

UNIVERSIDADE FEDERAL DE SANTA CATARINA

**PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE AUTOMAÇÃO E SISTEMAS**

**SOLUÇÕES PARA REDES DE SENSORES SEM FIO
COM MOBILIDADE:
PROTOCOLO DE ROTEAMENTO COM PRIORIZAÇÃO
DE MENSAGENS E MECANISMO DE PREDIÇÃO DE
CONECTIVIDADE LOCAL**

Gustavo Medeiros de Araújo

Florianópolis, 16 de abril de 2013.

UNIVERSIDADE FEDERAL DE SANTA CATARINA

**PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE AUTOMAÇÃO E SISTEMAS**

**SOLUÇÕES PARA REDES DE SENSORES SEM FIO
COM MOBILIDADE:
PROTOCOLO DE ROTEAMENTO COM PRIORIZAÇÃO
DE MENSAGENS E MECANISMO DE PREDIÇÃO DE
CONECTIVIDADE LOCAL**

Gustavo Medeiros de Araújo

Florianópolis, 16 de abril de 2013.

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Medeiros de Araújo, Gustavo
SOLUÇÕES PARA REDES DE SENSORES SEM FIO COM MOBILIDADE:
PROTOCOLO DE ROTEAMENTO COM PRIORIZAÇÃO DE MENSAGENS E
MECANISMO DE PREDIÇÃO DE CONECTIVIDADE LOCAL / Gustavo
Medeiros de Araújo ; orientador, Leandro Buss Becker -
Florianópolis, SC, 2013.
199 p.

Tese (doutorado) - Universidade Federal de Santa
Catarina, Centro Tecnológico. Programa de Pós-Graduação em
Engenharia de Automação e Sistemas.

Inclui referências

1. Engenharia de Automação e Sistemas. 2. Redes de
Sensores Sem Fio. 3. Mobilidade. 4. Roteamento. 5.
Conectividade. I. Buss Becker, Leandro. II. Universidade
Federal de Santa Catarina. Programa de Pós-Graduação em
Engenharia de Automação e Sistemas. III. Título.

**PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE AUTOMAÇÃO E SISTEMAS**

Gustavo Medeiros de Araújo

'Esta Tese foi julgada adequada para obtenção do Título de Doutor em
Engenharia de Automação e Sistemas,
Área de Concentração em *Sistemas Computacionais*,
e aprovada em sua forma final pelo Programa de Pós-Graduação em
Engenharia de Automação e Sistemas da
Universidade Federal de Santa Catarina.'

Prof. Leandro Buss Becker, Dr.
Orientador

Prof. Jomi Fred Hübner, Ph.D.
Coordenador do Programa de Pós-Graduação em Engenharia de
Automação e Sistemas

Banca Examinadora:

Prof. Leandro Buss Becker, Dr.
Presidente

Prof. Flávio Moraes de Assis Silva , Dr.

Prof. Eraldo Silveira e Silva, Dr.

Prof. Frank Augusto Siqueira, Dr.

Prof. Marcelo Maia Sobral, Dr.

Prof. Jean Marie Alexandre Farines, Dr.

Dedico esta Tese aos meus pais, Gilvan Cordeiro de Araújo e Maria Cleonildes Borges Medeiros, a minha esposa Tula Beck Bisol, ao meu cachorro Fred e a todos os meus irmãos e amigos.

Agradecimentos

À Deus,
aos meus Pais
Irmãos
Esposa
Amigos do Laboratório EOS da Universidade Otto-von-Guericke de
Magdeburg, Alemanha
ao meu Cachorro Fred.

Aos Professores Leandro Buss Becker, Jörg Kaiser e todos os
professores do PGEAS que contribuíram para o meu crescimento pro-
fissional.

Resumo da Tese submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Doutor em Engenharia de Automação e Sistemas.

**SOLUÇÕES PARA REDES DE SENSORES SEM FIO
COM MOBILIDADE:
PROTOCOLO DE ROTEAMENTO COM
DIFERENCIAÇÃO DE SERVIÇO E MECANISMO DE
PREDIÇÃO DE CONECTIVIDADE LOCAL**

Gustavo Medeiros de Araújo

Fevereiro/2013

Orientador: Prof. Leandro Buss Becker, Dr.

Área de concentração: Sistemas Computacionais.

Palavras-chave: Redes de Sensores Sem Fio, Mobilidade, Diferenciação de Serviço, Roteamento, Conectividade.

RESUMO: As redes de sensores sem fio (RSSF) proporcionam uma ampla gama de aplicações em diversos setores da sociedade. A comunicação entre os nodos sensores é o elemento básico do funcionamento das redes de sensores. Por isso, o trabalho desta tese se concentra em duas questões essenciais para aperfeiçoamento de protocolos de rede. Primeiramente, uma arquitetura para protocolos de roteamento com garantia de qualidade de serviço foi proposta no escopo de cenários com mobilidade. O protocolo proposto utiliza métricas para tomada de decisão para realizar retransmissão de mensagens que refletem as condições da rede, sejam redes densas ou esparsas, com grande ou pouca carga de mensagem. A proposta do protocolo de comunicação foi avaliada sob a ótica de um cenário de aplicação envolvendo mobilidade e diferentes cargas de mensagens. Além disso, as métricas foram de-

sagregadas e analisadas para verificar a sua influência para diferentes condições de rede. Em conjunto à análise das métricas, diversos trabalhos da literatura foram analisados sob os mesmos aspectos e condições de rede para melhor avaliar o ganho do protocolo proposto, o qual se mostrou hábil para o atendimento de requisitos de qualidade de serviço demandado pela aplicação. O segundo aspecto igualmente importante para o desempenho das redes de sensores trata-se do aspecto fundamental para realização da comunicação em ambientes com mobilidade, que é a conectividade. A conectividade foi estudada na sua característica essencial que se refere ao enlace entre dois nodos móveis. Dois modelos foram projetados para proporcionar a estimativa de conectividade por meio da qualidade de enlace para protocolos e aplicações. O primeiro modelo proposto se baseia no comportamento estatístico dos padrões de mobilidade para realizar a estimativa da qualidade de enlace. O segundo modelo proposto se baseia no método de aprendizado sistema classificador para aprender o comportamento do padrão de mobilidade. Ambos os modelos foram implementados e testados com diferentes padrões de mobilidade. Ademais, alguns trabalhos da literatura também foram implementados, avaliados e comparados com os modelos propostos. O método de aprendizado com sistema classificador se mostrou eficiente na estimação de conectividade, bem como capaz de se adaptar a mudanças no padrão de mobilidade.

Abstract of Thesis presented to UFSC as a partial fulfillment of the requirements for the degree of Doctor in System and Automation Engineering.

**SOLUTIONS FOR WIRELESS SENSOR NETWORKS
WITH MOBILITY:
ROUTING PROTOCOL WITH DIFFSERV AND
MECHANISM FOR PREDICTION OF LOCAL
CONNECTIVITY**

Gustavo Medeiros de Araújo

February/2013

Advisor: Prof. Leandro Buss Becker, Dr.

Area of Concentrations: Computer Systems.

Keywords: Wireless Sensor Network, Mobility, Quality of Service, Routing, Connectivity.

ABSTRACT: Wireless sensor networks (WSN) provide a wide range of applications in various sectors of society. The communication among the sensor nodes is the basic element of sensor networks function. Given this premise, this thesis focuses on two key issues for the improvement of network protocols. Firstly, an protocol for routing protocols with guaranteed quality of service was proposed to deal with mobility scenarios. The proposed protocol uses several metrics that reflect the network conditions in dense or sparse networks with small or large load of messages for routing decisions. The proposed communication protocol was evaluated from the perspective of an application scenario involving mobility and different loads of messages. Furthermore, the metrics were disaggregated and analyzed to check their influence with different network conditions. Moreover, several algorithms from the

literature were analyzed under the same aspects and network conditions to better assess the gain of the proposed protocol. The proposed protocol proved to be able to fulfill the quality of service requirements demanded by the application. The second equally important aspect for the performance of WSN is the fundamental aspect for achieving communication in environments with mobility: the connectivity. Two models were designed to provide connectivity estimation by means of link quality to improve protocols and applications. The first proposed model is based on statistical behavior mobility patterns to perform the estimation of link quality. The second proposed model is based on a learning method classifier system to learn the behavior of the mobility pattern. Both models were implemented and tested with different mobility patterns. Furthermore, other existing approaches have also been implemented, evaluated, and contrasted against the proposed models. The learning method with classifier system is efficient for estimation of connectivity, as well as able to adapt to changes in mobility pattern.

Sumário

Lista de Figuras	xii
Lista de Tabelas	xvi
Lista de Abreviaturas e Siglas	xvii
I Introdução	1
1 Introdução	2
1.1 Motivações	7
1.1.1 Provimento de Priorização de Mensagens por Protocolo de Roteamento em Ambientes Móveis	7

1.1.2	Construção do Conhecimento de Conectividade Local pela Predição de Qualidade de Enlace em Ambientes Móveis	9
1.1.3	Problemas Abordados nesta Tese	10
1.2	Objetivos	12
1.3	Metodologia	13
1.4	Modelo do Sistema	14
1.5	Estrutura do Trabalho	15

II PROTOCOLO DE ROTEAMENTO COM PRIORIZAÇÃO DE MENSAGENS **17**

2 Trabalhos Relacionados a Protocolos de Roteamento para Redes *Ad Hoc* **18**

2.1	Protocolos de Roteamento Geográfico para RSSF . . .	20
2.1.1	<i>RAP: A Real-Time Communication Architecture for Large-Scale Wireless Sensor Network</i> .	22
2.1.2	<i>SPEED: A Stateless Protocol for Real-Time Communication In Sensor Networks</i>	25
2.1.3	<i>MMSPEED: Multipath Multi-SPEED Protocol for QoS Guarantee of Reliability and Timeliness in Wireless Sensor Networks</i>	27

2.1.4	<i>RPAR: Real-time Power-Aware Routing in Sensor Networks</i>	28
2.1.5	<i>PATH: A Novel Real-Time Power Aware Routing Protocol in Wireless Sensor Networks</i>	31
2.1.6	<i>TAAE: A Traffic-Aware Energy Efficient Routing Protocol for Wireless Sensor Networks</i>	32
2.1.7	Resumo	33
2.2	Protocolos de Roteamento para MANETs	34
2.2.1	<i>Destination-Sequenced Distance-Vector</i>	35
2.2.2	<i>Optimized Link State Routing</i>	36
2.2.3	<i>Better Approach To Mobile Adhoc Networking</i>	38
2.2.4	<i>Ad hoc On-Demand Distance Vector</i>	39
2.2.5	<i>Dynamic Source Routing</i>	40
2.2.6	<i>Dynamic MANET On-demand</i>	41
2.3	Considerações Finais	42
3	Protocolo de Roteamento Proposto	45
3.1	Introdução e Apresentação do Protocolo RACE	45
3.2	Modelo do Sistema	46
3.3	O Protocolo RACE	47

3.4	Arquitetura <i>Crosslayer</i> do Protocolo	50
3.4.1	Componente Gerenciador da Tabela de Vizinhança	51
3.4.2	Componente de Decisão de Roteamento	53
3.4.3	Componente de Ciência de Congestionamento	55
3.4.4	Estimador de Qualidade de Enlace	59
3.4.5	Estimador de Velocidade	60
3.4.6	Power Switch	62
3.4.7	Diferenciação de Serviço	63
4	Avaliação do Protocolo Proposto	65
4.1	Modelo de Rede	66
4.2	Descrição dos Experimentos	67
4.2.1	Primeiro Conjunto de Experimentos: Avaliações dos Protocolos RACE e RPAR	70
4.2.2	Segundo Conjunto de Experimentos: O Impacto da Métrica <i>Buffer</i>	78
4.2.3	Terceiro Conjunto de Experimentos: Análise de Desempenho dos Protocolos MANETs no Cenário da Corrida	83
4.2.4	Conclusão	87

III MECANISMO DE PREDIÇÃO DE CONECTIVIDADE LOCAL 90

5 Trabalhos Relacionados a Mecanismos de Predição de Conectividade para Ambientes Móveis 91

5.1	Considerações Iniciais	92
5.2	<i>Mobility Prediction in Wireless Networks</i>	95
5.3	<i>Disconnection prediction in mobile ad hoc networks for supporting cooperative work</i>	96
5.4	<i>Effective Link Operation Duration: a New Routing Metric for Mobile Ad Hoc Networks</i>	97
5.5	<i>Breadcrumbs: Forecasting mobile connectivity</i>	98
5.6	<i>Link quality prediction in mesh networks</i>	99
5.7	<i>Performance Enhancement of Fuzzy Logic Based Transmission Power Control in WSN using Markov Based RSSI Prediction</i>	101
5.8	<i>BD Model</i>	103
5.9	Considerações Finais	107

6 Predição de Conectividade Local 109

6.1	Modelo do Sistema	109
6.2	Introdução	110

6.3	<i>Oriented Birth-Death</i>	111
6.3.1	Autômato de Estados Finitos do modelo O-BD	114
6.4	<i>Genetic Machine Learning Approach for Link Quality Prediction</i>	117
6.4.1	Sistemas Classificadores	117
6.4.2	Funcionamento do GMLA	120
7	Avaliação dos Mecanismos Propostos para Predição de Co- nectividade Local	127
7.1	Análise Individual dos Mecanismos O-DB e GMLA . .	128
7.1.1	Modelos de Mobilidade	129
7.1.2	Conclusões Parciais	142
7.2	Aplicações do GMLA em Algoritmo de Roteamento MANET	144
7.2.1	Visão Geral das Mensagens do Protocolo AODV	145
7.2.2	AODV Aprimorado com GMLA	146
7.3	Estudo da Simulação	148
7.3.1	Configurações da Simulação	150
7.3.2	Resultados Obtidos	151
7.3.3	Considerações Finais	153

IV	Conclusões e Trabalhos Futuros	156
8	Conclusões Finais e Trabalhos Futuros	157
	Referências Bibliográficas	164

Lista de Figuras

1.1	Classificação para Aplicação de Redes de Sensores Sem Fio.	3
1.2	Cenário de Corrida Onde Vários Atletas com Sensores Anexados Percorrem um Trajeto com Nodos <i>Sinks</i> Espalhados.	8
2.1	Arquitetura RAP.	25
3.1	Arquitetura do Protocolo RACE	51
3.2	Evitando Regiões Congestionadas	57
3.3	Mecanismo de Diferenciação de Serviço	63
4.1	Simulação do Cenário da Corrida. Disposição dos Nodos Sorvedouros e Fontes.	68

4.2	Taxa de Entrega de Dados: Velocidade Fixada e Carga de Mensagens Variável	74
4.3	Taxa de Entrega de Dados: Carga de Mensagens Fixada e Velocidades Variáveis	75
4.4	Taxa de Atraso com Variação de Velocidade	76
4.5	Taxa de Atraso com Variação de Carga de Mensagens .	77
4.6	Média de Energia Consumida em mW-sec	78
4.7	Redistribuição dos Nodos Fontes.	78
4.8	Taxa de Entrega de Dados Para o Nodo Sorvedouro 1 com Reconfiguração de Métricas.	82
4.9	Taxa de Entrega de Dados Para o Nodo Sorvedouro 2 com Reconfiguração de Métricas.	82
4.10	Taxa de Entrega de Dados Variando a Carga de Dados .	84
4.11	Taxa de Entrega de Dados Variando as Velocidades . .	84
4.12	Atraso fim-a-fim Variando a Carga de Dados	85
4.13	Atraso fim-a-fim Variando as Velocidades	85
5.1	Topologia de Rede Representada na Matriz de Transição de Estados.	97
5.2	Fluxo do Algoritmo Filtro de Kalman Baseado no Modelo de Auto-Regressão.	101
5.3	Matriz de Transição de Estados do Modelo MTCP. . .	102

5.4	SNR Dividido em Faixas de Valores para Determinar os Limites dos Estados.	103
5.5	Autômato representando os estados e transições do modelo <i>birth-death</i>	104
5.6	Nodos se Movimentando em Sentidos Opostos Entretanto em Contato.	105
5.7	Nodos se Movimentando em Sentidos Opostos Perdendo o Contato.	106
6.1	Variação do SNR sobre o tempo devido à mobilidade do nodo e interferência do ambiente.	110
6.2	Autômato representando os estados e transições	114
6.3	Esquema do Sistema Classificador.	118
6.4	O Cromossomo como uma Matriz e o Gene como uma Linha.	122
6.5	GMLA - Genetic Machine Learning Approach.	123
6.6	O Processo de Cruzamento do GMLA Gera Dois Novos Indivíduos.	125
6.7	O processo de Mutação do GMLA Randomicamente Escolhe Um Gene e Aplica o Algoritmo 7.	126
7.1	Percentual de Precisão para escolher o próximo estado correto em 1,5,10,15 e 20 segundos à frente para o modelo de mobilidade <i>Manhattan</i>	135

7.2	Percentual de Precisão para escolher o próximo estado correto em 1,5,10,15 e 20 segundos à frente para o modelo de mobilidade <i>Reference Point Group</i>	136
7.3	Percentual de Precisão para escolher o próximo estado correto em 1,5,10,15 e 20 segundos à frente para o modelo de mobilidade <i>Gauss-Markov</i>	137
7.4	Processo de Reaprendizado com Mudança de Padrão de Mobilidade.	141
7.5	Mensagens de Controle Normalizadas.	152
7.6	Taxa de Entrega de Dados.	152

Lista de Tabelas

2.1	Atributos Utilizados pelos Protocolos	34
4.1	Parâmetros Gerais de Simulação	69
4.2	Conjunto de Velocidades	69
4.3	Conjunto de Dados	72
4.4	Pesos das Métricas	79
4.5	Conjunto de Dados Para os Nodos Sorvedouros	81
7.1	Parâmetros Gerais do OMNET	132
7.2	Parâmetros do GMLA	133
7.3	Parâmetros de Taxa de Dados	150

Lista de Abreviaturas e Siglas

ACK - Acknowledgment

APS - Ad-Hoc Positioning System

CSMA - Carrier Sense Multiple Access

CTS - Clear To Send

GPS - Global Position System

LQE - Link Quality Estimator

LQI - Link Quality Indicator

MAC - Medium Access Control

MANET - Mobile Ad Hoc Network

MWSN - Mobile Wireless Sensor Network

PRR - Packet Reception Rate

QoL - Quality of Link

QoS - Quality of Service

RSSF - Redes de Sensores Sem Fio

RSSI - Received Signal Strength Indicator

RTS - Request To Send

RPGM - Reference Point Group Mobility

SNR - Signal To Noise Ratio

WLAN - Wireless Local Area Network

WMEWMA - Window Mean Exponentially Weighted Moving Average

WSN - Wireless Sensor Network

Parte I

Introdução

Capítulo 1

Introdução

A tecnologia de Redes de Sensores Sem Fio (RSSF) tem sido largamente estudada e aplicada. O seu surgimento se deu na área militar para fins de monitoramento de campos de batalha. Entretanto, rapidamente a RSSF foi difundida em diversas outras aplicações e se tornou objeto de estudos científicos. Há uma grande possibilidade de aplicações de RSSF, que vão desde o monitoramento de florestas, vigilância de indústrias, fronteiras ao rastreamento de condições fisiológicas de pacientes, dentre outras [Akyildiz et al. 2002].

Dada a variada gama de aplicação das RSSFs existe uma grande quantidade de requisitos que devem nortear a utilização desta tecnologia. Segundo [Mottola e Picco 2011], as RSSFs podem ser caracterizadas quanto a cinco aspectos: objetivo, padrão de interação, mobilidade, espaço e tempo. Uma visão geral dos aspectos pode ser vista na figura 1.1.



Fig. 1.1: Classificação para Aplicação de Redes de Sensores Sem Fio.

A classificação proposta por [Mottola e Picco 2011] pode ser descrita como:

1. **Objetivo:** as RSSFs podem realizar apenas o sensoriamento, ou seja, realizar a coleta de dados e transmitir a informação, ou pode além de monitorar, atuar sobre o objeto de monitoramento.
2. **Padrão de Interação:** está relacionada ao propósito da comunicação. O padrão "Um para Muitos" está normalmente relacionado à transmissão de comandos para configuração da rede. "Muito para Muitos" normalmente é utilizada quando há vários nodos sorvedouros e a informação deve repassada a todos eles. "Muito para Um" é a situação mais comum, quando os nodos enviam informação para um nodo sorvedouro que realiza a coleta de dados.
3. **Mobilidade:** a maioria da RSSFs são propostas para a formação em topologia fixa [Chen e Varshney 2004]. Entretanto, pode-se

ter mobilidade de duas formas: i) utilizando nodos sorvedouros móveis, que se movimentam em uma rede fixa realizando coletas de dados ou ii) nodos sensores móveis, que neste caso, eles estão anexados à algum agente, como robôs [Erman-Tüysüz et al. 2008], [Freitas et al. 2010], [Valente et al. 2011], pessoas [Araújo e Becker 2011], ou animais [Liu et al. 2004], ou estão em algum ambiente que lhes proporcionem a mobilidade, como correnteza de rios.

4. Espaço: está relacionado ao significado da informação coletada. O modo "Global" refere-se ao fato de que o dado de um único sensor não é relevante, mas somente os dados coletados de toda a rede que faz sentido. No modo "Regional" refere-se à relevância da informação coletada por regiões da área sensoreada. Pode-se ter uma terceira classificação quanto ao espaço, que seria a "Única", quando o dado de um nodo sensor é tão importante quanto qualquer outro. Um exemplo de aplicação da classificação única, seria o monitoramento de animais [Liu et al. 2004] ou em seres humanos [Araújo e Becker 2011].
5. Tempo: está relacionada ao tempo de geração da informação. As operações de monitoramento e envio de dados podem ser periódicas, quando há um intervalo de tempo pré-definido para a realização destas tarefas, ou podem ser geradas a eventos, quanto a partir de monitoramento é detectado algum valor fora de limites pré-estabelecidos para que os dados sejam enviados.

O foco desta tese está relacionada à aplicações envolvendo mobilidade, tanto com nodos sensores como nodos sorvedouros móveis. Nota-se que ao longo dos últimos anos, a tecnologia de redes sem fio

IEEE 802.15.4¹, usualmente empregada para RSSF, passou a ser largamente utilizada em aplicações envolvendo mobilidade. Além disso, o desenvolvimento da robótica móvel é outro fator que impulsiona o avanço de tecnologias de comunicação que lidam com redes tanto estáticas como móveis [Wang Z. e Zhou 2005], [Zavlanos, Egerstedt e Pappas 2011]. Dessa forma, se torna natural a união das tecnologias de RSSF com a robótica móvel para ampliar a gama de aplicações de sensoriamento e monitoramento.

Podem-se citar exemplos da interação entre a robótica móvel e as RSSFs. Um exemplo do trabalho cooperativo entre estas duas tecnologias é descrito em [Valente et al. 2011]. Neste exemplo, veículos aéreos não tripulados (VANTs) podem realizar coletas de informações de redes de sensores espalhadas em um campo agrícola a fim de aumentar a produtividade e qualidade dos produtos. Em outro exemplo, o uso dos VANTs em cooperação com uma RSSF pode ser usado para auxiliar missões críticas como resgate de pessoas em desastres e monitoramento ostensivo e inteligente de fronteiras [Erman-Tüysüz et al. 2008]. Os robôs podem realizar sensoriamento em locais onde pode ser difícil o acesso para a implantação de uma RSSF. Por fim, a robótica aliada às redes de sensores pode trabalhar cooperativamente para trocar e transmitir informações a fim de ampliar os cenários de aplicação.

Além da interação entre RSSF e robótica móvel, há diversos outros cenários envolvendo RSSF com mobilidade: em [Allred et al. 2007] os pesquisadores anexaram nodos sensores em veículos aéreos para realizar monitoramento do nível de poluição em cidades, em [Eisenman et al. 2009] os autores anexaram sensores em bicicletas para

¹<http://standards.ieee.org/findstds/standard/802.15.4e-2012.html>

monitorar atletas, em [Eriksson et al. 2008] foram anexados sensores em veículos de patrulha e, por fim, em [Dunbabin, Usher e Corke 2005] sensores foram acoplados a sistemas motorizados para coletar informações de sensores estáticos debaixo d'água.

Em todos os exemplos citados, a comunicação é o fator determinante que torna possível a convergência destas tecnologias. Manter a comunicação estável e confiável requer o desenvolvimento de protocolos e aplicações que sejam projetados para lidar com o fator mobilidade. O fator mobilidade traz duas questões importantes, em relação à comunicação, que são abordadas nesta tese.

A primeira questão refere-se aos protocolos de rede, em especial os protocolos de roteamento. Dentre os protocolos de rede, o protocolo de roteamento tem recebido bastante atenção devido à sua grande importância para o provimento de diferenciação de serviço [Jha e Hassan 2002] no que se refere a priorização de mensagens e para o desempenho das redes de sensores sem fio [Akkaya e Younis 2005]. Os protocolos de roteamento são projetados de acordo com a necessidade de aplicação, muitas vezes tendo que reduzir o atraso na entrega de mensagens, serem energeticamente eficientes e, dependendo do cenário, lidar com mobilidade dos nodos sensores.

A segunda questão atribui-se à conectividade dos nodos. A conectividade é o elemento básico da comunicação, e ela se torna volátil devido à constante quebra e estabelecimento dos enlaces entre os nodos móveis. Por esse motivo, a obtenção da noção de conectividade é um conhecimento importante para melhorar o desempenho de protocolos e aplicações. O conhecimento de conectividade pode evitar a perda de mensagens e melhorar o uso da banda de rede, evitando o uso

indiscriminado de mensagens de controle por parte de protocolos de roteamento.

O incentivo para o estudo das questões citadas vem do potencial de aplicações das redes de sensores móveis e os desafios no projeto destas aplicações. A seguir são apresentadas as motivações do trabalho, os objetivos e, por fim, é mostrada a estruturação do documento.

1.1 Motivações

As motivações para este trabalho são abordadas de acordo com cada questão levantada anteriormente. A primeira diz respeito ao desenvolvimento de um protocolo de roteamento que proporcione a priorização de mensagens em cenários com mobilidade. A segunda questão concerne construir o conhecimento de conectividade local pela estimativa de qualidade de enlace também em cenários móveis.

1.1.1 Provimento de Priorização de Mensagens por Protocolo de Roteamento em Ambientes Móveis

O cenário de aplicação que motivou esta parte do trabalho de tese foi apresentado em [Araújo e Becker 2011]. O mesmo consiste em uma competição de corrida ou de *triathlon*. Em competições como estas, centenas de corredores percorrem um longo trajeto para completar a prova. O objetivo do sistema proposto é realizar o monitoramento das condições fisiológicas, como batimento cardíaco e pressão sanguínea, de cada atleta. Cada atleta, então, teria anexado um nodo sensor para realizar as medições e enviar dados. As mensagens seriam entregues

por meio de multi-saltos aos diversos nodos sorvedouros espalhados pelo trajeto. A ideia do cenário de corrida é ilustrada na Figura 1.2.

A camada de aplicação de cada nodo sensor pode enviar dois tipos de mensagens: (i) mensagens normais de sensoriamento, nas quais os dados a serem entregues não requerem qualidade de serviço e (ii) mensagens de alerta, as quais são mensagens que necessitam chegar com atraso mínimo de tempo, necessitando assim serem priorizadas pelo protocolo de comunicação. As mensagens de alerta representam o resultado de um monitoramento, no qual alguma condição anormal foi detectada no comportamento fisiológico do atleta. Como condição anormal nas condições fisiológicas do atleta, considera-se uma parada cardíaca ou qualquer problema que possa comprometer a saúde ou a própria vida do competidor. Sendo assim, é imperativo que mensagens de alerta possam acionar as equipes médicas, para que a situação possa ser tratada com a devida urgência.

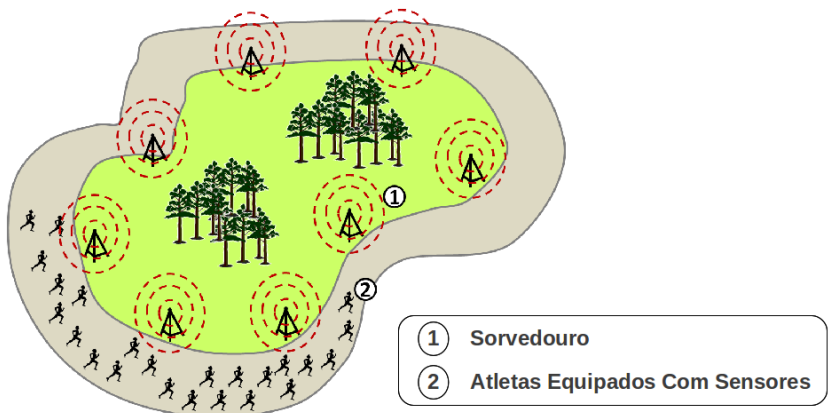


Fig. 1.2: Cenário de Corrida Onde Vários Atletas com Sensores Anexados Percorrem um Trajeto com Nodos Sinks Espalhados.

A partir da definição da aplicação da corrida, devem-se considerar as questões de mobilidade juntamente com uma alta sobrecarga na

rede para propor novos protocolos de roteamento. A questão de alta sobrecarga na rede deve-se ao fato de que todos os nodos, neste caso, são nodos fontes, ou seja, todos os nodos realizam o sensoriamento do atleta e, portanto, enviam mensagens. Além disso, os nodos também podem realizar o roteamento de mensagens de outros nodos. As características presentes neste cenário podem ser encontradas em outros cenários envolvendo mobilidade dos nodos sensores. A topologia dinâmica é um elemento básico presente em qualquer cenário envolvendo mobilidade.

1.1.2 Construção do Conhecimento de Conectividade Local pela Predição de Qualidade de Enlace em Ambientes Móveis

A mobilidade dos nodos afeta a qualidade do enlace (QoL - *Quality of Link*). Quando os nodos se aproximam a qualidade do enlace tende a melhorar, pois o sinal fica mais forte devido à menor distância. O contrário também é verdadeiro, quando os nodos se afastam a qualidade do enlace tende a piorar, pois o sinal enfraquece com o afastamento dos nodos. Esse comportamento da QoL devido à mobilidade é o outro objeto de estudo deste trabalho. Em cenários dinâmicos, os enlaces são formados e rompidos frequentemente. Por isso, ter o conhecimento da durabilidade do enlace pode evitar transmissões de mensagens de dados e de controle desnecessárias e até infrutíferas. Essa propriedade é particularmente útil para os protocolos MAC e de roteamento.

O conhecimento das mudanças na conectividade em ambientes móveis é tipicamente resolvido pelo emprego de alguma estratégia ba-

seada em localização. Tais estratégias dependem de dados adquiridos na placa de interface rede [Ali, Latiff e Faisal 2005], [Deak, Curran e Condell 2011] e/ou de dispositivos de posicionamento, como GPS [Su, Lee e Gerla 2000] ou *Cricket Compass* [Priyantha et al. 2001]. No caso dos sistemas de posicionamentos baseados em GPS, além do custo, a aplicação se limita a cenários *outdoor*. As soluções que trabalham com posicionamento e não dependem de GPS em cada nodo (aplicações *in-door*) são chamadas de *GPS-free*. Nestas soluções, o dispositivo GPS é substituído por outros dispositivos como sensores de ultra-som, como em *Cricket Compass* ou antenas extras como na solução *Angle of arrival* (AoA) [Peng e Sichitiu 2006].

A principal desvantagem das soluções *GPS-free* é o custo adicional de hardware, pois são necessários antenas extras ou sensores de ultra-som. Além disso, é necessário mais processamento para lidar com as informações de posicionamento, o que gera mais sobrecarga na CPU. Por estas razões, este trabalho foca em soluções de estimativa de conectividade que não necessitam da informação de posicionamento.

Dadas as motivações do trabalho, a seguir é apresentado um resumo dos problemas a serem enfrentados nas tarefas de prover um protocolo de roteamento com priorização de mensagens e fornecer um mecanismo que construa o conhecimento de conectividade local para cenário de RSSF móveis.

1.1.3 Problemas Abordados nesta Tese

Dois temas são tratados nesta tese. O primeiro tema, como já mencionado, é o desenvolvimento de um protocolo de roteamento com

priorização de mensagens. O segundo tema é o provimento de um mecanismo que construa o conhecimento de conectividade local para cenário de RSSF móveis. Os dois temas não têm uma relação direta, entretanto os desafios de cada um deles se encontram sob o mesmo grande assunto de RSSF móveis. O primeiro tema aborda questões referentes à priorização de mensagens pelo protocolo de roteamento, utilizando como premissa o conhecimento irrestrito de posicionamento provido por um sistema de GPS. Por outro lado, o segundo tema acerca as questões de provimento de conhecimento de conectividade local, utilizando apenas a qualidade do enlace, não dependendo de nenhum sistema de posicionamento. Pode-se dizer que a única relação entre os dois temas são os estudos do benefício de se ter um sistema de posicionamento e as suas limitações e o benefício de não usar um sistema de posicionamento, apenas a qualidade de enlace e suas limitações.

Os problemas referentes ao primeiro tema são:

1. A manutenção de rotas em topologia dinâmica com a menor quantidade possível de mensagens de controle.
2. O Provimento de diferenciação de serviço no que se refere a atraso de mensagens, levando em consideração a alta carga na rede, proveniente de vários nodos fontes.

Os problemas referentes ao segundo tema são:

1. A variação da qualidade do enlace mediante interferência e enfraquecimento do sinal.

2. A variação da qualidade de enlace refletida pelo padrão de mobilidade.
3. Caso o padrão de mobilidade mude, como fazer o mecanismo se adaptar?

A seguir são apresentados os objetivos a serem alcançados pela tese.

1.2 Objetivos

Os objetivos da tese são atender a cada desafio delineado na seção 1.1.3 para que as aplicações possam ter uma rede estável em um cenário móvel, e que seus requisitos possam ser atendidos. Os objetivos da tese podem ser divididos em dois objetivos principais:

1. Propor um protocolo de roteamento com priorização de mensagens para uma rede de sensores em ambientes móveis.
2. Propor mecanismo de predição de conectividade local para redes móveis.

Para atingir os objetivos principais da tese, pode-se delinear os seguintes objetivos específicos:

- Quanto ao protocolo de roteamento com priorização de mensagens:

1. Definir uma arquitetura de protocolo de roteamento que faça uso das informações das camadas de rede, MAC e física para nortear as decisões de roteamento.
 2. Definir mecanismos de priorização de mensagens para favorecer mensagens marcadas como prioritárias.
 3. Realizar entrega de mensagens marcadas sem prioridade, por melhor esforço.
 4. Definir métricas para decisão de roteamento que possam refletir a situação de uma rede congestionada e de uma rede descongestionada.
- Quanto ao mecanismo de predição de conectividade local para redes móveis:
 1. Prover um mecanismo de predição de conectividade local para redes móveis baseado na informação da qualidade de enlace.
 2. Prover um mecanismo que seja independente do padrão de mobilidade.
 3. Prover um mecanismo que possa ser utilizado em protocolos de roteamento e protocolos MAC.

1.3 Metodologia

Os dois trabalhos desenvolvidos nesta tese seguiram a mesma metodologia que consistiu em levantamento bibliográfico, desenvolvimento da proposta, validação das propostas por meio de simulações e discussão dos resultados.

Em particular, o trabalho sobre protocolo de roteamento com priorização de mensagens, primeiramente foi visualizado um cenário de aplicação descrito na seção 1.1. A partir deste cenário, foram estudados diferentes protocolos de roteamentos que pudessem ser aplicados ao cenário da corrida. Após a análise dos requisitos presentes no cenário e a análise dos protocolos presentes na literatura, pôde-se projetar um novo protocolo de roteamento. O protocolo proposto herdou as características presentes nos protocolos da literatura e criou novas funcionalidades com intuito de prover o melhor desempenho para o dado cenário.

A metodologia seguida para o segundo trabalho desta tese, o mecanismo de predição de conectividade local, foi similar ao primeiro trabalho, com a diferença de que o mecanismo não foi inspirado a partir de um cenário, mas de qualquer cenário que envolva mobilidade. Além disso, na questão de avaliação, alguns experimentos práticos foram realizados em laboratório para estudar os efeitos da propagação do sinal sem fio em ambiente com presença de interferência e mobilidade. Por fim, o mecanismo criado também foram validados por meio de simulações, além de ser realizado um estudo comparativo com trabalhos correlatos da literatura.

1.4 Modelo do Sistema

O modelo de sistema é composto por nodos móveis se comunicando de maneira ad hoc. A comunicação é realizada por *broadcast* e *unicast*. Para o caso da aplicação da corrida descrita na seção 1.1 de motivações, os nodos sensores enviam dois tipos de mensagem: mensagens com e sem prioridade. As mensagens sem prioridade devem

ser enviadas periodicamente para nodos sorvedouros fixos no trajeto. As mensagens com prioridade seguem a caracterização de mensagens geradas por evento. Neste caso, a aplicação pode gerar aleatoriamente mensagens de prioridade. Para cada parte do trabalho desenvolvido nesta tese, há uma descrição mais detalhada do modelo de sistema.

Em seguida, será apresentada a estruturação do restante do trabalho.

1.5 Estrutura do Trabalho

O documento desta tese está dividido em quatro partes. A primeira parte é composta por este capítulo introdutório. As partes seguintes são delineadas de acordo com cada tema da tese e, por fim, a última parte contém as conclusões finais de todo o trabalho e as perspectivas futuras. Os capítulos estão organizados da seguinte forma:

Parte II: Protocolo de Roteamento com Provimento de Priorização de Mensagens

Capítulo 2: Análise do estado da arte para protocolos de roteamento geográfico para RSSFs e os principais protocolos para redes MANETs.

Capítulo 3: Apresentação da proposta de um novo protocolo de roteamento chamado RACE.

Capítulo 4: Apresentação da avaliação do protocolo RACE e suas comparações com demais protocolos relevantes da literatura.

Parte III: Mecanismo de Predição de Conectividade Local

Capítulo 5: Análise do estado da arte de mecanismos de predição de qualidade de enlace para ambientes móveis.

Capítulo 6: Apresentação dos modelos desenvolvidos para predição de qualidade de enlace para ambientes móveis.

Capítulo 7: Apresentação da avaliação de desempenho dos modelos mediante aos modelos similares da literatura e apresentação de um estudo de caso.

Parte IV: Conclusões e Trabalhos Futuros

Capítulo 8: Conclusões do trabalho e perspectivas futuras.

Parte II

PROTOCOLO DE ROTEAMENTO COM PRIORIZAÇÃO DE MENSAGENS

Capítulo 2

Trabalhos Relacionados a Protocolos de Roteamento para Redes *Ad Hoc*

Neste capítulo são apresentados os trabalhos relacionados a protocolos de roteamento geográfico para redes de sensores sem fio, assim como protocolos de roteamento para MANETs (*Mobile Ad hoc NETWORKs*). Apesar das diferenças entre as RSSFs e as MANETs, existem grandes semelhanças entre elas [Jindal A.K.Verma 2011]. Enquanto, as RSSFs foram projetadas para serem implantadas em larga escala com alta densidade [Warneke et al. 2001], as MANETs foram projetadas para redes esparsas [Garcia-Macias e Gomez 2007].

Entretanto, ambas são redes *ad hoc*, que por definição, são redes que não possuem infraestrutura central para dar suporte as aplica-

ções e protocolos. Sendo assim, as rotas de mensagens não são pré-estabelecidas e por isso, elas precisam ser descobertas. O princípio básico para a descoberta e a formação de rotas está na realização de anúncios por parte dos nodos para registrar a sua presença e/ou algum pedido referente à roteamento. Além disso, a topologia para RSSFs móveis e MANETs é dinâmica. Com uma topologia dinâmica, os protocolos de rede devem se adaptar ao fato de que a vizinhança passa a ser volátil, ou seja, podem surgir novos vizinhos e os vizinhos correntes podem ser desassociados. Segundo [Lenders, Wagner e May 2006], a mudança constante na vizinhança pode tornar as rotas inválidas, o que pode trazer impacto no desempenho dos protocolos de roteamento, pois há um aumento na sobrecarga de mensagens para manutenção da rotas.

Ademais, atualmente, os nodos das RSSFs e MANETs têm recursos computacionais limitados, o que influencia o projeto de aplicações e protocolos. Contudo, pode-se dizer que os nodos da RSSF possuem um menor recurso que os nodos de uma MANET. Um típico nodo em uma RSSF pode possuir um *chipset* como Atmel Atmega128RFA1, que opera com 16 MHz e tem 8 KB de memória SRAM e 128 KB de memória *flash*. Já um nodo para MANET pode ser um *smartphone* com 128 MB de memória [Huang et al. 2010] e capacidade de processamento de executar diversos aplicativos como um *web browser* [Karl e Willig 2005]. Apesar das tecnologias estarem em constante evolução, o que tende a melhorar os recursos computacionais dos nodos sensores e nodos MANETs, note-se, atualmente, uma considerável diferença entre ambos.

Apesar das semelhanças e diferenças entre RSSF e MANET, existem vários protocolos de roteamento para ambas as redes [Hanzo

e Tafazolli 2007]. Neste capítulo são descritos os protocolos proativos, (*table-driven*) e os reativos (*on-demand*). Segundo [Taneja e Kush 2010], os protocolos proativos e reativos são as duas categorias principais de protocolos, dando origem a outros tipos, como os protocolos híbridos, que são uma união entre os protocolos proativos e reativos. Os protocolos geográficos também são descritos, mas eles igualmente fazem parte da classificação de proativo e reativo.

Os protocolos de roteamento para redes móveis *ad hoc* também se diferenciam pelo provimento de diferenciação de serviço para as aplicações. O trabalho realizado em [Abbas e Kure 2010] descreve várias métricas para avaliar o provimento de diferenciação de serviço para as aplicações. De acordo com [Abbas e Kure 2010], as métricas mais comuns são o *throughput*, atraso fim a fim das mensagens e energia consumida para os nodos que utilizam bateria. Pode-se dizer que não há limitação de tipos de métricas de provimento de diferenciação de serviço, pois são as aplicações que as definem de acordo com a necessidade, e que devem ser atendidas pela pilha de protocolo.

A seguir são descritos alguns dos principais trabalhos referentes a protocolos de roteamento geográfico com provimento de diferenciação de serviço para RSSFs estáticas e móveis, bem como, os protocolos MANETs proativos e reativos para MANETs.

2.1 Protocolos de Roteamento Geográfico para RSSF

Dentre vários protocolos de roteamento, os protocolos de roteamento geográfico são os que apresentam o melhor desempenho no aspecto de entrega de mensagens e cumprimento de requisitos tempo-

rais, além de prover escalabilidade [Jin et al. 2009]. Estes protocolos são também caracterizados pelo seu aspecto “guloso”¹, escolhem o menor caminho entre a fonte e o destino. Essa característica tem uma desvantagem, pois pode causar congestionamento nos caminhos escolhidos, além de poder causar esgotamento de bateria dos nodos que forem constantemente escolhidos para realizar a retransmissão.

Os protocolos de roteamento geográficos se baseiam no conhecimento do posicionamento para escolher os vizinhos e construir rotas mais curtas para o destino. O conhecimento sobre o posicionamento pode ser obtido por meio de algum algoritmo *GPS-free* [Ali, Latiff e Faisal 2005], [Latiff, Ali e Faisal 2005] como *Angle of arrival (AoA)* [Peng e Sichitiu 2006] ou Cricket Compass [Priyantha et al. 2001] ou mesmo pela utilização de um dispositivo GPS. As abordagens que não utilizam GPS são conhecidas como soluções *indoor*, pois podem ser implementadas em ambientes onde o dispositivo GPS não pode ser utilizado, como interior de prédios ou subterrâneos.

Em [Amundson e Koutsoukos 2009] são discutidos os aspectos de sistemas de localização *GPS-free* para uma MWSN (*Mobile Wireless Sensor Network*). Um ponto interessante da análise dos autores é o tempo para calcular a localização dos nodos. Enquanto os nodos calculam as posições, estas podem mudar dependendo da velocidade dos nodos. Sendo assim, as localizações podem ficar desatualizadas e prejudicar os protocolos e as aplicações que dependem do sistema de localização. Outra técnica adotada é o uso de “nodos âncoras” para calcular o posicionamento via triangulação. Como os “nodos âncoras” devem enviar periodicamente mensagens para toda a rede sobre a suas localizações, por consequência, a rede pode ficar sobrecarregada com

¹do termo inglês “*greedy*”

informações de localização.

Esses aspectos podem prejudicar o provimento de diferenciação de serviço em uma MWSN. Por outro lado, não faz parte do escopo deste trabalho o estudo sobre a influência dos sistemas de posicionamento no provimento de diferenciação de serviço, assume-se apenas que os nodos tem o conhecimento do posicionamento por meio de GPS.

Importantes trabalhos sobre roteamento geográfico com provimento de diferenciação de serviço são descritos na próxima subseção. Apesar de alguns não serem projetados para lidar com mobilidade, são relevantes quanto ao provimento de diferenciação de serviço.

2.1.1 RAP: A Real-Time Communication Architecture for Large-Scale Wireless Sensor Network

Um dos principais trabalhos voltados para atender aos requisitos específicos de diferenciação de serviço, como restrições temporais é o RAP [Lu et al. 2002], em que os autores propuseram uma arquitetura de comunicação para rede de sensores em larga escala com centenas de nodos sensores. RAP foi o primeiro estudo detalhado sobre o desempenho das questões de cumprimento de prazo em uma rede de sensores multi-saltos.

A arquitetura do RAP, ilustrada na figura 2.1 é composta por cinco componentes: (i) o componente *Query/Event Service APIs* é uma interface entre a aplicação de sensoriamento e as camadas inferiores do protocolo; (ii) o componente *Location Addressed Protocol* (LAP)

é responsável pelo provimento de três tipos de comunicação: *unicast*, *area multicast* e *area anycast*; (iii) o componente *Geographic Forwarding (GF) routing protocol* é responsável por fazer o roteamento guloso dos pacotes; (iv) o componente *Velocity Monotonic (packet) Scheduling (VMS)* é responsável por escalonar os pacotes de acordo com o prazo e a distância do nodo fonte ao *sink*; (v) o MAC é priorizado para comportar a política definida pelo VMS.

Pode-se destacar o terceiro componente, GF, da arquitetura RAP por se tratar do protocolo de roteamento, que é o foco do trabalho desenvolvido na primeira parte da tese. Dotado do conhecimento posicionamento, o protocolo realiza o roteamento geográfico de pacotes. O método adotado para rotear os pacotes é descrito em [Karp e Kung 2000]. Neste método, a decisão de repassar o pacote para o vizinho consideram dois fatores: i) o vizinho precisa ter a menor distância do destino dentre todos os vizinhos e ii) o vizinho precisa estar mais perto do que o nodo que precisa enviar o pacote. Segundo os autores, este método é escalável em relação ao número de nodos na rede e tamanho da rede. Outra característica do método GF, quanto mais densa a rede, maior a probabilidade de se encontrar o melhor caminho entre o nodo fonte e o nodo destino.

Os resultados em [Lu et al. 2002] mostram que o RAP provê um cumprimento de prazos de até 60% com uma alta taxa de dados de \cong 70 pacote por segundo e carga gerada por um terço dos nodos da rede. Entretanto, o RAP não foi projetado para o redes móveis. Além disso, o RAP não considera o tamanho do *buffer* como um aspecto limitador para o desempenho do protocolo.

No cenário da corrida apresentado no capítulo 1, estão presentes

duas características importantes que devem ser levadas em consideração por um protocolo de roteamento, que não foram consideradas pelo RAP. A primeira característica é a carga gerada por todos os nodos da rede, pois 100% dos nodos da rede, com exceção dos nodos sorvedouros, são nodos fontes. Isso gera não somente uma grande carga de dados, mas uma intensa carga gerada por mensagens de controle do protocolo. No caso do cenário da corrida, todos os nodos são geradores de mensagens, pois cada nodo realiza o sensoriamento das condições físicas do atleta, por isso, cada medição precisa ser entregue ao nodo sorvedouro mais próximo.

A segunda característica é o fato de todos os nodos fontes se movimentarem, o que pode aumentar a sobrecarga na rede devido à manutenção das rotas. Outro aspecto importante é a política de escolha de rota. A política de escolha de rota é feita puramente pela posição geográfica dos nodos, o que pode gerar congestionamento em caso de larga escala de nodos fontes, além de drenar a energia dos nodos fontes que são sempre escolhidos para retransmissão.

Estes elementos presentes no protocolo RAP não são adequados para o cenário visualizado na corrida. Por outro lado, a diferenciação de serviço proposta pelo RAP é uma estratégia adequada para priorizar mensagens com requisitos de diferenciação de serviço. A diferenciação de serviço é um ponto importante no cenário da corrida, pois há dois tipos de mensagens trafegadas, com e sem requisitos de diferenciação de serviço.

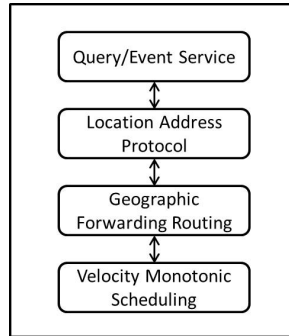


Fig. 2.1: Arquitetura RAP.

2.1.2 SPEED: A *Stateless Protocol for Real-Time Communication In Sensor Networks*

O protocolo SPEED [He et al. 2003] é um protocolo de roteamento com diferenciação de serviço para aplicações *soft real-time*. O SPEED cunhou a estratégia denominada “velocidade de transmissão dos pacotes”, o que influenciou vários protocolos desde então. O protocolo SPEED também é um protocolo de roteamento geográfico e também faz uso da estratégia gulosa para realizar a retransmissão.

O SPEED constrói as rotas dos nodos fontes ao *sink* baseando-se na posição geográfica e na capacidade de “velocidade de transmissão” dos nodos. A velocidade de transmissão é obtida de acordo com a equação 2.1:

$$Velocidade = \frac{D_{vizinho} - D_{destino}}{RTT} \quad (2.1)$$

onde $D_{vizinho} - D_{destino}$ computa a distância entre o nodo vizi-

no e o nodo destino e o RTT computa o *round trip time* da mensagem que é expresso pela fórmula 2.2:

$$RTT = T_{ack} - T_{envio} \quad (2.2)$$

onde T_{ack} é o tempo de recebimento do ack e T_{envio} é o tempo que o pacote foi enviado, ou seja, o tempo do envio de uma mensagem e o recebimento do ACK da mesma.

Quanto a manutenção de de vizinhança o SPEED faz pela troca periódica de mensagens. Assim, os vizinhos que têm o menor atraso são escolhidos para retransmissão. O protocolo SPEED tem um mecanismo chamado de *backpressure-rerouting* que é usado para lidar com os problemas causados por congestionamento nas rotas ou desconexão provocada quando um nodo na rota falha. Neste mecanismo, quando os nodos detectam uma situação de congestionamento ou ruptura na rota, os nodos enviam mensagens de alerta para o nodo fonte, para que esse reduza a taxa de transmissão ou procure novas rotas. Dessa forma, os nodos na rota enviam mensagens de aviso para o nodo fonte a fim de evitar congestionamento, sinalizando ao nodo fonte para procurar novas rotas.

O protocolo SPEED faz a busca pelas rotas mais rápidas para atender aos requisitos temporais das mensagens, além de prover um mecanismo para evitar o congestionamento das rotas. Entretanto, o protocolo foi projetado para redes estáticas, e o fato de realizar manutenção de rotas em ambiente móvel não é viável devido à constante mudança de topologia. Fazer a manutenção de rota em ambiente móvel pode onerar bastante a rede de sensores, além de tornar difícil o

cumprimento de prazos na entrega de mensagens, pois seria necessário reconstruir as rotas caso os enlaces deixassem de existir.

2.1.3 **MMSPEED: *Multipath Multi-SPEED Protocol for QoS Guarantee of Reliability and Timeliness in Wireless Sensor Networks***

O protocolo MMSPEED [Felemban e Ekici 2006] é uma derivação do SPEED. Enquanto o protocolo SPEED procura por rotas mais rápidas para entrega de mensagens, o MMSPEED procura também por rotas alternativas às rotas mais rápidas. Dessa forma, o protocolo aumenta a quantidade de rotas para atender aos diferentes tipos de requisitos temporais.

A diferença do MMSPEED em relação ao SPEED, é prover a diferenciação de rotas pela necessidade das mensagens. Isso significa que mensagens com prazos mais longos podem percorrer rotas alternativas, muitas vezes mais longas. Neste caso, as rotas são formadas de acordo com a velocidade de transmissão (fórmula 2.1) provida pelos nodos na vizinhança. Esta característica descongestiona as rotas mais rápidas, aumentando a probabilidade de cumprimento de prazos.

Assim como o protocolo SPEED, o MMSPEED também foi projetado para redes estáticas, assim pode apresentar os mesmos problemas que o protocolo SPEED em ambiente móvel.

2.1.4 RPAR: *Real-time Power-Aware Routing in Sensor Networks*

Em [Chipara et al. 2006] os autores propuseram o RPAR, que considera o cumprimento de prazo fim a fim na entrega de mensagens, enquanto propõem economizar energia dos nodos. O RPAR adapta a potência de transmissão de acordo com o prazo do pacote, dessa forma garantindo o cumprimento de requisitos temporais. Com o ajuste da potência de transmissão para valores baixos sempre que possível, os autores alegam que o protocolo pode gastar menos energia dos nodos. Além disso, o protocolo não mantém tabela de rota, apenas tabela de vizinhança.

Assim, a escolha para retransmissão é feita nodo a nodo. Essa característica é viável para ambiente de mobilidade, pois não é necessário manter rota, e sim somente a vizinhança imediata de um salto. A manutenção de rotas em redes móveis requer o constante monitoramento e reconstrução dos caminhos escolhidos. Essa característica pode gerar sobrecarga na rede.

Outra particularidade do RPAR é que o controle de vizinhança é feita de modo reativo, ou seja, quando há uma mensagem a ser transmitida, o nodo faz uma busca na vizinhança pelo melhor nodo que possa enviar a mensagem. O fato de manter reativamente a tabela de vizinhança pode causar atrasos no envio de mensagem, pois antes da mensagem ser enviada, é necessário iniciar o processo de busca pelo melhor vizinho retransmissor. Isto pode causar o não atendimento aos prazos de entrega de mensagens impostos pela aplicação. O RPAR possui cinco componentes: i) atribuição dinâmica de velocidade, ii) política de retransmissão, iii) estimador de atraso, iv) gerenciador de vizinhança e o v) adaptador de potência de transmissão.

O componente de atribuição dinâmica de velocidade é responsável por atribuir a velocidade requerida pelo pacote. Defini-se velocidade requerida pelo pacote, como sendo a relação entre a distância euclidiana do nodo fonte ao nodo destino pelo tempo restante que o pacote possui para cumprir o prazo. Para o cálculo do tempo restante para atingir o prazo, é considerado o tempo de armazenamento do pacote na fila. A fórmula 2.3 expressa a atribuição de velocidade requerida:

$$VelocidadeRequerida = \frac{d(S,D)}{P - (T_{chegada} - T_{saida})} \quad (2.3)$$

sendo $d(S,D)$ a distância euclidiana entre o nodo fonte \underline{S} e o nodo destino \underline{D} , \underline{P} o prazo a ser cumprido, $\underline{T_{chegada}}$ o tempo de chegada do pacote na fila e $\underline{T_{saida}}$ o tempo que o pacote sai da fila.

O componente de política de encaminhamento realiza a escolha do nodo vizinho que tem maiores condições de encaminhar o pacote. Para escolher o vizinho, o componente calcula a velocidade de transmissão provida por cada vizinho a uma potência de transmissão \underline{p} e a compara com a velocidade requerida pelo pacote, calculada pelo componente de atribuição dinâmica de velocidade. O cálculo da velocidade de transmissão a uma potência de transmissão \underline{p} provida pelos nodos vizinhos é expresso pela fórmula 2.4:

$$VelocidadeProvida_p = \frac{d(S,D) - d(N,D)}{delay(N)} \quad (2.4)$$

onde $d(S,D)$ é a distância euclidiana entre o nodo fonte \underline{S} e o

nodo destino \underline{D} , $d(N,D)$ a distância euclidiana entre o nodo vizinho \underline{N} e o nodo destino \underline{D} e $delay(N)$ o atraso de transmissão entre o nodo fonte \underline{S} e o nodo vizinho \underline{N} .

O componente estimador de atraso computa o tempo atraso do canal, ou seja, o tempo que uma mensagem leva para chegar com sucesso ao nodo vizinho. Para o cálculo do atraso do canal entre o nodo fonte e o nodo vizinho é calculado o *round trip time* (RTT) da mensagem (vide fórmula 2.2).

O componente gerenciador de vizinhança realiza a manutenção da tabela de vizinhança de um salto por requisição/resposta. O método de requisição/resposta consiste em que o nodo fonte faz uma busca por nodos na vizinhança e estes respondem. O componente gerenciador de vizinhança mantém a tabela de vizinhança de forma a atualizar as velocidades providas pela vizinhança com o conhecimento de distância e atraso entre os nodos.

O quinto componente, adaptador de potência de transmissão, é responsável por adaptar a potência de transmissão de acordo com a necessidade de cumprimento de prazos. Caso o RPAR não encontre um nodo na sua tabela de vizinhança que possa prover a velocidade requerida pelo pacote, o protocolo escolhe o nodo com a maior velocidade provida e aumenta a potência de transmissão. Por outro lado, se a velocidade de transmissão provida pelo nodo vizinho for maior que a velocidade requerida, o protocolo diminui a potência de transmissão com intuito de economizar energia. O aumento da potência é feito de forma multiplicativa por um fator α . Enquanto que a atenuação da potência de transmissão é feita de forma linear por um fator β .

A escolha dos fatores α e β podem influenciar no acordo entre

cumprimento de prazo e economia de energia.

2.1.5 *PATH: A Novel Real-Time Power Aware Routing Protocol in Wireless Sensor Networks*

Seguindo o mesmo objetivo do RPAR, um trabalho mais recente chamado de PATH [Rezayat et al. 2010] foi proposto para garantir qualidade de serviço similar aquela atendida pelo RPAR. Entretanto, o PATH lida com os requisitos de forma diferente. O protocolo PATH utiliza duas métricas para a escolha do nodo retransmissor. Na primeira, o protocolo se preocupa com a energia consumida pelo nodo, computando a energia residual de cada nodo vizinho. A segunda métrica calcula a velocidade do pacote, que é feita pela distância entre o nodo fonte e o nodo *sink*.

A manutenção da tabela de vizinhança utiliza o método de requisição/resposta (similar ao RPAR) e é realizada com informações baseadas na distância euclidiana de dois saltos. No PATH, a escolha do nodo retransmissor depende de métricas que levam em consideração a velocidade da retransmissão que um nodo vizinho pode prover, bem como a sua eficiência energética para retransmitir o pacote. Entretanto, caso não haja nodos elegíveis, o protocolo realiza uma busca reativa por nodos na vizinhança que possam atender aos requisitos. Essa característica pode ocasionar a perda de cumprimento de prazo na entrega de mensagens. Além disso, caso o cenário de aplicação envolva uma grande de nodos fontes, o PATH não utiliza nenhuma política de retransmissão que considere situações de congestionamento.

2.1.6 TAAE: *A Traffic-Aware Energy Efficient Routing Protocol for Wireless Sensor Networks*

O protocolo chamado de TAAE [Liu e Hong 2007] foi projetado para prolongar o tempo de vida da rede, mantendo o conhecimento da energia residual dos nodos e também ficando ciente das condições do tráfego de pacotes. Entretanto, o TAAE assume que a rede seja estática, então a informação da topologia da rede é espalhada somente em tempo de implantação dos nodos.

Trata-se de um protocolo de roteamento geográfico, em que os autores provêm a informação de posicionamento dos nodos manualmente por um GPS. No TAAE, o nodo *sink* utiliza o algoritmo de Floyd-Warshall [Cormen et al. 1990] para construir a árvore espalhada mínima entre o *sink* e os nodos folhas da rede. Como a informação da árvore construída pelo *sink* é espalhada pela rede, os nodos da árvore proativamente espalham suas condições de energia residual e de tráfego pelos caminhos da árvore, assim essas informações são utilizadas para o roteamento das mensagens.

O TAAE apresenta características que são: a preocupação com o consumo de energia e com o congestionamento. Entretanto, o protocolo foi projetado para redes de sensores estáticas. O próprio fato de criar árvores em ambiente móvel deveria ser feito com cuidado, pois a constante mudança de topologia vai causar a desatualização das informações da árvore ou, caso haja uma manutenção a um curto período, pode causar um esgotamento da rede de sensores.

Tabela 2.1: Atributos Utilizados pelos Protocolos

Protocolos	Velocidade de Transmissão	Qualidade do <i>Link</i>	Ocupação de <i>Buffer</i>	Energia
RAP	X	-	-	-
SPEED	X	-	X	-
MMSPEED	X	-	X	-
TAE	-	-	X	X
PATH	X	X	-	X
RPAR	X	-	-	X

2.1.7 Resumo

A Tabela 2.1 resume os atributos utilizados pelos protocolos descritos para realizar as decisões de roteamento. Como se pode notar, a velocidade é uma preocupação para a maioria dos protocolos que atendem a requisitos temporais. Entretanto, não se observa nenhum protocolo que considere além da velocidade e energia, as seguintes características: ocupação do *buffer*, estimação de qualidade de enlace. A ocupação de *buffer* é um dos fatores que podem influenciar no desempenho do protocolo, visto que o estouro de *buffer* causa a perda de pacotes. Já a estimativa da qualidade do enlace reflete situações como colisão e desistência de aquisição do canal.

O cenário da corrida descrito no capítulo 1 aponta para uma grande quantidade de mensagens de dados geradas, pois todos os nodos da rede são nodos fontes. Sendo assim, considerar a situação de *buffer* e qualidade de enlace podem ser importantes métricas para escolha do melhor nodo retransmissor, pois a rede poderá estar congestionada devido ao intenso tráfego de dados. Também se pode concluir que não há protocolo capaz de lidar ao mesmo tempo com mobilidade, alta carga na rede e priorização de mensagens.

Nenhum dos protocolos apresentados considerava a mobilidade

nos seus projetos, mas proveem diferenciação de serviço para aplicações. Já protocolo RPAR é um candidato a ser utilizado em ambientes de mobilidade, pois não faz a manutenção de rotas. Além de o RPAR ser um protocolo clássico, ele tem características de protocolo reativo, pois ao receber mensagens para retransmitir, o nodo consulta a sua vizinhança, a priori, para saber se algum vizinho atende aos requisitos exigidos pela mensagem.

Por esses motivos, o protocolo RPAR foi implementado e comparado com o protocolo proposto que é descrito no capítulo 3. Na próxima seção são apresentados os principais protocolos de roteamento para *Mobile Ad Hoc Networks* (MANETs).

2.2 Protocolos de Roteamento para MANETs

Os protocolos para MANETs são projetados para lidar essencialmente com a mobilidade dos nodos, o que acarreta em tornar a topologia dinâmica. Os protocolos MANETs podem ser basicamente divididos em proativos e reativos. Os protocolos proativos fazem manutenção periódica das rotas entre os nodos fontes e os nodos destinos. Por outro lado, os protocolos reativos procuram as rotas por demanda.

Os protocolos proativos foram projetados para terem as rotas para os destinos sempre atualizadas. A vantagem dessa abordagem é que os pacotes sofrem menos com o atraso devido à falta de rota, pois, a princípio, sempre há uma rota disponível. Essa característica é importante para aplicações que precisam de qualidade de serviço, como minimizar o atraso fim-a-fim. Entretanto, manter a disponibilidade de rotas gera muita sobrecarga na rede. Mesmo que em algum período não

haja necessidade de transmissão de dados, as rotas são mantidas. Essa medida consome mais energia dos nodos, o que pode ser um problema para dispositivos alimentados a bateria.

Pelo fato dos protocolos proativos gerarem grande sobrecarga na rede para manutenção de rotas, os protocolos reativos trabalham de forma contrária. As rotas são geradas por demanda, isto é, caso haja necessidade de se ter uma rota, ela é construída. Essa abordagem reduz bastante a sobrecarga na rede, pois em períodos que não há necessidade de transmissão de dados as rotas não são mantidas. Entretanto, há um aumento no atraso fim-a-fim do pacote, caso seja necessário realizar o processo de descoberta de rota.

Em seguida, são detalhados três dos principais protocolos de roteamento proativos conhecidos na literatura.

2.2.1 *Destination-Sequenced Distance-Vector*

O protocolo *Destination-Sequenced Distance-Vector* (DSDV) [Perkins e Bhagwat 1994] é classificado como *distance vector*, por isso utiliza o algoritmo de Bellman-Ford para calcular as rotas. Além disso, no DSDV os nodos compartilham suas tabelas de vizinhança com os vizinhos. Cada nodo mantém as informações de rotas para todos os destinos. As mudanças de topologia são difundidas pela rede por meio do envio das informações das tabelas de rota, ao contrário dos protocolos *link-state*, que difundem apenas o estado da topologia para toda a rede. Nos protocolos *link-state* todos os nodos da rede constroem um mapa de conectividade de seus vizinhos e distribuem este mapa para toda a rede.

O DSDV garante a não ocorrência de *loops* pela utilização de um número sequencial para cada entrada na tabela de rota. Toda entrada na tabela recebe um número sequencial par, e este número é atualizado periodicamente para um número também par, à medida que os nodos enviam a atualização das suas tabelas. Há, também um campo chamado de métrica, que é utilizado no cálculo da melhor rota para o destino. Caso seja detectada uma quebra de enlace, o número sequencial do vizinho é incrementado de 1 e o campo métrica é modificado para o valor infinito. Essa mudança na topologia da vizinhança é repassada para os nodos da rede. A quebra do enlace é detectada por meio de *time-out* de mensagens *Hello*.

Além dos problemas de sobrecarga causados pelos protocolos proativos, o DSDV precisa gerar novos números sequenciais e informá-los aos nodos da rede antes que a rede estabilize. Como a atualização das tabelas precisa ser transmitida para toda a rede cada vez que a topologia muda, essa característica não é muito adequada para redes cuja topologia muda constantemente gerando um alto custo na rede [Taneja e Kush 2010].

2.2.2 *Optimized Link State Routing*

O protocolo *Optimized Link State Routing* (OLSR) [Jacquet et al. 2001] é classificado como protocolo *link-state*. Assim, a informação de topologia fica difundida por toda a rede. A difusão da topologia é periodicamente executada a cada vez que a vizinhança muda, assim, o protocolo difunde a nova topologia pela rede.

O protocolo OLSR otimiza o comportamento dos protocolos

link-state elegendo nodos roteadores (*multipoint relay* - MPR) para executar a difusão do mapa da rede, a qual é feita somente para os nodos roteadores. Os mapas são construídos a partir do envio periódico de mensagens de controle (*Hello*) por todos os nodos da rede. As mensagens de controle *Hello* são difundidas até dois saltos. Dessa forma, cada nodo seleciona os nodos roteadores para enviar os dados para os nodos destinos. O conjunto de nodos MPR forma um *backbone* que periodicamente difunde as informações de topologia da rede.

O protocolo OLSR herda as vantagens e desvantagens de um protocolo proativo. Além disso, o OLSR pode ser também classificado como um protocolo *link-state*, pois herda característica de difusão da topologia da rede, o que a sobrecarrega. Entretanto, o uso de roteadores para difusão da informação da topologia minimiza a quantidade de informações transmitidas na rede, pois somente um conjunto de nodos poderá realizar essa tarefa.

O trabalho apresentado em [Couto et al. 2003], chamado de OLSR-ETX (Expected Transmission Count), propôs uma extensão ao protocolo OLSR. A proposta utiliza a taxa de perda de pacotes para determinar o melhor caminho, ou seja, é levada em consideração a qualidade do enlace para calcular a melhor rota, não somente se baseando no menor caminho. A qualidade do enlace é determinada tanto da origem para o destino, como do destino para a origem. Os experimentos realizados pelos autores demonstram um aumento significativo na vazão de dados em relação ao OLSR original.

2.2.3 *Better Approach To Mobile Adhoc Networking*

O protocolo *Better Approach To Mobile Adhoc Networking* (BATMAN) [Neumann C. Aichele e Wunderlich 2007] originalmente foi projetado para substituir o protocolo OLSR. A ideia básica é a descentralização do conhecimento do melhor nodo roteador através da rede. Os nodos chamados de originadores constroem e atualizam suas tabelas de rota, inundando a rede com pacotes chamados BATMAN *packets*. A estratégia para encontrar as melhores rotas pela rede é determinar para cada destino um melhor vizinho de um salto até o destino.

Ao contrário dos protocolos *link-state*, o protocolo BATMAN descentraliza a informação das melhores rotas para o destino. Isto elimina a necessidade de enviar as mudanças na topologia para todos os nodos da rede. O BATMAN não determina o caminho completo da origem ao destino, mas utiliza as mensagens dos nodos fontes. O pacote somente é repassado para o vizinho próximo que está na “direção” do destino, ou seja, para o nodo que esteja a um menor número de saltos do destino. Os pacotes são entregues ao vizinho na mesma direção que, por sua vez usa o mesmo mecanismo. Esse processo é repetido até que o pacote chegue ao destino.

A estratégia adotada pelo protocolo BATMAN elimina a necessidade de espalhar a informação de mudança da rede para todos os nodos, assim reduzindo a sobrecarga na rede. A técnica de escolha do próximo vizinho baseado na sua direção em relação ao destino é similar à técnica utilizada por alguns protocolos de roteamento geográficos, como o RPAR, o PATH, diferenciando do modo como a noção de direção é determinada.

As próximas subseções descrevem três importantes protocolos de roteamento reativos.

2.2.4 *Ad hoc On-Demand Distance Vector*

O protocolo *Ad hoc On-Demand Distance Vector Routing* (AODV) [Perkins e Royer 1999] é baseado no conceito dos protocolos *distance-vector*, ou seja, utiliza o algoritmo de Bellman-Ford para calcular a melhor rota. O AODV herdou do DSDV a técnica de sequenciamento numérico para manutenção da tabela de rotas para evitar o problema de *loops*.

Como o AODV é um protocolo reativo, ele permanece inativo até que haja necessidade de criação de uma rota. A partir do momento em que é preciso encontrar uma rota, o protocolo inicia o processo de descobrimento. Neste processo, mensagens de requerimento de rota são difundidas por toda a rede. Quando as mensagens de requerimento atingem o nodo destino, este responde difundindo a resposta por toda a rede até que chegue ao nodo origem.

Uma vez que a rota é criada, ela é mantida por um tempo pré-definido pelos nodos que participam da rota por meio de mensagens *Hello* por um salto de distância. Este tempo pré-definido estabelece a validade da rota. Caso seja detectada uma quebra de enlace na rota, dentro de tempo de validade da rota, o nodo que a detectou inicia um processo de envio de mensagem para o nodo fonte, informando sobre a quebra de enlace. Posteriormente, o nodo fonte reinicia o processo de reparo para reconstruir a rota.

A vantagem do AODV é que não é criado um tráfego extra para a manutenção das rotas existentes, visto que somente a vizinhança de um salto é periodicamente verificada, ao invés de toda a rota. Entretanto, como todo protocolo reativo, o estabelecimento inicial da rota é um processo bastante custoso e pode provocar o atraso no envio das mensagens [Taneja e Kush 2010].

2.2.5 *Dynamic Source Routing*

O protocolo *Dynamic Source Routing* (DSR) [Johnson e Broch 1999] é similar ao protocolo AODV na questão de construção das rotas, ou seja, somente a faz quando é necessário. Entretanto, a diferença básica é que a rota é definida pelo nodo fonte, ou seja, os pacotes do nodo fonte carregam a informação completa do endereço de todo o caminho até o destino. Já no AODV, são os nodos destinos e intermediários que armazenam a informação do próximo salto para cada fluxo de dados.

Os principais mecanismos que compõem o protocolo são o descobrimento e manutenção de rota. Os nodos intermediários que pertencentes à rota mantêm uma cache da rota. Outra diferença em relação ao AODV é que o DSR não utiliza o processo de manutenção periódica com mensagens *Hello*, pois essa característica é da natureza dos protocolos *table-driven*. A quebra de enlace é percebida quando o caminho está sendo utilizado e o nodo que participa da rota não responde a confirmação de recebimento da mensagem. Caso aconteça uma quebra de enlace na rota, o protocolo inicia um processo de reparo da rota.

É interessante observar que o protocolo DSR é completamente *on-demand*, inclusive no seu processo de manutenção de rota. Já o

AODV utiliza a filosofia proativa para manutenção da vizinhança de um salto pertencente à rota. Esse comportamento proativo é menos oneroso que o processo de manutenção de rota dos protocolos naturalmente proativos, contudo o DSR apresenta menos sobrecarga para rede.

2.2.6 *Dynamic MANET On-demand*

O protocolo *Dynamic MANET On-demand* (DYMO) [Chakeres e Perkins 2009] também é similar ao protocolo AODV quanto aos processos de criação e manutenção de rotas. Entretanto, o DYMO elege nodos como roteadores para fazer o processo de retransmissão. Sendo assim, o processo de descobrimento de rota não é feito até o destino, mas para os nodos roteadores alvo. Os roteadores intermediários armazenam as rotas construídas até o roteador alvo nas duas direções, do nodo fonte ao nodo roteador alvo e o contrário.

O processo de manutenção consiste em manter as rotas para os destinos por meio do monitoramento do fluxo de dados para os roteadores. Caso a rota esteja inválida por quebra de enlace, o processo similar a outros protocolos de roteamento reativo como AODV e DSR é executado. Uma mensagem de erro de rota é gerada a partir do ponto de origem do problema ao nodo fonte. Posteriormente, um processo de reparo da rota é iniciado. O DYMO também utiliza a estratégia do número sequencial proveniente do DSDV e também utilizada pelo AODV para evitar *loops* na rota.

2.3 Considerações Finais

Após o trabalho de levantamento da bibliografia a respeito dos protocolos de roteamento em cenários com mobilidade para RSSFs, notou-se que havia algumas questões em aberto que poderiam ser melhores trabalhadas. A primeira questão foi o fato dos protocolos de roteamento que provêm qualidade de serviço não considerarem a capacidade de *buffer* da rede. Este fator deve ser observado para melhorar a capacidade de retransmissão de mensagens em cenários onde há uma grande quantidade de nodos fontes na RSSF.

Outra questão é a manutenção de rotas por parte de alguns protocolos de roteamento geográficos. Como a manutenção de rota gera sobrecarga na rede, realizar a manutenção apenas da vizinhança de um salto, como o RPAR, configura uma estratégia mais adequada em ambientes com constante mudança de topologia. Observa-se também que as métricas consideradas para retransmissão encontradas na literatura são: velocidade de transmissão e energia residual dos nodos. A velocidade de transmissão está diretamente relacionada a roteamentos geográficos, que computam o tempo com que uma mensagem é transmitida com sucesso, levando em consideração o posicionamento dos nodos. Outra métrica considerada na literatura é a energia residual dos nodos, que tem por objetivo escolher os caminhos com mais energia residual para retransmissão.

Um aspecto interessante de alguns protocolos de roteamento geográficos apresentados é que eles apresentam características de protocolos reativos e proativos das MANETs. O protocolo PATH faz a manutenção proativa da tabela de vizinhança de dois saltos, já o RPAR faz a manutenção da tabela de vizinhança de um salto por demanda. En-

tretanto, mesmo sendo de modo proativo, a manutenção da tabela de vizinhança é feita somente para os vizinhos até dois saltos, o que é diferente da característica proativa dos protocolos MANETs, que fazem a manutenção de todas as rotas, ou seja, a quantidade de informação trafegada na rede é maior.

Notou-se, também, que o AODV compartilha de uma característica dos protocolos proativos. A criação das rotas é feita de modo reativo todavia, após a rota ser estabelecida, o AODV faz a manutenção proativa da rota durante um determinado período de tempo. O tempo de manutenção da rota é regido pela validade da rota estabelecida. Apesar de a manutenção ser proativa, no AODV somente os vizinhos de 1 salto da rota anunciam a sua presença, evitando a inundação da rede com a informação completa da rota.

Pode-se dizer que, pela natureza dos protocolos reativos para MANETs, eles seriam mais adequados para serem implementados em redes de sensores. A arquitetura Zigbee², proposta para pilha de protocolos das redes de sensores, considera o AODV como protocolo de roteamento no contexto de rede *Mesh*.

As questões discutidas nesta seção, como grande fluxo de dados gerados por todos os nodos da rede, além da mobilidade que gera a necessidade de manutenção de vizinhança, são características encontradas no cenário da corrida e descritas no capítulo 1. Estas questões foram consideradas para o projeto do protocolo proposto que será apresentado no próximo capítulo.

No capítulo 3 é apresentada a primeira parte do trabalho reali-

²<http://www.zigbee.org/>

zado no desenvolvimento da tese. Como primeiro resultado foi desenvolvido um protocolo de roteamento para lidar com as questões abordadas como mobilidade, alta carga de mensagens e provimento de diferenciação de serviço. A arquitetura do protocolo e seus componentes serão descritos a seguir.

Capítulo 3

Protocolo de Roteamento Proposto

3.1 Introdução e Apresentação do Protocolo RACE

Como introduzido no capítulo 2, vários protocolos de roteamento foram propostos com intuito de prover qualidade de serviço para aplicações em RSSF. Entretanto, devido às características reunidas no cenário de aplicação da corrida, proposto neste trabalho (vide seção 1.1.1), foi observado que existem questões que não foram consideradas nos protocolos descritos no capítulo anterior, pois estes não atendem todos os requisitos da aplicação. Diante dos desafios apresentados para prover qualidade de serviço em aplicações de RSSF com mobilidade, um novo protocolo de roteamento foi desenvolvido.

O protocolo foi nomeado de RACE (*netwoRk conditions Aware*

geographiCal forwarding protocol for rEal-time applications in mobile wireless sensor network) [Araújo e Becker 2011]. O RACE foi projetado para suportar aplicações de nodos sensores móveis que se comuniquem com nodos *sinks* estacionários. Essencialmente, o RACE foi implementado sob a ótica de um algoritmo de roteamento geográfico, pois os nodos são cientes de suas localizações geográficas. Além disso, o RACE considera vários aspectos da rede para realizar as decisões de roteamento. Para isto, o RACE mantém uma tabela de vizinhança atualizada a fim de realizar decisões de roteamento de maneira rápida. O roteamento é feito nodo a nodo, e cada nodo calcula uma pontuação da vizinhança para escolha do melhor nodo para retransmissão das mensagens.

O mecanismo de transmissão adotado pelo RACE tem dois objetivos principais: (i) fazer diferenciação de serviço entre pacotes com prioridade de entrega; (ii) evitar congestionamento local para prover um balanceamento de carga na rede. Como o RACE foi projetado para redes de sensores, ele também deve ser energeticamente eficiente.

3.2 Modelo do Sistema

O modelo de sistema considerado para concepção do protocolo de roteamento geográfico com garantias de priorização de mensagens, apresentado nesta tese, consiste de uma rede com nodos móveis realizando comunicação de maneira ad hoc. Os nodos se movimentam conforme uma situação de uma corrida de *triathlon* ou São Silvestre, por exemplo. A velocidade relativa entre os nodos é maior que zero, assim a topologia da rede se torna dinâmica. Os nodos devem manter periodicamente uma tabela de vizinhança de um salto. Os dados

trocados para a manutenção da tabela de vizinhança devem conter a identificação do nodo, as coordenadas X e Y e a situação de *buffer* local.

Além disso, são considerados centenas de nodos que geram periodicamente mensagens resultantes de medições fisiológicas dos atletas e de posicionamento. Estas mensagens são consideradas mensagens normais que contém um valor de medição de pressão sanguínea e o posicionamento geográfico do atleta. Os nodos também podem gerar mensagens por evento, que são resultados de medições anormais do atleta. As mensagens anormais são consideradas mensagens prioritárias e devem chegar ao destino com o menor atraso possível.

Os nodos podem entregar as mensagens para qualquer nodo sorvedouro espalhado ao longo do trajeto. Os nodos sorvedouros são dispostos de maneira simétrica no trajeto de maneira que não haja um grande espaço vazio. Da mesma forma, os nodos sorvedouros não devem estar ao alcance uns dos outros para que haja a necessidade de entrega de mensagens, via multi-saltos, por parte dos nodos móveis.

3.3 O Protocolo RACE

Como observado em [Chipara et al. 2006], situações de congestionamento degradam o desempenho dos protocolos de roteamento, pois mais pacotes são perdidos e assim aumenta a necessidade de retransmissão, e em consequência disso, aumenta a disputa pelo canal. Por esse motivo, foi decidido que o RACE estaria ciente do *buffer* residual (quantidade de *buffer* restante) para poder tomar ações visando evitar a perda de pacotes devido ao congestionamento da rede. Para

tanto, cada nodo precisa manter o conhecimento médio da situação de *buffer* da vizinhança de um salto. Combinado com essa estratégia, o RACE implementa um mecanismo de diferenciação de serviço que prioriza pacotes marcados como preferenciais, dos pacotes comuns.

Outra característica importante que precisa ser ressaltada é o fato de o protocolo não manter rota para o destino, dada a constante mudança de topologia em cenários móveis. A escolha de transmissão é executada nodo a nodo e somente uma tabela de vizinhança de um salto é mantida de maneira proativa.

Em comparação com os trabalhos discutidos na seção 2.1, a proposta do protocolo RACE considera três métricas para atender às necessidades de qualidade de serviço: velocidade do enlace (como no SPEED, RPAR e PATH), qualidade do enlace e para lidar com alta carga de mensagem foi incluído o *buffer* residual como uma métrica. Trabalhar com *buffer* residual é uma estratégia para lidar com o congestionamento e suas consequentes perdas de pacotes. Entretanto, segundo [Kurose e Ross 2006], as perdas de pacotes são provocadas não somente por estouro de *buffer*, mas por colisões e desistência de aquisição do canal.

Cada métrica tem uma finalidade no modelo de sistema considerado. A métrica de velocidade tem como objetivo identificar os nodos vizinhos que provejam a maior velocidade de transmissão para estes possam ser escolhidos em casos onde há a necessidade de minimizar o atraso na entrega mensagens com prioridade. A métrica de qualidade de enlace tem como objetivo identificar os nodos com enlaces mais estáveis para minimizar a perda de pacotes, e por consequência diminuir a contenção do canal. A métrica de *buffer* também tem o objetivo de

minimizar a perda de pacotes, mas considerando a perda de pacotes por esgotamento de *buffer*. Pode-se dizer, alinhado ao modelo de sistema, que as métricas de *qualidade de enlace* e *buffer* tem como objetivo maximizar a entrega de dados normais pela escolha de caminhos que não sejam necessariamente o menor caminho para se chegar ao destino. Essa escolha de caminhos alternativos em relação aos caminhos mais curtos tem como intuito deixar sempre que possível os caminhos mais curtos para mensagens com prioridades. Sendo assim, as mensagens com prioridades teriam mais chances de chegarem aos destinos com menor atraso possível.

As questões de colisão e desistência de aquisição do canal estão diretamente relacionadas com a contenção do canal, o que reflete na qualidade do enlace. Por isso, a qualidade do enlace está sendo considerada no protocolo RACE como umas das métricas para escolha dos nodos retransmissores. Estas métricas estão disponíveis para cada nodo, e estão distribuídas entre os componentes que constituem a arquitetura do protocolo RACE, como é detalhado na sessão 3.4.

Entretanto, para o uso das métricas é necessário um balanceamento apropriado. Por essa razão, o protocolo RACE lida com os pesos de cada métrica que são utilizados para computar a pontuação. O uso de pesos serve para direcionar a escolha do melhor nodo para retransmissão que provenha as melhores condições para atender aos requisitos de qualidade de serviço exigidos pela mensagem.

Como exemplo, para uma mensagem que seja marcada como preferencial, as métricas de velocidade e qualidade de enlace são os principais atributos a serem observados para a escolha do nodo retransmissor. O uso da velocidade tende a escolher o nodo geograficamente

mais perto do nodo *sink*. Entretanto, o uso exclusivo desta métrica tende a tornar a retransmissão gulosa, o que pode causar congestionamento ao longo do caminho pela escolha viciada dos mesmos nodos. Além disso, de acordo com [Krogmann et al. 2009], o uso da qualidade do enlace para escolha do melhor nodo diminui a probabilidade de retransmissão dos pacotes, pois enlaces mais estáveis tendem a ter mais sucesso na entrega do pacote.

Para uma mensagem cujo objetivo é a entrega pelo melhor esforço, a qualidade do *link* e a situação de *buffer* residual devem ser as principais características a serem observadas para a escolha do melhor nodo. Isto ocorre, pois a maioria das mensagens geradas pela aplicação da corrida não teriam prioridade, enquanto que as mensagens com prioridade seriam em uma proporção menor. Por esta razão, o uso da métrica de *buffer* tende a escolher os nodos de maneira a balancear a carga de mensagens na rede.

Para implementação do protocolo proposto é fundamental ter acesso às métricas adotadas. A próxima seção descreve os componentes que foram projetados para suportar o protocolo RACE.

3.4 Arquitetura *Crosslayer* do Protocolo

A arquitetura do protocolo RACE é constituída por sete componentes, sendo que quatro estão localizados na camada de rede, dois na camada MAC e um na camada física, como mostrado na Figura 3.1. Nas subseções seguintes todos os componentes são descritos em mais detalhes.

A arquitetura *crosslayer* proporciona a facilidade de aproveitar informações disponíveis em diferentes camadas. Como exemplo, o atraso no encaminhamento de mensagens pode ser obtido pelo tempo que leva uma mensagem de reconhecimento. Essa informação é essencial para o cálculo da velocidade de transmissão do nodo.

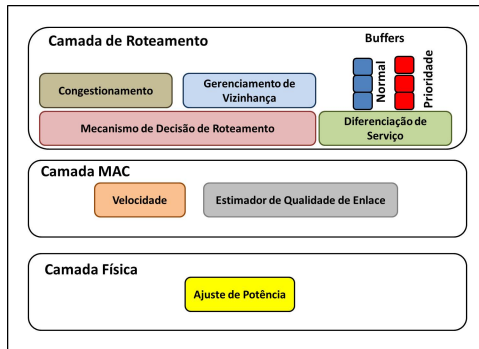


Fig. 3.1: Arquitetura do Protocolo RACE

3.4.1 Componente Gerenciador da Tabela de Vizinhança

O componente gerenciador da tabela de vizinhança é responsável por mantê-la atualizada. A tabela de vizinhança contém um subconjunto de nodos vizinhos na distância de um salto e que tenham a distância igual ou menor para o nodo *sink* mais próximo (Equação 3.1). F_n representa todos os nodos armazenados na tabela, N_j é o conjunto de nodos vizinhos a um salto do nodo fonte e n_i são os nodos que retransmitirão as mensagens. d_i e d_s representam a distância euclidiana dos nodos vizinhos e fonte ao nodo *sink* mais próximo ao nodo fonte.

$$F_n = \{n_i \in N_j | d_s - d_i \geq 0\} \quad (3.1)$$

O motivo da tabela de vizinhança armazenar os vizinhos que estejam em uma distância igual ou menor do nodo fonte ao nodo *sink* deve-se ao fato de que o protocolo tende a entregar as mensagens preferenciais pelo menor caminho. Se uma mensagem é preferencial, os nodos que estiverem com distâncias menores que o nodo fonte terão chances maiores de serem escolhidos para retransmissão. Já para as mensagens comuns, os nodos na mesma distância ou menor terão chances iguais de retransmitir a mensagem. Dessa forma, espera-se evitar a estratégia gulosa para retransmissão de mensagens sem prioridade, realizando um balanceamento de carga na rede.

O mecanismo de atualização da tabela de vizinhança é executado de maneira proativa. A principal vantagem do uso de um mecanismo proativo é permitir que a decisão para escolha do melhor nodo a retransmitir uma mensagem seja feita de maneira rápida. O processo de atualização é executado por cada nodo, o qual realiza um *broadcast* de uma mensagem *beacon* na distância de um salto.

O tamanho da mensagem *beacon* para atualização da tabela de vizinhança é de 26 *bytes* (tamanho típico de um pacote de uma rede de sensores). O conteúdo da mensagem *beacon* é subdividido em três campos: (i) 16 *bits* para identificar o nodo; (ii) 64 *bits* para representar o *buffer* residual; (iii) 128 *bits* para representar as coordenadas *x* e *y* do posicionamento do nodo.

Para evitar colisões, cada nodo envia o *beacon* utilizando um deslocamento temporal randômico (*backoff*) que pode variar entre 1 a 3 segundos. Além disso, se o nodo escutar uma transmissão, ele reinicia o temporizador de transmissão. O mecanismo de manutenção de tabela de vizinhança do RACE é eficiente por não utilizar requisição

e resposta para troca de mensagens *beacon*.

Com intuito de evitar problemas de *loop*, o protocolo RACE anexou no pacote de dados a identificação dos nodos por onde o pacote passava. Dessa forma, um nodo somente pode ser escolhido para retransmitir o dado se a sua identificação não estivesse contida no pacote. Para evitar que o pacote crescesse de acordo com a quantidade de saltos e de acordo com o tamanho da rede, a partir do terceiro salto, a primeira identificação é removida. Com isso, o pacote ficaria com até dois identificadores, representando dois saltos. Esta estratégia não elimina a possibilidade de *loops* mas a reduz. Um outra solução seria memorizar os identificadores por onde a mensagem passasse, entretanto o tamanho da mensagem iria crescer de acordo com a quantidade de saltos.

3.4.2 Componente de Decisão de Roteamento

O componente de decisão de roteamento é responsável por executar a equação 3.2 e escolher o nodo que irá retransmitir a mensagem. Quando uma mensagem chega do *buffer*, sendo de prioridade ou normal, o componente calcula a pontuação, S_i , para cada nodo $F_i \in F_n$. O peso para cada métrica é definido pela aplicação. O nodo com a maior pontuação S_i é o escolhido para retransmitir o pacote.

$$S_i = \gamma_1 * LQE + \gamma_2 * B_r + \gamma_3 * S \quad (3.2)$$

O cálculo para a computação da escolha do melhor nodo pode ser visto na Equação 3.2. O S_i representa a pontuação de cada nodo

i existente na tabela de vizinhança. O LQE representa o resultado de um estimador de qualidade de enlace. O B_r representa o *buffer* residual médio e S é a velocidade provida pelo nodo vizinho. O peso de cada métrica é atribuído ao γ_i , cujo $\sum_{i=1}^3 \gamma_i$ deve ser igual a 1. Cada peso pode ser dinamicamente ou estatisticamente definido, dependendo das necessidades da aplicação.

O processo de escolha do melhor nodo para retransmitir a mensagem é mostrado no algoritmo 1. A saída do algoritmo é o índice do nodo da tabela de vizinhança com a melhor pontuação. Como entrada no algoritmo tem-se as métricas (velocidade, LQE e *buffer* residual), os pesos e o conjunto de nodos vizinhos.

Input: Conjunto de Nodos Vizinhos, Métricas e Pesos das Métricas

Output: Índice do nodo retransmissor ($fwdNode$)

$maxScore = -1$;

foreach Neighbor i **do**

$currentScore = \gamma_1 * Neighbor[i].lqe + \gamma_2 *$

$Neighbor[i].br + \gamma_3 * Neighbor[i].s$;

if $maxScore < currentScore$ **then**

$maxScore = currentScore$;

$fwdNode = i$;

end

end

return $fwdNode$;

Algorithm 1: Algoritmo para Escolha do Melhor Nodo Retransmissor

3.4.3 Componente de Ciência de Congestionamento

Frente aos trabalhos encontrados na literatura, a utilização o conhecimento da quantidade de *buffer* para tomada de decisão de roteamento é um aspecto inovador presente neste trabalho. Todos os trabalhos sobre protocolos de roteamento que utilizam métricas para construir rotas, consideram a velocidade, qualidade do enlace e a energia residual dos nodos. Porém, como observado em [Hull, Jamieson e Balakrishnan 2004] e recentemente em [Agarkhed, Biradar e Mytri 2012], o estouro de *buffer* é um dos fatores que causam a perda de pacotes e por consequência degrada o desempenho dos protocolos. Por isso, a informação da situação de *buffer* pode ajudar a reduzir a perda de pacotes.

O componente de ciência de congestionamento é responsável por prover as medições sobre a situação de *buffer* dos nodos. Esta métrica é importante, pois o conhecimento da situação de congestionamento pode evitar consequentes retransmissões provocadas por perda de pacotes, o que ocasiona uma grande disputa pelo canal. O trabalho [Fdili Youssef Fakhri 2012], posterior ao RACE, também utilizou o conhecimento de *buffer* para realizar decisão de roteamento melhorando o desempenho do protocolo MMSPEED, discutido na subseção 2.1.3.

Na camada de rede do protocolo RACE, cada nodo tem dois tipos de *buffers*, para pacotes com e sem prioridade, como pode ser visto na Figura 3.1. A fila normal é organizada de maneira FIFO, enquanto a outra é uma fila de prioridade. As prioridades na fila são computadas considerando o tempo de criação do pacote. Os pacotes que foram criados há mais tempo, têm maior prioridade. O tempo de criação do

pacote indica que ele está a mais tempo viajando pela rede, então para garantir que ele possa chegar com o menor atraso possível, ele recebe a prioridade de ser transmitido.

Para ficar ciente da situação de congestionamento, cada nodo precisa manter o conhecimento da situação de *buffer* da vizinhança. Isto é feito pela cálculo do grau de congestionamento médio (utilizando média móvel) M_{Br} , que consiste da situação local do nodo vizinho no subconjunto F_n . Esse processo é ilustrado pela Equação 3.3, onde Br_i é a normalização do *buffer* local. O termo \underline{Br} é calculado de acordo com a equação 3.4 onde Bc e Bd são, respectivamente, o espaço (*buffer*) utilizado das filas normal e de prioridade e Bc_{max} e Bd_{max} são respectivamente a capacidade máxima dos *buffers* das filas normal e de prioridade.

$$M_{Br} = \frac{\sum_i^N Br_i}{N} \quad (3.3)$$

$$B_r = (Bc_{max} - Bc) + (Bd_{max} - Bd) \quad (3.4)$$

Considera-se que as aplicações de todos os nodos tenham períodos diferentes para os envios das mensagens. Sendo assim, espera-se que haja nodos que enviem mais mensagens que outros. Dessa forma, serão geradas regiões mais congestionadas que outras, e é este fato que a métrica de *buffer* pretende explorar. A Figura 3.2 ilustra uma situação de congestionamento, onde a métrica de *buffer* é utilizada para evitar regiões congestionadas. Nesta figura, os nodos encaminham mensagens por regiões alternativas onde os nodos estão descongestionados,

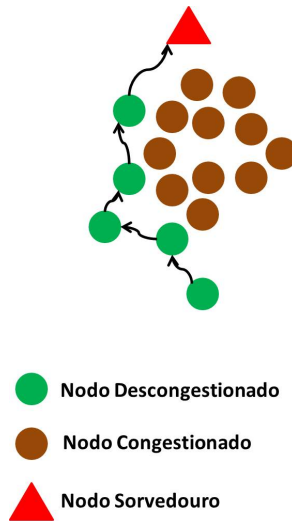


Fig. 3.2: Evitando Regiões Congestionadas

evitando que nodos com os limites máximos de *buffer*s atingidos sejam escolhidos. Dessa forma, que novas mensagens possam ser descartadas.

O processo para o cálculo da métrica de *buffer* residual é executado em duas etapas. A primeira, descrita pelo algoritmo 2, é executado por todos os nodos para informarem aos seus vizinhos sobre as suas situações atuais de *buffer*. Os espaços utilizados dos *buffer*s das filas normal e prioridade e as suas capacidade máximas são os parâmetros de entrada do algoritmo. Como resultado o algoritmo devolve o *buffer* residual do nodo local e essa informação é anexada a mensagem *beacon* que será transmitida para os nodos vizinhos no próximo período

de envio.

Input: *Buffer* Normal Utilizado (Bc), *Buffer* Prioridade Utilizado (Bd), *Buffer* Normal Máximo (Bc_{max}), *Buffer* Prioridade Máximo (Bd_{max})

Output: *Buffer* Residual (Br)

$Br = (Bc_{max} - Bc) + (Bd_{max} - Bd);$

return Br ;

Algorithm 2: Algoritmo para o Cálculo do *Buffer* Residual Local

A segunda etapa, ilustrada no algoritmo 3, ocorre no nodo fonte que mantém o conhecimento da média da situação de *buffer* dos vizinhos. O conjunto de nodos vizinhos é a entrada do algoritmo. Como resultado, o *buffer* residual médio de cada nodo vizinho é atualizado para uso posterior na escolha das métricas.

Input: Conjunto de Nodos Vizinhos *Neighbor*

$TamanhoJanela = 5;$

foreach *Neighbor* i **do**

for $j = 1$ to $TamanhoJanela$ **do**

 | $Neighbor[i].br = Neighbor[i].br + Neighbor[i].br[j];$

end

$Neighbor[i].br = Neighbor[i].br / TamanhoJanela;$

end

Algorithm 3: Algoritmo para o Cálculo do *Buffer* Residual Médio dos Vizinhos

3.4.4 Estimador de Qualidade de Enlace

O componente estimador de qualidade de enlace foi implementado na camada MAC e pode ser usado sob demanda da camada de roteamento para computar a métrica de qualidade de enlace. De acordo com [Woo, Tong e Culler 2003], é essencial ter um estimador de qualidade de enlace, pois a escolha do melhor enlace está diretamente relacionada com o sucesso na transmissão de pacotes. Como consequência da perda de pacotes, os nodos sensores gastam mais energia devido à necessidade de retransmissão, além do aumento da disputa pelo canal.

Dentre os diversos estimadores de qualidade de enlace existentes, foi escolhido o WMEWMA (*Window Mean Exponentially Weighted Moving Average*) [Woo e Culler 2003], pois, além de ser um estimador clássico da literatura, a sua implementação é simples e usa de maneira eficiente a memória, o que, para um nodo sensor é uma característica importante em virtude das restrições de recursos. A Equação 3.5 é usada para calcular o WMEWMA, sendo $LQE_{\alpha,W}(n) \in [0,1]$, $\alpha \in [0,1]$ e $PRR = \frac{N_{ts}}{N_t}$ e $PRR \in [0,1]$, onde N_{ts} é o número de pacotes transmitidos com sucesso e N_t é o número de pacotes transmitidos. O parâmetro W é o tamanho da janela de tempo de observação do LQE e α é o peso dado ao histórico.

$$LQE_{\alpha,W}(n) = \alpha * LQE_{\alpha,W}(n - 1) + (1 - \alpha) * PRR \quad (3.5)$$

O processo para o cálculo da qualidade de enlace é expresso pelo algoritmo 4. A cada vez que uma mensagem é enviada para um nodo vizinho, a taxa de pacotes recebida é computada (PRR). Após

um tempo W o PRR é filtrado com WMEWMA para o cálculo final do LQE. O conjunto de nodos vizinhos e os parâmetros α e W são as entradas do algoritmo. Como resultado, o LQE de cada nodo vizinho é atualizado para uso posterior na escolha das métricas.

```

Input: Conjunto de Nodos Vizinhos  $Neighbor, \alpha, W$ 
foreach  $Neighbor\ i$  do
    | Neighbor[i].prp = Neigh-
    | bor[i].qtdPacotesEnviadosSucesso/Neighbor[i].qtdPacotesEnviados;
end
foreach  $Neighbor\ i$  do
    | for  $j = 1$  to  $W$  do
    | | Neighbor[i].lqe[j] =  $\alpha * Neighbor[i].lqe[j] + (1-\alpha) *$ 
    | | Neighbor[i].prp;
    | end
end

```

Algorithm 4: Algoritmo para o Cálculo do LQE dos Vizinhos

3.4.5 Estimador de Velocidade

O conceito de velocidade de transmissão foi introduzido por [He et al. 2003] e inspirou vários protocolos de roteamento. O componente estimador de velocidade usa o atraso da transmissão de pacotes e a distância relativa do nodo vizinho ao sorvedouro mais próximo. O conhecimento da velocidade provida por um nodo vizinho tem um peso importante para decisão da escolha do melhor nodo f_i , onde $f_i \in F_n$, pois quanto mais perto do próximo sorvedouro o nodo vizinho estiver, maior a chance de satisfazer a priorização das mensagens. Para calcular a velocidade S do pacote, foi utilizada a Equação 3.6. O termo $d(F, S)$ é a distância euclidiana entre o nodo fonte F e o sorvedouro

mais próximo, $d(N,S)$ é a distância euclidiana entre o nodo vizinho N e o sorvedouro mais próximo do nodo fonte F . RTT é o *round trip time* do vizinho N . O somatório da velocidade de todos os nodos, S_m , em F_n , é expressa pela fórmula 3.7.

$$S = \frac{\frac{d(F,S) - d(N,S)}{RTT}}{S_m} \quad (3.6)$$

$$S_m = \sum_i^n \frac{d(F,S) - d(N_i,S)}{RTT_i} \quad (3.7)$$

Para o cálculo do *round trip time* é considerado o momento em que um pacote é transmitido e o momento em que a mensagem *ack* é recebida, como mostrado na Equação 2.2 demonstrada no capítulo 2.

O algoritmo 5 expressa o cálculo do estimador de velocidade de transmissão dos nodos vizinhos. O algoritmo é executado nos nodos fontes com informações do RTT de cada vizinho e suas distâncias euclidianas em relação ao nodo sorvedouro mais próximo do nodo fonte. Ao final, cada vizinho tem a velocidade de transmissão normalizada. O conjunto de nodos vizinhos e distância euclidiana do nodo fonte em relação ao nodo sorvedouro mais próximo do nodo fonte são as entradas do algoritmo. Como resultado, a velocidade de transmissão de cada

nodo vizinho é atualizada para uso posterior na escolha das métricas.

```
Input: Conjunto de Nodos Vizinhos  $Neighbor, d(F, S)$   
foreach Neighbor  $i$  do  
    | Neighbor[i].velocidade =  $(d(F, S) -$   
    | Neighbor[i]. $d(N, S)) / Neighbor[i].rtt;$   
end  
 $velocidadeTotal = 0;$   
foreach Neighbor  $i$  do  
    |  $velocidadeTotal = velocidadeTotal +$   
    | Neighbor[i].velocidade;  
end  
foreach Neighbor  $i$  do  
    | Neighbor[i].velocidade =  
    | Neighbor[i].velocidade/velocidadeTotal;  
end
```

Algorithm 5: Algoritmo para o Cálculo da Velocidade de Transmissão dos Nodos Vizinhos

3.4.6 Power Switch

O componente *Power Switch* foi implementado na camada física, pois permite configurar a potência de transmissão para atingir uma velocidade de transmissão desejável. Dessa forma, o componente estimador de velocidade pode interagir com o componente *Power Switch* em tempo de execução para alterar a velocidade de acordo com a necessidade da aplicação.

O aumento na potência de transmissão aumenta a chance do pacote chegar ao nodo vizinho sem necessidade de retransmissão, o que

acarreta no aumento da velocidade do pacote, pois diminui o seu RTT (*round trip time*). Entretanto, o aumento da potência, aumenta a área de atuação da transmissão, com isso, aumenta a interferência na vizinhança e prejudica a transmissão dos outros nodos. Outra consequência é o aumento no consumo de energia do nodo. Sendo assim, o componente deve ser utilizado somente em casos de necessidade, como a transmissão de um pacote com prioridade.

3.4.7 Diferenciação de Serviço

O mecanismo de diferenciação de serviço foi projetado para priorizar pacotes com qualidade de serviço sobre os pacotes por melhor esforço, aqui chamados de pacotes normais. O componente provê uma diferenciação no processo de enfileiramento. Assim os pacotes na fila de prioridades são despachados primeiro, e somente depois de esvaziar esta fila é que a fila de pacotes normais é processada. Este mecanismo é ilustrado na Figura 3.3.

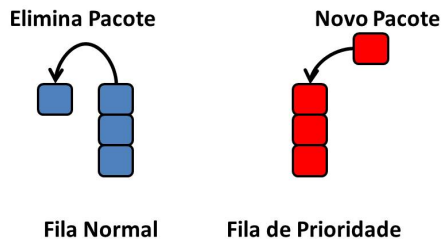


Fig. 3.3: Mecanismo de Diferenciação de Serviço

Se um nodo tiver seu *buffer* de prioridade cheio, ele toma espaço emprestado do *buffer* normal. Além disso, se o *buffer* normal também estiver cheio, n pacotes normais são dispensados para arma-

zenar n novos pacotes com qualidade de serviço. Assim, o tamanho total disponível para armazenar pacotes com prioridade pode chegar à soma total dos dois *buffers* projetados.

Com essa estratégia pretende-se aumentar a disponibilidade de espaço nas filas para os pacotes que tenham prioridades associadas. Dessa forma, acredita-se que as chances dos pacotes com prioridades sofrerem menos atraso aumentem. Como o cenário motivador para o desenvolvimento do protocolo RACE apresenta uma grande carga de mensagens, é provável que os *buffers* dos nodos que estiverem mais próximos ao *sink* sofram de sobrecarga e, com esse mecanismo, os pacotes com entrega por melhor esforço podem dar lugar aos pacotes com prioridade.

No próximo capítulo, são apresentados e discutidos os resultados da avaliação do protocolo RACE. Foram realizados experimentos com intuito de verificar o desempenho do protocolo em relação à taxa de entrega de mensagens e à taxa de atraso, além da verificação da eficiência energética. Além disso, será discutido o impacto da métrica *buffer* nas taxas de entrega e de atraso sobre o cenário simulado.

Capítulo 4

Avaliação do Protocolo Proposto

Neste capítulo são apresentados os resultados e discussões sobre o protocolo RACE descrito no capítulo anterior. Foram feitos três conjuntos de experimentos para avaliar o protocolo proposto e alguns protocolos da literatura. No primeiro conjunto de experimentos, foi testada a capacidade do RACE de entregar mensagens e verificar o atraso na entrega de mensagens com prioridade. Além disso, o RACE foi comparado com um protocolo da mesma classe, ou seja, de roteamento geográfico.

No segundo conjunto de experimentos, foi avaliado o impacto da métrica de *buffer* proposta nesta tese sobre a taxa de entrega de dados. Os resultados mostram que, dependendo da situação do ambiente, o peso dado à métrica de *buffer* traz um significativo impacto à rede que será apresentado nas próximas seções. No terceiro conjunto

de experimentos, alguns protocolos de roteamento MANETs clássicos da literatura, discutidos no capítulo 2, foram simulados nas mesmas condições do primeiro conjunto de experimentos.

4.1 Modelo de Rede

O modelo de rede utilizado para as avaliações consistiu em nodos em movimento circular com variação entre as velocidades de 4mps, escolhida aleatoriamente. Os nodos realizam duas formas de comunicação: i) *broadcast* de mensagens de controle de um salto para manutenção da tabela de vizinhança e ii) *unicast* de um salto para o repasse de mensagens de dados. A periodicidade de envio de mensagens de controle foi regulado em 1 segundo com um deslocamento temporal aleatório de 2 segundos, com intuito de reduzir as colisões. A periodicidade de envio de dados foi regulado em cinco níveis com intuito de aumentar gradativamente a carga de mensagens de dados geradas. Os detalhes quanto a carga de mensagens geradas é detalhado na próxima seção. O tamanho da mensagem de controle foi fixado em 26 bytes: 16 bits para identificação do nodo, 128 bits para suas coordenadas X/Y e 64 bits para o valor do *buffer* residual. Para mensagens de dados o tamanho ficou em 30 bytes: 16 bits para identificação do nodo, 32 bits para memorização das identificações de nodos referentes à 2 saltos, 64 bits para representar o dado coletado e 128 bits para representar as coordenadas X/Y no nodo fonte. Em todos os experimentos foi utilizado o modelo de propagação de sinal *free-space path loss*.

4.2 Descrição dos Experimentos

Para todos os experimentos, foi gerado um cenário de simulação para representar a competição de corrida, na qual cada atleta tem um nodo sensor anexado que envia dados periodicamente. Os dados devem ser roteados até o *sink* mais próximo, dentre os diversos espalhados ao longo do trajeto da corrida (vide Figura 1.2). Este cenário é usado para avaliar como a mobilidade e a carga de dados gerada por cada nodo afetam as taxas de entrega de mensagens e de atraso fim-a-fim.

O cenário de corrida foi implementado com oito *sinks* estáticos espalhados pelo trajeto, como é mostrado na Figura 4.1. Esta configuração garante que haja nodos *sinks* disponíveis por todo o trajeto da corrida, entretanto a área de atualização de recepção de sinal dos nodos *sinks* é limitada. Isto força a necessidade de transmissão de dados de modo *multi-hop* por parte dos nodos fontes.

O modelo de mobilidade escolhido foi o circular, dentre os diversos modelos conhecidos na literatura, pois é o modelo que mais se assemelha à configuração de um trajeto de corrida. Todos os nodos sensores são pré-configurados para saber a localização geográfica de cada nodo *sink*.

Para realizar as avaliações, o protocolo RACE foi implementado na ferramenta de simulação OMNET++ versão 4.x [Varga et al. 2001], sob o *framework* INETMANET [Quintana e Cabrera 2012]. Como camada MAC foi utilizado o IEEE 802.15.4 com CSMA, e topologia *peer-to-peer*. As configurações da camada física e outros aspectos dos experimentos são mostrados na Tabela 4.1.

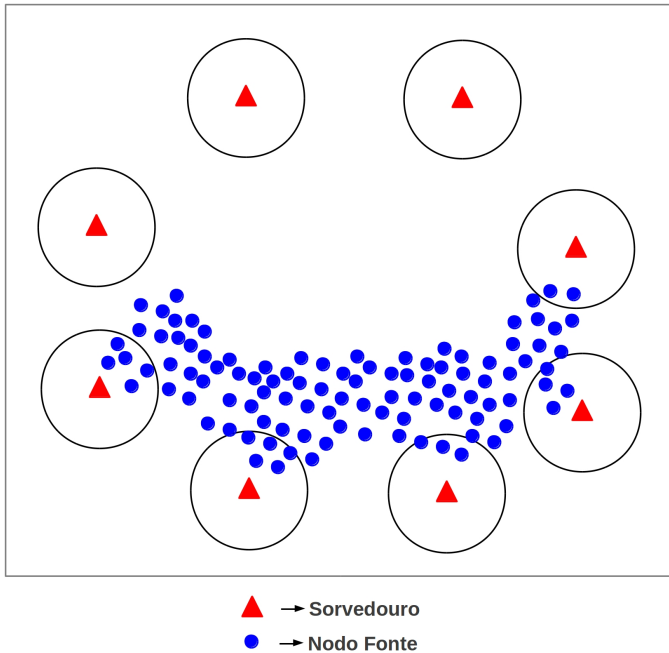


Fig. 4.1: Simulação do Cenário da Corrida. Disposição dos Nodos Sorvedouros e Fontes.

Para simular a mobilidade, foram usados quatro conjuntos de velocidades, definidas em metros por segundo (mps), como mostrado na Tabela 4.2. Em cada simulação, foi fixado um conjunto de velocidades para ser utilizado e cada nodo seleciona a velocidade randomicamente dentro do conjunto. A ideia de ter várias regiões de velocidades é simular diferentes cenários da competição de corrida. Por exemplo, em competições de triatlon o atleta precisa primeiro nadar (*slow speed*), depois andar de bicicleta (*high speed*) e finalmente correr (*moderate speed*). O conjunto de velocidade *very high speed* mostrado na Tabela 4.2 foi utilizado para avaliar o protocolo RACE ao seu limite e então medir o seu desempenho.

Tabela 4.1: Parâmetros Gerais de Simulação

Parâmetros de Simulação	Valor
<i>Sensibilidade</i>	-85 dBm
<i>Potência de Transmissão</i>	5 mW
<i>Thermal Noise</i>	-110 dBm
<i>Tamanho do Buffer</i>	100
<i>Número de Nodos Fontes</i>	300
<i>Número de Nodos Sorvedouros</i>	8
<i>Área de Simulação</i>	2600 x 2600 m ²
<i>Simulation Time</i>	1000 segundos
<i>Modelo de Mobilidade</i>	Circular
<i>Raio do Círculo</i>	2500 metros
<i>Modelo de Propagação de Sinal</i>	Free-Space Path Loss

Tabela 4.2: Conjunto de Velocidades

Parâmetros de Simulação	Valor (mps)
<i>Slow Speed</i>	3 to 7
<i>Moderate Speed</i>	8 to 12
<i>High Speed</i>	13 to 17
<i>Very High Speed</i>	18 to 22

O protocolo RPAR [Chipara et al. 2006] também foi implementado e o seu desempenho foi comparado com desempenho do protocolo RACE.

4.2.1 Primeiro Conjunto de Experimentos: Avaliações dos Protocolos RACE e RPAR

A escolha dos pesos da equação 3.2 foi realizada empiricamente. Como a aplicação envia dois tipos de mensagens, aquelas que necessitam de qualidade de serviço e aquelas cuja entrega é realizada por melhor esforço, os pesos foram escolhidos de acordo com a prioridade das mensagens. Para os dados com qualidade de serviço os pesos foram: 0,4; 0,2 e 0,4 para LQE, *Buffer* e Velocidade de Transmissão respectivamente. Para os dados com entrega por melhor esforço os pesos foram: 0,3; 0,6 e 0,1 para LQE, *Buffer* e Velocidade de Transmissão respectivamente.

Quanto ao protocolo RPAR é necessário configurar os parâmetros α e β do componente de adaptação da potência de transmissão. O valor utilizado para β foi 1, de acordo com [Chipara et al. 2006], mas o valor α foi configurado para 2, para que pudesse ter quatro interações também de acordo com [Chipara et al. 2006]. Os valores mínimos e máximos da potência de transmissão foram de 1 a 5mW. Neste caso o valor da potência de transmissão poderia ser aumentada, de 1 a 5mW com fator 2, mas não podendo ultrapassar o valor máximo da potência de transmissão. O algoritmo 6 expressa a ideia do ajuste de potência. Como parâmetros de entrada são necessários os fatores α e β e uma *flag* que indica se potência deve ser aumentada ou reduzida. Como

resultado, o algoritmo retorna o valor da nova potência.

Input: $\alpha, \beta, flag$

Output: Nova Potência de transmissão (pot)

if $flag == aumentar$ **then**

$pot = pot * \alpha$;

if $pot > 5$ **then**

$pot = 5$;

end

end

else

$pot = pot - \beta$;

if $pot < 1$ **then**

$pot = 1$;

end

end

return pot ;

Algorithm 6: Algoritmo para Ajuste de Potência

É importante notar que no começo da competição, todos os nodos estão localizados muito próximos uns dos outros, mas com o passar do tempo eles criam uma formação parecida com uma fila. Isto mostra que no início da competição há uma densidade maior de nodos e com o passar do tempo há uma dispersão dos mesmos. Essa característica afeta o tamanho da vizinhança, o que gera diferenças no congestionamento da rede.

Para este primeiro conjunto de experimentos, o cenário adotado foi caracterizado por uma grande carga de mensagens, pois todos os nodos enviam periodicamente dados com medições dos sinais vitais dos atletas. Além disso, os nodos precisam lidar com a possibilidade

de ter que rotear os dados de outros nodos. Para simular esta situação, foram utilizados diferentes conjuntos de dados (em número de mensagens), como mostrado na Tabela 4.3.

Cada conjunto de dados representa o número total de mensagens enviadas na simulação por todos os nodos. Enquanto os dados com prioridade de entrega representam 10% do total de dados enviados por cada nodo, os outros 90% são representados pelos dados normais (sem prioridade). Neste cenário de aplicação, os dados preferenciais com prioridade seriam resultados de uma medição de alerta nas condições fisiológicas dos atletas, enquanto os dados comuns seriam registros com variações normais.

Tabela 4.3: Conjunto de Dados

Parâmetros de Simulação	Valor (nº de dados)
<i>Workload 1</i>	12500
<i>Workload 2</i>	14000
<i>Workload 3</i>	16500
<i>Workload 4</i>	20500
<i>Workload 5</i>	28500

A avaliação foi executada utilizando dois conjuntos de experimentos. Primeiramente, foi fixada a velocidade na configuração *Slow Speed* (vide Tabela 4.2) e a quantidade de mensagens enviadas foi variada de acordo com a Tabela 4.3. No segundo conjunto de experimentos foi realizado o contrário, ou seja, foi fixada a quantidade de dados na configuração *Workload 2* e a velocidade foi variada da configuração *Slow Speed* à *Very High Speed*. Para ambos os conjuntos de experi-

mentos, foram medidas as taxas de entrega de mensagens e as taxas de atraso. Ademais, observou-se a energia consumida média ao final da simulação. Todos os experimentos foram realizados cinco vezes e os resultados são as médias obtidas. Para o experimento variando a carga de mensagens de dados, o desvio padrão para a taxa de entrega ficou em 2,4% para o RACE 2,1% para o RPAR. Para os resultados quanto a taxa de atraso ficou em 1,9% para o RACE e 2,5% para o RPAR.

Na Figura 4.2 são mostrados os resultados da taxa de entrega de mensagens dados para o experimento em que foi fixada a velocidade e variada a carga de mensagens. O protocolo RACE teve um desempenho melhor que o protocolo RPAR, pois o RACE conseguiu manter uma alta taxa de entrega enquanto a carga de mensagens aumenta. O protocolo RACE também conseguiu lidar com vários sorvedouros escolhendo sempre o que está mais geograficamente mais perto do nodo fonte.

Com a velocidade relativa entre os nodos sendo maior que zero, há a possibilidade de quebra de enlace entre os vizinhos, o que justifica a utilização de um mecanismo de manutenção proativa de vizinhança. Entretanto, esse mecanismo de manutenção de vizinhança não pode onerar a rede. Por esta razão, o protocolo RACE foi projetado para manter a tabela de vizinhança atualizada sem gerar uma sobrecarga muito grande na rede em comparação com protocolos MANETs que mantêm as rotas para os destinos.

Durante o desenvolvimento do RACE, primeiramente, foi testada a estratégia requisição-resposta, para atualização da tabela de vizinhança. Nesta estratégia somente respondem a requisição os nodos com distâncias menores ou iguais ao nodo fonte em relação ao

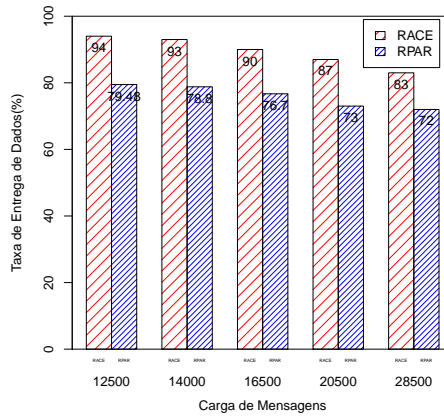


Fig. 4.2: Taxa de Entrega de Dados: Velocidade Fixada e Carga de Mensagens Variável

sink mais próximo do nodo fonte. Entretanto, observou-se que o mecanismo adotado e descrito no item 3.4.1 do capítulo 3 teve um aumento de 22% na taxa de entrega em relação ao método de requisição-resposta, que é adotado pelo protocolo PATH, descrito no capítulo 2.

A intenção do segundo conjunto de experimentos foi estimar a influência da velocidade sobre a taxa de entrega, com a carga de mensagens fixada. O protocolo RACE também teve um melhor desempenho em relação ao RPAR para todos os conjuntos de velocidades, como mostrado na Figura 4.3. Em velocidade baixa, o RACE consegue entregar 93% dos pacotes enviados pelos 300 nodos fontes. Mesmo em movimentação com alta velocidade, o RACE alcançou a taxa de entrega de 87%.

Analisando os dois conjuntos de experimentos, é possível observar que o aumento da velocidade teve um maior impacto no desem-

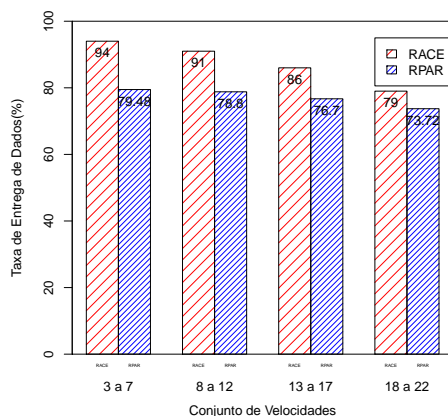


Fig. 4.3: Taxa de Entrega de Dados: Carga de Mensagens Fixada e Velocidades Variáveis

penho do protocolo em relação ao aumento da carga de mensagens. O protocolo RACE teve um desempenho melhor que o RPAR com perdas de mensagens de dados de até 17% e o RPAR de até 28% em relação ao primeiro experimento variando a carga de mensagem.

Mediu-se também a taxa de atraso dos pacotes com prioridade nos dois conjuntos de experimentos discutidos. O protocolo RACE teve um melhor desempenho em relação ao protocolo RPAR, como mostrado nas Figuras 4.4 e 4.5. Foi fixado um tempo de referência de 100 milissegundos para determinar a taxa de atraso das mensagens. As mensagens que chegassem acima do tempo de referência estabelecido seriam contabilizadas como atraso. Para o RACE, a taxa de atraso dos pacotes com prioridade foi de, no máximo, 4% em ambos os experimentos, com carga de mensagens e velocidades variáveis. Por outro lado, o RPAR foi sensível a variações de carga de mensagens e velocidade, e teve uma perda que variou de 34% a 46% no primeiro experi-

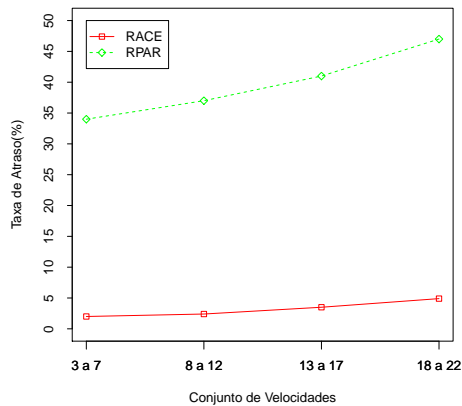


Fig. 4.4: Taxa de Atraso com Variação de Velocidade

mento (variação de carga de mensagens) e de 34% a 44% no segundo experimento (variação da velocidade). Outra informação importante, é que a média de saltos realizados pelos pacotes para atingir o sorvedouro, tanto para o RACE quanto para o RPAR, foi de 6 saltos, sendo que o máximo de saltos observado foi 10.

O comportamento do protocolo RACE em relação às taxas de atraso pode ser explicado pelo uso do componente de Diferenciação de Serviço, que prioriza a transmissão de pacotes com prioridade associada. Por isso, nos dois experimentos com variação de velocidade e carga de mensagens, o protocolo RACE consegue sustentar uma taxa quase linear de atraso.

A última questão que deve ser analisada é em relação ao consumo de energia. Embora o consumo de energia não seja uma questão muito importante para o cenário de aplicação da corrida, pois os nodos seriam utilizados no máximo por 16 horas (tempo de duração de uma

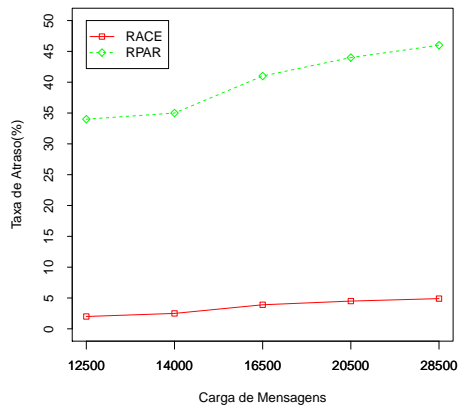


Fig. 4.5: Taxa de Atraso com Variação de Carga de Mensagens

competição como *triathlon*), é desejável que os nodos sejam energeticamente eficientes para que possam ser reutilizados em outras competições sem necessidade de recarga. Dessa forma, foi mensurada a energia média consumida e o resultado pode ser visualizado na Figura 4.6. Como se pode ver, a média de consumo de energia do protocolo RACE foi próxima ao protocolo RPAR, o qual foi especialmente projetado para ser um protocolo energeticamente eficiente. A diferença de consumo entre os dois protocolos foi menor que 1% a favor do protocolo RPAR. O modelo de consumo de energia adotado, que foi implementado pelo *framework* INETMANET, foi baseado em [Landsiedel, Wehrle e Götz 2005].

Na subseção seguinte são apresentados os detalhes do segundo conjunto de experimentos para avaliar o impacto da métrica *buffer*.

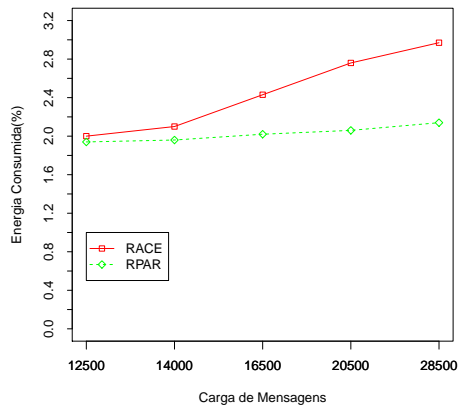


Fig. 4.6: Média de Energia Consumida em mW-sec

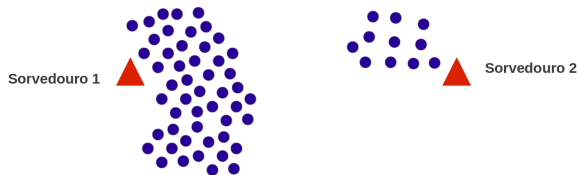


Fig. 4.7: Redistribuição dos Nodos Fontes.

4.2.2 Segundo Conjunto de Experimentos: O Impacto da Métrica *Buffer*

Os experimentos seguintes foram projetados para avaliar em detalhes o impacto da métrica de *buffer*. Para isso, a configuração dos primeiros experimentos foi alterada para criar regiões mais congestionadas que outras. O número de nodos sorvedouros foi diminuído para 2 e os nodos fontes foram redistribuídos de forma a ter uma maior concentração de nodos próxima ao nodo sorvedouro 1 e uma concentração menor próxima ao nodo sorvedouro 2, como é mostrado na Figura 4.7.

Além disso, o experimento durou 100 segundos, para medir a taxa de entrega de mensagens, antes que os nodos começassem a se dispersar e formar o efeito de fila observado no experimento anterior. A dispersão dos nodos provoca o descongestionamento da rede, visto que a rede fica esparsa. Com redes esparsas, há *buffer* disponível para processar as mensagens, sendo assim, manter parte da rede densa é o interesse deste experimento para tornar escassa a disponibilidade de *buffer*. Igualmente ao primeiro experimento foram realizados cinco repetições e os resultados são as médias obtidas. Para o experimento com o nodo sorvedouro 1 o desvio padrão foi de 2,3% e para o nodo sorvedouro 2 foi 2,1%.

Os pesos das métricas foram reconfigurados a fim de verificar a relevância de cada métrica nas duas situações criadas. As duas situações foram: i) 50 nodos fontes próximos ao nodo sorvedouro 1, assim cria-se uma situação de congestionamento; ii) 10 nodos fontes próximos ao nodo sorvedouro 2, a fim de criar uma rede esparsa. Os pesos foram configurados de acordo com a Tabela 4.4.

Tabela 4.4: Pesos das Métricas

Configuração das Métricas	Pesos
BUFFER	$buffer = 0.8; lqe = 0.1; velocity = 0.1$
LQE	$buffer = 0.1; lqe = 0.8; velocity = 0.1$
VELOCITY	$buffer = 0.1; lqe = 0.1; velocity = 0.8$

Neste experimento, foram geradas cargas de mensagens maiores a fim de verificar os limites do protocolo RACE. A Tabela 4.5 de cargas de mensagens geradas mostra as configurações. Outra diferença

importante, em relação ao experimento da subseção 4.2.1, é que toda a carga de mensagens foi destinada a somente um nodo sorvedouro. Dessa forma, nos experimentos desta seção, os nodos sorvedouros concentraram muito mais carga de mensagens, como é mostrado na Tabela 4.5.

Quanto à velocidades dos nodos, foi utilizada a configuração *Slow Speed* da Tabela 4.2. As medições foram realizadas com base nos dois nodos sorvedouros, ou seja, para cada conjunto de nodos próximos ao nodo sorvedouro, as configurações de métricas da Tabela 4.4 foram utilizadas. Com isso, pode-se ter uma ideia da influência de cada métrica sobre as duas situações da rede.

Os resultados do experimento são mostrados nas Figuras 4.8 e 4.9. Os resultados para o nodo sorvedouro 1, representados na Figura 4.8, mostram que a métrica de *buffer* nas três configurações de carga de mensagens superou as métricas *velocity* e *lqe*. Para uma carga de mensagens menor, o RACE teve uma alta taxa de entrega para as três métricas, chegando a 97.3% para a métrica de *buffer*. Para cargas de mensagens maiores, o RACE teve uma baixa taxa de entrega, entre 11% e 23%, para *workload* 2 e 3.

Entretanto, a métrica de *buffer* fez mais diferença que a métrica puramente *velocity*, que leva mais em consideração o posicionamento geográfico. A métrica de *buffer* pode entregar a uma alta carga de mensagens destinadas somente a um nodo sorvedouro cerca de 9.7% e 6% a mais que a métrica *velocity* e cerca de 2.9% a mais que a métrica *lqe*. Uma informação importante para o experimento com o nodo sorvedouro 1 foi que, com 50 nodos, foi observado uma vizinhança de quatro saltos. O tamanho da vizinhança é importante, mesmo que a

carga de mensagem seja baixa, pois quanto maior a vizinhança, maior a interferência, o que reflete nas condições da rede.

Tabela 4.5: Conjunto de Dados Para os Nodos Sorvedouros

	Número Total de Dados		
	<i>Workload 1</i>	<i>Workload 2</i>	<i>Workload 3</i>
Nodo Sorvedouro 1	2500	11000	27000
Nodo Sorvedouro 2	500	1600	3500

Os resultados obtidos para o nodo sorvedouro 2 são mostrados na Figura 4.9. A quantidade reduzida de nodos próximos ao nodo sorvedouro 2 gerou uma baixa carga de mensagens, apesar de todos os nodos serem nodos fontes. Nota-se que a métrica *velocity*, neste caso, superou as outras duas métricas, mantendo uma alta taxa de entrega de dados. Já as métricas de *buffer* e *lqe* não foram eficientes na situação onde não há um congestionamento e há uma baixa carga de mensagens. Além disso, foi observado que a vizinhança do nodo sorvedouro 2 teve dois saltos.

Com esse experimento, pode-se concluir que as métricas devem ser dosadas de acordo com a situação da rede. Em cenários de uma maior densidade da rede com alta carga de mensagens, deve-se levar em consideração a situação de *buffer* da vizinhança e a qualidade do enlace para escolha do melhor nodo retransmissor. Para o caso de uma rede esparsa, a métrica *velocity* é a melhor opção para a escolha do melhor nodo vizinho.

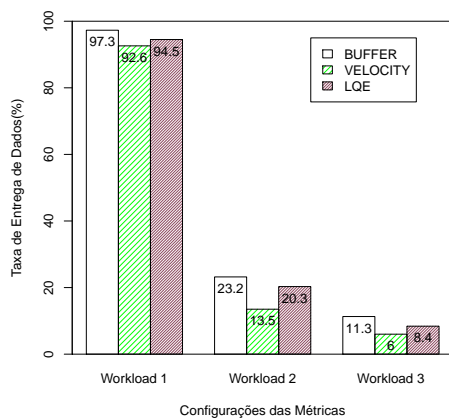


Fig. 4.8: Taxa de Entrega de Dados Para o Nodo Sorvedouro 1 com Reconfiguração de Métricas.

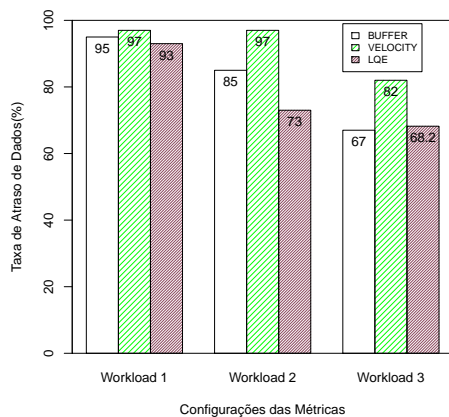


Fig. 4.9: Taxa de Entrega de Dados Para o Nodo Sorvedouro 2 com Reconfiguração de Métricas.

Na próxima subsecção são apresentadas algumas comparações com as mesmas configurações de simulação da subsecção 4.2.1 para os

protocolos de roteamento MANETs.

4.2.3 Terceiro Conjunto de Experimentos: Análise de Desempenho dos Protocolos MANETs no Cenário da Corrida

Os protocolos para redes MANETs foram projetados especificamente para ambiente de mobilidade. Por esse motivo, alguns dos protocolos descritos no capítulo 2 foram analisados no mesmo cenário da corrida descrito na subseção 4.2.1. É importante notar que os protocolos MANETs foram projetados tanto para ambientes *indoor* como *outdoor*, pois não necessitam de nenhum sistema de localização. Apesar de serem de diferentes classes, é fundamental entender o desempenho dos protocolos MANETs frente aos protocolos geográficos sob o mesmo cenário.

As configurações da simulação foram as mesmas descritas na subseção 4.2.1 com 300 nodos fontes e 8 nodos sorvedouros. Quatro protocolos MANETs foram testados, sendo dois reativos e dois proativos:

1. Protocolos Reativos:

- (a) AODV (*Ad hoc On-Demand Distance Vector*)
- (b) DYMO (*Dynamic MANET On-demand*)

2. Protocolos Proativos:

- (a) OLSR (*Optimized Link State Routing*)
- (b) BATMAN (*Better Approach To Mobile Adhoc Networking*)

Os resultados podem ser vistos nas Figuras 4.10, 4.11, 4.12 e 4.13. Foram utilizadas as métricas de taxa de entrega de dados e atraso fim-a-fim para avaliar os protocolos.

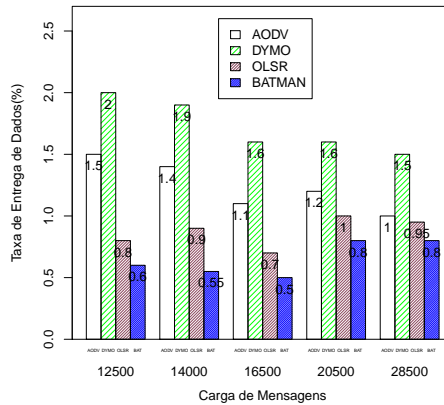


Fig. 4.10: Taxa de Entrega de Dados Variando a Carga de Dados

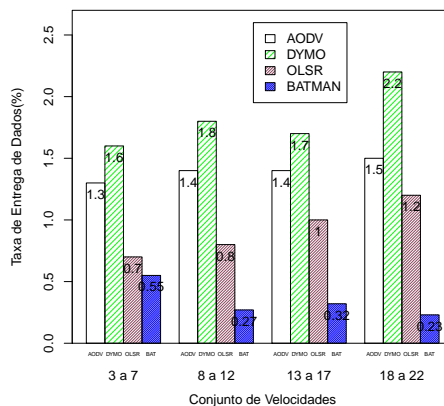


Fig. 4.11: Taxa de Entrega de Dados Variando as Velocidades

Primeiramente, nota-se que os resultados para os protocolos MA-

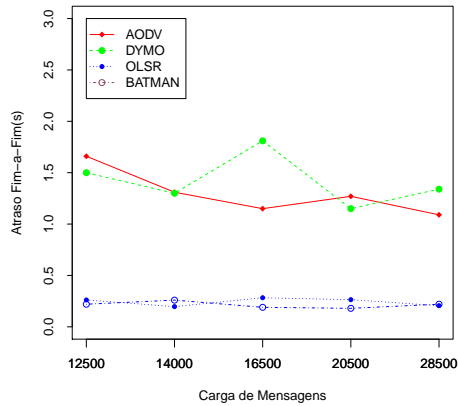


Fig. 4.12: Atraso fim-a-fim Variando a Carga de Dados

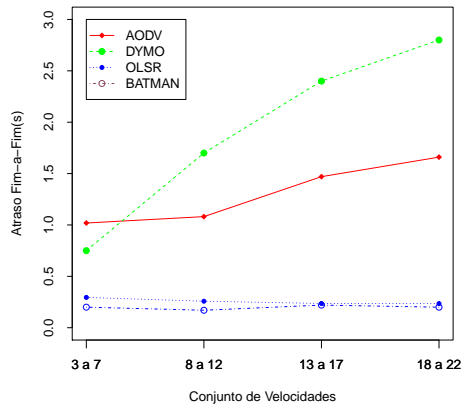


Fig. 4.13: Atraso fim-a-fim Variando as Velocidades

NETs na configuração do cenário da corrida foram muito inferiores comparados aos resultados dos protocolos de roteamento geográfico mostrados nas figuras 4.2, 4.3, 4.4, 4.5 da subseção 4.2.1. As taxas de entrega de dados para todos os protocolos MANETs foram inferiores

a 3% para todas as configurações de velocidades e carga de dados. Entretanto, o protocolo que teve a melhor taxa de entrega foi o DYMO seguido do AODV e o pior desempenho foi do protocolo BATMAN.

A variação da velocidade teve mais efeito sobre o DYMO e o AODV. Contudo, este efeito foi positivo, pois aumentou a sua taxa de entrega. Por outro lado, o efeito foi negativo sobre os protocolos proativos BATMAN e OLSR, diminuindo a sua taxa de entrega. A melhor taxa de entrega de dados foi de 2.2% para o DYMO no experimento variando a velocidade e a pior taxa de entrega de dados, de 0.23%, foi para o protocolo BATMAN variando a velocidade.

Para os resultados de atraso fim-a-fim, os protocolos proativos, OLSR e BATMAN tiveram o melhor resultado, não apresentando muita influência variando a carga ou velocidade. Os protocolos reativos tiveram um desempenho inferior em relação ao atraso fim-a-fim e apresentaram uma maior influência variando a velocidade piorando os resultados. Apesar dos protocolos reativos ficarem com o desempenho inferior em relação ao atraso fim-a-fim, as suas taxa de entrega chegaram a ser, em alguns casos, 100% superiores. Além disso, o resultado do atraso fim-a-fim foi inferior para os protocolos reativos, pois o aspecto de formar rotas por demanda tem um custo temporal para a entrega de dados.

É importante ressaltar que os protocolos para redes MANETs foram projetados para redes esparsas com baixa carga de mensagens, por isso, os resultados foram muito inferiores aos dos protocolos geográficos. Entretanto, a experimentação destes protocolos sobre a mesma configuração do cenário da corrida é fundamental para termos a noção de que os protocolos são projetados visando um cenário de aplicação.

Com esses resultados, confirmamos que, para cenários com grande quantidade de nodos fontes, os protocolos geográficos são mais adequados devido à baixa carga de mensagens de controle geradas. Entretanto, sabe-se que há limitações dos protocolos geográficos, que estão diretamente relacionadas às limitações ao uso de GPS.

Na próxima subseção são apresentadas as conclusões do trabalho desenvolvido até o momento.

4.2.4 Conclusão

O desenvolvimento do protocolo RACE proporcionou um melhor entendimento de como a mobilidade e a geração de carga de mensagens por vários nodos fontes podem afetar os protocolos de rede, em especial, os de roteamento. Notou-se que no cenário da corrida, com grande número de nodos fontes, sendo que todos os nodos também desempenham o papel de roteador, o protocolo RACE teve um desempenho satisfatório na entrega de dados e apresentou uma baixa taxa de atraso para mensagens com prioridade.

Além disso, notou-se que os pesos das métricas e as condições de rede têm uma relação direta para o desempenho do protocolo RACE. No caso de grande carga de mensagens de dados, alguns fatores são responsáveis pela perda de pacotes e redução no desempenho dos protocolos de rede, como interferência da vizinhança (o que gera colisão), desistência de aquisição do canal e tamanho do *buffer*. A qualidade do enlace, que basicamente verifica a taxa de pacotes transmitidos com sucesso (PRR - *Packet Reception Rate*) e a quantidade de *buffer* são elementos considerados pelo protocolo RACE para re-

presentar alguns dos fatores responsáveis pela redução de desempenho dos protocolos de rede.

Com o segundo conjunto de experimentos, perceber-se a influência que a métrica de *buffer* apresenta em determinada situação da rede. Em casos com grande quantidade de mensagens e vários nodos fontes, a métrica de *buffer* aumentou o desempenho do RACE em detrimento das outras métricas. Entretanto, nos casos em que a rede é esparsa e a quantidade de mensagens é pequena, o *buffer* deixa de ser um problema pois, neste caso, haverá *buffer* disponível para processar as mensagens. Sendo assim, a métrica de *buffer* não faz diferença, deixando que a característica gulosa dos protocolos geográficos torne-se mais importante para o desempenho do protocolo.

Em relação ao último conjunto de experimentos pode-se confirmar a superioridade dos protocolos de roteamento geográficos diante dos protocolos MANETs no que tange o desempenho em condição de rede com grande número de nodos fontes. Pode-se concluir que, em cenários com grande carga de nodos, os protocolos MANETs não são apropriados, pois os seus desempenhos são muito baixos. Entretanto, com uma baixa carga, os protocolos MANETs melhoram bastante os seus desempenhos. Essa última observação poderá ser notada no experimento realizado no capítulo 7, no estudo de caso com o protocolo AODV, seção 7.2, no qual observa-se que, com 1 nodo fonte, o AODV pode entregar cerca de 70% de mensagens. Contudo, no mesmo experimento com dez nodos fontes, o desempenho do AODV cai para cerca de 15% de taxa de entrega de mensagens. Sendo assim, podemos concluir que a carga de mensagens de controle e de dados têm mais influência sobre o desempenho dos protocolos do que a mobilidade.

Apesar do melhor desempenho dos protocolos de roteamento geográfico, eles se baseiam em uma premissa muito forte, que é o conhecimento do posicionamento, como o uso de um GPS. Essa premissa não leva em consideração os recursos computacionais envolvidos e as limitações em cenários *indoor*, além de considerar uma alta precisão de posicionamento. Nos experimentos com o protocolo RACE, o custo de energia medida não levou em consideração o uso do GPS, somente a contabilização do envio de mensagens. Entretanto, o envio de mensagens é um fator que consome mais energia que o processamento de dados, sendo assim, o resultado pode apontar para uma tendência no consumo de energia.

A utilização de GPS pode drenar a bateria do nodo sensor. Essa seria a maior limitação de todas as propostas de roteamento geográfico para RSSF. Seria necessário investigar o quanto o uso do GPS requer de energia e fazer uma relação com a acurácia que se precisa ter. Contudo, no caso do cenário da corrida, se espera que os nodos sejam utilizados por, aproximadamente, 16 horas. Dessa forma, a bateria do nodo pode durar por esse tempo. Se tais questões referentes ao uso de GPS forem solucionadas no futuro, pode-se dizer que os roteamentos baseados em posicionamento são os mais indicados em cenários *outdoor*.

No próximo capítulo, será apresentada a segunda parte da tese que concerne ao estudo e desenvolvimento de métodos de predição de conectividade em ambientes móveis. Os métodos propostos foram estendidos e comparados com métodos similares da literatura. A apresentação da segunda parte da tese está estruturada em: (i) estado da arte de mecanismos de predição de qualidade de enlace para ambiente móveis, (ii) apresentação do modelo proposto, (iii) avaliação do modelo proposto, (iv) estudo de caso e (v) conclusões finais.

Parte III

MECANISMO DE PREDIÇÃO DE CONNECTIVIDADE LOCAL

Capítulo 5

Trabalhos Relacionados a Mecanismos de Predição de Conectividade para Ambientes Móveis

Neste capítulo é apresentado o estado da arte sobre mecanismos de predição de conectividade para ambientes móveis. Além da apresentação dos trabalhos relacionados, são analisados os aspectos das tecnologias e modelos utilizadas para dar suporte aos mecanismos de predição de conectividade, bem como as suas vantagens e limitações de acordo com os cenários.

5.1 Considerações Iniciais

Antes de apresentar os trabalhos relacionados, é importante discutir alguns aspectos que são inerentes a todas as propostas que lidam com predição de conectividade. Tais aspectos podem ser divididos em três tipos: i) sistemas de posicionamento, ii) histórico de dados, iii) tipo de modelo utilizado para executar a predição de qualidade de enlace.

O primeiro aspecto refere-se à utilização de sistemas de posicionamento como GPS ou GPS-*free*. A utilização de GPS têm algumas desvantagens, como custo para implementação em larga escala, o esgotamento de bateria de dispositivos devido à captação do sinal GPS e as aplicações são restritas a cenários *outdoor*. Entretanto, os sistemas GPS-*free* superam uma destas limitações.

Os sistemas GPS-*free* foram projetos para cenários *indoor*, contudo, esses sistemas requerem antenas extras, como o *Time Difference of Arrival* (TDOA), ou dispositivos como ultrassom, como na solução Cricket Compass [Priyantha et al. 2001], ou ainda, o uso de nodos âncoras. Os nodos âncoras são nodos estacionários que sabem previamente a sua posição e transmitem para toda a rede a informação de posicionamento. Dessa forma, os nodos móveis podem calcular sua própria posição por triangulação.

As desvantagens dos sistemas GPS-*free* vão da necessidade de antenas ou dispositivos extras até a necessidade de inundar a rede periodicamente com o posicionamento dos nodos âncoras. Neste último caso, a estratégia deve fazer um *trade-off* entre manter a precisão do posicionamento e ter o cuidado para não sobrecarregar a rede.

A segunda característica interessa a todas as abordagens que necessitam de histórico de dados. O registro de dados pode ser proveniente de diferentes variáveis, tal como qualidade de enlace (RSSI ou SNR) ou o registro de mudanças no posicionamento geográfico de um nodo em movimento. O histórico de registro de dados é gravado em modo *off-line* para ser processado posteriormente.

As abordagens que necessitam do histórico de dados são alimentadas por um padrão de comportamento contido no histórico. Desse modo, após o período de processamento, é esperado que o padrão corrente tenha o mesmo comportamento quando os dados foram coletados. A principal desvantagem dessa abordagem está relacionada a possíveis modificações de padrões. Se isto ocorrer, um novo histórico de dados é necessário.

O terceiro aspecto refere-se ao modelo escolhido para estimar a conectividade. Existem três modelos principais encontrados na literatura que são largamente utilizados para realizar as previsões de conectividade: i) Modelos de Markov, ii) Séries Temporais e iii) Velocidade Relativa.

Os modelos de Markov e as Séries Temporais necessitam de histórico de registro de dados para computar a estimativa, já o modelo de Velocidade Relativa necessita de um sistema de posicionamento preciso. A principal desvantagem dos modelos de Markov é que, dependendo de como o processo é modelado, o número de estados fica muito grande. Por exemplo, se o estado é considerado como um nodo, e o cenário tem um grande número de nodos, o tamanho do modelo pode crescer na proporção do tamanho da rede. Entretanto, esse fator depende de como o problema é modelado.

As Séries Temporais podem ter o mesmo problema em relação à quantidade de informação armazenada em memória. Neste caso, todo o histórico de dados precisa ser carregado na memória para ser processado. Isto pode ser um fator limitante para dispositivos com recursos computacionais restritos, como os nodos de redes RSSF. O uso do modelo de Velocidade Relativa requer um sistema de posicionamento para dar a informação de coordenada a fim de calcular o vetor de velocidade de cada nodo. As limitações com o modelo de Velocidade Relativa estão relacionadas às limitações dos sistemas de posicionamento, como discutido anteriormente.

O problema de predição de conectividade para cenários móveis tem sido tratado na forma de prover o conhecimento de durabilidade de conectividade entre nodos. Visto que a conectividade é o elemento básico para a comunicação nas redes, é imperativo criar mecanismos que possam lidar com a volatilidade do enlace em cenários móveis.

A questão de conectividade pode ser tratada de maneira global. Isto requer o conhecimento de toda a topologia da rede. Esse conhecimento deve ser constantemente atualizado em caso de mobilidade dos nodos para refletir as mudanças de topologia. Com a visão global da topologia da rede, pode-se estimar quando um nodo se desconectará da rede, pelo conhecimento, por exemplo, da rede k -conectada [Atay e Bayazit 2007].

Entretanto, o trabalho apresentado nesta tese, bem como a maioria dos trabalhos relacionados descritos nesta seção, trata da questão de predição de conectividade de maneira local. Na abordagem local, a conectividade é vista nodo a nodo na vizinhança de um salto. Pode-se dizer que as duas abordagens são complementares, pois se pode

construir um grafo da rede, no qual o peso pode ser a estimativa de conectividade calculada nodo a nodo.

Nas próximas seções são discutidos os trabalhos relacionados ao tratamento de estimação da conectividade local.

5.2 Mobility Prediction in Wireless Networks

O trabalho desenvolvido por [Su, Lee e Gerla 2000] foi um dos precursores na criação de mecanismos para realizar a predição de conectividade em ambientes móveis. Os autores apresentam um modelo determinístico para prever a conectividade como mostra a equação 5.1. A equação proposta calcula a conectividade restante entre dois nodos móveis. O conjunto de informações adotado é composto pelo conhecimento de posicionamento, provido por GPS, ângulo θ entre os nodos, velocidade v e a área de alcance do sinal r . Como premissa, este modelo necessita do posicionamento provido por um dispositivo GPS e é necessário o conhecimento da área de alcance do sinal.

$$Dt = \frac{-(ab + cd) + \sqrt{(a^2 + c^2)r^2 - (ad - bc)^2}}{a^2 + c^2} \quad (5.1)$$

onde Dt é a quantidade de tempo em que dois nodos móveis estarão conectados, e

$$a = v_i \cos \theta_i - v_j \cos \theta_j$$

$$b = x_i - x_j$$

$$c = v_i \sin \theta_i - v_j \sin \theta_j$$

$$d = y_i - y_j$$

Este trabalho foi implementado e testado com três modelos de mobilidade e comparado com outras propostas da literatura e com a proposta da tese. Apesar de ser um modelo determinístico, é importante saber como ele se comporta quando os modelos de mobilidade apresentam certo grau de comportamento randômico. Mais detalhes a respeito dos experimentos realizados são descritos no capítulo 7 de avaliação da proposta de predição de conectividade.

5.3 Disconnection prediction in mobile ad hoc networks for supporting cooperative work

Em [De Rosa, Malizia e Mecella 2005], os autores desenvolveram uma técnica para predição de desconexão de enlace visando aplicações em MANETs. Os autores assumiram que todos os nodos tem o conhecimento de seus posicionamentos provido por um sistema *GPS-free*. No momento da inicialização, todos os nodos precisam estar conectados para a construção de um grafo, o que pode sinalizar uma limitação desta proposta.

O peso para cada aresta representa a distância entre os nodos vizinhos. Para calcular e prever a distância entre os nodos, a proposta utiliza cadeia de Markov baseada no domínio de espaço para construir a matriz. Cada nodo da rede é um elemento da matriz. Dessa forma, um coordenador calcula qual será a próxima posição entre os nodos.

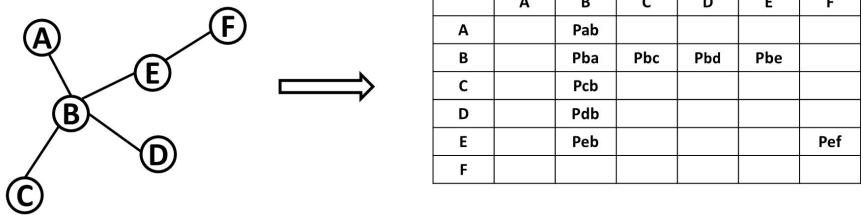


Fig. 5.1: Topologia de Rede Representada na Matriz de Transição de Estados.

Na figura 5.1 é mostrado um exemplo de uma topologia, na qual os nodos estão conectados entre si. Nesta topologia de exemplo, cada nodo representa um estado da cadeia de Markov e cada enlace representa a transição do autômato. A matriz de transição gerada a partir da topologia também é mostrada na figura.

Esta é uma proposta baseada em conhecimento global da rede para estimar a conectividade entre nodos móveis. O fato de ser preciso ter todo o conhecimento da topologia pode gerar sobrecarga na rede para manter o conhecimento atualizado. Ademais, como o modelo considera que cada nodo da rede seja um estado na cadeia de Markov, então o tamanho da cadeia cresce de acordo com o tamanho da rede. Essa última característica pode ser limitante para o uso da solução em redes maiores ou com número variável de nodos.

5.4 Effective Link Operation Duration: a New Routing Metric for Mobile Ad Hoc Networks

No trabalho realizado em [Chen, Jones e Jayalath 2007], os autores propõem medir a durabilidade do enlace entre os nodos. O es-

queima de predição adotado combina o conhecimento do posicionamento com a informação do enfraquecimento do sinal. Com o conhecimento do posicionamento pode-se obter o conhecimento da mobilidade do nodo. A mobilidade do nodo é representada pela sua velocidade e direção.

Primeiramente, os autores utilizam o conhecimento da mobilidade do nodo para prever o tempo de vida do enlace. Então, o tempo de vida do enlace é combinado à estatística do enfraquecimento do sinal para obter a duração do enlace. A estatística de enfraquecimento do sinal é calculada por meio do modelo de enfraquecimento do sinal *free-space path loss*.

Para calcular a duração do enlace, assume-se que todos os nodos tenham GPS para prover as informações de localização e velocidade. Assim, combinando o conhecimento da velocidade relativa, utilizando a equação 5.1 do trabalho [Su, Lee e Gerla 2000], com a estatística de enfraquecimento do sinal (*free-space path loss*), a duração do enlace é obtida. As limitações desta proposta recaem sobre as mesmas limitações do trabalho predecessor em [Su, Lee e Gerla 2000] que serão discutidas no capítulo 7 de avaliação da proposta.

5.5 Breadcrumbs: Forecasting mobile connectivity

Em [Nicholson e Noble 2008], é apresentado um modelo para fazer reserva de banda antecipada para uma rede WLAN, levando em consideração o histórico da mobilidade do nodo que é proveniente do registro de mudanças no posicionamento geográfico do nodo em movimento. Esse histórico é utilizado como entrada para um modelo de

Markov, que visa prever a mobilidade do nodo. No modelo, cada estado é considerado como uma localização fixa onde há algum roteador para prover serviço de acesso. Os nodos móveis transitam de uma área para outra, dessa forma, construindo um histórico de mudança de área.

Este esquema é baseado em predição de localização, considerando que os nodos e roteadores tenham o conhecimento de posicionamento por GPS. O modelo é construído baseado na probabilidade de mudança de localização. Por exemplo, se um nodo se move de uma área para outra, os roteadores podem prever a mudança e, assim, antecipar a reserva de banda.

As desvantagens destas propostas são a necessidade de utilização do dispositivo GPS, bem como a limitação do modelo estar atrelado às áreas que contém roteadores fixos. Caso seja necessário incluir novas áreas, o modelo estará desatualizado. Além disso, o tamanho do modelo cresce de acordo com a quantidade de áreas.

5.6 *Link quality prediction in mesh networks*

Em uma abordagem diferente, os autores de [Farkas et al. 2008] utilizam o conhecimento da qualidade do enlace, com SNR, para prever a conectividade entre os nodos. Baseando-se no modelo de Séries Temporais, os autores registraram as mudanças na qualidade do enlace, ao longo do tempo, enquanto os nodos se moviam. A abordagem necessita de histórico de registro do SNR.

Em um cenário pré-determinado, o histórico das variações do SNR, devido à mobilidade do usuário, foi utilizado como entrada para

construção do modelo. Por essa razão, não foi necessária a utilização de nenhum *hardware* adicional para adquirir a informação de posicionamento. Entretanto, o SNR está sujeito a perturbação do ambiente, gerando variações aleatórias.

Por isso, os autores desenvolveram uma variação de filtro de kalman, baseada em janela, para obter um sinal livre de ruído. O filtro desenvolvido é uma variação do filtro de kalman, pois o modelo de variação do SNR não segue um modelo conhecido como um modelo Gaussiano. Ao invés de utilizar um modelo Gaussiano, o modelo de auto-regressão (AR) [McQuarrie e Tsai 1998] de ordem 1 foi utilizado. A figura 5.2 representa o fluxo de funcionamento do algoritmo desenvolvido. O algoritmo funciona da seguinte forma:

1. O valor puro do SNR é amostrado e enfileirado em uma janela de tamanho N .
2. Pelo modelo auto-regressão, utilizando o método de quadrados mínimos [Jain 1991], a estimativa do SNR é calculada.
3. Com o resultado da estimativa do SNR, a covariância do ruído é calculada.
4. Com a covariância do ruído, a covariância do erro de estimativa é calculada.
5. Por fim, o processo de filtragem do ruído faz uma combinação linear da covariância do erro de estimativa com a estimativa do SNR. Assim, gera-se o SNR filtrado.
6. A cada vez que a janela é atualizada com novos valores, um novo valor do SNR filtrado é gerado.

Pode-se considerar que a principal colaboração deste trabalho foi o desenvolvimento de um mecanismo de filtragem de ruído *on-the-fly*, no qual não é necessário ter o conhecimento prévio do modelo do sinal. Entretanto, a principal reserva quanto à abordagem, é que ela não considera enlaces futuros. Isto significa que, caso novos nodos apareçam no ambiente, o modelo não poderá prever a conectividade entre eles. Além disso, se o padrão de mobilidade mudar, o modelo ficará desatualizado e será necessária uma nova etapa de treinamento *off-line*.

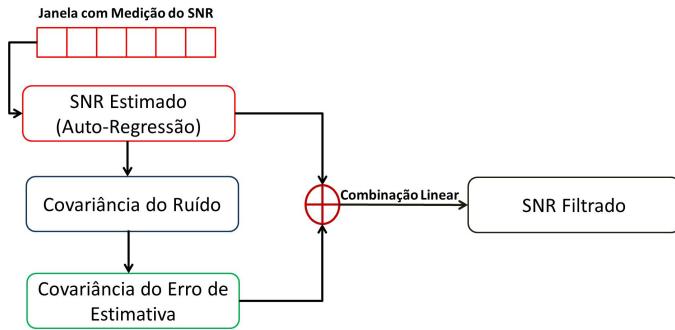


Fig. 5.2: Fluxo do Algoritmo Filtro de Kalman Baseado no Modelo de Auto-Regressão.

5.7 Performance Enhancement of Fuzzy Logic Based Transmission Power Control in WSN using Markov Based RSSI Prediction

A abordagem apresentada em [Sabitha e Thangavelu 2011] (MTCP) representa as variações de RSSI utilizando um modelo de Markov com cinco estados. Os estados basicamente representam o aumento, diminuição, manutenção e um valor mínimo do RSSI. O quinto estado seria

o recebimento de um pacote controle. O modelo é usado para prever a qualidade do enlace entre dois nodos.

Os estados do modelo formam um grafo quase completamente conectado, em que, para cada estado, há uma transição para todos os outros estados. Para facilitar a visualização do modelo, ao invés de mostrar um automato, é mostrado a figura 5.3 que representa a matriz de transição gerada pelo autômato. Os elementos da matriz são as probabilidade de transição dos estados.

	A	B	C	D	E
A	Paa	Pab	Pac	Pad	Pae
B		Pba	Pbc	Pbd	Pbe
C	Pca	Pcb	Pcc	Pcd	Pce
D	Pda	Pdb	Pdc	Pdd	Pde
E	Pea	Peb	Pec		Pe

Fig. 5.3: Matriz de Transição de Estados do Modelo MTCP.

A intenção dos autores é utilizar o MTCP para melhorar o mecanismo de *backoff* do protocolo MAC. Essa melhoria teria reflexo na economia de energia e também no aumento de largura de banda disponível. Os resultados obtidos são promissores considerando a predição para um passo à frente.

Como os autores proviram um modelo que para realizar a predição da variação do RSSI, entende-se que esta possa ser uma proposta válida para prever a conectividade de nodos em movimento. Sendo assim, este modelo foi implementado e testado igualmente com três modelos de mobilidade diferentes, a fim de verificar a sua adequação

ao propósito de predição de conectividade em cenários móveis.

5.8 *BD Model*

O modelo de Markov *Birth-Death* (BD) apresentado em [Guha e Sarkar 2008] tem a vantagem de também evitar o uso de informações de posicionamento. Em vez disso, ele prevê a qualidade do enlace para rede WiFi por meio do SNR. Os estados do modelos são determinados por faixas de valores do SNR. A figura 5.4 mostra a ideia de como o SNR é delimitado em faixas de valores, sendo que cada faixa de valor é representado por um estado que contem um conjunto de valores do SNR.

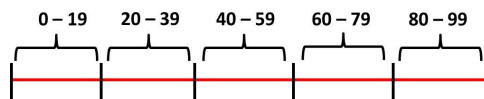


Fig. 5.4: SNR Dividido em Faixas de Valores para Determinar os Limites dos Estados.

Assumindo-se uma periodicidade de envio de mensagens a cada segundo, pode-se obter o histórico de variação do SNR para construir a matriz de transição. Para construir a matriz de transição dos estados, a frequência nas mudanças entre as faixas de valores do SNR são contabilizadas. Assume-se-se que a ocorrência de transições seja suave entre os estados, ou seja, as transições somente ocorrem entre os estados adjacentes.

Os autores mostraram, por meio de experimentos que, para caracterizar o canal em um modelo de Markov, é necessário ter mais que dois estados para representar a variação estatística do canal sem fio. O

autômato que representa os estados e transições do modelo é ilustrado na figura 5.5. Cada estado representa uma faixa de valor do SNR. Dessa forma, o estado mais à esquerda representa a faixa de menor valor e o estado mais à direita representa a faixa de maior valor.

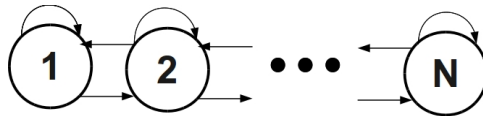


Fig. 5.5: Autômato representando os estados e transições do modelo *birth-death*.

Apesar de o modelo representar a variação da qualidade do enlace devido a interferência ou mobilidade, ele não representa adequadamente o movimento dos nodos. Ele ignora o princípio da inércia, que estabelece que um objeto não pode mudar sua direção imediatamente. Tal problema é expresso pelo modelo de Markov proposto, o que sugere que um nodo tem a mesma probabilidade de aumentar ou diminuir a intensidade do sinal, ou seja, manter ou mudar de direção. No entanto, na verdade, a probabilidade de o nodo continuar na mesma direção é muito mais elevada do que de mudar a sua direção.

A direção neste caso, entende-se como o sentido do movimento, que é expresso pelo fato de os nodos se aproximarem ou se afastarem. Sendo assim, independente da direção angular de os nodos, a tendência de se continuar no mesmo sentido (afastando ou aproximando) é maior que do mudar. Pelo exemplo a seguir, pode-se demonstrar que a probabilidade de o nodo continuar na mesma direção (afastando ou aproximando) é maior que ele mudar de direção:

1. Considere dois nodos A e B em movimento, mas em sentidos opostos: O nodo A se movimenta da esquerda para direita e o

nodo B o contrário.

2. Considere, também, que em certo momento, eles entram em contato um com o outro, ou seja, eles entram no raio de ação do sinal, como mostrado na figura 5.6.
3. Para verificar o contato entre os nodos A e B, eles enviam mensagens periódicas a cada 1 segundo.
4. A partir do primeiro contato (figura 5.6) até os nodos se desconectarem (figura 5.7) o histórico do SNR é contabilizado.
5. De maneira simplificada, o histórico poderia ser da seguinte forma: 10-20-30-40-50-60-70-60-50-40-30-20-10, sendo 10 o menor valor do SNR e 70 o maior valor.
6. Neste caso, verifica-se que a tendência do SNR continuar aumentando de 10 para 70 (aproximando) ou continuar diminuindo de 70 para 10 (afastando) é maior que a mudança de sentido (aproximação para afastamento) mostrada pela redução do SNR de 70 para 60. Sendo assim, uma vez que o nodo está em um sentido, a probabilidade dele continuar neste sentido é maior que mudar.

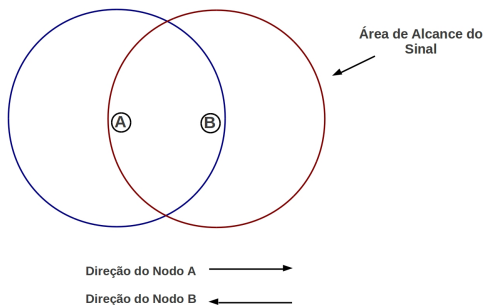


Fig. 5.6: Nodos se Movimentando em Sentidos Opostos Entretanto em Contato.

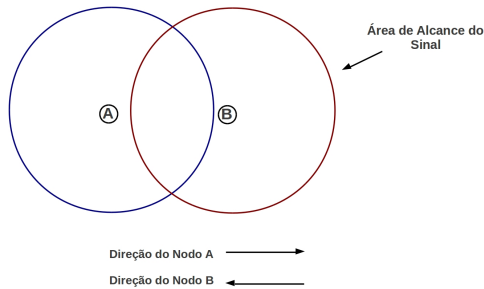


Fig. 5.7: Nodos se Movimentando em Sentidos Opostos Perdendo o Contato.

Pode-se pensar em outros padrões de mobilidade, na qual os nodos ao entrarem em contato, podem mudar de sentido a qualquer momento, não seguindo exatamente o exemplo dado. Entretanto, o momento da mudança de sentido (de aproximação para afastamento) será menor que o tendência de continuar no sentido.

Com base no exemplo citado, notou-se que há uma deficiência conceitual para representar a variação da qualidade de enlace devido ao movimento. No modelo BD, cada estado, que delimita uma faixa de valores do SNR, representa tanto o aumento do SNR devido a aproximação dos nodos, quanto a redução do SNR devido ao afastamento dos nodos. Sendo assim, as probabilidades das transições entre os estados, vista na figura 5.5, podem não representar a tendência da movimentação de aproximação e afastamento. Com base na premissa de Markov, que todas as transições de um estado devem somar 100%, pode-se observar a limitação do modelo com o seguinte exemplo:

- Caso os nodos estejam se aproximando, como mostrado na figura 5.6, as transições que representam a saída do estado (vide figura 5.5), para alcançar o estado N, deveriam ter uma maior

probabilidade que as transições no sentido oposto, pois a tendência do movimento, como visto no primeiro exemplo, é de continuar no mesmo sentido.

- Entretanto, caso os nodos estejam se afastando, como mostrado na figura 5.7, as transições que representam a saída do estado (vide figura 5.5), para alcançar o estado 1, deveriam ter uma maior probabilidade que as transições no sentido oposto, pois a tendência do movimento, como visto no primeiro exemplo, é de continuar no mesmo sentido.
- Sendo assim, seriam necessários dois autômatos BD (figura 5.5), para se representar os dois sentidos do movimento.

Apesar da limitação observada no modelo BD, ele foi adotado como base para o desenvolvimento da presente proposta de tese a fim de caracterizar a variação da qualidade de enlace devido à mobilidade dos nodos. A nova proposta estendeu o modelo BD para melhor representar a questão de fortalecimento e enfraquecimento do sinal devido ao movimento de aproximação e afastamento dos nodos. O modelo BD também foi implementado e simulado para que pudesse ser comparado com a nova proposta.

5.9 Considerações Finais

A mobilidade dos nodos traz novos desafios para a comunicação, afetando protocolos e aplicações. Como a conectividade é o elemento básico da comunicação, faz-se necessário criar mecanismos que possam mitigar a influência da mobilidade e os seus efeitos nas redes,

para melhorar a qualidade de serviço oferecida para as aplicações. Por isso, várias propostas têm sido apresentadas ao longo dos anos.

A questão da conectividade pode ser vista, como mencionado anteriormente, de forma global ou local. As propostas que enxergam a conectividade do nodo relacionada à toda rede, tem uma visão global. Por meio de conhecimento da topologia da rede se verifica o quanto o nodo está conectado à rede, não somente a um vizinho. Este tipo de abordagem tem como objetivo a manutenção da conectividade global da rede, podendo atuar para evitar que qualquer nodo ou conjunto de nodos fique particionado.

Já a visão da conectividade local pode ser vista como um bloco construtor da visão global. A conectividade vista nodo a nodo é o componente primário para uma perspectiva topológica. O comportamento de cada enlace é analisado devido a interferência e/ou mobilidade. As informações coletadas do comportamento do enlace podem ser bastante úteis para que protocolos e aplicações se adequem à conduta da rede, de maneira a melhorar seus desempenhos.

Por esses motivos, o trabalho da presente tese se concentrou em apresentar soluções que pudessem avançar na área de estimação de conectividade local visando trazer benefícios para protocolos e aplicações. No próximo capítulo são apresentadas duas soluções para estimar a conectividade local. A primeira solução é uma extensão do trabalho [Guha e Sarkar 2008] para melhor lidar com o comportamento de mobilidade. A segunda é uma extensão da primeira solução, a fim de lidar com ambientes onde o histórico de registro de dados pode mudar, tornando o modelo baseado em histórico desatualizado.

Capítulo 6

Predição de Conectividade Local

6.1 Modelo do Sistema

O modelo de sistema adotado consiste em nodos em movimento realizando comunicação ad hoc em uma rede com densidade esparsa. A comunicação é periódica e feita por *broadcast* em um salto. Os nodos são capazes de realizar a medição da qualidade de enlace por meio do SNR (*Signal-to-Noise Ratio*) ou RSSI (*Received Signal Strength Indicator*). O modelo de propagação do sinal é o *free-space path loss*. Além disso, é considerado que o ambiente esteja livre de obstáculos e que a qualidade de enlace tenha uma variação gradual e suave ao longo do tempo. Não é considerada uma perda completa do sinal.

6.2 Introdução

Neste capítulo são apresentadas as duas propostas para predição de conectividade local desenvolvidas nesta tese. As duas se baseiam na análise do comportamento da qualidade de enlace devido à mobilidade. A concepção do modelo de predição de conectividade surgiu a partir da observação de experimentos realizados em laboratório. Os experimentos foram feitos com nodos robôs em movimento realizando comunicação entre si e com nodos sensores espalhados pelo ambiente. A partir da medição do SNR dos nodos robôs, verificou-se que havia um padrão que era diretamente relacionado com a movimentação dos nodos. Este fenômeno pode ser observado na figura 6.1. Esta figura representa a medição de um nodo robô se movimentando ao redor da sala de laboratório enviando dados periódicos à um nodo sensor localizado em um canto da sala.

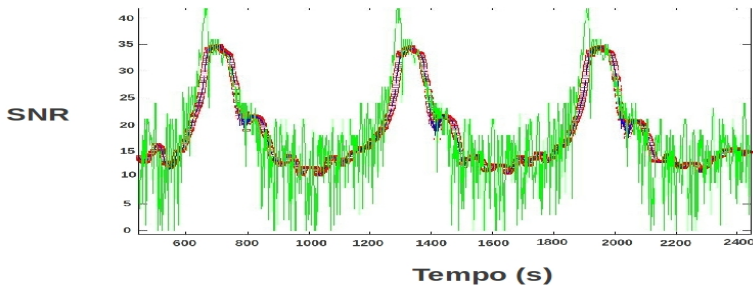


Fig. 6.1: Variação do SNR sobre o tempo devido à mobilidade do nodo e interferência do ambiente.

A variação do SNR mostrada na figura 6.1 demonstra que quando o nodo robô está longe do nodo sensor, a qualidade do enlace atinge o menor nível. A medida que o robô se aproxima do nodo sensor, a qualidade do enlace aumenta gradativamente até atingir o ponto máximo

devido a aproximação entre os nodos. O mesmo fenômeno acontece quando o nodo robô se afasta, a qualidade do enlace reduz gradativamente até atingir o menor nível observado. O aumento e a redução da qualidade do enlace são repetidos ao longo do tempo, mostrando que há uma relação entre o padrão de movimento e a variação da qualidade do enlace.

Além do comportamento da variação da qualidade do enlace em relação ao movimento, também foi observado a variação da qualidade do enlace devida a interferência. O experimento foi realizado com presença de obstáculos e interferência de outro nodo robô utilizando o mesmo canal. Os resultados mostraram uma grande perturbação do sinal, como é mostrado na Figura 6.1. A curva mais clara mostrada na Figura 6.1 representa o SNR puro, enquanto a curva mais escura representa o SNR filtrado.

Devido ao fenômeno de perturbação do sinal, notou-se a necessidade de filtrar o ruído. Dessa forma, poderia se ter uma variação suave que apenas representasse a relação observada da variação da qualidade do enlace e o padrão de movimento. Para se obter o sinal mais suave livre de ruído, o filtro de Kalman com janela desenvolvido no trabalho [Farkas et al. 2008], descrito no capítulo 5, foi utilizado.

6.3 *Oriented Birth-Death*

Seguindo o trabalho desenvolvido por [Guha e Sarkar 2008], descrito no capítulo 5, o SNR foi utilizado para verificar a conectividade entre dois nodos vizinhos de um salto. O trabalho em [Guha e Sarkar 2008] foi implementado no experimento mostrado na figura 6.1

para verificar o comportamento do modelo em relação a variação da qualidade de enlace e o padrão de mobilidade.

O modelo Oriented Birth-Death (O-BD) foi aqui proposto para prever o estado da qualidade do enlace (conectividade) para certo tempo à frente. Um ponto importante da solução O-BD é o fato de não precisar de sistema de localização para computar a estimativa de conectividade. Ao invés disso, o O-BD utiliza a qualidade do enlace como SNR ou RSSI.

Como em seu predecessor, o modelo BD, um modelo de cadeia de Markov é usado para prever a qualidade do enlace no futuro. Modelos de Markov têm a vantagem de não precisar manter o histórico em memória, o que poderia ser limitante para dispositivos com restrições computacionais para calcular a estimativa. Na verdade, o histórico está embutido nas probabilidades de transição entre os estados da cadeia de Markov. Tais probabilidades são armazenadas em uma matriz de tamanho fixo, chamada matriz de transição.

A principal diferença entre os modelos O-BD e BD é que o primeiro leva em consideração a orientação/direção do movimento. Isto não significa que o nodo não pode mudar de direção, mas que a mudança de direção tem uma baixa probabilidade de ocorrer. Para representar esta questão, todos os estados intermediários do BD (vide 5.5) são duplicados, como é mostrado na Figura 6.2. Os estados no topo representam os nodos se movimentando em uma direção e os estado em baixo representam os nodos na direção contrária. Os dois estados nos extremos não são duplicados, porque somente faz sentido para eles irem em uma única direção.

Cada estado representa uma faixa de valor da qualidade de en-

lace. O estado 1 representa a pior qualidade de enlace e o estado N a melhor qualidade de enlace. Para efeitos de simplificação do modelo, o estado 1 é considerado como estado de desconexão, pois a qualidade de enlace é muito ruim e instável. O uso do enlace no estado 1 torna o esforço do uso muito custoso, sendo muitas vezes necessárias retransmissões, o que aumenta a interferência na vizinhança e desgasta recursos importantes como energia.

O modelo pode ter qualquer número de estados. Entretanto, como discutido em [Guha e Sarkar 2008], é necessário ter pelo menos três estados para representar a variação estatística da qualidade do enlace. Por outro lado, se o modelo tem muitos estados, cada estado representaria uma pequena variação do valor qualidade do enlace, o que poderia trazer dois problemas: i) aumento no efeito *ping-pong* e ii) aumento do tamanho da matriz em memória. O efeito *ping-pong* refere-se ao fato do valor da variável em questão pular de um estado para o outro e vice-versa pelo motivo de perturbação e não pela variabilidade devido à mobilidade. Então, quanto menor a faixa de valor que representa o estado, mais suscetível o modelo fica a esse efeito. O aumento da matriz, como anteriormente comentado, pode ser um problema para nodos sensores com capacidade limitada de memória e processamento.

Encontrar o número de estados apropriado para o autômato não é simples. O autômato pode ser visto como o resultado de uma heurística que representa a variação da qualidade de enlace durante o movimento. É importante ressaltar que o presente modelo, como em [Guha e Sarkar 2008], assume que a qualidade do enlace tenha uma variação suave no tempo. A variação suave pode ser proporcionada por um filtro, como descrito anteriormente, mas não é considerada a perda completa de

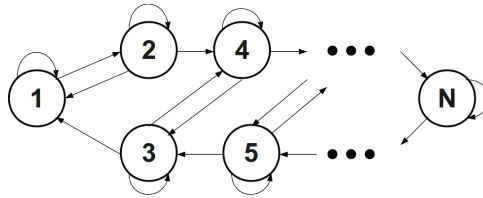


Fig. 6.2: Autômato representando os estados e transições

sinal.

O modelo O-DB é formulado como uma cadeia de Markov discreta, onde o tempo é discretizado para observar as variações da qualidade do enlace. A próxima seção descreve os detalhes de como o modelo foi construído.

6.3.1 Autômato de Estados Finitos do modelo O-BD

O conhecimento provido pelo modelo permite estimar o próximo estado da qualidade de enlace. Portanto, foi utilizada a teoria de cadeia de Markov discreta, na qual cada transição ocorre em um tempo discreto t . Assumiram-se intervalos de 1 segundo para marcar a probabilidade de transição entre os estados. A cadeia de Markov discreta pode ser descrita como a tripla $\langle \mathbf{S}, \mathbf{T}, \pi \rangle$. \mathbf{S} representa o conjunto de estados, como mostrado na Figura 6.2. \mathbf{T} é a matriz de transição, que representa a probabilidade de ir do estado i para o estado j . Finalmente, π é a distribuição de probabilidade inicial do conjunto de estados \mathbf{S} .

A matriz de transição \mathbf{T} e a distribuição de probabilidade inicial π estão relacionadas a cada cenário, neste caso ao padrão de mobilidade. Estes termos podem ser obtidos pelo uso de dados coletados em

simulação ou experimentação. Os detalhes de como os dados são obtidos serão apresentados no capítulo 7 de avaliação dos modelos. Depois da fase de coleta dos dados, no caso o SNR, a matriz \mathbf{T} e a distribuição de probabilidade inicial π são construídas pelo processo de contagem. Dessa forma, retira-se da simulação a distribuição de probabilidade que representa as variações do SNR para o cenário específico, ou seja, para o padrão de mobilidade simulado. O cálculo para obtenção da matriz \mathbf{T} pode ser definido como mostra a equação 6.1.

$$T_{ij} = \frac{A(s_i, s_j)}{\sum A(s_i, s_j)} \quad (6.1)$$

$A(s_i, s_j)$ representa o número de vezes que o valor da qualidade de enlace saiu do estado s_i para o estado s_j . A matriz \mathbf{T} é normalizada pela variação da qualidade de enlace de todas as transições ocorridas no processo. Além disso, π é definido pela equação 6.2.

$$\pi = \frac{V(e_i)}{\sum V(e_k)} \quad (6.2)$$

$V(e_i)$ é o número de vezes que o estado e_i foi visitado. O termo π é normalizado pela soma do número de visitas de todos os estados. Com os termos T e π obtidos, é possível calcular a escolha do próximo estado pela equação 6.3 de vetor de probabilidade de estado. Dessa forma, se obtém a distribuição futura do vetor de probabilidade de estado.

$$\pi^n = \pi * T^n \quad (6.3)$$

onde T^n é a matriz de transição elevada à potência n , onde n representa n passos a frente. Depois de calculada a equação 6.3, é possível adquirir o novo vetor de probabilidade π^n . Para, finalmente, escolher o próximo estado, opta-se pelo estado s_k^n com maior probabilidade no vetor de estado π^n , como mostra a equação 6.4.

$$s^n = \text{Max}[s_1^n, s_2^n, \dots, s_k^n] \quad (6.4)$$

A construção do modelo O-BD baseia-se na obtenção de histórico de dados, que neste caso é a qualidade de enlace representada pelo SNR. Notou-se, via experimentos e simulações, que a mobilidade do nodo pode ser capturada pela variação da qualidade do enlace, como é mostrado na Figura 6.1. Sendo assim, o formato da variação da qualidade do enlace está diretamente relacionado ao padrão de mobilidade do nodo. Isso significa que o modelo alimentado por certo padrão de mobilidade deve ser realimentado caso o padrão de mobilidade mude. Dessa forma, o modelo poderá refletir adequadamente o padrão de mobilidade coletado.

Como comentado no capítulo 5, essa necessidade de realimentação do modelo, no caso de mudança de padrão de mobilidade, pode causar a queda da qualidade da resposta do modelo, caso este fique desatualizado. Em vista desta limitação, foi desenvolvida uma nova abordagem baseada no aprendizado *on-the-fly* para preencher o modelo de acordo com o padrão de mobilidade corrente.

A seguir será apresentada a segunda proposta de estimação de conectividade local por meio da qualidade do enlace.

6.4 Genetic Machine Learning Approach for Link Quality Prediction

A proposta *Genetic Machine Learning Approach for Link Quality Prediction* (GMLA) [Araújo et al. 2012] é baseada no método de sistema classificador [Goldberg 1989]. Sistema classificador é uma técnica de aprendizado de máquina baseada em algoritmos genéticos (AG), que é capaz de aprender sistematicamente regras simples. A técnica de sistemas classificadores foi utilizada na segunda proposta para descobrir, em tempo de execução, os parâmetros do modelo de Markov que representa a variação da qualidade de enlace. Combinar a técnica de sistemas classificadores com modelo de Markov também foi utilizado em [Jha, Tan e Maxion 2001] com a finalidade de realizar a detecção de intrusão em redes de computadores.

6.4.1 Sistemas Classificadores

Os sistemas classificadores (SC) são sistemas de aprendizado de máquina baseados em algoritmo genético que, por sua vez, é um mecanismo fundamentado em regras e que pode aprender por meio de recompensa. Como os classificadores são baseados em algoritmos genéticos, eles evoluem a cada fim de iteração de aprendizagem. As regras para evolução, como cruzamento e mutação, são herdadas dos algoritmos genéticos. A diferença entre sistema classificador e algoritmo genético é que o sistema classificador realiza N consultas para demarcar o fim de interação de aprendizagem e assim realizar o processo de evolução.

A Figura 6.3 apresenta uma visão geral da abordagem proposta por [Goldberg 1989] sobre sistemas classificadores. A mensagem recebida do ambiente pode ativar um ou mais classificadores no conjunto de classificadores. Todos os classificadores são avaliados e suas respostas são comparadas. Após um período de avaliação, os classificadores são recompensados de acordo com o seu desempenho.

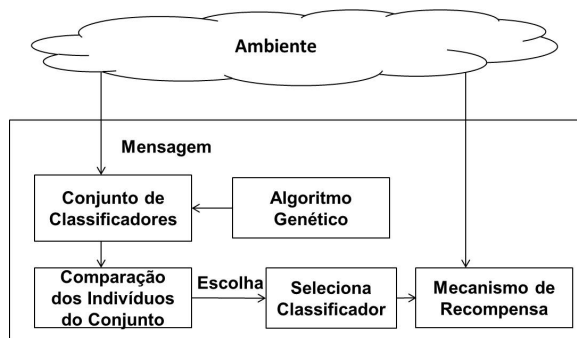


Fig. 6.3: Esquema do Sistema Classificador.

A interação do sistema classificador para escolher o melhor classificador para resolver o problema, passa pelo processo de evolução. Ao final de N interações entre os sistemas classificadores e o ambiente, os classificadores são evoluídos. O processo de evolução é composto pelo cruzamento e a mutação. O processo de cruzamento tem como objetivo trocar as experiências adquiridas entre os classificadores durante o seu processo de interação com o ambiente. No processo de cruzamento são escolhidos os dois melhores indivíduos, com melhor desempenho, e as suas experiências são cruzadas, gerando dois novos indivíduos. Estes dois novos indivíduos substituem os indivíduos menos adaptados, ou seja, os indivíduos que têm os menores desempenhos.

O objetivo dos sistemas classificadores é encontrar o melhor classificador para o problema em questão. Pode-se resumir o processo dos sistemas classificadores da seguinte forma:

1. Uma população de tamanho P é gerada para encontrar a melhor solução (melhor classificador) para um dado problema.
2. Os classificadores são consultados N vezes e recompensados mediante a melhora de desempenho do sistema.
3. Após N consultas, os classificadores são evoluídos:
 - (a) O processo de cruzamento é realizado e dois novos indivíduos são gerados e substituem os dois indivíduos menos adaptados.
 - (b) O processo de mutação é realizado e parte da composição do classificador é modificada por um processo aleatório.
4. Caso algum classificador represente a solução para o problema em questão, o processo é terminado, senão o processo é reiniciado no item 2.

Algoritmos genéticos consideram uma população como um problema de otimização. Neste caso, os classificadores são indivíduos representando seus genótipos. Os genótipos podem ter qualquer formato para representar o problema a ser modelado. A população de classificadores evolui como nos algoritmos genéticos, depois de algumas consultas pré-determinadas.

A cada ciclo de avaliação, um novo conjunto de indivíduos (classificadores) é gerado. O novo conjunto de indivíduos é baseado em

fragmentos dos indivíduos mais adaptados ao ambiente, ou seja, os indivíduos que estão mais próximos de encontrar a solução para o problema.

A principal característica dos algoritmos genéticos é a robustez [Goldberg 1989]. Um sistema robusto requer um menor número de intervenções e redefinições. Além disso, sistemas robustos alcançam um alto grau de adaptação e mantêm a qualidade por mais tempo. A principal diferença da abordagem clássica do algoritmo genético e dos sistemas classificadores é que os sistemas classificadores somente evoluem depois de certo número de consultas ao conjunto de indivíduos. Então os SCs executam o processo de evolução *on-the-fly*.

6.4.2 Funcionamento do GMLA

A proposta *Genetic Machine Learning Approach* (GMLA) é uma extensão do modelo O-BD [Araújo, Kaiser e Becker 2012], pois também utiliza cadeias de Markov para realizar a estimativa futura da qualidade de enlace.

Essencialmente, o sistema classificador foi utilizado para encontrar a melhor matriz de transição T que reflita a variação da qualidade do enlace de acordo com o padrão de mobilidade corrente. Já o vetor de probabilidades iniciais π foi assumido que seria equiprovável, ou seja, a probabilidade de o processo iniciar em qualquer estado é $\frac{1}{M}$, onde M é o número de estados do autômato. A hipótese sobre a qualidade do enlace entre os nodos poder iniciar em qualquer estado, justifica-se pelo fato de os nodos poderem iniciar a comunicação a qualquer distância entre si, o que reflete no estado da qualidade de enlace.

Um nodo pode iniciar a comunicação a qualquer distância na área do sinal. O tempo necessário para treinar o modelo e encontrar o melhor T varia de acordo com a quantidade de amostra da qualidade do enlace. O tempo depende de cada cenário de aplicação. Caso as amostras de qualidade de enlace cheguem com alta frequência, o modelo pode ser construído mais rapidamente. E o contrário também é verdadeiro. Entretanto, não é sempre o caso, no qual a fase de aprendizado é requerida. Por exemplo, os nodos podem receber a matriz T de outros nodos que já aprenderam.

Cada cromossomo no GMLA representa uma matriz de transição e cada gene representa uma linha da matriz, que por sua vez, representa as transições de um estado do autômato. A Figura 6.4 representa a ideia de um cromossomo. Como cada gene é uma linha, ele contém valores das probabilidades de transição que vai de um estado i para o outro estado j . Além disso, cada gene precisa garantir uma regra básica do modelo de Markov, que diz que a soma de todas as transições que saem de um estado precisa ser 100%. Para garantir essa propriedade, cada gene aplica o algoritmo 7.

Input: StateTransitions St , numbertransitions N

```

foreach  $St$   $i$  do
    if  $i == N$  then
         $P_i = 100 - (\sum_{j=0}^{N-1} P_j)$ ;
    end
    else
         $P_i = random(0, 100 - \sum_{j=0}^{j-1})$ ;
    end
end

```

Algorithm 7: Seleção Randômica das Probabilidades de Transição.

Exemplificando a propriedade do algoritmo 7 para um gene com três transições, como na Figura 6.4: i) a primeira transição seleciona randomicamente um número P_1 entre 0 e 100, ii) a segunda transição seleciona outro número randomicamente entre 0 e $(100 - P_1)$, iii) a terceira transição obtém o resto, $P_3 = 100 - (P_1 + P_2)$. No final a soma de $P_1 + P_2 + P_3$ é 100%. Desse modo, cada cromossomo é criado de maneira a representar o modelo do autômato 6.2.

Gene ₁	15	35	50
Gene ₂	10	80	10
Gene ₃	50	5	45

Fig. 6.4: O Cromossomo como uma Matriz e o Gene como uma Linha.

A seguir será descrito o funcionamento do GMLA em todas as etapas, desde a amostragem da qualidade de enlace à evolução da população e escolha do cromossomo mais adaptado.

6.4.2.1 Etapas do GMLA

O funcionamento do GMLA se inicia pela geração da população. Cada indivíduo é gerado randomicamente, sendo que os genes são gerados pelo algoritmo 7. Dessa forma, fica garantido que as linhas da matriz gerada somem 100%.

Após a criação da população, os indivíduos evoluem de acordo com as fases mostradas na Figura 6.5. Primeiramente, a qualidade do enlace é amostrada. Depois ela é filtrada para retirar o ruído provocado pela interferência, assim, o GMLA utiliza a amostra suavizada da qualidade do enlace. Em seguida, a qualidade do enlace é mapeada de

acordo com os estados do autômato (vide Figura 6.2). No próximo estágio, os indivíduos estimam o próximo estado da qualidade de enlace para N transições à frente.

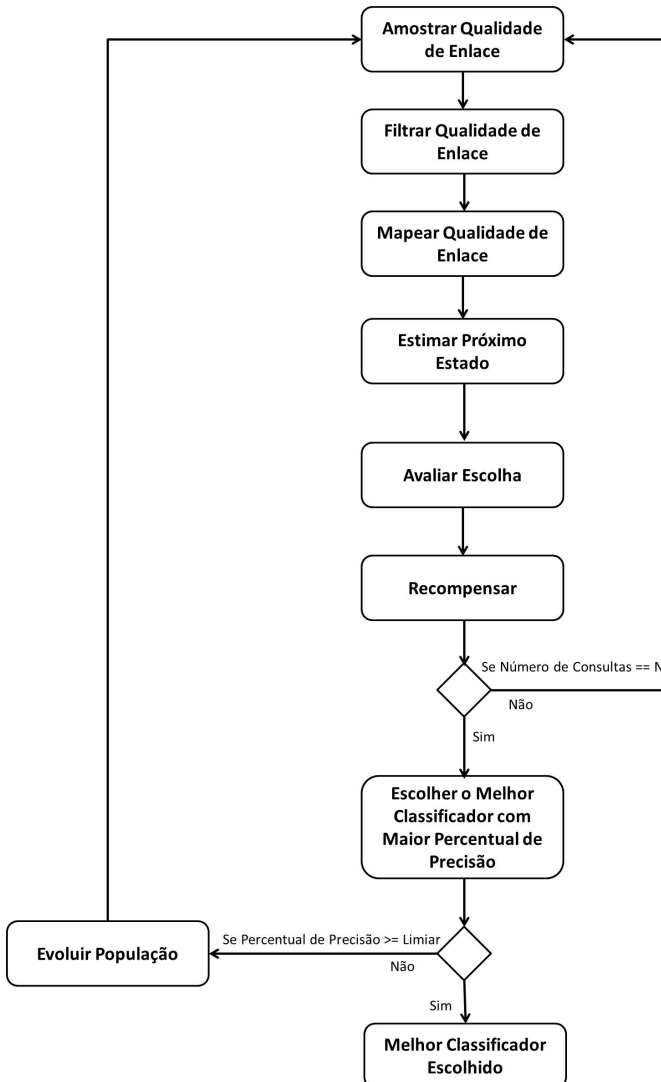


Fig. 6.5: GMLA - Genetic Machine Learning Approach.

Para estimar o próximo estado, os indivíduos computam as equações 6.3 e 6.4. Depois de passados N passos (tempo), todos os indivíduos são avaliados. A qualidade do enlace é amostrada, filtrada e mapeada novamente, e depois comparada com o estado sugerido na etapa de estimativa. Todos os indivíduos que estimaram corretamente são recompensados. Para recompensar cada indivíduo que estima corretamente o próximo estado, é dado um ponto para o indivíduo, e os indivíduos que erram não ganham o ponto. Esse processo continua até o número de consultas acabarem.

Após certo número de consultas, a próxima etapa é escolher o melhor indivíduo. A escolha do melhor indivíduo é feita aplicando a função de aptidão expressa pela equação 6.5. O percentual de precisão, AP , é o número de estimativas corretas, P_t , dividido pelo número de consultas Q . O indivíduo com maior AP é escolhido. Se o indivíduo com o maior AP for maior ou igual a um determinado limiar, então o processo do GMLA termina. Entretanto, caso o indivíduo com maior AP esteja abaixo do limiar, toda a população evolui.

$$AP = \frac{\sum P_t}{Q} \quad (6.5)$$

Seguindo o processo de evolução descrito em [Goldberg 1989] foram utilizados o cruzamento e a mutação. O cruzamento é aplicado aos dois melhores indivíduos, com maiores AP s. No processo de cruzamento, os genes (estado) são cruzados. Por exemplo, em um modelo com 3 estados, o estado 1 é escolhido do indivíduo A e os estados 2 e 3 são escolhidos do indivíduo B, gerando dois novos indivíduos. Uma ilustração do processo de cruzamento pode ser vista na Figura 6.6.

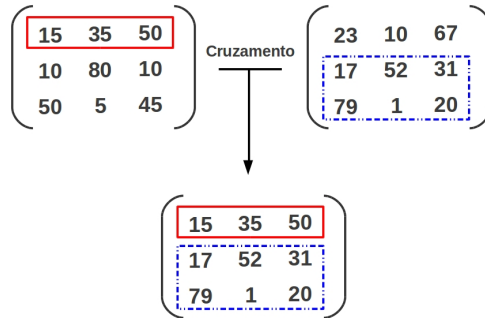


Fig. 6.6: O Processo de Cruzamento do GMLA Gera Dois Novos Indivíduos.

Após o cruzamento, todos os indivíduos que estiverem com o valor de AP abaixo do limiar sofrem mutação. A mutação é aplicada a um gene randomicamente escolhido. Neste processo, novos valores para o gene são escolhidos, utilizando o algoritmo 7, assim novas probabilidades para as transições são geradas. O processo de mutação é ilustrado na Figura 6.7, onde novas probabilidades são escolhidas para as transições do terceiro estado. Um gene (linha da matriz) é randomicamente escolhido e aplicado o algoritmo 7. Assim novas probabilidades são geradas.

Depois de realizados os processos de cruzamento e mutação, os dois piores indivíduos, com piores valores de AP, são substituídos pelos dois novos indivíduos gerados pelo cruzamento. Estes passos são conhecidos como fase de aprendizado. O processo é repetido até que haja um indivíduo com valor de AP maior ou igual ao limiar estabelecido.

No próximo capítulo são apresentadas as avaliações realizadas com as duas propostas, *Oriented Birth-Death* e o *Genetic Machine Learning Approach for Link Quality Prediction*. Serão analisadas as capacidades de predição de estimativa para N passos à frente. Além

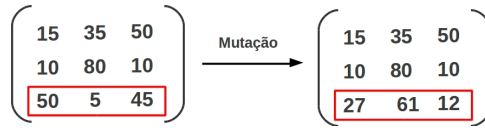


Fig. 6.7: O processo de Mutaçao do GMLA Randomicamente Escolhe Um Gene e Aplica o Algoritmo 7.

disso, outros trabalhos correlatos da literatura, que foram descritos no capitulo 5 tambem foram implementados e suas precisoes foram comparadas com as propostas descritas neste presente capitulo. Por fim, um estudo de caso envolvendo a aplicacao do GMLA a um protocolo de roteamento classico da literatura sera apresentado.

Capítulo 7

Avaliação dos Mecanismos Propostos para Predição de Conectividade Local

Neste capítulo são apresentadas as avaliações dos mecanismos de predição de conectividade local descritos no capítulo 6. Para melhor avaliar o desempenho dos mecanismos, alguns trabalhos correlatos da literatura foram implementados. A métrica utilizada para avaliar as propostas foi a precisão da estimativa da qualidade do enlace. A qualidade do enlace foi representada pelo SNR (*Signal-to-Noise Ratio*). Além disso, foi realizada uma avaliação entre as propostas O-BD e GMLA sob o aspecto de mudança no ambiente. Neste caso, o padrão de mobilidade foi alterado durante a simulação para verificar como as duas propostas reagem à mudança de ambiente.

Como o modelo de cadeia de Markov foi utilizado pelos métodos propostos e pela maioria dos trabalhos correlatos da literatura implementados, foi avaliado não exatamente o valor do SNR, mas o estado estimado do SNR. O único trabalho correlato da literatura implementado que não utiliza cadeia de Markov foi o [Su, Lee e Gerla 2000]. Nesta proposta foi utilizado um modelo determinista para calcular a durabilidade do enlace. Sendo assim, é importante comparar a eficiência entre o modelo determinista com modelos probabilísticos, afim de estimar a durabilidade da conectividade.

Após a apresentação das avaliações individuais dos modelos propostos, será apresentado um estudo de caso com a implementação do GMLA em um protocolo de roteamento MANET clássico. Neste estudo de caso foi visualizada a possibilidade de melhoria no desempenho dos protocolos de roteamento MANETs. Os detalhes sobre esse estudo são descritos na seção 7.2 de Estudo de Caso.

7.1 Análise Individual dos Mecanismos O-DB e GMLA

Foram realizados dois conjuntos de simulações. No primeiro conjunto foi avaliada a habilidade de o modelo em estimar corretamente o próximo estado da qualidade do enlace. No segundo conjunto os modelos O-BD e GMLA foram avaliados por uma simulação que muda o padrão de mobilidade após certo tempo.

Na avaliação dos dois conjuntos de experimentos, não é estimado o próximo estado da qualidade de enlace somente para o próximo passo no futuro, mas para n passos à frente. Retomando a hipótese dos modelos propostos, um passo à frente representa o tempo de um se-

gundo, pois foi assumido que poderia ocorrer uma transição de estado por segundo.

As avaliações foram feitas por meio de simulações. A escolha de realizar simulações vem do fato de ser possível avaliar os modelos com diferentes padrões de mobilidade. A avaliação com diferentes padrões de mobilidade é importante, pois os modelos propostos para predição de conectividade local em ambiente de mobilidade foram baseados na hipótese de que a variação da qualidade do enlace poderia representar o padrão de mobilidade dos nodos móveis.

O simulador escolhido foi o OMNET++ [Varga et al. 2001] juntamente com o *framework* INETMANET [Quintana e Cabrera 2012]. Todos os nodos utilizaram como protocolo MAC padrão o IEEE 802.15.4 modo *non-beacon*. Todos os modelos foram implementados sobre a camada MAC, entretanto eles podem ser utilizados pelas camadas superiores.

7.1.1 Modelos de Mobilidade

Como discutido em [Camp, Boleng e Davies 2002], existem vários modelos de mobilidades sintéticos, que tentam representar um comportamento realista. Foram escolhidos três modelos que são largamente utilizados na literatura.

O primeiro modelo adotado foi o *Gauss-Markov* (GM) [Liang e Haas 2003], criado para superar o alto grau de comportamento randômico do conhecido modelo *Random Way Point Mobility*. O modelo de mobilidade *Random Way Point Mobility* tem o problema de ge-

rar movimentos irrealis como súbitas paradas e reinícios. O modelo *Gauss-Markov* adiciona um parâmetro de aleatoriedade α . Este parâmetro rege a aleatoriedade de variáveis como velocidade e direção que seguem uma distribuição Gaussiana. O valor de α varia entre 0 (movimento Browniano) e 1 (movimento linear).

O modelo *Reference Point Group Mobility Model* (RPGM) [Hong et al. 1999] foi outro modelo adotado. O RPGM foi projetado para representar o comportamento de nodos se movendo em grupos. O líder do RPGM é aleatoriamente escolhido e os outros nodos seguem o líder. Muitos cenários podem representar o movimento do modelo RPGM, como equipes de resgate em operações de salvamento ou tropas militares em campos de batalha, ou mesmo robôs móveis fazendo algum trabalho cooperativo, como a vigilância de um perímetro de segurança.

O terceiro modelo adotado foi o *Manhattan Grid Mobility Model* (MGM) [Bai, Sadagopan e Helmy 2003], que foi essencialmente proposto para representar o movimento em áreas urbanas. Os nodos fazem movimento em grade no modelo MGM, pois somente podem se deslocar em direções horizontais e verticais, ou seja, eles podem escolher ir reto ou virar para direita ou esquerda.

Os três modelos descritos foram utilizados para avaliar as propostas apresentadas no capítulo 6 e alguns trabalhos correlatos apresentados no capítulo 5. A maioria das propostas testadas utiliza o conhecimento do histórico de variação da qualidade do enlace devido à mobilidade. Somente a proposta GMLA e o trabalho correlato [Su, Lee e Gerla 2000] não utilizam conhecimento do histórico de variação da qualidade do enlace. O GMLA utiliza um sistema de aprendizado de máquina baseado em algoritmo genético, enquanto que o trabalho

correlato [Su, Lee e Gerla 2000] utiliza um modelo determinista fundamentado no conhecimento da velocidade relativa e área de alcance do sinal.

Para todos os modelos que necessitam de histórico, foram realizadas 20 simulações de 1000 segundos para cada modelo de mobilidade, a fim de criar o histórico para poder processá-lo e alimentar os modelos. Apesar de o histórico ser o mesmo para todos os modelos, as matrizes de transições geradas pelo processamento do histórico são diferentes para cada um deles, pois cada modelo possui um autômato único.

7.1.1.1 Configurações dos Experimentos

O primeiro conjunto de experimentos foi feito com 100 nodos, sendo que 10% eram estacionários e 90% eram móveis. Os nodos estáticos podem ser representados por nodos sensores fixos espalhados em uma área delimitada como uma fábrica. Quanto aos nodos móveis, eles podem ser representados por agentes móveis, que podem ser robôs ou pessoas. Os nodos móveis escolheram velocidades randomicamente entre 1 a 10 m/s. Todos os nodos enviavam mensagens a uma taxa seguindo uma distribuição uniforme entre 1 a 5 segundos.

Nestes experimentos, 10% dos nodos foram aleatoriamente escolhidos para realizar a estimativa de conectividade entre a vizinhança. Para o modelo de mobilidade *Gauss-Markov*, o parâmetro α foi escolhido pelos nodos móveis entre 0.5 e 0.9 para evitar os movimentos Browniano e Linear.

Tabela 7.1: Parâmetros Gerais do OMNET

Parâmetros	Valor
<i>Sensibilidade</i>	-85 dBm
<i>Potência de Transmissão</i>	1.0 mW
<i>Ruído</i>	-110 dBm
<i>Área de Simulação</i>	1000 x 1000 m ²
<i>Tempo de Simulação</i>	20.000 segundos
<i>Modelo de Atenuação de Sinal</i>	<i>Path Loss Reception Model</i>

A métrica utilizada foi o percentual de precisão, descrito na fórmula 7.1. O termo P_p é o percentual de precisão, o $N_{acertos}$ é o número de acertos e o $N_{consultas}$ é o número de consultas. O percentual de precisão P_p foi avaliado considerando n passos à frente. Mais especificamente, para cada modelo foram conduzidos experimentos utilizando 1, 5, 10, 15, 20, 25 e 30 passos à frente. Além disso, para melhor entender a influência do número de estados, o O-BD e o GMLA foram implementados em três versões de 6, 8 e 10 estados. O modelo original do BD foi implementado com 8 estados, pois foi sugerido por [Guha e Sarkar 2008] que seria o melhor número de estados para representar a variação da qualidade de enlace.

$$P_p = \frac{N_{acertos}}{N_{consultas}} \quad (7.1)$$

Os parâmetros gerais referentes à configuração da camada física e modelo de propagação de sinal podem ser vistos na Tabela 7.1.

Os parâmetros para o GMLA são apresentados na Tabela 7.2. O número de indivíduos foi escolhido para que o GMLA pudesse ser

Tabela 7.2: Parâmetros do GMLA

Parâmetros	Valor
<i>Número de Indivíduos</i>	16
<i>Limiar de Mutação</i>	70%
<i>Número de Consultas</i>	100
<i>Tempo de Fase de Aprendizado</i>	600 segundos

implementado em dispositivos com recursos computacionais limitados como nodos sensores. Como cada indivíduos representa uma matriz de transição, o espaço em memória necessário é diretamente proporcional à quantidade de indivíduos. Sendo assim, se for necessário ter uma grande quantidade de indivíduos, a solução GMLA ficaria limitada a ser implantada em dispositivos computacionais com mais recursos que os nodos sensores usuais.

No segundo conjunto de experimentos, foram utilizados apenas 5 nodos para avaliação da mudança no padrão de mobilidade, sendo quatro nodos sorvedouros fixos que implementaram os modelos O-BD(6) e GMLA(6) para realizar a estimativa para 10 passos a frente. O quinto nodo se movia pela área e enviava mensagens. O tamanho da área também foi reduzido para $300m^2$.

Este experimento foi realizado em menor escala, pois o objetivo era capturar o comportamento dos modelos devido à mudança no padrão de mobilidade. A escala simulada foi suficiente para verificar a mudança no padrão de mobilidade e seu efeito sobre os modelos propostos.

Para o segundo conjunto de experimentos, foi utilizada o mo-

delo de mobilidade chamada de *Chain Mobility* [Aschenbruck et al. 2010]. A mobilidade *Chain Mobility* não é um padrão de mobilidade de fato, mas uma configuração que permite unir dois padrões de mobilidades conhecidos. Neste conjunto de experimentos, os padrões de mobilidade *Gauss-Markov* e *Manhattan* foram unidos. Foi realizada uma simulação com a configuração *Chain Mobility* para cada modelo O-BD e GMLA.

Neste caso, a simulação inicia-se por *Gauss-Markov* e depois de 1000 segundos de experimento, o padrão de mobilidade muda para *Manhattan*. Dessa forma, os nodos podem experimentar uma mudança repentina no padrão de mobilidade. As configurações de rádio estão de acordo com a Tabela 7.1. Entretanto, o tempo de simulação descrito na Tabela 7.1 foi aumentado para 2500 segundos. Ademais, a configuração do modelo GausMarkov foi a mesma do primeiro conjunto de experimentos.

A seguir serão apresentados os resultados obtidos nos dois conjuntos de experimentos com os modelos propostos, O-BD e GMLA, bem como alguns trabalhos correlatos.

7.1.1.2 Resultados Obtidos

No primeiro conjunto de experimentos, foram realizadas comparações entre três versões diferentes dos modelos GMLA e O-BD (6,8 e 10 estados), além dos modelos BD, MTCP [Sabitha e Thangavelu 2011] e o modelo apresentado em [Su, Lee e Gerla 2000], que será chamado aqui de modelo GPS, pois as informações de entrada deste modelo são provenientes do sistema de posicionamento GPS. Os

resultados estão ilustrados nas Figuras 7.1, 7.2 e 7.3.

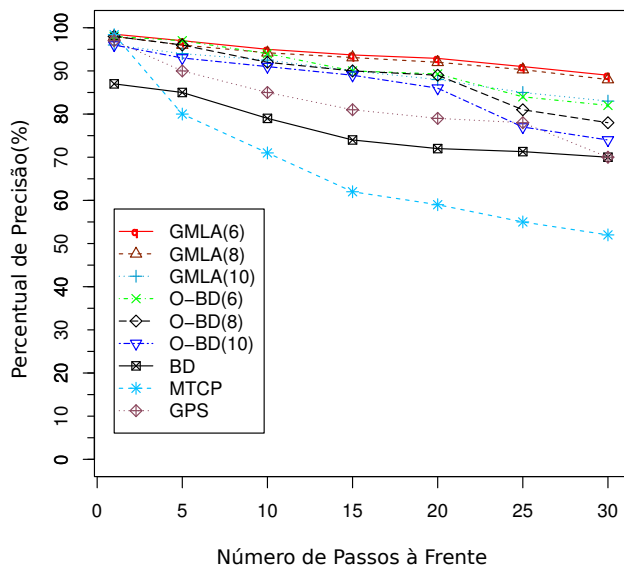


Fig. 7.1: Percentual de Precisão para escolher o próximo estado correto em 1,5,10,15 e 20 segundos à frente para o modelo de mobilidade *Manhattan*.

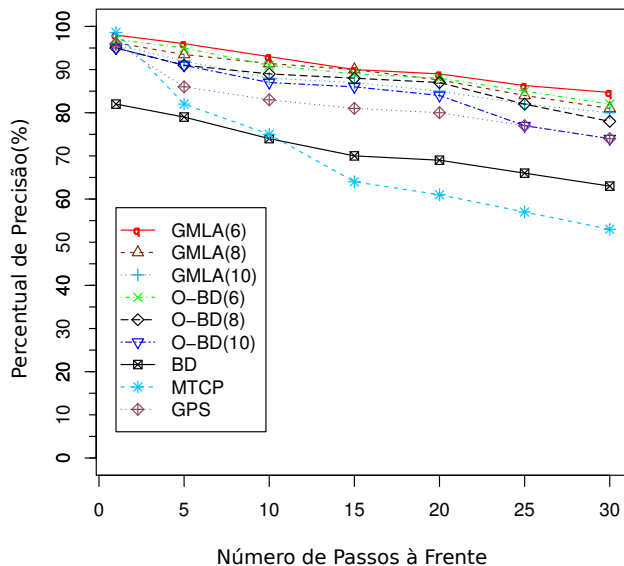


Fig. 7.2: Percentual de Precisão para escolher o próximo estado correto em 1,5,10,15 e 20 segundos à frente para o modelo de mobilidade *Reference Point Group*.

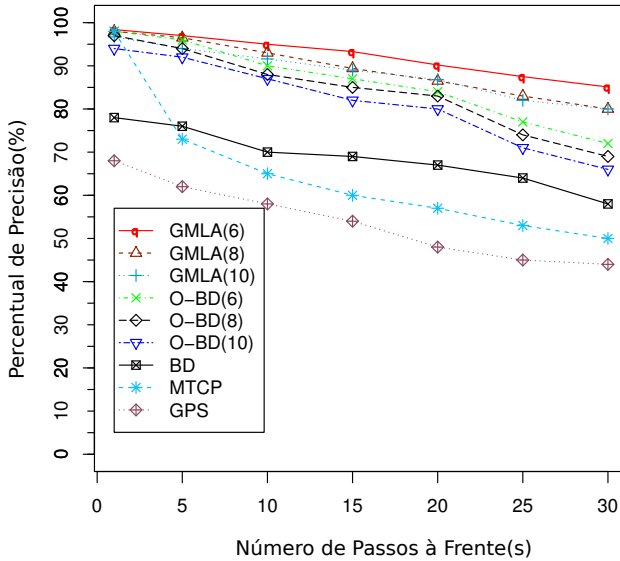


Fig. 7.3: Percentual de Precisão para escolher o próximo estado correto em 1,5,10,15 e 20 segundos à frente para o modelo de mobilidade *Gauss-Markov*.

Os modelos foram comparados pela métrica de precisão de predição, considerando n passos à frente. Faz-se necessário ressaltar que a intenção do experimento não foi somente avaliar os modelos quantitativamente. De fato, a intenção é observar a viabilidade de se usar a qualidade do enlace para estimar a conectividade entre nodos móveis.

Nos três experimentos realizados, os modelos propostos O-BD(n) e GMLA(n) superam os modelos correlatos, BD, MTCP e GPS em relação à métrica percentual de precisão. Primeiramente, nota-se que há diferenças entre as versões entre o O-BD(n) e GMLA(n). Este comportamento confirma o fato de que, quanto maior o número de estados,

mais imprecisa se torna a estimativa, pois aumenta o número de opções de escolha do próximo estado.

Além disso, observa-se, que os resultados entre as propostas O-BD e GMLA são ligeiramente diferentes. O GMLA superou o O-BD em todas as versões. Isto se deve ao fato de que em cada simulação a *seed* do simulador varia, o que varia um pouco o histórico, mas não ao ponto deste ficar completamente desatualizado. Já o GMLA aprende com a simulação corrente (*seed* corrente), sendo assim, não é preciso resimulá-lo.

Os resultados também mostram a habilidade do GMLA (para 6,8 e 10 estados) de estimar corretamente o próximo estado j a partir do estado corrente i . No melhor caso, os percentuais de precisão ficaram em 98.5%, 98.0%, e 96.7% para um passo à frente no modelo de mobilidade *Manhattan*. No pior caso, para 30 passos à frente, os percentuais de precisão foram de 80%, 79%, e 77% para o modelo de mobilidade *Gauss-Markov*.

Para o O-BD, os resultados também mostram a sua habilidade de predição, apesar de terem ficado inferiores aos do GMLA. No melhor caso, os percentuais de precisão ficaram em 98%, 97% e 94% para o modelo de mobilidade *Manhattan*. Para o pior caso, os percentuais de precisão ficaram em 72%, 69% e 66% para o modelo de mobilidade *Gauss-Markov*.

Pode-se notar que, houve um maior distanciamento entre o GMLA e o O-BD para o modelo *Gauss-Markov*. Como comentado anteriormente, isto se deve ao fato de que, para os modelos baseados em histórico, é necessário realizar experimentos prévios com *seeds* diferentes, para construir a matriz de transição. Sendo assim, ao realizar o expe-

rimento, após processar o histórico, escolhe-se novamente uma *seed* aleatoriamente. Essas diferenças entre os históricos, provocadas pelas diferentes *seeds* geraram uma pequena diferença nos resultados. Como o modelo *Gauss-Markov* apresenta maior aleatoriedade, as diferentes *seeds* tiveram uma influência maior, o que gerou um maior distanciamento entre o GMLA e o O-DB.

Os resultados também devem ser analisados sob outras duas perspectivas: (i) a influência no número de estados e (ii) a influência dos modelos de mobilidades. No primeiro aspecto, nota-se que quando o número de estados cresce, o percentual de precisão diminui. Com mais estados, o tamanho do vetor de estados π aumenta e a distribuição de probabilidades fica mais esparsa, dando ao modelo uma precisão mais granular. Sendo assim, com um número menor de estados é possível ter uma visão geral da qualidade de enlace, embora mais acurada. Entretanto, a decisão quanto ao número de estados precisa ser um compromisso no projeto de protocolos e aplicações, pois se o número de estados é muito pequeno, não há uma visão mais precisa da qualidade do enlace, visto que esta estaria em uma faixa muito grande de valores para a definição do tamanho do estado.

Com relação aos modelos de mobilidade, apesar das diferenças dos resultados dos experimentos, notou-se que, para os modelos de mobilidade *Manhattan* e RPGM, os resultados são similares aos dos GMLA e O-BD. Entretanto, para o modelo *Gauss-Markov*, o P_p decresce acentuadamente após 5 passos à frente. A característica de aleatoriedade do modelo *Gauss-Markov* dificulta a criação de um padrão de mobilidade, o que leva aos erros de estimativa.

Quanto aos outros modelos correlatos, o modelo BD teve re-

sultados que variaram entre 87% (para a mobilidade *Manhattan* em 1 passo à frente) e 58% (para o *Gauss-Markov* em 30 passos à frente) sendo estes o melhor e o pior caso, respectivamente. O modelo MTCP teve bons resultados, por volta de 98%, para um passo à frente, em todos os modelos de mobilidade. Contudo, o rendimento cai depois de 1 passo à frente atingindo 50% no pior caso para 30 passos à frente para o modelo de mobilidade *Gauss-Markov*. O modelo MTCP é um autômato *full connected*, o que significa que todos os estados estão conectados entre si. Esta característica do autômato de ser completamente conectado reduz a probabilidade para cada transição, o que leva a mais estados destinos possíveis, sendo assim, torna mais imprecisa a escolha do estado final.

Por último, o modelo GPS obteve resultados interessantes. Os resultados foram bons para o modelo de mobilidade *Manhattan*, entre 97% e 70%, e para o RPGM variando entre 96% e 74%. Entretanto, para o modelo *Gauss-Markov*, o resultado foi o pior de todos os modelos experimentados, variando entre 68% e 44%. Os resultados do GPS apontam uma limitação do uso de métodos determinísticos para estimativa de conectividade em ambientes de mobilidade com presença de aleatoriedade. Observa-se que todos os parâmetros que são necessários para o modelo GPS podem sofrer mudanças instantes após o cálculo, o que leva a crer que o método precisa ser recalculado a cada momento que for detectada uma mudança nos parâmetros.

Para o segundo conjunto de experimentos, o resultado pode ser visto na Figura 7.4. Há três fatores importantes a serem discutidos com relação a estes experimentos. Primeiramente, pode-se observar que o O-BD não necessita de um período de aprendizado, pois o modelo está previamente computado. Contudo, o GMLA precisou de um período

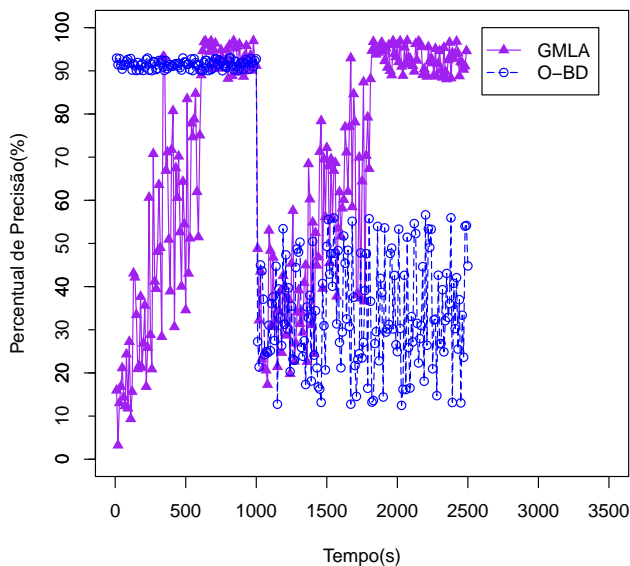


Fig. 7.4: Processo de Reaprendizado com Mudança de Padrão de Mobilidade.

de aprendizado de 600 segundos para alcançar o modelo O-BD.

Além disso, nota-se que após 1000 segundos de simulação, a mudança no padrão de mobilidade prejudicou os dois modelos, reduzindo as suas capacidades de estimar corretamente o estado da qualidade do enlace, além de torná-los instáveis, variando entre 12% e 55% o percentual de acerto. O terceiro fator é a capacidade de readaptação do GMLA frente ao O-BD. Enquanto o modelo O-BD se manteve instável até o final da simulação, o GMLA iniciou o processo de reaprendizagem, depois de detectada a queda na estimativa da qualidade de enlace.

Enfim, os resultados também mostraram que é possível utilizar o conhecimento de qualidade de enlace para estimar a conectividade local com uma precisão adequada. Além disso, é possível aprender um padrão de mobilidade utilizando o comportamento da qualidade de enlace para construir o conhecimento de conectividade local.

7.1.2 Conclusões Parciais

O desenvolvimento das propostas de estimação de conectividade local foram motivadas para melhorar as soluções existentes em duas características. A primeira melhora foi modelar o movimento de um nodo em seus aspectos mais simples, que são o afastamento e aproximação. Independente do modelo de mobilidade, todos os nodos se afastam ou se aproximam. Essas duas características foram identificadas nos primeiros experimentos com robôs móveis. Então, foi desenvolvido a solução O-BD (*Oriented-Birth Death*) para superar a limitação conceitual do modelo *birth-death* em representar o afastamento e aproximação presentes nos padrões de mobilidade.

A segunda característica que foi objeto de estudo, foi o fato de a variação da qualidade do enlace ficar representada pelo padrão de mobilidade corrente. Isso indica que essa relação direta pode prejudicar os protocolos e aplicações, caso o histórico que representa a variação da qualidade de enlace mude. Por isso, foi desenvolvido o mecanismo GMLA para que este pudesse aprender com a variação da qualidade de enlace em tempo de execução. Além disso, as soluções apresentadas dispensaram o uso de sistema de posicionamento, o que amplia a gama de aplicações para ambientes *indoor* e *outdoor*.

Os experimentos mostram que apesar de os modelos BD e MTCP serem projetados para estimar a qualidade do enlace, eles tiveram rendimentos inferiores, pois não representaram adequadamente a variação da qualidade de enlace devido à mobilidade. De fato o MTCP não foi projetado com esse intuito, contudo, por se tratar de um modelo proposto para realizar a predição da qualidade de enlace, foi importante a sua experimentação para melhor entender os seus limites.

Além disso, pode-se observar que a proposta com GPS somente é adequada para padrões de mobilidade determinísticos ou com pouca aleatoriedade. Quanto mais aleatório é o movimento, aumenta a necessidade de o modelo GPS recalcular a estimativa devido às mudanças nos parâmetros.

As soluções propostas podem ser combinadas a protocolos de roteamento existentes, tanto proativos, como OLSR, como reativos, como AODV. Esses protocolos adotam um mecanismo de mensagem para ficarem cientes da conectividade, ou seja, enviam mensagens para manter o contato com a vizinhança. Por considerarmos o GMLA uma evolução do O-BD, o GMLA foi combinado ao protocolo clássico da literatura AODV, a fim de melhorar o desempenho do protocolo de roteamento provendo o conhecimento da estimativa de conectividade entre os nodos vizinhos.

Os detalhes do estudo de caso realizado com o protocolo de roteamento AODV, bem como os resultados e discussões a respeito, serão descritos na próxima seção.

7.2 Aplicações do GMLA em Algoritmo de Roteamento MANET

O conhecimento da conectividade é uma questão crucial para o desempenho de protocolos de roteamento MANETs. A topologia dinâmica dos cenários MANETs gera um desafio para manutenção de rotas para os destinos. Como a vizinhança muda constantemente, os protocolos de roteamento precisam ter mecanismos para rastrear seus vizinhos a fim de manterem as tabelas de rotas atualizadas. Normalmente, o mecanismo de mensagem é utilizado para garantir o conhecimento de conectividade entre a vizinhança. Mensagens de controle são enviadas periodicamente via *broadcast* em um salto para atualizar a tabela de vizinhança dos protocolos. Por isso, este é o mecanismo utilizado para verificar o contato com a vizinhança.

O protocolo *Ad-Hoc On-Demand Distance Vector (AODV)*, descrito no capítulo 5, é um exemplo típico de protocolo de roteamento MANET que utiliza mensagens denominadas *HELLO* para prover a ciência de conectividade entre a vizinhança. Como os nodos na mesma vizinhança de um salto podem estar a diferentes distâncias entre si, eles podem deixar a área de comunicação em tempos diferentes. Se um nodo para de receber mensagens *HELLO*, o protocolo assume que o nodo não pertence mais à vizinhança e o remove da tabela de vizinhança.

Esse método de manutenção do conhecimento da conectividade é eficiente para manter a tabela de roteamento. Entretanto, a desvantagem deste método é que o mesmo causa uma grande sobrecarga na rede. A quantidade de mensagens requerida para manter o conhecimento de conectividade pode degradar a largura de banda e, conseqüentemente, reduzir a taxa de mensagens entregue, bem como au-

mentar o consumo de energia utilizado pelos nodos.

Afim de superar a desvantagem do método adotado para trazer o conhecimento de conectividade, o mecanismo GMLA foi implementado ao AODV para reduzir a quantidade de mensagens *HELLO*. Os detalhes de como o mecanismo de mensagens no AODV trabalha são apresentadas na subseção seguinte.

7.2.1 Visão Geral das Mensagens do Protocolo AODV

Como descrito no capítulo 5, o AODV é um protocolo reativo, o que significa que ele somente cria uma rota sob demanda. Ao estabelecer uma rota, o AODV mantém o conhecimento dos nodos vizinhos que estão nela incluídos por meio de mensagens *HELLO*.

De fato, o AODV utiliza três mensagens adicionais além das mensagens *HELLO*, como descrito a seguir:

- *Route Request* (RREQ): mensagem de requisição de rota é transmitida por toda rede a fim de encontrar o nodo destino.
- *Route Reply* (RREP): mensagem de resposta à mensagem de requisição RREQ enviada pelo nodo destino encontrado.
- *Route Error* (RERR): mensagem enviada ao nodo fonte, quando é detectada uma rota interrompida. Assim o nodo fonte pode iniciar o processo de reparo da rota, feito pelas mensagens RREQ e RREP.

A detecção de rota interrompida é feito por meio de mensagens

HELLO. O protocolo determina a perda de conectividade na rota e inicia o mecanismo de reparo, caso algum nodo vizinho que participa da rota deixe de enviar certa quantidade de mensagens *HELLO*.

O mecanismo de mensagens *HELLO* pode ser reduzido ao mínimo necessário. Essa melhora é obtida pela definição de um prazo limite que representa a durabilidade da rota. Quando o prazo limite expira, não é mais necessário enviar mensagens *HELLO*. Para fins de identificação nos experimentos realizados, essa versão melhorada do AODV foi chamada de U-AODV. A versão U-AODV é simplesmente uma parametrização do próprio AODV para reduzir a quantidade de mensagens *HELLO*.

Entretanto, a quantidade de mensagens *HELLO*, pode ser ainda mais suprimida. O objetivo da integração do GMLA e o AODV é demonstrar que a quantidade de mensagens *HELLO* pode ser mais reduzida. Os detalhes desta integração são descritos na subseção seguinte.

7.2.2 AODV Aprimorado com GMLA

O aprimoramento do AODV utilizando o GMLA consiste em reduzir a quantidade de mensagens *HELLO*. Para isso, o tempo de intervalo para o envio das mensagens *HELLO* foi regulado. Sendo assim, o tempo de duração do enlace foi computado para cada vizinho de um salto. A ideia é estimar quando os nodos estarão no estado 1 do autômato 6.2. O estado 1 representa uma qualidade de enlace muito ruim, o que consideramos como falta de conectividade. Quanto mais perto os nodos estão entre si, mais longe do estado 1 eles estão e o contrário é verdadeiro.

O algoritmo 8 expressa a ideia principal. Quando a probabilidade da qualidade de enlace de estar no estado 1 é maior do que a de estar nos outros estados, este tempo é considerado como o possível intervalo para as mensagens *HELLO*. Isto é feito para todos os vizinhos, então o intervalo para envio de mensagens *HELLO* é retirado do menor valor de todos os vizinhos.

```

Output: TimeInterval
TimeInterval, t = 10;
foreach Neighbor nh do
    for i = 1 to t do
         $\pi^i = \pi * T^i$ ;
         $s_{n,nh}^i = \text{Max}[s_1^i, s_2^i, \dots, s_k^i]$ ;
        if  $s_{n,nh}^i == 1$  then
            if TimeInterval > i then
                | TimeInterval = i;
            end
            break;
        end
    end
end

```

Algorithm 8: Seleciona o Intervalo para Mensagens *HELLO*.

O AODV possui um mecanismo de agendamento para gerar as mensagens *HELLO*. O AODV cria o evento de envio de mensagem *HELLO* e o agenda para envio em um intervalo determinado. A chamada ao GMLA foi realizada exatamente no ponto de determinação do intervalo de envio da mensagem *HELLO*. Sendo assim, neste ponto, o AODV invoca o GMLA, que executa o algoritmo 8 e retorna o intervalo que será utilizado. A cada vez que a mensagem *HELLO* é enviada, uma nova é agendada, e o período de envio é consultado pelo GMLA. O

GMLA utilizou qualquer mensagem, tanto de dados como de controle para verificar o SNR. O componente GMLA foi ligado diretamente a camada MAC para capturar o SNR das mensagens que chegaram para o nodo.

No começo da simulação o GMLA está em fase de treinamento. Neste caso, o AODV funciona de forma padrão até que o GMLA sinalize o AODV informando que está pronto para ser utilizado. O algoritmo 9 expressa a sinalização do GMLA ao AODV.

```
if  $\overline{GMLA.isReady() == True}$  then  
  |  $TimeInterval = GMLA.getTimeInterval()$ ;  
end  
else  
  |  $TimeInterval = AODV_{PARAMETER}$ ;  
end
```

Algorithm 9: Sinalização do GMLA ao AODV.

O AODV melhorado com o GMLA foi analisado por meio de simulação, como é apresentado na próxima subseção.

7.3 Estudo da Simulação

O objetivo do estudo da simulação é avaliar o quanto o GMLA pode melhorar o desempenho de protocolos de roteamento MANETs como o AODV. Isto foi feito pela regulação da quantidade de mensagens de controle (*HELLO*) de acordo com o tempo de conectividade restante. A hipótese é que é possível reduzir a quantidade de mensagens de controle.

Como consequência, é possível reduzir a sobrecarga da rede causada pelas mensagens de controle e também aumentar a taxa de entrega de mensagens. Ademais, a redução da sobrecarga da rede leva a decréscimo no consumo de energia para dispositivos alimentados a bateria.

Além de analisar o AODV padrão e sua combinação com o GMLA, chamada de GMLA-AODV, duas outras variações do AODV também foram avaliadas neste estudo: o U-AODV, apresentado na subseção 7.2.1, e o GPS-AODV. O mecanismo GPS, apresentado no capítulo 5, também foi incluído na avaliação.

Duas métricas foram alvos para a avaliação apresentada nesta seção. A primeira é chamada de *Normalized Control Messages*, que representa a relação entre a quantidade de mensagens de controle enviadas pelas versões modificadas do AODV (U-AODV, GMLA-AODV, e GPS-AODV) e o padrão AODV. Essa métrica é calculada de acordo com a equação 7.2.

$$N_c = 1 - \frac{C_{Improved-AODV}}{C_{AODV}} \quad (7.2)$$

O termo N_c representa a métrica *Normalized Control Messages* do padrão AODV e $C_{Improved-AODV}$ é a quantidade de mensagens de controle das versões do AODV melhorado, enquanto C_{AODV} é o AODV padrão.

A segunda métrica adotada é o *Data Delivery Rate*, que representa a taxa de mensagens enviadas que chegaram aos nodos destinos. A seguir, as configurações da simulação são detalhadas.

Tabela 7.3: Parâmetros de Taxa de Dados

Parameters	Value
<i>Carga Baixa (LL)</i>	0.5 mensagem/segundo
<i>Carga Média (ML)</i>	1.0 mensagem/segundo
<i>Carga Alta (HL)</i>	2.0 mensagem/segundo

7.3.1 Configurações da Simulação

As simulações foram igualmente realizadas com o simulador OMNET++ sob o framework INETMANET. Como camada MAC, foi utilizado o padrão IEEE 802.15.4 modo sem *beacon*. Os parâmetros de rede e do GMLA(6) utilizados estão descritos nas Tabelas 7.1 e 7.2 que estão na seção 7.1.1.1.

O padrão de mobilidade que foi usado nas simulações foi o *Manhattan*. O número de nodos utilizados foi 50 nodos, sendo 10 nodos sorvedouros aleatoriamente escolhidos por uma distribuição uniforme e de 1 a 10 nodos fontes. Cada nodo fonte utilizou três perfis diferentes de geradores de tráfego, como mostra a tabela 7.3. Os valores da tabela representam o número de pacotes enviados por segundo, e cada pacote possui 64 Bytes.

Em cada simulação, o número de nodos fontes foi incrementado de 1 a 10. Isto significa que foram feitas 10 simulações para cada protocolo. Adicionalmente, cada simulação foi repetida 10 vezes para aumentar o nível de confiança. Os resultados apresentados são as médias das simulações repetidas.

7.3.2 Resultados Obtidos

A primeira métrica *Normalized Control Message* é mostrada na figura 7.5. Estas medições representam a quantidade de mensagens de controle enviadas pelas implementações U-AODV, GPS-AODV, e GMLA-AODV em relação ao AODV padrão (equação 7.2). Como já mencionado, três diferentes tráfegos foram utilizados (tabela 7.3).

Analisando os resultados, o U-AODV enviou cerca de 15% a 0.4% menos mensagens que o AODV. Com mais de dois nodos fontes, o U-AODV cai para 5% a menos mensagens de controle até chegar perto de 0% de diferença em relação ao AODV. Além disso, o U-AODV teve o mesmo desempenho em termos de *Data Delivery Rate* que o AODV padrão.

Para o GPS-AODV, os resultados foram melhores nas duas métricas em relação ao U-AODV. A versão GPS-AODV utilizou menos mensagens de controle, variando entre 69% menos mensagens com 1 nodo fonte em configuração com baixa carga a 15% com 10 nodos fontes em configuração com alta carga.

Entretanto, a versão GMLA-AODV teve um desempenho melhor em termos de *Normalized Control Messages*. Com 66% a menos de mensagens de controle enviadas com um nodo fonte, em baixa carga, e até 27% com 10 nodos fontes em alta carga.

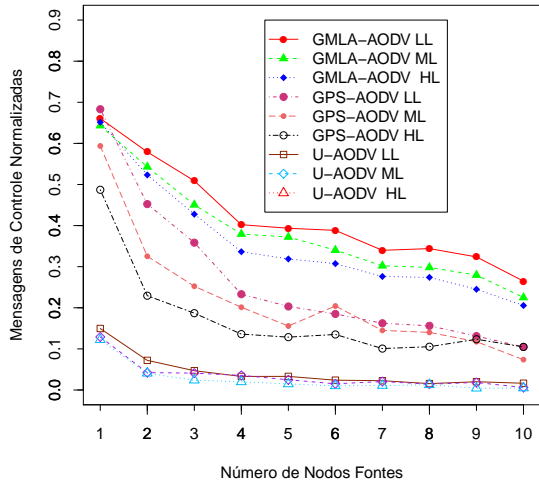


Fig. 7.5: Mensagens de Controle Normalizadas.

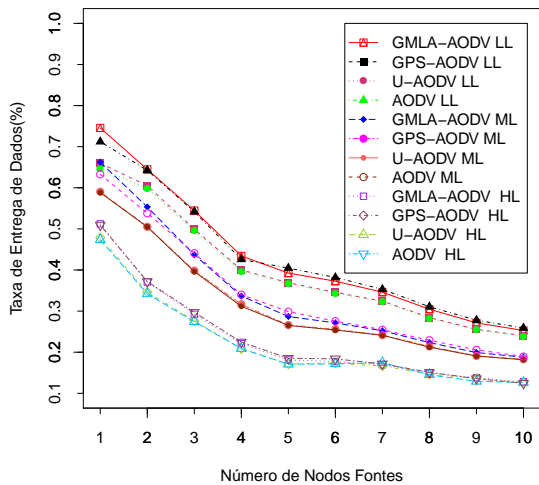


Fig. 7.6: Taxa de Entrega de Dados.

Em baixa carga, a taxa de entrega de dados (*Data Delivery Rate*) do GMLA foi 10% melhor que o U-AODV e 11% melhor que o AODV para 1 nodo fonte e cai para 1.3% melhor que o U-AODV e 2% melhor que o AODV com 10 nodos fontes. Com a configuração de média, a taxa de entrega de dados do GMLA foi de 8% melhor que o U-AODV e 9% melhor que o AODV para 1 nodo fonte e cai para menos de 1% melhor para ambos AODV e U-AODV para 10 nodos fontes.

Com a configuração de alta carga, a taxa de entrega de dados do GMLA-AODV foi 4% melhor do que o U-AODV e 5% melhor que o AODV para 1 nodo fonte e, também, cai para menos de 1% para ambos AODV e U-AODV para 10 nodos fontes. A versão GPS-AODV teve essencialmente a mesma taxa de entrega de dados que o GMLA-AODV, variando entre 1% melhor para o GPS-AODV em todas as configurações, com exceção para 1 nodo fonte em baixa carga de mensagens.

Os resultados mostraram que o GMLA pode trazer grande melhoria ao AODV, reduzindo o número de mensagens de controle e aumentando a taxa de entrega de dados. Esta melhoria trouxe um impacto significativo no tráfego da rede, provendo mais largura de banda para os nodos. O consumo de energia não foi medido, mas é possível assumir que quando o número de mensagens transmitidas é reduzido, a tempo de vida da rede aumenta.

7.3.3 Considerações Finais

Nesta seção, foi apresentado um estudo de caso para avaliar o impacto que o GMLA pode trazer se combinado com protocolos de

roteamento MANETs, como o AODV. De fato, todos os protocolos de roteamento MANETs, tanto reativos como proativos, podem ser combinados ao GMLA a fim de aumentar seus desempenhos.

Pelos experimentos realizados, pode-se notar quais foram os limites do GMLA para a solução do problema identificado. Em síntese, o problema identificado foi que a quantidade de mensagens de controle poderia ser reduzida para manter o conhecimento de conectividade entre os nodos em um salto. Com isso, observou-se que o mecanismo GMLA poderia atuar no escalonamento de envio de mensagens com intuito de reduzir a quantidade de mensagens de controle.

Para esse problema identificado, verificou-se que os limites dos benefícios que a solução proposta pôde trazer para a rede. Nos casos em que há poucos nodos fonte, em torno de 3, a solução possibilitou um aumento na taxa de entrega de dados de até 10%. Mas, a medida que o número de nodos fontes cresce, a taxa de entrega provida pela solução proposta reduz ao nível do AODV padrão. Observa-se, também, a redução de ganho pela métrica de controle normalizado, devido ao aumento de nodos fontes. Isto indica que para redes com grande número de nodos fontes, como a aplicação da corrida descrita no capítulo 1 e avaliada no 4, a solução não proveja ganho aparente.

Apesar das limitações discutidas, soluções como o GMLA, que não requerem sistemas de posicionamento, são mais baratas de implementar em comparação às que adotam dispositivos GPS. Além disso, quando se tem um padrão de mobilidade com certo nível de aleatoriedade, o GMLA é melhor que as soluções baseadas em GPS.

A proposta RACE apresentada no capítulo 3 poderia se beneficiar da combinação do GMLA, entretanto, a implementação foi re-

alizada no protocolo MANET AODV por se tratar de uma proposta largamente aceita e debatida na literatura. Além disso, como a taxa de entrega do RACE ficou em torno de 90% não seria necessário uma implementação com GMLA para aumentar o seu desempenho. Além disso, como já discutido, a grande quantidade de nodos fontes presentes no cenário da corrida poderia eliminar os benefícios da integração do GMLA a qualquer outro protocolo de roteamento, para o fim de aumento na taxa de entrega.

A seguir serão apresentadas as conclusões finais do trabalho de tese e perspectivas futuras.

Parte IV

Conclusões e Trabalhos Futuros

Capítulo 8

Conclusões Finais e Trabalhos Futuros

A área de RSSF tem grandes desafios a serem superados para que a vasta gama de possíveis aplicações possa ser implementada em cenários reais e assim trazer benefícios à sociedade moderna. Um destes desafios é a união das RSSF com o fator de mobilidade. RSSF em cenários com mobilidade estão presentes em duas situações: i) quando o agente que tem um sensor anexado pode se mover e/ou ii) quando os sensores são fixos e um agente móvel interage com a RSSF. O fator mais importante que a mobilidade influencia em qualquer rede é a mudança constante de topologia. Esta característica tem grandes influências na pilha de protocolos de rede.

A constante mudança de topologia da rede torna a manutenção de rotas um desafio para os protocolos de roteamento. Como os enlaces entre os nodos se formam e se desfazem frequentemente, os protoco-

los de rede necessitam de um mecanismo para manter o conhecimento da vizinhança e de todo o caminho entre os nodos fontes e destinos. Esse mecanismo é composto por mensagens de controle e torna possível a entrega de mensagens, entretanto pode provocar uma sobrecarga na rede. Os protocolos de roteamento MANET foram projetados para construir todo o caminho da rota. Apesar de cumprirem o objetivo de disponibilizar rotas, observa-se pelos experimentos que a carga gerada pelas mensagens de controle prejudica o desempenho destes protocolos quando se faz necessário construir e manter várias rotas.

Por esse motivo, a primeira questão tratada nesta tese foi o desenvolvimento de um protocolo de roteamento que pudesse realizar a entrega de mensagens nodo a nodo minimizando a sobrecarga gerada pelas mensagens de controle. Sendo assim, o protocolo RACE foi idealizado com intuito não de construir rotas, mas sim de realizar a manutenção de vizinhança de um salto. O simples fato de não manter uma rota proporcionou uma redução substancial na sobrecarga da rede. O nodo sensor consome energia para sensorar, comunicar e processar os dados. Entretanto, mais energia é necessária para realizar a comunicação de dados do que outros processos. Como cada mensagem tem um custo na rede, conclui-se que é substancial minimizar ao máximo as mensagens de controle da rede. Isto pode garantir a melhora no desempenho dos protocolos e aumentar a vida útil da rede.

Pôde-se notar, nos experimentos feitos com protocolos de redes MANET, os efeitos gerados pelas mensagens de controle. Visto que as mensagens de dados eram as mesmas para todos os protocolos testados, tanto para os protocolos geográficos como para os protocolos de redes MANET. O desempenho, em parte, foi influenciado pela carga de mensagens de controle gerada. A necessidade de descobrir e man-

ter rota, característica dos protocolos de redes MANET, foi o fator que pode ser apontado como causa do baixo desempenho. Essa afirmação vem da observação dos experimentos realizados no capítulo 7, seção 7.2 de estudo de caso. Pode-se notar que com apenas 1 nodo fonte, a taxa de entrega ficou em torno de 70% e quanto mais nodos fontes foram acrescentados, a taxa de entrega foi reduzida. Observando o desempenho dos protocolos por esse prisma, é conclusivo que com 300 nodos fontes a taxa de entrega se aproxime dos valores observados nos experimentos realizados no capítulo 4 subseção 4.2.3. Pode-se afirmar, então, que com o aumento de nodos fontes, aumentaram os descobrimentos e manutenções de rotas, o que influenciou diretamente no desempenho dos protocolos.

Com uma grande quantidade de nodos fontes, conclui-se então que os protocolos de roteamento para redes MANET são inadequados por proverem um baixo desempenho na entrega de mensagens. Os protocolos de roteamento geográfico são mais apropriados para esse fim. Entretanto, os protocolos de roteamento geográfico se valem da premissa de ter um sistema de posicionamento. Como não foram feitos estudos sobre a influência dos sistemas de posicionamento no consumo de energia e precisão, seria necessário uma investigação futura para determinar tais influências.

Outra questão abordada na tese foi o provimento de priorização de mensagens para mensagens com prioridade em cenários onde há uma grande quantidade de nodos gerando mensagens de dados. As mensagens marcadas com prioridade pela aplicação tinham preferências na ocupação de *buffer* e no encaminhamento. Como resultado, observou-se um atraso fim-a-fim substancialmente inferior em relação a um protocolo da literatura projetado para lidar com garantias *soft-real*

time. Pode-se concluir que o protocolo RACE cumpriu o seu papel no que se refere a priorização de mensagens para reduzir o atraso na entrega fim-a-fim.

Além disso, o RACE introduziu o conceito de conhecimento de *buffer* da vizinhança para ajudar na retransmissão de mensagens. Sabe-se que a perda de mensagens é provocada pela desistência de aquisição do canal, pela interferência ou pela perda de enlace. Contudo, em cenários onde há uma grande quantidade de nodos fontes realizando comunicação "muitos para um", a falta de espaço em *buffer* é outro fator de perda de mensagens. Esse fator foi igualmente observado por outro trabalho da literatura [Agarkhed, Biradar e Mytri 2012], posterior ao RACE, que propôs um mecanismo parecido ao do RACE para lidar com a questão de esgotamento de *buffer*. Sendo assim, nota-se pelos experimentos realizados, que prover o conhecimento de *buffer* na vizinhança, em certas situações, pode prover um aumento no desempenho dos protocolos de roteamento no que se refere à taxa de entrega de dados.

O desenvolvimento e estudo do protocolo RACE aponta para outro trabalho de investigação futura. A escolha dos pesos das métricas foi influenciada pelas diferentes densidades da rede. Entretanto, para avaliar qual seria a melhor distribuição de pesos, seria necessário realizar, a priori, experimentos com ajustes nos pesos para identificar a melhor configuração. Com isso, nota-se que há espaço para pesquisa de um mecanismo que possa realizar o ajuste dinâmico dos pesos em tempo de execução, o que tornaria o protocolo mais fácil de ser implementado e reduziria a questão de administração de configurações dependentes do cenário.

Após o desenvolvimento do RACE, do estudo da influência das suas métricas para o desempenho do protocolo e uma análise comparativa com protocolos de roteamento MANET, o trabalho foi focado para mitigar os efeitos da mobilidade na conectividade dos nodos. Tratar tais efeitos foram as questões trabalhadas na segunda parte da tese.

Embora os modelos de obtenção do conhecimento de conectividade projetados terem obtido um desempenho satisfatório nos experimentos realizados, a questão de perda completa do sinal devido à presença de obstáculos não foi estudada. Esta questão pode prejudicar os modelos visto que foi considerada somente a variação do sinal de forma suave. Como trabalhos futuros pretende-se estender os estudos em ambientes com presença de obstáculos para verificar o impacto nos modelos e realizar as adaptações necessárias.

A primeira questão desta segunda parte da tese, refere-se à obtenção da informação da estimativa de qualidade de enlace para construir o conhecimento de conectividade local. Para suplantiar os desafios contidos na construção do conhecimento de conectividade local, foram apresentadas duas propostas. A primeira proposta, O-BD, foi exclusivamente projetada para lidar com qualquer padrão de mobilidade, baseando-se em um modelo capaz de capturar as variações mais básicas do movimento, aproximação e afastamento, por meio da variação da qualidade do enlace. O segundo modelo, GMLA, foi uma extensão do primeiro para lidar com mudanças no ambiente, neste caso, com mudanças no padrão de mobilidade. O GMLA mostrou-se eficiente na questão de readaptação ao ambiente devido a mudanças no padrão de mobilidade.

Como a mobilidade e a sua influência sobre os enlaces é o ob-

jeto de estudo da segunda parte desta tese, foram necessários avaliar os modelos propostos e os modelos da literatura com diferentes padrões de mobilidade. Os modelos de mobilidade são modelos sintéticos, que de certa forma refletem alguns padrões de mobilidades reais. Como o modelo de mobilidade Manhattan, que reflete a mobilidade em áreas urbanas ou o modelo *Reference Point Group Mobility* que espelha a mobilidade em grupo. Já o modelo de mobilidade Gauss-Markov não representa um padrão de mobilidade real [Liang e Haas 2003], mas tem a finalidade de avaliar os limites dos modelos de predição de conectividade, que têm a mobilidade como o seu principal objeto de estudo. Por isso, o desenvolvimento dos modelos de conectividade local levou em consideração o estudo de diferentes padrões de mobilidade para determinar as suas influências sobre os modelos propostos e sobre os trabalhos encontrados na literatura.

Pode-se dizer que aplicações e protocolos de redes móveis podem se beneficiar com a adoção de modelos de predição de conectividade. Por esse motivo, foram dispendidos esforços com intuito de verificar os benefícios que os modelos de predição de conectividade podem trazer. Foi identificado um problema específico dos protocolos para redes MANTE quanto ao seu baixo desempenho na taxa de entrega visto o dispendioso mecanismo de descobrimento e manutenção de rotas. A intenção dos modelos foi atuar especificamente no escalonamento das mensagens *HELLO*, que é o mecanismo utilizado pelos protocolos MANTE para manter o contato com a sua vizinhança, ou seja, ter a noção de conectividade.

Existem outras questões que podem ser trabalhadas nos protocolos MANET para melhorar os seus desempenhos com a adoção de modelos de predição de conectividade local. Podem-se construir ro-

tas baseada na conectividade entre os nodos. Neste caso, poderiam ser construídas rotas baseadas na maior duração de enlaces, aumentando o tempo de duração da rota, e por consequência tornando as rotas mais confiáveis. Esta modificação pode ser aliada à modificação já experimentada sobre o escalonamento de envios de mensagens *HELLO*. Entretanto, retomando a observação do comportamento dos protocolos MANTE, quanto ao número de nodos fontes, nenhuma solução de conectividade poderia melhorar os protocolos MANET em casos de grande quantidade de nodos fontes.

Referências Bibliográficas

[Abbas e Kure 2010]ABBAS, A. M.; KURE, O. Quality of service in mobile ad hoc networks: a survey. Int. J. Ad Hoc Ubiquitous Comput., Inderscience Publishers, Inderscience Publishers, Geneva, SWITZERLAND, v. 6, p. 75–98, 2010.

[Agarkhed, Biradar e Mytri 2012]AGARKHED, J.; BIRADAR, G. S.; MYTRI, V. D. Energy efficient qos routing in multi-sink wireless multimedia sensor networks. IJCSNS International Journal of Computer Science and Network Security, v. 12, p. 25–31, 2012.

[Akkaya e Younis 2005]AKKAYA, K.; YOUNIS, M. A survey on routing protocols for wireless sensor networks. Ad Hoc Networks, Elsevier, v. 3, n. 3, p. 325–349, 2005.

[Akyildiz et al. 2002]AKYILDIZ, I. F. et al. Wireless sensor networks: a survey. [S.l.]: Computer Networks, 2002. 393–422 p.

[Ali, Latiff e Fisal 2005]ALI, A.; LATIFF, L.; FISAL, N. GPS-free indoor location tracking in mobile ad hoc network (MANET) using [S.l.], 2005. 251–255 p.

- [Allred et al. 2007]ALLRED, J. et al. Sensorflock: an airborne wireless sensor network of micro-air vehicles. In: Proceedings of the 5th international conference on Embedded networked sensor systems. New York, NY, USA: ACM, 2007. p. 117–129.
- [Amundson e Koutsoukos 2009]AMUNDSON, I.; KOUTSOUKOS, X. D. A survey on localization for mobile wireless sensor networks. In: Proceedings of the 2nd international conference on Mobile entity localization and tracking in GPS-less environments. Berlin, Heidelberg: Springer-Verlag, 2009. p. 235–254.
- [Araújo, Kaiser e Becker 2012]ARAÚJO, G. M. de; KAISER, J.; BECKER, L. B. An optimized markov model to predict link quality in mobile wireless sensor networks. 2012 IEEE Symposium on Computers and Communications (ISCC), IEEE Computer Society, Los Alamitos, CA, USA, v. 0, p. 000307–000312, 2012.
- [Araújo et al. 2012]ARAÚJO, G. M. de et al. An evolutionary approach to improve connectivity prediction in mobile wireless sensor networks. Procedia Computer Science, v. 10, n. 0, p. 1100–1105, 2012.
- [Araújo e Becker 2011]ARAÚJO, G. Medeiros de; BECKER, L. B. A network conditions aware geographical forwarding protocol for real-time applications in mobile wireless sensor networks. In: Proceedings of the 2011 IEEE International Conference on Advanced Information Networking and Applications. Washington, DC, USA: IEEE Computer Society, 2011. (AINA '11), p. 38–45.
- [Aschenbruck et al. 2010]ASCHEBRUCK, N. et al. Bonnmotion: a mobility scenario generation and analysis tool. In: Proceedings of the 3rd International ICST Conference on Simulation Tools and

- Techniques. [S.l.]: ICST (Institute for Computer Sciences, Social- Informatics and Telecommunications Engineering), 2010. (SIMUTo- ols '10), p. 1–10.
- [Atay e Bayazit 2007]ATAY, N.; BAYAZIT, B. Mobile Wireless Sensor Network Connectivity Repair with K-Redundancy. 2007.
- [Bai, Sadagopan e Helmy 2003]BAI, F.; SADAGOPAN, N.; HELMY, A. Important: A framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks. In: IEEE. INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies. [S.l.], 2003. v. 2, p. 825–835.
- [Camp, Boleng e Davies 2002]CAMP, T.; BOLENG, J.; DAVIES, V. A survey of mobility models for ad hoc network research. Wireless communications and mobile computing, Wiley Online Library, v. 2, n. 5, p. 483–502, 2002.
- [Chakeres e Perkins 2009]CHAKERES, I.; PERKINS, C. Dynamic manet on-demand (dymo) routing. In: IETF Internet-Draft, draft-ietf-manet-dymo-17.txt. [S.l.: s.n.], 2009.
- [Chen e Varshney 2004]CHEN, D.; VARSHNEY, P. K. QoS support in wireless sensor networks: A survey. In: Proc of the 2004 International Conference on Wireless Networks ICWN 2004 Las Vegas Nevada USA. [S.l.]: Citeseer, 2004. v. 13244, p. 227–233.
- [Chen, Jones e Jayalath 2007]CHEN, S.; JONES, H.; JAYALATH, D. Effective link operation duration: a new routing metric for mobile ad hoc networks. In: CITeseer. International Conference on Signal Processing and Communication Systems. [S.l.], 2007.

- [Chipara et al. 2006]CHIPARA, O. et al. Real-time power-aware routing in sensor networks. [S.l.]: IEEE, 2006. 83–92 p.
- [Cormen et al. 1990]CORMEN, T. H. et al. Introduction to Algorithms, first edition. 1990. 558–565 p.
- [Couto et al. 2003]COUTO, D. S. J. D. et al. A high-throughput path metric for multi-hop wireless routing. In: Proceedings of the 9th annual international conference on Mobile computing and networking. [S.l.: s.n.], 2003. (MobiCom '03).
- [De Rosa, Malizia e Mecella 2005]De Rosa, F.; MALIZIA, A.; MECCELLA, M. Disconnection prediction in mobile ad hoc networks for supporting cooperative work. Pervasive Computing, IEEE, IEEE, v. 4, n. 3, p. 62–70, 2005.
- [Deak, Curran e Condell 2011]DEAK, G.; CURRAN, K.; CONDELL, J. Filters for rssi-based measurements in a device-free passive localisation scenario. International Journal on Image Processing & Communications, v. 15, p. 23–34, 2011.
- [Dunbabin, Usher e Corke 2005]DUNBABIN, M.; USHER, K.; CORKE, P. Data collection, storage and retrieval with an underwater sensor network. Proc. IEEE SenSys, p. 154–165, 2005.
- [Eisenman et al. 2009]EISENMAN, S. et al. BikeNet: A mobile sensing system for cyclist experience mapping. ACM Transactions on Sensor Networks (TOSN), ACM, v. 6, p. 1–39, 2009.
- [Eriksson et al. 2008]ERIKSSON, J. et al. The pothole patrol: using a mobile sensor network for road surface monitoring. In: Proceedings of the 6th international conference on Mobile systems, applications, and services. New York, NY, USA: ACM, 2008. p. 29–39.

- [Erman-Tüysüz et al. 2008]ERMAN-TÜYSÜZ, A. et al. Enabling mobility in heterogeneous wireless sensor networks cooperating with uavs for mission-critical management. IEEE Wireless Communications, IEEE Communication Society, v. 15, n. 6, p. 38–46, December 2008.
- [Farkas et al. 2008]FARKAS, K. et al. Link quality prediction in mesh networks. Computer Communications, Elsevier, v. 31, n. 8, p. 1497–1512, 2008.
- [Fdili Youssef Fakhri 2012]FDILI YOUSSEF FAKHRI, D. A. O. A. Impact of queue buffer size awareness on single and multi service real-time routing protocols for wsns. International Journal of Communication Network and Information Security, Kohat University of Science and Technology, v. 4, n. 2, p. 104–111, 2012.
- [Felemban e Ekici 2006]FELEMBAN, E.; EKICI, E. MMSPEED: multipath Multi-SPEED protocol for QoS guarantee of reliability and. Timeliness in wireless sensor networks. Ieee Transactions On Mobile Computing, v. 5, n. 6, p. 738–754, 2006.
- [Freitas et al. 2010]FREITAS, E. Pignaton de et al. Coordinating aerial robots and unattended ground sensors for intelligent surveillance systems. International Journal of Computers, Communications & Control, University of Oradea, v. 5, n. 1, p. 52–70, 2010.
- [Garcia-Macias e Gomez 2007]GARCIA-MACIAS, J.; GOMEZ, J. Manet versus wsn. In: MAHALIK, N. (Ed.). Sensor Networks and Configuration. [S.l.]: Springer Berlin Heidelberg, 2007. p. 369–388.
- [Goldberg 1989]GOLDBERG, D. Genetic algorithms in search, optimization, and machine learning. [S.l.]: Addison-wesley, 1989.

- [Guha e Sarkar 2008]GUHA, R.; SARKAR, S. Characterizing temporal snr variation in 802.11 networks. Vehicular Technology, IEEE Transactions on, IEEE, v. 57, n. 4, p. 2002–2013, 2008.
- [Hanzo e Tafazolli 2007]HANZO, L.; TAFAZOLLI, R. A Survey of QoS Routing Solutions for Mobile Ad hoc Networks. [S.l.]: IEEE, 2007. 50–70 p.
- [He et al. 2003]HE, T. et al. SPEED: A Stateless Protocol for Real-Time Communication In Sensor Networks. In: Proceedings of International Conference on Distributed Computing Systems [C], Providence, RI. [S.l.: s.n.], 2003.
- [Hong et al. 1999]HONG, X. et al. A group mobility model for ad hoc wireless networks. In: ACM. Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems. [S.l.], 1999. p. 53–60.
- [Huang et al. 2010]HUANG, J. et al. Anatomizing application performance differences on smartphones. In: Proceedings of the 8th international conference on Mobile systems, applications, and services. New York, NY, USA: ACM, 2010. p. 165–178.
- [Hull, Jamieson e Balakrishnan 2004]HULL, B.; JAMIESON, K.; BALAKRISHNAN, H. Mitigating congestion in wireless sensor networks. In: Proceedings of the 2nd international conference on Embedded networked sensor systems. [S.l.]: ACM, 2004. p. 134–147.
- [Jacquet et al. 2001]JACQUET, P. et al. Optimized link state routing protocol for ad hoc networks. In: IEEE UNMIC. Pakistan: [s.n.], 2001. p. 62–68.

- [Jain 1991]JAIN, R. The art of computer systems performance analysis. [S.l.]: John Wiley & Sons New York, 1991.
- [Jha e Hassan 2002]JHA, S.; HASSAN, M. Engineering Internet QoS. [S.l.]: Artech House Publishers, 2002.
- [Jha, Tan e Maxion 2001]JHA, S.; TAN, K.; MAXION, R. A. Markov chains, classifiers, and intrusion detection. In: Proceedings of the 14th IEEE workshop on Computer Security Foundations. Washington, DC, USA: IEEE Computer Society, 2001. p. 206–.
- [Jin et al. 2009]JIN, Z. et al. A Survey on Position-Based Routing Algorithms in Wireless Sensor Networks. Algorithms, v. 2, n. 1, p. 158–182, fev. 2009.
- [Jindal A.K.Verma 2011]JINDAL A.K.VERMA, S. B. V. How the two adhoc networks can be different:manet & wsns. International Journal on Computer Science and Technology, TechRepublic, v. 2, n. 4, p. 122–126, 2011.
- [Johnson e Broch 1999]JOHNSON, D. A. M. D. B.; BROCH, J. Dsr: The dynamic source routing protocol for multi-hop wireless ad hoc networks. Ad Hoc Networking, Addison-Wesley, v. 5, p. 139–172, 1999.
- [Karl e Willig 2005]KARL, H.; WILLIG, A. Protocols and Architectures for Wireless Sensor Networks. [S.l.]: John Wiley e Sons, 2005.
- [Karp e Kung 2000]KARP, B.; KUNG, H.-T. Gpsr: Greedy perimeter stateless routing for wireless networks. In: ACM. Proceedings of the 6th annual international conference on Mobile computing and networking. [S.l.], 2000. p. 243–254.

- [Krogmann et al. 2009]KROGMANN, M. et al. Impact of link quality estimation errors on routing metrics for wireless sensor networks. In: IEEE. Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on. [S.l.], 2009. p. 397–402.
- [Kurose e Ross 2006]KUROSE, J.; ROSS, K. Computer Networks: A Top Down Approach Featuring the Internet. [S.l.]: Pearson Addison Wesley, 2006.
- [Landsiedel, Wehrle e Götz 2005]LANDSIEDEL, O.; WEHRLE, K.; GÖTZ, S. Accurate prediction of power consumption in sensor networks. In: CITeseer. Proceedings of The Second IEEE Workshop on Embedded Networked Sensors (EmNetS-II). [S.l.], 2005.
- [Latiff, Ali e Faisal 2005]LATIFF, L.; ALI, A.; FISAL, N. O. Development of an indoor GPS-free self-positioning system for MANET. 2005.
- [Lenders, Wagner e May 2006]LENDERS, V.; WAGNER, J.; MAY, M. Analyzing the impact of mobility in ad hoc networks. In: Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality. New York, NY, USA: ACM, 2006. p. 39–46.
- [Liang e Haas 2003]LIANG, B.; HAAS, Z. J. Predictive distance-based mobility management for multidimensional pcs networks. Networking, IEEE/ACM Transactions on, IEEE, v. 11, p. 718–732, 2003.

- [Liu e Hong 2007]LIU, J.; HONG, X. A traffic-aware energy efficient routing protocol for wireless sensor networks. ASAP, p. 142–147, 2007.
- [Liu et al. 2004]LIU, T. et al. Implementing software on resource-constrained mobile sensors: experiences with impala and zebra-net. In: Proceedings of the 2nd international conference on Mobile systems, applications, and services. [S.l.: s.n.], 2004. p. 256–269.
- [Lu et al. 2002]LU, C. et al. Rap: A real-time communication architecture for large-scale wireless sensor networks. In: IEEE. Real-Time and Embedded Technology and Applications Symposium, 2002. Proceedings. Eighth IEEE. [S.l.], 2002. p. 55–66.
- [McQuarrie e Tsai 1998]MCQUARRIE, A. D.; TSAI, C.-L. Regression and time series model selection. [S.l.]: World Scientific, 1998.
- [Mottola e Picco 2011]MOTTOLA, L.; PICCO, G. P. Programming wireless sensor networks: Fundamental concepts and state of the art. ACM Computing Surveys (CSUR), ACM, v. 43, n. 3, p. 19, 2011.
- [Neumann C. Aichele e Wunderlich 2007]NEUMANN C. AICHELE, M. L. A.; WUNDERLICH, S. Better Approach To Ad-Hoc Networking (B.A.T.M.A.N). [S.l.]: Network Working Group, 2007.
- [Nicholson e Noble 2008]NICHOLSON, A.; NOBLE, B. Breadcrumbs: Forecasting mobile connectivity. In: ACM. Proceedings of the 14th ACM international conference on Mobile computing and networking. [S.l.], 2008. p. 46–57.
- [Peng e Sichitiu 2006]PENG, R.; SICHITIU, M. L. Angle of arrival localization for wireless sensor networks. In: Sensor and Ad Hoc

- Communications and Networks (SECON). [S.l.: s.n.], 2006. v. 1, p. 374–382.
- [Perkins e Bhagwat 1994]PERKINS, C. E.; BHAGWAT, P. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. SIGCOMM Comput. Commun. Rev., v. 24, n. 4, 1994.
- [Perkins e Royer 1999]PERKINS, C. E.; ROYER, E. Ad-hoc on-demand distance vector routing. In: Proceeding of 2nd IEEE Workshop on Mobile Computing & Systems and Applications. [S.l.: s.n.], 1999.
- [Priyantha et al. 2001]PRIYANTHA, N. et al. The cricket compass for context-aware mobile applications. In: ACM. Proceedings of the 7th annual international conference on Mobile computing and networking. [S.l.], 2001. p. 1–14.
- [Quintana e Cabrera 2012]QUINTANA, A. A.; CABRERA, A. T. Chapter 7: Simulation of Multihop Wireless Networks in OMNeT++ in Simulation in Computer Network Design and Modeling: Use and Analysis. [S.l.]: Information Science Reference (an imprint of IGI Global), 2012.
- [Rezayat et al. 2010]REZAYAT, P. et al. A Novel Real-Time Power Aware Routing Protocol in Wireless Sensor Networks. Journal of Computer Science, v. 10, n. 4, p. 300–305, 2010.
- [Sabitha e Thangavelu 2011]SABITHA, R.; THANGAVELU, T. Performance enhancement of fuzzy logic based transmission power control in wireless sensor networks using markov based rssi prediction. European Journal of Scientific Research (EJSR), Euro Journal Publishers, v. 59, n. 1, p. 68–84, 2011.

[Su, Lee e Gerla 2000]SU, W.; LEE, S.-J.; GERLA, M. Mobility prediction in wireless networks. In: Proceedings of IEEE International Conference on Computer Communication Networks. [S.l.]: IEEE, 2000. (ISCC '00), p. 4–9.

[Su, Lee e Gerla 2000]SU, W.; LEE, S. ju; GERLA, M. Mobility prediction in wireless networks. In: Proceedings of IEEE MILCOM 2000. [S.l.: s.n.], 2000.

[Taneja e Kush 2010]TANEJA, S.; KUSH, A. A survey of routing protocols in mobile ad hoc networks. International Journal of innovation, Management and technology, v. 1, p. 2010–0248, 2010.

[Valente et al. 2011]VALENTE, J. a. et al. An air-ground wireless sensor network for crop monitoring. Sensors, v. 11, n. 6, p. 6088–6108, 2011.

[Varga et al. 2001]VARGA, A. et al. The OMNeT++ discrete event simulation system. In: Proceedings of the European Simulation Multiconference. [S.l.: s.n.], 2001. p. 319–324.

[Wang Z. e Zhou 2005]WANG Z., L. L.; ZHOU, M. C. Protocols and applications of ad-hoc robot wireless communication networks: An overview. International Journal of Intelligent Control and Systems, v. 10, p. 296–303, 2005.

[Warneke et al. 2001]WARNEKE, B. et al. Smart dust: Communicating with a cubic-millimeter computer. Computer, IEEE Computer Society Press, Los Alamitos, CA, USA, v. 34, n. 1, p. 44–51, 2001.

[Woo e Culler 2003]WOO, A.; CULLER, D. Evaluation of efficient link reliability estimators for low-power wireless networks. [S.l.]: Computer Science Division, University of California, 2003.

[Woo, Tong e Culler 2003]WOO, A.; TONG, T.; CULLER, D. Taming the underlying challenges of reliable multihop routing in sensor networks. In: ACM. Proceedings of the 1st international conference on Embedded networked sensor systems. [S.l.], 2003. p. 14–27.

[Zavlanos, Egerstedt e Pappas 2011]ZAVLANOS, M. M.; EGERS-TEDT, M. B.; PAPPAS, G. J. Graph-theoretic connectivity control of mobile robot networks. Proceedings of the IEEE, v. 99, p. 1525–1540, 2011.