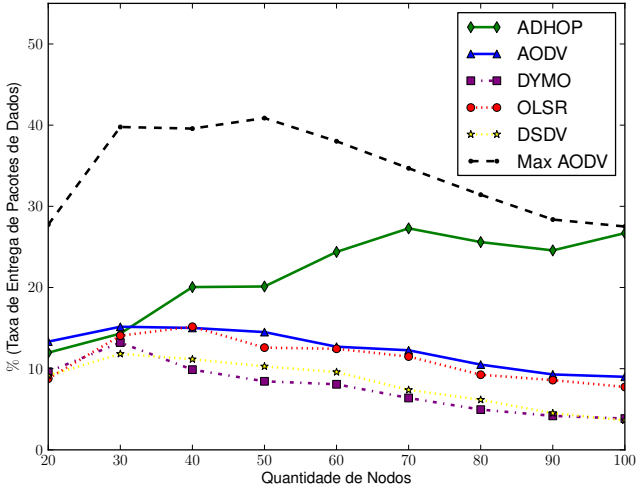


Figura 49: Taxa de entrega de pacotes de dados.

Tabela 6: Configuração energética do OMNeT++.

Parâmetros de energia	
Tempo de simulação	900s
Quantidade de nodos	20 ~ 200
Capacidade da bateria	850mAh
Voltagem da bateria	3V
Consumo do rádio - dormindo	0.001mA
Consumo do rádio - recebendo	22mA
Consumo do rádio - transmitindo	29mA

A Figura 50 retrata o desvio padrão do consumo de energia dos nodos após 900 segundos de intensa transmissão de dados. A abordagem de balanceamento de energia mantém o desvio padrão menor em todos os cenários de simulação. Em média, tal abordagem conseguiu melhorar o balanceamento de energia em torno de 23% em relação à abordagem de baixa latência. Além disso, o ADHOP consciente de energia tende a manter um balanceamento de energia melhor que o AOER, que melhora o balanceamento de energia em torno de 20% em relação à abordagem de baixa latência. Portanto, como estabelecido pela heurística de balanceamento de energia, o ADHOP consegue balancear o consumo de energia na rede.

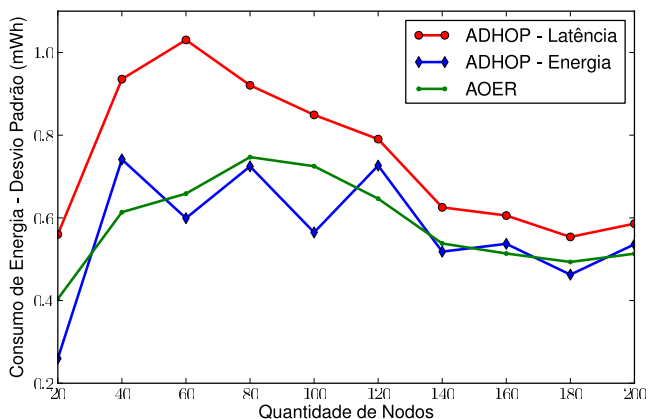


Figura 50: Desvio padrão do consumo de energia.

A Figura 51 retrata o consumo médio de energia por nodo. Embora o ADHOP consciente de energia tenha melhor balanceamento de energia, o consumo médio alcança o máximo de 25% de melhora em determinado cenário, enquanto a média do consumo médio de energia produz uma melhora de 6% em relação à abordagem da baixa latência. Nota-se que o consumo médio tende a aumentar conforme a quantidade de nodos aumenta na rede. Considerando que a quantidade de dados gerados durante o experimento é o mesmo para todos

os cenários, a figura deveria apresentar o consumo médio de energia na decrescente. Porém, a figura apresenta o que realmente é esperado em uma rede *IEEE 802.15.4*. Comparando com a Figura 44, que apresenta o consumo médio na decrescente e utiliza à rede *IEEE 802.11*, conclui-se que o consumo em redes *IEEE 802.15.4* tende a ser maior conforme a quantidade de nodos aumenta, devido à grande diferença na taxa de *bits* entre ambos os padrões. A diferença é **8 vezes maior** em redes *IEEE 802.11* (ver Tabelas 3 e 5), isto é, cada nodo *IEEE 802.15.4* tende a ocupar o canal 8 vezes mais tempo. Isto significa uma maior disputa pelo meio em redes escaláveis e densas, implicando em colisões, retransmissões, congestionamento, perda de dados e um maior consumo de energia. Embora o ADHOP consciente de energia possua melhor balanceamento de energia que o AOER, o consumo médio de energia tende a ser pior que o AOER. O AOER melhora o consumo médio em torno de 18% em relação ao ADHOP de baixa latência. O motivo desta grande diferença consiste no modo que o AOER lida com o consumo de energia da rede. O AOER usa a energia residual média de uma rota, a energia residual mínima na rota e a média da energia residual da rede para combinar e avaliar a aptidão das rotas. Se a aptidão de uma rota é maior do que o limiar estabelecido, então a rota será desativada e um novo procedimento de descoberta de rota será acionado caso necessário. Com isso, o AOER mantém o consumo médio de energia da rede bem inferior ao ADHOP.

A Figura 52 retrata a taxa de entrega de dados neste ambiente de simulação. Nota-se que a abordagem de balancear o consumo de energia gerou melhores resultados. Isto ocorre devido ao “efeito colateral” desta abordagem, em que, ao tentar balancear o consumo de energia, o algoritmo também balanceou a carga de dados trafegados pela rede. Ou seja, os dados foram distribuídos entre diversas rotas, e com isso, situações de colisões e congestionamento foram evitadas ou “suavizadas”. Em contraste com o ADHOP, a taxa de entrega do AOER tende a ser muito inferior. O procedimento de economia do consumo de energia do AOER é eficiente, mas tal procedimento afeta a taxa de entrega negativamente. A fim de reduzir o consumo de energia e o uso de rotas com pouca energia residual, o AOER “sacrifica” a taxa de entrega, como apresentado na Figura 52.

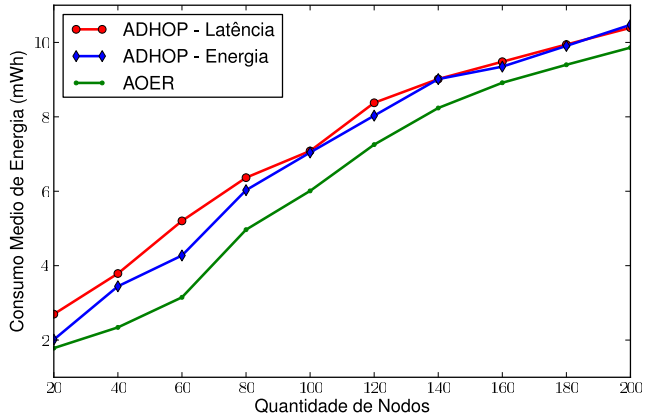


Figura 51: Consumo médio de energia.

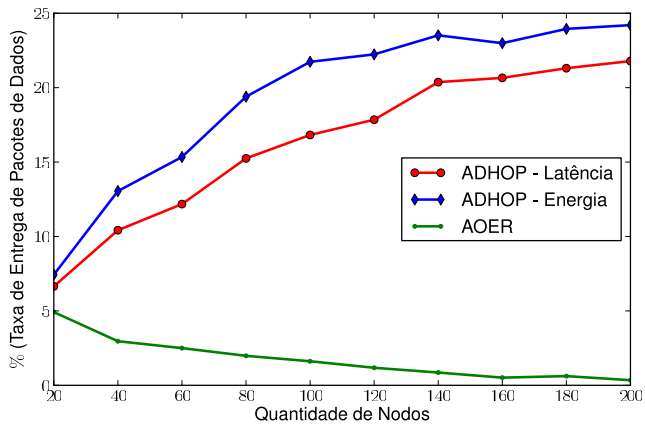


Figura 52: Taxa de entrega de pacotes de dados.

6 CONCLUSÃO

Esta dissertação apresentou o ADHOP, um algoritmo de roteamento inspirado pela inteligência coletiva das formigas para RSSFs móveis que utiliza um método de roteamento baseado em *saltos dinâmicos*. Como um algoritmo baseado em ACO, o ADHOP utiliza feromônio como métrica para as tomadas de decisões, e permite a utilização de heurísticas no cálculo da taxa de depósito e evaporação de feromônio. O objetivo é validar o uso de informações heurísticas de modo a tornar o ADHOP um protocolo flexível o bastante para se adaptar aos diferentes contextos que uma rede sensor pode ter. No entanto, esta dissertação apresentou o AD-ZRP como primeira proposta, cuja maior contribuição é o baixo *overhead* de roteamento. Porém, o AD-ZRP carece de mecanismos mais simples na manutenção de feromônio (uso de heurísticas) a fim de obter uma maior taxa de entrega de dados e uma melhor implantação em RSSFs. Em consequência disso, o ADHOP foi desenvolvido como uma evolução do AD-ZRP.

O ADHOP foi avaliado e comparado com o AD-ZRP, bem como outros protocolos bem conhecidos para redes *ad hoc*, tais como AODV, OLSR e DSDV. O ADHOP se mostrou superior em termos de taxa de entrega, *overhead* de roteamento e congestionamento. Principalmente em termos de *overhead* de roteamento, onde ambos o AD-ZRP e o ADHOP visam menor um tráfego de dados na rede. Concluímos que, em redes propensas a erros e altamente móveis, protocolos de roteamento reativos que não são conscientes do fluxo de dados da rede tendem a ser piores em termos de *overhead* até mesmo para protocolos pró-ativos. Mas também se conclui que apesar de muitas das características de MANETs, o ADHOP é capaz de ser utilizado em RSSFs móveis, sempre visando o uso eficiente dos recursos disponíveis.

A redução do *overhead* de roteamento deve ser o primeiro curso de ação para lidar com redes móveis (VLAJIC; STEVANOVIC, 2009). O ADHOP segue esta concepção desde o início do projeto, utilizando o método de saltos dinâmicos que tende a gerar um menor *overhead* na

rede devido à técnica de otimização com feromônio. Na verdade, o ADHOP também pode ser considerado um protocolo de fluxo de rede pela eficiência que lida com congestionamentos. O algoritmo proposto se baseia muito nos trabalhos do AOER e FROMS, algoritmos que visam à simplicidade nas estruturas de dados e no roteamento, bem como mecanismos que visam um baixo *overhead* de roteamento. Com isso, a abordagem proposta consegue gerar bons e interessantes resultados na taxa de entrega também. O método de saltos dinâmicos não apenas assegura simplicidade no roteamento, mas também nas estruturas de dados, tais como a tabela de roteamento e a própria formiga. Tal simplicidade pode ser observada na descrição do algoritmo e nos resultados obtidos em ambos os estágios de simulações.

Com isso, conclui-se que o ADHOP é um protocolo eficiente em ambientes cuja mobilidade dos nodos é alta e imprevisível. Os bons resultados obtidos sob a heurística de energia e a comparação com o algoritmo de roteamento AOER contribuem para a eficiência do algoritmo em RSSFs móveis, implicando em diversos cenários de aplicações e casos de estudo que necessitam de um esquema de roteamento eficiente para redes de topologia dinâmica. Como dito anteriormente, sensores podem ser anexados a veículos ou animais (domésticos, silvestres ou robôs autônomos), portanto, aplicações que possuem a finalidade de exploração de ambientes de difícil acesso humano são os principais cenários de aplicações imaginados. Animais, robóticos ou não, e veículos coletando informações ou até mesmo atuando no ambiente é um sonho que não está muito longe de ocorrer, como se pode observar no projeto desenvolvido em Esslingen (FESTO, 2012). Acompanhamento, tratamento e/ou manutenção de poluição em rios, lagos e mares é uma opção de sensoriamento móvel e intermitente que possibilita aplicações mais simples, e que pode ser realizado até mesmo através de boias aquáticas.

O protocolo do ADHOP está implementado no *Sistema Operacional Paralelo Embarcado* (do inglês *Embedded Parallel Operating System* - EPOS) (FRÖHLICH, 2001). O algoritmo trabalha em torno da segunda geração de nodos sensores do EPOS, o EPOSMoteII, que possui processador ARM e transceptor de rádio IEEE 802.15.4 (EPOS-Mote, 2012).

As próximas etapas de desenvolvimento do ADHOP deverão permitir o uso mais eficiente das informações heurísticas e o refinamento das equações para o uso de energia e outras heurísticas probabilísticas. Os cálculos e equações para a geração destas informações tendem a ser realizadas de forma empírica. Um gerador automático de equações para tais redes, considerando o tipo de heurística utilizada, pode gerar boas contribuições para o algoritmo. Outra ideia é a associação do ADHOP com algum protocolo de MAC a fim de compartilhar informações e melhorar a taxa de falha de *links* da rede. Uma integração entre o ADHOP e o *MAC-Configurável* (do inglês *Configurable-MAC* - C-MAC) está em desenvolvimento (STEINER; MÜCK; FRÖHLICH, 2010). No entanto, ADHOP ainda permite uma infinidade de pesquisa para ser realizada no algoritmo: explorar outras heurísticas e as diversas combinações delas, avaliar o comportamento juntamente com as heurísticas em redes estáticas, incluir módulos de segurança e qualidade de serviço e utilizar o algoritmo em uma aplicação de rede escalável, móvel e real.

REFERÊNCIAS

- AKKAYA, K.; YOUNIS, M. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, v. 3, n. 3, p. 325 – 349, 2005. ISSN 1570-8705.
- ANASTASI, G. et al. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Networks*, v. 7, n. 3, p. 537 – 568, 2009. ISSN 1570-8705.
- ARAÚJO, G. de; BECKER, L. A network conditions aware geographical forwarding protocol for real-time applications in mobile wireless sensor networks. In: *Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on*. [S.l.: s.n.], 2011. p. 38 –45. ISSN 1550-445X.
- CARO, G. D. *Ant Colony Optimization and its application to adaptive routing in telecommunication networks*. Tese (Doutorado) — Faculté des Sciences Appliquées, Université Libre de Bruxelles, Brussels, Belgium, November 2004.
- CARO, G. D. *Ant Colony Optimization and its application to adaptive routing in telecommunication networks*. Tese (Doutorado) — Faculté des Sciences Appliquées, Université Libre de Bruxelles, Brussels, Belgium, November 2004.
- CARO, G. D.; DUCATELLE, F.; GAMBARDELLA, L. M. Anthocnet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks. *European Transactions on Telecommunications*, John Wiley & Sons, v. 16, n. 5, p. 443–455, 2005. ISSN 1541-8251. Disponível em: <<http://dx.doi.org/10.1002/ett.1062>>.
- CHAKERES, I. D.; PERKINS, C. E. *Dynamic MANET On-demand (DYMO) Routing*. mar. 2009. IETF Internet Draft.

- CHOI, D.; MOH, S.; CHUNG, I. Variable area routing protocol in wsns: A hybrid, energy-efficient approach. In: . [S.l.: s.n.], 2008. p. 397–403.
- CLAUSEN, T.; JACQUET, P. *Optimized Link State Routing Protocol (OLSR)*. 2003. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.11.620>>.
- DEY, A. K.; ABOWD, G. D. Towards a better understanding of context and context-awareness. In: *In HUC '99: Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing*. [S.l.]: Springer-Verlag, 1999. p. 304–307.
- DHURANDHER, S. et al. Ant colony optimization-based congestion control in ad-hoc wireless sensor networks. In: . [S.l.: s.n.], 2009. p. 492–497.
- DORIGO, M.; BIRATTARI, M.; STUTZLE, T. Ant colony optimization. *Computational Intelligence Magazine, IEEE*, v. 1, n. 4, p. 28–39, nov. 2006. ISSN 1556-603X.
- DORIGO, M.; BLUM, C. Ant colony optimization theory: A survey. *Theoretical Computer Science*, v. 344, n. 2-3, p. 243–278, 2005. ISSN 0304-3975.
- EPOSMote. *EPOSMote II Project*. 2012. <http://epos.lisha.ufsc.br/EPOSMote+II>.
- FESTO, C. *Bionic Learning Network*. 2012. <http://www.festo.com>.
- FÖRSTER, A.; MURPHY, A. Froms: Feedback routing for optimizing multiple sinks in wsn with reinforcement learning. In: . [S.l.: s.n.], 2007. p. 371–376.
- FÖRSTER, A. et al. An efficient implementation of reinforcement learning based routing on real wsn hardware. In: *WIMOB '08: Proceedings of the 2008 IEEE International Conference on Wireless & Mobile Computing, Networking & Communication*. Washington, DC, USA: IEEE Computer Society, 2008. p. 247–252. ISBN 978-0-7695-3393-3.

- FRÖHLICH, A. A. *Application-Oriented Operating Systems*. Sankt Augustin: GMD - Forschungszentrum Informationstechnik, 2001. (GMD Research Series, 17).
- GANESAN, D. et al. Networking issues in wireless sensor networks. *J. Parallel Distrib. Comput.*, Academic Press, Inc., Orlando, FL, USA, v. 64, n. 7, p. 799–814, 2004. ISSN 0743-7315.
- GARCIA-MACIAS, J.; GOMEZ, J. Manet versus wsn. In: MAHALIK, N. P. (Ed.). *Sensor Networks and Configuration*. [S.l.]: Springer Berlin Heidelberg, 2007. p. 369–388. ISBN 978-3-540-37366-7.
- HAAS, Z. A new routing protocol for the reconfigurable wireless networks. In: . [S.l.: s.n.], 1997. v. 2, p. 562–566 vol.2.
- HEINZELMAN, W.; CHANDRAKASAN, A.; BALAKRISHNAN, H. Energy-efficient communication protocol for wireless microsensor networks. In: . [S.l.: s.n.], 2000. v. 2, p. 10 pp.
- IYENGAR, S. et al. Biologically inspired cooperative routing for wireless mobile sensor networks. *Systems Journal, IEEE*, v. 1, n. 1, p. 29–37, sep. 2007. ISSN 1932-8184.
- JACOBSON, V. et al. Networking named content. In: *Proceedings of the 5th international conference on Emerging networking experiments and technologies*. New York, NY, USA: ACM, 2009. (CoNEXT '09), p. 1–12. ISBN 978-1-60558-636-6.
- KUMAR, N.; KUMAR, M.; PATEL, R. Coverage and Connectivity Aware Neural Network Based Energy Efficient Routing in Wireless Sensor Networks. *ArXiv e-prints*, Mar 2010.
- LI, Z.; SHI, H. A data-aggregation algorithm based on adaptive ant colony system in wireless sensor networks. In: . [S.l.: s.n.], 2008. v. 4, p. 449–453.
- LIU, C. A survey of mobile ad hoc network routing protocols. *Analysis*, Citeseer, v. 1, n. 2003, p. 125–136, 2005.

- LU, Y.; ZHAO, G.; SU, F. Adaptive ant-based dynamic routing algorithm. In: . [S.l.: s.n.], 2004. v. 3, p. 2694 – 2697 Vol.3.
- MARRÓN, P. J.; MINDER, D. *Embedded WiSeNts Research Roadmap*. [S.l.]: Logos Verlag Berlin, 2006. ISBN 3-8325-1424-4.
- MATROUK, K.; LANDFELDT, B. Rett-gen: A globally efficient routing protocol for wireless sensor networks by equalising sensor energy and avoiding energy holes. *Ad Hoc Networks*, v. 7, n. 3, p. 514–536, 2009. ISSN 1570-8705.
- OKAZAKI, A. M.; FROHLICH, A. A. Ad-zrp: Ant-based routing algorithm for dynamic wireless sensor networks. In: *Telecommunications (ICT), 2011 18th International Conference on*. [S.l.: s.n.], 2011. p. 15 –20.
- OKAZAKI, A. M.; FROHLICH, A. A. Ant-based dynamic hop optimization protocol: A routing algorithm for mobile wireless sensor networks. In: *GLOBECOM Workshops (GC Wkshps), 2011 IEEE*. [S.l.: s.n.], 2011. p. 1139 –1143. ISBN 978-1-4673-0039-1.
- OKAZAKI, A. M.; FRÖHLICH, A. A. Adapting hopnet algorithm for wireless sensor networks. In: . Florianópolis, Brazil: [s.n.], 2009. p. 191–194. ISBN 978-85-89264-10-5.
- OKDEM, S.; KARABOGA, D. Routing in wireless sensor networks using an ant colony optimization (aco) router chip. *Sensors*, v. 9, n. 2, p. 909–921, 2009. ISSN 1424-8220.
- PERKINS, C.; ROYER, E. Ad-hoc on-demand distance vector routing. In: *Proceedings. Second IEEE Workshop on Mobile Computing Systems and Applications*. [S.l.: s.n.], 1999. p. 90–100.
- PERKINS, C. E.; BHAGWAT, P. Highly dynamic destination-sequenced distance-vector routing (dsv) for mobile computers. *SIGCOMM Comput. Commun. Rev.*, ACM, New York, NY, USA, v. 24, p. 234–244, October 1994. ISSN 0146-4833.

- RAO, A. et al. Geographic routing without location information. In: *Proceedings of the 9th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2003. (MobiCom '03), p. 96–108. ISBN 1-58113-753-2.
- SHELBY, Z.; BORMANN, C. *6LoWPAN: The Wireless Embedded Internet*. [S.l.]: Wiley Publishing, 2010. ISBN 0470747994, 9780470747995.
- SHUANG, B.; LI, Z.; CHEN, J. An ant-based on-demand energy route protocol for ieee 802.15.4 mesh network. *International Journal of Wireless Information Networks*, Springer Netherlands, v. 16, p. 225–236, 2009. ISSN 1068-9605. 10.1007/s10776-009-0110-4.
- SONG, Y.; LIU, M.; WANG, Y. Power-aware traffic engineering with named data networking. In: *Mobile Ad-hoc and Sensor Networks (MSN), 2011 Seventh International Conference on*. [S.l.: s.n.], 2011. p. 289–296.
- STANKOVIC, J. Wireless sensor networks. *Computer*, v. 41, n. 10, p. 92–95, oct. 2008. ISSN 0018-9162.
- STEINER, R.; MÜCK, T. R.; FRÖHLICH, A. A. C-MAC: a Configurable Medium Access Control Protocol for Sensor Networks. In: *9th IEEE Sensors*. Waikoloa, HI, USA: [s.n.], 2010. p. 845–848. ISBN 978-1-4244-8168-2.
- TANENBAUM, A. *Computer Networks*. 4th. ed. [S.l.]: Prentice Hall Professional Technical Reference, 2002. ISBN 0130661023.
- VLAJIC, N.; STEVANOVIC, D. Performance analysis of zigbee-based wireless sensor networks with path-constrained mobile sink(s). In: . [S.l.: s.n.], 2009. p. 61–68.
- WANG, J. et al. Hopnet: A hybrid ant colony optimization routing algorithm for mobile ad hoc network. *Ad Hoc Networks*, v. 7, n. 4, p. 690–705, 2009. ISSN 1570-8705.
- WANG, Y.-C.; WU, F.-J.; TSENG, Y.-C. Mobility management algorithms and applications for mobile sensor networks. *Wireless*

Communications and Mobile Computing, John Wiley & Sons, Department of Computer Science, National Chiao-Tung University, Hsin-Chu 30010, Taiwan, ROC, Jan. 2010.

- WATTEYNE, T. et al. From manet to ietf roll standardization: A paradigm shift in wsn routing protocols. *Communications Surveys Tutorials, IEEE*, PP, n. 99, p. 1–20, 2010. ISSN 1553-877X.
- YU, Y.; GOVINDAN, R.; ESTRIN, D. *Geographical and Energy Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks*. [S.l.], 2001.
- ZEN, K. et al. Performance evaluation of ieee 802.15.4 for mobile sensor networks. In: *Wireless and Optical Communications Networks, 2008. WOCN '08. 5th IFIP International Conference on*. [S.l.: s.n.], 2008. p. 1–5.
- ZENG, K. et al. Energy aware efficient geographic routing in lossy wireless sensor networks with environmental energy supply. *Wirel. Netw.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 15, n. 1, p. 39–51, 2009. ISSN 1022-0038.