

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

Lucas Guardalben

**Auto-Otimização e Auto-Configuração nos
Protocolos de Roteamento OLSR e AODV para
Redes em Malha Sem Fio**

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

Prof. Dr. João Bosco Mangueira Sobral

Florianópolis, fevereiro de 2009

Auto-Otimização e Auto-Configuração nos Protocolos de Roteamento OLSR e AODV para Redes em Malha Sem Fio

Lucas Guardalben

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, Área de Concentração Sistemas de Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação

Prof. Dr. Frank Augusto Siqueira
**Coordenador do Curso de Pós-Graduação
em Ciência da Computação**

Prof. Dr. João Bosco Manguiera Sobral
(Orientador)

Banca Examinadora

Prof. Dr. Eduardo Camponogara

Prof. Dr. Frank Augusto Siqueira

Prof. Dr. Roberto Willrich

*"Transportai um punhado de terra
todos os dias e fareis uma montanha."*

Confúcio

Dedico este trabalho à tríade: família,
amigos e Deus

AGRADECIMENTOS

Primeiramente gostaria de agradecer a todos os membros da minha família, em especial meus pais Aleri Luiz Guardalben e Ivete Ana Guardalben juntamente com meus irmãos, Cássio Fernando Guardalben e Laís Guardalben por todo apoio, ensinamentos e palavras confortantes nos momentos certos. Sem essa base de apoio tão sólida esse trabalho não poderia ser concluído.

Ao meu orientador João Bosco Mangureira Sobral, pela amizade, confiança, ensinamentos e orientação durante o início meio e fim deste trabalho. Um grande amigo, que aprendi a respeitar e admirar pelo seu caráter e sua honestidade.

Gostaria também de agradecer a minha namorada Claudia pela compreensão e ajuda nas correções parciais deste trabalho.

Gostaria de agradecer ao professor Azzedine Boukerche e a professora Mirela S.M.A Notare por todo suporte oferecido antes e durante a minha estada no Canadá, para apresentação do meu trabalho no evento LCN 2008, na cidade de Montreal.

Agradeço em especial ao professor Eduardo Camponogara pelas contribuições no modelo de auto-otimização proposto e por sua paciência em me atender e ajudar no amadurecimento das técnicas de otimização utilizadas nessa dissertação.

Meu muito obrigado as minhas queridas tias, Iliane Remonatto Tussi e Isete Remonatto Rizzi, pelo excelente suporte dado nas correções dos artigos aceitos tanto em inglês e no resumo em espanhol.

Aos professores da Universidade Federal de Santa Catarina que contribuíram para o meu aprendizado.

Ao Programa de Pós-Graduação em Ciência da Computação e à Universidade Federal de Santa Catarina que contribuiu com a infra-estrutura necessária para o desenvolvimento deste trabalho.

A Vera Lúcia Sodr e Teixeira (Verinha) pela paci ncia e prestatividade em atender todas as minhas requisi es.

Agradeço a CAPES pelo apoio financeiro parcial durante o desenvolvimento da pesquisa.

Agradeço aos membros da banca pela aten o despendida nas corre es, avalia es e contribui es deste trabalho.

As minhas amigas Vivian Cremer e Clytia H. Tamashiro que reservaram um tempo para ler, discutir e contribuir para a concretiza o do meu trabalho, meus sinceros agradecimentos.

Aos amigos Jeferson (Bomba), Joelson (Tio), Denise Ferreira, Alexandre Parra Carneiro da Silva, Vinicius da Cunha Martins Borges (Cresio), Augusto Venâncio Neto (Brutal), Eduardo Cerqueira (Dudu), Rafael Brazão (Estranho), Almir Santos Albuquerque (Mikinha), Vasco da Silva Pinto, Wanderlan Paiva de Freitas (Manin Skurim), Rodrigo Mendonça de Souza (Pudim), Rodrigo Paiva Mendonça (Brodin), Rodrigo Cândido Borges (Grozop). Obrigado pela confiança, amizade, troca de conhecimento e momentos de descontrações.

Ao pessoal do LAED (Laboratório de Ambientes de Ensino a Distância), obrigado pelo apoio.

E por fim, o meu muito obrigado por todos que direta e indiretamente contribuíram para o sucesso deste trabalho.

Sumário

Lista de Figuras	v
Lista de Tabelas	viii
Lista de Abreviaturas	ix
Resumo	x
Abstract	xi
1 Introdução	1
1.1 Objetivo Geral	2
1.2 Trabalhos Relacionados	3
1.2.1 Arquiteturas Auto-Organizadas	3
1.2.2 Abordagens para Auto-Otimização e Auto-Configuração dos Parâmetros de Roteamento em Redes Sem Fio	5
1.2.3 Considerações	6
1.3 Contribuições	7
1.4 Apresentação do Trabalho	7
2 Redes em Malha Sem Fio	9
2.1 Definição	10
2.2 Diferenças entre as RMSF e as Redes <i>Ad Hoc</i> Móveis	12
2.3 Infra-Estrutura da Rede	13
2.4 Características	15
2.5 Cenários de Aplicação	16
2.6 Fatores que Afetam a Performance das RMSF	18
2.6.1 Projetos	20
2.7 Padronização das Redes em Malha sem Fio	26

2.8	Algumas Considerações	26
3	Auto-Organização	27
3.1	Conceituação	27
3.2	Definições	29
3.3	Propriedades dos Sistemas Auto-Organizados	30
3.4	Abordagens <i>Top-Down</i> e <i>Bottom-up</i>	33
3.4.1	Computação Autônoma (<i>Autonomic Computing</i>)	34
3.4.2	Projeto Para Computação Autônoma Proposto Por Kephart e Chess (2003)	36
3.5	Exemplos de Abordagens Auto-Organizadas em Redes de Computadores	40
3.6	Projetos	41
3.6.1	ANA: Autonomic Network Architecture	42
3.6.2	HAGGLE: An innovative Paradigm for Autonomic Opportunistic Communication	43
3.6.3	CASCADAS: Componentware for Autonomic, Situation-aware Communications and Dynamically Adaptable Services	44
3.7	Considerações Finais	45
4	Roteamento em Redes Em Malha Sem Fio	47
4.1	Introdução	47
4.2	Classificação dos Protocolos de Roteamento	48
4.2.1	Protocolos de Roteamento Pró-Ativos	49
4.2.2	<i>Optimised link state routing (OLSR)</i>	49
4.2.3	Protocolos de Roteamento Reativos	51
4.2.4	<i>Ad hoc on-demand distance vector (AODV)</i>	51
4.3	Visão Geral sobre Protocolos de Roteamento para Redes em Malha Sem Fio	56
4.4	Algumas Considerações	57
5	Proposta de uma Arquitetura para Auto-Organização	58
5.1	Introdução	58
5.2	Arquitetura de Auto-Organização em Nível Conceitual	59

5.2.1	Arquitetura de Auto-Organização Aplicada às Redes em Malha Sem Fio	60
5.2.2	Plataforma dos Agentes	62
5.2.3	Processos de Auto-Organização	64
5.2.4	Auto-Otimização	65
5.2.5	Auto-Configuração	66
5.2.6	Auto-Cura e Auto-Proteção	68
5.3	Considerações Finais	68
6	Modelagem e Simulação	70
6.1	Introdução	70
6.1.1	Software de Simulação Opnet (<i>Optimized Network Engineering Tools</i>)	71
6.2	Etapa 1: Construção dos Modelos de Processo	73
6.2.1	Modelo de Processo Agentes	73
6.2.2	Modelos de Processo das Capacidades de Auto-Otimização e Auto-Configuração	75
6.3	Etapa 2: Simulação	79
6.4	Etapa 3: Análise	82
6.4.1	Métricas Globais	82
6.4.2	Métricas Específicas	83
6.4.3	Métricas Customizadas	85
6.5	Etapa 4: Resultados	85
6.5.1	Capacidades de Auto-Otimização e Auto-Configuração Aplicadas ao Protocolo OLSR	85
6.5.2	Capacidades de Auto-Otimização e Auto-Configuração Aplicado ao Protocolo AODV	92
6.5.3	Comparação entre OLSR:Auto-Organizado e AODV:Auto-Organizado	99
6.6	Algumas Considerações	101
7	Conclusões e Trabalhos Futuros	103
7.1	Limitações	105
7.2	Trabalhos Futuros	105

7.3	Considerações Finais	106
Referências		107
Apêndice A – Publicações		116
A.1	A Self-X Approach for OLSR Routing Protocol in Large-Scale Wireless Mesh Networks	116
A.2	A Performance Evaluation of OLSR and AODV Routing Protocols Using a Self-Configuration Mechanism for Heterogeneous Wireless Mesh Networks	116
A.3	A Self-Configuration Property for OLSR Routing Protocol in Wireless Mesh Networks	117
A.4	An Architectural Self-Organization Model for Large-Scale Wireless Mesh Networks Based in Technology of Agents	117
A.5	Towards an Architecture for Self-Organization in Wireless Mesh Networks	117
A.6	Uma Arquitetura para Auto-Organização em Redes em Malha Sem Fio	118
Apêndice B – MatLab		119
B.1	Código do Processo de Auto-Otimização para Matlab	119

Lista de Figuras

2.1	(a) Topologia em malha regular. (b) Topologia em malha irregular . . .	10
2.2	Comparação de topologias sem fio (<i>Peer-to-Peer, Infrastructure, Ad-Hoc e Wireless Mesh</i>) adaptado de (HELD, 2005)	11
2.3	Exemplo de um conexão cliente-para-cliente em uma RMSF adaptado de (AKYLDIZ et al., 2005)	14
2.4	Exemplo de um <i>backbone</i> híbrido para RMSF adaptado de (AKYLDIZ et al., 2005)	15
2.5	Mapa da cidade de Cambridge onde está localizado o projeto Roofnet (AGUAYO et al., 2004)	21
2.6	Modelo da arquitetura da rede Mesh da Universidade de Thessaly (TSARMPOPOULOS et al., 2005)	22
2.7	Modelo da rede do projeto Re-Mesh indoor e outdoor (PASSOS et al., 2006)	24
2.8	Arquitetura do projeto Canopy Hot Zone (MOTOROLA, 2007a) . . .	25
2.9	Plataforma de <i>Hardware</i> do <i>Mesh Cube</i> (4G-SYSTEMS, 2007)	25
3.1	Comparação de uma abordagem de rede auto-organizada com uma abordagem de rede convencional centralizada. Adaptada de (PREHOFER; BETTSTETTER, 2005)	31
3.2	Comparação entre escalabilidade versus determinismo entre controle centralizado e sistema auto-Organizado (DRESSLER, 2006)	32
3.3	Controle realimentado padrão (HERRMANN et al., 2005)	35
3.4	Exemplo de políticas baseadas em ações (KEPHART; WALSH, 2004) . .	36
3.5	Controle realimentado proposto por (KEPHART; CHESS, 2003)	37
3.6	Comparativo entre computação atual e auto-organizada (autonômica), conforme as Capacidades de Auto-x adaptado de (KEPHART; CHESS, 2003)	39
3.7	Modelo conceitual da arquitetura do projeto ANA (ANA, 2006)	43
3.8	Arquitetura de um nodo do projeto Hagggle (HAGGLE, 2006)	44
3.9	Visão geral da arquitetura do projeto CASCADAS (MANZALINI; ZAMBONELLI, 2006)	45

4.1	Exemplo de difusão de mensagens, utilizando a técnica de MPR's no protocolo OLSR	50
4.2	Exemplo da descoberta de rota e propagação inversa do protocolo AODV (ROYER; TOH, 1999)	51
5.1	Visão geral da arquitetura proposta, conforme modelo em camadas TCP/IP	60
5.2	Visão geral do modelo de auto-organização proposto, adaptado às redes em malha sem fio	61
5.3	Visão da arquitetura para auto-organização para um nodo roteador em malha	62
6.1	Etapas do processo de modelagem e simulação utilizado para validar a arquitetura proposta no simulador <i>Opnet</i>	72
6.2	Modelo da plataforma de agentes acoplado ao nodo roteador em malha	74
6.3	Estrutura do pacote utilizado para descoberta do tamanho da rede pelo NS-Agents	75
6.4	Exemplo da análise das escalas do parâmetro Hello-Interval conforme a métrica de vazão e a variação do intervalo de tempo	76
6.5	Exemplo do cenário intra-campus utilizado para simulação dos nodos dispostos em uma área de $2000 \times 2000 m^2$	80
6.6	Configuração dos parâmetros da interface sem fio de um nodo roteador	81
6.7	Configuração dos parâmetros da interface sem fio de um nodo cliente	81
6.8	Comparação do total de mensagens HELLO enviadas do protocolo OLSR:Auto-Organizado e OLSR:Padrão	86
6.9	Relação do tráfego de pacotes recebidos entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão	86
6.10	Relação do tráfego de pacotes enviados entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão	87
6.11	Comparação do tráfego de mensagens TC (<i>Topology Control</i>) enviadas do protocolo OLSR:Auto-Organizado e OLSR:Padrão	88
6.12	Comparação do número de mensagens MPR entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão	88
6.13	Comparação da quantidade de pacotes descartados do protocolo OLSR:Auto-Organizado e OLSR:Padrão	89
6.14	Relação do atraso fim-a-fim entre OLSR:Auto-Organizado e OLSR:Padrão	89
6.15	Comparação do atraso de acesso ao meio entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão	90

6.16	Relação da carga da rede entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão	91
6.17	Comparação da Vazão entre o Protocolo OLSR:Auto-Organizado x OLSR:Padrão	91
6.18	Comparação do número de saltos por rota entre o AODV:Auto-Organizado e AODV:Padrão	92
6.19	Comparação do total de réplicas de roteamento enviadas entre AODV:Auto-Organizado e AODV:Padrão	93
6.20	Comparação do tráfego de pacotes recebidos entre AODV:Auto-Organizado e AODV:Padrão	93
6.21	Comparação do tráfego de pacotes enviados entre AODV:Auto-Organizado e AODV:Padrão	94
6.22	Comparação do número de pacotes descartados do protocolo AODV:Auto-Organizado e AODV:Padrão	95
6.23	Comparação do total de requisições de rotas enviadas pelo AODV:Auto-Organizado e AODV Padrão	95
6.24	Comparação do atraso do protocolo AODV:Auto-Organizado e AODV:Padrão	96
6.25	Relação do total de erros de rotas entre AODV:Auto-Organizado e AODV:Padrão	96
6.26	Comparação do atraso de acesso ao meio do Protocolo AODV:Auto-Organizado e AODV:Padrão	97
6.27	Comparação da carga da rede entre AODV:Auto-Organizado e AODV Padrão	97
6.28	Comparação do tempo de descoberta de rota entre AODV:Auto-Organizado e AODV:Padrão	98
6.29	Comparação da vazão do protocolo AODV:Auto-Organizado e AODV:Padrão	99
6.30	Relação da vazão entre OLSR:Auto-Organizado e AODV:Auto-Organizado	100
6.31	Relação do atraso de descoberta de rota entre OLSR:Auto-Organizado e AODV:Auto-Organizado	100
6.32	Relação da sobrecarga entre OLSR:Auto-Organizado e AODV:Auto-Organizado	101

Lista de Tabelas

2.1	Principais Características que Diferem as Redes <i>ad hoc</i> das RMSF . .	12
6.1	Exemplo dos dados captados do parâmetro Hello-Interval usando a vazão como métrica de análise	77
6.2	Descrição dos Parâmetros do protocolo OLSR depois do processo de auto-otimização	78
6.3	Descrição dos Parâmetros do protocolo AODV depois do processo de auto-otimização	78
6.4	Descrição dos parâmetros de simulação utilizados no cenário intra-campus	81

Lista de Abreviaturas

AODV	<i>Ad Hoc On-Demand Distance Vector Routing</i>
CGSR	<i>Cluster-Head Gateway Switch Routing</i>
DHCP	<i>Dynamic Host Configuration Protocol</i>
DSDV	<i>Destination Sequenced Distance Vector</i>
DSR	<i>Dynamic Source Routing</i>
EMA	<i>External Module Application</i>
ETT	<i>Estimated Transmission Time</i>
ETX	<i>Estimated Transmission Count</i>
FSM	<i>Finite State Machine</i>
FSR	<i>Fisheye State Routing</i>
IBM	<i>International Business Machines</i>
IP	<i>Internet Protocol</i>
LAN	<i>Local Area Network</i>
MANET	<i>Mobile Ad Hoc Network</i>
MIMO	<i>Multiple Input Multiple Output</i>
MIT	<i>Massachusetts Institute of Technology</i>
MMRP	<i>Mobile Mesh Routing Protocol</i>
MPR	<i>multipoints relay</i>
NIC	<i>Network Interface Card</i>
OLSR	<i>Optimized Link State Routing</i>
OPNET	<i>Optimized Network Engineering Tools</i>
PDA	<i>Personal Digital Assistant</i>
REMESH	<i>Rede Mesh de Acesso Universitário Faixa Larga Sem Fio</i>
RMSF	<i>Redes em Malha Sem Fio</i>
RREQ	<i>Route Request</i>
TCP	<i>Transport Control Protocol</i>
TORA	<i>Temporally-Ordered Routing Algorithm</i>
TTL	<i>Time to live</i>
WiFi	<i>Wireless Fidelity</i>
WIMAX	<i>Wireless Interoperability for Microwave Access</i>
WLAN	<i>Wireless Local Area Network</i>
WRP	<i>Wireless Router Application Plataform</i>
WRP	<i>Wireless Routing Protocol</i>

Resumo

Redes em Malha Sem Fio (RMSF) ou *Wireless Mesh Networks*(WMN) são redes basicamente formadas por nodos roteadores e clientes. Os nodos roteadores têm a função de formar a malha de comunicação da rede, enquanto os clientes usam essa comunicação, por exemplo, para acesso à Internet. A crescente complexidade em gerenciar essas redes e a grande heterogeneidade de equipamentos envolvidos na mesma dificultam o trabalho do administrador em configurar e manter a rede funcionando de forma satisfatória. Ainda, a inclusão de equipamentos móveis e a combinação de diferentes tipos de redes suportadas, tais como: redes de telefonia celular, redes sem fio WLAN e redes do tipo Ad Hoc móveis, tornam a administração manual de cada nodo um processo dispendioso, caro e difícil de gerenciar. Sendo este processo difícil de ser gerenciado, ele consome um tempo extra do administrador, podendo ocorrer falhas e também problemas de segurança (intrusos e roubo de informações).

Uma alternativa para minimizar estes problemas é a utilização do conceito de auto-organização aliado às capacidades denominadas auto-x (configuração, otimização, cura e proteção), da rede. Nesta dissertação é proposta uma abordagem para a auto-organização em redes em malha sem fio, utilizando como base para aquisição de informações da rede a tecnologia de agentes de software, juntamente com a construção de módulos para auto-configuração, auto-otimização, auto-cura e auto-proteção. Como primeira experiência enfatizou-se o desempenho da rede, implementando as capacidades de auto-otimização e auto-configuração nos protocolos de roteamento OLSR e AODV.

Através de simulações pôde ser demonstrado que as capacidades implementadas nos protocolos supracitados resultam em uma melhora na vazão e diminuem a sobrecarga de mensagens de controle e manutenção de rotas provocada pela excessiva injeção dessas mensagens na rede. Por fim, foi observado através das métricas de vazão, atraso e sobrecarga, que os protocolos OLSR:Auto-Organizado e AODV:Auto-Organizado obtêm melhor desempenho quando as escalas da rede em malha sem fio passam de médio para grande porte.

Palavras-Chave: Auto-Configuração e Otimização, Auto-Organização, Redes em Malha Sem Fio, Agentes de Software, Roteamento em Redes Sem Fio.

Abstract

Wireless Mesh Networks(WMN) are networks basically formed by router and client nodes. The router nodes have the function to form the mesh of network's communication, while the clients use this communication to access, for example, the Internet. The growing complexity to manage these networks and the great heterogeneity of equipments involved, make hard the administrator's work to configure and to maintain the network working in a satisfactory way. Still, the inclusion of mobile equipments and the combination of different types of supported networks, such as: cellular telephony networks, Wireless WLAN Ad Hoc, networks make the manual administration of each node a costly process, expensive and hard to manage. Being this process hard to be managed, it consumes an extra administrator's time, may happen faults and also security problems (intruders and unsafe information).

An alternative to minimize these problems is the use of self-organization concept together with the capacities denominated self-x (configuration, optimization, cure and protection) of the network. In this dissertation is proposed an approach to self-organization in wireless mesh networks using as base for acquisition of network's information the technology of software agents, jointly with the implementation of modules for self-configuration, self-optimization, self-cure and self-protection.

As first experience, was emphasized the network's performance implementing the self-optimization and self-configuration capacities in the routing protocols OLSR and AODV. Through simulations it could be shown that the implemented capacities in the OLSR and AODV protocols, results in the improvement of throughput and decrease of control messages overhead and maintenance of routes provoked by the excessive injection of messages in the network. Finally, it was observed through the throughput, delay and overhead metrics, that the OLSR:Self-Organized and AODV:Self-Organized protocols obtain better performance when the wireless mesh network scales range from medium to large.

Keywords: *Self-Optimization and Self-Configuration, Self-Organization, Wireless Mesh Networks, Software Agents, Wireless Routing*

1 *Introdução*

Nos últimos anos, as redes em malha sem fio (RMSF) ou *Wireless Mesh Networks* (WMN) têm atraído considerável atenção da indústria (MICROSOFT, 2005) (MOTOROLA, 2005) e da comunidade científica devido à gama de aplicações que podem ser desenvolvidas com base nessas redes, principalmente em ambientes metropolitanos. De acordo com Zhang (2006), uma rede em malha sem fio é qualquer rede sem fio que tenha uma topologia parcial ou completamente ligada. Particularmente, redes em malha sem fio são caracterizadas por nodos sem fio estáticos que provêem uma infra-estrutura distribuída para os nodos clientes sobre uma topologia em malha parcial. Redes em malha sem fio comunicam-se através de múltiplos saltos e consistem basicamente de nodos-clientes, que suportam mobilidade, ou não, e nodos-roteadores. Ambos os nodos possuem interfaces sem fio para comunicação entre eles e, no caso específico dos nodos-roteadores, esses são responsáveis por formar a malha da rede de uma forma fixa, autônoma ou pré-determinada. Os nodos-roteadores podem trabalhar como *gateways* ou *bridges*, permitindo a interconexão de diferentes tipos de redes (AKYLDIZ et al., 2005). Os nodos-roteadores são usualmente colocados em alturas elevadas, de forma a evitar interferências físicas na área de cobertura de suas antenas. Redes em malha sem fio é uma tecnologia emergente e possui vantagens em relação a maximizar sua área de cobertura, devido à possibilidade de introdução de nodos-roteadores com função de *gateways* conectados à rede cabeada, a fim de estender as aplicações para a malha sem fio e também ao seu baixo custo de implementação, pois pode utilizar tecnologias sem fio disponíveis no mercado para sua construção.

As RMSF são comumente aplicadas em ambientes metropolitanos e consistem de uma grande quantidade de nodos. As principais vantagens das redes em malha sem fio são: a tolerância a falhas, a simplicidade de configuração e a capacidade para largura de banda (HELD, 2005). Comparando-se as RMSF com as redes de telefonia celular, esta última, em casos de falhas, podem deixar uma área muito grande não comunicável, se for utilizada uma única Estação Base (EB). Redes em malha sem fio

provêem alta tolerância a falhas mesmo que uma quantidade de nodos falhe. Apesar da similaridade com as redes *ad hoc*, os protocolos e arquiteturas desenvolvidos para este tipo de rede não são bem aplicados nas redes em malha sem fio. Por exemplo, se a rede em malha sem fio utilizar múltiplas interfaces de rádio, o protocolo de roteamento inicialmente projetado para redes *ad hoc* móveis deve ser adaptado para funcionar com múltiplos-rádios, como também as métricas de roteamento devem ser melhoradas.

No caso específico das redes em malha sem fio são desejáveis protocolos de roteamento que minimizem a sobrecarga de mensagens e que evitem um acúmulo de mensagens de descoberta de caminhos sobre os nodos na rede. A maioria dos protocolos para redes em malha sem fio são derivados das redes *ad hoc* e necessitam adequar-se às características das redes em malha sem fio. Nas redes *ad hoc* móveis há uma grande mobilidade entre os nodos, mas no caso das rede em malha sem fio essa mobilidade é limitada, ou seja, os nodos roteadores em malha são pré fixados. Um protocolo de roteamento especificamente projetado para redes *ad hoc* móveis não conseguirá obter o máximo de performance na rede em malha sem fio. Devido a essa limitação, há uma crescente necessidade de novos protocolos de roteamento, ou adaptações dos protocolos já existentes para redes *ad hoc* móveis, para obter maior performance em redes em malha sem fio (PASSOS et al., 2006)(ABELÉM et al., 2007)(LE et al., 2007)(PIRZADA; PORTMANN, 2007)(BAUMANN et al., 2007). Além das limitações dos protocolos de roteamento, outra característica desejável em redes em malha sem fio é a capacidade de auto-organização (*self-organization*), ou seja, forma de organizar a rede sem intervenção manual ou externa do administrador (VASUDEVAN, 2006).

1.1 Objetivo Geral

Propor uma arquitetura para auto-organização em redes em malha sem fio, utilizando como base para aquisição de informações da rede, a tecnologia de agentes de software (WEISS, 1999) juntamente com os módulos oriundos da computação autônoma conhecidos como auto-x: auto-otimização (*self-optimization*), auto-configuração (*self-configuration*), auto-cura (*self-healing*) e auto-proteção (*self-protection*) da rede. Os módulos desenvolvidos nesta dissertação limitam-se aos dois primeiros: auto-otimização e auto-configuração, pois os demais são propostos como trabalhos futuros. Como primeira experiência adaptou-se a arquitetura proposta nos protocolos de roteamento OLSR e AODV. Com a finalidade de validar a abordagem proposta

foram executadas exaustivas simulações em um cenário de uma rede em malha sem fio intra-campus utilizando-se o simulador de redes Opnet (*Optimized Network Engineering Tools*) (OPNET, 1987a) (OPNET, 1987b), nos quais foram comparadas características originais dos protocolos de roteamento citados e posteriormente aplicadas as capacidades de auto-otimização e configuração propostas.

1.2 Trabalhos Relacionados

Este trabalho foi iniciado através de estudos sobre redes em malha sem fio e auto-organização encontrados na literatura e, nesta fase de estudos, alguns trabalhos se destacaram como apoio para a pesquisa apresentada nesta dissertação. Entretanto, observou-se a necessidade de subdividir os trabalhos relacionados em: arquiteturas auto-organizadas e abordagens utilizando os conceitos de auto-configuração e auto-otimização aplicados ao roteamento em redes sem fio.

1.2.1 Arquiteturas Auto-Organizadas

Devido ao crescimento em número de dispositivos e serviços que fazem parte das redes sem fio atuais, alguns esforços estão sendo feitos para que o processo de instalação, gerenciamento e manutenção dessas redes não se tornem uma dor de cabeça para o administrador da mesma. Algumas pesquisas no sentido de que a rede possua autonomia em suas decisões estão sendo desenvolvidas.

A capacidade de auto-organização é útil ou mesmo necessária em função da quantidade de nodos na rede, pois a configuração manual de cada nodo dentro dela se torna um problema complexo, conforme a quantidade de nodos que entram e saem. Neste caso a capacidade de auto-organização permite que a rede não perca sua autonomia, por causa da mínima intervenção manual do administrador para configuração de cada nodo-roteador. Uma outra vantagem de redes auto-organizadas é a redução do custo de instalação e manutenção dos nodos envolvidos na rede, uma vez que os nodos-clientes e nodos-roteadores entram na rede de uma forma transparente para os usuários. No entanto, as abordagens atuais para redes em malha sem fio permitem realizar apenas parcialmente o objetivo da auto-organização (AKYLDIZ et al., 2005). Isso se deve ao fato de que as abordagens encontradas na literatura não se preocupam na implementação da forma completa de auto-organização da rede, não contemplando assim todos os aspectos como desempenho, tolerância a falhas e segurança. Devido à complexa natureza das redes em malha

sem fio, novos paradigmas são necessários para projeto, gerenciamento e manutenção (SARMA et al., 2006).

Em Kephart e Chess (2003), os autores apresentam um projeto de computação autônoma que demonstra que a capacidade de auto-gerenciamento pode ser alcançada de maneira evolutiva e prescrevem as iniciativas para aplicação dessa capacidade em ambientes heterogêneos. A complexidade em configurar as políticas de gerenciamento é uma desvantagem eminente nesta abordagem.

Algumas arquiteturas são propostas para redes em malha sem fio. Em Raniwala e Chiueh (2005) apresenta-se uma arquitetura multi-canais chamada *Hyacinth*. Esta arquitetura é composta por nodos sem fio equipados com múltiplas interfaces de rede (NIC's) no padrão 802.11. Os autores demonstram um modelo inteligente para escolha dos melhores canais utilizando múltiplos rádios. O alto custo de implantação dessa abordagem em larga escala inviabiliza o projeto, devido à necessidade de *hardware* dedicado para agregar o tráfego da rede, além de não enfatizar a auto-organização em nível de camada de rede.

Wu (2006) considera que as redes em malha sem fio são boas candidatas a resolver o problema da última milha ¹ para serviços de banda larga sem fio. As redes em malha sem fio se deparam com o problema da escalabilidade devido às características de auto-organização e conexões através de múltiplos saltos. Para este contexto o autor propõe uma arquitetura hierárquica para redes em malha sem fio. Esta abordagem consiste em dividir a rede como um todo dentro de pequenas sub-redes, as quais são operadas independentemente e são conectadas através de um *backbone* confiável. Um problema encontrado em abordagens utilizando arquiteturas hierárquicas é que o *cluster head* (nodo eleito para ser o nodo principal dentro de um subgrupo de redes) se torna um ponto único de falha, além de necessitar políticas bem definidas para eleição de um novo nodo principal em caso de falhas. Neste caso, abordagens utilizando arquiteturas híbridas são mais tolerantes a falhas.

Malheiros e Madeira (2006) propõem uma arquitetura para redes em malha sem fio auto-gerenciáveis, utilizando o paradigma de computação autônoma. As propriedades de auto-configuração, auto-otimização, auto-cura e auto-proteção são propostas e adaptadas às redes em malha sem fio. Os autores esperam contribuir na evolução da abordagem de sistemas autônomos voltados às redes em malha sem fio, além de aprimorar as técnicas de gerência e desenvolvimento de redes em malha sem fio auto-gerenciáveis. Entretanto, os autores estão ainda em fase de estudos e

¹Última milha é considerado um problema de área de cobertura, ou seja, é necessária uma cobertura máxima do sinal da rede sem fio.

implementação da arquitetura, não havendo resultados concretos para avaliação e comparação.

Em Bash et al. (2007), os autores apresentam um projeto denominado *BaselMesh* que tem por objetivo integrar o estilo de roteamento utilizado em redes *ad hoc* móveis e adaptá-lo em uma rede em malha sem fio, levando em consideração diferenças de roteamento entre os dois tipos de rede.

1.2.2 Abordagens para Auto-Otimização e Auto-Configuração dos Parâmetros de Roteamento em Redes Sem Fio

Em Malatras et al. (2005), os autores apresentam um sistema denominado SCOMAN (Self-Configuring and Optimizing MANET's), que implementa auto-configuração e auto-otimização para as redes *ad hoc* móveis (MANET). Neste trabalho é proposto um gerenciamento colaborativo para MANET's baseado em políticas de gerenciamento pré-definidas e implementado utilizando os princípios de *context awareness* e *cross layer design*. Observa-se que a abordagem proposta é direcionada às redes *ad hoc* móveis, enquanto que, se adaptada às redes em malhas sem fio, deve-se levar em consideração as diferenças encontradas entre as redes.

Nos trabalhos Huang et al. (2008b), Huang et al. (2006) e Huang et al. (2008a) os autores analisam os diferentes impactos do ajuste do tempo de *refresh interval* do protocolo OLSR em diferentes cenários simulados, variando-se a densidade e a velocidade dos nodos. Os autores enfatizam que através de simulação pode-se comprovar que a performance do OLSR é mais sensível para o ajuste do *HELLO_Interval* do que para o *TC_Interval*, ou seja, o protocolo OLSR é diretamente influenciado pelo ajuste do tempo de *HELLO_Interval*, pois a cada decremento do intervalo das mensagens HELLO, os nodos teriam a capacidade de descoberta mais rápida de quebra de um enlace. Nestes trabalhos os autores propõem uma solução para diminuir o intervalo de envio das mensagens de HELLO para acelerar o processo de descoberta de quebra de enlaces entre dois nodos na rede. Mas em contrapartida, essa diminuição do intervalo de tempo injetaria mais mensagens na rede, provocando uma sobrecarga na mesma.

Já em outro trabalho os autores Gomez et al. (2005) analisam como o protocolo de roteamento OLSR encontra caminhos alternativos, quando um nodo destino falha. Os autores propõem um RCL (*Route Change Latency*) que depende de como os parâmetros são ajustados. Como resultados observa-se que usando altas taxas de envio de mensagens HELLO, o protocolo OLSR tem uma melhora significativa do

atraso fim-a-fim na rede. Mas segue o mesmo problema dos trabalhos dos autores Huang et al. (2008b), Huang et al. (2006), Huang et al. (2008a), que diminuindo o tempo de envio das mensagens de HELLO, mais mensagens em curtos espaços de tempos seriam injetadas na rede obtendo assim uma sobrecarga de mensagens.

(MALATRAS et al., 2005) (HUANG et al., 2006), (HUANG et al., 2008a), (HUANG et al., 2008b) e (GOMEZ et al., 2005) apresenta abordagens que ajustam automaticamente os parâmetros dos protocolos de roteamento são direcionadas às redes MANET's. Com isso é justificada a carência de métodos e técnicas para o ajuste automático dos parâmetros para redes em malha sem fio.

1.2.3 Considerações

Na literatura pesquisada encontram-se poucas abordagens relacionadas ao projeto de pesquisa proposto nesta dissertação. Foram analisados trabalhos relacionados e observou-se que somente algumas soluções wu:2006, (RANIWALA; CHIUH, 2005) (SARMA et al., 2006) (MALHEIROS; MADEIRA, 2006) enfatizam o problema da auto-organização em redes em malha sem fio. A maioria das abordagens de auto-organização propostas não levam em consideração as peculiaridades das redes em malha sem fio em comparação com as redes *ad hoc* móveis, tais como: pouca mobilidade dos nodos roteadores, o tráfego para a internet é originado e retornado aos nodos *gateways*, pouco gasto de energia, infra-estrutura parcial ou completamente ligada, necessidade de planejamento para sua construção, roteamento parcial ou completamente distribuído, tráfego típico do usuário ou tráfegos adicionais (nodos sensores), dentre outras. Com isso, nota-se que as redes em malhas sem fio não abstraem um desempenho satisfatório, se forem simplesmente acoplados, por exemplo, protocolos de roteamento originalmente propostos para funcionar em redes do tipo *ad hoc* móveis, como é o caso do OLSR e do AODV. Deve-se levar em conta essas peculiaridades e aplicá-las ao roteamento em malha sem fio, a fim de se obter um desempenho satisfatório na mesma. Com isso justifica-se o desenvolvimento da arquitetura para auto-organização em redes em malha sem fio, com o intuito de prover como primeira experiência uma significativa melhora da performance da rede em malha sem fio, utilizando-se técnicas e conceitos oriundos da computação autônoma e auto-organizada. Sumarizando, ressalta-se que a auto-organização é um conceito interessante para as próximas gerações de redes sem fio e ainda, percebe-se muitas lacunas que devem ser preenchidas em direção à auto-organização completa de uma rede em malha sem fio.

1.3 Contribuições

O principal contribuição deste trabalho é a definição de uma arquitetura para auto-organização em redes em malha sem fio adaptada aos protocolos de roteamento OLSR e AODV, mas não limitada aos mesmos. A metodologia aplicada para desenvolver este trabalho integra agentes de software para monitoramento e controle do tamanho da rede, juntamente com as capacidades de auto-otimização e auto-configuração, aplicadas no controle dos parâmetros dos protocolos de roteamento supracitados.

Como contribuição secundária mostra-se a melhora no desempenho dos protocolos OLSR e AODV na rede, implementando a interação entre os agentes de controle de tamanho da rede NS_AGENTS e as capacidades de auto-otimização e auto-configuração nesses protocolos.

A terceira contribuição deste trabalho se aplica na pesquisa bibliográfica sobre redes em malha sem fio e auto-organização, pois envolve conceitos atuais que possuem pouco material em língua portuguesa.

1.4 Apresentação do Trabalho

Esta dissertação está subdividida em seis capítulos, os quais são descritos a seguir:

No Capítulo 2 é apresentada uma visão geral sobre redes em malha sem fio, destacando definições, diferenças entre as RMSF e as redes *ad hoc*, infra-estrutura da rede, características, cenários de aplicação, fatores que afetam a performance, padronização e projetos.

O Capítulo 3 é destinado a apresentar o conceito de auto-organização de acordo com diferentes pontos de vista, mostrando as definições, propriedade dos sistemas auto-organizados, abordagens *top-down* e *bottom-up* e enfatizando conceitos relacionados a computação autônoma, redes autônomas e alguns projetos.

No Capítulo 4 é apresentada, de forma geral, a classificação dos protocolos de roteamento para redes em malha sem fio, enfatizando os protocolos de roteamento OLSR e AODV, como sendo a base do modelo de auto-organização proposto nesta dissertação.

O Capítulo 5 detalha a arquitetura de auto-organização proposta, ressaltando a

arquitetura em nível conceitual juntamente com toda a interação entre a plataforma dos agentes, os módulos de auto-organização e a extensão nos protocolos de roteamento OLSR e AODV.

No Capítulo 6 são apresentados a modelagem e a simulação parcial da arquitetura proposta, dispondo os resultados experimentais obtidos através de simulações em cenários específicos projetados especialmente para validar a presente proposta.

No Capítulo 7 são abordadas as devidas conclusões e discutidos os benefícios em se utilizar a arquitetura proposta juntamente com a descrição dos trabalhos futuros, os quais dariam continuidade ao projeto de pesquisa proposto nessa dissertação.

2 *Redes em Malha Sem Fio*

Desde o surgimento do padrão IEEE 802.11 (802.11, 1997), em meados do ano de 1997, observa-se um aumento da utilização das redes sem fio, principalmente em locais de acesso público como: aeroportos, shoppings, restaurantes, hospitais, escolas e universidades. Esse aumento deveu-se ao barateamento dos equipamentos utilizados e ao amadurecimento dos padrões de comunicação. As redes sem fio baseadas no padrão 802.11 possuem duas formas de comunicação: modo infra-estruturado, no qual um conjunto de clientes se comunica por meio de um ponto de acesso; e modo *ad hoc*, no qual os nodos se comunicam diretamente, sem a necessidade de um ponto de acesso central. Uma alternativa para prover acesso sem fio baseado na forma de comunicação *ad hoc* é a chamada Rede em Malha Sem Fio - RMSF ou em inglês *Wireless Mesh Network* - WMN. Com as RMSF é possível construir redes capazes de garantir uma grande área de cobertura e, possivelmente, reduzir a interferência causada pelos padrões de rádio frequência utilizados. Além dessas vantagens, as RMSF possuem outras como: a redução do custo de implementação, alta confiabilidade, capacidade de crescimento sem degradação da largura de banda e eliminação de pontos únicos de falhas e gargalos. Entretanto, os nodos pertencentes à rede devem manter a conectividade de forma automática e possuir mecanismos para encontrar a melhor rota em caso de falha de algum nodo. Neste caso é desejável também a baixa intervenção do administrador para manutenção e configuração da rede, devido ao custo de se configurar manualmente todos os nodos. Assim, observa-se que as RMSF podem ser utilizadas em projetos de redes faixa larga sem fio.

Neste capítulo são abordados, de forma geral, a definição de RMSF juntamente com um comparativo entre as RMSF e as redes *ad hoc*. É apresentada também a infra-estrutura da rede e algumas características. São observados exemplos de cenários de aplicação e alguns fatores que afetam a sua performance, juntamente com os projetos relevantes da área.

2.1 Definição

Para entender como as RMSF são interligadas faz-se necessária uma breve contextualização sobre a topologia em malha. Um dos pontos chave para a constituição de uma rede de computadores é a topologia da rede, isto é, a interligação dos nodos na rede que são denominadas topologia da rede. Uma das formas de interligar nodos na rede é através de topologias ponto-a-ponto. Nas topologias ponto-a-ponto, a rede é composta de diversos enlaces de comunicação, formando um par entre os nodos envolvidos na rede. Dentre as topologias ponto-a-ponto, pode-se citar: estrela, anel, árvore, malha regular e malha irregular (TANENBAUM, 1996).

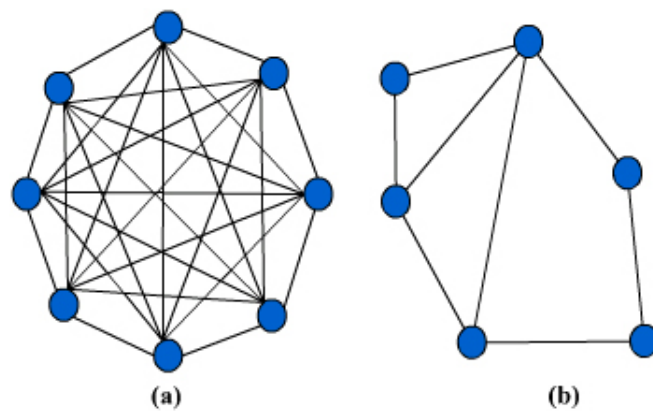


Figura 2.1: (a) Topologia em malha regular. (b) Topologia em malha irregular

Na topologia em malha regular, como pode ser observado na Figura 2.1 (a), todos os nodos possuem um link de comunicação entre eles. Neste caso, a rede possui alto grau de redundância e confiabilidade. Já na topologia em malha irregular, alguns nodos possuem enlaces de comunicação e outros não, conforme se observa na Figura 2.1 (b). Ao abstrair o conceito de redes em malha originalmente proposto, para redes que utilizam cabos como enlaces de comunicação nas RMSF os enlaces de comunicação são formados por padrões de rádio frequência como 802.11a/b/g, 802.16 e 802.15.

Na Figura 2.2, Held (2005) mostra um comparativo entre as redes sem fio ponto-a-ponto (*peer-to-peer*), infra-estruturadas (*Infrastructure*), (*Ad Hoc*) e redes em malha sem fio (*Wireless Mesh Network*). No ambiente ponto-a-ponto a transmissão é limitada a somente dois nodos; na comunicação infra-estruturada, os nodos comunicam-se através de um ponto de acesso centralizado; e no ambiente da rede em malha, cada nodo funciona como roteador e repetidor, transmitindo pacotes para o próximo nodo. Held (2005) enfatiza que agrupando-se um ambiente ponto-a-ponto pode-se formar um ambiente em malha e define as RMSF como:

“Uma série de transmissões ponto-a-ponto, onde cada nodo funciona tanto como um roteador ou como um repetidor.”

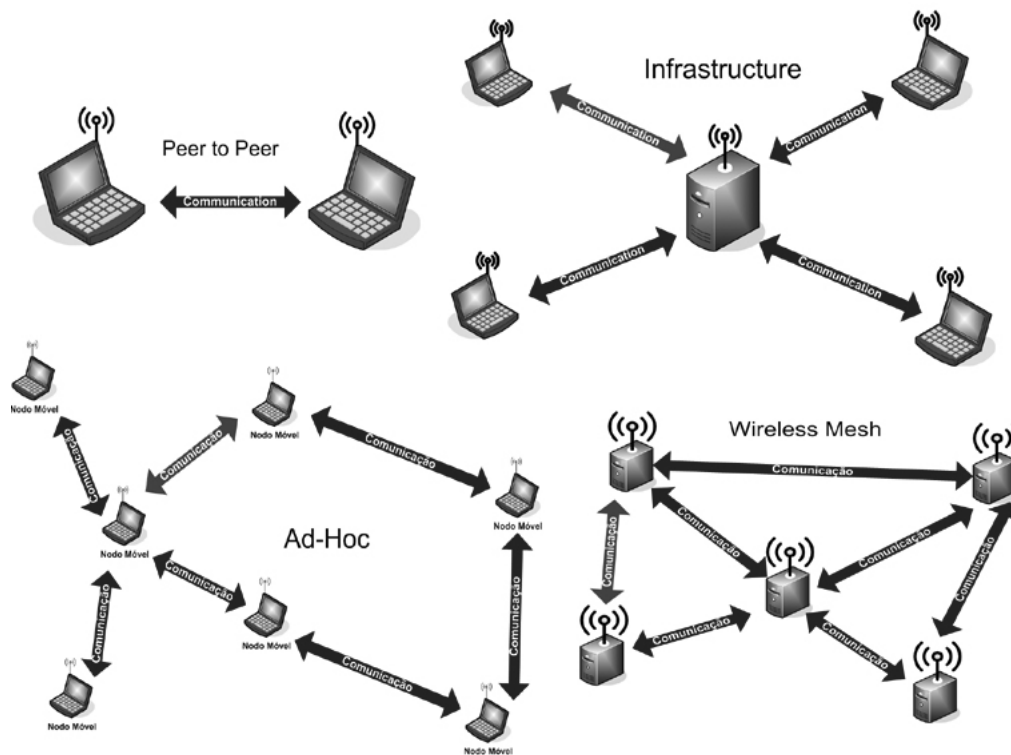


Figura 2.2: Comparação de topologias sem fio (*Peer-to-Peer*, *Infrastructure*, *Ad-Hoc* e *Wireless Mesh*) adaptado de (HELD, 2005)

Enquanto que Zhang (2006), define RMSF como:

“Tipos de redes radicais, formadas por conexões sem fio tradicionais, que marcam a divergência entre as redes de telefonia celular e as redes locais sem fio (WLAN’s). Similar ao paradigma em malha utilizado durante a década de 60 e 70 em redes que utilizavam cabos para sua comunicação”.

De forma geral, o conceito de RMSF é apresentado em Akyldiz et al. (2005):

“Redes em malha sem fio consistem basicamente de nodos clientes e roteadores, os nodos roteadores possuem mobilidade limitada e formam o backbone da rede e podem integrar funções de gateway ou bridge e permitem interconectar vários tipos de redes, como: à Internet, telefonia celular, 802.16 (WIMAX), 802.15 (bluetooth), redes sensores sem fio, dentre outras; enquanto que os nodos clientes podem ser tanto móveis ou fixos e podem formar redes entre eles mesmos ou conectar-se com a malha de roteadores.”

2.2 Diferenças entre as RMSF e as Redes *Ad Hoc* Móveis

Muitas técnicas utilizadas em redes *ad hoc* se aplicam às RMSF, mas com limitações de performance. Por serem baseadas na forma de comunicação *ad hoc*, existem diferenças que devem ser levadas em consideração na hora de projetar uma RMSF. Na Tabela 6.4 (ZHANG, 2006) são enfatizadas algumas características que se diferem quanto à topologia, mobilidade, gasto de energia, aplicação, requisitos de infra-estrutura, comunicação, performance de roteamento, desenvolvimento, características de tráfego e cenários de aplicação.

Mais diferenças podem ser observadas em Mahmud et al. (2006).

Tabela 2.1: Principais Características que Diferem as Redes *ad hoc* das RMSF

Características	Redes <i>ad hoc</i>	Rede em malha
Topologia	Altamente dinâmica	Relativamente Estática
Mobilidade dos Nodos	Média para alta	Baixa
Gasto de Energia	Alto	Baixo
Aplicação	Temporária	Semipermanente ou permanente
Requisitos de infra-estrutura	Sem infra-estrutura	Parcialmente ou completamente
Comunicação	Nodos móveis	Nodos fixos
Performance de roteamento preferencial	Completamente distribuído	Completamente distribuído ou parcialmente distribuído
Desenvolvimento	Fácil	Requer algum planejamento
Características de tráfego	Tráfego típico do usuário	Tráfego típico e pode haver tráfego de sensores
Cenários de aplicação	Militares, resgates	Militar, resgates redes comunitárias etc.

A premissa é que as redes em malha não são um conceito novo, pois a topologia em malha é largamente utilizada e, já no caso específico das RMSF, o que difere são os links de comunicação e tendo como base a já conhecida infra-estrutura de comunicação *ad hoc*. Redes do tipo em malha têm a vantagem de ser redes de baixo custo, possuem simplicidade de configuração, capacidade para largura de banda (HELD, 2005) e são tolerantes a falhas, ou seja, caso algum nodo roteador deixe de funcionar momentaneamente, os demais nodos roteadores que estão funcionando podem estender sua força de sinal, a fim de cobrir aquela área que foi debilitada pela falha do roteador. Seguindo essa premissa, as RMSF provêm alta tolerância a falhas mesmo quando uma quantidade de nodos não se encontram operantes e comunicam-se através de múltiplos-saltos com intuito de encaminhar as mensagens da origem aos seus destinos. Esse processo é similarmente encontrado nas redes do tipo *ad hoc*. Apesar da similaridade entre as redes *ad hoc*, os protocolos de roteamento e arquiteturas desenvolvidos para essas redes não são bem aplicados nas RMSF. Por exemplo, em redes *ad hoc* há uma grande mobilidade entre os nodos pertencendo

centes a redes, mas no caso das RMSF essa mobilidade é limitada. Se um protocolo especificamente projetado para uma rede *ad hoc* não conseguir obter a máxima performance na RMSF, portanto nota-se a necessidade de abordagem específicas para trabalharem em RMSF. As RMSF possuem mais recursos se comparadas às redes *ad hoc*. Em algumas aplicações para RMSF, a rede pode ter uma topologia específica, então os protocolos e algoritmos devem ser projetados para beneficiar-se da mesma.

A topologia da RMSF, por ser parcialmente estática e formada por nodos roteadores que são fixos e possuem conexão direta com uma fonte de energia, faz com que a preocupação com desgaste de energia em RMSF seja descartada. De acordo com Akyldiz et al. (2005), as RMSF surgiram para suprir as limitações encontradas nas redes *ad hoc*, no que tange a interferências entre os nodos e a área de cobertura. Outra diferença, além do roteamento, seria no padrão de tráfego da rede. Nas redes RMSF todo o tráfego origina-se ou destina-se ao nodo *gateway*, que pode ser responsável por prover acesso à Internet aos demais nodos sem fio pertencentes à rede.

2.3 Infra-Estrutura da Rede

As RMSF consistem basicamente de dois tipos de nodos: clientes e roteadores. Os nodos roteadores são responsáveis por formar a malha e integram funções específicas para rotear e interconectar diferentes tipos de redes. Podem ser equipamentos dedicados a este fim (MOTOROLA, 2007b)(4G-SYSTEMS, 2007)(LINKSYS, 2007), ou podem ser construídos utilizando computadores pessoais e softwares específicos (MESH LINUX, 2007)(QUAGGA, 2007)(CUWIN, 2007)(OPENWRT, 2005). Os roteadores comunicam-se através de múltiplos-saltos, ou seja, um nodo roteador comunica-se com o seu vizinho mais próximo, também denominado como nodo intermediário, de forma a montar um enlace de comunicação até chegar ao nodo roteador destino. Com isso é possível comunicar-se em uma grande área de cobertura com um baixo poder de transmissão, no caso do uso de antenas isotrópicas. Já os nodos clientes são menos robustos que os nodos roteadores, e em caso de falha de um roteador, clientes podem assumir a função de roteador após a execução de um algoritmo de eleição. Há uma maior variedade de nodos clientes no mercado: computadores pessoais, laptops, PDA's, pocketpc's, telefones celulares, etc. Além disso, os nodos clientes podem conectar-se aos nodos roteadores ou com outro nodo cliente, dependendo da forma de comunicação a ser implementada.

As formas de comunicação podem ser classificadas em três principais grupos e

se baseiam nas funcionalidades dos seus nodos (AKYLDIZ et al., 2005):

- **Infra-Estruturada Backbone:** este tipo de arquitetura é formada de roteadores responsáveis por constituir a malha infra-estruturada, que serve para a conexão de clientes. Pode ser construída utilizando as tecnologias de rádio frequência atualmente disponíveis no mercado, barateando assim o custo de implementação da rede. Em contrapartida, a tecnologia mais comumente utilizada é padrão 802.11, mas este padrão possui limitações conforme aumenta a quantidade de dispositivos na rede. Alguns padrões estão sendo estudados e aplicados, a fim de melhorar o desempenho da malha de roteadores. Em Zhang et al. (2006) os autores demonstram como o padrão 802.16, conhecido como WIMAX, pode ser aplicado para formar o *backbone* da RMSF de uma forma robusta e confiável.
- **Cliente-para-Cliente:** nesta arquitetura os nodos clientes comunicam-se diretamente com outros nodos clientes, sem a necessidade de utilização da malha de roteadores para intermediar a comunicação. Neste caso, os nodos clientes utilizam *hardwares* e *softwares* mais simples, devido à limitação do poder de processamento, armazenamento e energia, no caso dos PDA's, notebooks e celulares. Akyldiz et al. (2005) ainda enfatiza que os nodos clientes possuem uma única interface de comunicação para comunicação entre eles. Um exemplo de comunicação cliente-para-cliente pode ser observado na Figura 2.3 observa-se que alguns nodos são fixos e outros móveis, diferenciado-se das redes MANET's, nos quais todos os nodos são móveis.

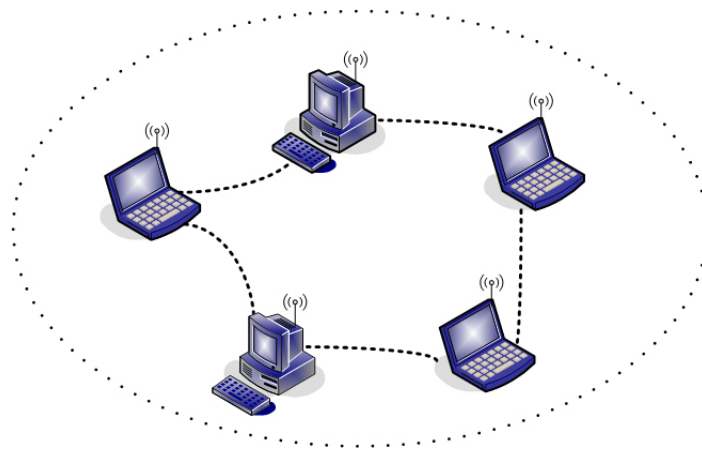


Figura 2.3: Exemplo de um conexão cliente-para-cliente em uma RMSF adaptado de (AKYLDIZ et al., 2005)

- **Híbrida:** já nas redes híbridas, há a união entre a arquitetura infra-estruturada e a cliente-para-cliente, de forma que os nodos clientes comunicam-se com os nodos roteadores, ou com outros clientes. Na Figura 2.4 pode ser observada a junção das arquiteturas.

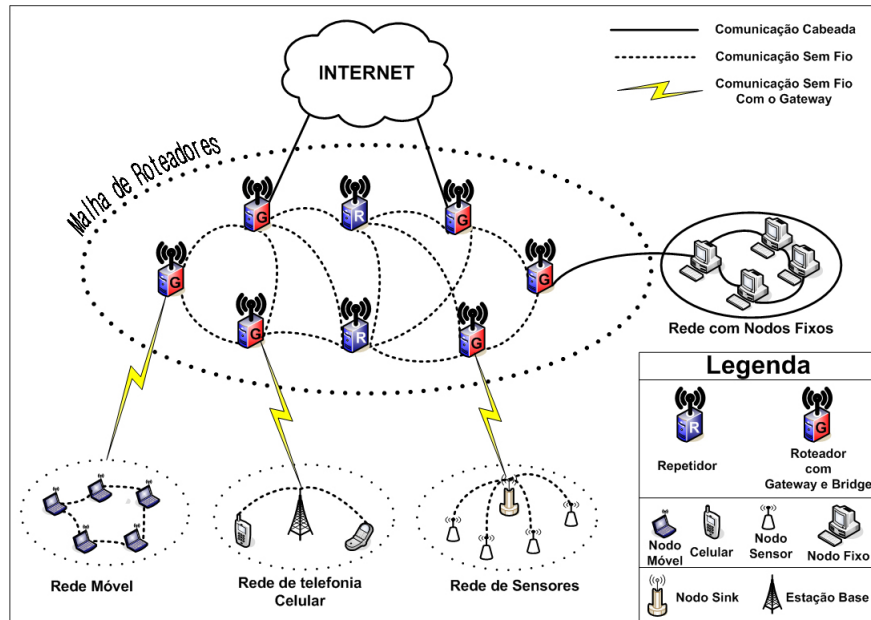


Figura 2.4: Exemplo de um *backbone* híbrido para RMSF adaptado de (AKYLDIZ et al., 2005)

A linha sólida demonstra a comunicação através de cabos, que conecta os nodos *gateways* à Internet e também à subrede de nodos fixos. Já as linhas tracejadas formam a malha de roteadores sem fio. As linhas pontilhadas mostram a comunicação direta com o nodo roteador *gateway* com as demais redes: móvel, telefonia celular, rede de sensores.

2.4 Características

Segundo Akyldiz et al. (2005), as principais características das RMSF são:

- **Múltiplos-Saltos:** as RMSF se enquadram na categoria de redes que utilizam múltiplos-saltos para a comunicação entre os nodos origem e destino. Este tipo de comunicação permite aumentar a área de cobertura da rede com baixa potência de transmissão, como é o caso da utilização de antenas isotrópicas, que utilizam o raio de cobertura do próprio nodo, dispensando assim a linha de visada. Com isso, há uma menor interferência entre os nodos e também uma melhor eficiência no reuso do sinal.

- **Suporte para Auto-Organização e Auto-Configuração:** algumas características são encontradas em RMSF como: arquitetura flexível , fácil construção e configuração, tolerância a falhas e a malha deve estar sempre conectada. Devido a essas características, as RMSF exigem menor investimento e podem crescer gradativamente, conforme a demanda de conexões.
- **Mobilidade Depende do Tipo do Nodo:** nas RMSF, os nodos roteadores possuem pouca ou nenhuma mobilidade e os nodos clientes podem ser tanto móveis ou fixos.
- **Restrições dos seus Nodos-Clientes:** geralmente, os nodos clientes que fazem parte da malha sem fio são dispositivos com baixo poder de processamento e armazenamento. Em função dessas restrições são necessárias políticas de eleições, no caso de um cliente precisar se tornar um nodo com capacidades maiores, como por exemplo, um nodo cliente se transformar em roteador.
- **Componentes *off-the-shelf*:** as RMSF podem ser constituídas utilizando equipamentos comercialmente disponíveis, como o caso do padrão 802.11.
- **Capacidade de integração com as demais redes:** a capacidade de integrar-se a diferentes tecnologias é bastante desejável em RMSF, pois atualmente o padrão mais utilizado para formar as RMSF é o 802.11 e suas variantes. Neste sentido as RMSF devem também ser compatíveis com outros tipos de redes e tecnologias como: redes de sensores sem fio, redes de telefonia celular, redes sem fio convencionais e com tecnologias como WIMAX, ZigBee, dentre outras.

2.5 Cenários de Aplicação

Devido à capacidade de comunicação com os mais variados tipos de tecnologias de rádio frequência disponíveis, diversos cenários de aplicação podem ser constituídos utilizando as RMSF:

- **Redes domésticas:** em redes domésticas, as RMSF podem substituir os pontos de acesso sem fio tradicionais por pontos de acesso em malha. Os pontos de acesso tradicionais necessitam de infra-estrutura cabeada para interligá-los dentro do ambiente, ao contrário dos pontos de acesso em malha que se interligam através de rádio frequência. Utilizando-se os pontos de acesso em

malha, é mais fácil cobrir zonas mortas; neste caso, basta posicioná-los de forma a obter um máximo de cobertura do sinal, ou simplesmente ajustar a potência de transmissão dele (AKYLDIZ et al., 2005). Neste tipo de aplicação, o ponto de acesso em malha não possui limitação de energia e nem mobilidade, pois os roteadores são conectados diretamente a uma fonte de energia e posicionados em locais pré-fixados.

- **Redes Comunitárias:** a maior gama de aplicações das redes em malha sem fio estão voltadas às redes sem fio de acesso comunitárias (MICROSOFT, 2005) (FLICKENGER, 2002), que possuem como principal objetivo fornecer acesso à Internet aos seus usuários. No Brasil, há o projeto Re-Mesh (REMESH, 2006), que consiste de nodos-roteadores instalados no topo dos prédios da universidade e se comunicam através de antenas visadas.
- **Redes Empresariais:** cenários como escritórios de pequeno porte ou até mesmo de grande porte, situados no mesmo prédio ou não, podem dispor da comunicação das RMSF (AKYLDIZ et al., 2005). Atualmente, utilizam-se pontos de acesso tradicionais que limitam a comunicação ao raio de cobertura, e a comunicação entre os demais pontos de acesso deve ser feita através de infra-estrutura cabeada ou da Internet. Neste caso inviabiliza-se o projeto, devido ao alto custo de construir a infra-estrutura cabeada para a comunicação. Com a adição das RMSF, a infra-estrutura cabeada seria eliminada e a comunicação entre os pontos de acesso em malha situados no mesmo prédio, ou não, seria bem mais acessível através de rádio frequência, não necessitando quebrar paredes, ou mesmo um projeto mais detalhado de implementação de uma rede.
- **Redes Metropolitanas / Cidades Digitais:** a extensão da área de cobertura das RMSF é uma das características desejáveis para a construção de redes metropolitanas ou, também conhecidas, como cidades digitais. Exemplos de aplicação podem ser vistos na Filadélfia e em Taipei (INFO, 2005). Na cidade de Taipei, foi implantada uma RMSF envolvendo cerca de 10 mil pontos de acesso, cobrindo cerca de 90 por cento da cidade, em uma área de 272 quilômetros quadrados. Neste projeto a idéia não se limitou somente à simples interligação de nodos clientes aos pontos de acesso, mas também em prestar serviços mais ambiciosos como serviços públicos e inclusão digital.
- **Redes Rurais:** a crescente demanda por conectividade à Internet mesmo em lugares de difícil acesso como as zonas rurais, está se tornando cada vez mais

comum. Um problema encontrado é que a tecnologia de comunicação sem fio é limitada e o custo de implementação é muito alto, tornando inviável sua aquisição. Neste caso, as RMSF podem ser uma alternativa de baixo custo para prover acesso à Internet às zonas rurais. Neste sentido algumas pesquisas (SEN; RAMAN, 2007) (DUTTA et al., 2007) estão sendo desenvolvidas em direção à melhora de performance das RMSF para redes rurais.

- **Redes de Segurança e Monitoramento:** como a área de cobertura das RMSF é extensa, elas podem suportar sistemas robustos de vigilância e monitoramento em grandes cidades. Dispositivos como sensores e câmeras sem fio são instalados em lugares estratégicos com o intuito de monitorar ações de pessoas maliciosas e, as informações coletadas e enviadas para a central de vigilância, aumentando assim a segurança do local e conseqüentemente inibindo possíveis problemas de segurança (BELAIR, 2007).
- **Sistemas de transportes:** segundo Akyldiz et al. (2005), as RMSF podem ser um atrativo em sistemas de transportes, nos quais podem prover acesso à rede nas estações e paradas e também no interior dos veículos. Neste caso, a rede não se limita somente em prover acesso à Internet aos seus passageiros. Um detalhe interessante neste tipo de rede é que, para prover acesso à Internet, é necessário um *link* de acesso rápido usado especialmente para veículos em alta velocidade. Atualmente, alguns padrões para este tipo de comunicação podem ser encontrados na literatura, como os padrões IEEE 802.20 e IEEE 802.16 (OLEXA, 2005).

2.6 Fatores que Afetam a Performance das RMSF

De acordo com Akyldiz et al. (2005) e também do ponto de vista de alguns pesquisadores, diversos são os fatores que afetam o desempenho das redes em malha sem fio. São eles:

- **Técnicas de Rádio Freqüência:** As tecnologias de rádio freqüência atualmente disponíveis ainda possuem limitações, apesar de seu rápido progresso. Muitas abordagens têm sido propostas para aumentar a capacidade e flexibilidade das redes sem fio, mas ainda é necessária uma maior maturidade em direção ao amadurecimento das técnicas de rádio freqüência. Fatores como ineficiência dos protocolos, interferência de fontes externas, compartilhamento do espectro e escassez do espectro eletromagnético, reduzem drasticamente a

capacidade de um único rádio para as RMSF. Uma alternativa para minimizar estes problemas seria a aplicação de múltiplos rádios e múltiplos canais (RANIWALA; CHIUEH, 2005) (COUTINHO, 2006) (ADYA et al., 2004) (SUBRAMANIAN et al., 2007), a fim de melhorar a capacidade das RMSF e ainda suportar a demanda de tráfego existente na mesma. Além da aplicação de múltiplos-rádios, os sistemas MIMO e WIMAX são bons candidatos a suprir as limitações encontradas no padrão 802.11 e suas variantes.

- **Degradação da Largura de Banda:** alguns pesquisadores estão direcionando suas pesquisas no sentido de estimar a capacidade de uma RMSF (GALLEGO et al., 2005). Nas RMSF, o tráfego é originado de um nodo *gateway* que atua como um gargalo. Devido à presença de gargalos, a real capacidade da rede é reduzida para $1/n$, onde n é o número de nodos associados a um *gateway* (GONZÁLEZ et al., 2007). Neste caso, há necessidade de novas técnicas para distribuir e melhorar o tráfego de pacotes, aumentando assim a capacidade da rede.
- **Latência:** cada pacote é enviado de um nodo origem para um nodo destino e conseqüentemente um atraso para este procedimento é naturalmente introduzido. Portanto, um aumento na latência e no atraso na entrega dos pacotes, pode afetar o desempenho total da rede. Neste sentido, observa-se a necessidade de técnicas para diminuir o atraso e a latência, de forma a não prejudicar o desempenho da rede.
- **Escalabilidade:** alguns problemas encontrados em redes em malha que utilizam cabos para comunicação, também existem em RMSF. O principal é a escalabilidade da rede, pois a entrada de um novo nodo implica na inserção de mais enlaces para sua comunicação, gerando um alto custo de comunicação e um baixo desempenho da rede.
- **Conectividade da Malha:** a tecnologia que garante a conectividade da malha de roteadores é o protocolo de roteamento. Nota-se que há uma crescente necessidade de projetar novos protocolos de roteamento ou adaptar os protocolos já existentes para que trabalhem em redes em malha sem fio, obtendo um melhor desempenho.
- **Banda Larga e Qualidade de Serviço (QoS):** a maioria das aplicações das redes em malha sem fio são para o acesso à banda larga e, neste caso, a qualidade do serviço é um requisito essencial para garantir a satisfação do cliente.

Novas métricas de roteamento devem ser consideradas a fim de aumentar o desempenho da rede;

- **Compatibilidade e Interoperabilidade:** os nodos roteadores devem possuir funções para interligar os mais variados tipos de redes heterogêneas de forma simples e transparente;
- **Segurança:** este assunto é bastante crítico na maioria dos sistemas computacionais e nas RMSF a questão da segurança se agrava, pois geralmente as RMSF são projetadas para ter uma grande área de cobertura. Com isso, a vulnerabilidade a ataques e anomalias por intrusos é maior, devendo-se aplicar mecanismos de segurança capazes de inibir ações neste sentido. Neste caso é desejável o uso de técnicas de detecção de intrusão, que se ajustem à demanda das RMSF.
- **Facilidade de uso:** é desejável que a rede seja autônoma e auto-organizada, a fim de se ter mínima intervenção do administrador para sua instalação e manutenção. Novos algoritmos e protocolos específicos para auto-organização e auto-gerenciamento em redes em malha sem fio devem ser projetados.

2.6.1 Projetos

Devido à grande demanda por conectividade, algumas empresas e universidades estão trabalhando em direção ao desenvolvimento de projetos em larga escala para implementar RMSF mais robustas e que consigam obter uma performance satisfatória à medida que os seus usuários entrem na rede. Dentre a gama de projetos acadêmicos e comerciais pode-se citar alguns como:

- **RoofNet:** é uma RMSF experimental baseada nas variantes do protocolo 802.11 em desenvolvimento pelo laboratório de Ciência da Computação e Inteligência artificial (CSAIL) do MIT (Massachusetts Institute of Technology)(AGUAYO et al., 2004), que fornece Internet banda larga para os usuários situados na cidade de Cambridge. Este projeto de pesquisa tem como principal objetivo avaliar o desempenho do protocolo 802.11 em redes de larga escala. Para isso testes foram feitos para encontrar a melhor vazão para as rotas e o melhor caminho com uma menor perda de pacotes. Para o experimento foram utilizados 50 nodos padronizados e fornecidos pelo próprio MIT e espalhados em perímetro urbano aos arredores da universidade. Os nodos consistem de

computadores pessoais (PCs) que utilizam sistema operacional Linux, com interfaces de redes sem fio 802.11b e antenas omnidirecionais. Os nodos são situados em áreas de altitude elevada, neste caso, o topo dos prédios dos participantes da rede. Para o roteamento dos pacotes é utilizado Srcr (AGUAYO et al., 2004), o qual é baseado no protocolo DSR e difere-se principalmente na métrica de custo para encontrar a melhor rota. Enquanto que DSR utiliza contagem dos saltos como métrica, o Srcr utiliza uma métrica denominada ETT (Estimated Transmission Time), que deriva da ETX (Estimated Transmission Count) (COUTO et al., 2003), que leva em consideração a taxa de perda de pacotes, como adicional para encontrar o melhor caminho. O baixo custo de implementação aliado à disponibilidade de equipamentos viabiliza o desenvolvimento deste tipo de rede em larga escala, mas a real preocupação é no protocolo de roteamento, como mostram as avaliações feitas por (ROOFNET, 2005).



Figura 2.5: Mapa da cidade de Cambridge onde está localizado o projeto Roofnet (AGUAYO et al., 2004)

Na Figura 2.5 pode ser observado o mapa de disposição dos nodos envolvidos na rede, e as respectivas taxas de transmissões dos enlaces, que podem variar de 1 KB/s a 600 KB/s.

- **VMesh:** Neste projeto a Universidade de Thessaly na Grécia apresenta um modelo de RMSF, que tem como objetivos pesquisar e avaliar um modelo

de baixo custo de rede em larga escala, utilizando roteadores 802.11b/g e sistema operacional Linux, com intuito de expandir a conectividade sem fio aos arredores da cidade de Volos na Grécia. O público alvo para utilização da rede são os alunos da própria universidade, pois o projeto disponibiliza a eles o acesso aos servidores situados na universidade, juntamente com o acesso à Internet banda larga das suas próprias casas. Outro objetivo deste projeto é criar uma plataforma de testes para pesquisa e avaliação de softwares e algoritmos na área de redes e sistemas distribuídos (TSARMPOPOULOS et al., 2005).

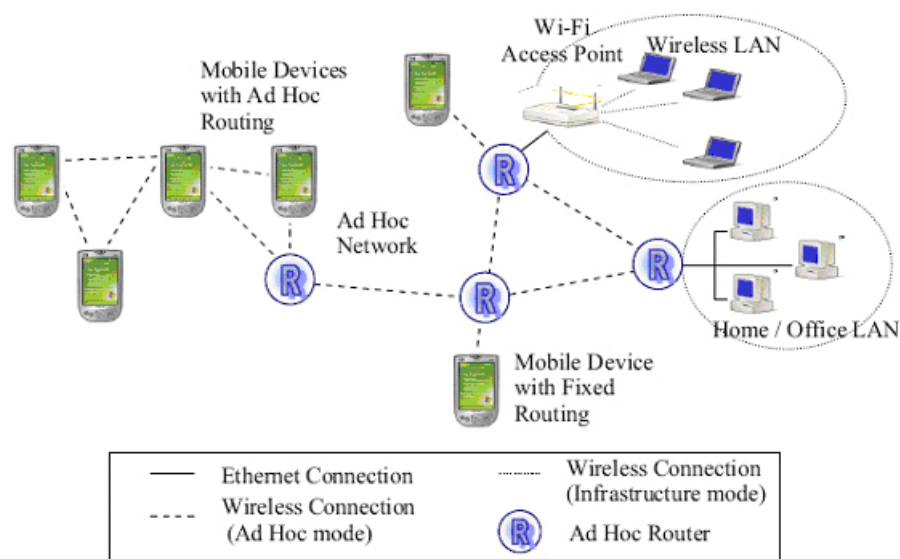


Figura 2.6: Modelo da arquitetura da rede Mesh da Universidade de Thessaly (TSARMPOPOULOS et al., 2005)

A Figura 2.6 demonstra uma visão geral da arquitetura da rede, o qual consiste de vários nodos estacionários e móveis juntamente com as sub-redes criadas e acopladas aos nodos roteadores, a fim de estender as funcionalidades da rede. Além disso, é interessante ressaltar que neste experimento foram usados, conexão cabeada para interligar as sub-redes Wi-Fi e LAN para comunicação direta entre os nodos roteadores, e para comunicação de alguns roteadores com seus clientes foram utilizados enlaces no padrão 802.11. Para alcançar uma melhor conectividade da rede, os nodos roteadores foram dispostos no alto dos prédios ou telhados de casas e possuem a função de permitir que os nodos clientes se conectem na rede. Para o roteamento dos pacotes é utilizado o protocolo OLSR, pois a rede é capaz de prever a quantidade de nodos móveis que desejam se conectar à mesma. Para os clientes que desejam

se conectar à rede é dado um endereço estático e cada roteador pode suportar até 29 clientes conectados pelas portas LAN. O hardware utilizado neste projeto foi desenvolvido por Pascal Dornier, conhecido como WRAP (Wireless Router Application Platform) (DORNIER, 2002). Devido ao alto custo da aquisição desta solução para os alunos, o projeto tornou-se inviável, e novas pesquisas tiveram que ser feitas pela equipe responsável pelo projeto a fim de encontrar uma solução mais adequada ao problema encontrado. A solução encontrada foi a utilização dos roteadores Linksys WRT54G e WRT54GS, onde seu *firmware* é baseado em uma versão mais enxuta do sistema operacional Linux, denominada OpenWRT (OPENWRT, 2005). Com essa nova adoção, testes e experimentos se mostraram satisfatórios aos requisitos buscados pelo projeto.

- **ReMesh - Rede Mesh de Acesso Universitário Faixa Larga Sem Fio:**

O projeto Re-mesh é uma iniciativa da RNP (Rede Nacional de Pesquisa), formado por professores e alunos do departamento de Engenharia de Telecomunicações (TET) e do instituto de Computação da Universidade Federal Fluminense (UFF)(REMESH, 2006). Este projeto implementou e realizou um teste piloto de uma RMSF em larga escala para acesso universitário utilizando o padrão 802.11 nos arredores do campus da UFF de Praia Vermelha. O objetivo principal deste projeto é demonstrar a viabilidade em construir uma rede a baixo custo para acesso à Internet para os alunos da universidade, pois o alto custo em adquirir um serviço ADSL ou cabo neste caso se torna inviável. Neste caso, o desenvolvimento e implantação de uma rede em larga escala sem fio para acesso comunitário, se torna uma alternativa economicamente viável para a comunidade universitária. O projeto consistiu em duas etapas: a primeira foi implementado um teste *indoor* e posteriormente foi implementado o projeto *outdoor*.

Na Figura 2.7 (a) pode ser observado o modelo *indoor*, primeiramente testado em laboratório para posteriormente construção do modelo *outdoor*, que pode ser observado na Figura 2.7 (b). O *testbed indoor* consiste de sete nodos que são identificados pelo nome dos laboratórios participantes (LCC, Add, Dout, IC, Rio, TCC e Midiacom), e estão localizados em dois andares adjacentes do mesmo prédio. O nodo seis está localizado no quarto andar, enquanto que, o restante dos nodos estão localizados no terceiro andar. Os enlaces são numerados de L1 ao L20 e as linhas sólidas demonstram a melhor qualidade dos enlaces, enquanto que as linhas tracejadas demonstram menor qualidade dos

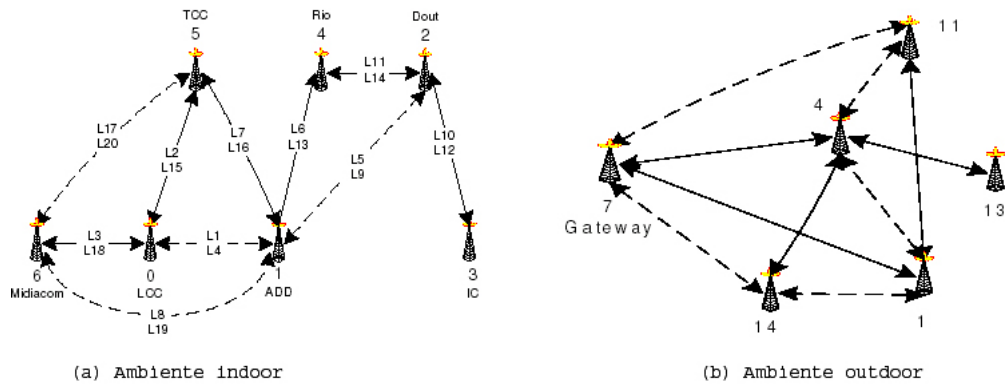


Figura 2.7: Modelo da rede do projeto Re-Mesh indoor e outdoor (PASSOS et al., 2006)

enlaces de comunicação.

Neste experimento os autores testaram o protocolo de roteamento OLSR com a métrica ETX (*Expected Transmission Count*) e constataram uma grande perda de pacotes e instabilidade das rotas, o que motivou os autores a desenvolver uma métrica mais eficiente para o protocolo OLSR. Esta métrica foi denominada OLSR-ML (minimum loss) (PASSOS et al., 2006). O teste *indoor* foi importante para validar o *hardware* e as escolhas do software para implementação do projeto *outdoor*. O projeto *outdoor* foi implementando utilizando a nova métrica testada inicialmente no ambiente *indoor*. O projeto *outdoor* consistiu de seis nodos-roteadores colocados no topo dos prédios participantes. Como pode ser observado na Figura 2.7 (b) o nodo-roteador sete tem a função de *gateway*, ou seja, ele é responsável pelo compartilhamento da Internet para os demais nodos-roteadores. Os resultados mostraram que a performance da rede foi melhorada com a adição da métrica desenvolvida pelos autores e houve uma maior estabilidade das rotas, menor perda de pacotes e atraso e uma melhora significativa na vazão da rede.

- **Motorola wireless Mesh Networks** : um dos projetos pioneiros de RMSF comercial é o da empresa Motorola.

Neste projeto a empresa investe em conectividade em larga escala, através de uma solução denominada *Canopy Hot Zone* (MOTOROLA, 2007a). Na Figura 2.8 pode ser visualizada a rede em malha sem fio do projeto *Canopy Hot Zone*. Observa-se que os nodos roteadores estão localizados nos telhados das casas, em postes de energia e topos de prédios, com isso o *backbone* possui pouca interferência e uma área de cobertura mais estendida.

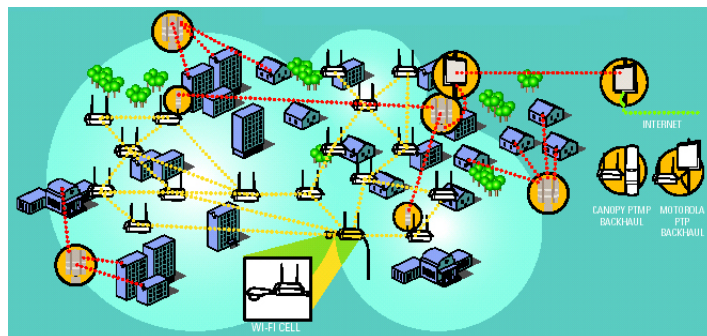


Figura 2.8: Arquitetura do projeto Canopy Hot Zone (MOTOROLA, 2007a)

- 4G Mesh Cube:** outro projeto comercial é o da empresa *4G-Systems*, o qual desenvolveu uma plataforma de *hardware* denominada *Mesh Cube* (4G-SYSTEMS, 2007) que tem por função prover acesso a redes sem fio especificamente as em malha sem fio. O protótipo do roteador como observado na Figura 2.9, possui dimensões reduzidas (7x5x7 cm) a fim de adequar-se a lugares bastante pequenos. A durabilidade e flexibilidade além da segurança e encriptação dos dados são características fortes neste protótipo. Além disso, possui baixo consumo de energia (4-6W), possui também uma entrada Ethernet 100Mbps e duas interfaces sem fio (802.11/a/b/g), um processador de 400Mhz e 64 Mb de memória RAM. O protocolo de roteamento utilizado neste protótipo é o MMRP (MITRE, 2005). O sistema operacional suportado pela plataforma é o Nylon, o qual é baseado no projeto *OpenEmbedded*, que tem como objetivo principal facilitar a integração e construção de sistemas para dispositivos embutidos.

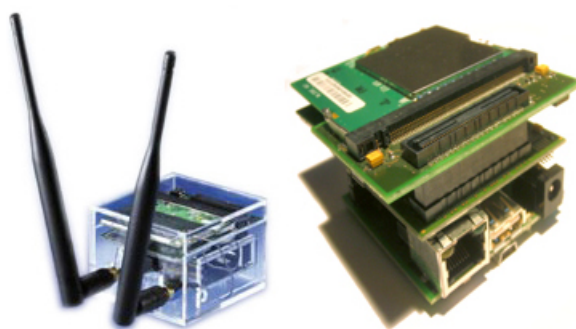


Figura 2.9: Plataforma de *Hardware* do *Mesh Cube* (4G-SYSTEMS, 2007)

Como pode ser observado há uma boa gama de projetos já desenvolvidos com base nas RMSF. Uma lista de mais projetos pode ser acessada em (MESH, 2007).

2.7 Padronização das Redes em Malha sem Fio

Em 2005 foi proposto o padrão 802.11s, que define como os dispositivos podem interconectar a malha sem fio. Duas propostas se destacam neste processo: a *Wi-Mesh Alliance*, desenvolvida pelas empresas Nortel, Thompson e Swisscom Innovation e Philips, e a SEEMesh (*Simple, Efficient and Extensible Mesh*) das empresas Nokia, NTT Docomo, Texas Instruments, Intel e Cisco com o intuito de discutir o futuro do padrão 802.11s. Um esboço da arquitetura proposta para o padrão 802.11s pode ser encontrado em (MESH, 2006).

2.8 Algumas Considerações

Este capítulo apresentou uma visão geral sobre RMSF, destacando-se a definição, as diferenças entre as RMSF e as redes *ad hoc*, a infra-estrutura da rede, suas características, os cenários de aplicação, os fatores que afetam a performance, a padronização e os projetos. As RMSF possuem inúmeras aplicações e benefícios em comparação com as redes *ad hoc*, porém, uma característica desejável em RMSF é a capacidade de se auto-organizar, ou seja, diminuir a intervenção do administrador da rede para seu gerenciamento e manutenção. As RMSF atuais só podem realizar parcialmente este objetivo (AKYLDIZ et al., 2005), pois as abordagens existentes tratam a auto-organização de forma independente, não destacando uma visão mais ampla da rede. No próximo capítulo serão abordados os conceitos relacionados à auto-organização de redes de computadores.

3 *Auto-Organização*

Diversos paradigmas computacionais, atualmente encontrados na literatura, originaram-se puramente da observação da natureza, ou de alguns fenômenos biológicos. Com a auto-organização não poderia ser diferente. A auto-organização é um fenômeno natural que pode ser encontrado no nosso dia a dia através de comportamentos emergentes, ou seja, as entidades de forma individual desempenham tarefas simples e específicas, que juntas se transformam na auto-organização de um sistema complexo. Um exemplo disso pode ser observado no comportamento das formigas que se auto-organizam para um bem global comum: a busca por alimento. Além das inúmeras aplicações que podem ser desenvolvidas utilizando-se como base a auto-organização, pode-se também aplicá-la a área computacional, com o intuito de resolver problemas de natureza complexos, sem qualquer intervenção manual ou externa. Já no caso da área computacional, mais precisamente em redes de computadores sem fio, a auto-organização é um atrativo, pois pode auxiliar o protocolo de roteamento nos processos de auto-configuração, auto-otimização, auto-cura e auto-proteção, diminuindo assim a intervenção do administrador da rede para tais procedimentos.

Neste capítulo é apresentada uma conceituação básica sobre auto-organização, mostrando um breve histórico, algumas definições e benefícios, bem como descrevendo algumas características relevantes, enfatizando redes de computadores sem fio auto-organizadas, o qual é o foco principal deste trabalho.

3.1 *Conceituação*

A auto-organização é um conceito que pode ser aplicado a diversas áreas, tais como: Engenharia, Física, Química, Biologia, Matemática, Computação, dentre outras. O estudo da auto-organização foi iniciado em 1947 pelo engenheiro e psiquiatra Willian Ross Ashby, que introduziu o conceito na cibernética e, também, foi o pioneiro no estudo de sistemas complexos e adaptativos. Posteriormente, Ashby escreveu diversos artigos analisando os mecanismos de auto-organização em diferen-

tes aspectos e, ainda nesta mesma época, desenvolveu uma máquina denominada ‘Homeostat’, com o intuito de imitar o comportamento cerebral através de conceitos mecânicos (ASBHY, 1947) (ASHBY et al., 1962). Inúmeros são os comportamentos encontrados na natureza que demonstram características auto-organizadas. Exemplos disso podem ser encontrados nas células do corpo humano, em pequenos organismos, no comportamento de alguns insetos, como em colônias de formigas, vespas, vaga-lumes, cupins, cardumes de peixes, revoadas de pássaros, dentre outros, cada qual com sua peculiaridade de organização.

Do ponto de vista dos sistemas biológicos, outro pesquisador, Camazine et al. (2003), mostrou como os sistemas complexos encontrados em partículas microscópicas, se auto-organizavam sem intervenção manual ou externa, a fim de proporcionar um objetivo global comum. Outros exemplos de comportamentos auto-organizados podem ser observados na sociedade, na psicologia e também no comportamento do cérebro humano (JUNIOR, 2001). Em todos esses exemplos observa-se um comportamento em que as entidades participantes estabelecem uma estrutura organizacional, que não necessita de controle centralizado ou intervenções externas para execução de suas tarefas. Algumas características naturais dos sistemas auto-organizados, como a capacidade de crescimento e a adaptabilidade a mudanças, tornam o conceito de auto-organização altamente desejável em sistemas complexos e adaptativos.

Deve-se ressaltar o entendimento de sistemas complexos como um sistema composto por uma junção de comportamentos simples interconectados, que juntos exibem um alto grau de complexidade, que emergem de um comportamento de alta ordem (WIKIPEDIA, 2007a) (CAMAZINE et al., 2003) (JOHNSON, 2003).

Dentro de sistemas complexos, o conceito de auto-organização é diretamente influenciado por entidades emergentes,¹ que surgem de um nível de elementos relativamente simples em direção a formas de comportamento mais sofisticados (JOHNSON, 2003), ou seja, cada entidade individual influencia na interação do sistema como um todo.

Particularmente, pode-se dizer que a auto-organização é tida como um paradigma aplicável à resolução de uma infinidade de problemas complexos e diferencia-se de técnicas convencionais, que muitas vezes são limitadas e não tratam da complexidade do problema de forma eficaz. Além disso, os vastos cenários de aplicação que podem ser constituídos com base na auto-organização a levam a ser um conceito

¹Entende-se emergência como um processo de formação de modelos complexos a partir de regras simples (WIKIPEDIA, 2007b).

promissor. Outra observação interessante é que os métodos e técnicas oriundas de modelos biológicos auto-organizados não estão inteiramente entendidos, ou seja, faz-se necessário ainda um amadurecimento por parte da comunidade científica, a fim por exemplo de aplicar os modelos auto-organizados para resolução de problemas computacionais complexos.

3.2 Definições

O conceito de auto-organização ocorre em diversas áreas da ciência e é considerado um campo interdisciplinar e heterogêneo (PREHOFER; BETTSTETTER, 2005), isso a torna difícil de ser definida. Diversos pesquisadores estudam o conceito de auto-organização, mas possuem diferentes pontos de vista. Devido à grande quantidade de definições que cercam o conceito, optou-se por enfatizar alguns pesquisadores influentes da área. Uma autoridade na área biológica, a visão de Camazine et al. (2003), define a auto-organização como:

“Um processo no qual padrões de nível global de um sistema emergem de inúmeras interações de mais baixo nível. Além disso, as regras executadas especificamente pelas interações entre os componentes do sistema são executadas usando somente informações locais, sem referência para um padrão global.”

Já Ashby et al. (1962) enfatizam que:

“A auto-organização pode ser definida como a interação de múltiplos componentes para um objetivo global comum.”

Enquanto que Johnson (2003) define auto-organização como:

“Sistemas que dispensam controle centralizado e que se auto-organizam de baixo para cima (bottom-up)”

Por sua vez, Collier e Taylor (2004), definem auto-organização como:

“Sistemas compostos por uma vasta coleção de unidades que coordenam-se umas com as outras, no sentido de alcançar o seu objetivo mais eficientemente”

Os autores (COLLIER; TAYLOR, 2004) ainda enfatizam que a definição de auto-organização pode ser enumerada dentro de uma lista de características:

1. O sistema deve ser composto de unidades que possam reagir individualmente a estímulos locais;
2. As unidades atuam de forma cooperativa com objetivo de dividir a carga de trabalho;
3. O sistema como um todo se adapta a fim de executar um ou vários objetivos de forma eficiente;

Já para este trabalho foi considerada a seguinte definição, correlacionada à área de redes de computadores sem fio (VASUDEVAN, 2006):

“Os nodos sem fio auto-organizados são entidades responsáveis por executar tarefas requeridas pela aplicação de forma eficaz e sem intervenção humana, obtendo como benefício a redução do custo de instalação.”

Sumarizando, observou-se que as definições de auto-organização direcionam-se para sistemas complexos e que exigem um alto grau de organização, em que muitas vezes seriam difíceis ou até mesmo impossíveis intervenções manuais ou externas.

3.3 Propriedades dos Sistemas Auto-Organizados

Nesta subseção é apresentada uma breve contextualização sobre as principais propriedades dos sistemas complexos e auto-organizados, de acordo com alguns pesquisadores (PREHOFER; BETTSTETTER, 2005) (HERRMANN, 2006) (TANG; TIAN-FIELD, 2006) (HEYLIGHEN; GERSHENSON, 2003).

- **Robustos a Falhas e Perturbações:** esta propriedades torna os sistemas auto-organizados mais robustos devido a redundância de entidades. Neste caso, mesmo que um número razoável de entidades falhem, o sistema como um todo continua funcionando. À medida que as entidades vão falhando, o sistema degrada-se de forma mais suave.
- **Randomização:** os componentes do sistema auto-organizado provêm um certo grau de randomização em seu comportamento, com o objetivo de explorar possíveis ações ou novos estados. Neste sentido um exemplo comumente encontrado é o comportamento das formigas, que se movem de forma aleatória com o objetivo de buscar comida para a colônia. Ao encontrar uma fonte de

comida, as formigas deixam marcas denominadas feromônios que servem como comunicação com as demais formigas, para sinalizar onde está a fonte de alimento. Pode-se dizer que a capacidade de randomização é vista como uma forma de conhecimento do ambiente através de ações aleatórias.

- **Sem Controle Centralizado:** os sistemas auto-organizados dispensam controle centralizado, pois cada falha de uma entidade central, por exemplo, pode acarretar em um problema geral do sistema. Cada entidade interage uma com a outra de forma distribuída e cada interação deve ser coordenada de forma local com intuito de organizar um comportamento global comum. Um exemplo disso pode ser visto em redes de computadores, qual podem ser entidades distribuídas e não possuem controle centralizado. Neste caso, a auto-organização pode ser uma alternativa à convencional arquitetura de rede centralizada, conforme pode ser observado na Figura 3.1 .

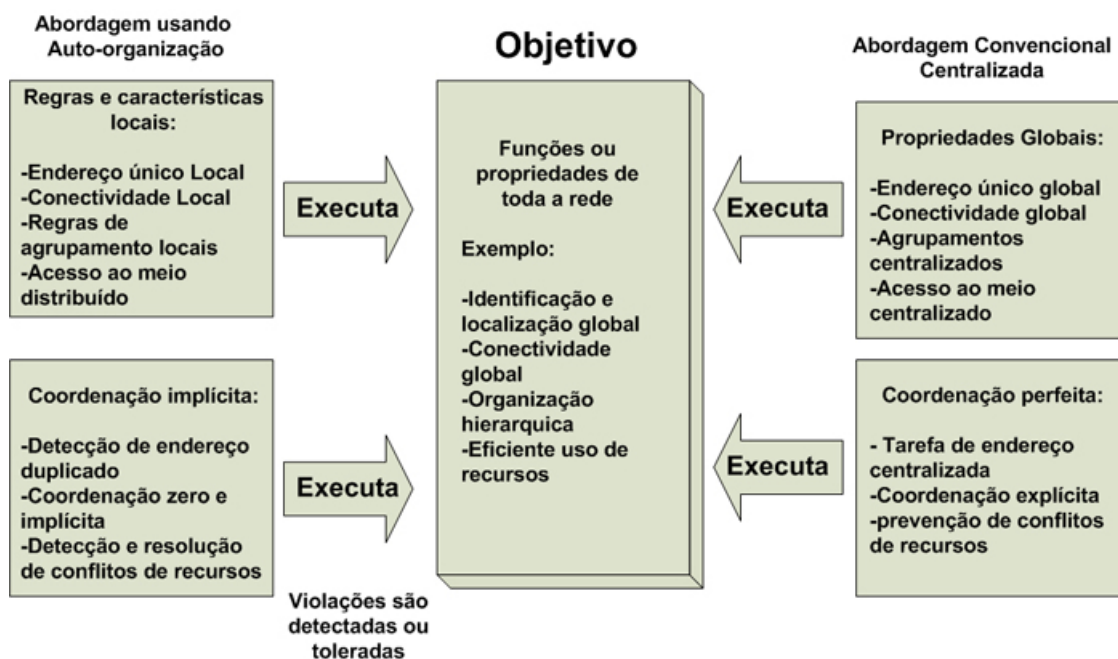


Figura 3.1: Comparação de uma abordagem de rede auto-organizada com uma abordagem de rede convencional centralizada. Adaptada de (PREHOFER; BETTSTETTER, 2005)

A Figura 3.1 representa um comparativo entre redes auto-organizadas e redes convencionais. Nas redes auto-organizadas a visão da rede limita-se a regras e propriedades locais e à coordenação implícita. Já nas redes convencionais a visão se limita às propriedades globais e à coordenação perfeita, ou seja, os mecanismos de endereçamento coordenação e detecção de conflitos trabalham

de forma integrada a fim de se ter um nível maior de confiabilidade do sistema. As visões tanto auto-organizadas como convencionais, convergem de forma diferenciada aos objetivos gerais da rede, que são: identificação, localização e conectividade global, organização hierárquica e o eficiente uso de recursos.

- **Adaptabilidade:** nesta propriedade as entidades que compõem o sistema reagem de forma dinâmica a mudanças e são capazes de se adaptar a mudanças no ambiente. Um exemplo de adaptabilidade neste caso, ainda citando o comportamento das formigas, seria quando um obstáculo é colocado no caminho traçado para a fonte de alimento e o ninho. Neste caso, observa-se a necessidade de encontrar caminhos alternativos para manter o trajeto e não prejudicar a coleta de alimento.
- **Escalabilidade:** isto significa que o sistema ainda trabalha satisfatoriamente conforme a quantidade de entidades aumenta no sistema. Um exemplo dessa afirmação pode ser encontrada em Dressler (2006), que demonstra a relação entre escalabilidade e determinismo (Figura 3.2). Observa-se que à me-

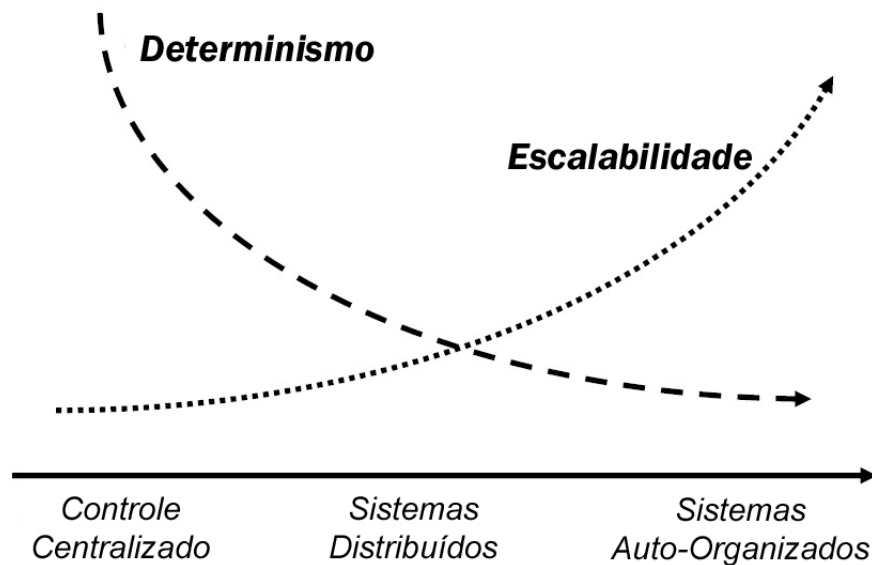


Figura 3.2: Comparação entre escalabilidade versus determinismo entre controle centralizado e sistema auto-Organizado (DRESSLER, 2006)

dida que o grau de evolução dos sistemas centralizados para os sistemas auto-organizados em direção aos sistemas distribuídos aumenta, pode-se notar que o nível de determinismo diminui. Em conseqüência, o nível de escalabilidade aumenta significativamente, demonstrando que sistemas auto-organizados possuem baixo determinismo e uma alta escalabilidade.

- **Comportamento Emergente:** ocorre quando uma quantidade de entidades que desempenham um papel simples, contribuem para formar comportamentos complexos e auto-organizados. Esta característica ocorre na maioria dos sistemas complexos, em que cada entidade possui um papel específico e interage de forma auto-organizada para um objetivo global comum. Um exemplo disso pode ser observado mais uma vez na colônia de formigas, pois a rainha não especifica quais regras devem ser seguidas pela colônia, mas cada formiga reage a estímulos químicos deixados no ambiente. Cada formiga é uma entidade autônoma que tem sua tomada de decisão baseada no conhecimento do ambiente.

As vantagens eminentes em sistemas auto-organizados, de acordo com Jelasy et al. (2006), devem ter implementação simples, potencialidade para escalabilidade e, além disso, trabalhar em ambientes com taxas razoáveis de dinamismo.

Sumarizando, muitas das propriedades supracitadas são desejáveis em modernas infra-estruturas de software e também em redes de computadores sem fio (HERMANN, 2006) (TANG; TIANFIELD, 2006) (MAMEI et al., 2006) (MEER et al., 2006). Em se tratando de rede de computadores sem fio é desejável que a rede seja robusta a falhas e perturbações, adaptável a mudanças no ambiente e possua uma boa capacidade para escalabilidade.

3.4 Abordagens *Top-Down* e *Bottom-up*

Nesta seção é feita uma breve contextualização sobre a divisão dos métodos de auto-organização encontrados na literatura, conforme o ponto de vista de Herrmann (2006). O autor divide as abordagens de auto-organização em *top-down* e *bottom-up*. É dado maior ênfase nas abordagens *top-down*, como base para o desenvolvimento da arquitetura de auto-organização proposta nesta dissertação.

De acordo com Herrmann (2006), nestes últimos anos diversas abordagens envolvendo gerenciamento de grandes infra-estruturas de informações foram desenvolvidas. Um conceito que está emergindo e está ligado de forma correlata a auto-organização é o auto-gerenciamento (JELASITY et al., 2006)(HERRMANN et al., 2005).

Auto-gerenciamento é um processo em que os sistemas computacionais dispensam a intervenção do administrador da rede para sua configuração e manutenção e são capazes de forma autônoma, de se auto-gerenciarem conforme mudanças no ambiente.

De acordo com Herrmann (2006), dois diferentes níveis de abstração são encontrados em sistemas auto-gerenciados, os quais são descritos a seguir:

- **Alto Nível de Abstração:** Neste nível são encontradas as abordagens *top-down*, que possuem um foco mais conceitual. Dentre as abordagens, pode-se citar: *Computação Autônômica*, *Redes Autônômicas*, *Computação Proativa*.
- **Baixo Nível de Abstração:** Neste nível se encontram as abordagens *bottom-up*, que possuem um foco mais específico. Dentre as abordagens podem ser encontradas soluções como auto-otimização de sistemas de software, auto-estabilização, inteligência coletiva e sistemas de segurança utilizando sistemas imunológicos. Neste caso, seriam abordagens mais específicas para um problema estritamente específico.

Nesta dissertação foram utilizados conceitos relacionados à computação autônômica, com isso observou-se a necessidade de se dar uma visão geral sobre as abordagens *top-down*, mais especificamente à computação autônômica. Suas características são aplicáveis em redes autônômicas, as quais são o foco de desenvolvimento da abordagem de auto-organização proposta para as RMSF.

3.4.1 Computação Autônômica (*Autonomic Computing*)

Nesta seção são explicados alguns conceitos voltados à computação autônômica, necessários ao entendimento de alguns aspectos do trabalho de pesquisa apresentado nesta dissertação.

A computação autônômica foi iniciada pela *International Business Machines* (IBM) em meados do ano de 2001, com o objetivo de desenvolver sistemas computacionais auto-gerenciáveis, no intuito de diminuir a complexidade de manutenção e gerenciamento desempenhada pelo administrador da rede. O projeto foi inspirado no comportamento do sistema nervoso humano, que tem comportamento autônomo para o gerenciamento das principais funções vitais do corpo, como: controle dos batimentos cardíacos e da temperatura corpórea.

De acordo com Herrmann et al. (2005), em sistemas autônômicos o que se destaca são os elementos autônômicos. Um elemento autônômico consiste de elementos a serem gerenciados, juntamente com um gerenciador autônômico, que possui como função controlar os elementos a serem gerenciados. Os elementos a serem gerenciados, neste caso, podem ser tanto um *hardware* ou como um *software*, ou a combi-

nação de ambos. De acordo com White et al. (2004), o comportamento recomendado oferecido por um elemento autônomo é garantir performance, confiança, disponibilidade e segurança ao elemento a ser gerenciado. Observa-se que a função principal dentro de um elemento autônomo é o controle realimentado, isto é, a combinação de componentes que atuam juntos para manter os objetivos do sistema (MULLER et al., 2006).

De acordo com Muller et al. (2006), o sistema de controle realimentado é basicamente dividido em: livre de ciclo, onde a saída não afeta a entrada, ou seja, uma ação de saída só é executada uma única vez, (exemplos: torradeiras automáticas e alarmes) e, em controle realimentado, que têm como a sua ação de saída influenciada pela entrada, como forma de manter o valor de saída desejado (exemplos: termostatos e sistemas de controle de velocidade) (MULLER et al., 2006). Um exemplo de um controle realimentado padrão pode ser observado na Figura 3.3 .

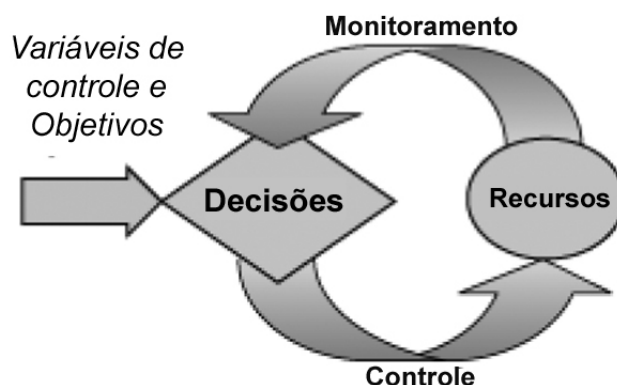


Figura 3.3: Controle realimentado padrão (HERRMANN et al., 2005)

Além do controle realimentado, observa-se a necessidade de interação, através de políticas ou regras pré-estabelecidas, com o intuito de especificar determinadas ações feitas pelos elementos autônomos. De acordo com White et al. (2004), uma regra ou política de gerenciamento é uma representação de um comportamento externo desejável. Basicamente existem três formas de políticas de gerenciamento conforme Kephart e Walsh (2004): políticas baseadas ações, objetivos e funções de utilidade. As políticas baseadas em ações baseiam-se em típicas formas de IF (Condição), THEN (Ação), em que a condição representa um estado ou um conjunto de estados, que satisfaçam à condição. Um exemplo das regras aplicadas às políticas baseadas em ações, pode ser vista na Figura 3.4 .

Neste exemplo observa-se que a ACTION(1) incrementa a carga da CPU(g) em cinco por cento e na ACTION(2) é incrementado a carga da CPU(s) em cinco

```

IF ((ACTION1: Increase CPUG by 5%)
    AND (ACTION2: Increase CPUS by 5%)
    AND (CPUG + CPUS > 90%))
THEN(Ignore ACTION2)

```

Figura 3.4: Exemplo de políticas baseadas em ações (KEPHART; WALSH, 2004)

porcento e se a carga da CPU(g) + CPU(s) for maior que noventa por cento então a ACTION(2) é ignorada.

Já as políticas baseadas em objetivos, representam a forma de como chegar a uma condição satisfeita, sem especificar os passos para tal, ou seja, neste tipo de política, ao invés de especificar exatamente o que um estado irá fazer, pode-se configurar objetivos onde as decisões são tomadas de forma dinâmica. Exemplo disso é o uso de *Workflow* para possíveis seqüências de ações.

Por último, as políticas baseadas em funções de utilidade representam uma forma de interação com as políticas baseadas em objetivo, pois identificam quais os estados desejáveis ou não dentro de uma seqüência de ações. Geralmente são representadas por funções binárias (KEPHART; WALSH, 2004). Na seção seguinte é contemplado, de forma conceitual, o projeto para computação autônômica proposto por Kephart e Chess (2003), que foi a base para a maioria dos projetos relacionados à computação autônômica encontrados na literatura.

3.4.2 Projeto Para Computação Autônômica Proposto Por Kephart e Chess (2003)

O projeto é formada por uma coleção de componentes, chamados elementos autônômicos, que possuem papéis específicos dentro da arquitetura. Um elemento autônômico é um agente que gerencia o comportamento interno e suas relações com outros agentes, respeitando regras e políticas. Cada elemento autônômico é dividido basicamente em duas partes integradas (KEPHART; CHESS, 2003) (HORN, 2001):

- **Elementos gerenciados:** São recursos gerenciados por um gerenciador, que pode ser um *hardware*, aplicação de software ou um sistema completo.
- **Gerenciador Autônômico:** o gerenciador autônômico interage no sistema, ou com os recursos, através de dois componentes de interface, denominados, sensores e executores. Os sensores são usados para checar a informação a res-

peito dos recursos, provendo uma interface para acesso ao estado dos recursos. Já os executores provêm interfaces, as quais são usados pelos gerenciadores autônomicos para mudar o estado dos recursos. Um gerenciador autônomico é responsável por prover um comportamento autônomico para o componente. Isto é implementado como um controle realimentado e incorporado às funções de monitoramento, análise, conhecimento, planejamento e execução.

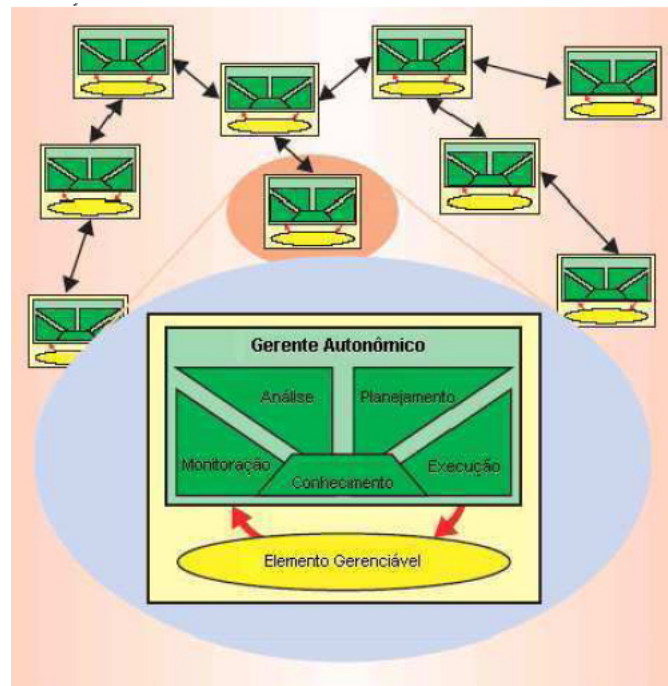


Figura 3.5: Controle realimentado proposto por (KEPHART; CHESS, 2003)

O controle realimentado proposto por Kephart e Chess (2003) pode ser observado na Figura 3.5 .

1. A fase de monitoramento coleta informações dos elementos a serem gerenciados;
2. A fase de análise reporta eventos com o intuito de verificar a situação do elemento e, em caso de falhas, sugere uma ação de recuperação do elemento;
3. O planejamento especifica quais ações devem ser tomadas, em caso de falha de algum elemento;
4. A fase de execução é responsável por executar as ações planejadas pela fase de planejamento. Essas ações são executadas nos elementos gerenciados;

5. A fase de conhecimento é compartilhada por todas as outras fases citadas anteriormente.

A computação autonômica ainda provê algumas capacidades que são desejáveis, no que tange ao gerenciamento de sistemas de grande porte. Essas capacidades formam o conceito conhecido como Auto- x (*Self-x*). O x representa as capacidades de auto-configuração (*self-configuration*), auto-cura (*self-healing*), auto-otimização (*self-optimization*) e auto-proteção (*self-protection*) (BABAOGU et al., 2005) Kephart e Chess (2003) (MULLER et al., 2006) (PARASHAR; HARIRI, 2006). Essas capacidades são conhecidas também pelo acrônimo CHOP (MILLER, 2005). Existem ainda outras capacidades como auto-monitoramento, auto-diagnóstico, auto-recuperação, auto-reparação, auto-ajuste, auto-adaptação, dentre outras, mas todas as capacidades são diferenciadas conforme o papel que desempenham no sistema (TANG; TIANFIELD, 2006). Do ponto de vista de Schmid et al. (2006), o termo autonômico ou sistema autonômico, refere-se à habilidade de um sistema em executar as capacidades de Auto- x .

Neste contexto destacam-se as quatro capacidades de Auto- x , que são descritas a seguir:

- **Auto-Configuração:** sistemas auto-configuráveis possuem como objetivo principal adaptar-se às constantes mudanças no ambiente; além disso, devem ter a capacidade de configurar-se de forma automática, diante de situações imprevisíveis (MULLER et al., 2006). Assim, os sistemas auto-configurados podem ser adaptados em ambientes dinâmicos, como por exemplo, uma rede sem fio faixa larga.
- **Auto-Otimização:** algumas aplicações necessitam que os parâmetros sejam ajustados de forma dinâmica, autônoma e otimizada, dessa forma, o sistema pode aumentar a sua performance. Além disso, a capacidade de auto-otimização pode prover uma maior eficiência no que tange a balanceamento de carga e o uso eficiente dos recursos. Para isso, o sistema monitora continuamente e otimiza seus recursos de forma a melhorar o desempenho do sistema (MULLER et al., 2006). De acordo com Schmid et al. (2006), o processo de otimização compreende em garantir que todo o sistema esteja rodando em níveis ótimos de performance.
- **Auto-Cura:** em termos de sistemas computacionais, essa capacidade é responsável por detectar, localizar e reparar falhas automaticamente no ambiente

(WHITE et al., 2004).

- **Auto-Proteção:** A capacidade de auto-proteção pode detectar possíveis ataques ou comportamentos intrusivos na rede. O sistema deve ser capaz de proteger a si próprio, através por exemplo, do uso de técnicas de reconhecimento de padrões (WHITE et al., 2004).

Deve ser observado que as capacidades de auto-x não são inteiramente independentes, ou seja, no caso da auto-configuração e a auto-otimização há uma forte correlação entre essas subcategorias. Em uma rede de computadores, por exemplo, enquanto uma das funções auto-configura os parâmetros de um protocolo de roteamento, a outra otimiza os parâmetros de forma a melhorar a performance da rede.

- **Auto-Configuração e Auto-Otimização:** Ambas as capacidades são fortemente correlacionadas. A otimização geralmente é o complemento da auto-configuração. A auto-configuração, como mencionado, é a habilidade de uma entidade autônoma de adaptar-se a mudanças dependendo das condições do ambiente. Já a auto-otimização tem como alvo a melhora de desempenho do sistema, configurando os parâmetros de forma otimizada.

Conceito	Computação Atual	Computação Auto-Organizada
Auto-Configuração	Diversos centros de informação possuem inúmeras plataformas e configurar e integrar tudo isso impacta em erros e complexidade de tempo elevada.	A Auto-Configuração dos componentes do sistema seguindo políticas de alto-nível. Impacta no ajuste automático do sistema, tendo como vantagem a economia de tempo de configuração e manutenção.
Auto-Otimização	Geralmente os sistemas possuem milhares de conjuntos de parâmetros não lineares. Este conjunto pode aumentar conforme novas atualizações vão sendo feitas.	Os componentes e o sistemas continuamente pode automaticamente otimizar os parâmetros do sistema a fim de melhorar a desempenho e a sua eficiência.
Auto-Cura	A localização de possíveis falhas no sistema é complexa e podem necessitar de um contingente grande de programadores.	O sistema automaticamente detecta, localiza, e repara possíveis problemas tanto de <i>hardware</i> como de <i>software</i> .
Auto-Proteção	Deteção e recuperação de ataques e falhas em cascata é feita de forma manual.	O sistema automaticamente pode se defender de ataques e também gerar mensagens de alerta para prevenir eles no futuro.

Figura 3.6: Comparativo entre computação atual e auto-organizada (autônômica), conforme as Capacidades de Auto-x adaptado de (KEPHART; CHESS, 2003)

Na Figura 3.6 destaca-se um comparativo entre computação atual e computação autônoma, enfatizando as quatro capacidades de Auto-x (configuração, otimização, cura e proteção), segundo Kephart e Chess (2003). Neste comparativo observou-se que as quatro capacidades de Auto-x estão direcionadas ao auto-gerenciamento de grandes infra-estruturas de sistemas de software.

3.5 Exemplos de Abordagens Auto-Organizadas em Redes de Computadores

Nesta seção são mostrados alguns exemplos do uso de auto-organização na área de redes de computadores.

As redes autônomicas representam os conceitos apresentados sobre computação autônoma, porém voltados às redes de computadores. A rede possui a capacidade de gerenciar cada nodo de forma autônoma e que dispense a intervenção do administrador para sua manutenção e configuração. Além disso, a rede pode ter a capacidade de se adaptar conforme as mudanças que possam ocorrer. Como exemplo de auto-organização em redes de computadores, pode-se citar o mecanismo de atribuição de endereços de IP (*Internet Protocol*), denominado DHCP (*Dynamic Host Configuration Protocol*). Neste exemplo observa-se que a atribuição manual dos IP's dentro de uma rede de computadores é feita geralmente pelo administrador da rede e ele deve manter uma tabela de todos os IP's atribuídos; isto se torna um processo dispendioso conforme a rede cresce. Neste sentido é desejável que a atribuição de IP's seja feita de forma dinâmica e auto-organizada. O protocolo DHCP ainda permite ao nodo adaptar-se a mudanças no ambiente, isto é, quando há necessidade de mudar de uma rede para outra será atribuído um novo endereço de IP de forma automática (PREHOFER; BETTSTETTER, 2005).

Outro exemplo de auto-organização pode-ser encontrado no protocolo TCP (*Transport Control Protocol*), o qual implementa um mecanismo de controle de congestionamento descentralizado. Isso reduz a taxa de perda de pacotes e aumenta significativamente a entrega correta de pacotes. Essa forma de adaptar-se ao tráfego é eficiente e escalável (PREHOFER; BETTSTETTER, 2005).

Além do DHCP e do TCP, outros exemplos podem ser encontrados. De acordo com Sarma et al. (2006), o mecanismo de IPV6 contém formas de auto-configuração das estações com o intuito de obter um endereço IP e outras informações que habilitam os nodos a entrarem na rede de forma transparente para os usuários. Essas

técnicas de auto-configuração podem ser classificadas dentro dos métodos de *stateless* e *stateful* (SARMA et al., 2006).

Além das inúmeras abordagens que podem utilizar auto-organização, alguns pesquisadores estão trabalhando para estender modelos auto-organizados, baseados em fenômenos naturais que supram a necessidade e organização em diferentes tipos de redes. Um exemplo é a auto-organização em redes *peer-to-peer* (BISON, 2006). Além disso, diversas aplicações têm sido desenvolvidas visando agregar desempenho em sistemas.

Em função das vantagens encontradas nos sistemas complexos e auto-organizados a abstração dessas características é desejável em redes de computadores sem fio. Geralmente as redes sem fio possuem uma grande quantidade de dispositivos; isso demanda organização e manutenção, muitas vezes não suportadas por técnicas convencionais. Neste caso a auto-organização se torna uma alternativa para a organização desses sistemas, de forma que os dispositivos mantenham-se sempre conectados. Este é o princípio das redes em malha sem fio, todos conectados a todos.

Segundo Tang e Tianfield (2006), as redes podem ser auto-organizadas conforme o modelo de referência OSI em duas camadas (TANG; TIANFIELD, 2006): **aplicação** e **rede**. O autor ainda enfatiza que, em alguns casos, a auto-organização pode ocorrer na camada de aplicação, exemplo disso seriam as redes overlays ². Já em outros casos, as redes físicas podem ser auto-organizadas na camada de rede.

As redes sem fio com capacidade de auto-organização se limitam a três categorias de acordo com Tang e Tianfield (2006): redes de sensores, redes *ad hoc* móveis e redes em malha sem fio. De acordo com Dressler (2006), em redes de computadores, em específico as redes do tipo *ad hoc*, a auto-organização é relativamente importante devido à interação espontânea de múltiplos componentes sobre conexões sem fio.

3.6 Projetos

Nesta seção são explanados alguns projetos que contemplam a área de redes autônômicas e auto-organizadas. Os projetos são iniciativas que estão em fase de desenvolvimento e testes, portanto destaca-se somente a descrição em nível conceitual das arquiteturas propostas em cada projeto. Entretanto é interessante ressaltar que a área de redes auto-organizadas está em constante evolução e mais iniciativas devem

²São redes que usam a infra-estrutura física já existente para comunicação ou para disponibilizar recursos na rede, um exemplo típico são as redes *peer-to-peer* P2P.

ser propostas para garantir um completo nível de auto-organização da rede.

3.6.1 ANA: Autonomic Network Architecture

Projeto ANA (JELGER et al., 2007) (TSCHUDIN et al., 2006) (ANA, 2006) é uma iniciativa fundada pela *European Foundation*, que possui várias empresas e universidades como parceiros tais como: ETH Zurich, NEC, Universidade de Oslo, Universidade de Waterloo, Universidade de Basel, Universidade de Lancaster, Fokus, Universidade de Liege, Universidade Pierre et Marie Curie, NKUA e Telecom Áustria. Tem como objetivo o desenvolvimento de uma meta-arquitetura para redes autônomicas.

Os principais objetivos do projeto são:

- **Objetivo Científico:** Neste objetivo os autores enfatizam a preocupação de identificar os fundamentos da computação autônomicas, que permite com que a rede possa ser escalável não somente em tamanho mas sim em funcionalidades também. Para isso alguns atributos Auto-x são desejáveis como a auto-otimização, auto-monitoramento, auto-reparo e auto-proteção.
- **Objetivo Tecnológico:** O objetivo tecnológico concerne em construir um *framework* que seja capaz de interconectar diferentes tipos de redes de forma autônoma e sem intervenção direta do administrador da rede. O conceito de auto-organização é desenvolvido considerando-se cada nodo individualmente dentro da rede.

A arquitetura permite a adaptação dinâmica e re-organização de tráfego de acordo com a necessidade do usuário. Além da adaptação dinâmica, a arquitetura ainda suporte nodos móveis e múltiplos domínios administrativos.

Na Figura 3.7 pode ser visualizado o modelo conceitual do projeto, subdividido em duas camadas principais. A arquitetura do nodo e a arquitetura da rede. Na arquitetura de cada nodo individual pode-se perceber a camada de aplicação, que provê serviços como www, email, ftp, DNS, peer-to-peer (kazzaa, emule) voip (skype), dentre outras aplicação suportadas. Ainda na arquitetura de um nodo é introduzido o conceito de compartimento. O conceito de compartimento é implementado conforme as regras operacionais e políticas administrativas para um dado contexto de comunicação. No qual define como um nodo entra e deixa a rede, quais os membros registrados, o modelo de autenticação e como o nodo comunica-se com

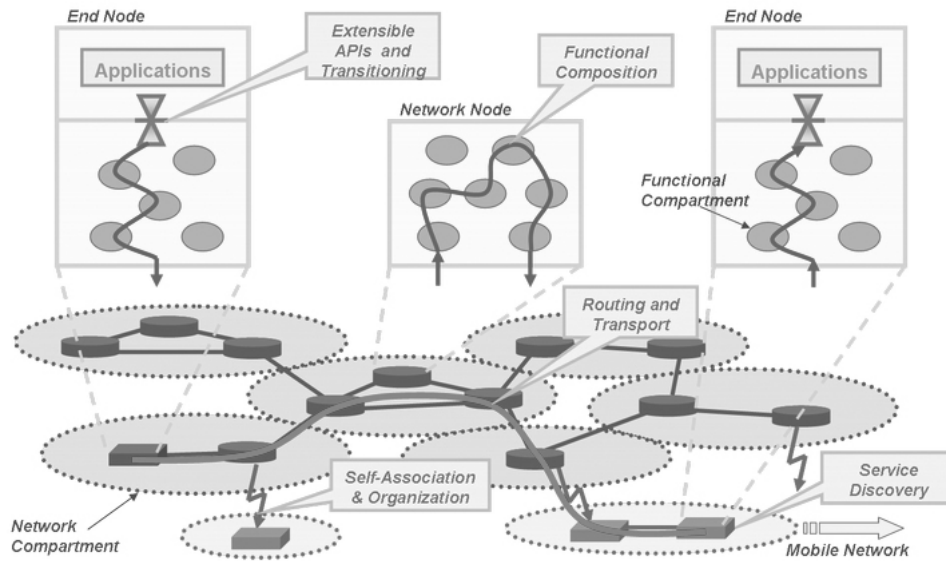


Figura 3.7: Modelo conceitual da arquitetura do projeto ANA (ANA, 2006)

outro membro da rede (roteamento, comunicação *peer-to-peer*, endereçamento). O conceito de compartimento configura o sistema de comunicação em unidades pequenas de gerenciamento; isso facilita a rede em ter uma visão local de cada nodo, a fim de prover uma auto-organização em nível global. O processo de auto-associação e organização tem a função de associar os novos nodos fixos ou não, que desejam entrar na rede. Este processo possui um alto nível de abstração para o usuário. A arquitetura da rede possui a propriedade de descoberta de serviços ou recursos, no qual abstraem uma lista de serviços que cada nodo pode prover, um exemplo seria um nodo com maior capacidade de armazenamento e outro com capacidade para impressão.

3.6.2 HAGGLE: An innovative Paradigm for Autonomic Opportunistic Communication

Haggle é um projeto iniciado pela *European Foundation* que consiste em projetar uma arquitetura para redes autônomicas que suporte a presença de conectividade intermitente, ou seja, que explore a comunicação oportunista.

Na Figura 3.8 pode ser observada a arquitetura de um nodo específico do projeto Haggle que provê todos os aspectos de comunicação, abrangendo desde da camada física, até a camada de aplicação. Observando-se da camada física, podem ser encontrados alguns dos métodos de rádio frequência suportados como: 802.11 e bluetooth. Além dos métodos de rádio frequência, é suportando também o padrão

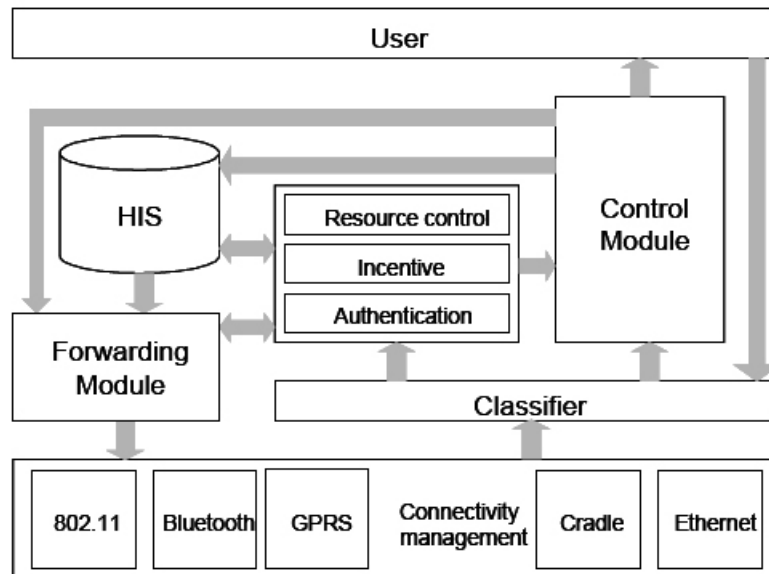


Figura 3.8: Arquitetura de um nó do projeto Hagggle (HAGGLE, 2006)

para redes cabeadas Ethernet 802.3. A arquitetura de um nó Hagggle é baseada na noção de HIS (*Hagggle Information Space*) contendo *Application Data Units* (ADUs) em cada dispositivo do usuário. O *Classifier* tem o papel de classificar a informação que chega de uma das camadas mais inferiores.

3.6.3 CASCADAS: Componentware for Autonomic, Situation-aware Communications and Dynamically Adaptable Services

O projeto Cascadas, iniciado em janeiro de 2006, foi fundado pela *European Commission* e tem como objetivo principal definir uma proposta geral de desenvolvimento de uma arquitetura para auto-organização para serviços de comunicação.

O projeto Cascadas utiliza o modelo de *Autonomic Communication Element - ACE*. Um ACE atua como entidade, que pode implementar uma forma distribuída de comunicação e serviços e também atuar como um ponto central de serviços (MANZALINI; ZAMBONELLI, 2006). Neste caso um modelo de ACE deve integrar todas as capacidades de auto-organização e deve ser a base para a implementação dos serviços de comunicação situados na camada de aplicação, como também a base de implementação para a camada de rede e a camada do nível de *middleware*, conforme observado na Figura 3.9.

Observando-se ainda a Figura 3.9 pode-se destacar as três camadas propostas

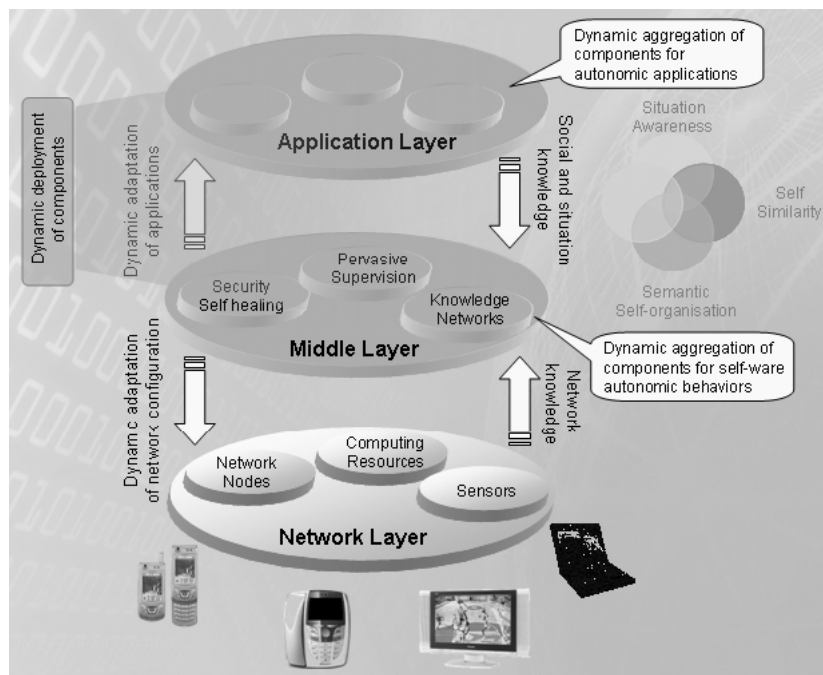


Figura 3.9: Visão geral da arquitetura do projeto CASCADAS (MANZALINI; ZAMBONELLI, 2006)

para a implementação do ACE. Na camada de aplicação, destaca-se o desenvolvimento dinâmico de componentes, juntamente com acoplamento de componentes para aplicações autônomicas. Na camada de *middleware* ou *middle* observa-se algumas capacidades como auto-cura (self-healing), supervisão pervasiva e conhecimento da rede, que são funções agregadas dinamicamente conforme o comportamento autônômico da rede. Na camada de rede, nota-se a integração dos diferentes dispositivos como: televisores, computadores pessoais e até sensores. Além disso, nota-se também a questão de compartilhamento de recursos computacionais.

Os autores consideram um cenário de redes dinâmicas e heterogêneas, possivelmente com interações com sensores e dispositivos conectados à *Internet* (MANZALINI; ZAMBONELLI, 2006).

Uma lista de mais projetos, envolvendo arquiteturas auto-organizadas pode ser acessada em (AUTONOMIC, 2007).

3.7 Considerações Finais

Este capítulo apresentou uma visão geral sobre auto-organização, mostrando as definições e as propriedades dos sistemas auto-organizados. Optou-se também em

dividir os métodos de auto-organização encontrados na literatura, em abordagens *top-down* e *botton-up*, enfatizando as primeiras, mais precisamente redes autonômicas, o foco principal de desenvolvimento da abordagem de auto-organização proposta para as redes em malha sem fio. Analisando a literatura pode-se perceber que não existem muitos projetos e abordagens voltados a redes autonômicas, pois a área está em constante crescimento e realmente há uma carência deste tipo de abordagem para redes sem fio. Os aspectos discutidos sobre redes autonômicas restringem-se às capacidades de Auto-x (auto-configuração, auto-otimização, auto-cura e auto-proteção). Estas capacidades são usadas para deixar a rede auto-organizada. O próximo capítulo discute sobre roteamento em redes em malha sem fio, enfatizando os protocolos de roteamento OLSR e AODV, como base para o desenvolvimento da abordagem de auto-organização proposta neste trabalho.

4 *Roteamento em Redes Em Malha Sem Fio*

4.1 Introdução

Os protocolos de roteamento em uma rede de computadores têm como principal função manter a conectividade entre os nodos na rede. Este passo é muito importante ou mesmo essencial, em uma rede de computadores sem fio (TANENBAUM, 2003). No entanto alguns desafios são encontrados, tais como: freqüente mudanças na topologia, limitação de largura de banda, capacidade variada dos enlaces e mudanças de conectividade. Com a concepção das RMSF, alguns desses problemas foram minimizados, mas em compensação a dificuldade em se adaptar os protocolos de roteamento já existentes, se torna um problema eminente. As RMSF, por serem baseadas nas redes *ad hoc*, possuem suas próprias peculiaridades que devem ser levadas em consideração na hora da escolha do protocolo de roteamento. Com isso, observou-se a necessidade de estudar os protocolos de roteamento já existentes para as redes sem fio *ad hoc*, a fim de alcançar uma melhor performance nas RMSF. Nota-se que o maior desafio em RMSF é projetar protocolos de roteamento eficientes, que dinamicamente encontram as rotas de menor custo e que também obtenham um máximo proveito das características das redes em malha. Este desafio tem motivado inúmeros pesquisadores, tais como (ABELÉM et al., 2007) (LEE; WARD, 2006) (CAMPISTA et al., 2007) (SUBRAMANIAN et al., 2007) (SUBRAMANIAN et al., 2006) a direcionarem suas contribuições nesta área. Porém, ainda observa-se uma grande carência de abordagens, especializadas em adaptar os protocolos de roteamento já existentes para as peculiaridades das RMSF. Como mencionado no capítulo 2, a maioria dos protocolos de roteamento para as RMSF são derivados das redes sem fio *ad hoc*; sendo assim, é inerente dar uma visão geral sobre alguns conceitos relacionados ao roteamento em redes sem fio *ad hoc*. O emprego de protocolos de roteamento *ad hoc* nas RMSF representa uma solução mais simples, pois assim como as redes *ad hoc*, as RMSF também possuem comunicação múltiplos-saltos (CAMPISTA et

al., 2007). Sumariando, as RMSF usam a comunicação *ad hoc* para formar o seu *backbone* entre seus nodos roteadores.

Neste capítulo são apresentados os protocolos de roteamento OLSR e AODV utilizados como base para a proposta de auto-organização, juntamente com uma breve explanação sobre a classificação dos protocolos de roteamento para redes em malha sem fio, subdivididos conforme a característica de roteamento.

4.2 Classificação dos Protocolos de Roteamento

Nesta seção é apresentada a classificação dos protocolos de roteamento conforme o ponto de vista de alguns autores, juntamente com a descrição específica do funcionamento dos protocolos de roteamento AODV e OLSR, os quais foram utilizados nesta dissertação.

O principal objetivo de um protocolo de roteamento é garantir que o nodo origem comunique-se com o nodo destino, com o menor custo possível. Observa-se que entre as técnicas tradicionais que garantem o menor custo, destacam-se as baseadas em estado de enlace (*link-state*) e a vetor de distância (*distance-vector*).

De acordo com (TANENBAUM, 2003), a técnica de estado de enlace possui cinco fases e é estabelecida em cada nodo roteador. Nota-se que cada nodo roteador deve proceder desta forma: primeiro, o nodo roteador deve conhecer seus nodos vizinhos e posteriormente guardar informações do mesmo; segundo, cada nodo roteador deve medir o tempo de retardo ou o custo de cada um dos seus nodos vizinhos; terceiro, cada nodo roteador pode criar um pacote com intuito de enviar a informação que foi colhida através do primeiro passo; quarto, enviar este pacote para todos os outros nodos roteadores; por fim, no quinto e último passo, cada nodo roteador deve calcular o menor custo possível até os outros nodos roteadores. Seguindo-se estes passos, os nodos roteadores conterão informações relevantes para o cálculo do menor custo possível para chegar ao nodo roteador destino. O menor custo possível pode ser calculado utilizando o algoritmo de Dijkstra (DIJKSTRA, 1960).

Na abordagem vetor de distância, os nodos roteadores mantêm uma tabela (ou seja, um vetor) que tem por objetivo fornecer a melhor distância conhecida entre o nodo roteador origem e o seu nodo destino. Nota-se que a tabela é atualizada através de troca de mensagens entre seus nodos roteadores vizinhos. As mensagens de atualização possuem o cálculo da estimativa do custo da comunicação entre cada nodo roteador conhecido. (TANENBAUM, 2003)

Com o intuito de dar uma visão geral sobre a classificação dos protocolos de roteamento para redes sem fio, diversos autores como, (ABOLHASAN et al., 2004) (ROYER; TOH, 1999) (MAUVE et al., 2001) (TANENBAUM, 2003) classificam os protocolos de roteamento em pró-ativos, reativos e híbridos, os quais serão explanados nas seções seguintes.

4.2.1 Protocolos de Roteamento Pró-Ativos

Nos protocolos pró-ativos ou *table-driven*, cada nodo mantém as rotas de todos os possíveis nodos destinos armazenados em uma tabela, de forma que quando um nodo deseja enviar um pacote ao nodo-destino, este consulta a tabela para saber qual a rota disponível e de forma imediata envia o pacote ao seu destino. Para isso, cada nodo deve manter a tabela sempre atualizada e mandar a sua tabela para os demais nodos, de forma que todos os nodos possuem as atualizações periódicas das rotas. Dentre alguns exemplos de protocolos pró-ativos pode-se citar OLSR (*Optimized Link State Routing*), DSDV (*Destination Sequenced Distance Vector*), WRP (*Wireless Routing Protocol*), CGSR (*Cluster-Head Gateway Switch Routing*), FSR (*Fisheye State Routing*).

Neste contexto é apresentado o funcionamento básico do protocolo OLSR, por ser usado também como base para a abordagem de auto-organização proposta neste trabalho.

4.2.2 *Optimised link state routing (OLSR)*

O protocolo de roteamento OLSR (JACQUET et al., 2001) (CLAUSEN et al., 2003) (ABOLHASAN et al., 2004) trabalha de forma pró-ativa trocando informações entre os demais nodos da rede, com o objetivo de atualizar sua tabela de roteamento. Com essa forma de troca de informações o protocolo possui uma visão global da rede, ou seja, cada nodo possui o conhecimento completo da rede, através do conceito denominado de *multipoints relay* (MPR). Os nodos que foram escolhidos para serem nodos MPR, são responsáveis pela difusão controlada de mensagem de HELLO (que serve para identificar se o nodo está ativo na rede). Isso minimiza a sobrecarga de mensagens na rede, pois cada MPR tem um limite de difusão de mensagens.

Na Figura 4.1 podem ser observados os nodos MPR, representados pelos quadrados, e os nodos que receberão a difusão de mensagens dos MPR's, representados pelos círculos.

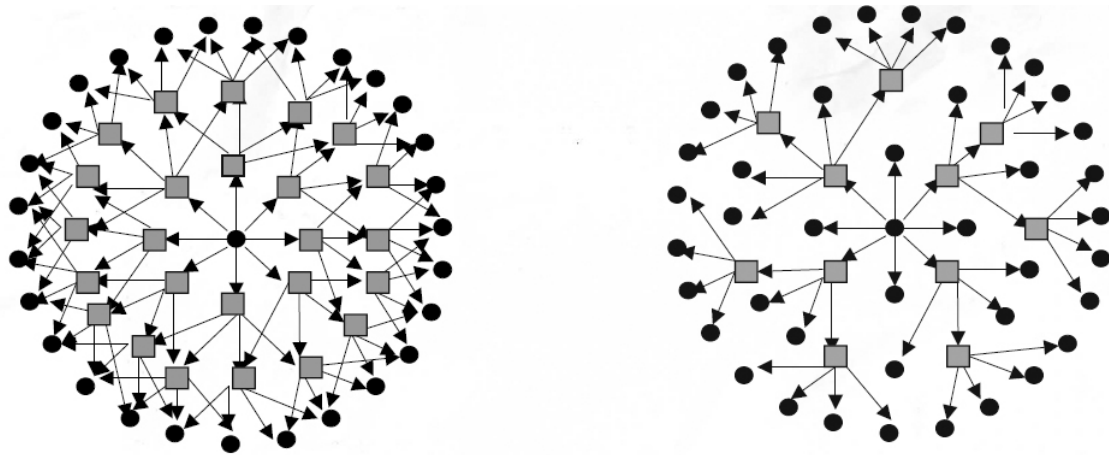


Figura 4.1: Exemplo de difusão de mensagens, utilizando a técnica de MPR's no protocolo OLSR

Descrição dos Parâmetros do Protocolo OLSR

Nesta seção foi descrito cada parâmetro do protocolo OLSR juntamente com o tipo de variável (inteira, ponto flutuante ou booleana) e suas respectivas funcionalidades.

- **Hello_Interval** (*ponto flutuante*) : Neste atributo é especificado o intervalo de tempo entre um pacote do tipo Hello.
- **Willingness** (*inteira*): Este atributo possui a capacidade de encaminhar tráfego dele mesmo ou para outros nós.
- **TC_interval** (*ponto flutuante*) : Especifica o tempo entre uma mensagem de controle TC (*Topology Control*). As mensagens de TC são geradas pelos nodos MPR e são usadas para auxiliar no cálculo da tabela de roteamento.
- **Neighbor_hold_time** (*ponto flutuante*): Este parâmetro especifica o tempo em que o enlace vai expirar. Ele é configurado para ser $3 * \text{Hello Interval}$.
- **Top_hold_time** (*ponto flutuante*): Este atributo especifica o tempo de expiração das entradas da tabela de roteamento. Ele é configurado por $3 * \text{TC Interval}$.

4.2.3 Protocolos de Roteamento Reativos

Nos protocolos de roteamento reativos ou por demanda, constata-se que quando um nodo origem deseja enviar um pacote para o nodo destino, antes, o pacote é armazenado em uma fila denominada *buffer*, que serve para aguardar até que o processo de descoberta de rota consiga encontrar a rota desejada e assim encaminhar o pacote armazenado. Com o estabelecimento da rota, os protocolos reativos mantêm-na até o momento em que o nodo não seja mais alcançado ou, por ventura, o nodo não pertença mais à rede. Existe uma grande quantidade de protocolos de roteamento reativos na literatura, alguns exemplos são: DSR(*Dynamic Source Routing*), AODV(*Ad Hoc On-Demand Distance Vector Routing*), TORA (*Temporally-Ordered Routing Algorithm*).

4.2.4 *Ad hoc on-demand distance vector (AODV)*

O protocolo AODV (ROYER; TOH, 1999)(ABOLHASAN et al., 2004) é basicamente uma combinação dos protocolos DSR e DSDV, pois utiliza a forma de descoberta de enlaces e manutenção das rotas, encontradas no DSR em conjunto com o roteamento por saltos e número de seqüência utilizados pelo DSDV.

O mecanismo de requisição de rota do AODV é realizado através de *broadcast* de pacotes RREQ (*Route Request*) na rede. Por exemplo, quando um Nodo Origem (NO), deseja constituir uma rota com o Nodo Destino (ND), passando por vários Nodos Intermediários (NI), é iniciado um processo de *broadcast* de pacotes RREQ até chegar ao ND. Durante o processo de descoberta da rota, os nodos que receberam os pacotes RREQ incluem entradas temporárias em suas tabelas de roteamento, registrando assim qual o NO que enviou a requisição RREQ.

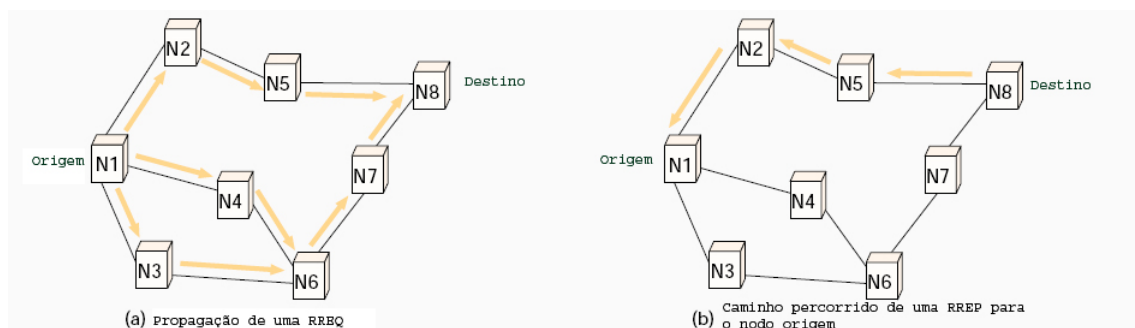


Figura 4.2: Exemplo da descoberta de rota e propagação inversa do protocolo AODV (ROYER; TOH, 1999)

Na Figura 4.2 (a) pode ser exemplificado o mecanismo de descoberta de rotas

utilizado pelo AODV. Quando o NO deseja comunicar-se com o ND, como o ND não está ao seu alcance, o NO deverá propagar mensagens de RREQ através dos NI, a fim de encontrar o ND. Para isso cada nodo transmite mensagens de *Hello* periódicas aos seus nodos vizinhos que tenham rotas que passam por eles. Isso serve para que os nodos vizinhos possam manter-se atualizados sobre a existência ou não de uma rota. Se a mensagem de *Hello* não for recebida em um período de tempo pré-estabelecido, conclui-se que ocorreu algum problema em algum enlace que pertencia à rota isso a torna inválida. Quando há problemas com a rota, o NO deve requisitar novamente o processo de descoberta de enlaces RREQ para selecionar uma nova rota. Observando-se a Figura 4.2 (b), é demonstrado o procedimento de propagação inversa, ou seja o caminho percorrido pelo ND para o NO. Um RREQ inclui o endereço do NO e ND, e um identificador, que é usado como identificador do último número seqüencial do ND, como também, o número seqüencial do NO. Os números seqüências são usados para prevenir problemas de *loops* no roteamento e servem também para garantir que os pacotes de atualizações das rotas sejam os mais atuais possíveis.

Desempenho depende dos Valores dos Parâmetros do protocolo AODV

O desempenho do protocolo AODV é diretamente influenciado pela escolha dos valores dos parâmetros. Seguindo se essa premissa observa-se que a escolha do melhor valor para o parâmetro de roteamento é um problema que pode ser tratado através de técnicas de otimização. A melhor escolha resulta em um melhor desempenho do protocolo diante de situações adversas. Alguns desses parâmetros influenciam drasticamente o desempenho do protocolo na rede são:

1. **Intervalo de mensagens de Hello:** Neste parâmetro um nodo qualquer pode manter informações sobre a conectividade de seus nodos vizinhos através do envio periódicos de mensagens de Hello. Neste caso o HELLO_INTERVAL é um dos parâmetros mais importantes do protocolo de roteamento AODV, pois é nele que são configurados os tempos para garantir a conectividade entre os nodos vizinhos. Então, se o intervalo de tempo é muito pequeno, é gerada uma grande quantia de mensagens de controle, impactando em uma sobrecarga de mensagens na rede. Agora, se o intervalo de tempo é muito grande, observa-se a dificuldade em se detectar se algum nodo vizinho deixou de responder à mensagem, isso impacta diretamente na forma de descoberta mais rápida de uma falha de conectividade de algum nodo vizinho. Sumariando, O HELLO_INTERVAL, deve ser configurado de forma a reduzir a sobrecarga de

mensagens de controle e identificar de forma satisfatória a quebra de conectividade de algum nodo vizinho. É válido ressaltar também que reduzindo o *broadcast* desnecessário de mensagens de controle, reduz-se o tráfego dessas mensagens, aumentando assim a vazão da rede e em consequência diminuindo a perda de pacotes na rede.

2. **Controle de *Broadcasting* através da técnica *Expanding Ring Search*:**

O controle de *broadcasting* pode ser feito através da técnica denominada *Expanding Ring Search*. Nesta técnica, um nodo origem qualquer utiliza um TTL (Time-To-Live) que está encapsulado no cabeçalho do RREQ. Este cabeçalho é denominado TTL_START = 1, que por padrão recebe o valor 1. Neste caso o TTL_START é utilizado para verificar que se o RREQ esgotou o tempo de espera de um RREP (um RREP é a resposta de uma rota), o nodo origem envia novamente um pacote RREQ, incrementando o TTL_INCREMENT para 2. Isso evita que o nodo origem envie duplicatas desnecessária de pacotes RREQ que podem sobrecarregar a rede. Entretanto, essa técnica pode afetar no atraso da descoberta de rotas, ou seja, a técnica descrita pode prevenir duplicatas desnecessárias de pacotes RREQ mas acaba impactando no atraso da descoberta de uma rota ao nodo destino.

3. **Mantendo a conectividade local:** Cada nodo na rede pode manter uma tabela de conectividade local, ou seja, nesta tabela cada nodo mantém atualizados quais são os nodos vizinhos que estão ativos em um determinado instante de tempo. A troca de mensagens é feita através de pacotes *Hello*, conforme descrito anteriormente. A demora em atualizar a tabela de conectividade impacta diretamente na escolha de rotas confiáveis, pois o nodo pode escolher uma rota onde o nodo pode ter deixado de funcionar.

Descrição dos Parâmetros do Protocolo AODV

Nesta seção é descrito cada parâmetro, o tipo de variável (inteira, ponto flutuante ou booleana) e suas respectivas funcionalidades.

- **route_request_retries (inteira):** É um parâmetro que especifica o número máximo de tempo que um nodo tentaria novamente descobrir uma rota através de *broadcasting* para outro pedido de rota.
- **route_request_rate (inteira):** Especifica a taxa limite de geração de pacotes de requisição de rota. Neste caso, o nodo fica limitado a não originar

mensagens de requisição de rota maiores que o limite imposto pelo ‘Route Request Rate Limit’.

- **grat_route_reply_flag** (*booleana*): Este parâmetro indica o estado que uma resposta de rota deve ser enviado para o nodo especificado pelo endereço IP de destino.
- **dest_only_flag** (*booleana*): Quando configurado para verdadeiro, indica que somente o nodo destino pode responder ao pedido de rota (RREP). Neste caso, se todos os nodos intermediários não podem responder um pedido de rota solicitado.
- **ack_required** (*booleana*): Quando configurado para verdadeiro, a cada requisição (RREP), uma RREP de rota com ACK (*acknowledgement*) deve ser enviada para o nodo que enviou a requisição.

Estes parâmetros citados fazem parte da classe de parâmetros de descoberta de rotas. Os próximos parâmetros são voltados ao controle do *broadcasting* de mensagens na rede.

- **active_route_timeout** (*ponto flutuante*): Este parâmetro configura o tempo de vida de uma entrada de rota na tabela de roteamento.
- **hello_interval** (*ponto flutuante*): Este atributo é usado para calcular o tempo antes que o nodo determine que a conectividade com o nodo vizinho está perdida. Se o nodo não recebe um pacote de ‘Hello’ ou qualquer outro pacote de um nodo vizinho dentro de um determinado período de tempo, configurado por (4.1), este assume que a conexão para o nodo vizinho está comprometido.

$$\text{AllowedHelloLoss} * \text{HelloInterval}. \quad (4.1)$$

- **allowed_hello_loss** (*inteira*): Este atributo é responsável por configurar o número de pacotes de ‘Hello’ que podem ser perdidos, enquanto é iniciado o processo de *broadcasting* de mensagens de ‘Hello’. Se uma mensagem de ‘Hello’ não for recebida de um nodo vizinho com (4.1), então a conectividade para o nodo vizinho está comprometida.

- **net_diameter** (*inteira*): Este atributo tem como objetivo setar o número máximo de saltos (*hops*) entre dois nodos.
- **node_traversal_time** (*ponto flutuante*): O “Node Traversal Time” é uma estimativa da média de um salto(*hop*) por pacote. Nesta estimativa deve ser incluído o tempo de atraso, tempos de interrupções de processamento e tempos de transferência.
- **route_error_rate** (*inteira*): Este parâmetro limita a taxa na qual um nodo gera mensagens de erros de rotas. Neste caso, o nodo não deve gerar mensagens de erro por segundo que sejam maiores que “Route Error Rate Limit”.
- **timeout_buffer** (*inteira*): Este parâmetro é responsável por prover um *buffer* para o intervalo de recebimento de uma resposta de rota para um pedido de rota, no qual foi enviado. Para desativar o buffer é só configurar o “Timeout Buffer” = 0. Usa-se este atributo para calcular o *ring traversal time*, conforme observado em (4.2);

$$RingTraversalTime = 2 * NodeTraversalTime * (TTL_Value + TimeoutBuffer). \quad (4.2)$$

- **pkt_queue_size** (*inteira*): Número de pacotes em fila; este atributo configura o máximo de pacotes que podem esperar na fila. O seu valor padrão é setado para -1 ou infinito.
- **local_repair** (*booleana*): Se este atributo é configurado para ser verdadeiro, o nodo tentará reparar um possível problema na rota ativa.

Os parâmetros seguintes são usados para controle do tempo-de-vida (TTL) de um determinado pacote na rede.

- **ttl_start** (*inteira*): Inicia a disseminação de um pacote RREQ (*Route Request*). O nodo originalmente usa um TTL(*Time-to-live*) = “TTL Start” para a pesquisa.

- **ttl_increment** (*inteira*): Enquanto uma requisição de rota não recebe uma resposta de rota em um tempo pré-estabelecido, o nodo que originou o pedido de rota requisita novamente um pedido, com o valor do TTL incrementado pelo atributo “TTL Increment”. Isso aumenta o tempo de vida do pacote até chegar ao seu nodo destino.
- **ttl_threshold** (*inteira*): Neste parâmetro o valor do TTL é incrementado para cada timeout de um pedido de requisição de rota, sem receber uma resposta de rota válida. O valor do TTL é incrementado de acordo com o TTL *Threshold* máximo. Além do *Threshold*, o valor do TTL é configurado também de acordo com o atributo *Net Diameter* para cada tentativa de envio de uma requisição de rota.
- **local_add_ttl** (*inteira*): Quando um nodo decide executar um reparo local, este atributo é usado para calcular o TTL, que será colocado na mensagem para um novo pedido de rota. Este valor é calculado utilizando (4.3): o último *hop count* conhecido para o nodo destino.

$$\max(MIN_REPAIR_TTL, 0.5 * Hops) + Local_Add_TTL \quad (4.3)$$

4.3 Visão Geral sobre Protocolos de Roteamento para Redes em Malha Sem Fio

Existem poucos protocolos de roteamento desenvolvidos para redes em malha sem fio. Dentre eles pode-se citar: o MIT (ScrRR) (AGUAYO et al., 2004), (*Mesh-Networks Scalable Routing*)(MESHNETWORKS, 2007) (MSR) e finalmente o *Hybrid Wireless Mesh Protocol* (HWMP) (BAHR, 2007). O ScrRR é uma variação do protocolo reativo DSR usando como métrica o tempo de transmissão esperado, ao invés do número de saltos utilizado pelo DSR. Já o MSR é um protocolo híbrido que suporta alta mobilidade dos usuários e possui a capacidade de adaptar-se em condições adversas do ambiente. Por fim o HWMP é um protocolo baseado no protocolo AODV e combina as melhores características do roteamento por demanda com o roteamento pró-ativo. Este protocolo ainda usa a métrica baseada em *airtime*, que calcula se o canal está disponível para transmissão ou não, em um enlace particular.

Uma lista completa dos protocolos de roteamento para redes sem fio, subdivididos conforme a técnica de roteamento, pode ser observada em (GRADES, 2005)

(TAMASHIRO, 2007). Informações adicionais sobre roteamento em redes em malha sem fio podem ser consultados em (WAHARTE et al., 2006).

4.4 Algumas Considerações

Em função das poucas opções de protocolos de roteamento específicos para redes em malha sem fio, observou-se que muitos protocolos utilizados em redes em malha sem fio são adaptações dos protocolos de roteamento já utilizados em redes do tipo *ad hoc*. Neste capítulo foi explanado de forma conceitual os protocolos de roteamento OLSR e AODV, sua classificação conforme as técnicas de roteamento. Entretanto, é interessante o entendimento do mecanismo de funcionamento desses protocolos, juntamente com a descrição dos seus parâmetros de roteamento, os quais foram utilizados para garantia de melhor desempenho utilizando as capacidades de auto-otimização e auto-configuração propostas nesta dissertação, pois essas capacidades são diretamente acopladas nos protocolos OLSR e AODV. O próximo capítulo detalha a arquitetura de auto-organização proposta para redes em malha sem fio, conforme o modelo conceitual proposto.

5 Proposta de uma Arquitetura para Auto-Organização

Este capítulo descreve a arquitetura para auto-organização proposta juntamente com uma descrição detalhada sobre os módulos que a compõem. Foram enfatizadas as capacidades de auto-otimização e auto-configuração aplicadas aos protocolos OLSR e AODV, com o intuito de aumentar o desempenho dos mesmos, quando adaptados às redes em malha sem fio.

5.1 Introdução

Conforme apresentado nos capítulos anteriores, notou-se uma crescente tendência em se utilizar redes sem fio cada vez mais robustas e que possuem muitos dispositivos interconectados, provendo serviços aos seus usuários. Com essa quantidade e heterogeneidade de dispositivos, o processo de manutenção e gerenciamento da rede torna-se dispendioso e difícil de ser feito pelo seu administrador. Observou-se então a necessidade de montar estratégias que facilitem o trabalho do administrador da rede. Em redes do tipo faixa larga sem fio, ou seja, redes que cobrem uma área de cobertura grande, por exemplo, as RMSF, há que se levar em conta o fator da auto-organização, que é a baixa intervenção do administrador para sua manutenção e configuração. O ideal nessas redes é que elas possam se auto-organizar conforme mudanças em seu ambiente. Um exemplo disso seria quando a quantidade de nodos que entram na rede impactam o seu desempenho, pois quanto maior o número de dispositivos entrando e saindo, maior será o impacto destes no desempenho e no trabalho do administrador em configurar manualmente cada nodo. Isso também afeta desde a largura da banda até o tráfego dos pacotes dentro da rede. Observa-se ainda que em determinados momentos alguns dispositivos essenciais para o funcionamento da rede podem falhar, deixando uma área não comunicável. Pode-se também citar os aspectos de segurança, pois como o sinal da rede sem fio propaga-se pelo ar, pos-

sibilita a qualquer pessoa mal intencionada gerar ataques que possam prejudicar o desempenho da mesma. Outro problema seria o da garantia de conectividade entre os nodos-roteadores e clientes. O que garante essa conectividade são os protocolos de roteamento, como pôde ser visto no Capítulo 4 desta dissertação. No mesmo capítulo observou-se que os protocolos atuais para redes em malha sem fio não tratam de aspectos relacionados à forma de auto-organizar as suas funções, o que impactaria em uma melhor performance, tolerância a falhas e segurança aplicado as RMSF. Vendo que estes protocolos não focam nesses aspectos, propõe-se neste trabalho a necessidade de acoplar as capacidades de auto-x (configuração, otimização, cura e proteção) discutidas no Capítulo 3 e adaptá-las aos protocolos de roteamento já existentes, por exemplo, OLSR e AODV, mas não limitados aos mesmos.

Para adaptar as capacidades de auto-x propostas deve-se analisar os protocolos de roteamento de forma específica, pois cada protocolo possui seus próprios parâmetros e suas próprias técnicas de roteamento.

Neste capítulo optou-se em descrever a arquitetura de auto-organização proposta e também enfatizar a arquitetura em nível conceitual. Neste último destaca-se a preocupação em pensar de forma mais ampla, ou seja, mostrar como é feita toda a interação entre os componentes envolvidos, a fim de enfatizar o completo funcionamento da arquitetura em uma rede em malha sem fio, conforme sugerido.

5.2 Arquitetura de Auto-Organização em Nível Conceitual

Nesta seção é mostrada uma visão geral da arquitetura proposta de acordo com o modelo em camadas TCP/IP.

Como observado na Figura 5.1, a arquitetura em nível conceitual trabalha em duas camadas: **aplicação** e **rede**.

Na camada de aplicação encontra-se a plataforma dos agentes, que é responsável pelo processo de aquisição de informações da rede. Estas informações são compostas: da coleta do tamanho da rede, que é uma lista da quantidade de dispositivos pertencentes à mesma, enviada de tempos em tempos para todos os nodos-roteadores; e também do monitoramento de cada nodo roteador (vazão, perda de pacotes, nodo ativo ou não).

Na camada de rede é situado o processo de auto-organização, acoplado ao pro-

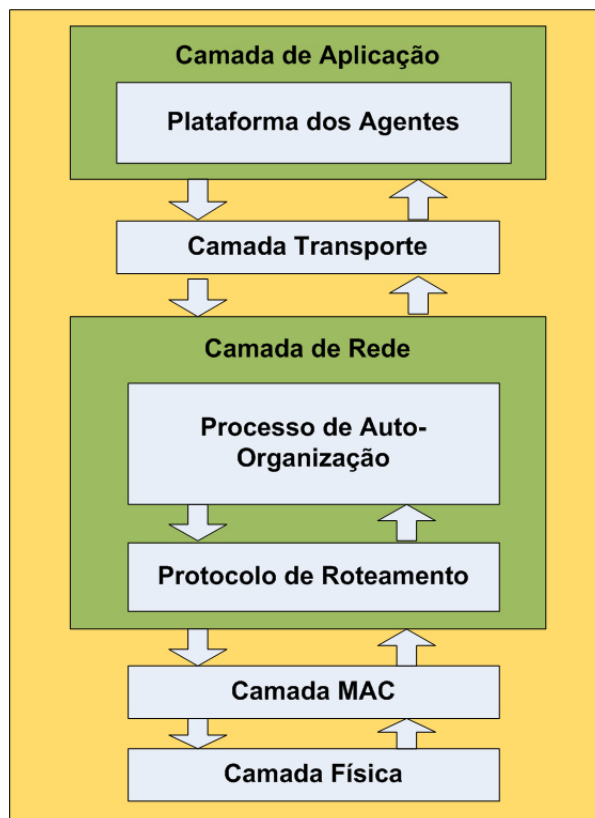


Figura 5.1: Visão geral da arquitetura proposta, conforme modelo em camadas TCP/IP

toloco de roteamento escolhido.

Detalhes da implementação do processo de auto-organização no simulador, pode ser visto no Capítulo 6 desta dissertação.

5.2.1 Arquitetura de Auto-Organização Aplicada às Redes em Malha Sem Fio

A arquitetura para redes em malha sem fio foi projetada para adaptar-se ao contexto dos nodos roteadores em malha. A Figura 5.2 mostra o uso da arquitetura de auto-organização em um cenário típico de redes em malha sem fio, na qual podem ser visualizados os nodos *gateways* conectados às redes *ethernet* cabeadas, com o intuito de prover acesso à Internet na malha de roteadores. Na malha de roteadores é observada a constituição do *backbone* utilizando o padrão 802.11g. A comunicação entre os nodos clientes e nodos roteadores é feita através do padrão 802.11g. Este processo é classificado como rede em malha sem fio únicos rádios, pois o *backbone* e a ligação entre os clientes é feita utilizando a mesma interface de rede (NIC). Na

camada dos agentes foi desenvolvido dois tipos de agentes que são: NS (*Network-Size*) e NM (*Network-Monitoring*). No processo de auto-organização optou-se em integrar as capacidade de Auto-x (configuração, otimização, cura e proteção) em cada nodo-roteador em malha. Os nodos clientes estão associados a determinados nodos roteadores em malha, conforme a área de cobertura. A definição da RMSF pode ser representada através de um grafo unidirecional $G = (R,L)$, onde R é o conjunto de roteadores em malha, e L é o conjunto de enlaces que representam a ligação entre quaisquer dois nodos roteadores x,y. Alguns nodos x podem manter uma conexão com o nodo *gateway* g. O nodo x pode conectar-se à internet via nodo *gateway* g.

Cada nodo cliente pode estabelecer ligações bidirecionais a um determinado nodo roteador x, conforme pode ser observado na Figura 5.2

Os nodos clientes podem ser tanto móveis, quanto fixos e diferenciam-se pelo tipo do dispositivo: classe de dispositivos móveis (PDA's, telefones celulares e laptops) e classe de dispositivos fixos (computadores desktop, servidores e nodos-roteadores).

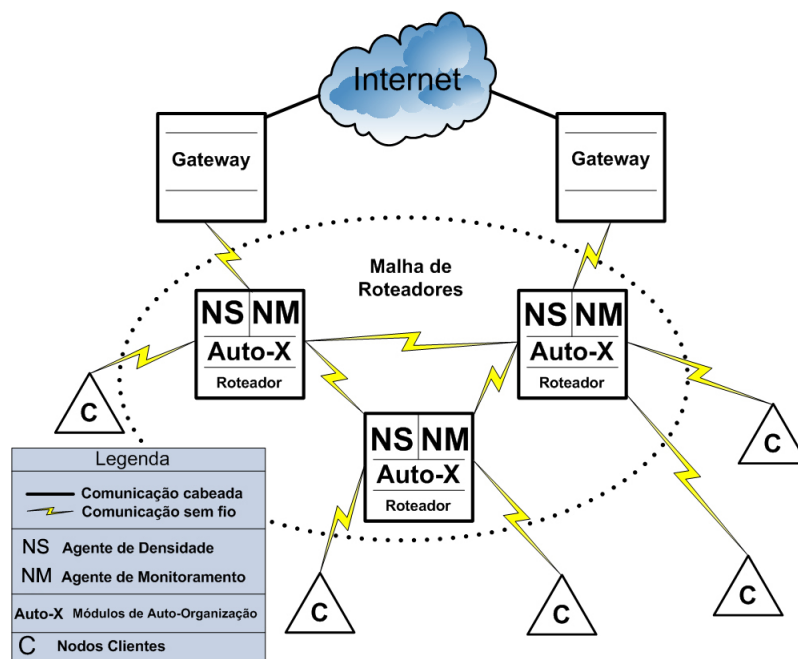


Figura 5.2: Visão geral do modelo de auto-organização proposto, adaptado às redes em malha sem fio

Para melhor explicar a lógica de interação entre os módulos envolvidos neste trabalho, desenvolveu-se a arquitetura baseada em camadas. As abordagens baseadas em camadas garantem flexibilidade e simplicidade no desenvolvimento, no sentido de que todas as funções envolvidas são bem definidas e tratadas de forma independente. Isso se torna uma vantagem com relação à detecção de falhas isoladas, como

pode ser visto na Figura 5.3 . Nesta mesma Figura observa-se que as capacidade de auto-cura e auto-proteção que estão em tons de cinza escura, não foram implementadas no escopo deste trabalho, somente foram implementadas as capacidade de auto-otimização e auto-configuração juntamente com o NS_Agent.

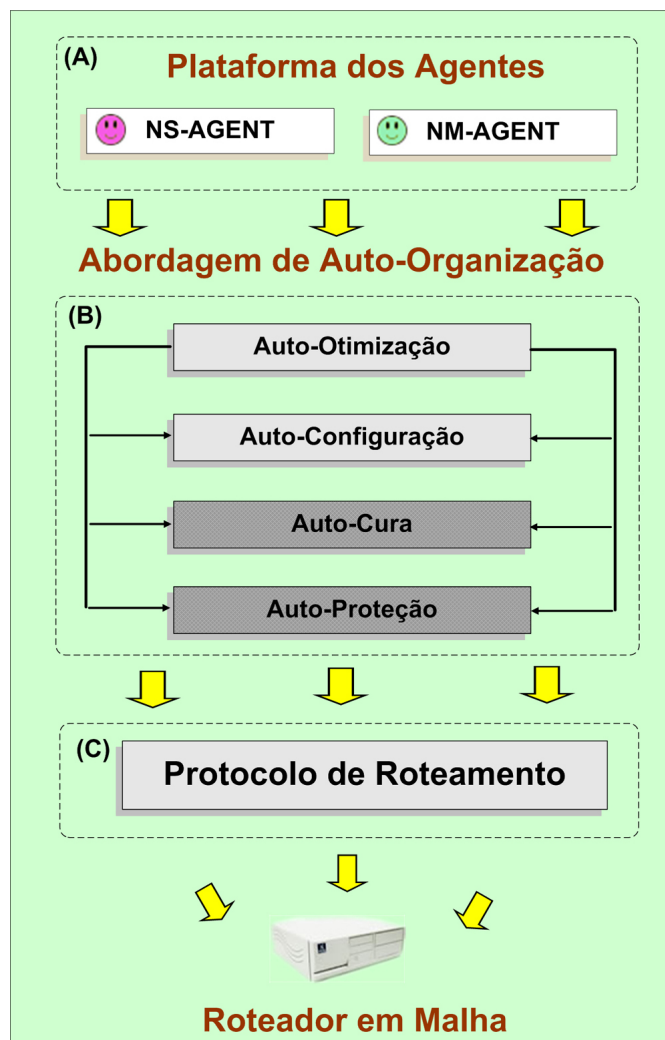


Figura 5.3: Visão da arquitetura para auto-organização para um nodo-roteador em malha

Os detalhes de cada camada são descritos nas seções seguintes e destacam-se de forma conceitual na interação entre todas as camadas propostas.

5.2.2 Plataforma dos Agentes

Com o intuito de fazer o monitoramento e controle da rede, foram utilizados agentes de software (WEISS, 1999) para o gerenciamento do tamanho da rede e também do monitoramento dos nodos roteadores. Os agentes de software são consider-

ados uma alternativa à convencional arquitetura Cliente/Servidor, diferenciando-se pela interação local com mobilidade de código (HARRISON et al., 1997; BIESZCZAD et al., 1998; RAIBULET; DEMARTINI, 2000).

Agentes para Monitoramento e Controle do Tamanho da Rede

Como visto no Capítulo 3, o monitoramento é um fator importante para o conhecimento do sistema a ser gerenciado. Observando-se esta importância foi proposta a tecnologia de agentes de software para fazer o monitoramento dos nodos roteadores em malha dentro da rede. Além do monitoramento, os agentes são responsáveis por manter atualizada a lista de nodos da rede, ou seja, o número de equipamentos na rede conectados em determinados períodos de tempo. Essa informação é relevante para que o processo de auto-configuração através do tamanho da rede, possa configurar os parâmetros dos protocolos de roteamento de forma dinâmica. Optou-se em utilizar agentes de software devido às características encontradas neste tipo de abordagem.

Algumas características dos agentes são:

- **Tolerância a Falhas:** Têm habilidades de reagir dinamicamente a situações adversas;
- **Interação Autônoma e Assíncrona:** São capazes de se comportar de forma autônoma com outros agentes;
- **Adaptação Dinâmica:** São capazes de se adaptar a mudanças no ambiente;

Dentre as muitas características encontradas no comportamento dos agentes de software, essas são as mais desejáveis no contexto deste trabalho. Outro fator importante para a escolha dos agentes de software como abordagem para monitoramento distribuído, é que abordagens como SNMP possuem como características modelos centralizados que não são bem aplicados as RMSF, deixando assim um ponto único de falhas, acarretando problemas nos módulos que necessitam dessa funcionalidade.

Os agentes que compõem a camada aplicação possuem um papel específico na arquitetura e são descritos a seguir:

- **(A) NS-AGENTS:** são agentes estáticos criados nos nodos roteadores e possuem como função manter os mesmos informados sobre o tamanho da rede.

Cada nodo roteador através do agente NS possui uma lista atualizada da quantidade de nodos conectados (clientes fixos, móveis, nodos roteadores e nodos *gateways*) e a enviam periodicamente para os seus nodos roteadores vizinhos, a fim de manter esta informação atualizada. O controle do tamanho da rede possui três escalas: pequena (SMALL), média (MEDIUM), grande (LARGE) (vide seção 5.2.5). Esta informação é obtida contando-se a quantidade de nodos que estão associados em um determinado nodo-roteador local e depois é enviada aos demais roteadores, com a finalidade de montar uma somatória de todos os nodos clientes pertencentes à rede.

- **(A) NM-AGENTS:** são agentes estáticos criados nos nodos-roteadores e a cada pedido de associação de um novo nodo roteador, o agente inicia o processo de aquisição de informações: endereço (IP e MAC), se o nodo está ativo ou não, e, o caminho com melhor vazão e menor perda de pacotes. Estas informações são gravadas em uma tabela que é compartilhada com os demais nodos roteadores, para que os mesmos possam ter um conhecimento específico de cada novo nodo roteador que entra na rede. O agente de monitoramento forma a base para o processo de auto-cura proposta nesta dissertação.

5.2.3 Processos de Auto-Organização

O processo de auto-organização consiste basicamente em acoplar as propriedades Auto-x (otimização, configuração, cura e proteção), discutidas no Capítulo 3, como forma de estender as funcionalidades do protocolo de roteamento para as redes em malha sem fio. Com isso o protocolo de roteamento poderá ter um comportamento autônomo e assim aumentar o desempenho, tolerância a falhas e segurança da rede. Nas subseções seguintes é detalhada a forma, na qual as propriedades Auto-x foram acopladas nos protocolos de roteamento, dando ênfase às propriedades de auto-otimização e auto-configuração. É interessante ressaltar que as propriedades Auto-x são acopladas à camada de rede como extensão às funcionalidades dos protocolos de roteamento OLSR e AODV, e cada propriedade Auto-x desempenha um papel específico para estender as funcionalidades dos protocolos de roteamento.

As funcionalidades são detalhadas nas seções seguintes.

5.2.4 Auto-Otimização

A auto-otimização, neste contexto, utiliza-se de técnicas de otimização de sistemas. Segundo Camponogara (2006) a otimização de sistemas é considerada uma área da Matemática Aplicada que se preocupa em calcular valores ótimos para variáveis de decisão que induzem ao desempenho ótimo, satisfazendo as restrições de modo matemático.

O módulo de auto-otimização é responsável pela escolha dos melhores valores para os parâmetros dos protocolos AODV e OLSR. O problema, neste caso, é como escolher os melhores valores para os parâmetros, a fim de maximizar o desempenho, conforme a RMSF cresce e decresce em quantidade de dispositivos clientes.

Alguns elementos do problema são conhecidos:

- Principais parâmetros dos protocolos OLSR e AODV que influenciam a rede;
- Tipo de dados dos parâmetros (inteiro, real e booleano);
- Tamanho da rede em um instante k de tempo. Essa informação é obtida através do agente NS-Agent;

A solução, neste caso, foi o uso de otimização não linear sem restrições, pois todas as características do problema se encaixam nesta subárea da otimização.

No módulo de auto-otimização dos parâmetros nota-se que deve ser encontrado um mínimo local que possa substituir os valores padrões dos parâmetros dos protocolos OLSR e AODV nas escalas de tamanho da rede propostas como: pequena, média ou grande.

Na equação 5.1, pode se ver a função objetivo utilizada para otimizar os dados obtidos das simulações (vide seção 6.2.2).

A equação utilizada para adequar-se ao comportamento da análise obtida de cada valor dos parâmetros dos protocolos OLSR e AODV, foi a função cotangente hiperbólica (MATHWORLD, 1999), que pode ser vista a seguir:

$$FO = \left(\frac{e^{2x} + 1}{e^{2x} - 1} \right) \quad (5.1)$$

Na equação 5.3 foi usada para minimizar a função objetivo 5.1. Onde o y é resultado da análise da vazão para cada intervalo de tempo proposto (vide seção 6.2.2).

$$\text{Minimize } \sum_{i=1}^n [FO(i) - y(i)] \quad (5.2)$$

Detalhes sobre a análise feita através de simulações com a finalidade de obter os intervalos de valores que foram utilizados para alimentar as equações propostas, podem ser vistas na seção 6.2.2 do Capítulo 6.

5.2.5 Auto-Configuração

Este módulo é responsável por configurar os parâmetros dos protocolos de roteamento OLSR e AODV, conforme as informações do NS-Agent.

Os parâmetros devem ser configurados de forma a aperfeiçoar o desempenho do cômputo das rotas.

Os fatores que podem influenciar os parâmetros são: o tamanho da rede, a taxa de mobilidade, a distância de transmissão, a força do sinal de transmissão, o *overhead*, o *jitter* e o *delay*. O ato de configurar os parâmetros em redes complexas necessita de um acompanhamento do administrador da rede para configurar e gerenciar os parâmetros de forma correta e que não comprometam o desempenho da mesma. Na maioria dos casos esse acompanhamento não acontece e o administrador da rede, por falta de um conhecimento macro, acaba deixando valores padrões que em determinados estágios da rede podem influenciar drasticamente em seu desempenho, além de inferir em problemas de segurança. Implementando a capacidade de auto-configuração, nota-se que algumas vantagens são eminentes: a baixa intervenção do administrador e a consequente redução de custos de sua manutenção e instalação. Com isso, o administrador da rede pode direcionar seu tempo para a tomada de decisão em níveis mais críticos.

Para isto foi proposto o algoritmo 5.1, que é responsável por dividir o tamanho da rede em três diferentes escalas: pequena (SMALL), média (MEDIUM) e grande (LARGE), que são configuradas para adaptar-se aos parâmetros já otimizados pelo processo de auto-otimização, descrito na seção anterior. A

política de verificar o tamanho da rede depende da informação vinda do NS-Agent, pois quando aumenta-se o número de dispositivos clientes na rede o algoritmo adapta-se conforme este crescimento.

Algoritmo 5.1 Controle dos parâmetros dos protocolos conforme o tamanho da rede

```

1: INÍCIO
2: network_size =? {Informação vinda do agente de Tamanho, NS-Agents.size}
3: small_scale = 30 {número aproximado de nodos na rede, que compõem a escala
   pequena}
4: medium_scale = 70 {número aproximado de nodos na rede, que compõem a escala
   média}
5: large_scale = 100 {número aproximado de nodos na rede, que compõem a escala
   grande}
6: while Tiver_nodos_na_rede do
7:   if network_size < small_scale then
   {Configura os parâmetros do AODV e OLSR, para trabalharem na escala pe-
   quena} {Os valores desses parâmetros são oriundos da capacidade de auto-
   otimização}
8:   end if
9:   if network_size > small_scale and network_size <= medium_scale then
   {Configura os parâmetros do AODV e OLSR, para trabalharem na escala média}
   {Os valores desses parâmetros são oriundos da capacidade de auto-otimização}
10:  end if
11:  if network_size >= medium_scale and network_size <= large_scale then
   {Configura os parâmetros do AODV e OLSR, para trabalharem na escala grande}
   {Os valores desses parâmetros são oriundos da capacidade de auto-otimização}
12:  end if
13: end while
14: FIM

```

A idéia principal do algoritmo de escalas é adaptar os valores dos parâmetros dos protocolos OLSR e AODV nas três escalas propostas. Para isso, usa-se uma lista de quantidade de nodos clientes na rede, que é obtida através do NS-Agent em um determinado instante de tempo. Com este processo, o administrador da rede não precisa preocupar-se em configurar os valores dos parâmetros do protocolos, conforme a rede cresce em tamanho (aumento dos nodos clientes).

Para configurar os limites das escalas 30, 70 e 100 utilizou-se as referências dos trabalhos de (TAN; SEAH, 2004), (HELD, 2005). Limitou-se as escalas em: 0 a 30 para escala pequena, de 30 a 70 para escala média e 70 a 100 para escala grande. Os testes e validações são melhor explanados no Capítulo 6.

Por fim, o algoritmo pode ser estendido para que outras políticas sejam acopladas e configuradas para obter melhor desempenho em pontos particulares da rede.

5.2.6 Auto-Cura e Auto-Proteção

O módulo de Auto-Cura é responsável por detectar, localizar e reparar falhas nos protocolos de roteamento OLSR e AODV, ou seja, quando um nodo roteador deixa de funcionar por algum motivo, os outros nodos roteadores que estão na rede percebem isso através do Agente de Monitoramento (NM) e tentam reorganizar as rotas a fim de criar rotas alternativas para os nodos que foram prejudicados com a queda do nodo-roteador. Já o módulo de Auto-Proteção serve para garantir um nível de segurança na rede, observou-se a necessidade de adaptar técnicas utilizando conceitos de sistemas imunológicos dos trabalhos de (MACHADO et al., 2005) (SOMAYAJI et al., 1998) (BALASUBRAMANIYAN et al., 1998), a fim de garantir a detecção de intrusão em larga escala para às redes em malha sem fio.

No escopo dessa dissertação optou-se em enfatizar com mais detalhes aos módulos de auto-otimização e auto-configuração, deixando os módulos de auto-cura e auto-proteção como trabalhos futuros.

5.3 Considerações Finais

Foi visto neste capítulo que a arquitetura proposta serve para reutilizar os protocolos de roteamento já disponíveis para redes sem fio, por exemplo: OLSR e AODV e observou-se também que a arquitetura proposta pode ser estendida para demais protocolos. Para isso, deve-se modificar os módulos para dar suporte ao respectivo protocolo escolhido. Percebeu-se que a arquitetura proposta com todas as suas funcionalidades implementadas, tentaria suprir as carências que as redes em malha sem fio possuem, as quais são: desempenho, tolerância a falhas e segurança.

Como primeira experiência neste trabalho enfatizou-se a questão do desempenho nas RMSF, utilizando as capacidades de auto-otimização e auto-configuração. Neste trabalho não foi necessário o desenvolvimento de um novo protocolo de roteamento. A alternativa foi a adaptação dos protocolos existentes (OLSR e AODV) para as redes sem fio adaptando-os as RMSF. Isso resolve o problema da escassez de protocolos de roteamento eficientes para redes em

malha sem fio, como enfatizado no Capítulo 4. Por fim, para a realização de testes que permitam avaliar a arquitetura proposta é necessária a implementação dos seus módulos em um simulador. Nos artigos (GUARDALBEN et al., 2007b), (GUARDALBEN et al., 2007a), podem ser observados os passos iniciais da arquitetura em nível conceitual, enquanto que no artigo (GUARDALBEN et al., 2008b), observa-se o método de auto-configuração aplicado ao protocolo de roteamento OLSR.

Sumariando, a arquitetura pode ser tida como uma extensão aos protocolos de roteamento já existentes, pois a motivação deste trabalho não foi desenvolver um novo protocolo de roteamento auto-organizado, mas sim, adaptar os conceitos de auto-organização aos protocolos de roteamento para redes sem fio já existentes.

O próximo capítulo apresenta a modelagem e simulação da arquitetura proposta, enfatizando a implementação das capacidades de auto-otimização e auto-configuração, aplicadas aos protocolos OLSR e AODV e mostrando suas respectivas análises e resultados.

6 Modelagem e Simulação

Com o intuito de realizar experimentações e mensurar os resultados obtidos, foi utilizado o simulador de redes Opnet (*Optimized Network Engineering Tools*) (OPNET, 1987a) (OPNET, 1987b), o qual obtém a flexibilidade necessária para a implementação dos modelos de processo utilizados para validar a arquitetura proposta.

6.1 Introdução

De acordo com Freitas (2001), a simulação computacional de sistemas, ou simplesmente simulação, consiste em utilizar determinadas técnicas matemáticas empregadas em computadores digitais, permitindo imitar o comportamento de praticamente qualquer tipo de operação ou processo (sistemas) do mundo real. Basicamente três soluções podem ser usadas para avaliar o desempenho em uma rede de computadores, são elas: experimentos utilizando redes reais, utilização de métodos matemáticos (analíticos) e por último, a simulação computacional. Nesta dissertação foi utilizada a simulação computacional, motivada pela dificuldade em se criar uma implementação de uma rede em malha sem fio aplicada em um cenário real, em função do custo de aquisição dos equipamentos.

Através de simulações, pôde ser observado o comportamento dos protocolos OLSR e AODV auto-configurados e otimizados, em comparação à proposta padrão dos mesmos. No Capítulo 4 dessa dissertação foi apresentado o funcionamento básico dos protocolos de roteamento OLSR e AODV, estes foram utilizados como base à arquitetura de auto-organização proposta. Como primeira experiência, enfatiza-se o modelo de auto-configuração e auto-otimização que, como já mencionado, possui a tarefa de configurar e otimizar os parâmetros dos protocolos de roteamento citados.

A arquitetura proposta neste trabalho foi parcialmente modelada na linguagem

de especificação formal Object-Z (DUKE et al., 1991) e pode ser encontrada nos artigos (GUARDALBEN et al., 2007a) e (GUARDALBEN et al., 2007b).

Para realizar os experimentos com os protocolos OLSR e AODV, utilizou-se o simulador de redes Opnet, o qual tem sido utilizado por inúmeros pesquisadores e instituições de pesquisa para simular redes robustas e com grande capacidade para escalabilidade.

Os resultados foram analisados e comparados com os valores padrões produzidos pelos protocolos de roteamento OLSR e AODV. As métricas de desempenho analisadas foram coletadas conforme o cenário proposto: uma rede em malha sem fio intra-campus. Entretanto, como premissa, foi observado que as métricas de desempenho utilizadas para análise dos protocolos foram subdivididas em grupos de métricas: globais, específicas e customizadas.

Como resultado preliminar, comparou-se o desempenho entre a capacidade de auto-configuração adaptado aos protocolos de roteamento OLSR e AODV, gerando os resultados demonstrados em (GUARDALBEN et al., 2008a). Observou-se que o módulo de auto-configuração possui melhor desempenho em relação à vazão, descoberta de rotas e descarte de pacotes. No artigo citado não foi utilizado o método de auto-otimização, porém nesta dissertação os resultados foram verificados e analisados utilizando ambas as capacidades: auto-otimização e auto-configuração propostas.

Nas seções seguintes serão apresentados o desenvolvimento dos módulos propostos no simulador e os resultados experimentais dos protocolos de roteamento OLSR e AODV, aplicados às redes em malha sem fio.

6.1.1 Software de Simulação Opnet (*Optimized Network Engineering Tools*)

O *Opnet* é um software de simulação de redes que permite simular redes de pequeno, médio e grande porte. Foi desenvolvido pela empresa *Opnet Technologies* em meados da década de 80 e é mantido até sua versão atual 14.5. Utiliza-se linguagem de programação C/C++ e o conceito de FSM (*Finite State Machines*) para desenvolver os modelos de simulações. O *Opnet* permite a simulação de protocolos de rede como o UDP e o TCP, comportamento de tráfego como FTP, CBR, vídeo, email, telnet, voip, http e também inclui modelos dos protocolos de roteamento para redes sem fio como: OLSR, AODV, DSR e TORA. Além de disponibilizar um ambiente flexível e dinâmico para a simulação de redes sem fio, incluindo tipos de padrões 802.11a/b/g, Wimax

e Zigbee. Suporta modelos de mobilidade como trajetórias arbitrárias, mobilidade HLA e *Random Waypoint*. Em Lucio et al. (2003) é demonstrado que o simulador *Opnet*, comparado ao simulador de redes NS-2 (*Network Simulator 2*), possui melhor desempenho em relação à caracterização de tráfego, por causa da maior variação de tráfegos disponíveis, podendo até incorporar análises de tráfegos de redes reais utilizando, por exemplo, medições efetuadas com o software MRTG (OETIKER; RAND, 2008) na mobilidade e na adaptação dos parâmetros da simulação. É válido ressaltar que a dificuldade em se utilizar e implementar os *scripts* em NS-2 impactaram na produtividade ao se desenvolver os módulos de simulação, ao contrário do software *Opnet*, que possui uma interface gráfica flexível e produtiva, e, um ambiente para análise de eventuais erros no decorrer da implementação, além de possuir uma maior variedade de módulos aplicados às redes sem fio. Portanto, é justificado o uso do simulador *Opnet* como sendo o simulador escolhido para implementar os modelos de processos usados nesta dissertação.

Para seguir um fluxo lógico na modelagem e simulação deste trabalho, utilizou-se o diagrama proposto por Chang (1999), que divide a modelagem e simulação em cinco etapas. O diagrama foi adaptado para adequar-se a este projeto e pode ser visualizado na Figura 6.1, destacando-se o processo de validação feito com base nas etapas 1, 2 e 3.

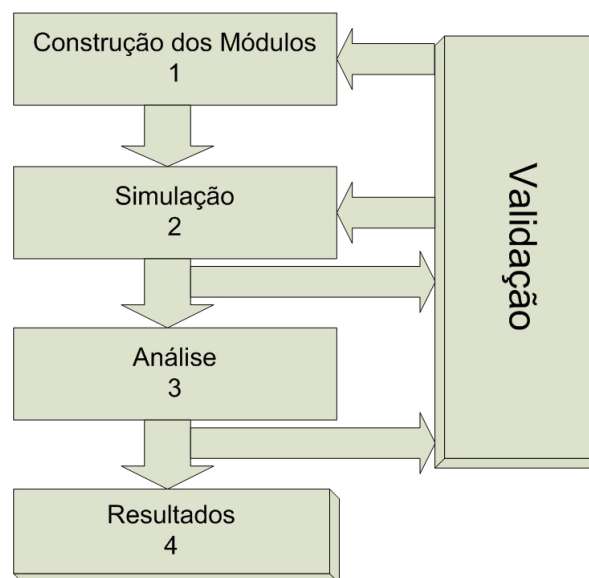


Figura 6.1: Etapas do processo de modelagem e simulação utilizado para validar a arquitetura proposta no simulador *Opnet*

6.2 Etapa 1: Construção dos Modelos de Processo

Nesta seção foi detalhada a construção dos modelos de processo utilizados para validar a presente proposta.

Os modelos de processo foram desenvolvidos com o propósito de simular e validar a arquitetura de forma específica, ou seja, cada capacidade de auto-organização sugerida juntamente com a troca de mensagens utilizadas pela plataforma de agentes.

Para simular a arquitetura proposta desenvolveu-se os módulos de processo em linguagem *C/C++* e utilizou-se como seqüência de ciclos as FSM (*Finite State Machines*), ou seja, máquinas de estados finitos para organizar as seqüência de passos de cada módulo de processo desenvolvido. A ferramenta de simulação *Opnet* possui suporte para redes móveis e estrutura para simular redes em larga escala. Para compilar os módulos de processo utilizou-se o compilador Visual C++ versão 6.0 e posteriormente converteu-se de forma automática pelo simulador *Opnet*, em arquivos no formato EMA (*External Module Application*), os quais são arquivos interpretados pela ferramenta de simulação.

Os modelos de processos apresentados nas seções seguintes foram desenvolvidos em camadas, começando pela implementação da plataforma dos agentes, mais tarde pelo processo de auto-organização (limitando-se às capacidades de auto-otimização e auto-configuração) e por fim aplicando-se essas capacidades como extensão para os protocolos de roteamento OLSR e AODV.

6.2.1 Modelo de Processo Agentes

A plataforma dos agentes foi implementada conforme o modelo de processo exemplificado na Figura 6.2 . Neste modelo, considerou-se os agentes de software como sendo estáticos em cada nodo roteador em malha. Cada agente está acoplado dentro do nodo-roteador. Para o desenvolvimento da plataforma não desenvolveu-se todas as funcionalidades de um agente de software como pode ser visto em Weiss (1999), pois isso não foi o escopo deste trabalho. Desenvolveu-se no simulador a troca de mensagens entre os agentes, conforme pode ser observado na Figura 6.2 . Os agentes NS-Agents foram divididos em NS-Agent (Envia), que possui a função específica de enviar mensagens para o NS-Agent (Recebe), com todas as informações sobre a quantidade de

dispositivos contidos na rede. Isso é feito por troca de mensagem, que é melhor detalhada na subseção abaixo.

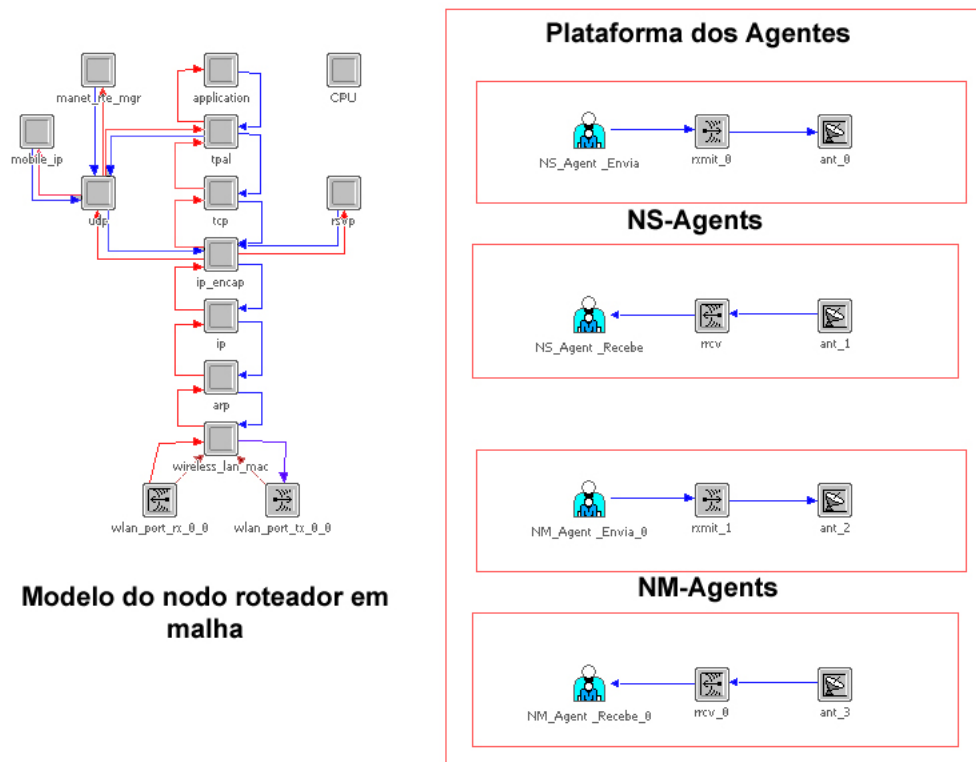


Figura 6.2: Modelo da plataforma de agentes acoplado ao nodo roteador em malha

O Modelo do nodo roteador no simulador Opnet contém todas as funções de um roteador, ou seja, desde do acesso às camadas físicas (Wireless_Lan_Mac), de rede, de transporte até a camada de aplicação.

Para a troca de mensagens entre os agentes criou-se um formato de pacote específico conforme pode ser observado na figura 6.3. Este formato possui quatro campos, que são: endereço de origem de quem enviou a mensagem, endereço para o destino, número de nodos fixos e número de nodos móveis. Cada nodo-roteador possui estes campos localmente. Posteriormente estes campos são unidos em um pacote que é enviado de tempos em tempos para todos os nodos vizinhos de um determinado roteador, com o intuito de obter a lista do tamanho da rede sempre atualizada.

Em função do método utilizado para envio destes pacotes ser por inundação, deve-se configurar um tempo de envio da atualização da lista, que pode variar conforme o acúmulo de mensagens na rede. O tempo padrão configurado é medido em segundos, portanto a cada 50 segundos os nodos roteadores enviam suas informações, evitando assim o acúmulo de troca de mensagens dos

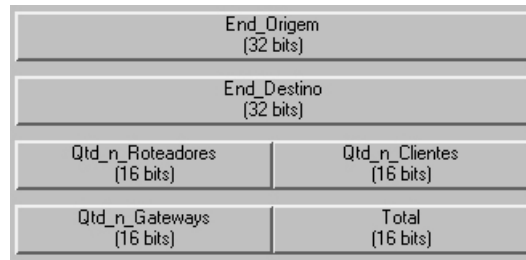


Figura 6.3: Estrutura do pacote utilizado para descoberta do tamanho da rede pelo NS-Agents

agentes.

Os agentes NM-Agents também são divididos em NM-Agent (Envia) e NM-Agent (Recebe) e têm a função de monitorar os nodos roteadores em malha que estão ativos ou não na rede. O NM-Agent (Envia) tem a função de contactar o NM-Agent (Recebe) de tempos em tempos. Esse processo forma a base de auto-cura proposta para esta dissertação, porém essa capacidade não foi implementada ainda neste trabalho.

6.2.2 Modelos de Processo das Capacidades de Auto-Otimização e Auto-Configuração

Como visto no Capítulo 3, as capacidades de auto-otimização e auto-configuração possuem uma forte correlação, ou seja, o método de auto-configuração é complementado com o método de auto-otimização, enquanto que uma capacidade configura os parâmetros conforme a escala da rede, a outra otimiza os valores para serem acoplados nos parâmetros dos protocolos OLSR e AODV como uma extensão aos mesmos.

O modelo de processo de auto-otimização foi implementado conforme o comportamento também descrito no Capítulo 5. Para implementar este processo, modificou-se os módulos originais dos protocolos OLSR e AODV (`aodv_rte.pr.c` e `olsr_rte.pr.c`) que fazem parte do suporte para redes MANET do simulador Opnet e adicionou-se a função de auto-otimização e auto-configuração. Foram feitas inúmeras modificações na estrutura dos módulos com intuito de adaptar essas capacidades.

Mencionou-se no Capítulo 5 que o método de auto-otimização é responsável pela escolha do valor otimizado para ser colocado em cada parâmetro dos protocolos de roteamento OLSR e AODV. Para obter-se o valor otimizado foi simulado uma rede em malha sem fio Intra-Campus conforme o cenário

apresentado na Figura 6.5 seção 6.3. Neste cenário simulou-se a variação da quantidade de nodos clientes associados em cada nodo roteador, juntamente com a variação do intervalo dos parâmetros escolhidos para este experimento. Na Figura 6.4 , pode-se observar um exemplo do comportamento da curva gerada através da relação vazão versus variação do tempo do Intervalo de mensagens HELLO, conforme mencionado no capítulo 5, na seção auto-otimização.

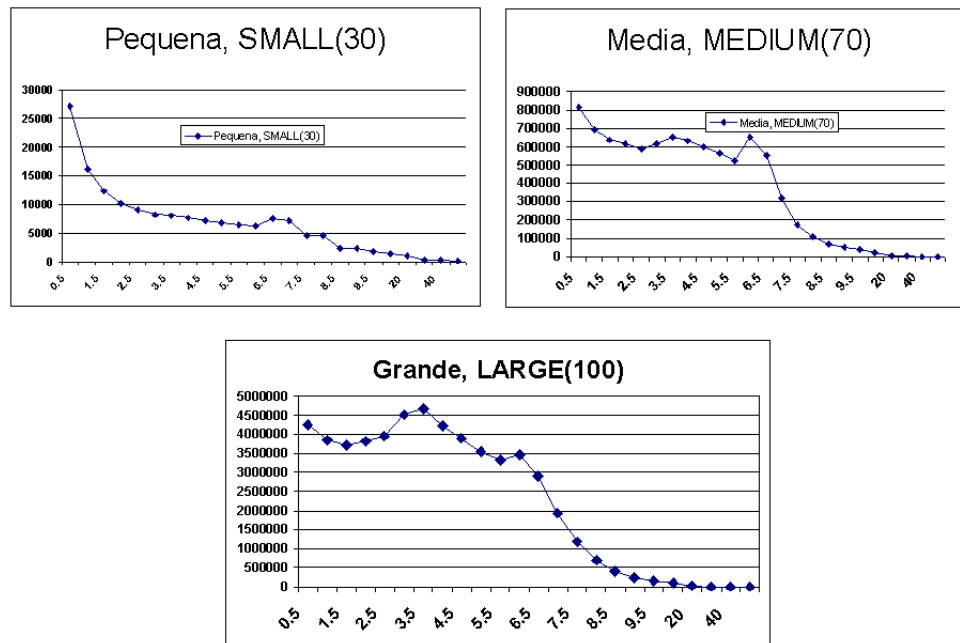


Figura 6.4: Exemplo da análise das escalas do parâmetro Hello-Interval conforme a métrica de vazão e a variação do intervalo de tempo

Neste exemplo nota-se que a análise da variação do valor do parâmetro foi efetuada de forma independente para as três escalas propostas, pequena, média e grande. Observa-se também que a função objetivo (vide seção 5.2.4) proposta adaptou-se de forma satisfatória aos gráficos da Figura 6.4 , resultando em um valor otimizado para cada escala proposta. Para o processo de otimização da curva gerada, optou-se em utilizar o software Matlab com a função de otimização *lsqnonlin*, esta função é regularmente utilizada para resolver problemas de ajuste não linear, e foi utilizada neste trabalho para auxiliar no ajuste da curva gerada pela análise específica de cada parâmetro. Utilizou-se a função objetivo cotangente hiperbólica e a função de minimização da função objetivo, que foram apresentadas no Capítulo 5, seção 5.2.4. Com os dados obtidos da Tabela 6.1 traçou-se a curva nas três escalas e, após isso, pesquisou-se na literatura uma função objetivo que melhor se adequasse ao comportamento da curva gerada. Após testes e análises no simulador escolheu-se a função cotan-

gente hiperbólica, pois esta obteve o melhor comportamento dentre as outras funções objetivas analisadas (interpolação linear e tangente hiperbólica).

Para exemplificar o processo de análise utilizou-se como referência o parâmetro Hello-Interval, mas o mesmo processo foi feito para os demais parâmetros escolhidos.

O código em Matlab utilizado para otimizar a escolha do melhor valor conforme a escala da rede, pode ser visto no Apêndice B desta dissertação.

Nodos	0.5	1	1.5	2	...	40	50
SMALL (10)	28.41469	24.5938	24.59264	71.20893		16.16489	16.16489
SMALL (20)	1290.703	861.9226	656.1949	575.5491		248.0969	248.0969
SMALL (30)	659.5588	443.2582	340.3938	323.379		132.1309	132.1309
MEDIUM(40)	2646.458	1704.578	1303.33	1163.118		765.0526	765.0526
MEDIUM(60)	6692.721	3773.002	2860.065	2285.047		882.2232	837.5746
MEDIUM(70)	4669.589	2738.79	2081.698	1724.083		823.6379	801.3136
LARGE(80)	8583.169	5175.946	3907.206	3290.569		1620.508	1664.479
LARGE(90)	11324.1	6876.763	5405.711	4612.651		2375.319	2165.133
LARGE(100)	9953.633	6026.355	4656.459	3951.61		1997.914	1914.806

Tabela 6.1: Exemplo dos dados captados do parâmetro Hello-Interval usando a vazão como métrica de análise

Conforme observado na Tabela 6.1 pode ser visto o exemplo dos dados obtidos somente do parâmetro Hello-Interval. Ainda neste exemplo, nota-se, que para a análise do Hello-Interval variou-se o intervalo de tempo de 0.5 até 50 segundos e também a quantidade de nodos-clientes que foi variada conforme a escala da rede: SMALL (0 a 30), MEDIUM (30 a 70) e LARGE (70 a 100).

Nas Tabelas 6.2 e 6.3 podem ser visualizados os parâmetros dos protocolos OLSR e AODV depois do processo de auto-otimização. Com intuito de analisar os parâmetros de forma específica, escolheu-se cinco parâmetros do protocolo AODV (Route-Request-Retries, Hello-Interval, Acitve-Route-Timeout, Allowed-Hello-Loss, Net-Diameter) e cinco parâmetros do protocolo OLSR (Hello-Interval, Tc-Interval, Neighbor-Hold-Time, Tc-Hold-Time, Duplicate-Hold-Time). Esta análise foi realizada variando-se o tempo de intervalo de cada parâmetro e analisando o crescimento da rede conforme as escalas. Para isso usou-se a métrica de vazão da rede.

Conforme pode ser observado nas Tabelas 6.2 e 6.3 elas contêm todos os parâmetros dos protocolos OLSR e AODV com seus respectivos parâmetros já otimizados, com o método de auto-otimização. Essa tabela foi acoplada no método de auto-configuração e posteriormente abstraiu-se os resultados que podem ser vistos na seção 6.5.

Além da análise específica dos cinco parâmetros também foram modificados alguns outros parâmetros de forma a privilegiar a rede conforme a escala. Como

Parâmetros OLSR	Padrão	Small(Pequena)	Medium(Média)	Large(Grande)
node_willingness	default	medium	high	always
hello_interval	2,0	8.7311	2.9802	2.8312
tc_interval	5	1.3388	0.0588	9.8348
neighbor_hold_time	6	2.3283	1.6764	1.4107
tc_hold_time	15	4.0745	3.7253	2.6822
dup_hold_time	30	20	35	10

Tabela 6.2: Descrição dos Parâmetros do protocolo OLSR depois do processo de auto-otimização

exemplo disso pode-se citar o parâmetro Node-Willingness que faz parte do conjunto de parâmetros do protocolo OLSR. Neste parâmetro foi configurado para *medium* na escala pequena, *high* para escala média e *always* para escala grande, que serve para encaminhar o tráfego do próprio nodo para os demais nodos conforme o *Willingness* apropriado. Observou-se que a medida que a rede cresce é interessante deixar o parâmetro *Willingness* encaminhar todo o tráfego sempre.

Parâmetros do AODV	Padrão	Small (Pequena)	Medium (Média)	Large (Grande)
route_request_retries	1	8.2171	3.2413	2.9104
route_request_rate	10	10	10	10
grat_route_reply_flag	False	True	False	False
dest_only_flag	False	True	False	True
ack_required	False	True	True	False
hello_interval	1,0	3.2906	5.9951	9.0137
active_route_timeout	3,0	2.7784	4.3656	5.8208
allowed_hello_loss	2	7.0288	2.5524	1.3097
net_diameter	35	8	3	1
node_traversal_time	0,04	0,04	0,04	0,04
route_error_rate	10	5	10	15
timeout_buffer	2	2	2	2
pkt_queue_size	-1	-1	-1	-1
local_repair	True	True	True	false
ttl_start	1	1	1	1
ttl_increment	2	2	2	2
ttl_threshold	7	7	7	7
local_add_ttl	2	2	2	2

Tabela 6.3: Descrição dos Parâmetros do protocolo AODV depois do processo de auto-otimização

Já no protocolo AODV os parâmetros que foram modificados são: grat-route-reply-flag (*true* para escala pequena, *false* para escala média e grande), dest-only-flag (*true* para escala pequena, *false* para escala média e *true* para escala grande), ack-required (*true* para escala pequena, *true* para escala média e *false* para escala grande), local_repair(*true* para escala pequena, *true* para escala média e *false* para escala grande). Já os parâmetros ttl's foram mantidos os mesmos que os valores originais. Estas mudanças nesses parâmetros contribuíram para privilegiar alguns momentos na rede, por exemplo, o parâmetro ack_required, neste parâmetro, verifica-se a necessidade de garantir um retorno de um ACK (*acknowledgement*) somente nas escalas pequenas e médias, pois

na escala grande não se torna necessário o envio de um ACK, pois neste caso pode haver um grande acúmulo de mensagens de ACK na rede. Este mesmo conceito do `ack_required` foi usado para os demais parâmetros acima citados. É válido ressaltar que nas Nas Tabelas 6.2 e 6.3, a auto-otimização é feita de forma *off-line*, ou seja, os dados são otimizados externamente pelo MATLAB e depois acoplada dentro do módulo de auto-otimização que se encontrada implementado dentro do simulador Opnet.

6.3 Etapa 2: Simulação

Esta seção descreve o equipamento e o cenário de simulação utilizado para a análise do desempenho entre os protocolos OLSR e AODV padrão e OLSR e AODV auto-otimizados e configurados. Utilizou-se um notebook Toshiba Satellite M105 com 1.66GHz Core Duo T2300, 1.5GB DDR2, 100GB com Windows XP para realizar as simulações. Já o cenário simulado limita-se a uma rede em malha sem fio intra-campus que utiliza o padrão 802.11g para comunicação sem fio e também o método de comunicação únicos-rádios, ou seja, uma única placa de rede conectada ao nodo roteador, o qual serve tanto para criar o *backbone* entre os nodos roteadores, como para distribuir acesso aos clientes. Neste cenário, diversas comparações foram realizadas com valores padrões e com valores obtidos através dos métodos de auto-otimização e auto-configuração. As métricas de comparação foram baseadas nas métricas globais, locais e específicas, descritas na seção análise.

A Figura 6.5 ilustra o cenário de uma rede em malha intra-campus. Este cenário é um ambiente em que os nodos estão dispostos em uma área de $2000 \times 2000m^2$ e apresenta uma quantidade de dispositivos móveis e fixos que se comunicam. Essa comunicação é feita utilizando múltiplos-saltos através de uma única interface de rede sem fio, no padrão 802.11g (*Direct Sequence*), com uma taxa de dados de 11 Mbps. Esse tipo de rede também é conhecido como rede em malha sem fio únicos-rádios. Foi utilizado o mesmo cenário para simular os protocolos OLSR e AODV Auto-Organizados comparando-se com a abordagem padrão. O crescimento da rede é configurado pela entrada de nodos clientes dentro da rede, ou seja, a medida que os nodos clientes entram na rede contabiliza-se através do agente de controle do tamanho (`NS_Agent`) o tamanho da rede.

Para realizar os experimentos com os métodos de auto-otimização e auto-

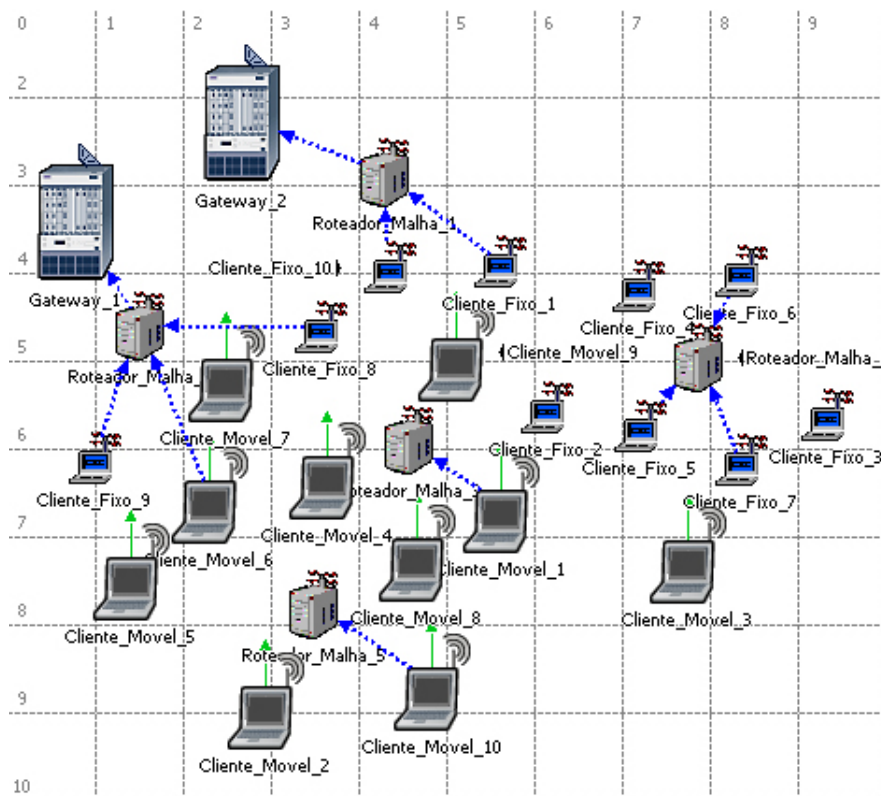


Figura 6.5: Exemplo do cenário intra-campus utilizado para simulação dos nodos dispostos em uma área de $2000 \times 2000 m^2$

configuração acoplados nos protocolos OLSR e AODV, utilizou-se o simulador Opnet versão 11.5, com o módulo para redes sem fio ativado. A Tabela 6.4 apresenta os parâmetros utilizados. Nas simulações utilizou-se o modelo de propagação *Two Ray Ground* que simula os fenômenos físicos: força, interferência e atraso na propagação do sinal, abstraindo as características reais de uma rede sem fio. Utilizou-se o padrão 802.11g para ser a variante base de nossos experimentos. Para o modelo de tráfego, diferentes demandas representadas pelas setas pontilhadas foram configuradas para o cenário proposto, tais como: HTTP, FTP, VOIP e também o tráfego real capturado pela ferramenta MRTG (MRTG, 2005). Este último foi importado para o simulador através da função *Import Baseline Loads*. O tipo de antena utilizado foi o omnidirecional, pois como já mencionado, cada nodo possui uma área de cobertura de alguns metros e comunica-se através de múltiplos-saltos.

Os nodos clientes móveis deslocam-se respeitando os limites da propagação do sinal utilizado, que neste caso utilizou o *Random Waypoint* (TOWSLEY, 2008), sendo que o percurso de cada nodo móvel é caracterizado como um zig-zag. O padrão de velocidade dos nodos clientes móveis foi definido como 1

Cenário:	Rede Intra-Campus de 2000 x 2000m ²
Nodos Clientes:	Fixos e Móveis com diferentes mobilidades (1, 2 m/s)
Canal:	Canal Sem Fio
Modelo de Propagação:	Two Ray Ground
Tipo de Interface:	Wireless NIC
Camada PHY:	802.11g
Modelo da Antena:	Omni-Antenna
Modelo de Mobilidade:	Random Waypoint
Protocolo de Transporte:	TCP e UDP
Quantidade de Roteadores em Malha Estáticos:	5
Tamanho da rede:	Escalas pequena(small), média(medium), grande(large)
Modelo de tráfego:	HTTP, FTP, VOIP, REAL
Largura de Banda dos Nodos:	2 Mbps
Protocolos de Roteamento:	AODV e OLSR
Tipo da Camada de Ligação:	LL
Tipo de Fila:	FIFO
Máximo de Pacotes em Fila:	50
Tempo de Simulação:	60 minutos
Intervalo de confiança	95%

Tabela 6.4: Descrição dos parâmetros de simulação utilizados no cenário intra-campus

e 2 m/s. Essa velocidade imita o comportamento de uma pessoa caminhando de forma aleatória entre o limite de propagação do cenário proposto, que é de 2000 x 2000m². O tempo de simulação foi de 60 minutos e as simulações foram repetidas cinco vezes, retirando-se a sua média para garantir a maior confiabilidade dos resultados.

Os detalhes da configuração dos parâmetros da rede sem fio para os nodos roteadores em malha e para os nodos clientes, podem ser visualizados nas Figuras 6.6 e 6.7 .

Wireless LAN	
Wireless LAN MAC Address	Auto Assigned
Wireless LAN Parameters (...)	
BSS Identifier	1
Access Point Functionality	Disabled
Physical Characteristics	Direct Sequence
Data Rate (bps)	2 Mbps
Channel Settings	
Transmit Power (W)	0.005
Packet Reception-Power Threshold	-95
Rts Threshold (bytes)	None
Fragmentation Threshold (bytes)	None
CTS-to-self Option	Enabled
Short Retry Limit	7
Long Retry Limit	4
AP Beacon Interval (secs)	0.02
Max Receive Lifetime (secs)	0.5
Buffer Size (bits)	256000
Roaming Capability	Disabled
Large Packet Processing	Drop
PCF Parameters	Disabled
HCF Parameters	(...)

Figura 6.6: Configuração dos parâmetros da interface sem fio de um nodo roteador

Wireless LAN (...)	
BSS Identifier	Auto Assigned
Access Point Functionality	Disabled
Physical Characteristics	Direct Sequence
Data Rate (bps)	1 Mbps
Channel Settings	
Transmit Power (W)	0.005
Packet Reception-Power Threshold	-95
Rts Threshold (bytes)	None
Fragmentation Threshold (bytes)	None
CTS-to-self Option	Enabled
Short Retry Limit	7
Long Retry Limit	4
AP Beacon Interval (secs)	0.02
Max Receive Lifetime (secs)	0.5
Buffer Size (bits)	256000
Roaming Capability	Disabled
Large Packet Processing	Drop
PCF Parameters	Disabled
HCF Parameters	Not Supported

Figura 6.7: Configuração dos parâmetros da interface sem fio de um nodo cliente

Nas Figuras 6.6 e 6.7 observa-se os parâmetros de configuração dos nodos

roteadores em malha e dos nodos clientes.

Os parâmetros que diferenciam essas configurações são: *BSS identifier* e taxa de dados (bps). O primeiro deve ser usado para identificar qual roteador o cliente vai se associar. Já o segundo, tem uma taxa de dados maior, de 2Mbps, enquanto que o cliente usa 1Mbps. Os demais parâmetros são iguais para ambos os nodos. É interessante ressaltar que o cliente associa-se a um nodo roteador conforme a área de cobertura do mesmo. Quando há mobilidade de um nodo cliente, este associa-se ao nodo roteador mais próximo. Cada nodo roteador possui um número único de associação.

6.4 Etapa 3: Análise

Nesta seção são abordadas as métricas utilizadas para comparar a arquitetura de auto-organização proposta, com os valores padrões dos protocolos de roteamento OLSR e AODV.

6.4.1 Métricas Globais

Neste tipo de métrica a rede é analisada de forma global, ou seja, as métricas são capturadas de todos os nodos pertencentes à rede.

As métricas podem ser observadas a seguir:

- **Número de pacotes que não foram entregues:** esta métrica identifica a quantidade de pacotes em bits/segundos, que foram rejeitados pela camada MAC. Isto pode ocorrer por diversos fatores, os mais prováveis são: sobrecarga na fila e falha no transmissor de pacotes.
- **Atraso fim-a-fim:** esta métrica é utilizada para representar o atraso fim-a-fim dos pacotes recebidos pela camada MAC de todos os nodos na rede. Posteriormente os pacotes recebidos são enviados para as camadas superiores.
- **Carga da rede:** Esta estatística é responsável por medir a carga da rede de cada BSS separadamente e posteriormente cada medição é adicionada como uma estatística global da rede.
- **Atraso de acesso ao meio:** é a representação de todos os pacotes em fila em função da contenção da rede.

- **Vazão:** representa o total de bits/segundo enviado das camadas inferiores para as superiores.

6.4.2 Métricas Específicas

Outro grupo de métricas utilizadas neste trabalho foi o de métricas específicas. Neste grupo, cada protocolo de roteamento possui sua própria métrica específica, que deve ser analisada de forma independente.

As descrições dessas métricas foram divididas conforme os seguintes protocolos de roteamento:

OLSR:

- **Tráfego de Mensagens HELLO:** Esta estatística coleta o total de mensagens HELLO enviadas, por segundo, na rede.
- **Número de MPR:** Esta estatística contabiliza o número de mensagens disseminadas pelos MPR's (*multipoint relays*) na rede, em um determinado momento.
- **Tráfego de Roteamento Recebido:** Esta estatística representa o total de tráfego de roteamento recebido, em bits/segundo, por todos os nodos na rede. O tráfego é composto por: HELLO Msgs + TC Msgs.
- **Tráfego de Roteamento Enviado:** Esta métrica representa o total de tráfego de roteamento enviado, em bits/segundo, por todos os nodos na rede.
- **Tráfego de Mensagens TC:** Esta métrica coleta o tráfego de mensagem TC (*Topology Control*) enviadas.

AODV:

- **Número de saltos por rota:** esta métrica representa o número de saltos (*hops*) para cada rota ao nodo destino, com base na tabela de roteamento.
- **Tamanho da fila de pacotes:** métrica responsável por representar o número de pacotes em fila, em um tempo qualquer, durante a simulação. A fila de pacotes é usada para armazenar os pacotes que não possuem

uma rota válida para serem despachados para o nodo destino, ficando assim em uma fila até a descoberta da rota correta para o nodo destino.

- **Tempo de descoberta de rota:** representa o tempo que um nodo leva para descobrir a rota para um nodo destino específico.
- **Tráfego de pacotes recebidos:** esta métrica representa a somatória do tráfego de pacotes recebidos. Pode ser em bits por segundo ou quantidade de pacotes por segundo.
- **Tráfego de pacotes enviados:** representa a somatória de pacotes enviados.
- **Número total de resposta de rotas locais enviadas:** representa o número total de respostas de rotas locais, de todos os nodos da rede, ou de um nodo específico somente. Por exemplo, quando um nodo recebe um pedido de rota e este possui, na sua tabela de roteamento local, o endereço para o nodo destino requisitado, o nodo envia uma resposta de que possui a rota desejada e não faz o *re-broadcast* do pacote de requisição.
- **Número total de pacotes descartados:** esta métrica representa o número total de pacotes descartados. Os pacotes podem ser descartados por inúmeros motivos, dentre eles: colisões de pacotes na rede, tamanho inadequado das filas e sobrecarga do tráfego entres os nodos.
- **Número total de respostas de rotas enviadas pelos nodos destinos:** representa o número total de pacotes de respostas de rotas enviadas pelo nodo destino da requisição de rota.
- **Número total de erros de rotas enviados:** esta métrica demonstra o número total de pacotes com erro de rota, enviados por todos os nodos ou por um nodo específico.
- **Número total de resposta de rotas:** representa o total de respostas de rotas, incluindo respostas de rotas locais e enviadas pelos nodos destinos.
- **Número total de requisições encaminhadas:** número total de requisições encaminhadas durante o processo de descoberta de rotas.
- **Número total de requisições de rotas:** representa a somatória das requisições de rotas enviadas por um nodo em específico, ou por todos nodos pertencentes à rede.

6.4.3 Métricas Customizadas

Além das métricas citadas, observou-se a necessidade de customizar algumas métricas com o intuito de capturar o comportamento de processos específicos.

A sobrecarga ou *overhead* na rede é uma das métricas específicas e é dada pela fórmula 6.1:

$$Sobrecarga = \frac{Total_Mensagens_HELLO_ou_Mensagens_TC}{Total_de_Pacotes_na_Rede} * 100 \quad (6.1)$$

Com essa métrica pôde ser observada a quantidade de pacotes de roteamento enviados na rede, durante a descoberta e manutenção das rotas.

6.5 Etapa 4: Resultados

Nesta seção são apresentados os resultados obtidos através da comparação entre os protocolos OLSR e AODV originais e OLSR e AODV Auto-Organizados, conforme as métricas descritas na seção 6.4 desta dissertação.

6.5.1 Capacidades de Auto-Otimização e Auto-Configuração Aplicadas ao Protocolo OLSR

Na Figura 6.8 é apresentada uma comparação de desempenho entre os protocolo OLSR:Auto-Organizado e OLSR:Padrão, com relação ao total de mensagens de HELLO enviadas.

Observa-se que a medida que a rede cresce em quantidade de dispositivos, a diferença começa a se acentuar. Concluiu-se com isto que quando a rede cresce o protocolo OLSR:Auto-Organizado injeta menos mensagens de HELLO na rede. Já nas Figuras 6.9 e 6.10, apresenta-se a comparação do protocolo OLSR:Padrão e OLSR:Auto-Organizado com relação ao tráfego de pacotes recebidos e enviados.

O protocolo OLSR:Auto-Organizado obteve melhor desempenho, ou seja, o mesmo recebeu mais pacotes de dados e injetou menos pacotes de manutenção e controle de rotas na rede. O fato de receber mais pacotes pode ser explicado

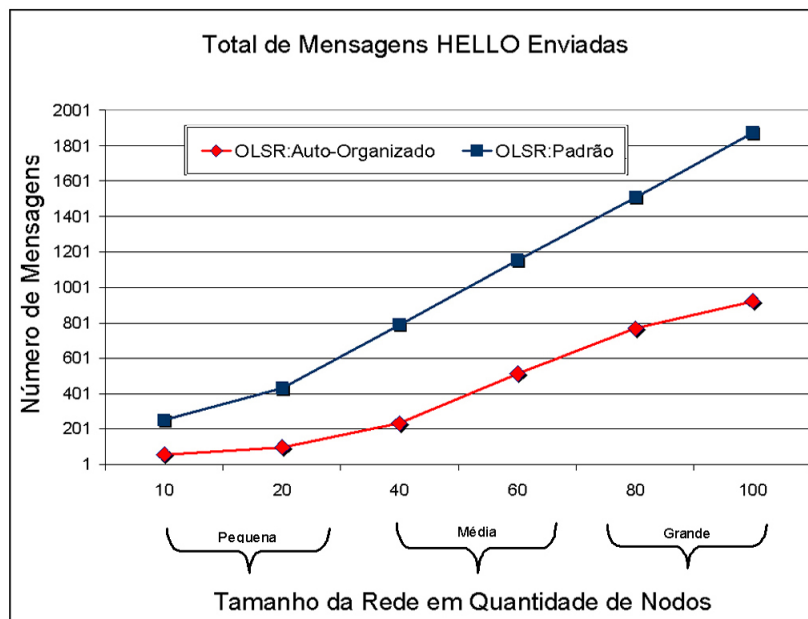


Figura 6.8: Comparação do total de mensagens HELLO enviadas do protocolo OLSR:Auto-Organizado e OLSR:Padrão

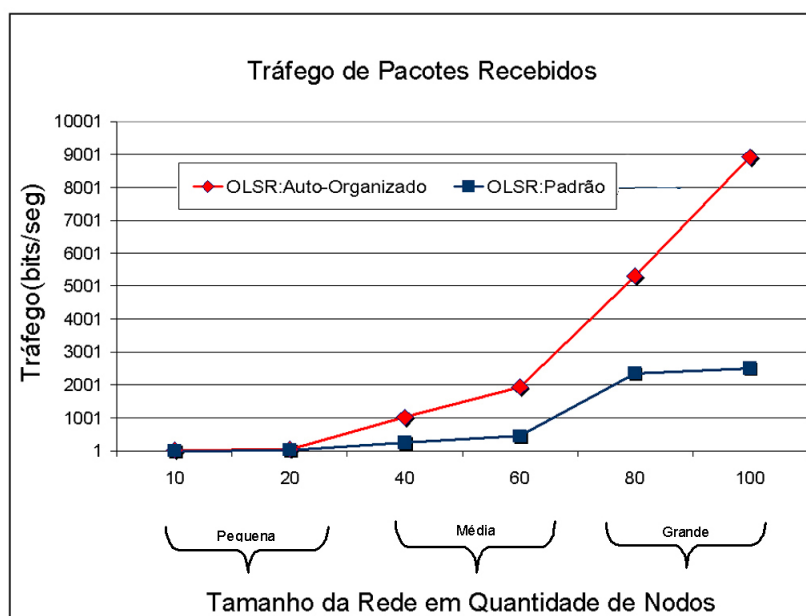


Figura 6.9: Relação do tráfego de pacotes recebidos entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão

pela menor perda de pacotes do protocolo OLSR:Auto-Organizado (ver Figura 6.13).

Pôde ser ainda observado, que na escala pequena o mesmo comportamento de ambas as comparações. Já nas escalas média para grande o OLSR:Auto-Organizado recebeu mais pacotes que o OLSR:Padrão. Em uma análise mais

específica do número total de mensagens HELLO enviadas, como mostrado na Figura 6.8, pôde ser observado que as escalas pequenas, médias e grandes, injetaram um número menor de mensagens HELLO na rede.

Observou-se também na escala média, um envio um pouco mais acentuado de pacotes pelo protocolo OLSR:Auto-Organizado. Este valor foi originado da análise específica do parâmetro TC-Interval utilizando a função de otimização, pois houve um maior envio de mensagens de TC nesta escala pelo protocolo OLSR:Auto-Organizado.

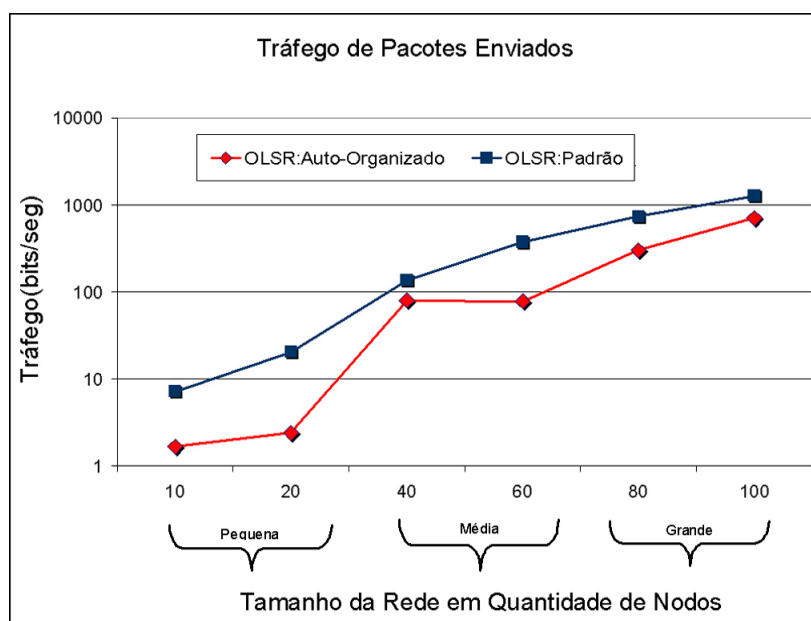


Figura 6.10: Relação do tráfego de pacotes enviados entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão

As Figuras 6.11 e 6.12 mostram a comparação entre mensagens TC (Controle de topologia) enviadas e o número de mensagens MPR entre o protocolo OLSR:Padrão e o OLSR:Auto-Organizado. Neste ponto o protocolo OLSR:Auto-Organizado injetou menos mensagens de controle e manutenção de rotas. Injetando menos mensagens na rede, evita-se a sobrecarga e o acúmulo excessivo de mensagens.

Esta é uma característica do protocolo OLSR, que tem uma alta sobrecarga de mensagens, devido à sua técnica para troca de mensagens na rede. Concluiu-se que o protocolo OLSR:Auto-Organizado obteve melhor desempenho que o OLSR:Padrão em todas as métricas analisadas.

Na Figura 6.13 é possível visualizar a quantidade de pacotes descartados pelos protocolos durante a simulação. Neste ponto nota-se que o protocolo

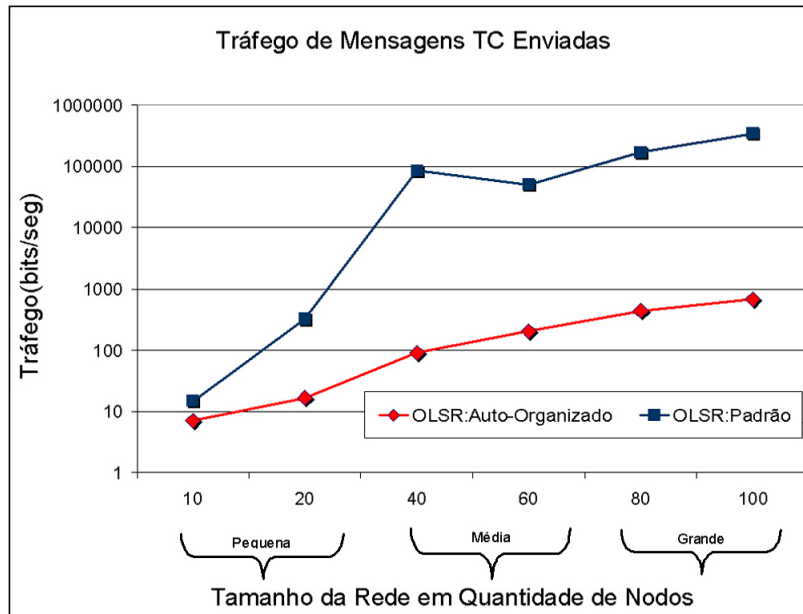


Figura 6.11: Comparação do tráfego de mensagens TC (*Topology Control*) enviadas do protocolo OLSR:Auto-Organizado e OLSR:Padrão

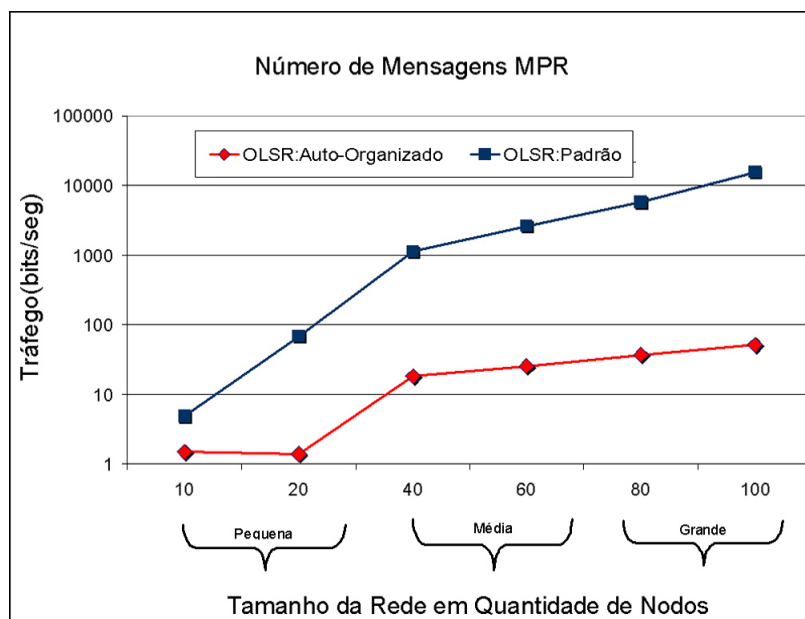


Figura 6.12: Comparação do número de mensagens MPR entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão

OLSR:Auto-Organizado obteve um menor descarte de pacotes em relação ao protocolo OLSR:Padrão, em todas as escalas. Observa-se também, que à medida que a rede cresce em quantidade de dispositivos, o descarte pelo protocolo OLSR:Padrão tende a ser maior.

Nota-se ainda, que em 10, 40 e 80 nós o descarte foi zero com relação a pa-

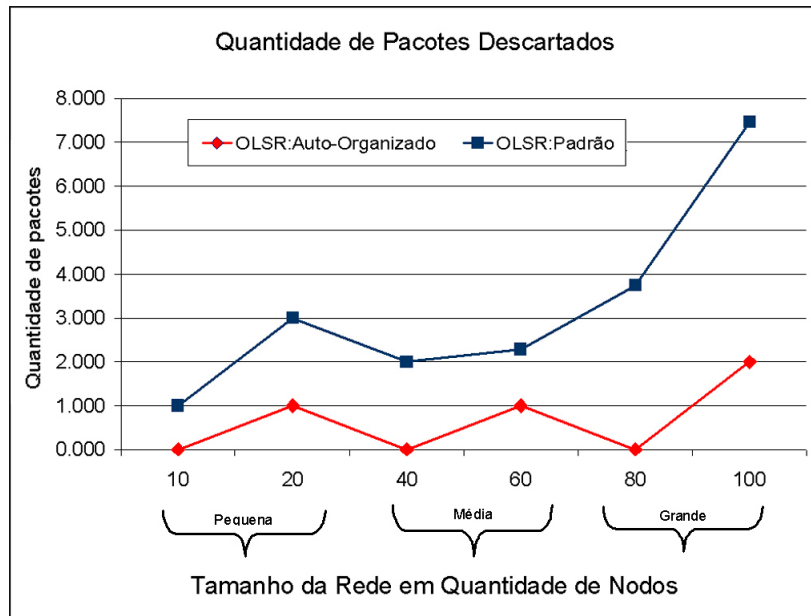


Figura 6.13: Comparação da quantidade de pacotes descartados do protocolo OLSR:Auto-Organizado e OLSR:Padrão

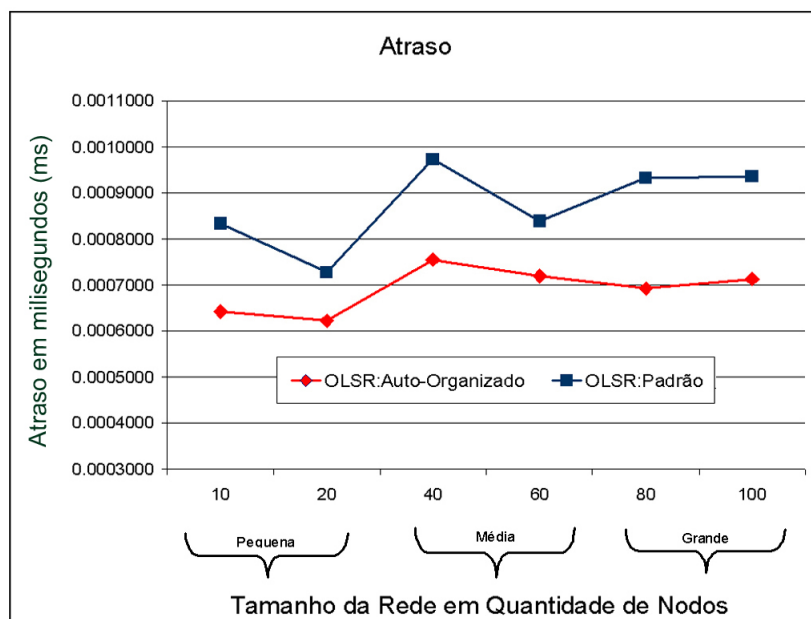


Figura 6.14: Relação do atraso fim-a-fim entre OLSR:Auto-Organizado e OLSR:Padrão

cotes em bits por segundo, enquanto que na Figura 6.14, observa-se o atraso fim a fim entre OLSR:Auto-Organizado e OLSR:Padrão. O OLSR:Auto-Organizado obteve menor atraso em comparação ao OLSR:Padrão.

Na Figura 6.15 é possível perceber que o atraso de acesso ao meio do protocolo OLSR:Auto-Organizado, é menor em relação ao OLSR:Padrão e pode

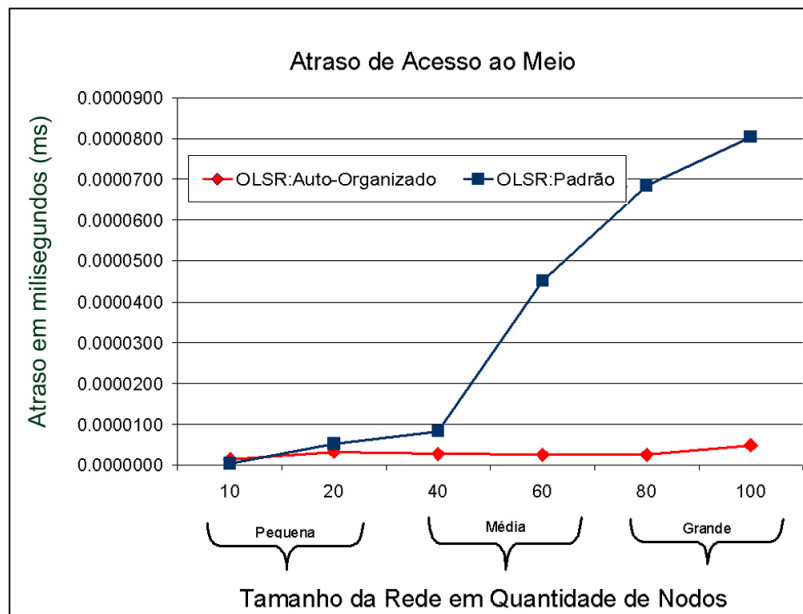


Figura 6.15: Comparação do atraso de acesso ao meio entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão

ser observado também que à medida que a quantidade de nodos cresce na rede, ou seja, da escala média para escala grande, a abordagem de Auto-Organização tem um desempenho mais significativo. Já na Figura 6.16 pode ser visualizada a carga da rede, que como observado no protocolo OLSR:Auto-Organizado, quando em escala pequena, obtém uma maior carga. O protocolo OLSR:Padrão obtém uma menor carga, mas em compensação nas escalas média para grande o OLSR:Auto-Organizado diminui a carga da rede.

Este fato é explicado pelo esforço do protocolo OLSR para encontrar as rotas de destino, pois como ele utiliza o método de inundação pelos MRP's, ou seja, necessita de um esforço extra para encontrar as rotas e esse esforço é compensando quando a rede cresce em tamanho, passando das escalas média para grande. O protocolo OLSR:Auto-Organizado obteve uma maior carga devido a configuração do parâmetro `TC_Interval` responsável pelo intervalo de envio de mensagens de controle da topologia como pode ser observado na Tabela 6.2. Nota-se que o valor do `TC_Interval` é menor que o intervalo do valor padrão, esse intervalo sendo menor conseqüentemente é injetado um maior número de mensagens de controle da topologia na rede aumentando em conseqüência a carga de mensagens na rede na escala pequena.

A Figura 6.17 simboliza a relação da vazão da rede entre OLSR:Auto-Organizado e OLSR:Padrão. Nesta figura nota-se que a medida que os nodos

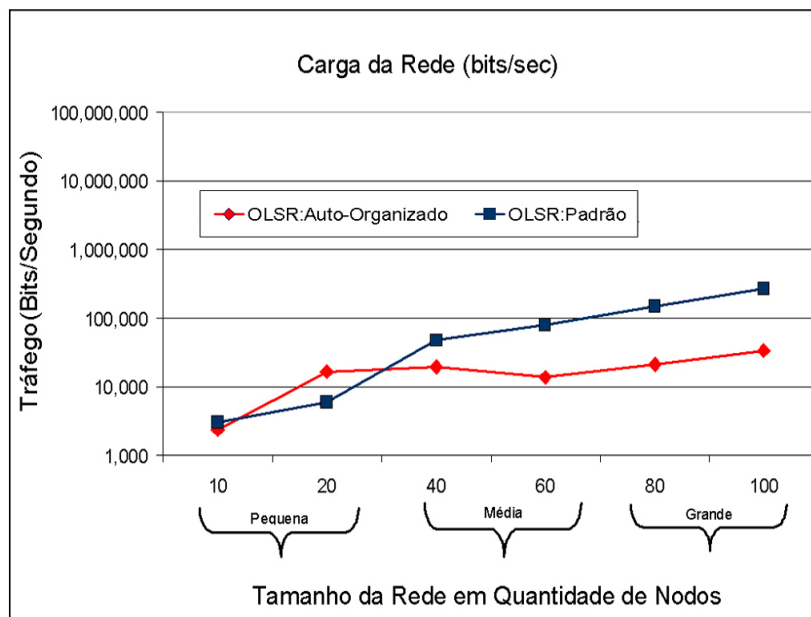


Figura 6.16: Relação da carga da rede entre o protocolo OLSR:Auto-Organizado e OLSR:Padrão

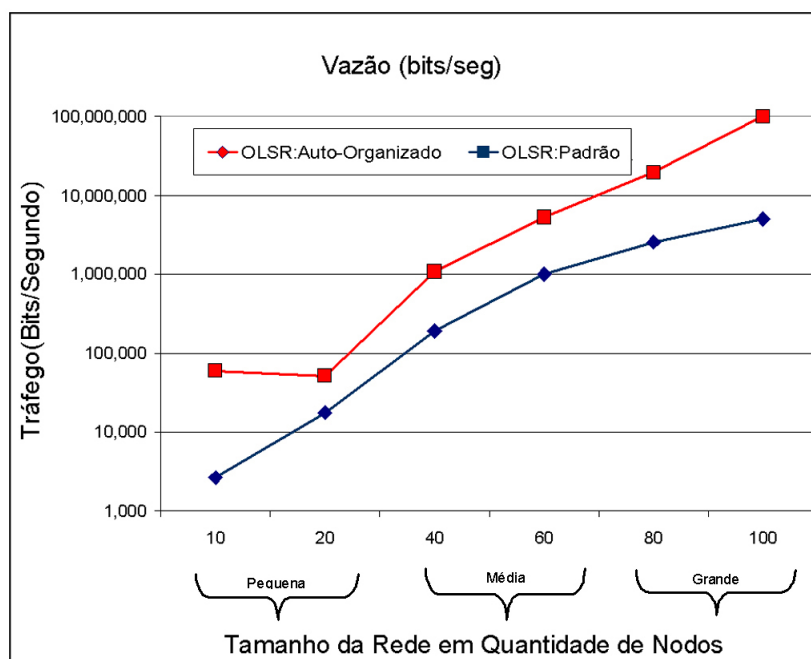


Figura 6.17: Comparação da Vazão entre o Protocolo OLSR:Auto-Organizado x OLSR:Padrão

entram na rede e ocorre a variação da escala pequena à escala grande, percebe-se uma melhora significativa na vazão do protocolo OLSR:Auto-Organizado.

Concluiu-se então, nesta análise específica, que o protocolo OLSR:Auto-Organizado obtém melhor desempenho quando é conhecida a posição dos nodos, pois a

técnica de encontrar as rotas pelo conhecimento geral da rede, demanda uma carga grande de mensagens de controle e manutenção de rotas. Por fim, notou-se que a auto-otimização e a auto-configuração dos valores dos parâmetros, podem garantir um melhor desempenho ao protocolo OLSR em ambientes de médio e grande porte.

6.5.2 Capacidades de Auto-Otimização e Auto-Configuração Aplicado ao Protocolo AODV

Na Figura 6.18 observa-se a relação do número de saltos por rota do protocolo AODV:Auto-Organizado.

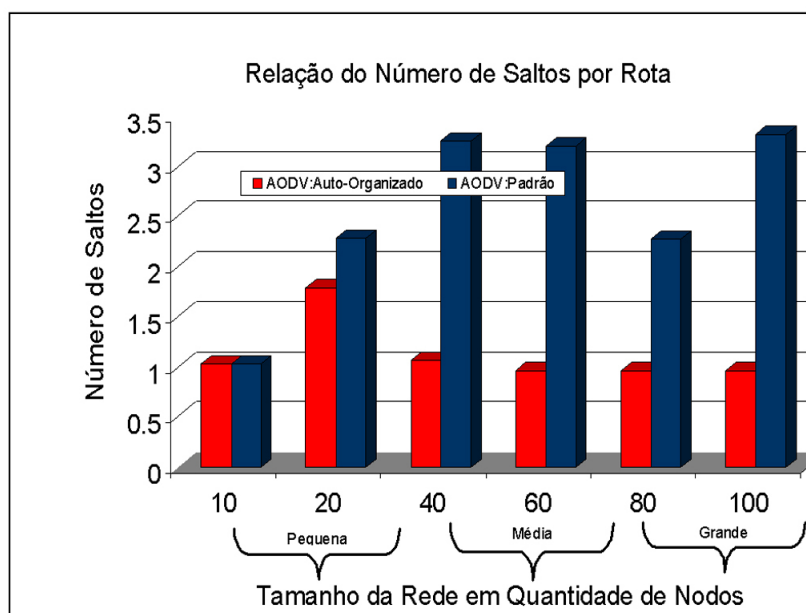


Figura 6.18: Comparação do número de saltos por rota entre o AODV:Auto-Organizado e AODV:Padrão

Verificou-se que o mesmo precisou de um número menor de saltos para encontrar o nodo destino nas escalas média para grande, enquanto que na escala pequena, o número de saltos foi praticamente o mesmo que o protocolo AODV:Padrão.

Observa-se que na Figura 6.19, o protocolo AODV:Auto-Organizado enviou um número menor de réplicas conforme a escala média para grande. Já na Figura 6.20 o número de pacotes recebidos pelo protocolo AODV:Auto-Organizado foi maior em relação ao protocolo AODV:Padrão. Essa melhora foi proporcionada pela menor perda de pacotes (vide Figura 6.22) e tam-

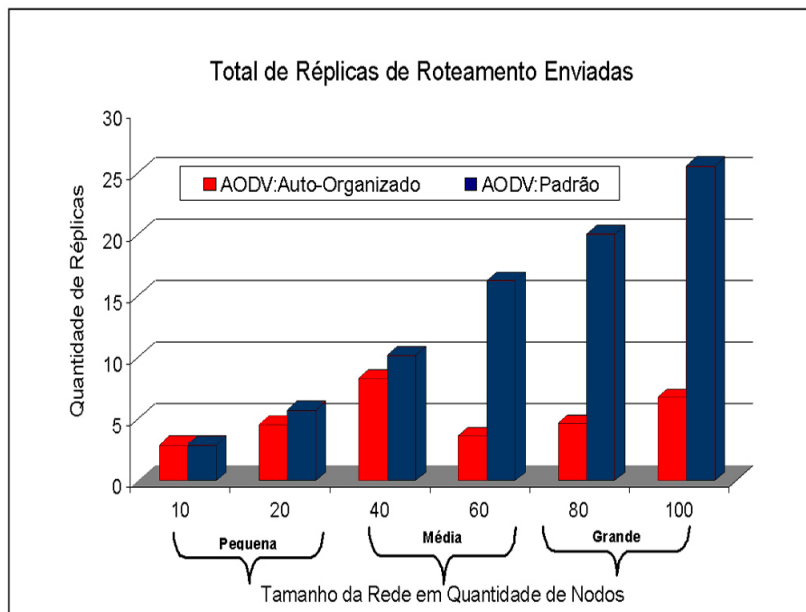


Figura 6.19: Comparação do total de réplicas de roteamento enviadas entre AODV:Auto-Organizado e AODV:Padrão

bém pelo ajuste do tempo dos pacotes que ficaram em *buffer*, esperando uma resposta de rota.

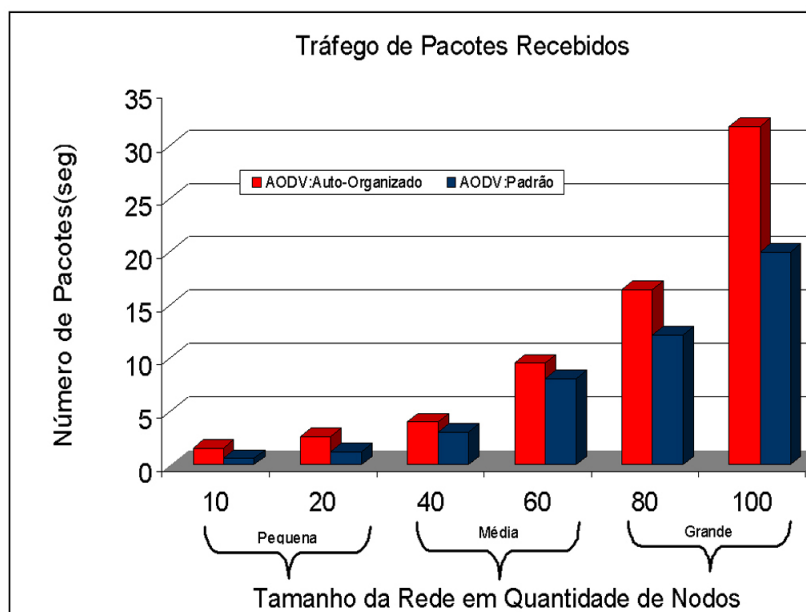


Figura 6.20: Comparação do tráfego de pacotes recebidos entre AODV:Auto-Organizado e AODV:Padrão

Na Figura 6.21 pode ser observada a relação de pacotes enviados, onde o protocolo AODV:Auto-Organizado obteve uma menor quantidade de pacotes

enviados em todas as escalas. É válido ressaltar, neste caso, que a quantidade de pacotes enviados é uma métrica que contabiliza todos os pacotes, inclusive os pacotes de controle e manutenção de rotas. Nota-se então, que o protocolo AODV:Auto-Organizado enviou um número menor de pacotes de controle e manutenção de rotas neste experimento.

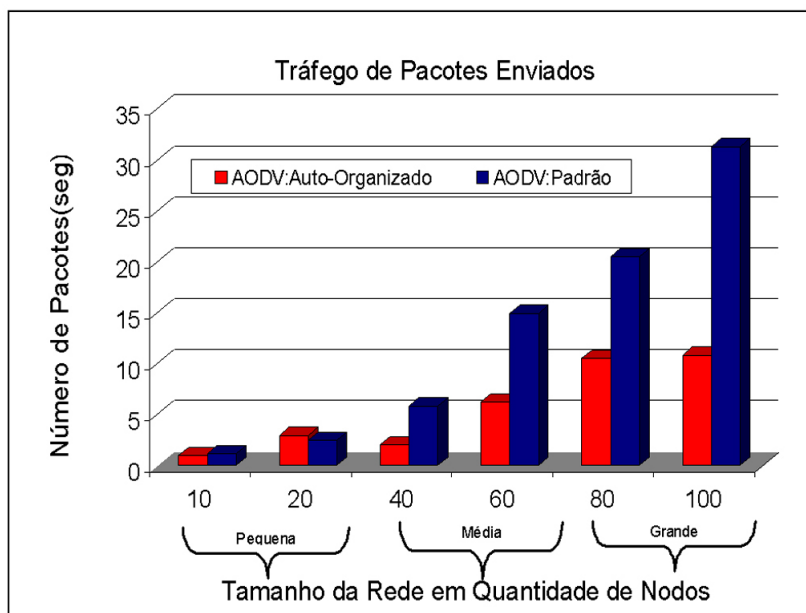


Figura 6.21: Comparação do tráfego de pacotes enviados entre AODV:Auto-Organizado e AODV:Padrão

A Figura 6.22 mostra que o protocolo AODV:Auto-Organizado obteve menor perda de pacotes em todas as escalas.

É interessante salientar que à medida que a rede cresce em tamanho, ou seja, em quantidade de dispositivos clientes e atinge as escalas média e grande, a perda de pacotes do AODV:Padrão é maior em comparação ao AODV:Auto-Organizado. Isso reforça que enquanto a rede cresce em tamanho há uma menor perda de pacotes proporcionada pelo protocolo AODV:Auto-Organizado. Nas Figuras 6.23 e 6.25 pode ser visualizado o total de requisições de rotas enviadas juntamente com o total de erros de rotas.

Na Figura 6.23, em particular, pode ser observado que o protocolo AODV:Auto-Organizado enviou um número menor de requisições de rotas, comparado ao protocolo AODV:Padrão. Com isso, a rede ganha em estabilidade.

Já na Figura 6.25 percebe-se um número menor de erros de rotas do protocolo AODV:Auto-Organizado em todas as escalas.

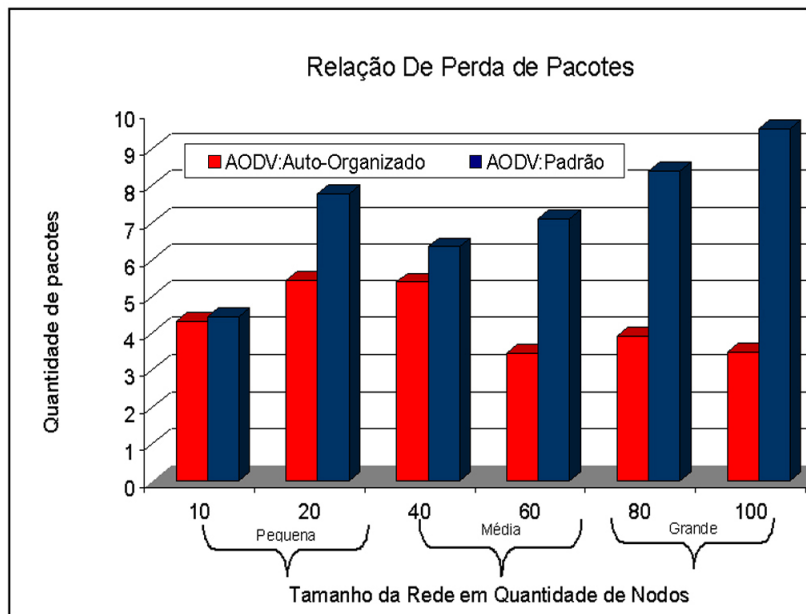


Figura 6.22: Comparação do número de pacotes descartados do protocolo AODV:Auto-Organizado e AODV:Padrão

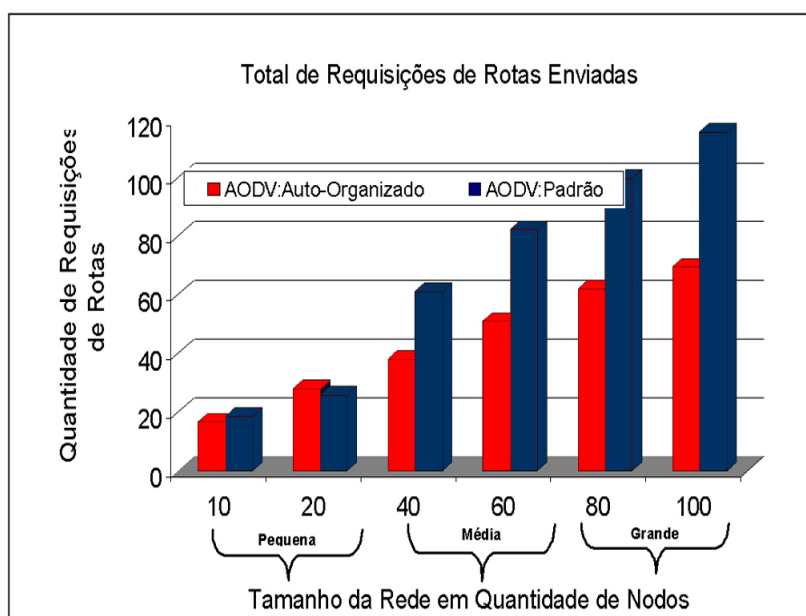


Figura 6.23: Comparação do total de requisições de rotas enviadas pelo AODV:Auto-Organizado e AODV Padrão

A Figura 6.24 apresenta o atraso entre os protocolos AODV:Auto-Organizado e o AODV:Padrão. Observa-se que o protocolo AODV:Auto-Organizado obteve um menor atraso de descoberta de rotas em todas as escalas.

Na Figura 6.26 nota-se o atraso de acesso ao meio, onde o protocolo AODV:Auto-Organizado obteve melhor performance em todas as escalas em comparação

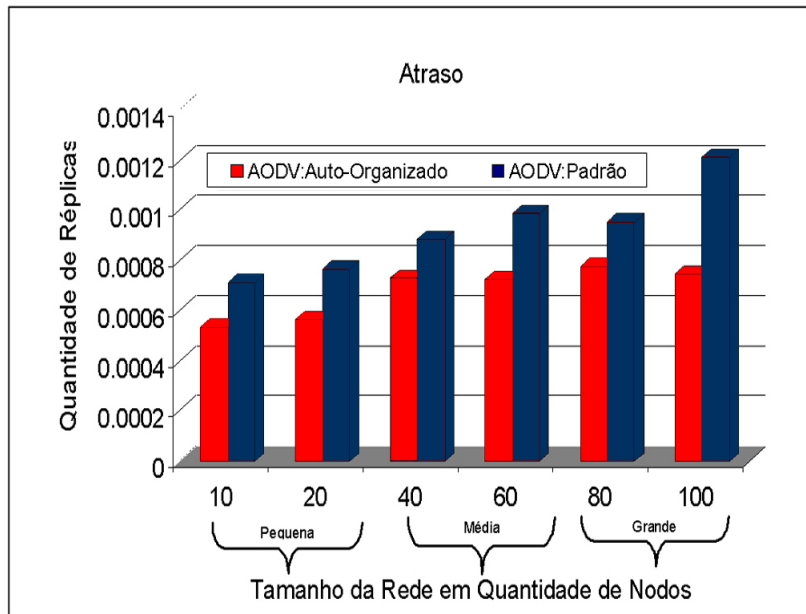


Figura 6.24: Comparação do atraso do protocolo AODV:Auto-Organizado e AODV:Padrão

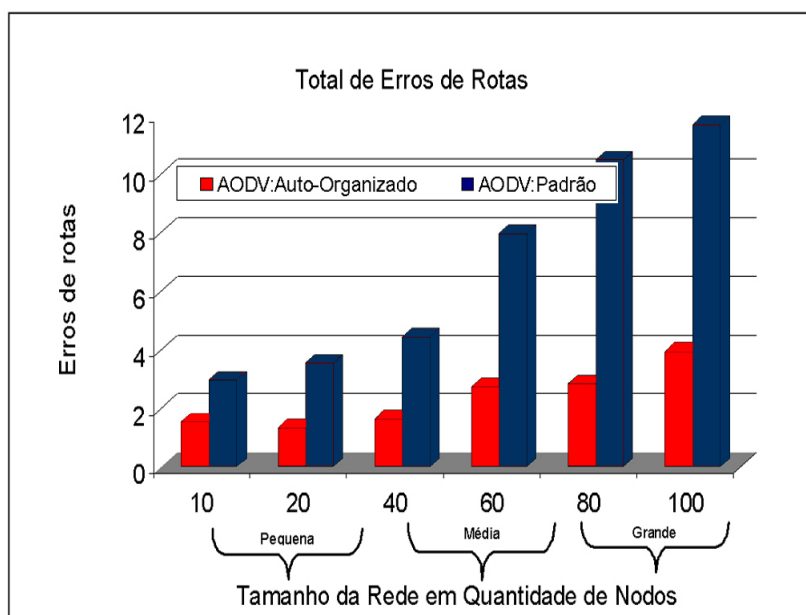


Figura 6.25: Relação do total de erros de rotas entre AODV:Auto-Organizado e AODV:Padrão

ao AODV:Padrão.

As Figuras 6.27 e 6.28 exibem a carga da rede e o tempo de descoberta da rota. Nota-se que a carga da rede usando o protocolo AODV:Auto-Organizado foi menor que o AODV:Padrão nas escalas média para grande e que na escala pequena o protocolo AODV:Padrão obteve uma melhor performance. Isso

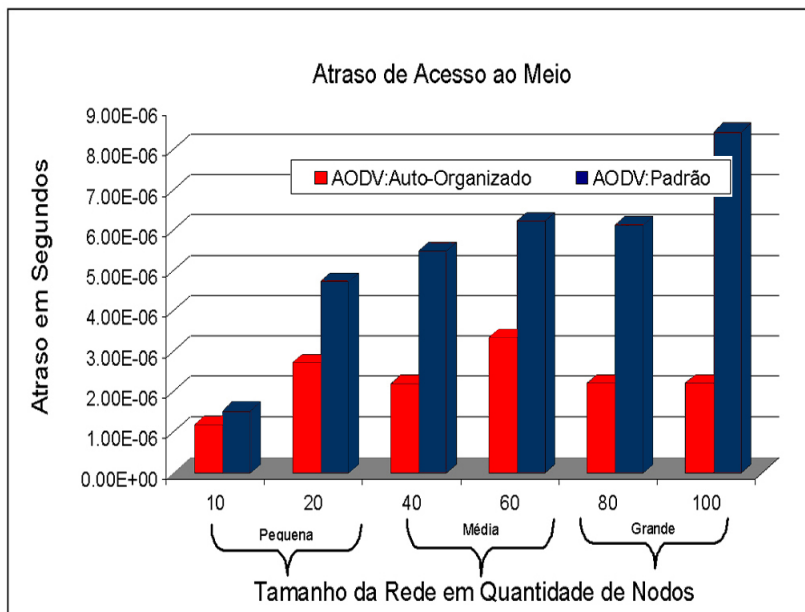


Figura 6.26: Comparação do atraso de acesso ao meio do Protocolo AODV:Auto-Organizado e AODV:Padrão

pode ser explicado devido ao valor configurado para verdadeiro (True) do parâmetro `grat_route_reply_flag` na escala pequena, conforme pode ser observado na Tabela 6.3.

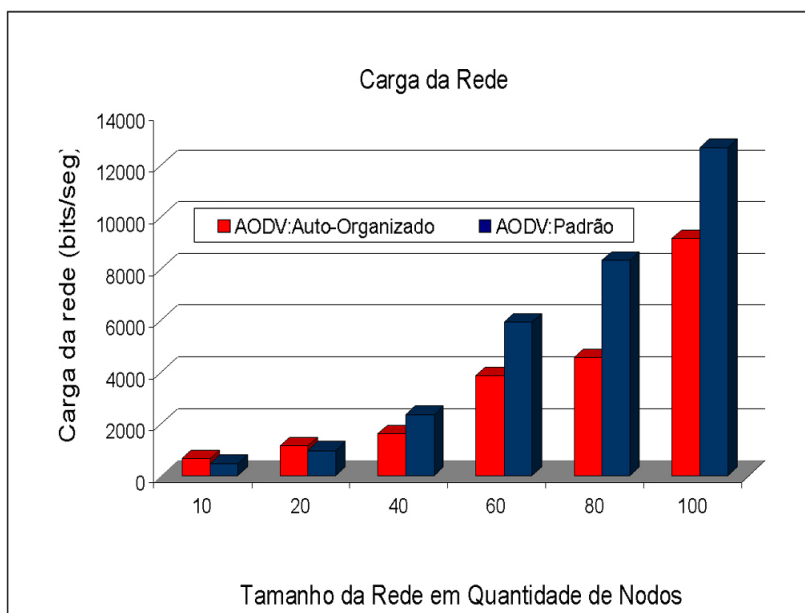


Figura 6.27: Comparação da carga da rede entre AODV:Auto-Organizado e AODV Padrão

Observa-se que se o parâmetro `grat_route_reply_flag` for configurado para verdade (True) este envia uma cópia do pacote de RREP (*Route Reply*) para

o nodo destino que enviou um RREQ(*Route Request*), com isso aumenta a confiabilidade na construção da rota mas em consequência aumenta a carga da rede na escala pequena devido a essa cópia do RREP na rede, como pode ser observado na Figura 6.27, mas em consequência a rede pode garantir uma maior confiabilidade das rotas. Com o parâmetro `grat_route_reply_flag` desabilitado (False) nas escalas média para grande observa-se que o protocolo AODV:Auto-Organizado garante uma menor carga de mensagens de RREP na rede.

Na Figura 6.28 visualiza-se o tempo de descoberta da rota, em que o protocolo AODV:Auto-Organizado obteve o melhor tempo em todas as escalas.

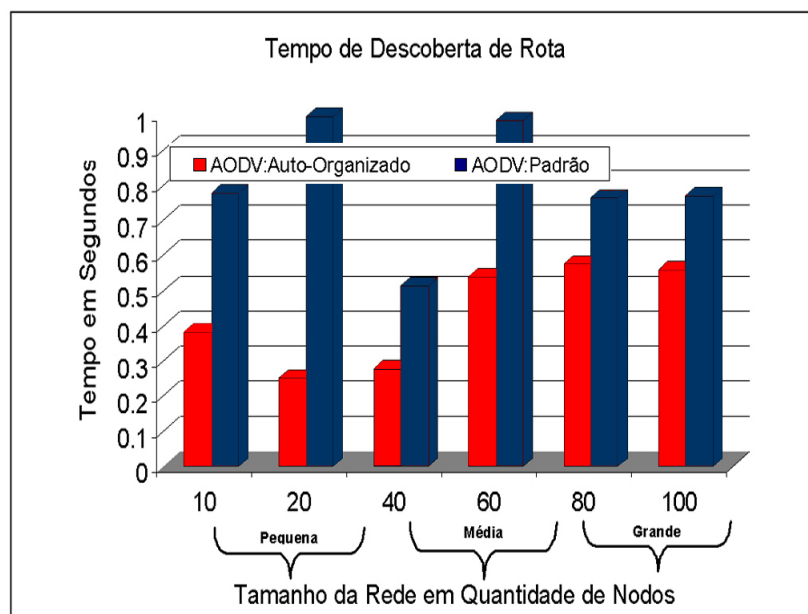


Figura 6.28: Comparação do tempo de descoberta de rota entre AODV:Auto-Organizado e AODV:Padrão

Neste experimento fez-se a comparação da vazão da rede entre o protocolo AODV:Auto-Organizado e AODV:Padrão. O AODV:Auto-Organizado obteve uma melhora significativa na vazão, conforme a escala da rede cresce de média para grande porte, enquanto que na escala pequena nota-se uma pequena diferença na vazão. Conforme analisado nas análises anteriores, pôde-se concluir que à medida que a rede cresce em tamanho, o protocolo AODV:Auto-Organizado obtém melhor desempenho em praticamente em todas as métricas analisadas.

Isso reforça a idéia que aplicando os conceitos de auto-otimização e auto-configuração nos protocolos de roteamento AODV e OLSR nas rede em malha

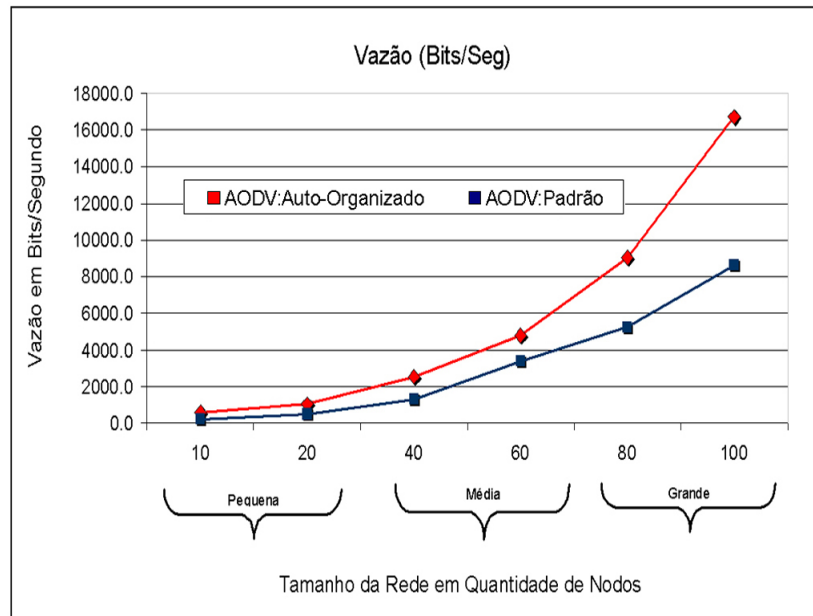


Figura 6.29: Comparação da vazão do protocolo AODV:Auto-Organizado e AODV:Padrão

sem fio, obtêm-se uma performance melhorada em ambientes em que a rede em malha sem fio pode crescer em quantidade de nodos clientes que entram na mesma.

6.5.3 Comparação entre OLSR:Auto-Organizado e AODV:Auto-Organizado

Além das análises específicas de cada protocolo, conforme visto nas seções anteriores, foram comparados os protocolos OLSR e AODV após aplicar as capacidades de Auto-Configuração e Auto-Otimização.

Na Figura 6.30 observa-se que o protocolo OLSR:Auto-Organizado obteve o melhor desempenho na escala pequena, enquanto que nas escalas média para grande o protocolo AODV:Auto-Organizado obteve o melhor desempenho. Este fato pode ser explicado em função das características de cada protocolo. Por exemplo, o protocolo AODV utiliza-se de demanda para encontrar as rotas para os nodos destinos, enquanto que o protocolo OLSR possui uma tabela de rotas pré-atualizada com as rotas para os nodos destinos.

Analisando por escalas, nota-se que à medida que a escala aumenta em tamanho, ou seja, quantidade de dispositivos clientes na rede, o protocolo OLSR gera uma maior sobrecarga, afetando assim a vazão de toda a rede. Já o protocolo AODV:Auto-Otimizado obtém as rotas por demanda, aumentando a vazão e

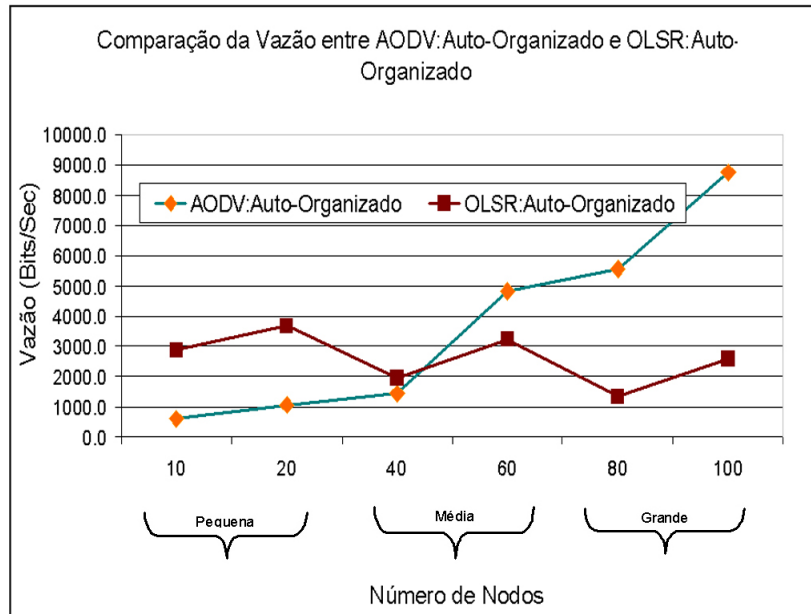


Figura 6.30: Relação da vazão entre OLSR:Auto-Organizado e AODV:Auto-Organizado

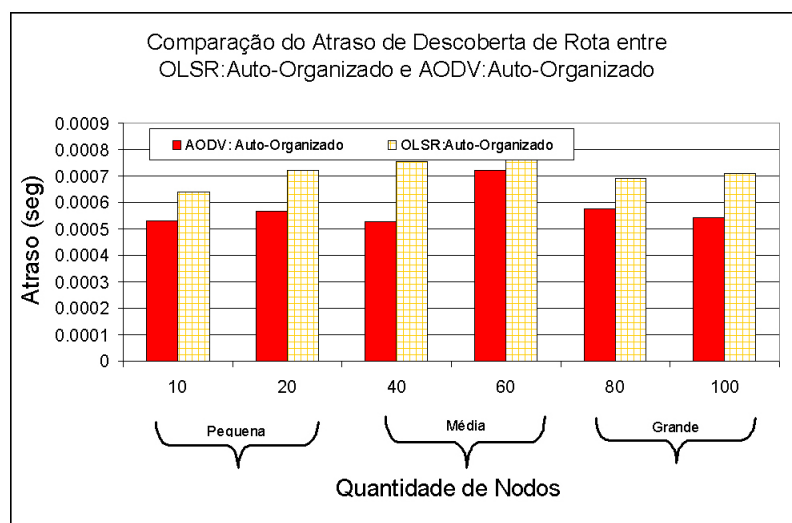


Figura 6.31: Relação do atraso de descoberta de rota entre OLSR:Auto-Organizado e AODV:Auto-Organizado

evitando a sobrecarga.

Concluiu-se neste experimento que o protocolo AODV:Auto-Organizado obtém melhor vazão quando a escala da rede passa de média para grande porte.

O atraso de descoberta da rota é mensurado na Figura 6.31. Observa-se que o protocolo AODV:Auto-Organizado obteve um menor atraso em todas as escalas comparado ao OLSR:Auto-Organizado. É interessante ressaltar que o protocolo AODV utiliza-se da técnica por demanda para descobrir os nodos

destino e isso gera um maior atraso que a técnica que o protocolo OLSR utiliza. Conclui-se neste experimento que o protocolo AODV:Auto-Organizado obteve um menor atraso em comparação ao OLSR:Auto-Organizado.

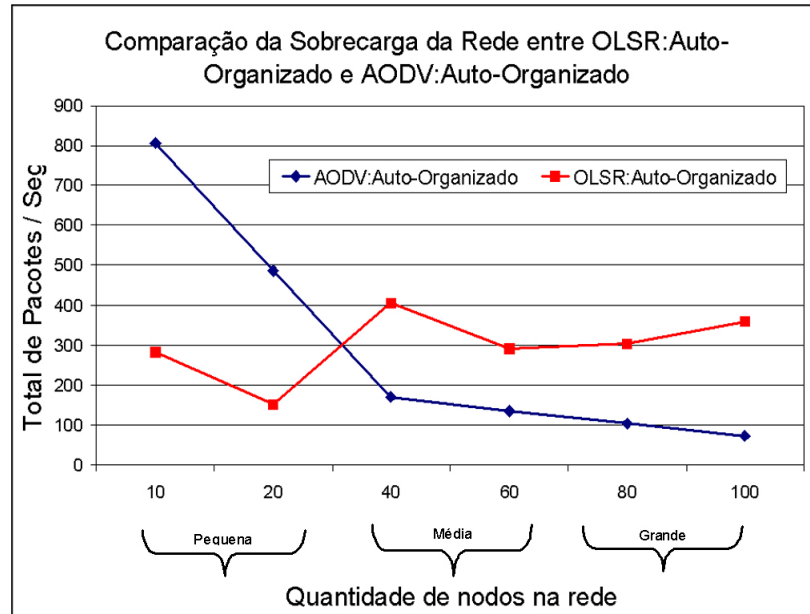


Figura 6.32: Relação da sobrecarga entre OLSR:Auto-Organizado e AODV:Auto-Organizado

A sobrecarga da rede é analisada na Figura 6.32, onde pode ser observado que o protocolo AODV:Auto-Organizado obteve maior sobrecarga para encontrar as rotas na escala pequena. O protocolo OLSR:Auto-Organizado possui uma menor sobrecarga na escala pequena, porém quando essa escala passa de média para grande, observa-se que a sobrecarga de pacotes do protocolo AODV:Auto-Organizado diminui e a do OLSR:Auto-Organizado aumenta.

Conclui-se neste experimento que o protocolo AODV:Auto-Organizado adapta-se melhor em redes em malha sem fio de médio e grande porte.

6.6 Algumas Considerações

Neste capítulo foram apresentadas a modelagem e a simulação utilizadas para validar o presente trabalho. Inicialmente foram simuladas as capacidades de auto-otimização e auto-configuração aplicadas aos protocolos de roteamento OLSR e AODV. Como resultado obteve-se uma melhora significativa no desempenho desses protocolos, para redes em malha sem fio de pequeno, médio e grande porte. As métricas propostas para análise desse desempenho

mostraram em primeira instância que as capacidades de auto-configuração e auto-otimização, aplicadas aos protocolos citados, melhoram o desempenho da rede em malha sem fio em comparação à abordagem original, que utiliza os parâmetros padrões dos protocolos. Por fim, conforme as análises de comparações entre os protocolos OLSR:Auto-Organizado com o AODV:Auto-Organizado, observou-se uma melhor utilização do protocolo AODV:Auto-Organizado em ambientes em que a rede em malha sem fio, alterna entre as escalas média e grande porte. Como resultados demonstrou-se uma melhora significativa no desempenho dos protocolos OLSR e AODV após implementação das capacidade de auto-otimização e auto-configuração propostas neste trabalho. Observou-se também uma redução das mensagens de manutenção e controle de rotas em ambos os protocolos, contribuindo para uma menor sobrecarga geral da rede.

7 *Conclusões e Trabalhos Futuros*

Neste trabalho foi proposta uma arquitetura para auto-organização em redes em malha sem fio utilizando como base os protocolos de roteamento pró-ativo OLSR e reativo AODV, juntamente com a tecnologia de agentes de software, para auxiliar as capacidades de auto-organização (Auto-Otimização e Auto-Configuração) no ajuste dos valores dos parâmetros dos protocolos de roteamentos supracitados. Para tal, foi proposta a divisão do tamanho da rede em escalas.

Notou-se que ao revisar a literatura especializada, que ainda existem poucos trabalhos que focam a questão da auto-organização para redes em malha sem fio. A arquitetura proposta ainda envolve a definição de conceitos oriundos da computação autônoma, integrados à tecnologia de agentes de software.

O algoritmo que verifica a escala da rede em malha sem fio foi acoplado na capacidade de Auto-Configuração, enquanto que a capacidade de Auto-Otimização obtém um valor otimizado e este foi colocado dentro dos cinco parâmetros dos protocolos OLSR (Hello-Interval, Tc-Interval, Neighbor-Hold-Time, Tc-Hold-Time, Duplicate-Hold-Time) e AODV (Route-Request-Retries, Hello-Interval, Active-Route-Timeout, Allowed-Hello-Loss, Net-Diameter). Utilizou-se de otimização não linear sem restrições para adequar a função objetivo cotangente hiperbólica ao comportamento da análise da variação do intervalo dos parâmetros citados. Esse valor otimizado foi diretamente adicionado nesses parâmetros conforme a escala em que a rede estava em um determinado momento.

Comparando-se as capacidades de auto-otimização e auto-configuração implementadas nesta dissertação, aos trabalhos de Huang et al. (2006) Gomez et al. (2005) (MALATRAS et al., 2005). Pode-se concluir que as capacidades implementadas obtiveram melhor performance nas RMSF. Além que analisamos cinco parâmetros de cada protocolo de roteamento, enquanto que, os trabalhos de Huang et al. (2006) Gomez et al. (2005) analisaram a taxa de mensagens de HELLO na rede que é configurada através do parâmetro Hello-Interval.

Já o trabalho de (MALATRAS et al., 2005) somente analisou a capacidade de auto-configuração e auto-otimização em redes MANET, e não em RMSF.

Para as escalas optou-se em dividir a rede em malha sem fio em três tamanhos: pequena (SMALL), média (MEDIUM) e grande (LARGE). O tamanho da rede foi obtido através do agente de software denominado NS-Agent, que é responsável pelo processo de coleta do tamanho da rede e posteriormente, por enviar essa informação para os demais nodos roteadores. Observou-se que as capacidades de Auto-Otimização e Auto-Configuração propostas neste trabalho possuem forte correlação, pois uma é complemento da outra: enquanto uma otimiza, a outra configura os valores nos parâmetros dos protocolos citados.

A descrição da arquitetura em nível conceitual com todas as capacidades (Auto-Otimização, Auto-Configuração, auto-cura e auto-proteção), como apresentado no capítulo 5 desta dissertação, foi importante pois mostrou o escopo global de todo o trabalho.

Com todas as capacidades implementadas, espera-se que a rede em malha sem fio possa ter:

- Maior desempenho, aplicando-se as capacidades de Auto-Otimização e Auto-Configuração;
- Alta tolerância a falhas, aplicando-se a capacidade de auto-cura;
- Mais segurança a ataques e intrusões, aplicando-se a capacidade de auto-proteção.

Nesta dissertação enfatizou-se o desempenho com a implementação das capacidades de Auto-Otimização e Auto-Configuração, com a implementação dos agentes NS-Agents.

Verificando os resultados das simulações observou-se uma melhora significativa do desempenho da rede em malha sem fio, utilizando as capacidades de Auto-Otimização e Auto-Configuração acopladas nos protocolos de roteamento OLSR e AODV.

As simulações ainda mostraram resultados específicos nos protocolos OLSR:Auto-Organizado e AODV:Auto-Organizado, dentre os quais pode-se citar.

- Melhora na vazão da rede em malha sem fio;
- Melhora na entrega dos pacotes, pois percebeu-se uma menor perda de pacotes;

- Menor atraso da descoberta das rotas entre os nodos vizinhos;
- Menor injeção de mensagens de controle e manutenção na rede.

Também pôde ser constatado que o protocolo AODV:Auto-Organizado obteve melhor desempenho comparado ao protocolo OLSR:Auto-Organizado, enquanto a escala da rede em malha sem fio mudava de média para grande porte.

É valido ressaltar, que a arquitetura proposta está ainda em fase de desenvolvimento, faltando acoplar as capacidades de auto-cura e auto-proteção juntamente com o agente NM-Agent. Esforços devem ser feitos para que futuramente a arquitetura possa se tornar uma referência em auto-organização para redes em malha sem fio.

7.1 Limitações

No desenvolvimento deste trabalho observou-se algumas limitações que devem ser enfatizadas:

- Os protocolos OLSR e AODV não possuem suporte para simular redes em malha sem fio múltiplos-rádios, portanto neste trabalho foram simulados às redes em malha sem fio únicos rádios, que usam a mesma interface de rede tanto para formar o *backbone* da rede quanto para prover a comunicação para os nodos clientes;
- Devido a um problema de compatibilidade na API, que faz a integração do software Matlab com o simulador Opnet, os valores otimizados pelo software Matlab são acoplados de forma manual dentro das escalas da capacidade de Auto-Configuração. O ideal seria que essa análise, juntamente com a integração desses valores, fossem feitas de forma automática.

7.2 Trabalhos Futuros

No desenvolvimento deste trabalho observou-se uma gama de possibilidades para contribuição e aperfeiçoamento da arquitetura de auto-organização para redes em malha sem fio, dentre as quais pode-se citar:

- Necessidade de validar a arquitetura proposta empregando-a em um ambiente real;

- Implementar nos protocolos AODV e OLSR, uma extensão para que possam ser simuladas redes em malha sem fio múltiplos-rádios;
- Desenvolver as capacidades restantes (auto-cura e auto-proteção), juntamente com a integração do agente de monitoramento NM-Agent.
- Estender a arquitetura de auto-organização para funcionar em outros protocolos de roteamento, tais como: DSR, TORA, DSDV, WRP, dentre outros;
- Necessidade de aperfeiçoar o modelo de Auto-Otimização, aplicando uma técnica mais genérica para análise do comportamento do crescimento da rede em malha sem fio;

7.3 Considerações Finais

Por fim, todo contexto deste trabalho possibilitou um aprendizado abrangente em diversos aspectos referentes às redes em malha sem fio e às capacidades de auto-organização (Auto-Otimização, Auto-Configuração, Auto-Cura e Auto-Proteção), permitindo que os conhecimentos adquiridos fossem repassados para o projeto e concepção da arquitetura proposta.

Referências

4G-SYSTEMS. *meshcube - 4G Access Cube*. 2007. Acessado em 14 de junho. Disponível em: <<http://meshcube.org/>>.

802.11, I. *LAN/MAN Wireless LANS*. Junho 1997. Acessado em 12 março 2007. Disponível em: <<http://standards.ieee.org/getieee802/802.11.html>>.

ABELÉM, A. J. G. et al. Redes mesh: Mobilidade, qualidade de serviço e comunicação em grupo. In: _____. [S.l.]: Livro de Minicursos: SBRC 2007, 2007. cap. 2, p. 59–112.

ABOLHASAN, M.; WYSOCKI, T.; DUTKIEWICZ, E. A review of routing protocols for mobile ad hoc networks. **Ad Hoc Networks (Elsevier)**, p. 1–22, Junho 2004.

ADYA, A. et al. A multi-radio unification protocol for ieee 802.11 wireless networks. **Broadband Networks. BroadNets 2004. Proceedings**, v. 1, p. 344 – 354, Outubro 2004.

AGUAYO, D. et al. **MIT Roofnet: Construction of a Community Wireless Network**. [S.l.], 2004.

AKYLDIZ, I. F.; WANG, X.; WANG, W. Wireless mesh networks: a survey. **Computer Networks**, v. 47, n. 4, p. 445–487, Março 2005.

ANA, P. **ANA: Autonomic Network Architecture**. 2006. Acessado em 12 de outubro de 2007. Disponível em: <<http://www.ana-project.org/>>.

ASBHY, W. R. Principles of the self-organizing dynamic system. **General Psychology**, v. 37, p. 125–, 1947.

ASHBY, W. R.; FOERSTER, H. V.; ZOPF, G. W. Principles of the Self Organizing System. **Pergamon Press, London, UK**, v. 6, p. 255–278, 1962.

AUTONOMIC, C. P. Novembro 2007. Acessado em 12 de novembro de 2007. Disponível em: <<http://autonomiccomputing.org/projects.php>>.

BABAOGLU, O. et al. **SelfStar Properties in Complex Information Systems**. [S.l.]: Springer, 2005.

BAHR, M. Update on the hybrid wireless mesh protocol of ieee 802.11s. In: **Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE International Conference on**. [S.l.: s.n.], 2007. p. 1–6.

BALASUBRAMANIYAN, J. S. et al. An architecture for intrusion detection using autonomous agents. In: **Proceedings of the 14th Annual Computer Security Applications Conference**. [S.l.]: IEEE Computer Society, 1998. p. 13. ISBN 0-8186-8789-4.

- BASH, L. M.; JELGER, C.; TSCHUDIN, C. A case study in designing an autonomic wireless mesh network. **7th Scandinavian Workshop on Wireless Ad-hoc Networks - ADHOC, Stockholm, Sweden**, p. 1–5, Maio 2007.
- BAUMANN, R. et al. Routing packets into wireless mesh networks. In: **Wireless and Mobile Computing, Networking and Communications, 2007. WiMOB 2007. Third IEEE International Conference on**. [S.l.: s.n.], 2007. p. 38–38.
- BELAIR, N. **Security Solutions**. 2007. Acessado em 26/09/2007. Disponível em: <<http://www.belairsecurity.com/index.html>>.
- BIESZCZAD, A.; PAGUREK, B.; WHITE, T. Mobile agents for network management. **IEEE Communications Surveys**, 1998. Disponível em: <<http://citeseer.ist.psu.edu/bieszczad98mobile.html>>.
- BISON. **BISON: Biology-Inspired techniques for Self-Organization in dynamic Networks**. Abril 2006. Acessado em 12 agosto 2007.
- CAMAZINE, S. et al. **Self-Organization in Biological Systems**. [S.l.]: Princeton: Princeton University Press, 2003.
- CAMPISTA, M. E. M.; COSTA, L. H.; DUARTE, O. C. M. B. Um algoritmo eficiente de disseminação dos estados de enlace para redes em malha sem fio. **SBRC - Simpósio Brasileiro de Rede de Computadores**, Maio 2007.
- CAMPONOGARA, E. **Métodos de Otimização Teoria e Prática - Versão Preliminar**. [S.l.], Setembro 2006.
- CHANG, X. Network simulator with opnet. **Proceedings of the 1999 Winter Simulation Conference**, p. 307–314, 1999.
- CLAUSEN, T.; JACQUET, P.; LAOUITI, A. **Optimized Link State Routing Protocol (OLSR)**. [S.l.], October 2003.
- COLLIER, T. C.; TAYLOR, C. Self-organization in sensor networks. **Journal of Parallel and Distributed Computing**, v. 64, p. 866 – 873, 2004.
- COUTINHO, A. A. da C. **Wireless Mesh Network Múltiplos-Rádios e Múltiplos-Canais**. [S.l.], 2006.
- COUTO, D. S. J. D.; BICKET, D. A. J.; MORRIS, R. A high-throughput path metric for multi-hop wireless routing. **Proceedings of the 9th annual international conference on Mobile computing and networking, San Diego, CA, USA**, p. 134 – 146, September 2003.
- CUWIN, W. S. **Cuwin**. 2007. Acessado em 23/09/2007. Disponível em: <<http://www.cuwireless.net/download>>.
- DIJKSTRA, E. W. Recursive programming. **Numerische Mathematik** 2, p. 312318, 1960.
- DORNIER, P. **WRAP - Wireless Router Application Platform**. 2002. Disponível em: <<http://www.pcengines.ch/wrap.htm>>.
- DRESSLER, F. **Self-Organization in Ad Hoc Networks: Overview and Classification**. [S.l.], fevereiro 2006.

- DUKE, R. et al. **The Object-Z Specification Language**. [S.l.], 1991. v. 1.
- DUTTA, P.; JAISWAL, S.; RASTOGI, R. Routing and channel allocation in rural wireless mesh networks. **INFOCOM 2007. 26th IEEE International Conference on Computer Communications**. IEEE, p. 598–606, May 2007.
- FLICKENGER, R. **Building Wireless Community Networks**. [S.l.]: O'Reilly, 2002.
- FREITAS, P. J. F. **Introdução à Modelagem e Simulação de Sistemas**. [S.l.]: Visual Books, 2001. 320 p.
- GALLEGO, D. M. et al. Capacity and qos of wireless mesh networks. **4th International Information and Telecommunication technologies Symposium**, Dezembro 2005.
- GOMEZ, C.; GARCIA, D.; PARADELLS, J. Improving performance of a real ad-hoc network by tuning olsr parameters. In: **Computers and Communications, 2005. ISCC 2005. Proceedings. 10th IEEE Symposium on**. [S.l.: s.n.], 2005. p. 16–21.
- GONZÁLEZ, P. C. et al. An ip-video surveillance indoor system using mesh networks. in: 6th international information and telecommunication technologies symposium. **Proceedings of the I2TS 2007**, 2007.
- GRADES zur Erlangung des. **Modeling and Simulation of a Routing Protocol for Ad Hoc Networks Combining Queuing Network Analysis and Ant Colony Algorithms**. Tese (Doutorado) — Institut für Informatik und Wirtschaftsinformatik, Abril 2005.
- GUARDALBEN, L. et al. A performance evaluation of olsr and aodv routing protocols using a self-configuration mechanism for heterogeneous wireless mesh networks. In: **Local Computer Networks, 2008. LCN 2008. 33rd IEEE Conference on**. [S.l.: s.n.], 2008. p. 697–704.
- GUARDALBEN, L. et al. A self-x approach for olsr routing protocol in large-scale wireless mesh networks. In: **Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE**. [S.l.: s.n.], 2008. p. 1–6.
- GUARDALBEN, L.; M., S. J. B.; TAMASHIRO, C. H. Towards an architecture for self-organization in wireless mesh networks. **Anais da XXXIII Conferência Latinoamericana de Informática - CLEI**, p. 51–52, 2007.
- GUARDALBEN, L.; M., S. J. B.; TAMASHIRO, C. H. Uma arquitetura para auto-organização em redes em malha sem fio. **XXV Simpósio Brasileiro de Telecomunicações**, p. 51–52, 2007.
- HAGGLE. A novel communication paradigm for autonomic opportunistic communication. Janeiro 2006.
- HARRISON, C.; CHESS, D.; KERSHENBAUM, A. Mobile agents: Are they a good idea? **Second International Workshop on Mobile Object Systems, MOS '96**, Springer-Verlag, v. 1222, p. 25–47, July 1997.

- HELD, G. **Wireless Mesh Networks**. [S.l.]: Auerbach, 2005. 160 p.
- HERRMANN, K. **Self-Organizing Infrastructure for Ambient Services**. Tese (Doutorado) — Universidade de Berlim - Instituto de Engenharia, Julho 2006.
- HERRMANN, K.; MÜHL, G.; GEIHS, K. Self management: The solution to complexity or just another problem? **IEEE Distributed Systems Online**, v. 6, n. 1, p. 1–17, Janeiro 2005.
- HEYLIGHEN, F.; GERSHENSON, C. The meaning of self-organization in computing. **IEEE Intelligent Systems: Section Trends & Controversies - Self-organization and Information Systems**, Maio 2003.
- HORN, P. **Autonomic Computing: IBM Perspective on the State of Information Technology**. Outubro 2001.
- HUANG, Y.; BHATTI, S.; HANDURUKANDE, S. Autonomic tuning of routing for manets. In: **Network Operations and Management Symposium Workshops, 2008. NOMS Workshops 2008. IEEE**. [S.l.: s.n.], 2008b. p. 314–320.
- HUANG, Y.; BHATTI, S.; PARKER, D. Tuning olsr. In: **Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on**. [S.l.: s.n.], 2006. p. 1–5.
- HUANG, Y.; BHATTI, S.; SORENSEN, S.-A. Self-tuning network support for manets. In: **Network Operations and Management Symposium, 2008. NOMS 2008. IEEE**. [S.l.: s.n.], 2008a. p. 1037–1042.
- INFO, R. **Reportagem: O Mesh faz a Rede**. Dezembro 2005.
- JACQUET, P. et al. Optimized link state routing protocol for ad hoc networks. In: **Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International**. [S.l.: s.n.], 2001. p. 62–68.
- JELASITY, M.; BABAOGU, O.; LADDAGA, R. Self-management through self-organization. **IEEE Computer Society: Intelligent Systems**, Março 2006.
- JELGER, C.; TSCHUDIN, C.; SCHMID, S. Basic abstractions for an autonomic network architecture. **World of Wireless, Mobile and Multimedia Networks (WoWMoM) in IEEE International Symposium**, p. 1–6, Junho 2007.
- JOHNSON, S. **Emergência - A dinâmica de Rede em Formigas, Cérebros, Cidades e Softwares**. 3. ed. [S.l.]: Jorge Zahar, 2003. 217 p.
- JUNIOR, O. P. **Auto-Organização e Complexidade: Uma Introdução Histórica e Crítica**. [S.l.], Janeiro 2001.
- KEPHART, J. O.; CHESS, D. M. The Vision of Autonomic Computing. In: **IEEE Computer Networks 36**. [S.l.: s.n.], 2003. v. 1, p. 41–50.

- KEPHART, J. O.; WALSH, W. E. An artificial intelligence perspective on autonomic computing policies. **Proceedings of Policy 2004**, p. 1–10, 2004.
- LE, A. N. et al. Directional aodv routing protocol for wireless mesh networks. In: **Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on**. [S.l.: s.n.], 2007. p. 1–5.
- LEE, A.; WARD, P. A. **A Study of Routing Algorithms in Wireless Mesh Networks**. [S.l.], Maio 2006. Department of Electrical and Computer Engineering, documento técnico - University of Waterloo - Ontario - CA.
- LINKSYS. **Wireless-G Broadband Router (Wrt54G)**. 2007. Acessado em 20/06/2007. Disponível em: <<http://www.linksys.com/products>>.
- LUCIO, G. F. et al. Opnet modeler and ns-2: Comparing the accuracy of network simulators for packet-level analysis using a network testbed. **WSEAS - World Scientific and Engineering Academic and Society**, v. 2, p. 700–707, Julho 2003.
- MACHADO, R. B. et al. A hybrid artificial immune and mobile intrusion detection based model for computer network operations. **International Parallel & Distributed Processing Symposium**, 2005.
- MAHMUD, S. A. et al. A comparison of manets and wmns: Commercial feasibility of community wireless networks and manets. **AccessNets**, p. 1–7, Setembro 2006.
- MALATRAS, A. et al. Self-configuring and optimizing mobile ad hoc networks. In: **Autonomic Computing, 2005. ICAC 2005. Proceedings. Second International Conference on**. [S.l.: s.n.], 2005. p. 372–373.
- MALHEIROS, N.; MADEIRA, E. **Autonomic Wireless Mesh Networks: Uma Arquitetura para Redes Mesh Sem Fio Auto-Gerenciáveis**. Dezembro 2006. Workshop de Teses e Dissertações: Instituto de Computação Universidade Estadual de Campinas. Acessado em 12 março. Disponível em: <<http://solimoes.ic.unicamp.br:8080/wtd/apresentacoes/malheiros>>.
- MAMEI, M. et al. Case studies for self-organization in computer science. **Journal of Systems Architecture**, v. 52, p. 443–460, Abril 2006.
- MANZALINI, A.; ZAMBONELLI, F. Towards autonomic and situation-aware communication services: the cascadas vision. **Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS)**, p. 383–388, 2006.
- MATHWORLD, W. **Hyperbolic Cotangent**. 1999. Disponível em: <<http://mathworld.wolfram.com/HyperbolicCotangent.html>>.
- MAUVE, M.; WIDMER, J.; HARTENSTEIN, H. A survey on position-based routing in mobile ad hoc networks. **IEEE Network**, v. 15, p. 30–39, Novembro 2001.
- MEER, H. de; WÜCHNER, P.; HOUYOU, A. Self-organizing systems: New trends in architectures and performance modeling. In: **International Workshop on Self-Organizing Systems (IWSOS)**. [S.l.: s.n.], 2006.

- MESH. **Wireless Mesh Network Testbed**. Julho 2007. Acessado em 30/09/2007. Disponível em: <<http://www.cse.msu.edu/wangbo1/testbed/doku.php?id=mesh>>.
- MESH alliance. **Wi-Mesh Proposal for IEEE 802.11s**. 2006. Disponível em: <<http://wi-mesh.org/>>.
- MESHLINUX. **Linux distribution geared to build mesh super-nodes**. 2007. Acessado em 22/09/2007. Disponível em: <<http://open-mesh.net/meshlinux>>.
- MESHNETWORKS. **Meshnetworks Protocol**. 2007. Disponível em: <<http://www.meshnetworks.com>>.
- MICROSOFT. **Self-Organizing Wireless Mesh Networks** . <http://research.microsoft.com/mesh/> 2005.
- MILLER, B. **The autonomic computing edge: Can you CHOP up autonomic computing?** Janeiro 2005. Acessado em 15 de setembro. Disponível em: <<http://www.ibm.com/developerworks/autonomic/library/acedge4/>>.
- MITRE. **Corporation: Mobile Mesh**. http://www.mitre.org/work/tech_transfer/mobilemesh/ 2005. Acessado em 12/02/2007.
- MOTOROLA. **Multi-Radio Broadband Solution**. <http://www.motorola.com/mesh/pages/products/motomesh.htm> 2005.
- MOTOROLA. **Motorola Canopy HotZone**. 2007. Acessado em 12/06/2007. Disponível em: <<http://motorola.canopywireless.com/products/hotzone/>>.
- MOTOROLA. **Motorola HotZone Duo Mesh Wireless Router (MWR)**. 2007. Acessado em 26 de Maio.
- MRTG. **The Multi Router Traffic Grapher**. 2005. Disponível em: <<http://oss.oetiker.ch/mrtg/>>.
- MULLER, H. A. et al. **Autonomic Computing: Software Architecture Technology**. [S.l.], Abril 2006.
- OETIKER, T.; RAND, D. **Multi Router Traffic Grapher (MRTG)**. 2008. Acessado em 13 de janeiro de 2008. Disponível em: <<http://www.mrtg.com/>>.
- OLEXA, R. **Implementing 802.11, 802.16 and 802.20 Wireless Networks**. [S.l.]: Newnes, 2005.
- OPENWRT. **OpenWRT Wireless Freedom**. 2005. Acessado em 23/09/2007. Disponível em: <<http://openwrt.org/>>.
- OPNET, T. **Modeler Accelerating Network R&D**. 1987. Acessado em 05/08/2007. Disponível em: <http://www.opnet.com/solutions/network_rd/modeler.html>.
- OPNET, T. **Opnet Solutions**. 1987. Acessado em 15/08/2007. Disponível em: <http://www.opnet.com/university_program/research_with_opnet/>.

- PARASHAR, M.; HARIRI, S. **Autonomic Computing: Concepts, Infrastructure and Applications**. [S.l.]: CRC Press, 2006. 539 p.
- PASSOS, D. et al. Mesh network performance measurements. **Proceedings of the 5th International Information and Telecommunication Technologies Symposium**, p. 48–55, Dezembro 2006.
- PIRZADA, A.; PORTMANN, M. High performance aodv routing protocol for hybrid wireless mesh networks. In: **Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on**. [S.l.: s.n.], 2007. p. 1–5.
- PREHOFER, C.; BETTSTETTER, C. Self-organization in communication networks: Principles and design paradigms. In: **Advances in Self-Organizing Networks**. [S.l.]: IEEE Communications Magazine, 2005.
- QUAGGA. **Routing Suite**. 2007. Acessado em 23/09/2007. Disponível em: <<http://www.quagga.net/commercial.php>>.
- RAIBULET, C.; DEMARTINI, C. Mobile agent technology for the management of distributed systems a case study. **Comput. Networks**, Elsevier North-Holland, Inc., v. 34, n. 6, p. 823–830, 2000. ISSN 1389-1286.
- RANIWALA, A.; CHIUH, T. Architecture and Algorithms for an IEEE 802.11-Based Multi-Channel Wireless Mesh Network. **24th Annual Joint Conference of the IEEE Computer and Communications Societies (Infocom). Proceedings IEEE**, v. 3, p. 2223–2234, Março 2005.
- REMESH. **Rede Mesh de Acesso Universitário Faixa Larga Sem Fio**. 2006. Disponível em: <<http://mesh.ic.uff.br/>>.
- ROOFNET. **MIT Roofnet**. 2005. Disponível em: <<http://pdos.csail.mit.edu/roofnet/doku.php>>.
- ROYER, E.; TOH, C.-K. A review of current routing protocols for ad-hoc mobile wireless networks. **IEEE Personal Communications Magazine**, v. 6, p. 46–65, 1999.
- SARMA, A.; BETTSTETTER, C.; DIXIT, S. **Self-Organization in Communication Networks**. [S.l.]: Technologies for the Wireless Future: Wireless World Research Forum (WWRF), 2006. 423-451 p.
- SCHMID, S.; SIFALAKIS, M.; HUTCHISON, D. Towards autonomic networks. In **proceedings of 3rd Annual Conference on Autonomic Networking, Autonomic Communication Workshop (IFIP AN/WAC), Paris France**, Setembro 2006.
- SEN, S.; RAMAN, B. Long distance wireless mesh network planning: Problem formulation and solution. **WWW 2007 / Track: Technology for Developing Regions**, May 2007.
- SOMAYAJI, A.; HOFMEYR, S.; FORREST, S. Principles of a computer immune system. In: DEPARTMENT OF COMPUTER SCIENCE, NEW MEXICO UNIVERSITY, ALBUQUERQUE, NM, USA. **Meeting on New Security Paradigms, 23-26 Sept. 1997, Langdale, UK**. New York, NY, USA: ACM, 1998. p. 75–82.

SUBRAMANIAN, A. P.; BUDDHIKOT, M. M.; MILLER, S. Interference aware routing in multi-radio wireless mesh networks. **Wireless Mesh Networks, 2006. WiMesh 2006. 2nd IEEE Workshop Multi-Hop**, p. 55–63, 2006.

SUBRAMANIAN, A. P.; GUPTA, H.; DAS, S. R. Minimum interference channel assignment in multi-radio wireless mesh networks. **Proceedings of the Fourth Annual IEEE Communications Society Conference on Sensor, Mesh, and Ad Hoc Communications and Networks (SECON)**, p. 481–490, 2007.

TAMASHIRO, C. H. **Uma Análise de Protocolos de Roteamento Anônimo para Redes Sem Fio Ad Hoc Móveis**. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, Setembro 2007.

TAN, H.; SEAH, W. Dynamically adapting mobile ad hoc routing protocols to improve scalability. **Communication Systems and Networks**, p. 1–6, 2004.

TANENBAUM, A. S. **Computer Networks**. [S.l.]: Prentice Hall, 1996. 813 p.

TANENBAUM, A. S. **Redes de Computadores**. 4. ed. [S.l.]: Campus, 2003. 903 p.

TANG, H.; TIANFIELD, H. Self-Organizing Networks of Communications and Computing. In: **International Transactions on Systems Science and Applications**. [S.l.: s.n.], 2006. v. 1, n. 4, p. 421–431.

TOWSLEY, D. Mobility models for wireless networks: Challenges, pitfalls, and successes. In: **Principles of Advanced and Distributed Simulation, 2008. PADS '08. 22nd Workshop on**. [S.l.: s.n.], 2008. p. 3–3.

TSARMPOPOULOS, N.; KALAVROS, I.; LALIS, S. A low-cost and simple-to-deploy peer-to-peer wireless network based on open source linux routers. **Proceedings of the First International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMmunities (TRIDENTCOM-05)**, p. 92– 97, Fevereiro 2005.

TSCHUDIN, C. et al. **ANA Project: Autonomic Network Architecture**. [S.l.], Dezembro 2006.

VASUDEVAN, S. **Self-Organization in Large-Scale Wireless Networks**. Tese (Doutorado) — University of Massachusetts - Dept Computer Science, September 2006.

WAHARTE, S. et al. Routing protocols in wireless mesh networks: challenges and design considerations. **Multimedia Tools Appl.**, Kluwer Academic Publishers, Hingham, MA, USA, v. 29, n. 3, p. 285–303, 2006. ISSN 1380-7501.

WEISS, G. **Multiagent System: A Modern Approach to Distributed Artificial Intelligence**. London: MIT Press, 1999.

- WHITE, S. et al. An architectural approach to autonomic computing. **In Proceedings of the International Conference on Autonomic Computing - ICAC, IEEE Computer Society**, p. 2–9, 2004.
- WIKIPEDIA. **Sistemas Complexos**. 2007a. Acessado em 11/07/2007. Disponível em: <http://en.wikipedia.org/wiki/Complex_system>.
- WIKIPEDIA. **Emergência**. 2007b. Acessado em 11/07/2007. Disponível em: <<http://pt.wikipedia.org/wiki/Emerg%C3%Aancia>>.
- WU, X. Hierarchical MESH Architecture: Toward Practical Applications. **iwnas, pp. 69-72, 2006 International Workshop on Networking, Architecture, and Storages (IWNAS'06)**, 2006.
- ZHANG, Y. **Wireless Mesh Networking: Architectures, Protocols and Standards (Wireless Networks and Mobile Computing)**. [S.l.]: Auerbach, 2006. 592 p.
- ZHANG, Y. et al. An effective qos scheme in wimax mesh networking for maritime its. **International Conference on ITS Telecommunications Proceedings**, Junho 2006.

APÊNDICE A – Publicações

A.1 A Self-X Approach for OLSR Routing Protocol in Large-Scale Wireless Mesh Networks

Tipo: Full-Paper

Qualificação: A Internacional

Autores: Lucas Guardalben, Azzedine Boukerche, João Bosco M. Sobral e Mirela S.M.A. Notare

Evento: *IEEE Global Communications Conference (IEEE GLOBECOM 2008)*

Status: Publicado

Local: New Orleans, LA, USA

Data: Novembro - 2008

A.2 A Performance Evaluation of OLSR and AODV Routing Protocols Using a Self-Configuration Mechanism for Heterogeneous Wireless Mesh Networks

Tipo: Full-Paper

Qualificação: C Internacional

Autores: Lucas Guardalben, Azzedine Boukerche, João Bosco M. Sobral e Mirela S.M.A. Notare

Evento: *Fourth IEEE International Workshop on Performance and Management of Wireless and Mobile Networks - P2MNET 2008*

Status: Publicado

Local: Montreal, Canadá

Data: Outubro - 2008

A.3 A Self-Configuration Property for OLSR Routing Protocol in Wireless Mesh Networks

Tipo: Short-Paper

Qualificação: A Internacional

Autores: Lucas Guardalben e João Bosco M. Sobral

Evento: *IEEE International Symposium on Network Computing and Applications - NCA 2008*

Status: Aceito para publicação

Local: Cambridge, MA USA

Data: Julho - 2008

A.4 An Architectural Self-Organization Model for Large-Scale Wireless Mesh Networks Based in Technology of Agents

Tipo: Short-Paper

Qualificação: Em julgamento

Autores: Lucas Guardalben e João Bosco M. Sobral

Evento: *6th International Information and Telecommunication Technologies Symposium*

Status: Publicado

Local: Brasília - DF

Data: Dezembro - 2007

A.5 Towards an Architecture for Self-Organization in Wireless Mesh Networks

Tipo: Full-Paper

Qualificação: B Nacional

Autores: Lucas Guardalben, João Bosco M. Sobral e Clytia H. Tamashiro

Evento: *XXXIII Conferencia Latinoamericana de Informática - CLEI*

Status: Publicado

Local: San José - Costa Rica

Data: Outubro - 2007

A.6 Uma Arquitetura para Auto-Organização em Redes em Malha Sem Fio

Tipo: Full-Paper

Qualificação: B Nacional

Autores: Lucas Guardalben e João Bosco M. Sobral

Evento: *SBRT - XXV Simpósio Brasileiro de Telecomunicações*

Status: Publicado

Local: Recife - PE

Data: Setembro - 2007

