Capítulo 2

Redes *Mesh*: Mobilidade, Qualidade de Serviço e Comunicação em Grupo

Antônio Jorge Gomes Abelém[1], Célio Vinicius Neves Albuquerque[2],
Débora Christina Muchaluat Saade[3], Elisangela Santana Aguiar[4], Jairo Lino Duarte[2],
José Eduardo Mendonça da Fonseca[3] e Luiz Claudio Schara Magalhães[3]

[1] Departamento de Informática, Universidade Federal do Pará (DI/UFPA)
[2] Instituto de Computação, Universidade Federal Fluminense (IC/UFF)
[3] Departamento de Engenharia de Telecomunicações, Universidade Federal Fluminense (TET/UFF)
[4] Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Pará (PPGEE/UFPA)

Abstract

Mesh networks are multi-hop wireless networks that can be used as a low cost infrastructure for community and city-wide access networks. In this context, support for killer applications such as cooperative services and mobile multimedia applications are in great demand. This mini-course presents the main challenges and solutions for providing mobility, quality of service (QoS) and multicasting in wireless mesh networks. It begins by describing techniques for supporting device mobility, presenting advantages and disadvantages of each solution. Then it presents QoS issues in wireless mesh networks, discussing the main proposals found in the literature. In addition, multicast routing protocols for ad-hoc networks are described and their main advantages and disadvantages are analized. Finally, mesh network initiatives are cited and challenges that demand future research are pointed out.

Resumo

Redes mesh são redes em malha sem fio auto-configuráveis e de crescimento orgânico. Recentemente vêm sendo consideradas para a criação de infra-estrutura de baixo custo para a construção de redes de acesso comunitárias e de cidades digitais. Neste contexto, é grande o interesse em suportar aplicações multimídia como telefonia IP móvel e aplicações cooperativas. Este minicurso tem como objetivo apresentar os principais problemas, soluções e desafios sobre mobilidade, provisão de qualidade de serviço e comunicação em grupo (multicast) em redes ad-hoc sem fio do tipo mesh.

Primeiro será abordado o estado da arte de técnicas para suporte à mobilidade de dispositivos com as suas implicações, padrões de suporte e deficiências. O segundo tópico versa sobre questões de qualidade de serviço em redes em malha sem fio, apresentando as principais propostas encontradas na literatura. O terceiro tópico aborda a comunicação em grupo em redes mesh, discutindo os protocolos de roteamento multicast propostos para redes ad-hoc mais citados na literatura e destacando suas principais vantagens e desvantagens. Nas considerações finais, são apresentados exemplos de redes mesh que oferecem tais facilidades e apontados os principais desafios que demandam novas pesquisas na área.

2.1. Introdução

Redes *mesh* (redes em malha sem fio) são redes com topologia dinâmica, variável e de crescimento orgânico, constituídas por nós cuja comunicação, no nível físico, é feita através de variantes dos padrões IEEE 802.11 e 802.16, e cujo roteamento é dinâmico. A Figura 2.1 exibe um exemplo de rede mesh. Essas redes evoluíram a partir das redes móveis *ad-hoc* (*Mobile Ad-hoc NETworks*, ou MANETs).

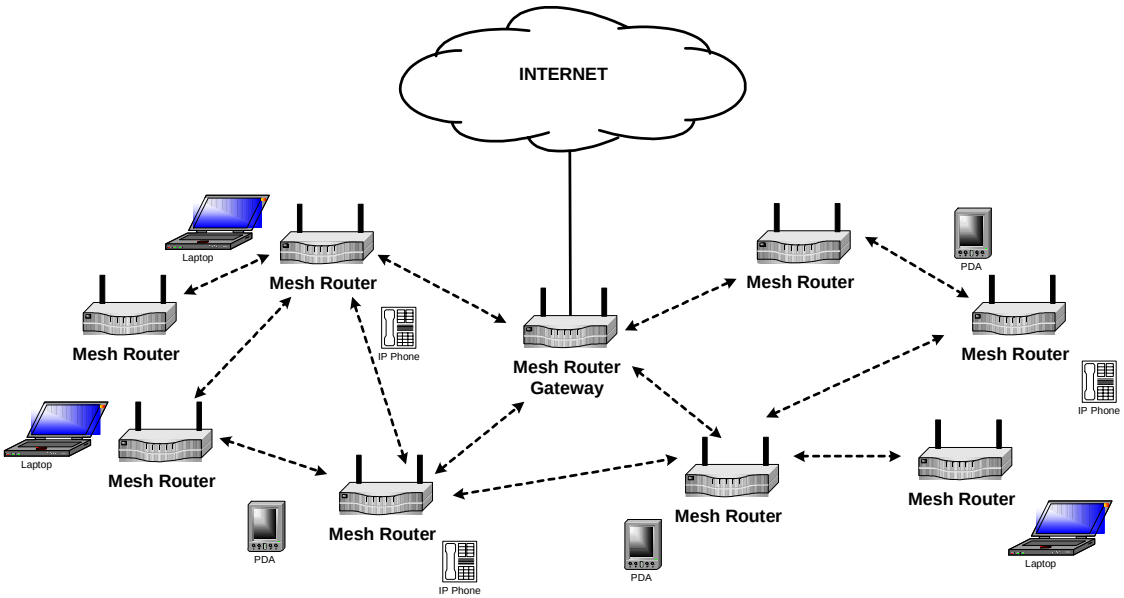

Figura 2.1. Rede *mesh*

Ao contrário das redes *mesh*, as redes *ad-hoc* têm uso limitado devido ao pouco incentivo de seus participantes no compartilhamento de recursos, especialmente quando são escassos (por exemplo, a energia disponível em um laptop) [Kravets 2001]. Por esta razão, as redes *ad-hoc* são empregadas em situações onde não existe uma infra-estrutura de comunicação disponível e a necessidade de compartilhamento de recursos é imperativa para a troca de dados.

Por outro lado, as redes *mesh* invertem o paradigma de usar uma rede cabeada para a espinha dorsal (*backbone*) da rede, e acesso sem fio na última milha. O *backbone* de uma rede *mesh* é sem fio, como pode ser observado pelos roteadores da Figura 2.1 e o acesso dos nós clientes pode ser com ou sem fio. Como os nós do *backbone* deste tipo de rede têm localização fixa, estes podem facilmente ser alimentados, não possuindo,

desta forma, limitação de energia, eliminando, por conseguinte, muitas das restrições das redes *ad-hoc*.

Nos últimos anos, diversos *campi* de universidades e centros de pesquisa ao redor do mundo têm desenvolvido e vêm amplamente utilizando redes *mesh* como redes de acesso ao campus por usuários residentes nas suas proximidades. Exemplos de projeto piloto de redes de acesso sem fio do tipo *mesh* são o *ReMesh* em Niterói/RJ [ReMesh 2007] , *RoofNet* no MIT [Aguayo et al 2004], *Google Mesh* na Califórnia [Google 2007], *VMesh* na Grécia [Tsarmpopoulos 2005], *MeshNet* na UCSB [Ho et al 2004], *Microsoft Mesh* [Draves et al 2004], entre outros.

Um outro potencial de redes *mesh*, além da implantação de redes de acesso à Internet próximas a universidades e escolas, é a construção de cidades digitais, oferecendo infra-estrutura de comunicação sem fio em ambiente metropolitano a todos os cidadãos, o que já vendo realizado em cidades, como por exemplo, Dublin, Taipei, Pittsburgh e Filadélfia. A tecnologia de redes *mesh* é ideal para a construção de redes de acesso comunitárias permitindo o acesso à Internet para aqueles que não possuem condições de arcar com os altos custos de uma conexão faixa larga tradicional do tipo xDSL ou cabo.

O crescente interesse em torno de aplicações multimídia em redes *mesh* traz consigo desafios próprios que tornam a provisão de qualidade de serviço (QoS) e comunicação em grupo (*multicasting*) uma tarefa mais complexa. Tal complexidade é resultado, dentre outros fatores, da própria mobilidade das estações, que implica na necessidade do gerenciamento de suas localizações; das limitações do ambiente e dos dispositivos envolvidos, como por exemplo, a qualidade da transmissão no meio sem fio; dos recursos escassos de largura de banda; da grande variabilidade da qualidade do enlace; das limitações dos componentes de hardware, etc.

Este capítulo tem como principal objetivo apresentar os principais problemas, soluções e desafios sobre mobilidade, provisão de qualidade de serviço e comunicação em grupo (*multicast*) em redes em malha sem fio.

O restante do texto está estruturado da seguinte maneira. A Seção 2.2 aborda o estado da arte de técnicas para suporte à mobilidade de dispositivos com as suas implicações, padrões de suporte e deficiências. A Seção 2.3 versa sobre questões de qualidade de serviço (QoS – *Quality of Service*) em redes em malha sem fio, apresentando as principais propostas encontradas na literatura. A Seção 2.4 aborda a comunicação em grupo em redes mesh, discutindo os protocolos de roteamento *multicast* propostos para redes *ad-hoc* mais citados na literatura e destacando suas principais vantagens e desvantagens. Para finalizar, a Seção 2.5 apresenta as conclusões, apontando os principais desafios atuais que demandam novas pesquisas na área.

2.2. Mobilidade em Redes *Mesh*

Esta seção define alguns cenários típicos de mobilidade em redes *mesh* e em seguida descreve soluções de problemas relacionados à mobilidade em diversos níveis. No nível de rede são descritos o uso de IP Móvel [Perkins 2002], o impacto da técnica *Network Address Translation* (NAT) [Srisuresh 2001] e a técnica MobileNAT [Buddhikot 2005], que procura resolver os problemas causados pelo NAT tradicional. No nível

intermediário entre rede e transporte, é discutido o uso do protocolo *Host Identity Protocol* (HIP) [Koponen 2005]. As camadas de transporte e aplicação oferecem técnicas adicionais de suporte a mobilidade não abordadas neste minicurso por limitações de espaço, como o uso de URIs [Handley 1999, Wedlund 1999], técnicas [Balakrishnan 1995, Ratnam 1998, Brown 1997] que melhoram o desempenho fraco do protocolo TCP [Petrovic 2003, Xylomenos 2001, Tsaoussidis 2002] em redes sem fio e técnicas que colocam o suporte à mobilidade no nível de transporte [Magalhães 2002].

Diferentes cenários de interconexão entre redes com e sem fio demonstram diferentes níveis de problemas de mobilidade e endereçamento. Em uma rede IP, os nós possuem um endereçamento estruturado de forma hierárquica. Cada endereço IP é utilizado para identificar os nós e suportar o roteamento hierárquico. Em redes *mesh*, nós móveis conectam-se a redes IP através de *gateways*, que fazem a ligação entre as sub-redes sem e com fio. Quatro cenários são apresentados para demonstrar os problemas, em vários graus de complexidade, que devem ser tratados quando desejamos suportar mobilidade de clientes sem fio.

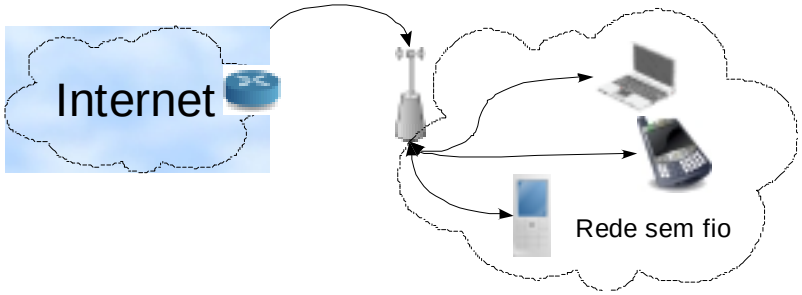

Figura 2.2. Cenário 1: Rede com apenas um *gateway* e distantes a um salto

Cenário 1: O cenário da rede sem fio mais simples é ilustrado pela Figura 2.2**,** onde apenas um *gateway* fornece acesso a Internet a todos os nós sem fio móveis. Todos os nós da rede sem fio estão a um salto do *gateway*, sem qualquer intermediário. Este cenário é o mais básico e não representa desafios em relação ao endereçamento que deve garantir identificação única e correta. O endereço IP dos nós da rede *mesh* pode ser privado ou não, mas deve ter o mesmo prefixo da interface de rede *mesh* do *gateway*.

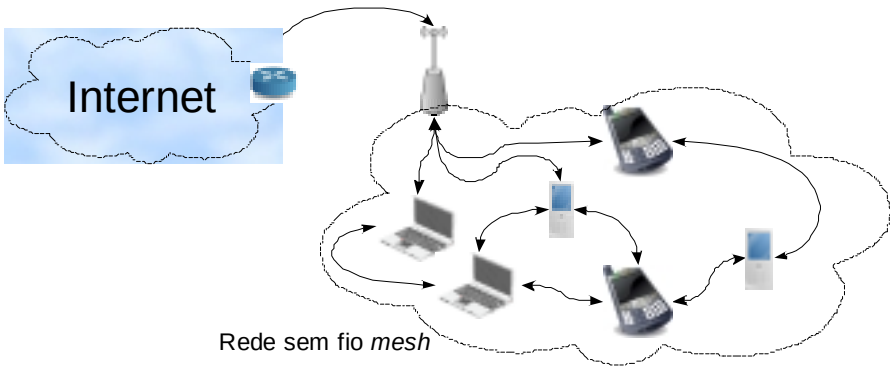

Figura 2.3. Cenário 2: gateway oferece serviço a maiores distâncias pelo uso de múltiplos saltos

Cenário 2: Com o suporte de protocolos de roteamento com múltiplos saltos, uma quantidade maior de clientes pode ser inserida na rede pois estes passam a ter acesso ao

gateway com a ajuda de outros nós. A Figura 2.3 ilustra o caso, onde um único *gateway* conectado a Internet fornece acesso à mesma a todos os clientes da rede *mesh*. Neste cenário, apesar da existência de múltiplos saltos, e uma relação hierárquica entre os nós da rede *mesh*, o endereçamento continua o mesmo do primeiro cenário.

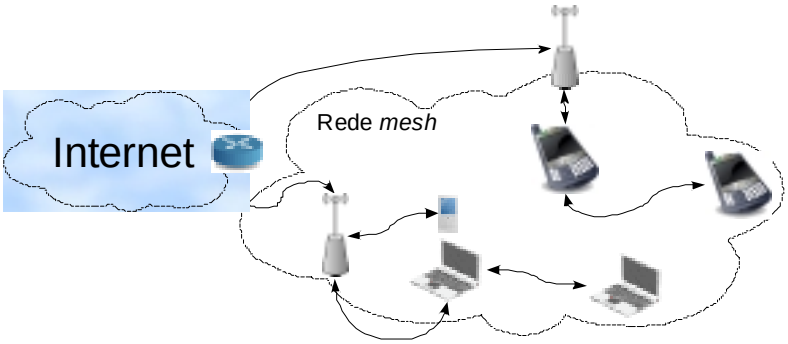

Figura 2.4. Cenário 3: Rede com múltiplos *gateways* e por múltiplos saltos

Cenário 3: Redes *mesh* de grande porte que utilizam protocolos de roteamento de múltiplos saltos podem ter mais de um *gateway*. Consequentemente, os nós podem procurar se conectar ao *gateway* mais próximo em busca de um melhor acesso a Internet, como visto na Figura 2.4. Isto minimiza a carga da rede e pode levar ao balanceamento da carga nos *gateways*. Uma característica deste cenário que simplifica ou evita diversos problemas de endereçamento e mobilidade é que todos os *gateways* conectam-se a Internet por meio de um único nó roteador. Logo teremos apenas micro-mobilidade ou mobilidade intra-domínio, onde o nó movimenta-se dentro do domínio de uma mesma rede *mesh*. O endereçamento neste caso deve suportar a mudança no *gateway* escolhido para dar acesso a Internet aos nós internos e cuidados com as conexões abertas quando na alternância entre os *gateways* de saída.

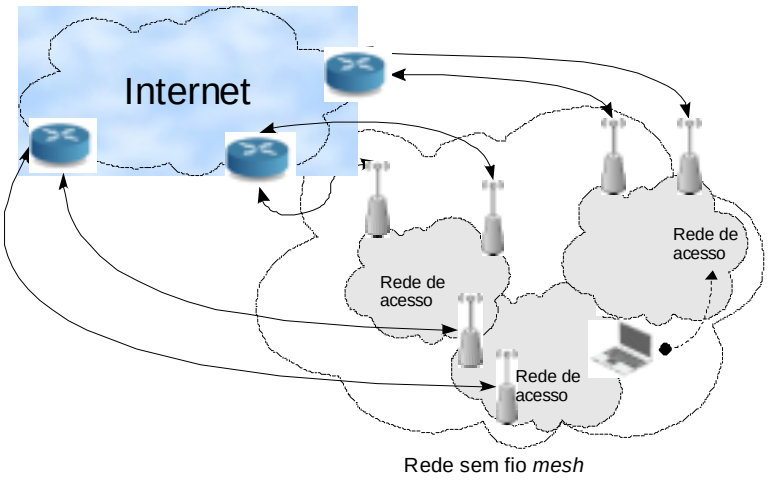

Figura 2.5. Cenário 4: Interação entre múltiplas redes de acesso

Cenário 4: A Figura 2.5 mostra o caso mais complexo e de maior escala de redes *mesh*, onde diferentes redes de acesso, cada qual com múltiplos *gateways* e com múltiplos saltos, interceptam-se formando em conjunto uma rede *mesh*, onde os nós irão conectar-se à rede de acesso mais próxima para ter acesso a Internet.

Neste cenário, quando os nós se movem, novas conexões são criadas para as redes mais próximas. Contudo as conexões de transporte ativas que o nó móvel tinha com a antiga rede de acesso devem continuar a funcionar. Este tipo de mobilidade é conhecido como macro-mobilidade ou mobilidade inter-domínio, pois diferentes domínios, cada um relacionado a uma rede de acesso, são envolvidos e devem trabalhar em conjunto em favor do nó em movimento.

Como cada rede de acesso potencialmente tem endereços IP pertencentes a domínios distintos, a troca de rede pode requerer a troca de endereço IP do nó móvel no cenário 4. Dado que os conexões de transporte (TCP) dependem do endereço IP como parte do seu identificador de conexão, esta mudança de endereço tem que ser levada em conta ou todas as conexões de transporte existentes no momento da troca serão perdidas. Entre as formas de solucionar este problema estão o IP móvel [Perkins 2002], o TCP *Migrate* [Snoeren 2000] e a mobilidade via nível de transporte [Magalhães 2002].

Dentre os cenários apresentados, podemos destacar alguns critérios para analisar a capacidade dos métodos de suporte a alocação de endereço e de mobilidade em redes *mesh*:

- Endereços alocados corretamente segundo a topologia do nó e a rede de acesso;
- Suporte a múltiplos *gateways*;
- Suporte a mobilidade;
- Suporte às conexões de transporte ativas quando houver migração entre redes ou *gateways* de acesso a Internet.

Uma observação sobre *multihoming* é referente à situação ao qual o nó possui duas ou mais conexões simultaneamente ativas em diferentes redes de acesso, o que usualmente significa que o nó possui diversas interfaces, cada qual para uma determinada rede. Nesta situação só existe o problema de endereçamento se houver mudança na rede de conexão de cada interface, pois cada interface pode ser tratada isoladamente[1].

Na próxima subseção, iremos apresentar algumas soluções de endereçamento em redes *mesh* e na seqüência como suportar a manutenção da conectividade a Internet em caso de mobilidade.

2.2.1 Suporte a mobilidade no Nível de Rede

O suporte a mobilidade no nível de rede possui vários aspectos. Veremos inicialmente como resolver o problema de endereçamento, seguindo do padrão IP Móvel que foi desenvolvido com o objetivo de dar suporte a mobilidade em redes IP, passando pela análise sobre o impacto que a mobilidade tem sobre uma técnica comumente utilizada, chamada de NAT e a técnica MobileNAT que procura resolver os problemas causados pelo NAT tradicional.

[1] No entanto, é possível usar múltiplas interfaces para uma única conexão, aumentando a banda disponível [Magalhães 2001].

2.2.1.1 Alocação de Endereço

Para um esquema de alocação de endereços em redes *mesh* é importante identificar cada nó por um endereço localmente único. Para tal algumas estratégias podem ser adotadas.

Na alocação *stateless* não existe um mecanismo centralizado para distribuir os endereços e os nós configuram endereços únicos automaticamente, através de três possibilidades:

- Cada nó cria um endereço de forma aleatória e posteriormente verifica pelo protocolo de detecção de endereço duplicado (*Duplicate Address Detection*-DAD [Perkins 2001]) a unicidade do endereço criado,
- Uso do mecanismo de configuração de endereço automático definido pelo IPv6 em conjunto com os mecanismos *One-hop* [Zhao 2004] e *Weak* DAD [Vaidya 2002],
- Utilizar somente os mecanismos do IPv6 como sugerido por [Bechler 2003].

Para obter endereços alcançáveis na Internet, cada nó pode então usar o prefixo da rede que é periodicamente anunciado pelo *gateway* ou enviar um pedido ao *gateway* [Perkins et al 2002] ou o prefixo pode ser incluído pelo *gateway*.

Na alocação *statefull*, um mecanismo de controle central gerencia a alocação de endereços únicos, que pode ser por DHCP ou pelo uso de agente externo (*Foreign Agent*), se for utilizado IPv4 móvel [Perkins 2002].

Depois de adquirir um endereço único e possivelmente roteável pela Internet, veremos na próxima subseção alguns métodos que podem suportar mobilidade em redes *mesh*, como *IP Móvel*, *Network Address Translation* (NAT) e *Host Identity Protocol*.

2.2.1.2 IP Móvel

IP Móvel [Perkins 2002] é um protocolo desenvolvido pela *Internet Engineering Task Force* (IETF) com o objetivo de permitir que nós, ao se movimentarem entre diferentes redes de acesso, mantenham um endereço IP permanente.

No protocolo IP Móvel, cada nó móvel possui dois endereços: um permanente, conhecido como "*home address*", e outro dinâmico, chamado de "*care-of-address*", que é relativo à rede de acesso que o nó está visitando e se modifica quando o nó associa-se a outra rede de acesso.

Existem dois tipos de entidades no IP Móvel. O primeiro é o agente externo (*Foreign Agent*) que é responsável por gerenciar informações como o endereço dinâmico (*care-of-address*). Os nós móveis devem entrar em contato com o agente externo para registro e aquisição de um endereço dinâmico, que é topologicamente relacionado à rede do agente externo. O segundo é o agente permanente (*Home Agent*) que é um nó fixo e conhece o endereço permanente (*Home address*) do nó móvel e, portanto, pode ser visto como representante (*proxy*) do nó móvel.

A Figura 2.6 exibe o fluxo de comunicação do protocolo IP Móvel. Inicialmente, um nó que quer se comunicar com o nó móvel envia pacotes de dados ao agente permanente (*Home Agent*), que os redireciona ao nó móvel. O redirecionamento utiliza o último endereço dinâmico (*care-of-address*) conhecido do nó móvel, encapsulando os

pacotes IP enviados pelo nó correspondente em pacotes IP com o novo endereço de destino, criando um túnel entre a rede antiga e a nova do nó móvel. Se o nó móvel mudar de rede e por conseqüência alterar também o agente externo (*Foreign Agent*), este deve enviar ao agente permanente o novo endereço obtido depois da movimentação.

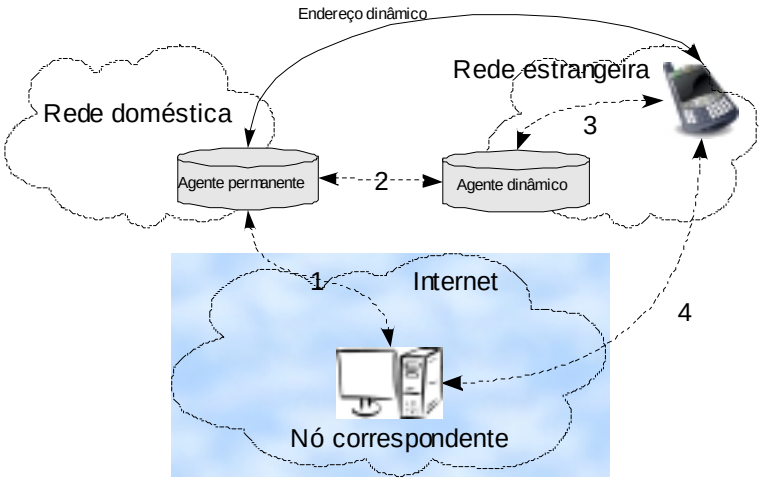

Figura 2.6. IP Móvel

O IP Móvel suporta o uso de múltiplos *gateways*. Também é capaz de manter as conexões do nível de transporte em funcionamento durante as migrações de rede de acesso e agentes externos, portanto o protocolo IP Móvel é apropriado para ser usado em todos os cenários apresentados, de 1 a 4. Contudo este protocolo possui algumas deficiências destacadas a seguir:

- Os nós móveis devem sempre informar sobre a sua movimentação ao seu agente permanente, que pode provocar um significativo aumento na latência do restabelecimento do fluxo de dados, quando o nó móvel muda de ponto de acesso freqüentemente;
- Cada nó móvel deve possuir um agente permanente, que por sua vez deve ter um endereço globalmente roteável e, portanto, um endereço Internet. Deve ainda estar sempre disponível nos momentos em que se deseja estabelecer alguma comunicação com o nó móvel, criando assim um ponto de falha central. Se o agente permanente falhar, o nó móvel deixa de ser alcançável;
- O filtro de ingresso [Ferguson 1998] é uma técnica recente que dificulta alguns tipos de acesso maliciosos a Internet, onde em cada domínio administrativo os roteadores de borda só retransmitem pacotes em direção à Internet se eles tiverem o endereço IP correto, ou seja, de um IP pertencente a sub-rede do roteador. Este filtro prejudica o funcionamento normal do IP Móvel, que passa a ser obrigado a utilizar uma adaptação como encontrada em [Montenegro 1998]. Esta adaptação consiste no uso de um túnel entre o nó móvel e o agente permanente para encaminhar os pacotes saindo do nó móvel em direção ao nó correspondente em questão, removendo assim uma otimização anterior (roteamento triangular), onde o nó móvel enviava pacotes diretamente ao nó correspondente. Com isso, em ambos os sentidos da comunicação, entre o nó móvel e correspondente, o agente permanente é usado como intermediário;

- Em redes *mesh,* os pacotes enviados em *broadcast* podem causar grande impacto no desempenho da rede, pois o agente externo *(foreign agent)* não pode resolver o endereço ARP, o que pode provocar uma inundação de pacotes por toda rede;
- O fato de redes *mesh* realizarem comunicação em múltiplos saltos dificulta ou impede o correto funcionamento dos mecanismos de detecção de movimentos encontrados no padrão IP Móvel.

Além do IP Móvel, que foi originalmente desenvolvido com o objetivo de suportar macro-mobilidade, os seus pesquisadores criaram outros dois protocolos para Internet, Celular IP [Valkó 1999] e Hawaii [Ramjee 1999] para oferecerem suporte a micro-mobilidade.

A seguir, serão apresentadas pesquisas adicionais [Perkins 1996, Perkins 1997] que amadureceram o IP Móvel, de modo a permitir o uso contínuo da Internet em redes *ad-hoc* sem fio com nós móveis. Neste sentido, ao se incluir estas extensões ao IP Móvel, passa-se a permitir que dispositivos móveis usem um endereço dinâmico (*care-of-address*) mesmo que estejam a mais de um salto do agente externo (*Foreign Agent*). Resolve-se ainda o conflito na manipulação de tabela de rotas, pois tanto a rede *mesh* quanto o IP Móvel tentam manipular a tabela de roteamento. Uma solução para tal conflito é instalar um terceiro gerente para coordenar as manipulações.

Como o agente externo envia periodicamente pacotes de anúncio de agente (*Agent Advertisement messages*) por difusão (*broadcast*), cada nó da rede *ad-hoc* com suporte ao IP Móvel deve retransmitir o pacote para que os nós distantes do agente possam receber tal anúncio. Cada nó faz um controle dos anúncios com vários objetivos: manter atualizada a disponibilidade do agente, descobrir a realização de movimentação por detecção de outros agentes e evitar a propagação duplicada de um pacote de anúncio. Estes mesmos nós irão utilizar o agente como *gateway*.

Cada nó móvel pode tentar descobrir um agente externo (*Foreign Agent*) pró-ativamente, em vez de esperar a chegada de algum pacote de anúncio, como visto anteriormente. Para tal, o nó envia um pacote RREQ tendo como destino o IP 224.0.0.11, que é o endereço do grupo *multicast* reservado para todos os agentes móveis. Cada nó vizinho que receber o pacote pode consultar a sua própria tabela de agentes para procurar um agente válido, dando uma resposta local pelo pacote RREP. Caso contrário, o nó vizinho em questão retransmite o pacote, seja por não suportar o IP Móvel ou por não ter um agente válido na sua própria tabela.

Em uma solução alternativa, MIPANET [Jönsson 2000], os nós móveis de uma rede *ad-hoc* que desejam ter acesso a Internet realizam um registro no agente externo e passam a utilizar o seu endereço permanente *(home address)* para todas as comunicações.

O nó móvel cria um túnel com o agente externo, para encaminhar todos os pacotes com destino a Internet que, ao chegarem ao agente externo, são desencapsulados e enviados em seguida ao destinatário. Dentro da rede sem fio o protocolo de roteamento *ad-hoc* é responsável por rotear o pacote do nó móvel ao agente externo. Adicionalmente o MIPMANET utiliza um novo algoritmo, chamado *MIPMANET Cell Switching (MMCS),* que indica ao nó móvel a necessidade de realizar um novo registro em outro novo agente externo.

A seguir, passa-se a examinar uma técnica muito utilizada atualmente, NAT, que possui um grande impacto sobre IP Móvel, pois na prática impede o uso do formato padrão, obrigando, assim, a realização de modificações para criar compatibilidade.

2.2.1.3 Impacto da Técnica *Network Address Translation* (NAT) na Mobilidade

A técnica NAT (*Network Address Translation*) [Srisuresh 2001] é utilizada basicamente por dois motivos. Primeiro, pela indisponibilidade de endereços IP globalmente roteáveis pela Internet e, segundo, para aumentar o nível de segurança e controle de uma rede privada. Tal técnica consiste no procedimento de tradução do endereço IP original, de um pacote que sai de um nó interno de uma rede privada para uma rede pública.

A técnica NAT opera no *gateway* de uma rede, que é o ponto de união entre a rede interna e as redes externas, manipulando os pacotes originados dentro da rede interna com destino a algum nó da rede externa. Um mecanismo do NAT corresponde em substituir o par endereço IP/porta internos por um par endereço da interface externa/porta do *gateway*. O novo endereço é globalmente roteável, o que permite que nós da rede externa se tornem capazes de estabelecer comunicações com o *gateway*. Os pacotes que seguem o caminho contrário, que originam da rede externa, passam pelo *gateway*, que os redirecionam para a interna. Tal situação é ilustrada na Figura 2.7.

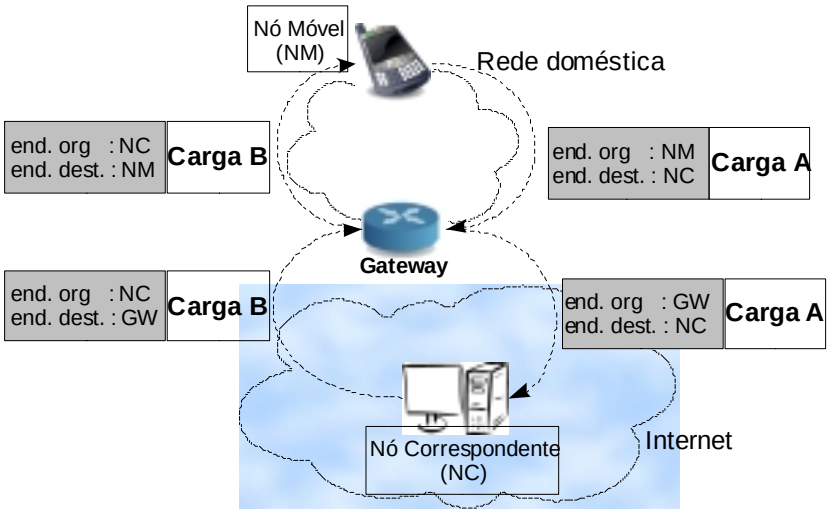

Figura 2.7. NAT substitui o endereço IP interno pelo endereço global do *gateway*

Nas redes *mesh* com múltiplos *gateways* (Figura 2.5)**,** a técnica NAT pode ser implementada em cada *gateway* da rede *mesh* ou em outro roteador externo a essa rede, caso este seja usado por todos os *gateways* da rede *mesh*.

Quando a técnica NAT for implementada em cada *gateway* da rede *mesh* (no primeiro caso), problemas de quebra de transparência irão afetar a semântica das conexões, pois os pacotes que são originários de qualquer nó da rede interna, parecem ter origem no próprio *gateway* NAT. Assim, se um nó interno mudar de *gateway* NAT, o nó correspondente não vai reconhecer os pacotes seguidos da mudança como sendo da conexão, pois a origem dos pacotes será o novo gateway, quebrando assim a conexão antiga e a perda do seu contexto nos níveis acima de rede.

O segundo caso ocorre quando um nó realiza a função NAT. Nesta situação**,** não existe nenhum desafio em relação à mobilidade dos nós da rede interna em acessarem a rede externa, bastando apenas que o protocolo de roteamento da rede interna seja capaz de dar suporte à mobilidade. Contudo, quando os dois extremos de uma comunicação são móveis e estão dentro na mesma rede, ou seja sobre o mesmo *gateway* que realiza o NAT, um suporte extra deve ser desenvolvido.

Utilizar NAT em um único nó pode não ser possível ou pode apresentar um desempenho inadequado. Usualmente, apenas topologias onde todos os *gateways* são suportados por uma única infra-estrutura (*backbone*), e o *gateway* que realiza o NAT está localizado neste *backbone* (e é alcançável através de um ou poucos saltos), são capazes de suportar com qualidade a demanda de roteamento.

Essas técnicas são modos de encapsular um pacote original, com destino ao nó correspondente, em um novo pacote, com destino ao *gateway*. Ao chegar no *gateway*, este desencapsula o pacote original e realiza a técnica NAT, enviando-o ao nó correspondente. Um efeito do túnel é tornar transparente a rota (*gateways* intermediários) que foi utilizada pela rede, para enviar o pacote do nó móvel ao *gateway* que está no final do túnel. Contudo, isso iria obrigar uma implementação específica tanto no *gateway*, que está no fim do túnel, quanto no nó móvel para tratar o encapsulamento e desencapsulamento necessários ao funcionamento do túnel.

Por fim, o uso de NAT é suficiente nos três primeiros cenários, quando o suporte a macro-mobilidade não é necessário e o *gateway* NAT é feito em um nó central, possuindo a vantagem, em relação ao IP Móvel, de não precisar ter um endereço fixo (*Home Address*) ou de algum processo de registro no *gateway*.

Em seguida, apresenta-se uma evolução de NAT, que busca dar o suporte a mobilidade, pelo uso da técnica NAT.

2.2.1.4 MobileNAT

Examina-se, neste tópico, uma adaptação da técnica NAT vista anteriormente, chamada MobileNAT [Buddhikot 2005], através da adição do suporte tanto a micro quanto a macro-mobilidade, entre diversos espaços de endereçamento heterogêneos. As três principais características são, (1) uso de dois endereços – um virtual e outro globalmente roteável, (2) extensão do protocolo DHCP *(Dynamic Host Configuration Protocol)* para gerenciar os dois endereços em conjunto e (3) um gerente de mobilidade, que altera as regras NAT em resposta aos eventos de mobilidade. Esta técnica foi desenvolvida para não modificar as redes de acesso e não interferir nos mecanismos existentes de IP Móvel, buscando assim suporte a mobilidade entre redes com e sem fio (veja a Figura 2.8).

MobileNAT possui dois componentes que realizam tradução de endereço. Uma é realizada em um roteador chamado nó âncora (*archor node)* e possui um comportamento similar com o *gateway* NAT visto anteriormente. A segunda tradução é feita em um novo nível chamado *shim* que é colocado no nó móvel. Com essas traduções, o MobileNAT é capaz de influenciar na forma em que o pacote será roteado entre o nó âncora e móvel, realizando adaptações sob os eventos de mobilidade.

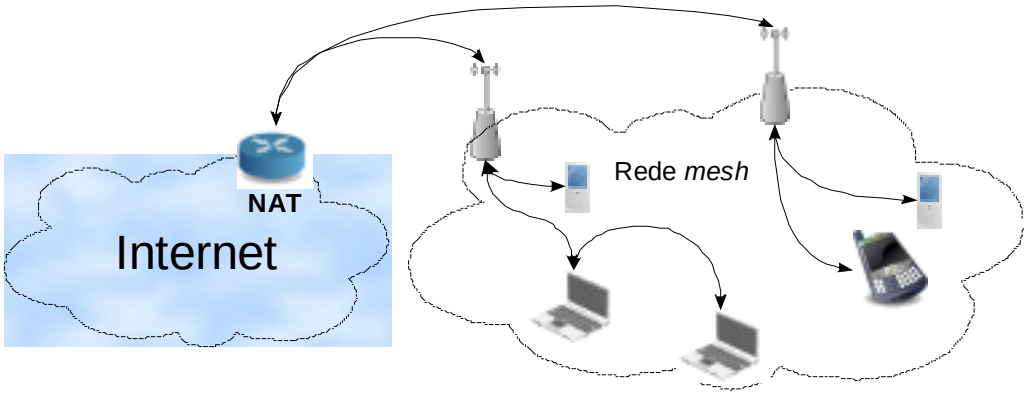

Figura 2.8. Suporte a mobilidade entre redes com e sem fio e a comunicação entre nós móveis da mesma sub-rede ou de redes distintas (Fonte [Buddhikot 2005])

O endereço IP possui uma deficiência em relação à mobilidade por empenhar duas funções distintas, uma é identificar o nó no nível de transporte e segundo servir como referência a sua localização física no nível de rede. A deficiência fica evidente pelo evento de mobilidade, pois apesar do nó continuar sendo o mesmo em outra rede de acesso, a sua localização é diferente, gerando assim um conflito das funções, vez que uma precisa da alteração do endereço e a outra não.

Com fins de resolver tal deficiência e conflito, a técnica MobileNAT utiliza dois endereços para cada nó. O primeiro é o endereço virtual E_v (*Virtual IP*) que é fixo, cuja função é dar uma identificação que não se altera no evento de mobilidade, e por tal é utilizado pelos níveis de transporte e nos acima. O segundo é o endereço físico E_F (*Physical IP*) que identifica a localização corrente do nó móvel, utilizada para rotear os pacotes para o nó móvel dentro de um domínio (espaço de endereçamento ou sub-rede) ou pela Internet. Claramente o E_F deve ser modificado sempre que o evento de mobilidade levar o nó à uma nova sub-rede.

Como ambos os endereços podem ser públicos ou privados, existem quatro combinações possíveis. A técnica MobileNAT oferece suporte a todas as formas. Veja a seguir a combinação mais provável em redes *mesh*, onde ambos endereços são privados.

A Figura 2.9 exemplifica a mobilidade intra-domínio, com conexões TCP ao serviço da página de notícias *CNN* na porta destino 80 (http) e de origem SP. O nó móvel possui a tupla de endereços ($E_{f,}$,E_v). Na camada de transporte esta sessão possui a tupla de endereço (TCP, E_v, CNN, SP,80). Como o E_v é o endereço de origem e não pode ser usado no roteamento do pacote, deve ser substituído pelo E_f na técnica NAT de origem (*Source* NAT ou SNAT). O mesmo ocorre quando o pacote de resposta chega ao nó móvel com a técnica NAT de destino (*Destination* NAT ou DNAT). Estas substituições são realizadas entre nível de rede e de enlace, em um nível chamado *shim.*

Quando o pacote chega ao nó âncora, é realizada a técnica NAT convencional, o E_f é substituído pelo endereço público do nó âncora (E_{na}), pois o $E_{f,}$, por ser privado, pode somente ser utilizado dentro do domínio controlado pelo nó âncora. No caso do E_f ser público, não será necessário realizar NAT, porém o endereço deve ser o da sub-rede, no qual o nó âncora é responsável por realizar o roteamento. O nó âncora mantém o

mapeamento de endereços realizado na tradução. Por fim, o pacote é encaminhado ao nó correspondente, que é o servidor *CNN*.

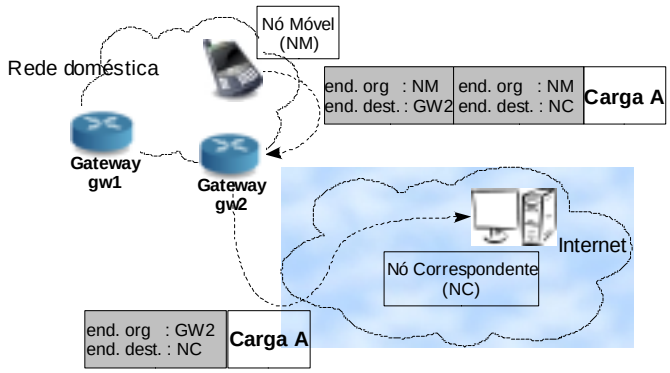

Figura 2.9. Suporte a mobilidade intra-domínio para sessões na Internet

O servidor *CNN* mantém a conexão no nível de transporte com a tupla (TCP, CNN, E_{na} ,80,SP). Quando a resposta é enviada, o pacote que a contém sofre as mesmas traduções do pacote de requisição, mas em ordem inversa.

Quando o nó realiza algum movimento, para a conexão não ser invalidada quando o nó chegar a nova rede de acesso, as tuplas nos nós móveis e correspondentes não devem ser alteradas. Neste momento, o uso do E_v é necessário, pois como não é alterado no evento de mobilidade, permite-se que a tupla no lado do nó móvel fique estável. No outro extremo da conexão ao nó correspondente, *CNN*, a tupla utiliza o endereço do nó âncora, que igualmente não é alterado no mesmo evento de mobilidade. Somente o E_f utilizado pelo nível *shim* deve ser atualizado para refletir a nova localização física. O nó âncora e o nó móvel, quando informados da mobilidade, atualizam o mapeamento com o novo E_f. Sendo assim, o nó correspondente não precisa realizar qualquer ação, pois este é protegido de qualquer alteração causada pela mobilidade, somente o nó âncora e o nó móvel precisam ser modificados para suportar o MobileNAT.

Uma alteração no MobileNAT pode tornar dispensável qualquer modificação no nó móvel. Isso é possível quando o nó âncora é o *gateway* de uma sub-rede e existem outros roteadores NAT adaptados dentro da mesma sub-rede. Quando o nó móvel realiza uma movimentação dentro da sub-rede, cada roteador pode detectar a presença de um novo nó ao verificar que houve uma falha na tabela de tradução. Quando a detecção ocorrer, o *gateway*, que também é o nó âncora, será notificado pelo roteador sobre a nova localização do nó móvel, sendo assim, o *gateway* vai utilizar o roteador que fez a notificação como sendo o novo ponto do nó móvel.

No MobileNAT, cada nó móvel recebe o par (E_v,E_f) e o endereço do nó âncora por um servidor DHCP capaz de trabalhar com o MobileNAT. Ambos endereços do nó móvel possuem um tempo de validade, de forma que precisam ser renovados. Ao realizar uma movimentação, o nó móvel deve requisitar um novo E_f e, caso na nova rede o E_v estiver em uso por outro nó, deve ser feita sua renovação. Sempre que os endereços forem modificados, o nó âncora deve ser informado para que a tabela de tradução seja atualizada. O nó móvel pode ter mais de um E_v após cada evento de mobilidade, E_v

antigos serão utilizados enquanto as conexões antigas estiverem em uso, e os novos, para cada nova rede acessada.

A modificação citada acima no DHCP não cria incompatibilidade com clientes que não entendam ou precisem de MobileNAT, pois tratam-se apenas de campos adicionais de informações que podem ser ignorados.

Acima foi mostrado um exemplo de mobilidade intra-domínio. Apresenta-se, a seguir, um exemplo de mobilidade inter-domínio, ilustrado pela Figura 2.10.

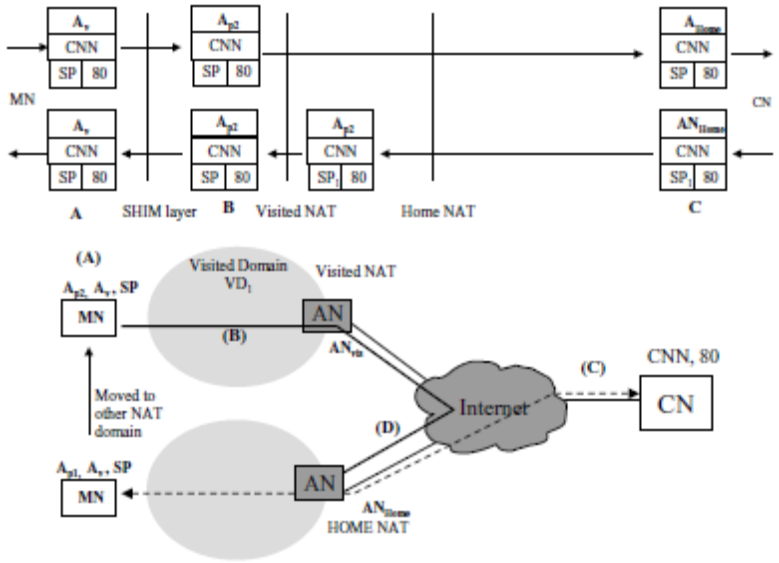

Figura 2.10: Mobilidade entre diferentes domínios, preservando as conexões no nível de transporte (Fonte [Buddhikot 2005])

No caso de mobilidade inter-domínio, diferentes nós âncoras serão envolvidos, pois cada domínio possui um nó responsável. O primeiro e antigo domínio é chamado de NAT nativo *(home* NAT) e o novo de NAT visitado (*visited* NAT), que devem cooperar para a manutenção das conexões existentes ativas do nó que realizou a movimentação. Quando o nó móvel entra no novo domínio e adquire outro $E_{f,}$ o nó âncora do domínio nativo deve ser informado do novo $E_{f,}$ assim como do endereço do nó âncora do domínio visitado.

Todo tráfego das conexões antigas continua sendo enviado ao nó âncora nativo (*home* NAT), pois os nós correspondentes não podem sofrem qualquer modificação causada pelo evento de mobilidade.

O nó âncora nativo repassa todos os pacotes destinados ao antigo E_f do nó móvel, pertencentes às conexões iniciadas durante a presença do nó móvel em seu domínio, ao novo do domínio visitado, utilizando o nó âncora visitado como intermediário. Após chegar ao nó âncora visitado, os pacotes seguem o caminho para a nova localização do nó móvel.

O MobileNAT pode ser comparado às variações do IP Móvel:

- **IPv6** cria a possibilidade do nó correspondente enviar respostas a um endereço diferente de quem enviou a requisição, pelo uso da opção IP de destino

(*destination* IP). Similar a técnica de otimização de rotas, que permite ao nó correspondente descobrir o endereço atual do nó móvel. Como a implementação deste padrão requer mudanças em larga escala, que ainda não foram realizadas na infra-estrutura da Internet, não é, portanto, uma opção viável.

- **IP Móvel** com **NAT** exige o uso de túneis e de adaptações no agente permanente (*Home Agent),* pois tanto o nó móvel quanto o agente externo *(foreign agent)* passam a possuir endereços privados. O agente permanente é capaz de perceber que o nó móvel e o agente externo estão sob NAT, ao verificar que o endereço anunciado no pacote de comunicação entre os agentes difere do endereço do pacote, pois o endereço do pacote que chega ao agente permanente é do *gateway* que realizou o NAT.

O padrão IP móvel possui deficiências no suporte a micro-mobilidade, assim, como na migração rápida (*fast handoff*). Algumas técnicas adicionais foram desenvolvidas para melhorar tais deficiências:

- **IP celular (*Cellular IP* – CIP)** [Valkó 1999] utiliza protocolos proprietários dentro dos domínios móveis que não permitem a operação com outros nós que utilizem somente o padrão IP dentro do mesmo domínio. É adequado somente em ambientes homogêneos que tenham um único fornecedor.
- **Hawaii** [Ramjee 1999] permite que nós móveis mantenham um IP permanente dentro de um domínio, e para tal os roteadores devem manter regras de roteamento para cada nó, além de oferecerem suporte especializado a mecanismos de atualização das tabelas de roteamento, para responderem a movimentação de cada nó. Portanto, não possui alta escalabilidade e exige uma ampla modificação na infra-estrutura do domínio.
- **IP Móvel hierárquico** [Perkins 2003] (*Hierarchical mobile IP* – HMIP) oferece a idéia de registro por regiões (*regional registration*), substituindo os registros locais em algumas situações, em vez de registros nos agentes permanentes que podem estar longe. Estas situações que permitem registros locais são quando ocorre a mobilidade dentro de um domínio. O agente externo é dividido em dois, agentes externos de *gateway* que definem um domínio e agentes externos regionais que são ligados ao tipo anterior. O registro no agente permanente é realizado tão-somente quando o nó móvel entra em um novo domínio. Enquanto a mobilidade for dentro do mesmo domínio, os registros são feitos aos agentes regionais. É criado um túnel entre o agente externo regional e o de gateway para conduzir toda comunicação do nó móvel.
- **Protocolo de mobilidade intra-domínio** *(intra-domain mobility protocol* IDMP) é similar ao HMIP, suportando ainda múltiplos agentes de mobilidade (similar ao agente externo de *gateway*). Utiliza DHCP para sinalização e é adequado para ser usado em redes com NAT.
- **Virtual-NAT** [Su 2002] é uma proposta recente de migração de conexões TCP ao evento de mobilidade, provendo suporte a mobilidade no nível de transporte [Magalhães 2002]. A tradução é realizada nos dois extremos, ou seja, nos nós clientes. Utiliza sinalização explícita entre estes nós, levando assim a implementação obrigatória no nó correspondente mesmo se este não for móvel.

Na próxima subseção, apresenta-se uma proposta de um novo nível de suporte a mobilidade chamada HIP [Koponen 2005], que possui alguns aspectos avançados de segurança, como o uso de criptografia na comunicação e de mecanismos de prevenção contra ataque de negação de serviço *(Denial of Service* – DoS *).* O MobileNAT é uma proposta mais simples, que permite a integração de nós com e sem mobilidade. Apenas o lado de quem realiza a movimentação precisa ter o MobileNAT implementado. O protocolo HIP (*Host Identity Protocol*), por sua vez, representa uma solução interessante a longo prazo, por ser uma proposta para a próxima geração de Internet.

2.2.2 Suporte a Mobilidade no Nível Intermediário entre Rede e Transporte: *Host Identity Protocol* (HIP) – Protocolo de Identificação de Nós

Foi visto anteriormente que o nível de rede possui inúmeras questões que dificultam o suporte à mobilidade. Existem alternativas que fogem do nível de rede e de seus problemas. A seguir será examinado o protocolo *Host Identity Protocol* (HIP) que utiliza esta abordagem.

HIP [Koponen 2005] é uma proposta de um nível intermediário que deve ser inserido entre os níveis de rede e de transporte. Tem como objetivo prover um método de identificação de nós, que separa a identificação da localização hierárquica do endereço IP. HIP introduz um novo espaço de nomes para identificação de um nó (*HI, Host Identiy*) entre os níveis de rede e de transporte (veja Figura 2.11**)**. Durante a comunicação entre nós, o HIP provê uma identificação única (HI) para o estabelecimento e atualização da comunicação.

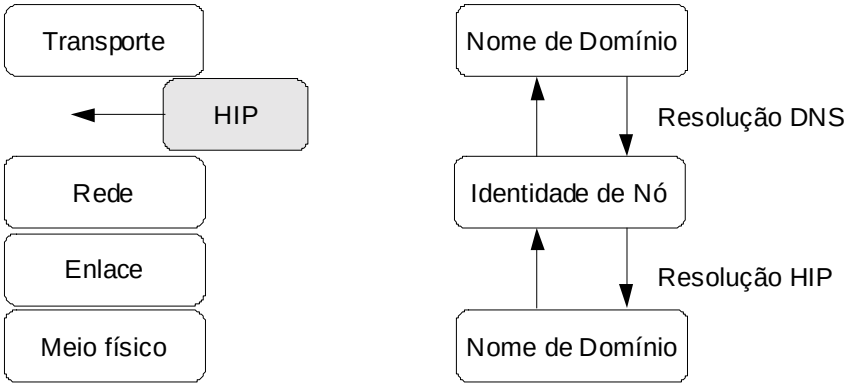

Figura 2.11. À esquerda a nova camada e à direita a resolução de identificações

Um nó móvel usualmente estabelece um contato inicial com um nó correspondente (potencialmente móvel também), através da consulta a um servidor DNS. É possível que ocorra um problema em situações em que o servidor DNS não possui a localização atualizada do nó móvel correspondente (endereço IP referente a esta nova localização).

O mesmo problema de falta de atualização da localização pode ocorrer, mesmo quando o DNS responsável pela identificação é atualizado sempre que o nó realiza alguma mudança. Isto porque o servidor DNS utilizado pelo primeiro nó possivelmente guardou em sua *cache* a antiga localização da identificação. No modelo não móvel, em que os nós raramente realizam a movimentação, o uso de *cache* permite grandes ganhos

de desempenho e escalabilidade, porém torna-se inconveniente quando se tratam de nós móveis que podem a todo momento mudar de localização.

O HIP alivia o problema relacionado ao DNS, pois através de extensões *rendezvous* (*rendezvous extensions*), em que um nó com identificação HI pode ser alcançado pelo IP de um nó intermediário, o chamado servidor *rendezvous*. Este servidor possui uma localização estável e fixa para ser sempre alcançável. Com HIP, a configuração DNS para um nó consiste nos dois itens anteriores, a identificação HI e a localização do servidor *rendezvous*, de forma que quando um nó iniciar uma nova conexão com um outro nó qualquer, a primeira mensagem será enviada ao servidor fixo, que o reenvia ao nó destinatário pela última localização conhecida (veja a Figura 2.12**)**. O restante das mensagens é enviado diretamente pelos nós finais, sem passar pelo servidor, a fim de evitar o efeito do roteamento triangular. Quando o nó móvel muda de lugar, este informa ao servidor a sua nova localização.

O comportamento do servidor *rendezvous* é similar ao agente permanente (*home agent*) da arquitetura IP Móvel.

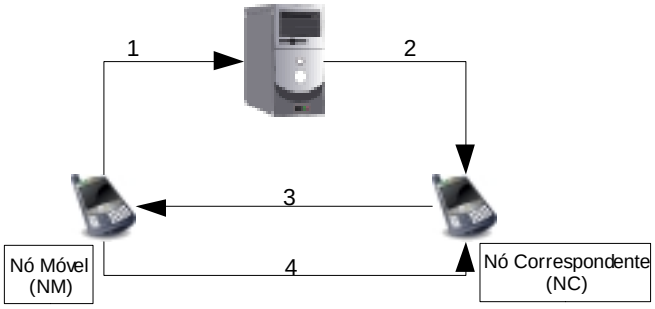

Figura 2.12. Estabelecimento da associação em HIP

HIP permite a mobilidade IP fim-a-fim de forma transparente para a camada de transporte, logo HIP é uma solução para todos os quatro cenários apresentados anteriormente, possuindo apenas uma falha: o uso do HIP iria obrigar a modificação da pilha de camadas de rede e a introdução de um serviço global *rendezvous*.

2.2.3 Detecção de mobilidade

Idealmente o melhor nível de gerência sobre a mobilidade é realizado por quem pratica a mobilidade, no caso o próprio dispositivo móvel. Entretanto, nem sempre isso é possível. Para discutirmos esse tópico, são definidas duas classes [Bondareva 2006] de detecção de mobilidade:

1. **Classe I: o nó sabe da sua macro-mobilidade**

   a) **O nó conhece o *gateway* que deve utilizar:** neste caso o nó controla de forma explícita qual *gateway* será utilizado. Este é o caso padrão da detecção de mobilidade, o que é razoável em redes cujo protocolo de roteamento permite ao nó saber todo o caminho necessário até o *gateway* e permite ao nó cuidar de sua própria mobilidade.

   b) O *gateway* informa ao nó de sua mobilidade: quando o nó desconhece a rota até o *gateway*, o que ocorre em redes com roteamento em cada salto e, portanto o nó móvel não é capaz de identificar uma mudança de rota, pois a

decisão de alterar a rota foi estabelecida por outro nó durante o trajeto até ao *gateway*. Contudo, o *gateway* ao receber os pacotes é capaz de identificar que foram originados de um cliente recém chegado e, assim, enviar uma notificação ao nó sobre a sua mobilidade, permitindo que o nó cuide de sua mobilidade e realizando as operações necessárias. Ilustra-se tal situação na Figura 2.13.

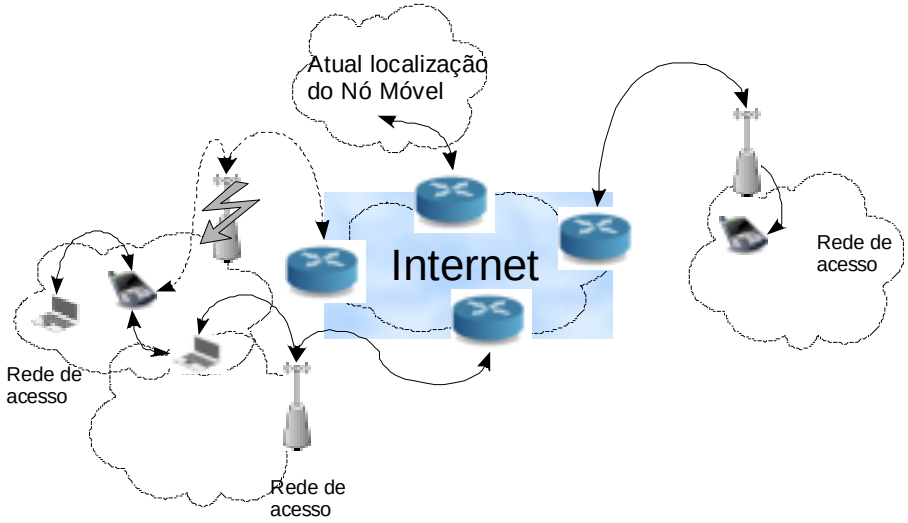

Figura 2.13. Problema de detecção de mobilidade: Por causa da auto-reparação da rede *mesh* em algum salto o nó móvel tem alterado o seu *gateway*

2. **Classe II: o nó não sabe da sua macro-mobilidade**

   Em diversas situações, pode ser desejável que os protocolos de roteamento da rede sem fio e os nós móveis não precisem gerenciar as questões de mobilidade. Uma opção é delegar esta função ao *gateway* e passar a suportar a mobilidade. Neste caso toda sobrecarga necessária a este suporte é transferida na sua totalidade ao *gateway*.

2.2.3.1 Implementação da Detecção de Mobilidade na Classe I

O melhor nível de gerência sobre a mobilidade é viável se o nó móvel possuir mecanismos necessários e suficientes para perceber o evento de mobilidade assim como tomar medidas de adaptação e reconfiguração necessárias para manter a conectividade. Os detalhes de como a detecção de mobilidade na classe I pode ser implementada em seus dois tipos são dados a seguir:

a) Ambos os protocolos, IP Móvel e HIP, utilizam detecção de mobilidade apenas da Classe Ia. Esta classe implica que o nó móvel estabelece e mantém a sua conexão padrão para o mesmo *gateway*. Somente no caso do *gateway* padrão ficar fora de alcance, o nó móvel deve descobrir e estabelecer uma nova rota a um novo *gateway* e, se necessário, reconfigurar o seu endereço. Nesta classe de detecção de mobilidade, o próprio nó que realizou a movimentação deve informar ao nó correspondente e ao agente permanente (*Home Agent*) o novo endereço, caso tenham ocorrido alterações do mesmo. Um exemplo é ilustrado na Figura 2.14**;**

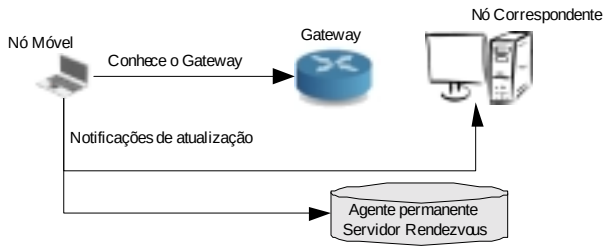

Figura 2.14. Classe Ia: o nó controla a sua mobilidade

b) No caso da Classe Ib, o nó móvel não tem controle sobre o *gateway* a que o pacote será realmente entregue. Logo, a técnica padrão do IP Móvel é incapaz de detectar a mobilidade utilizando mecanismos próprios. Contudo o *gateway* pode ajudar o nó móvel, pois possui a capacidade de perceber a mobilidade ao analisar o endereço de origem de cada pacote, e ao perceber alguma nova origem, informar ao nó de origem da mudança ocorrida. O nó móvel, ao ser notificado da alteração, pode cuidar dos ajustes necessários. Um exemplo é ilustrado na Figura 2.15.

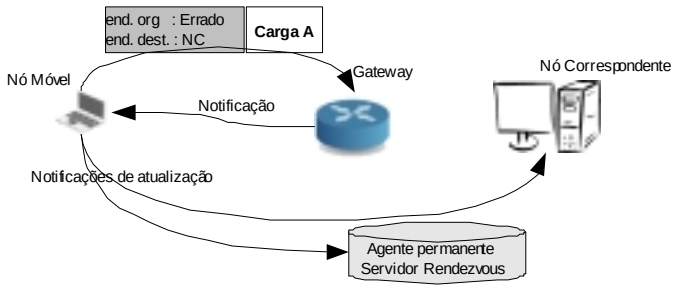

Figura 2.15. Classe Ib: o *gateway* notifica o nó sobre sua mobilidade

2.2.3.2 Implementação da Detecção de Mobilidade na Classe II

Em alguns casos, é interessante remover da rede sem fio e do nó móvel a manipulação da mobilidade, delegando ao *gateway* a sobrecarga de gerência das questões relacionadas à mobilidade. Em tal situação, a classe II de detecção de mobilidade pode ser mais a mais adequada.

Nos protocolos IP Móvel e HIP, o nó móvel possui a responsabilidade de sempre informar aos outros nós que participam dos protocolos sobre o evento de mobilidade e possíveis alterações na localização. O próprio *gateway* é candidato a tomar esta responsabilidade para si e absorver a carga de gerenciar a comunicação envolvida no evento de mobilidade.

Assim pode-se utilizar qualquer protocolo de roteamento dentro do domínio da rede sem fio para controlar o roteamento dos pacotes provenientes do nó móvel com destino ao *gateway*, mesmo que isso signifique passar por inúmeros intermediários. Considerando que o protocolo de roteamento deve enviar os pacotes de um nó móvel a um mesmo *gateway*, mudando somente no caso da topologia ser alterada para que se utilize outro *gateway* mais adequado a nova topologia.

A classe II, ao permitir que qualquer protocolo de roteamento seja utilizado na rede interna, torna-se adequada em redes sem fio do tipo *mesh*, pois redes deste tipo

usualmente empregam protocolos auto-gerenciáveis, com medidas de reparação de rotas em cada salto.

Para o IP Móvel, uma implementação proposta pode ser vista na Figura 2.16. Quando o nó móvel tenta enviar um pacote pelo *gateway*, este percebe o novo nó pela análise do campo endereço de origem (*Home Address*) contido no pacote. Então o *gateway* extrai o endereço do agente permanente (*Home Agent*) contido no pacote, para em seguida informar ao agente sobre a nova localização do nó móvel. O *gateway* modifica diversos campos do pacote, incluindo o endereço de origem usando o próprio endereço (do *gateway*), bem como colocando o endereço permanente (*Home Address*) na opção de agente permanente do cabeçalho móvel [Perkins 2000]. Depois destas modificações no pacote, o *gateway* repassa ao nó correspondente o pacote atualizado. No nó correspondente, o endereço IP de origem (do *gateway*) é trocado pelo endereço permanente do nó móvel e vice-versa. Logo o nível de transporte do nó correspondente não percebe que houve qualquer movimentação do nó móvel. Para o IP Móvel no padrão IPv4, é possível usar encapsulamento UDP a partir do *gateway*, em vez do cabeçalho móvel. Uma solução similar pode ser implementada no HIP, usando encapsulamento UDP, no caso de IPv4, ou o cabeçalho móvel, para IPv6.

Após a notificação realizada pelo *gateway* aos nós que participam dos protocolos (*Home Agent* ou *Rendezvous Server* e *Correspondent Hosts*), o agente móvel ou servidor *Rendezvous* pode notificar ao antigo *gateway* que desative o controle do cliente que o deixou.

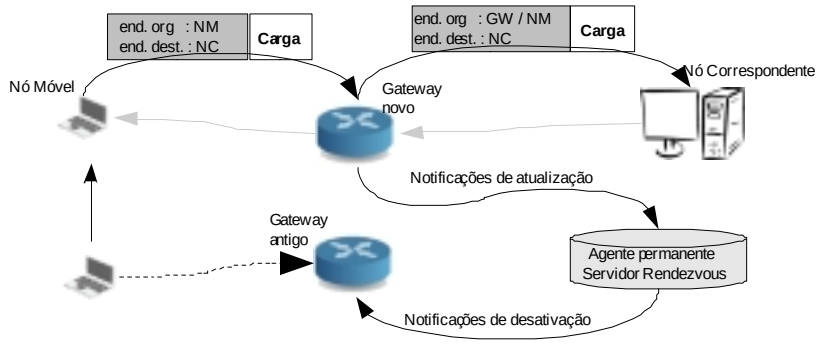

Figura 2.16. Classe II: *gateway* gerencia completamente a mobilidade

2.2.4 Comentários Finais

Para avaliar as soluções descritas anteriormente, podemos identificar alguns critérios importantes:

- Suporte a conectividade global por IP. Um nó é capaz de se conectar a Internet e ser alcançável por ela,
- Suporte à mobilidade nos pontos finais de uma comunicação,
- Endereços topologicamente corretos. Todos os nós na rede *mesh* devem possuir o prefixo das redes dos *gateways* que são utilizados para acessar a Internet,
- Suporte a múltiplos *gateways*. Um nó é capaz de se conectar ao melhor *gateway* de uma rede e de trocar para outro *gateway* enquanto realiza

uma movimentação,

- Suporte a manutenção da conectividade no nível de rede durante a movimentação,
- Suporte a mobilidade permitindo que um nó mantenha as conexões ativas da camada de transporte.

Além dos critérios mais importantes, outros critérios também são desejáveis:

- Não usar intermediários. É interessante não adicionar um novo nó externo com a finalidade de ser um intermediário fixo entre nós móveis,
- Compatível com a infra-estrutura atual da Internet. Soluções não devem exigir grandes modificações para serem suportadas,
- Permitir a um nó móvel ser um servidor. O nó móvel deve poder ser alcançado por um nó na Internet, sem ter que iniciar a comunicação. De preferência este último não precisa saber sobre a característica de mobilidade do primeiro,
- Escalabilidade em números de nós em cada *gateway*. A solução não deve impor grande carga no *gateway* que servir cada nó móvel,
- Ferramentas para suportar a solução devem provocar o menor impacto em qualquer dispositivo em que forem instaladas.

A Tabela 1 compara as soluções apresentadas considerando os critérios acima mencionados. Um critério por linha e uma técnica por coluna. Cada célula contém a avaliação da técnica segundo um critério, e esta avaliação é representada por dois símbolos, sendo o símbolo "+" representando um ponto positivo segundo o critério, e o símbolo "-" como ponto negativo.

Tabela 1. Resultado da avaliação das técnicas sob os critérios apresentados

	IP Móvel	**NAT**	**MobileNAT**	**HIP**
Critérios Importantes				
Conectividade IP global	+	+	+	+
Endereços topologicamente corretos	+	+	+	+
Suporte a múltiplos *gateways*	+	-	+	+
Suporte a Mobilidade	+	Apenas micro	+	+
Manutenção das conexões de transporte ativas	+	Apenas em micro	+	+
Critérios Desejáveis				
Sem ponto de apoio externo	-	+	-	-
Compatibilidade com a infra-estrutura atual da Internet	+	+	+	-
Nó móvel poder trabalhar como servidor	+	+	+	+
Escalabilidade no número de nós por *gateway*	+	-	+	+
Baixo custo de sobrecarga (*overhead*)	-	+	-	-

Como conclusão, o IP Móvel, apesar de suas deficiências, possui uma boa avaliação pelos critérios apresentados, seguido do MobileNAT que obteve a mesma avaliação. A principal diferença está na capacidade de conviver com implementações atuais da técnica NAT, amplamente utilizada no cenário atual da Internet pelo número limitado de endereços que o padrão IPv4 oferece. Possivelmente com a adoção do padrão IPv6, com seu amplo espaço de endereçamento e funcionalidades adicionais relativas a mobilidade, o IP Móvel será uma opção mais adequada.

O protocolo HIP possui uma avaliação interessante, contudo o seu ponto negativo é a necessidade dos nós fixos, que não realizam mobilidade, de sofrerem alterações iguais aos nós móveis.

Por fim, a técnica NAT tradicional, por não ter características de suporte a mobilidade, mesmo com uso de túneis, apresenta limitações, portanto é a técnica menos interessante.

2.3. Qualidade de Serviço em Redes *Mesh*

Qualidade de serviço (QoS) segundo a RFC 2386 [Crawley et al. 1998] é caracterizada como o conjunto de requisitos de serviços a ser atendido pela rede enquanto transporta fluxos de dados a partir da origem a um destino. Portanto, refere-se à capacidade das redes em oferecer algumas garantias de serviço para certos tipos de tráfego, por meio de tecnologias específicas.

Entre os parâmetros que a qualidade de serviço controla, destacam-se: largura de banda, atraso, variação do atraso (*jitter*), taxa de bloqueio, proteção, prioridade e resiliência, necessários, principalmente, para aplicações multimídia interativas de tempo real. Estes parâmetros, em geral, podem ser negociados pelo usuário (aplicação) com a prestadora de serviço através de contratos de nível de serviço (SLAs – *Service Level Agreements*).

A provisão de qualidade de serviço torna-se cada vez mais crítica no desenvolvimento das redes *ad-hoc*, em especial nas redes *mesh*, em função da necessidade do atendimento às aplicações multimídia, cada vez mais utilizadas pelo acesso à Internet em banda larga, uma vez que essas aplicações são tipicamente sensíveis ao atraso e necessitam de maior banda.

As soluções em qualidade de serviço para redes *ad-hoc* podem ser classificadas com base nas camadas da pilha clássica de protocolos de rede [Gavrilovska e Atanasovski 2005]. Cada camada possui mecanismos apropriados para tratar o problema, como por exemplo:

- Camada MAC/LL: as soluções representam extensões ao protocolo de acesso ao meio ou novos esquemas. Podemos citar o padrão IEEE 802.11e.
- Camada de rede: as soluções representam extensões e otimizações aos protocolos de roteamento *ad-hoc* tradicionais.
- *Cross-Layer*: as soluções representam propostas de novos esquemas que utilizem o conceito de *cross-layer*, ou seja, que permitam a troca de informações entre as camadas.

Qualidade de serviço em redes *ad-hoc* vem sendo estudada há certo tempo. [Chakrabarti e Mishra 2001] apresenta uma sólida discussão a respeito de qualidade de serviço em redes *ad-hoc* enquanto [Baoxian e Mouftah 2005] avalia e estuda protocolos de roteamento *unicast* em redes *ad-hoc* para prover QoS. De um modo geral, a maioria das propostas acaba sendo muito restrita, ou seja, apresentam bons resultados para cenários ou aplicações específicas. As propostas podem ser divididas entre aquelas que apresentam contribuições para os protocolos de roteamento e outras que sugerem mecanismos para complementar o provisionamento de QoS.

A maioria das propostas de QoS em redes *ad-hoc* apresentam modificações ou novas versões de protocolo de roteamento. [Badis e Agha 2004] propõe uma extensão do OLSR, denominado QOLSR (*QoS routing over OLSR*) como uma proposta de serviço integrado. [Badis e Al Agha 2004] analisa o QOLSR com uma métrica múltipla composta por banda e atraso. Já [Badis et al. 2004] utiliza o QOLSR sem reserva explícita. Uma das vantagens do QOLSR é que o protocolo é capaz de manipular várias métricas de QoS. Embora apenas rotas do tipo *shortest-widest paths* sejam calculadas por padrão, outras métricas também podem ser utilizadas. Uma das desvantagens do QOLSR é que as métricas de largura de banda e atraso, métricas básicas nesta variação do OLSR, são difíceis de medir utilizando a camada MAC IEEE 802.11.

Ainda na linha de trabalhos com propostas de extensões ao protocolo OLSR, [Nguyen e Minet 2005] propõe uma solução de roteamento fazendo uma extensão do protocolo OLSR, levando em consideração interferência e reserva de banda, sendo baseada no controle de admissão. O trabalho realiza, através de simulação, uma comparação da extensão proposta com o protocolo na sua versão original, validando melhor desempenho da extensão proposta para o cenário avaliado com 200 nós em uma área de 2500x2500m^2, com 7 fluxos CBR, fazendo uso mínimo de 5 saltos e máximo de 16 saltos.

Já na linha de trabalhos com novas propostas de protocolos, temos [Barua e Chakraborty 2002] e [Hsu et al. 2006]. No primeiro, é proposto um protocolo de roteamento denominado ACRQ (*Adaptive Cluster-based Routing with QoS support*) que realiza a descoberta de rotas baseada em *clusters*. Além de gerenciar as rotas dinâmicas, faz uso de um algoritmo distribuído para selecionar e rearrumar os clusters. No segundo, é apresentado um protocolo de roteamento com suporte a QoS através de reserva de banda sob demanda para redes MANETs baseadas na utilização do acesso TDMA (*Time Division Multiple Access*). Para tanto, trabalha com um algoritmo que escolhe a rota que mais atende os requisitos de QoS e outro algoritmo que reserva o *time slot* apropriado e mantém os demais livres para outras requisições.

Outro grupo de pesquisadores sugere contribuições complementares aos protocolos de roteamento com QoS, sendo que muitas são implementadas em níveis diferentes no nível de rede, considerando conceitos de avaliação de desempenho validados matematicamente. [Futernik at al. 2003], por exemplo, desenvolve um modelo analítico, através de uma cadeia de markov, para avaliar a qualidade de serviço em redes *ad-hoc*. Faz um estudo em cima da interferência de canal e QoS, comparando os resultados com simulações. Em [Alam et al. 2006], os autores propõem um esquema de cálculo de pesos para os fluxos de serviços de melhor esforço e fluxos de serviços garantidos, baseado em um modelo de filas usando selo de tempo. O objetivo é garantir

a banda necessária para os serviços de melhor esforço e possibilitar um compartilhamento igual de banda para os demais fluxos.

Diferente dos trabalhos citados acima, outros não fazem extensão a protocolos de nível 3 existentes e nem propõem novos protocolos. [Ying et al. 2003] apresenta uma técnica de otimização para o protocolo MAC que garanta QoS, através de um mecanismo de escalonamento distribuído, fazendo a reserva de canais de acesso para as sessões em tempo real. As transmissões dos pacotes e as novas sessões criadas são garantidas apenas após a verificação da reserva, para que esta não degrade as sessões de tempo real em andamento.

No que diz respeito ao oferecimento de QoS especificamente para redes *mesh*, requisitos especiais precisam ser considerados uma vez que as redes *mesh*, em especial no seu *backbone*, são organizadas de maneira diferenciada das redes *ad-hoc* tradicionais, através de múltiplos saltos (*hops*).

O primeiro aspecto a considerar é que o tráfego de *backbone* das redes *mesh* normalmente flui de/para *gateways* cabeados e entre estações móveis associadas a diferentes pontos de acesso (roteadores) sem fio. Em segundo lugar, deve-se ressaltar que existem dois pontos cruciais para as redes *ad-hoc* tradicionais: a mobilidade dos nós e o consumo de energia. Esses pontos não são relevantes para as redes *mesh*, em especial no seu *backbone*, uma vez que os roteadores geralmente são fixos e com energia cabeada.

Neste contexto, [Aoun et al. 2006] analisa QoS através da otimização do posicionamento mínimo de *gateways* na rede. O trabalho utiliza um algoritmo recursivo, validado por análise matemática e simulações, com comparações do algoritmo proposto e algoritmos interativos existentes, tendo como resultado a redução de 50% do número de *gateways* necessários, sendo que nenhum *cluster* é formado até que todos os requisitos de QoS tenham sido atendidos.

Considerando a arquitetura das redes mesh e sua escalabilidade, temos [Bok-Nyong et al. 2006], [Huang et al. 2006] e [Zhao et al. 2006]. O primeiro trabalho apresenta uma arquitetura de rede *mesh* voltada para o uso doméstico, expondo de forma geral a arquitetura e propondo um esquema de roteamento com suporte a QoS, em especial na descoberta e seleção das rotas a serem utilizadas, sendo validado por simulações. No segundo artigo, é realizada um análise da escalabilidade das redes *mesh*, considerando o planejamento de cobertura e QoS em termos de capacidade e atraso, com a proposta de um modelo analítico para validar a vazão considerando o CSMA, a distância em saltos relacionados com a camada física. O terceiro e último trabalho, baseado no esquema tradicional DCF, propõe o esquema MDCF (*Mesh* DCF) para interconectar um grande número de pontos de acesso para formar um eficiente ESS (*Extended Service Set*).

Já [Carlson et al. 2006] propõe um protocolo chamado DARE (*Distributed end-to-end Allocation of time slots for Real-timE traffic*) que funciona com reserva fim-a-fim para suportar qualidade de serviço na camada de controle de acesso ao meio nas redes *mesh*. Basicamente o protocolo reserva periodicamente, repetitivos *time slots* para as aplicações que necessitem de QoS, enquanto mantém a função de coordenação distribuída (DCF) para as aplicações de melhor esforço. O artigo realiza uma avaliação

de desempenho do protocolo DARE em comparação ao EDCA (*Enhanced Distributed Channel Access*) utilizado no padrão IEEE 802.11e.

Ainda que em menor número, atualmente os conceitos de qualidade de serviço e redes *mesh* já vêm sendo estudados em conjunto com a tecnologia de acesso WIMAX (*Worldwide Interoperability for Microwave Access*), como por exemplo em [Yan et al. 2006], que propõe e analisa um esquema que suporte a QoS em redes WIMAX no modo ponto-multiponto.

Com a mesma escassez, encontramos artigos que abordem o conceito de *cross-layer* em redes mesh, como em [Jiang et al.2006], onde é proposto um mecanismo de roteamento fazendo uso de *cross-layer* e serviços diferenciados. Mantendo a linha de análise de roteamento, e ainda envolvendo inteligência artificial, cita-se [Lianggui e Guangzeng 2005], onde é proposto um esquema de roteamento baseado em MFN (*Mean Field Network*).

No contexto das bandas não licenciadas de redes sem fio, onde se enquadram as redes *mesh*, em geral, há uma maior probabilidade de ocorrência de problemas que podem comprometer a qualidade dos serviços oferecidos, uma vez que a implantação está aberta a todos. No entanto, avanços nos padrões associados e nas tecnologias relacionadas ajudam a minimizar essas questões, tais como a interferência em múltiplos caminhos (*multipath interference*).

Além disso, outra característica de redes *mesh* é que seu o *backbone* pode ter algumas centenas de roteadores sem fio, caracterizando assim a escalabilidade da rede, sendo este um dos principais conceitos para a provisão de qualidade de serviço, discutido em [Jiang et al.2006] e que será abordado na próxima seção.

2.3.1 - *Wireless DiffServ*

Como apresentado na seção anterior, técnicas para o provisionamento de qualidade de serviço para as redes sem fio tradicionais vêm sendo intensamente investigadas, como por exemplo para as redes celulares [Lin e Lui 1999] e para as redes *ad-hoc* [Chakrabarti e Mishra 2001]. Entretanto, apesar das redes *mesh* herdarem muitas características das redes *ad-hoc*, no seu *backbone* percebem-se diferenças, as quais requerem tratamento diferenciado dos mecanismos de QoS tradicionais. Dentre essas características diferenciadas, citam-se duas:

- O *backbone* das redes *mesh* suporta o tráfego de/para o(s) *gateway*(s) cabeados e o tráfego entre os usuários associados com diferentes MPs (*Mesh Points*) com função de roteadores. Desta forma, pode-se ter conexões através de múltiplos saltos de forma hierárquica.
- Diferentemente das redes *ad-hoc* tradicionais, no *backbone* das redes *mesh*, não há restrição de consumo de energia e nem preocupação com mobilidade, uma vez que os MPs roteadores são usualmente fixos e bem alimentados com energia cabeada.

Considerando a escalabilidade como uma das principais características das redes *mesh*, em especial no seu *backbone*, o mecanismo *diffserv* é visto como uma solução aplicável ao *backbone*. A preferência pelo *diffserv* em relação ao mecanismo *intserv*, justifica-se pelos problemas listados abaixo:

- Escalabilidade: seria necessário manter o estado em cada MP roteador para cada fluxo que passa por ele. Isso causaria carga enorme nos MPs roteadores e grande *overhead* no *backbone*.
- Flexibilidade: as classes de QoS são poucas e quantitativas. Isso não permitiria dar tratamento preferencial a uma classe.

Em uma rede sem fio *diffserv*, o nó usuário pode iniciar uma conexão que passe por várias redes interconectadas. A rede sem fio *diffserv* está interconectada com outras redes *diffserv*. Cada rede *diffserv* pode, de forma independente selecionar, modificar ou trocar seu mecanismo de gerenciamento de recursos para implementar seu SLA com as redes vizinhas. Neste caso, a qualidade de serviço fim-a-fim seria garantida com os SLAs sendo atendidos em cada rede.

Cada MP roteador sem fio atuará como um roteador de borda (para classificação do tráfego, por exemplo) para o MP ou nó usuário sob a sua área de cobertura. O roteador coletará os requisitos de serviços dos seus usuários sob sua cobertura e os agregará em um requisito único de SLA para o *backbone*. O roteador também atuará como um roteador de núcleo (para encaminhamento dos pacotes de acordo com sua classe, por exemplo). Todos os roteadores usarão mecanismos de filas controlados por algoritmos de escalonamento para prover classes de serviços diferenciadas.

A Figura 2.17 representa a promissora interconexão heterogênea que podemos ter entre redes sem fio e cabeadas, de modo a prover qualidade e serviço fim-a-fim, buscando ainda a realização de *roaming* entre as estações móveis.

Nela, visualiza-se quatro domínios (redes) *diffserv*:

- O *backbone* da Internet, ilustrando os roteadores de borda e os roteadores de núcleo. Os roteadores de borda, fazendo a conexão com as demais redes através dos roteadores *gateways*.
- A rede *mesh*, ilustrando o seu *backbone* formado por pontos de acesso com funcionalidades de roteador e um acesso de usuário a um desses pontos de acesso.
- A rede cabeada.
- O núcleo de uma rede celular.

As principais diferenças entre o mecanismo *diffserv* tradicional e o *diffserv* sem fio são:

- O roteador pode servir tanto como roteador de borda quanto como roteador de núcleo. Entretanto quando atua como roteador de borda será apenas para um número limitado de nós usuários, no caso apenas para os nós sob sua área de cobertura.
- Não existe um controle centralizado (BB – *Bandwidth Broker*). A alocação de recursos deve ser executada de forma distribuída.
- No que diz respeito ao contrato de serviço, os SLAs podem ser associados a um ou vários *gateways* simultaneamente, de modo que a carga de tráfego seja distribuída.
- Em função do ambiente *broadcast* e do acesso ao meio ser compartilhado, além da camada de rede, as camadas física e de enlace são levadas em consideração quando for realizada a provisão da qualidade de serviço.

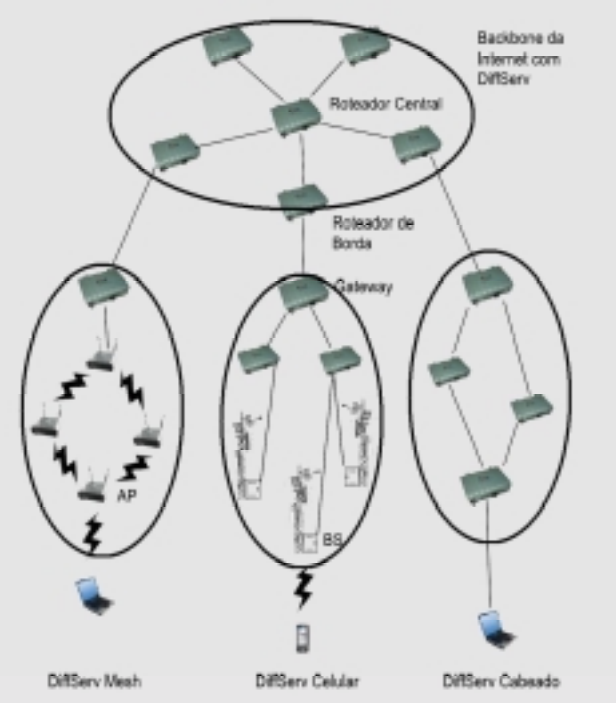

Figura 2.17. Interconexão de redes sem fio e cabeadas usando *diffserv* (Fonte [Jiang et al. 2006])

2.3.2 - Classes de serviços

Basicamente, não existem diferenças nas classes de serviços ofertadas nas redes cabeadas e nas redes sem fio. A rede sem fio *diffserv* define os serviços:

- Prêmio (*Premium*): provê baixa perda, baixos atraso e *jitter*, sendo recomendado para aplicações de tempo real como VoIP (*Voice over IP*) e videoconferência.
- Garantido (*Assured*): é satisfatório para usuários que requerem serviços garantidos do seu provedor de serviço com uma taxa designada e especificada no SLA.
- Melhor Esforço (*Best-effort*): qualidade conhecida na Internet por não apresentar garantia nenhuma.

Para o encaminhamento de tráfego no núcleo da rede, serão usados o EF PHB (*Expedited Forwarding Per-Hop Behavior*) associado ao serviço prêmio e o AF (*Assured Forwarding*) PHB associado ao serviço garantido. Entretanto, para garantir a diferenciação de serviços em cada salto e QoS fim-a-fim no *backbone* da rede *mesh*, alguns pontos precisam ser trabalhados, como por exemplo, roteamento com QoS e mecanismos de acesso ao meio.

2.3.3 – Roteamento e QoS

O número de saltos é o critério mais comum adotado pelos protocolos de roteamento tradicionais. Entretanto, torna-se claro que esses protocolos são inadequados para as aplicações multimídia, como por exemplo VoIP e videoconferência, as quais necessitam de garantias de QoS.

Protocolos de roteamento com QoS, não necessitam apenas achar a rota com o menor caminho, mas sim a melhor rota que satisfaça os requisitos de QoS fim-a-fim, independente do número de saltos.

A topologia de uma rede *mesh* envolverá roteamento através de múltiplos saltos. Desta forma, destacam-se três pontos importantes a serem estudados:

- Protocolos de roteamento: reativos, pró-ativos, adaptativos ou híbridos. Para redes *mesh*, recomenda-se o uso de protocolos híbridos, uma vez que de acordo com a abrangência da rede (número de nós ativos), as características dos protocolos pró-ativos serão vantajosas nas redes com menor número de nós ativos e as características dos protocolos reativos, serão válidas para redes com maior número de nós.
- Protocolos de roteamento de nível 2 (camada MAC) x de nível 3 (IP). Para utilizar roteamento na camada 3, seria necessário que todos os nós *mesh* fossem estações com funcionalidades desta camada. Entretanto, os nós *mesh* podem ser compostos por pontos de acesso, que são equipamentos de camada 2, portanto incapazes de decodificar pacotes IP. O ideal seria que todos os equipamentos da rede *mesh* suportassem uma integração entre as funcionalidades da camada 2 com a camada 3, conceito conhecido na literatura como *cross-layer*. Entretanto isso é considerado um desafio, uma vez que foge do escopo de padronização tanto do IETF (*Internet Engineering Task Force*) quanto do IEEE (*Institute of Electrical and Electronics Engineers*).
- Análise de métricas multidimensionais. A métrica tradicional, número de saltos (*hops*), passa a ser insuficiente nas redes *mesh*, uma vez que não conseguirá prover um roteamento consistente e nem suportará a complexidade de diferentes níveis de QoS, banda, latência e requisitos de segurança. O ideal será uma métrica multidimensional capaz de capturar informações de cada enlace (QoS, potência, segurança, etc).

Torna-se clara, a ligação da qualidade de serviço com o roteamento. Abaixo, citamos alguns trabalhos relacionados, que buscam de alguma forma efetuar o roteamento nas redes *ad-hoc* e *mesh* com garantias de QoS.

Matematicamente, [Alicherry et al. 2006] formula o problema de roteamento e utilização de canais em redes *mesh*, considerando interferência disponibilidade e alocação de canais nos pontos de acesso com funcionalidades de roteadores.

O conceito de *cross-layer* é abordado e analisado em [Krishnaswamy et al. 2006] e [Song e Fang 2006]. O primeiro apresenta uma arquitetura com a comunicação entre as camadas, que regularmente troca parâmetros dos serviços armazenados em bases de dados distribuídas, explorando a idéia de otimizar o roteamento com o uso de múltiplos fluxos, os quais extraem da rede parâmetros úteis a partir da base de dados, construindo rotas que satisfaçam os requisitos de atraso e banda das aplicações. Já o segundo propõe um algoritmo de roteamento com a comunicação entre as camadas, para obter controle de congestionamento e balanceamento de tráfego nas redes *mesh*.

Ainda que em menor número, extensões do protocolo AODV são apresentadas em [Farkas et al. 2006], [Kuo e Liang 2006] e [Subramanian et al. 2006]. O primeiro

artigo, através de simulações, analisa e avalia extensões de qualidade de serviço, como por exemplo, políticas de controle de taxas e fila de prioridades, para o roteamento em redes *ad-hoc*, considerando aplicações de tempo real. O segundo trabalho, propõe a extensão denominada de AODV-MM (*AODV with Meshed Multipath*), fazendo uso da criação de múltiplas rotas alternativas com baixo *overhead*, resultando em um melhor desempenho na entrega dos pacotes. Por último, o terceiro artigo, apresenta a extensão denominada AODV-MR (*AODV- Multi-Radio*), na qual uma métrica de roteamento denominada iAWARE (*interference aware*) seleciona a melhor rota em termos de interferência intra e inter fluxo.

Ainda na abordagem de proposição de novas métricas, em [Koksal e Balakrishnan 2006], são apresentadas duas métricas para roteamento com qualidade de serviço, denominadas de mETX (*Modified Expected Number of Transmissions*) e ENT (*Effective Number of Transmissions*). Enquano que em [Bin et al. 2006] é estudado o ganho no desempenho no roteamento quando a rota é escolhida baseada na codificação utilizada.

Em [Jaseemuddin et al. 2006], é proposto um sistema integrado de roteamento tanto para o *backbone* da rede quanto para o seu acesso, em conjunto com um esquema de descoberta de rotas. No *backbone* foi investigado o uso do protocolo OSPF (*Open Shortest Path First*), resultando em propostas de solução para configuração do protocolo no *backbone* e sua auto-configuração no demais enlaces sem fio.

Na linha de propostas de novos protocolos, temos [Nandiraju et al. 2006] e [Rozner et al. 2006]. O primeiro artigo propõe um protocolo de roteamento híbrido através de múltiplos saltos, denominado MMESH (*Multipath Mesh*) que efetivamente realiza a descoberta das rotas, e em conjunto propõe um algoritmo para realizar o balanceamento do tráfego objetivando o melhor desempenho da rede. No segundo trabalho, encontramos a proposta do protocolo SOAR (*Simple Opportunistic Adaptive Routing Protocol*), o qual maximiza o progresso de cada pacote através de temporizadores com prioridade para assegurar que o encaminhamento do pacote seja realizado pelo nó com menor *overhead* na rede, sendo validado através de simulações.

2.3.4 - Desafios

As redes *mesh*, quando comparadas com outros modelos de rede sem fio, apresentam desafios especiais, porque o meio sem fio é compartilhado pelos nós adjacentes e a topologia pode sofrer alterações dinamicamente conforme a mobilidade dos nós e a entrada/saída na rede pelos mesmos. Conseqüentemente, qualidade de serviço torna-se uma forte linha de pesquisa, uma vez que prover nível de QoS diferente do melhor esforço não é tarefa trivial, para redes onde os canais de comunicação são imprevisíveis.

Torna-se claro que prover qualidade de serviço em redes *mesh* passa a ser uma tarefa efetuada por todas as camadas, daí a importância do conceito de *cross-layer*, que mesmo não sendo o maior desafio, pode ser considerado como um dos mais complexos, por estar fora do âmbito dos órgãos de padronização.

Diferentes aplicações e seus tipos de tráfegos possuem requisitos diferenciados em termos de níveis de serviços providos pela rede. Esses requisitos são garantidos através de mecanismos de QoS. No contexto das redes *mesh*, dois aspectos são identificados [Faccin et al. 2006]:

- Oferta de QoS na presença de um tráfego misto, envolvendo o tráfego de acesso à rede e o tráfego de *backbone*. Desta forma, fica clara a necessidade de um controle de admissão de chamadas (CAC – *Call Admission Control*) e de um mecanismo para diferenciar esses dois tipos de tráfegos, de modo a garantir que ambos obtenham o nível de serviço apropriado.
- Ofertar garantias de QoS através de múltiplos saltos requer um mecanismo além de nível 2, de modo que fluxos fim-a-fim tenham garantidos seus serviços alocados e previstos.

O mecanismo de controle de admissão de chamadas é necessário em duas situações, conforme visualizado na Figura 2.18. Na primeira, na estação associada aos MPs, de modo a balancear o tráfego de entrada na rede e o tráfego encaminhado através do MP. Na segunda, o mecanismo de controle de admissão de chamadas é aplicado entre os MPs, para controle da carga da rede no *backbone*.

De certa forma esse mecanismo deverá primeiramente determinar a capacidade disponível na rede para aceitar um usuário baseado no tráfego de acesso e de *backbone*, depois aceitar o fluxo de tráfego baseado no tipo, se através de um salto ou múltiplos saltos.

Existe a possibilidade dos MPs trabalharem com agregação de pacotes e encaminhamento instantâneo do tráfego de entrada, baseado no tipo de serviço. Por exemplo, pacotes com alto *throughput*, porém requisitos de maiores atrasos, podem ser agregados, enquanto que pacotes sensíveis ao atraso (VoIP, videoconferência), são encaminhados imediatamente.

Para suportar qualidade de serviço fim-a-fim, pode-se trabalhar ainda com controle de fluxo de modo que os níveis de QoS possam ser mapeados entre o tráfego de acesso e de *backbone*.

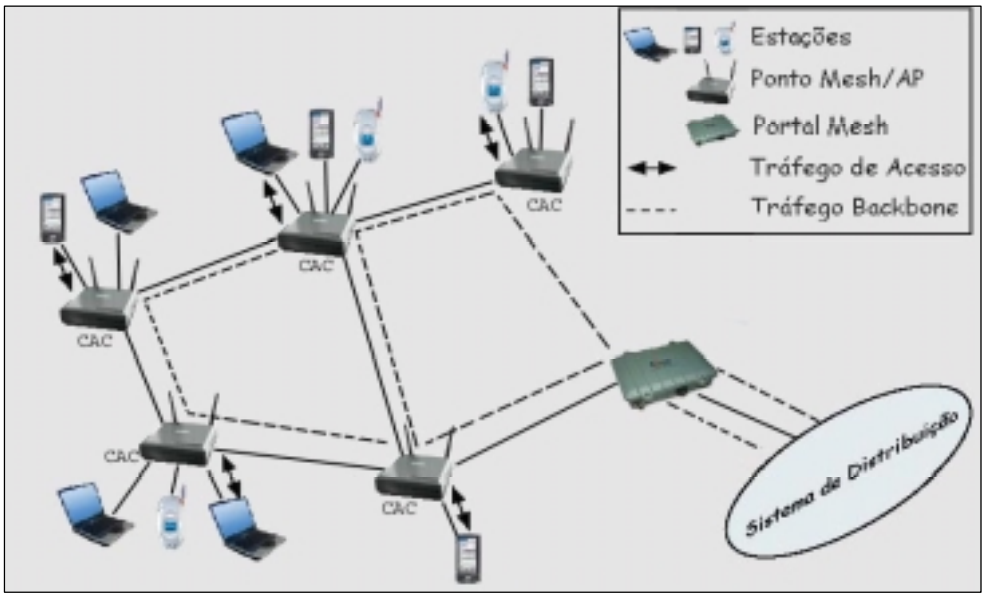

Figura 2.18. Caracterização do tráfego misto (Fonte [Faccin et al. 2006])

Desta forma, o controle de fluxo e o mecanismo de controle de admissão de chamadas estão fortemente ligados e relacionados, de modo que o primeiro é baseado

nos identificadores dos fluxos e o segundo, uma vez definidos os requisitos dos níveis de serviços, é realizado baseado nos cabeçalhos dos pacotes.

Além dos mecanismos de controle de admissão de chamadas e de controle de fluxo, não podemos esquecer a necessidade do balanceamento de carga, o qual deve ser realizado para evitar atrasos no tráfego e descarte de pacotes em função de um conjunto no núcleo da rede ou sobrecarga de um determinado MP.

Sabe-se que garantir qualidade fim-a-fim nas redes *mesh*, por mais que o mecanismo *diffserv* seja utilizado, significa que recursos deverão ser alocados a cada salto na rota escolhida. Isso significa que o protocolo de roteamento a ser utilizado necessita levar em consideração o conceito de qualidade de serviço.

A rota escolhida deverá satisfazer métricas múltiplas, ou seja, métricas que considerem várias informações simultaneamente, como por exemplo banda e atraso, além do tradicional conceito do menor caminho.

O *backbone* de uma rede *mesh* proverá banda larga sem fio, mas poderá também diferenciar entre classes de QoS levadas por suas redes associadas, conforme ilustrado na Figura 2.19 e a utilização dos seus recursos. Dependendo de quantos canais estarão sendo utilizados, a complexidade para essa implementação será maior.

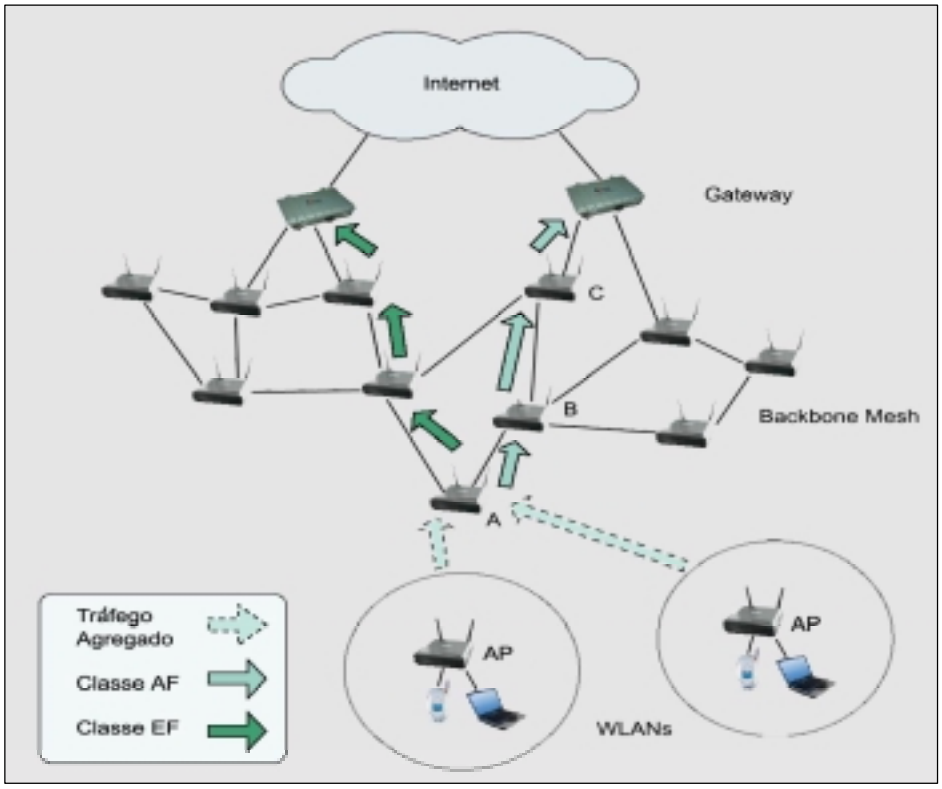

Figura 2.19. Roteamento com classes de serviços [Jiang et al. 2006]

2.4. Comunicação em Grupo em Redes *Mesh*

Comunicação em grupo (*multicasting*) é o processo de transmissão de pacotes de um nó de origem para um grupo de nós de destino identificados por um único endereço, conhecido por endereço de grupo. Oferecer a comunicação em grupo no nível de rede,

ao invés de *multicast* na camada de aplicação, evita a replicação desnecessária de cópias de pacotes transmitidos pelo nó de origem em partes da rede onde não é necessária a transmissão do conteúdo da sessão *multicast*. Transmissões *multicast* ainda não vêm sendo amplamente empregadas em redes *mesh* atualmente. Num futuro próximo, no entanto, tornar-se-ão muito importantes, pois são fundamentais para dar suporte à comunicação multimídia em redes sem fio. É sempre vantajoso usar comunicação em grupo (em relação a *unicast*), especialmente em se tratando de ambientes de redes sem fio, onde os recursos de banda disponíveis são limitados, comparados àqueles disponíveis em redes cabeadas. Comunicação em grupo em uma rede *mesh* é uma tarefa mais difícil que em redes cabeadas, devido às mudanças topológicas mais freqüentes, dadas pela natureza dinâmica das redes *mesh* e pela mobilidade dos nós clientes.

Muitos esforços vêm sendo despendidos em pesquisas sobre protocolos de roteamento desde o advento das redes *ad-hoc* sem fio. Diversos protocolos de roteamento *unicast* foram propostos, possuindo características distintas. A próxima subseção aborda, de forma resumida, as taxonomias mais encontradas na literatura [Murthy e Manoj 2004], [Cordeiro e Agrawal 2002] para classificação dos protocolos de roteamento *unicast*, que facilite o entendimento dos protocolos *multicast* a serem apresentados.

O foco principal desta seção é a discussão sobre transmissão *multicast* em redes em malha sem fio, abordando os principais protocolos de roteamento encontrados na literatura que oferecem comunicação em grupo em redes *ad-hoc* e fazendo uma comparação entre eles. A taxonomia mais referenciada na literatura para classificação dos protocolos de roteamento *ad-hoc multicast* toma por base características relacionadas à topologia empregada por estes algoritmos. De uma forma geral, os protocolos de roteamento *multicast* são divididos em duas categorias: *tree-based e mesh-based*, que serão discutidas detalhadamente a seguir.

Após a apresentação dos protocolos de roteamento *multicast*, é feita uma análise comparativa dos protocolos apresentados, discutindo suas principais vantagens e desvantagens. Além da comparação entre os diversos protocolos apresentados, é discutido o estado atual da provisão de serviço *multicast* nos projetos de redes *mesh* acadêmicas e comerciais.

2.4.1. Protocolos de Roteamento *Unicast*

A maior parte das aplicações em redes *mesh* está baseada em comunicações do tipo *unicast*. Ao contrário dos *hosts* de uma rede tradicional cabeada, os nós em uma rede sem fio estão livres para se movimentar arbitrariamente. Em função disto, a topologia da rede pode mudar com muito mais freqüência, quando comparada a de uma rede cabeada. Diversos protocolos de roteamento foram propostos na literatura, utilizando mecanismos de operação distintos, de modo a minimizar o impacto introduzido pela mobilidade.

A escolha da rota ocorre no nível de rede (camada IP), consistindo da transmissão de pacotes de dados de um nó de origem (N_O) até um nó de destino (N_D), através de nós intermediários (N_I). O procedimento de encaminhamento é relativamente simples: o N_I apenas compara o endereço constante do cabeçalho do pacote de dados com as entradas existentes em sua tabela de roteamento (T_{ROT}), através da qual obtém a

indicação do próximo salto (próximo N_I ou o próprio N_D). O problema maior é a maneira como são construídas e atualizadas as T_{ROT} dos nós numa rede, que deve ser resolvido pelos protocolos de roteamento.

As taxonomias mais referenciadas na literatura para classificação dos protocolos de roteamento *ad-hoc unicast* abrangem duas características essenciais dos algoritmos empregados nestas redes:

(1) Filosofia de roteamento:

- Roteamento plano: todos os nós são iguais, isto é, possuem a mesma responsabilidade na rede; o roteamento de pacotes é feito baseado em comunicações ponto-a-ponto, restringido apenas pelas condições de propagação. Todos os nós possuem uma visão global da rede, o que acarreta constante troca de informações sobre a disposição da topologia, aumentando sensivelmente a sobrecarga nos enlaces de comunicação. Porém, no que se refere ao descobrimento de rotas, o processo é simplificado pelo fato de todos os nós possuírem informações globais da rede; e
- Roteamento hierárquico: no roteamento hierárquico, os nós são divididos em zonas. Dentro de sua própria zona, o nó conhece os detalhes sobre como rotear pacotes para os demais; em contrapartida, o nó conhece muito pouco sobre a estrutura interna de outras zonas. Geralmente, em cada zona, ao menos um nó é designado para servir como *gateway*, de modo a prover a comunicação entre duas zonas distintas.

(2) Construção de rotas:

De uma forma geral, os protocolos de roteamento *unicast* são divididos nas seguintes categorias:

- Pró-ativos (ou *table-driven*): nos protocolos pertencentes a esta categoria, cada nó mantém informações atualizadas sobre a topologia da rede na forma de T_{ROT}, através da troca regular de informações de roteamento. Estas informações são, geralmente, difundidas utilizando *flooding* por toda a rede. Sempre que um nó precisar de um caminho até o destino, este utiliza um algoritmo apropriado para determinação de rota com base nas informações de topologia ou de melhor caminho mantidas pelo próprio nó. São exemplos de protocolos de roteamento *unicast* pró-ativos: **DSDV** (*Destination-Sequenced Distance-Vector*) [Perkins e Bhaqwat 1994], **WRP** *(Wireless Routing Protocol)* [Murthy e Garcia 1996], **OLSR** *(Optimized Link State Routing)* [Clausen e Jacquet 2003], **CGSR** *(Cluster-Head Gateway Switch Routing)* [Chiang 1997], **FSR** *(Fisheye State Routing)* [Gerla, Hong e Pei 2002] e **GSR** *(Global State Routing)* [Chen e Gerla 1998];
- Reativos (ou *on-demand*): os protocolos pertencentes a esta categoria não trocam informações de roteamento periodicamente, mas sim obtém o caminho, quando necessário, através de processo de descoberta de rotas entre os nós de interesse. São exemplos de protocolos de roteamento *unicast* reativos: **DSR** *(Dynamic Source Routing)* [Johnson, Maltz, Hu e Jetcheva 2002], **AODV** *(Ad Hoc On-Demand Distance Vector)* [Perkins, Belding-Hoyer e Das 2003], **TORA**

(Temporally-Ordered Routing Algorithm) [Park e Corson 1997] e **ABR** *(Associativity-Based Routing)* [Toh 1997]; e

- Híbridos: os protocolos pertencentes a esta categoria combinam as melhores características das duas categorias anteriores - pró-ativos e reativos. São exemplos de protocolos de roteamento *unicast* híbridos: **ZRP** *(Zone Routing Protocol)* [Haas 1997], [Haas, Pearlman e Samar 2002] e **CEDAR** *(Core Extraction Distributed Ad Hoc Routing)* [Sinha, Sivakumar e Bharghavan 1999].

2.4.2. Protocolos de Roteamento *Multicast*

Os protocolos de roteamento *multicast* representam um importante papel para prover a comunicação entre grupos de nós, pois são os responsáveis pela construção da Árvore *Multicast* (A_{MC}) ou Malha *Multicast* (M_{MC}), que consiste de um grafo conexo acíclico contendo todos os nós da árvore pertencentes ao Grupo *Multicast* (G_{MC}). Em uma rede cabeada o transmissor em uma sessão *multicast* envia pacotes, que são encaminhados a todos os nós da árvore *multicast*, percorrendo cada nó e cada enlace uma única vez. Esta estrutura *multicast* não é apropriada para ambientes *ad-hoc* porque a árvore pode facilmente ser rompida pela dinamicidade da rede.

As estruturas das redes *multicast* não são estáveis e necessitam ser reconstruídas continuamente na medida em que ocorrem mudanças na rede. Manter uma estrutura de roteamento *multicast* com o propósito de realizar transmissões *multicast*, quando a topologia de rede sofre alterações freqüentes, pode implicar um tráfego de controle substancial. As estruturas empregadas pelos protocolos de roteamento *multicast* de redes cabeadas convencionais utilizam técnicas do tipo *link state* ou *distance vector*. A troca freqüente de vetores de distância ou de tabelas de estado dos enlaces, provocadas por constantes mudanças de topologia, acrescenta um *overhead* de controle e processamento excessivo. Além disto, um período de instabilidade nas tabelas de roteamento pode levar à instabilidade da A_{MC}, e representar, por conseguinte, um maior retardo de transmissão, perda e retransmissão de pacotes. Desta forma, os protocolos de roteamento *multicast* convencionais para redes cabeadas não são aplicáveis às redes sem fio.

A disponibilidade de banda limitada, a mobilidade dos nós (com recursos de energia também limitados), segurança, dentre outros aspectos, tornam o projeto de protocolos de roteamento *multicast* para redes *ad-hoc* sem fio um verdadeiro desafio. Requisitos como robustez, eficiência, *overhead* de controle, qualidade de serviço, dependência de protocolos de roteamento *unicast* e uso eficiente de recursos limitados como energia e memória devem ser considerados no projeto de um protocolo de roteamento *multicast*:

- *Robustez*: devido à mobilidade dos nós, quedas nos enlaces de redes sem fio são freqüentes, ocasionando perda de pacotes que trafegam pela rede (resultando em uma taxa de entrega de pacotes mais baixa). Um protocolo de roteamento *multicast*, portanto, deve ser robusto o suficiente para suportar a mobilidade dos nós a garantir uma alta taxa de entrega de pacotes;
- *Eficiência*: em um ambiente de redes *ad-hoc*, onde banda é um recurso escasso, a eficiência de um protocolo de roteamento *multicast* é um fator de extrema importância. A eficiência de um protocolo *multicast* é definida como sendo a

razão entre o número de pacotes de dados recebidos pelos nós e o número total de pacotes (dados e controle) transmitidos pela rede [Murthy e Manoj 2004]. Quanto menor o *overhead* de controle do protocolo, maior é sua eficiência;

- *Overhead de controle*: de modo a manter o controle sobre os membros de um G_{MC}, é necessária a troca de pacotes de controle, o que consome uma parte considerável da banda disponível. Sendo esta um recurso limitado neste tipo de ambiente, o projeto de um protocolo de roteamento deve garantir que o número de pacotes de controle a serem transmitidos para manter o grupo seja o menor possível;
- *Qualidade de Serviço (QoS)*: a provisão de *QoS* em uma rede *ad-hoc* é um fator importante a ser considerado. Os principais parâmetros a serem considerados para provimento de *QoS* são vazão (*throughput*), retardo, *jitter* (variação do retardo) e confiabilidade;
- *Dependência de protocolos de roteamento unicast*: se um determinado protocolo de roteamento *multicast* necessita do suporte de um outro protocolo *unicast*, então é difícil para o protocolo *multicast* trabalhar em ambientes heterogêneos. Por esta razão, é desejável que os protocolos de roteamento *multicast* operem independentemente de outros protocolos, o que nem sempre acontece; e
- *Gerenciamento de recursos*: redes *ad-hoc* consistem de grupos de nós móveis com recursos limitados de energia e memória. Os protocolos de roteamento *multicast*, desta forma, devem minimizar o uso de energia (através da redução do número de transmissões de pacotes) e de memória (através da redução de informações e tabelas).

2.4.2.1. Operação de um Protocolo *Multicast*

Antes de passarmos à classificação dos protocolos de roteamento *multicast* e o detalhamento de alguns protocolos principais, faz-se necessário entender o funcionamento geral de uma transmissão *multicast*.

Baseado no tipo de operação, os protocolos *multicast* para redes sem fio são classificados em dois tipos:

- *Source-Initiated*: neste tipo de protocolo, N_O do G_{MC}, periodicamente, difunde (por *flooding*) um pacote *JoinReq*[2], o qual é propagados pelos demais nós da rede, e atinge, eventualmente, todos os nós pertencentes ao G_{MC}. Estes nós, ao receberem o pacote, expressam seu desejo de receber pacotes para o grupo, respondendo com um pacote de *JoinRep*[3], que se propaga pelo caminho reverso daquele do pacote *JoinReq*. Este pacote de *JoinRep* estabelece *forwarding states*[4] nos N_I (tanto na A_{MC}, quanto na M_{MC}), e, finalmente, atinge o N_O. Não existe um procedimento específico para a manutenção das rotas, já que os protocolos adotam diferentes estratégias com esta finalidade. Alguns protocolos

[2] Pacote de "*Join Request*": pacote encaminhado por um determinado nó, com a finalidade de estabelecer a associação de um determinado nó a um grupo G_{MC}.
[3] Pacote de "*Join Reply*": pacote encaminhado por um nó do G_{MC}, em resposta a um pacote de *JoinReq*.
[4] Refere-se às informações relacionadas ao G_{MC} mantidas pelos nós da A_{MC} ou M_{MC} , que auxiliam cada nó no encaminhamento dos pacotes *multicast* ao próximo salto (*next-hop*) vizinho.

empregam o mesmo procedimento acima descrito, ou seja, periodicamente (a cada período de *refresh*), N_O inicia a transmissão de um pacote de *JoinReq*. A Figura 2.20 (retirada de [Murthy e Manoj 2004]) ilustra o funcionamento de um protocolo *Source-Initiated*.

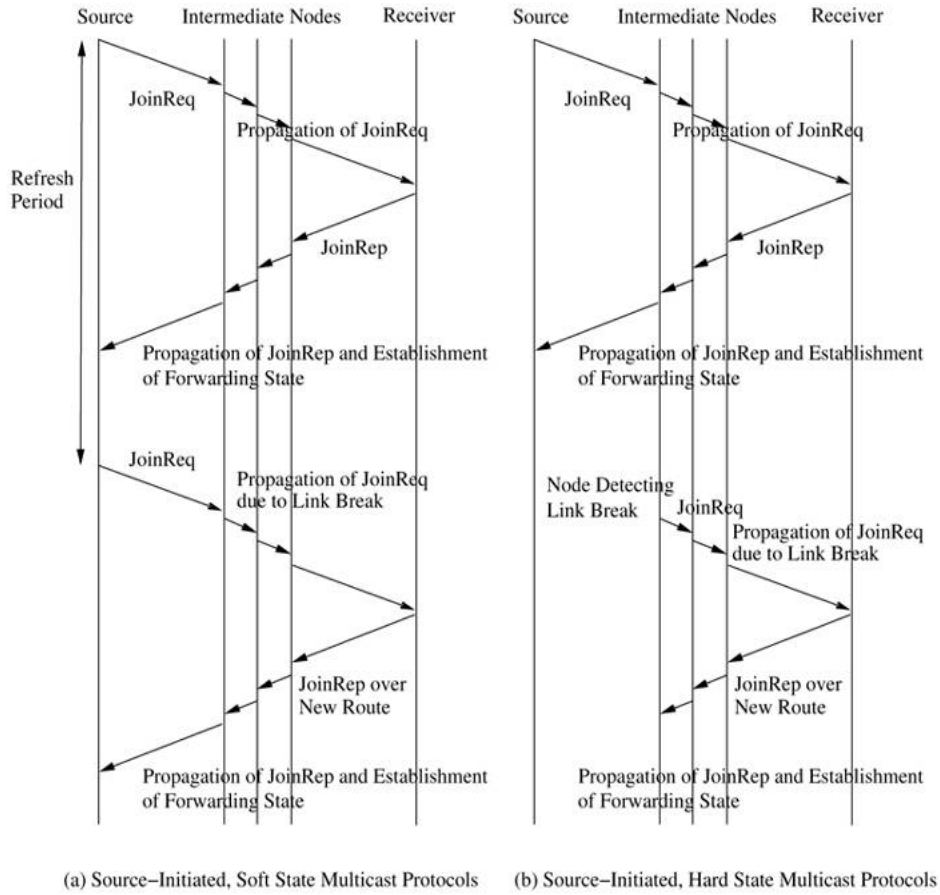

Figura 2.20. Funcionamento de um Protocolo *Multicast Source-Initiated* (Fonte [Murthy e Manoj 2004])

– *Receiver-Initiated*: neste tipo de protocolo, N_D utiliza-se da técnica de *flooding* para procurar pelos N_O dos G_{MC} aos quais pertence. Inicialmente, N_D difunde um pacote de *JoinReq*, que se propaga por outros nós. Usualmente, os N_O dos G_{MC} e/ou os N_I que já pertencem à A_{MC} ou M_{MC} podem responder ao pacote de *JoinReq* com um pacote de *JoinRep*, indicando a possibilidade do nó encaminhar pacotes de dados para o G_{MC}. O N_D escolhe o caminho por aquele nó com o menor valor de *hop-count* (ou outro critério qualquer, dependendo do protocolo) e envia um pacote de *JoinAck*[5] através do caminho reverso àquele tomado pelo pacote de *JoinRep*. Alguns protocolos empregam o mesmo procedimento acima descrito, ou seja, periodicamente (a cada período de *refresh*), N_D inicia a transmissão de um pacote de *JoinReq*. A Figura 2.21 (retirada de [Murthy e Manoj 2004]) ilustra o funcionamento de um protocolo *Receiver-Initiated*.

[5] Pacote de "*Join Acknowledgment*": pacote de confirmação trocado entre dois nós.

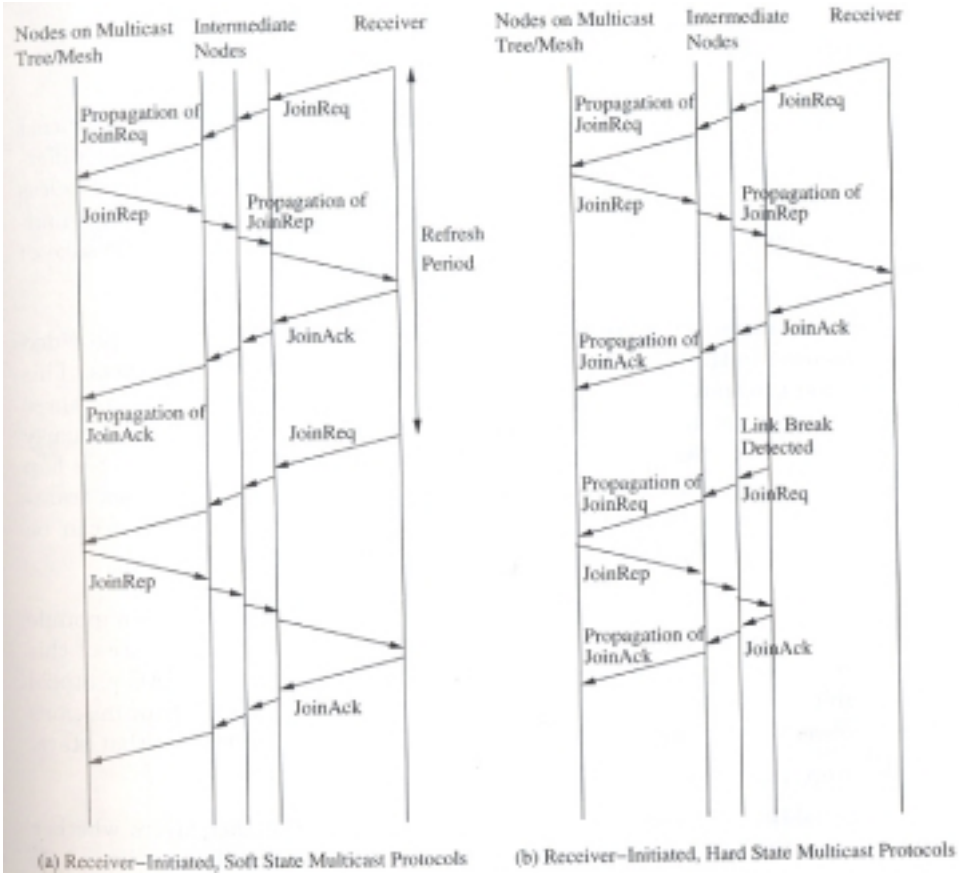

Figura 2.21. Funcionamento de um Protocolo *Multicast Receiver-Initiated* (Fonte [Murthy e Manoj 2004])

2.4.2.2. Classificação dos Protocolos *Multicast*

Uma vez entendido o funcionamento geral de uma transmissão *multicast*, passemos à classificação dos protocolos de roteamento *multicast*. A taxonomia mais referenciada na literatura [Murthy e Manoj 2004], [Cordeiro e Agrawal 2002] para classificação dos protocolos de roteamento *mesh multicast* toma por base características relacionadas à topologia empregada por estes algoritmos. De uma forma geral, os protocolos de roteamento *multicast* são divididos nas seguintes categorias:

- *Tree-based*: existe somente um caminho entre um N_O e um N_D. Estes protocolos apresentam uma altíssima eficiência na vazão dos dados; e
- *Mesh-based*: existem vários caminhos entre um N_O e um N_D. Estes protocolos apresentam uma altíssima robustez devido à disponibilidade de múltiplos caminhos entre os N_O e N_D.

A seguir, serão apresentadas as características dos principais tipos de protocolos empregados em uma rede *mesh* para transmissões *multicast.*

2.4.2.2.1. Protocolos de Roteamento *Tree-Based*

O conceito de *multicasting* do tipo *tree-based* é amplamente utilizado em diversos protocolos *multicast* tradicionais de modo a viabilizar uma maior eficiência no emprego dos recursos de uma rede. Nos protocolos *multicast* do tipo *tree-based*, há apenas um

caminho entre o N_O e o N_D, o que acarreta numa menor robustez, situação esta indesejada em ambientes de alta mobilidade dos nós de uma rede *mesh*.

Os protocolos de roteamento *tree-based* caracterizam-se pela formação de uma Árvore de *Multicast* (A_{MC}), que integra os membros de um grupo e, possivelmente, alguns não-membros. Quando um N_O transmite um pacote de dados, cada nó da árvore recebe o pacote de seu nó *upstream* e o encaminha pelo nó *downstream*.

Os protocolos de roteamento *tree-based* podem ser divididos em dois tipos - *source-tree-based* e *shared-tree-based*. Nos protocolos *multicast source-tree-based*, a árvore é constituída a partir da origem, ao passo que aqueles do tipo *shared-tree-based* compartilham uma mesma árvore por todos os nós pertencentes ao grupo *multicast*, constituída a partir de um nó núcleo denominado *core node* (C_N). Enquanto o primeiro tipo apresenta um melhor desempenho para um alto volume de dados, o segundo apresenta melhores características de escalabilidade, sendo, no entanto, mais vulnerável em função de existir um ponto comum de falhas, o C_N.

A seguir, serão descritos resumidamente os principais protocolos *tree-based* empregados em transmissões do tipo *multicast* em redes *mesh*.

(A) Protocolo MAODV *(Multicast Ad Hoc On-Demand Distance Vector)*

O AODV [Perkins, Belding-Hoyer e Das 2002] é um protocolo *unicast* que vem sendo amplamente testado em redes *ad-hoc*, motivo pelo qual foi estendido de modo a suportar transmissões *multicast* neste tipo de rede. Passou a denominar-se, para este caso, MAODV [Royer e Perkins 1999]. As operações básicas utilizadas pelo protocolo em transmissões *multicast* são similares àquelas empregadas em operações *unicast*, utilizando também mensagens do tipo *RREQ* (*route request*) e *RREP* (*route reply*).

No MAODV, o processo inclui a integração de determinados nós em um grupo *multicast*, e a criação de uma árvore *multicast*. A árvore consiste de um grupo de membros e nós conectados a ela. Isto permite a um *host* integrar-se a um grupo *multicast* mesmo que esteja afastado de mais de um *hop* deste grupo. A operação *unicast* do protocolo também se beneficia da informação coletada durante o processo de descobrimento de rotas para o tráfego *multicast*, o que diminui o tráfego de informações de sinalização pela rede.

Apesar de possuir operações similares àquelas empregadas em transmissões *unicast*, existe uma ligeira modificação. Em operações de roteamento *unicast*, todo N_D possui um único número de seqüência. De maneira análoga, todo grupo *multicast* também possui um único número de seqüência. Desta forma, apenas um líder de grupo é eleito para difundir mensagens periódicas de *group hello* através da rede de modo a manter o citado número de seqüência. O líder do grupo é sempre aquele primeiro nó a integrar o grupo em questão, podendo ser substituído por outro nó, quando vier a deixar o grupo.

De modo a suportar transmissões *multicast*, uma árvore *multicast* é formada *sob demanda* de modo a incluir todos os membros do grupo e alguns não-membros, que são nós retransmissores (*relaying nodes*). O processo de construção da árvore é similar àquele empregado em um procedimento de descobrimento de rotas em protocolos *unicast*: toda vez que um nó precisar se integrar ao *grupo multicast*, ou mesmo

transmitir pacotes de dados para um grupo *multicast*, uma mensagem de *RREQ* é encaminhada através da rede, conforme detalhamento a seguir:

(1) Quando um nó deseja encontrar uma rota para um grupo *multicast*, uma mensagem de *RREQ* é transmitida. O endereço de destino da mensagem é configurado para o endereço do grupo *multicast*. Se o nó deseja integrar-se ao grupo em questão, o *flag "J-Flag"* da mensagem é ativado;

(2) Qualquer nó pode responder à mensagem de *RREQ* simplesmente observando a rota, porém, somente um roteador na árvore de *multicast* em questão pode responder a um pedido de *"join RREQ"*. A rota, eventualmente, pode vir a incluir *hops* através de nós que não pertençam ao grupo *multicast*;

(3) Os nós da árvore de *multicast* deste grupo transmitem de volta uma mensagem de *RREP*. A mensagem *RREP multicast* é ligeiramente diferente daquelas empregadas em protocolos *unicast*. O endereço do líder do grupo *multicast* é armazenado no campo *'group-leader-addr'*. Além disto, existe um campo *'Mgroup-hop'*, que é iniciado em '0' e é incrementado a cada *hop* ao longo da rota. *'Mgroup-hop'* contém a distância em *hops* do N_O ao membro mais próximo da árvore *multicast*; e

(4) Os nós que encaminham as mensagens de *RREQ* e *RREP* gravam o caminho reverso para o N_O do pacote, a exemplo do que é feito no modo *unicast*. Caso múltiplos pacotes de *RREP* sejam recebidos, o N_O escolhe um ramo da árvore de *multicast* e se conecta a ele, evitando, com isto, a formação de *loops*.

(B) Protocolo AMROUTE *(Ad Hoc Multicast Routing Protocol)*

O protocolo AMRoute [Liu, Talpade e McAuley 1999] é outro protocolo de roteamento *multicast* do tipo *tree-based*, dependendo do protocolo de roteamento *unicast* para lidar com a dinâmica de uma rede *mesh*. Como no protocolo MAODV, existe somente um núcleo na árvore *multicast*, que é responsável pela criação da árvore e pela manutenção dos membros do grupo.

O protocolo constrói uma árvore de usuários *multicast* (*user-multicast*), onde são incluídos apenas os membros do G_{MC}. Pelo fato de nós não-membros não serem incluídos na A_{MC}, os enlaces da árvore são virtuais. Em outras palavras, são de fato túneis *multi-hop* IP-em-IP.

A operação *multicast* no protocolo AMRoute consiste de dois passos:

(1) C*riação da Malha*: no começo, um membro do G_{MC} forma o núcleo de sua própria malha (composta apenas por um nó) e inicia, periodicamente, a difusão de mensagens de *JoinReq*. Quando um membro do G_{MC} (que, a exemplo do nó anterior, também consiste do núcleo de sua própria malha) recebe a mensagem de outro núcleo, responde com uma mensagem *JoinAck*, que significa que os dois núcleos se encontraram e se fundiram; elege-se um dos membros como núcleo da nova malha, o C_N, empregando um algoritmo de resolução para o *core*; um túnel bidirecional é construído entre eles simultaneamente. Como resultado, uma malha é formada "a partir do zero", incluindo todos os membros do G_{MC}.

(2) *Criação da Árvore*: o C_N, periodicamente, realiza a difusão de mensagens *TreeCreate* através da malha (pelos túneis). Ao receber esta mensagem, um

membro do G_{MC} escolhe um enlace da malha de onde receberá a mensagem para ser enlace da A_{MC}, ignorando as mensagens duplicadas. Uma mensagem de *TreeCreate-nak* é transmitida de volta pelos enlaces da malha desconsiderados, de modo a eliminá-los da A_{MC}. Dependendo da disponibilidade de banda e da mobilidade da rede, um esquema baseado em confirmações pode ser empregado.

(C) Protocolo AMRIS *(Ad Hoc Multicast Routing protocol utilizing Increasing id-numberS)*

No protocolo AMRIS [Wu e Tay 1999], todo nó recebe um número de identificação – *id-number*. O N_O da sessão *multicast* possui o menor *id-number*. Os demais nós recebem *id-numbers* crescentes à medida que se afastam de N_O.

Para construir uma A_{MC} de entrega, o N_O gera o seu próprio *id-number*, difundindo, então, uma mensagem de *new-session*, que é encaminhada por este pela rede. Durante o trânsito da mensagem pela rede, cada nó escolhe seu próprio *id-number* (maior que aquele contido na mensagem recebida) e reencaminha a mensagem com seu *id-number*. Desta forma, cada nó possui seu *id-number.*

Quando um nó desejar ingressar naquela sessão *multicast*, este escolhe um de seus vizinhos com o menor *id-number* e encaminha uma mensagem de *join request*. Se o vizinho pertencer à árvore (caso a árvore tenha sido construída), responde com uma mensagem de *join ack*, o que representa que o ingresso transcorreu com sucesso; caso contrário (quando esta é a primeira vez que se constrói a árvore), o vizinho encaminha uma mensagem de *join request* aos seus vizinhos e aguarda a resposta, que é repetida até que a mensagem em questão chegue a um dos nós pertencentes à árvore ou ao próprio N_O. Como resultado, uma *árvore de entrega*, enraizada a partir do N_O, é formada de modo a incluir todos os membros do grupo e alguns não-membros.

(D) Protocolo MOLSR (*Multicast Optimized Link State Routing*)

O protocolo MOLSR [Jacquet, Clausen e Laouiti 2001] foi desenvolvido para ambientes de redes móveis e trabalha em redes heterogêneas compostas de roteadores OLSR [Clausen, Jacquet e Laouiti 2002] simples, roteadores MOLSR e *hosts*.

O protocolo MOLSR, desenvolvido como uma extensão do protocolo de roteamento *unicast* OLSR, beneficia-se das informações relacionadas à topologia de rede coletadas por este para construção de suas árvores *multicast*. Neste protocolo, uma árvore *multicast* é construída e mantida de forma distribuída sem qualquer entidade central, provendo rotas de menor caminho entre os nós transmissores de mensagens *multicast* e os diversos membros do grupo *multicast*.

O MOLSR pertence à família de protocolos do tipo *source-based*, e mantém ma Árvore *Multicast* (A_{MC}) para cada transmissor de cada grupo (N_O/G_{MC}). As A_{MC} são compostas apenas por nós com capacidade *multicast*.

Roteadores *multicast* declaram-se para toda a rede através da difusão de uma mensagem *MC_CLAIM* (empregando a técnica de *flooding* otimizada do OLSR). Estes nós disseminam estas informações periodicamente.

Uma vez que um N_O deseje transmitir dados para um G_{MC}, este transmite uma mensagem *SOURCE-CLAIM* (empregando a técnica de *flooding* otimizada do OLSR), possibilitando aos nós daquele G_{MC} detectarem a sua presença e se vincularem à A_{MC}

associada. Os ramos (*branches*) da A_{MC} são construídos de forma reversa: membros do G_{MC} que ainda não conhecem este N_O tentam se vincular a esta A_{MC}. Mais especificamente, quando um membro do G_{MC} recebe uma mensagem *SOURCE-CLAIM* e ainda não pertence a esta árvore (N_O/G_{MC}), este se vincula a esta árvore, conforme descrito a seguir:

(1) Observa em sua T_{ROT} *multicast* o próximo salto para alcançar o N_O (a T_{ROT} *multicast* fornece as rotas mais curtas para todos os nós com capacidade *multicast*). O próximo salto torna-se seu *nó pai* (N_{PAR}) na A_{MC};

(2) Envia uma mensagem *CONFIRM_PARENT* ao seu N_{PAR};

(3) O N_{PAR}, ao receber a mensagem vincula-se à A_{MC}, caso ainda não seja participante desta árvore; e

(4) Esta mensagem é analisada a cada salto pelos roteadores intermediários *multicast* que constroem o ramo da A_{MC}.

(E) Protocolo MZRP (*Multicast Zone Routing Protocol*)

O protocolo MZRP [Devarapalli, Selcuck e Sidhu 2001] é baseado (porém, não dependente), de um mecanismo de roteamento específico. O protocolo toma como base a estrutura hierárquica estabelecida pelo ZRP [Haas 1997], [Haas, Pearlman e Samar 2002], um protocolo de roteamento *unicast* híbrido. Uma rede ZRP é dividida em *Zonas de Roteamento* (Z_R)[6], onde cada nó define a sua própria Z_R (determinado pelo conjunto de nós dentro de um certo raio a partir deste).

Para o roteamento, são empregadas aproximações pró-ativa e reativa, respectivamente, dentro da Z_R (uso de protocolo pró-ativo) e entre zonas (uso de protocolo reativo), de tal forma a combinar as melhores características dos dois tipos de protocolos.

Para criar uma A_{MC} pela rede, o N_O inicia um processo dividido em dois estágios: no primeiro estágio, o N_O tenta formar a A_{MC} dentro da Z_R; no segundo estágio, este estende a árvore a toda a rede:

(1) Inicialmente, N_O encaminha uma mensagem *TREE-CREATE* aos nós localizados dentro da Z_R, utilizando roteamento *unicast* (topologia conhecida dentro de sua Z_R). Os nós interessados em participar deste G_{MC} respondem a N_O com uma mensagem *TREE-CREATE-ACK* , formando a rota; e

(2) De modo a estender a A_{MC} para fora da Z_R, N_O encaminha uma mensagem *TREE-PROPAGATE* a todos os nós de borda de sua zona. Ao receberem a mensagem, os nós de borda encaminham mensagens *TREE-CREATE* a cada um dos nós de suas respectivas Z_R. Desta forma, os nós interessados em participar deste G_{MC} respondem a N_O com mensagens *TREE-CREATE-ACK*.

(F) Protocolo MCEDAR (*Multicast Core-Extraction Distributed Ad Hoc Routing Protocol*)

---

[6] *Zona de Roteamento* é definida por Devarapalli como *"uma aproximação híbrida entre os protocolos de roteamento pró-ativos e reativos, onde o roteamento é pró-ativo dentro das zonas e reativo entre as zonas"*. Ele acrescenta que *"esta aproximação híbrida propicia um balanceamento entre a entrega eficiente de pacotes do roteamento pró-ativo e o baixo custo de manutenção do roteamento reativo"*.

O protocolo MCEDAR [Sinha, Sivakumar e Bharghavan 1999] apresenta uma concepção diferente dentro da família de protocolos do tipo *tree-based*, já que procura aumentar a robustez sem perder em eficiência (característica deste tipo de protocolo). Para isto, a A_{MC} – sobre uma infra-estrutura subjacente em malha denominada *mgraph*, que é empregada para encaminhamento dos pacotes de dados *multicast*. A classificação do MCEDAR mais apropriada é a de um protocolo do tipo *tree-based* sobre *mesh*.

A arquitetura definida pelo CEDAR [Sinha, Sivakumar e Bharghavan 1999] é utilizada pelo protocolo para construção da malha denominada *MDS* (*Minimum Dominating Set*), a qual é constituída por alguns nós na rede, denominados C_N da rede, escolhidos por um algoritmo especial, característica do Protocolo CEDAR. Após a formação da malha MDS (com a escolha dos C_N), cada C_N envia, em seu pacote *broadcast* de *beacon* (através de *piggy-backing*), uma mensagem para informar sobre sua presença aos três próximos *hops*. Este processo auxilia cada C_N na identificação de seus pares mais próximos, permitindo, desta forma, a criação de enlaces virtuais. Os demais nós (não pertencentes à MDS) selecionam um dos C_N como nó dominador**.** Além da criação da MDS, a arquitetura do protocolo CEDAR provê um mecanismo para *broadcast* no *core* baseada em transmissões *unicast* confiáveis, que dinamicamente estabelecem a A_{MC} .

Para participar de um G_{MC}, um determinado N_D solicita ao seu nó dominador que este transmita uma mensagem *JoinReq*, que possui uma opção chamada *JoinID*, que previne a formação de ciclos na *mgraph*) e cujo valor inicial é configurado para infinito. Ao receber a mensagem de *JoinReq*, o C_N de N_O responde com uma mensagem *JoinAck*, caso o *JoinID* seja menor que aquele do nó requisitante. Antes de encaminhar o pacote de *JoinAck*, o nó configura o valor de seu próprio *JoinID* no pacote de *JoinAck*. Ao receber uma mensagem de *JoinAck*, no caminho reverso, um N_I deve decidir se a aceita ou a rejeita baseado em um parâmetro, o fator de robustez (*robustness factor*[7]). Caso o número de pacotes *JoinAck* recebidos pelo N_I seja menor que o mencionado fator, este aceita a mensagem e adiciona o nó de *upstream* da *mgraph* ao seu conjunto de parentes; caso contrário, rejeita a mensagem. Em seguida, este N_I encaminha esta mensagem ao seu nó de *downstream*. E, assim sucessivamente, até chegar ao C_N que encaminhou a mensagem inicial de *JoinReq*. Conseqüentemente, o nó dominador pode receber mais de um pacote *JoinAck* dos membros do G_{MC}. Desta forma, o N_D estará integrado ao G_{MC}.

Apesar da mgraph do G_{MC} ser uma estrutura em malha, o encaminhamento dos pacotes de dados é feito apenas na A_{MC}, devido ao mecanismo de *broadcast* no *core*. De modo a reduzir o *overhead* em uma *mgraph*, uma otimização é realizada desacoplando as infra-estruturas de controle e de encaminhamento de pacotes de dados.

2.4.2.2.2 Protocolos de Roteamento *Mesh-Based*

Este grupo de protocolos de roteamento usa uma Malha *Multicast* (M_{MC}), ao invés de uma A_{MC} compartilhada como nos protocolos do tipo *tree-based*, o que garante enlaces redundantes entre os membros do grupo.

---

[7] *Robustness Factor*: este fator define basicamente o grau de segurança exigido pela *mgraph*, na medida em que define o número de caminhos alternativos máximos possíveis (entre os C_N) a partir de um determinado nó.

Há de ressaltar, no entanto, que estes protocolos consomem mais recursos da rede, quando comparados aos do tipo *tree-based.* No entanto, apresentam uma maior capacidade de se adaptar às características dinâmicas de uma rede *ad-hoc*.

(A) Protocolo ODMRP *(On-Demand Multicast Routing Protocol)*

O protocolo ODMRP [Lee, Gerla e Chiang 1999] é um protocolo de roteamento *multicast* reativo. No protocolo ODMRP, uma M_{MC} é formada por um grupo de nós de encaminhamento (*forwarding nodes*), chamado grupo de encaminhamento, responsável por encaminhar os pacotes de dados entre um N_O e um N_D. Estes nós de encaminhamento mantêm, em *cache*, as mensagens, as quais são usadas para detectar duplicações de pacotes de dados e pacotes de controle de *JoinReq*.

Deve-se ressaltar que o ODMRP não se trata apenas de um protocolo *multicast*, mas também provê capacidade de roteamento *unicast.*

Na fase de inicialização da malha, uma M_{MC} é formada entre os nós fontes e receptores. Para criar uma malha, cada nó fonte do grupo *multicast* encaminha, por *flooding*, pacotes de controle de *JoinReq* periodicamente. Ao receber um pacote de *JoinReq* de um N_O, os N_D podem enviar pacotes *JoinReply* como resposta, utilizando o caminho reverso mais curto. A rota entre um N_O e um N_D é estabelecida após o N_O receber o pacote de *JoinReply*. O pacote de *JoinReply* contém o ID do nó fonte e o ID do próximo nó correspondente (nó de *upstream* através do qual recebeu o pacote de *JoinReq*).

Nesta fase, tentativas são feitas para manter a M_{MC} formada por N_O, N_I (nós de encaminhamento) e N_D. A estrutura em malha garante um certo grau de proteção para que uma sessão *multicast* não seja afetada pela mobilidade dos nós, o que contribui para uma alta taxa de entrega de pacotes.

(B) Protocolo DCMP *(Dynamic Core-Based Multicast Routing Protocol)*

O protocolo DCMP [Das, Murthy e Manoj 2002] é uma tentativa de melhorar a eficiência do protocolo *multicast* do tipo *mesh-based*, já que os protocolos, como o ODMRP, apresentam algumas desvantagens:

- Muitos nós tornam-se *forwarding nodes*, resultando em um número excessivo de retransmissões de pacotes de dados. No ODMRP, por exemplo, todo N_I existente no caminho mais curto entre cada N_O e cada N_D torna-se um *forwarding node*, resultando em muitos nós de encaminhamento[8]; e
- No ODMRP, por exemplo, ocorre de cada N_O transmitir pacotes de *JoinReq* e a rede ser reconstruída periodicamente, acarretando em um alto *overhead* de controle.

O protocolo DCMP busca reduzir o overhead de controle e prover uma melhor razão de entrega de pacotes, através da redução do número de nós de encaminhamento que retransmitem pacotes de *JoinReq*.

[8] A vantagem de uma estrutura em malha contendo diversos nós de encaminhamento, como aquela criada pelo protocolo ODMRP, é, obviamente, a vazão de dados superior e a robustez em condições de mobilidade dos nós.

Na fase de inicialização da malha, o protocolo DCMP tenta reduzir o número de nós de encaminhamento, responsáveis pelo encaminhamento de pacotes de *JoinReq*. No DCMP, existem três tipos de nós: nós passivos, nós ativos e nós *core* ativos[9]. Cada nó passivo é associado a um nó ativo (neste caso, operando como *core* ativo), que desempenha papel de um *proxy* para o nó passivo, e responsabiliza-se pelo encaminhamento dos pacotes de dados dos nós passivos, através da malha criada por seus pacotes de *JoinReq*. Nós passivos não retransmitem pacotes de *JoinReq*, ao contrário dos nós ativos e *core* ativos. Os pacotes de dados dos nós ativos e *core* ativos são transmitidos através da malha por eles mesmos criada, enquanto que um nó passivo encaminha o pacote ao seu nó *core* ativo. Desta maneira, o overhead de controle é reduzido, quando comparado àquele gerado pelo protocolo ODMRP, haja vista que foi reduzido o número de nós que retransmitem pacotes de *JoinReq*.

De modo a manter a robustez de uma rede, é necessário que o protocolo possua parâmetros que garantam a existência de uma quantidade suficiente de nós de encaminhamento. O protocolo DCMP estabelece os seguintes parâmetros:

(1) Número de nós passivos associados ao um nó *core* ativo: o DCMP estabelece um limite para o número de nós passivos associados ao um nó *core* ativo, através do parâmetro *MaxPassSize*; e

(2) Distância máxima de um enlace: de modo a garantir que a razão de entrega de pacotes não seja reduzida em função de um aumento da distância média entre um nó passivo e seu *core*, a distância máxima de um enlace é também limitada pelo parâmetro *MaxHop*.

O protocolo DCMP reduz o número de nós de encaminhamento, se comparado ao ODMRP, sem muitas perdas em termos de robustez e razão de entrega de pacotes.

(C) Protocolo NSMP *(Neighbor Supporting Ad Hoc Multicast Routing Protocol)*

O protocolo NSMP [Lee e Kim 2000] é um protocolo de roteamento do tipo *mesh-based*, que realiza o encaminhamento de pacotes de controle seletivos e localizados. De maneira similar àquela do protocolo ODMRP, para formar uma malha *multicast*, são feitas transmissões de pacotes de controle pela rede. Porém, para propósitos de manutenção, é empregada uma técnica local para o descobrimento de rotas.

Para formar a malha *multicast*, inicialmente, um nó transmissor transmite, por *flooding*, uma mensagem de *flood_req* a todos os nós membros do grupo, e uma mensagem de *reply* é transmitida de volta por estes pelos caminhos reversos, estabelecendo suas rotas até N_O. O N_O, os nós retransmissores (*relaying nodes*) e os N_D são designados como nós de encaminhamento (*forwarding nodes*), formando uma malha *multicast* (*multicast mesh*). Todos os nós adjacentes a um dos nós de encaminhamento são designados como nós vizinhos (*multicast neighbor nodes*). Cada nó da malha *multicast* transmite, periodicamente, um pacote de *local_req*, o qual é retransmitido pelos nós pertencentes à malha *multicast* e seus vizinhos.

[9] Nó *core* ativo: possui características idênticas àquelas de um nó ativo, além de se responsabilizar por um ou mais nós passivos (operando como *proxy* de nó(s) passivo(s)).

Caso um novo nó deseje se integrar ao grupo *multicast*, deve aguardar por um período específico de tempo até que seja recebido um pacote de *local_req* de um nó da malha *multicast* ou de um vizinho. Ao receber a mensagem de *local_req*, o novo nó responde com uma mensagem de *reply*. Caso um novo nó que deseje se integrar à malha *multicast* esteja afastado da referida malha por dois ou mais saltos, e, desta forma, não recebendo qualquer mensagem de *local_req*, este deve transmitir, usando *flooding*, pacotes de controle denominados de *mem_req*. Qualquer nó pertencente à malha *multicast* que receba o referido pacote, transmite, em resposta, um pacote de *reply_route_discovery (RRD)*. Para o caso de serem recebidos múltiplos *RRD* pelo nó que originou a mensagem de *mem_req*, o protocolo NSMP utiliza uma métrica de peso relativo (*relative weight metric*) [Murthy e Manoj 2004], onde o caminho com menor valor relativo é escolhido.

Desta forma, o uso da banda disponível em uma rede empregando o protocolo NSMP é consideravelmente reduzido, quando comparado com o protocolo ODMRP. Para o caso de ocorrerem partições na rede e de novos vizinhos desejarem se integrar à malha, o protocolo NSMP efetua um *flooding* global.

2.4.2.3. Análise Comparativa dos Protocolos de Roteamento *Multicast*

A Tabela 2 apresenta as principais diferenças entre os tipos de protocolos de roteamento *multicast*, tomando por base a taxonomia por *tipo de topologia*, que os classifica em *tree-based* e *mesh-based*.

Tabela 2. Tipos Protocolos de Roteamento *Multicast* (Taxonomia)

	Características	*TREE-BASED*	*MESH-BASED*
1	**Organização** da Rede	Árvore	Malha
2	Definição de **Rotas**	Somente um caminho	Vários caminhos
3	Uso dos **recursos** da rede	Eficiente	Desperdício
4	Como lida com **Mobilidade**	Não possui caminhos alternativos	Caminhos alternativos
5	***Overhead***	Baixo	Alto

A Tabela 3 apresenta análise comparativa sobre aspectos essenciais dos protocolos *multicast* apresentados, de modo a melhor caracterizá-los (incluindo vantagens e desvantagens), notadamente quanto às suas respectivas aplicações em ambientes de redes *mesh*.

Tabela 3. Análise comparativa dos Protocolos *Multicast*

#	Protocolos	Análise Comparativa – Vantagens e Desvantagens	
		Vantagens	**Desvantagens**
Tree-	**MAODV**	- Integração do *unicast* e do *multicast* em uma mesma estrutura; e - Informações obtidas durante a fase de descobrimento de rotas *unicast* podem ser usadas pelo *multicast* e vice-versa => Redução do *overhead* de controle.	- A topologia baseada em árvore resulta em congestionamentos ao longo dos ramos da árvore; e - Uma falha do líder do grupo pode afetar significativamente todas as sessões em andamento.
	AMRoute	- Robustez do protocolo devido à sua estrutura em árvore.	- Eficiência do protocolo é reduzida pela possibilidade de existirem *loops* na estrutura da árvore; - Uma falha do *CN* pode levar a perda de pacotes e aumentar o retardo.

Based	**AMRIS**	- Menor *overhead* de controle; e - Simplicidade (quando comparado a outros protocolos *multicast*).	- Desperdício da banda disponível, devido ao uso de *beacons* e a perda de inúmeros pacotes em função da colisão com os referidos *beacons.*
	MOLSR	- Capacidade de operar em ambientes heterogêneos (roteadores OLSR e MOLSR); - Sendo uma extensão do OLSR, beneficia-se das informações sobre a topologia de rede recolhida pelo OLSR.	- Dependência do protocolo *unicast* OLSR; e - Menor robustez (quando comparado aos protocolos em malha).
	MZRP	- *Overhead* de controle reduzido; - Protocolos *unicast* e *multicast* podem trocar informações; e - Independência do protocolo *unicast* ZRP.	- Nós muito afastados do N_O esperam um tempo maior até serem incorporados à sessão *multicast*, uma vez que as mensagens *Tree-Propagate* levam muito tempo em seu encaminhamento.
	MCEDAR	- Devido à estrutura em malha *(mgraph)*, o protocolo é mais robusto que os demais protocolos do tipo *tree-based*; e - O uso de um protocolo *tree-based* sobre uma malha para o encaminhamento dos pacotes toma-o tão eficiente quanto os demais protocolos do tipo *tree-based*.	- Protocolo com alto grau de complexidade (quando comparado aos demais protocolos *tree-based*); e - Em redes dotadas de nós com alta mobilidade, os nós freqüentemente mudam seus C_N, aumentando, desta forma consideravelmente o *overhead* de controle.
Mesh-Based	**ODMRP**	- Robustez do protocolo devido ao processo de manutenção inerente para manter a malha.	- Pacotes entre N_O e N_D podem seguir por múltiplas rotas, resultando numa menor eficiência no uso da banda disponível.
	DCMP	- *Overhead* de controle reduzido; - Alta escalabilidade; e - Alta taxa de entrega de pacotes.	- Parâmetros do protocolo - *MassPassSize e MaxHop* - dependentes das condições de carga da rede e do número de nós.
	NSMP	- Os processos de descobrimento de rotas e de manutenção locais garantem uma redução do *overhead* de controle, e mantêm um alto *throughput* pela rede; e - Política de admissão de novos membros na malha *multicast* é mais eficiente que nos demais protocolos.	- Na determinação de uma rota para um determinado nó, a métrica de peso relativo leva em consideração apenas o número de saltos até o nó da malha *multicast*, não considerando, portanto, as diferentes condições de carga da rede.

2.4.3. Panorama Atual sobre Comunicação em Grupo em Redes *Mesh*

Um novo padrão vem sendo desenvolvido pelo IEEE, denominado IEEE 802.11s [IEEE 2006], que será capaz de dar suporte a transmissão através de múltiplos saltos no nível de enlace para redes *ad-hoc* sem fio, oferecendo comunicação em grupo. O padrão especifica um novo protocolo de roteamento default chamado HWMP (*Hybrid Wireless Mesh Protocol*), baseado no AODV [Perkins 2003], que deverá ser implementado em todas as interfaces 802.11s e ainda um protocolo de roteamento opcional, chamado RA-OLSR (*Radio Aware* OLSR), baseado no OLSR [Clausen 2003].

O projeto OLPC (*One Laptop per Child*) (laptop.org), sendo desenvolvido pelo MIT, que desenvolve um laptop de baixo custo voltado para crianças já está implementando a proposta atual do IEEE 802.11s em sua interface de rede. A transmissão usando redes *mesh* nos laptops OLPC já está operacional e a comunicação em grupo está em fase de implementação. No Brasil, a UFF – Universidade Federal

Fluminense – coordena os testes de conectividade do laptops OLPC no projeto RUCA – Rede do Projeto Um Computador por Aluno [RUCA 2007].

Enquanto o novo padrão para redes *mesh* ainda está sendo especificado, grupos de pesquisa nas várias universidades ao redor do mundo realizam pesquisas sobre comunicação em grupo em redes *mesh*.

A UCSB (University of California at Santa Barbara) realiza experimentos práticos com o protocolo *unicast* AODV, em sua plataforma de testes localizada no campus da universidade, formada por 25 (vinte e cinco) nós IEEE 802.11b. A UCSB desenvolveu um protocolo ligeiramente modificado em relação ao AODV, tendo sido chamado de AODV-UCSB [Chakeres 2007]. O protocolo encontra-se em testes em operações do tipo *unicast* e também suporta *gatewaying* para Internet, múltiplas interfaces e as funções básicas de *multicast*. Outras experiências envolvendo implementações práticas do AODV incluem o desenvolvimento do AODV-UU [Lundgren et al 2002] e do AODV-UIUC [Kawadia et al 2003]. As três diferentes versões do AODV suportam *multicast*.

Voltando ao cenário brasileiro, podemos citar o projeto *ReMesh* [ReMesh 2007] da Universidade Federal Fluminense (UFF), sendo desenvolvido atualmente com parcerias da RNP, UFPA, UTFPR e PUC-PR. O projeto *ReMesh* possui como proposta a implantação de uma rede de acesso do tipo *mesh* para usuários universitários que residem nas proximidades de suas universidades. Foi desenvolvido o *firmware* de um roteador *mesh* sem fio baseado numa implementação compacta do linux, chamada *OpenWRT* [OpenWRT 2007]. O projeto utiliza uma extensão do protocolo de roteamento OLSR, chamada OLSR-ML [Passos et al 2006], e pretende utilizar o protocolo MOLSR para oferecer comunicação em grupo em seus roteadores *mesh*.

2.5. Conclusões

Este capítulo apresentou uma discussão sobre três tópicos de pesquisa atuais em redes *mesh*, apresentando soluções para suporte à mobilidade de estações clientes, técnicas para provisão de qualidade de serviço e ainda protocolos de roteamento *multicast* que oferecem comunicação em grupo em redes *mesh*.

Os tópicos abordados são fundamentais para tornar o uso de redes *mesh* adequado a aplicações multimídia, tais como voz sobre IP e videoconferência, que exigem que a infra-estrutura de rede forneça tratamento diferenciado a fluxos de mídia contínua (voz e vídeo) e também permita a comunicação entre grupos de usuários evitando o desperdício de recursos disponíveis na rede. O suporte à mobilidade é fundamental a todas as aplicações em rede, permitindo que usuários utilizem equipamentos móveis tais como *laptops*, celulares e *palms* como estações clientes.

Pelo fato de não existir um padrão único para a construção de redes *mesh* e sim várias opções incluindo soluções acadêmicas e comerciais, são inúmeras as propostas encontradas na literatura tentando solucionar as questões apresentadas. Mesmo após a especificação do futuro padrão IEEE 802.11s que permitirá a transmissão em múltiplos saltos no nível de enlace, resolvendo os problemas de mobilidade e comunicação em grupo, as soluções propostas no nível de rede continuarão sendo investigadas, pois são

necessárias para permitir o uso de equipamentos já existentes nas redes *mesh* atuais e futuras.

2.6. Referências Bibliográficas

Aguayo, Daniel; Bicket, John; Biswas, Sanjit; Judd, G.; Morris, Robert (2004) "A Measurement Study of a Rooftop 802.11b Mesh Network", In Proceedings of ACM SIGCOMM Conference (SIGCOMM 2004), Setembro.

Alam, M.; Hamid, A.; Choong Seon Hong (2006) "QoS-aware fair scheduling in multihop wireless ad hoc networks". International Conference Advanced Communication Technology, Fevereiro.

Alicherry, M.; Bhatia, R.; Li, L.E. (2006) "Joint Channel Assignment and Routing for Throughput Optimization in Multiradio Wireless Mesh Networks". IEEE Journal on Selected Areas in Communications, Novembro.

Aoun, B.; Boutaba, R.; Iraqi, Y.; Kenward, G. (2006) "Gateway Placement Optimization in wireless mesh networks with qos constratints". IEEE Journal of Selected Areas in Communications, Novembro.

Badis, H. and Al Agha, K. (2004) "An Efficient QOLSR Extension Protocol for QoS in Ad Hoc Networks", IEEE Vehicular Technology Conference, Setembro.

Badis, H. and Al Agha, K. (2004) "QOLSR multi-path routing for mobile ad hoc networks based on multiple metrics: bandwidth and delay". IEEE Vehicular Technology Conference, Maio.

Badis, H.; Gawedzki, I.; Al Agha, K. (2004) "QoS routing in ad hoc networks using QOLSR with no need of explicit reservation". IEEE Vehicular Technology Conference, Setembro.

Balakrishnan, H., Seshan, S., Amir, E., Katz, R. (1995) "Improving TCP/IP Performance over Wireless Networks". In: Proceedings of the 1st ACM International Conference On Mobile Computing and Networking (Mobicom), Novembro.

Baoxian Z. and Mouftah, H. (2005) "QoS Routing for Wireless Ad Hoc Networks: Problems, Algorithms and Protocols". IEEE Communications Magazine, Outubro.

Barua, G.; Chakraborty, I. (2002) "Adaptive routing for ad hoc wireless networks providing QoS guarantees". IEEE International Conference on Personal Wireless Communications, Dezembro.

Bechler, M. ,W.J. Franz, L. Wolf (2003) "Fleet net as a IPv6 subnet with global fleet net prefix". Mobile Internet Access in FleetNet.

Bin Ni; Santhapuri, N.; Zifei Zhong; Nelakuditi, S. (2006) "Routing with opportunistically coded exchanges in wireless mesh networks". IEEE Workshop on Wireless Mesh Networks.

Bok-Nyong Park; Wonjun Lee; Sanghyun Ahn; Sungjoon Ahn (2006) "QoS-driven wireless broadband home networking based on multihop wireless mesh networks". IEEE Transactions on Consumer Electronics, November.

Bondareva, O. and Baumann, R., (2006) "Handling Addressing and Mobility in Hybrid Wireless Access Networks", TIK Report 250, ETH, Zürich.

Brown K., Singh, S. (1997) “M-TCP: TCP for mobile cellular networks”, ACM Computer Communications Review 27 p.19–43, Outubro.

Buddhikot, M., Hari, A., Singh, K., Miller, S. (2005) “MobileNAT: A Net Technique for Mobility Across Heterogeneous Address Spaces”. In: Journal of Mobile Networks and Applications,Volume 10, Number 3, Junho.

Carlson, E.; Prehofer, C.; Bettstetter, C.; Karl, H.; Wolisz, A. (2006) “A Distributed End-to-End Reservation Protocol for IEEE 802.11-Based Wireless Mesh Networks”. IEEE Journal on Selected Areas in Communications, Novembro.

Chakeres, I. D. (2007) “AODV-USCB Implementation from University of California Santa Barbara”, Santa Barbara, URL: http://moment.cs.ucsb.edu/AODV/aodv.html.

Chakrabarti, S. and Mishra, A. (2001) “QoS Issues in Ad Hoc Wireless Networks”. IEEE Communications Magazine, Fevereiro.

Chen, T. W., Gerla, M. (1998) “Global State Routing (GSR): A New Routing Scheme for Ad Hoc Wireless Networks”, In Proceedings of IEEE ICC 1998, pp. 171-175, Junho.

Chiang, C. (1997) “Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel”. In Proceedings of IEEE SICON 1997, pp. 197-211, Abril.

Clausen, T. and Jacquet, P. (2003), “Optimized Link State Routing Protocol (OLSR)”, IETF RFC 3626, Outubro.

Cordeiro, Carlos de Morais, Agrawal, Dharma P. (2002) “Mobile Ad Hoc Networking”; Livro texto dos minicursos do Simpósio Brasileiro de Redes de Computadores. Búzios, RJ.

Crawley, E., Nair, R., Rajagopalan,B., Sandick (1998) “A Framework for QoS-based Routing in the Internet”, RFC 2386, Agosto.

Das, S. K., Murthy, C. Siva Ram, Manoj, B. S. (2002) “A Dynamic Core-Based Multicast Routing Protocol for Ad Hoc Wireless Networks”, In Proceedings of ACM MOBIHOC 2002, pp. 24-35, Junho.

Devarapalli, V., Selcuk, A. A., Sidhu, D. (2001) “MZR: A Multicast Protocol for Mobile Ad Hoc Networks”, draft-vijay-manet-mzr-01.txt, Julho.

Draves, Padhye, Zill (2004), “Routing in Multi-radio, Multi-hop Wireless Mesh Networks”, ACM MobiCom, Philadelphia, PA, setembro. Disponível em: http://research.microsoft.com/mesh/.

Faccin, S., Wijting, C., Kneckt, J. e Damle, A. (2006) “Mesh WLAN Networks: Concept and System Design”. IEEE Wireless Communications. Abril.

Farkas, K.; Budke, D.; Plattner, B.; Wellnitz, O.; Wolf, L. (2006) “QoS Extensions to Mobile Ad Hoc Routing Supporting Real-Time Applications”. IEEE International Conference on Computer Systems and Applications, Março.

Ferguson, P., Senie, D. (1998) “Network ingress filtering:defeating denial of service attacks which employ IP source address spoofing”. IETF RFC 2267, Janeiro.

Futernik, A.; Haimovich, A.M.; Papavassiliou, S. (2003) “An analytical model for measuring QoS in ad-hoc wireless networks”. IEEE Global Telecommunications Conference, Dezembro.

Gavrilovska, L.M.; Atanasovski, V.M. (2005) “Ad hoc networking towards 4G: challenges and QoS solutions”. International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services, Setembro.

Gerla, Mario, Hong, Xiaoyan, Pei, Guangyu (2002) “Fisheye State Routing Protocol (FSR) for Ad Hoc Networks”, draft-ietf-manet-fsr-03.txt, Junho.

Google Mesh Network (2007), URL: http://www.google.com.

Haas, Z. J. (1997) “Algorithm for the Reconfigurable Wireless Network”, In Proceedings of ICUPC 1997, vol. 2, pp. 1415-1425, Outubro.

Haas, Z. J., Pearlman, M. R., Samar, P. (2002) “The Zone Routing Protocol (ZRP) for Ad Hoc Networks”, draft-ietf-manet-zrp-04.txt, Julho.

Handley, M., Schulzrinne, H., Schooler, E., Rosenberg, J. (1999) “SIP: session initiation protocol,” Request for Comments (Proposed Standard) 2543, Internet Engineering Task Force, Março.

Ho, Krishna Ramachandran, Kevin C. Almeroth and Elizabeth M. Belding-Royer (2004), “A Scalable Framework for Wireless Network Monitoring”, 2nd ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots (WMASH), Philadelphia, PA, setembro. Disponível em: http://moment.cs.ucsb.edu/meshnet/.

Hsu, Chih-Shun; Sheu, Jang-Ping; Tung, Shen-Chien (2006) “An On-Demand Bandwidth Reservation QoS Routing Protocol for Mobile Ad Hoc Networks”. IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing.

Huang, J.-H.; Wang, L.-C.; Chang, C.-J. (2006) “Capacity and QoS for a scalable ring-based wireless mesh network”. IEEE Journal of Selected Areas in Communications, Novembro.

IEEE P802.11s™/D0.02 (2006), “Draft Amendment to Standard for Information Technology - Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: Amendment: ESS Mesh Networking”, IEEE, Junho.

Jacquet, Philippe, Clausen, Thomas, Laouiti, Anis, et al. (2001) “Multicast Optimized Link State Routing Protocol (MOLSR)”, draft-ietf-manet-olsr-molsr-01.txt, Novembro.

Jaseemuddin, M.; Esmailpour, A.; Alwan, A.; Bazan, O. (2006) “Integrated Routing System for Wireless Mesh Networks”. Canadian Conference on Electrical and Computer Engineering, Maio.

Jiang, H., Zhuang, W., Shen, X., Abdrabou, A.; Wang, P. (2006) “Differentiated services for wireless mesh backbone”. IEEE Communications Magazine. Julho.

Johnson, David B., Maltz, David A., HU, Uih-Chun, JETCHEVA, Jorjeta G. (2002) “The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)”, draft-ietf-manet-dsr-07.txt, Fevereiro.

Jönsson, U., Alriksson, F., Larsson, T., Johansson P., Maguire Jr., G.Q. (2000) “MIPMANET - Mobile IP for Mobile Ad hoc Networks”. In MobiHoc, Agosto.

Kawadia, V., Zhang, Y., Gupta, B. (2003) “System Services for Implementing Ad-Hoc Routing: Architecture, Implementation and Experiences”, In Proceedings of 1st International Conference on Mobile Systems, Applications, and Services (MobiSys, pp. 99-112, San Francisco, CA, Junho.

Koksal, C.E.; Balakrishnan, H. (2006) “Quality-Aware Routing Metrics for Time-Varying Wireless Mesh Networks”. IEEE Journal on Selected Areas in Communications, Novembro.

Koponen, T., Gurtov, A., Nikander, P. (2005) “Application Mobility with HIP”. In: Proceedings of ICT'05, Maio.

Kravets, Robin; Carter, Casey; Magalhães, Luiz (2001). “A Cooperative Approach to User Mobility”. ACM Computer Communication Review, USA, v. 31, n. 5, 2001.

Krishnaswamy, D.; Hsien-Po Shiang; Vicente, J.; Conner, W.S.; Rungta, S.; Chan, W.; Kai Miao (2006) “A cross-layer cross-overlay architecture for proactive adaptive processing in mesh networks”. IEEE Workshop on Wireless Mesh Networks.

Kuo, Cheng-Ting; Liang, Chiu-Kuo (2006) “A Meshed Multipath Routing Protocol in Mobile Ad Hoc Networks”. International Conference on Parallel and Distributed Computing, Applications and Technologies, Dezembro.

Lee, S., Gerla M., Chiang, C-C. (1999) “On-Demand Multicast Routing Protocol (ODMRP)”, In Proceedings of IEEE WCNC'99, New Orleans, LA, Setembro.

Lee, S., Kim, C. (2000) “Neighbor Supporting Ad hoc Multicast Routing Protocol (NSMP)”, In Proceedings of 1st Annual Workshop on Mobile Ad Hoc Network & Computing, MobiHOC 2000, pp. 37-50, Boston, Agosto.

Lianggui L. and Guangzeng F. (2005) “Mean Field Network based Qos routing scheme in wireless mesh networks”. Proceedings in International Conference on Wireless Communications, Networking and Mobile Computing, Setembro.

Lin, C. R. and Liu, J. (1999) “QoS routing in ad hoc wireless networks”. IEEE Journal on Selected Areas in Communications, Agosto.

Liu, Mingyan, Talpade, Rajesh R., McAuley, Anthony (1999) “AMRoute: Ad hoc Multicast Routing Protocol”, Technical Report CSHCN TR 99-1, University of Maryland.

Lundgren, H., Lundberg D., Nielsen, J., Nordström, E., Tschudin, C. F. (2002) “A Large-scale Testbed for Reproductible Ad Hoc Protocol Evaluations”, In IEEE Wireless Communications and Networking Conference 2002 (WCNC), Março.

Magalhães, Luiz ; Kravets, Robin. (2001)”End-to-End Inverse Multiplexing for Mobile Hosts” Journal of the Brazilian Computer Society, Rio de Janeiro, v. 7, n. 2, p. 52-62

Magalhães, L. (2002) "A Transport Layer Approach to Host Mobility", Ph.D. Thesis, University of Illinois at Urbana Champaign, EUA.

Montenegro, G. (1998) “Reverse Tunneling for Mobile IP”. IETF RFC 2344, Maio.

Murthy, S., Garcia-Luna-Aceves, J. J. (1996) “An efficient Routing Protocol for Wireless Networks, Special Issue on Routing in Mobile Communication Networks”, pp183-197. Outubro.

Murthy, C. Siva Ram, Manoj B. S. “Ad Hoc Wireless Networks: Architectures and Protocols”. 2. ed. New Jersey: Prentice Hall, 2004.

Nandiraju, N.S.; Nandiraju, D.S.; Agrawal, D.P. (2006) “Multipath Routing in Wireless Mesh Networks”. IEEE International Conference on Mobile Adhoc and Sensor Sysetems, Outubro.

Nguyen, Dang-Quan; Minet, P. (2005) “Interference-aware QoS OLSR for mobile ad-hoc network routing”. International Workshop on Self-Assembling Wireless Networks, Maio.

OpenWRT Project (2007), URL: http://www.openwrt.org/.

Park, V. D., Corson, M. S. (1997) “A Highly Adaptative Distributed Routing Algorithm for Mobile Wireless Networks, In Proceedings of IEEE INFOCOM’97”, pp. 1405-1413, Abril.

Passos, Diego; Douglas Vidal Teixeira, Débora C. Muchaluat-Saade, Luiz C. Schara Magalhães e Célio Albuquerque (2006), “Mesh Network Performance Measurements”, 5th International Information and Telecommunicatios Technologies Symposium, Cuiabá, MT, Brasil, 6 a 8 de dezembro.

Perkins, C. (1996) “Mobile-IP, Ad-Hoc Networking, and Nomadicity”. In: Proceedings of 20th International Computer Software and Applications Conference. (Compsac 96), IEEE CS Press, pp. 472-476.

Perkins, C. (2002) “IP Mobility Support for IPv4”, RFC 3220, IETF, Janeiro.

Perkins, C., Belding, E., Sun, Y. (2002) “Internet connectivity for ad-hoc mobile networks”. In: International Journal of Wireless Information Networks, Abril.

Perkins, C., Gustafsson, E., Jonsson, A. (2003) “Mobile IPv4 regional registration”. Work in progress—Internet Draft, Novembro.

Perkins, C., Johnson, D., Arkko, J. (2000) “Mobility Support in IPv6”, IETF RFC3775, Junho.

Perkins, C., Johnson, D.B. (1996) “Mobility support in IPv6”, Proceedings of the 2nd annual international conference on Mobile computing and networking, p.27-37, Rye, New York, United States, Novembro.

Perkins, C., Lei, H. (1997) “Ad Hoc Networking with Mobile IP”. In: Proceedings 2nd European Personal Mobile Communications Conf. (EPMCC 97), pp. 197-202.

Perkins, C., Malinen, J.T., Wakikawa, R., Belding-Royer, E.M., Sun, Y. (2001) “IP Address Autoconfiguration for Ad hoc networks”. In: IETF Internet Draft, Novembro.

Perkins, Charles E., Belding-Hoyer, Elizabeth M., Das, Samir R. (2003) “Ad Hoc On-Demand Distance Vector (AODV) Routing”, IETF RFC 3561, Julho.

Perkins, Charles E., Bhaqwat, Pravin (1994) “Highly dynamic Destination-Sequenced Distance Vector routing (DSDV) for mobile computers”, In Proceedings of SIGCOM´94 Conference on Communications Architecture, Protocols and Applications, pp. 234-244, Agosto.

Petrovic, M., Aboelaze, M. (2003) “Performance of TCP/UDP over ad hoc IEEE 802.11”, In: International Conference on Telecommunications, pp. 700–708.

Ramjee, R., Varadhan, K.,Salgarelli, L., Thuel, S.R., Wang, S., Porta T. (1999) “HAWAII: A Domain-Based Approach for Supporting Mobility in Wide-Area Wireless Networks”, Proc. 7th Ann. International Conference Network Protocols (ICNP 99), IEEE CS Press, pp. 283-292.

Ratnam, K., Matta, I. (1998) “WTCP: An Efficient Mechanism for Improving TCP Performance over Wireless Links”. In: Proceedings of the Third IEEE Symposium on Computer and Communications (ISCC ’98), Junho.

ReMesh (2007), Projeto ReMesh, URL: http://mesh.ic.uff.br.

Royer, E. M., Perkins, C. E. (1999) “Multicast Operation of the Ad hoc On-Demand Distance Vector Routing Protocol (MAODV)”, In Proceedings of IEEE MOBICOM’99, pp. 207-218, Seattle, WA, Agosto.

Rozner, E.; Seshadri, J.; Mebta, Y.; Lili Qiu (2006) “Simple opportunistic routing protocol for wireless mesh networks”. IEEE Workshop on Wireless Mesh Networks.

RUCA (2007), Projeto RUCA, URL: hhtp://www.ruca.midiacom.uff.br.

Saltzer, J.H., Reed, D.P., Clark, D.D. (1984) “End-to-end arguments in system design”. ACM Transactions on Computer Systems (TOCS), vol. 2(4), p.277-288, Novembro.

Sinha, P., Sivakumar, R., Bharghavan (1999) “CEDAR: A Core Extraction Distributed Ad Hoc Routing Algorithm”, IEEE Journal on Selected Areas in Communications, vol. 17, nº 8, pp. 1454-1466, Agosto.

Sinha, P., Sivakumar, R., Bharghavan, V. (1999) “MCEDAR: Multicast Core Extraction Distributed Ad Hoc Routing”, In Proceedings of IEEE WCNC 1999, pp. 1313-1317, Setembro.

Snoeren, A., Balakrishnan, H., (2000) “An End-to-End Approach to Host Mobility” 6th ACM MOBICOM, Boston, Massachusetts, August.

Song W.; Fang, X. (2006) “Routing with Congestion Control and Load Balancing in Wireless Mesh Networks”. International Conference on Telecommunications Proceedings, Junho.

Srisuresh, P., Egevang, K. (2001) “Traditional IP Network Address Translator (Traditional NAT)”. RFC 3022.

Su, G., Nieh, J. (2002) “Mobile communication with virtual network address translation”. Department of Computer Science, Columbia University, CUCS-003-2, Fevereiro.

Subramanian, A.P.; Buddhikot, M.M.; Miller, S. (2006) “Interference aware routing in multi-radio wireless mesh networks”. IEEE Workshop on Wireless Mesh Networks.

Toh, C. K. (1997) “Associativity-Based Routing (ABR) for Ad Hoc Mobile Networks, Wireless Personal Communications”, vol. 4, no. 2, pp.1-36, Março.

Tsaoussidis, V., Matta, I. (2002) “Open Issues on TCP for Mobile Computing”. In: Journal of Wireless Communications and Mobile Computing – Special Issue on Reliable Transport Protocols for Mobile Computing, 2(1), Fevereiro.

Tsarmpopoulos, N., Kalavros, I., Lalis, S. (2005) “A Low-Cost and Simple-to-Deploy Peer-to-Peer Wireless Network base don Open Source Linux Routers”, In Proceedings of the 1st International Conference on Testbeds and Research Infrastructures for the Development of NeTworks and COMmunities (TRIDENTCOM’05).

Vaidya, N. (2002) “Weak Duplicate Address Detection in Mobile Ad Hoc Networks”. In: MobiHoc ’02, Lausanne , Suiça, Junho.

Valkó, A.G. (1999) “Cellular IP: A New Approach to Internet Host Mobility”. In: ACM Computer Communication Rev., vol. 29, no. 1, pp. 50-65.

Wedlund, E., Schulzrinne, H. (1999) “Mobility Suport using SIP”. In: IEEE/ACM Multimedia conference WOWMOM.

Wu, C. W., Tay, Y. C. (1999) “AMRIS: AMR with Increasing Sequence Numbers: a Multicast Protocols for Ad Hoc Wireless Networks”, In Proceedings of IEEE MILCOM’99, Atlantic City, Novembro.

Xylomenos, G., Polyzos, G.C., Mahonen, P., Saaranen, M. (2001) “TCP performance issues over wireless links”, IEEE Communications Magazine 39 p.52–58.

Yan Zhang; Mingtuo Zhou; Shaoqiu Xiao; Masayuki Fujise (2006) “An Effective QoS Scheme in WiMAX Mesh Networking for Maritime ITS”. International Conference on ITS Telecommunications Proceedings, Junho.

Ying, Z.; Ananda, A.L.; Jacob, L. (2003) “A QoS enabled MAC protocol for multi-hop ad hoc wireless networks”. Proceedings of the 2003 IEEE International Performance, Computing, and Communications Conference, Abril.

Zhao, R. Shi, X.Z. Yang, (2004) “One Hop-DAD Based Address Auto-configuration in MANET”. The Fifth International Conference on Parallel and Distributed Computing, Applications and Technologies.

Zhao, R.; Walke, B.; Hiertz, G.R. (2006) “An Efficient IEEE 802.11 ESS Mesh Network Supporting Quality-of-Service”. IEEE Journal on Selected Areas in Communications, Novembro.