UNIVERSIDADE
FEDERAL DO CEARÁ

Centro de Ciências

Programa de Mestrado e Doutorado em Ciência da Computação (MDCC)

Um Mecanismo de Melhoria de Handovers Verticais Utilizando Endereçamento Multicast e Serviços do MIH 802.21

Dissertação de Mestrado

Michel Sales Bonfim

Orientador: Prof. Miguel Franklin de Castro, PhD.

Fortaleza - CE

Setembro - 2011

Michel Sales Bonfim

Um Mecanismo de Melhoria de Handovers Verticais Utilizando Endereçamento Multicast e Serviços do MIH 802.21

Dissertação de Mestrado submetida ao Programa de Mestrado e Doutorado em Ciência da Computação (MDCC) da Universidade Federal do Ceará como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Miguel Franklin de Castro, PhD.

Fortaleza - CE

Setembro - 2011

Dedico esta dissertação
à minha família.

Agradecimentos

Durante este trabalho, descobri que, quando acreditamos em nós mesmos podemos alcançar qualquer objetivo traçado em nossas vidas. Entretanto, acreditar não é tudo, também existem as pessoas que surgem em nossa vida e a transforma, de forma a fazer com que o caminho para esses objetivos se torne bem menos difícil. Deste modo, não posso deixar de reconhecê-los e agradecê-los.

Agradeço à minha mãe, pela formação que me permitiu ter, pela força, por sempre acreditar em mim e no meu potencial de transformar escolhas em frutos. Agradeço também pelos os sacrifícios que só ela sabe quais foram e que me fizeram chegar até onde estou hoje. Também agradeço aos meus irmãos, Emanuel e Ligia, a Tia Sônia e a minha querida sobrinha, Jéssica, pela ajuda financeira e por sempre fortalecerem e engrandecerem o meu espirito com os seus conselhos cheios de positividade.

Agradeço à minha esposa e o meu grande amor, Sarah, por fazer parte da minha vida, tornando-me uma pessoa muito melhor. Obrigado meu amor, pela sua paciência, pela inspiração e pelo incentivo durante toda nossa, ainda curta, vida de casados. Também agradeço à sua família, Gilberto, Fátima e Lia, que me receberam como um membro da família, sempre me incentivando a seguir em frente na busca de minhas metas.

Agradeço em especial ao meu amigo e orientador, Prof. Miguel Franklin de Castro, pela oportunidade, por acreditar no meu potencial, pelas orientações, pela motivação e pela honra de poder ser ele a pessoa que me tornou apto a concretizar o meu sonho de alcançar o grau de Mestre.

Agradeço aos meus amigos e mentores, Prof. Stênio Fernandes e Prof. Ahmed Karmouch, pela grande ajuda durante a minha estada de 6 meses no Canadá e pelo grande impulso que deram ao meu trabalho. Também não posso deixar de agradecer a Karla Matias por ceder uma moradia neste país e dividir o convívio com o mundo fora dos muros da universidade.

Agradeço aos meus amigos do GREat, pelos importantes conselhos e por muitas vezes abdicarem do seu trabalho para me auxiliar. Obrigado Bruno, Camila, Giovanni, Bosco, Lincoln, Márcio, João Marcelo, João Borges, Marcos, Diana, Valéria, Darilu, Fabrício, Alexandre, Paulo Alexandre, Paulo Henrique, Marco Diego, Vinicius, Carlos Alberto e Windson.

Agradeço à CAPES pelo investimento feito durante essa caminhada e que tornou possível o sucesso deste trabalho.

Por fim, não posso deixar de agradecer a Deus, que sempre iluminou o meu caminho, permitindo-me colher os melhores frutos da vida.

Até agora os filósofos ficam preocupados na interpretação do mundo de várias maneiras. O que importa é transformá-lo.

Karl Marx

Resumo

O uso de dispositivos multi-interface, tais como *smart phones*, tem crescido ao mesmo tempo que as demandas por melhores serviços de mobilidade em redes heterogêneas. Neste cenário, a ideia da continuidade de serviços tornou-se um requisito crucial. Para atender essas demandas, esquemas eficientes de *handover* devem ser desenvolvidos com o objetivo de alcançar o chamado *Handover* Transparente, que significa a mudança de domínios de rede de uma forma transparente e sem a descontinuidade dos serviços para o usuário final. Atualmente, existem diferentes esquemas de *handover* e alguns deles podem envolver diferentes tecnologias de acesso (*Handover* Vertical). Entretanto, o tempo de interrupção do serviço ainda é um problema a ser resolvido. A principal proposta deste trabalho é uma melhoria para *Handovers* Verticais utilizando mobilidade IP, objetivando o tão desejado *Handover* Transparente. Neste trabalho, fez-se uso do *framework* MIH (*Media Independent Handover*) fornecido pelo padrão IEEE 802.21 para habilitar o *handover* vertical em redes heterogêneas. Além disso, propõe-se uma extensão do protocolo FMIPv6 (*Fast Handovers for Mobile IPv6*), o FaHMA (*Fast Handovers using Multicast Addressing*), utilizando endereçamento *multicast* para gerenciar a mobilidade nesses tipos de rede. Para fazer a análise de desempenho, simulações foram utilizadas considerando-se métricas tais com o atraso do *handover* e a perda de pacotes como os critérios mais importantes para avaliar a efetividade da solução. Os resultados destas simulações mostraram que o FaHMA obtêm melhores resultados que o FMIPv6, inclusive em relação aos fatores que determinam a qualidade do funcionamento de aplicações multimídias em rede.

Palavras-chave: Dispositivos Multi-Interface. Mobilidade. Continuidade dos Serviços. *Handover*. *Handover* Vertical. *Seamless Handover*. Endereçamento *Multicast*.

Abstract

The use of multi-interface devices such as smart phones has grown at the same time as the demands for efficient mobility services in heterogeneous networks. In this scenario, the idea of service continuity has become a crucial requirement. To achieve these demands, efficient handover schemes should be developed aiming to achieve Seamless Handover, which means the change of network domains in a transparent way and without services discontinuity to the end user. Currently, there are different schemes for handover and some of them may be used between different access technologies (Vertical Handover). However, the service time disruption is still a major problem to be solved. The main purpose of this study is to propose an improvement for Vertical Handovers using IP mobility, aiming at Seamless Handover. In this work, the framework provided by the MIH (*Media Independent Handover*) IEEE 802.21 is used to enable vertical handover in heterogeneous networks, and propose an extension of FMIPv6 (*Fast Handovers for Mobile IPv6*) called FaHMA (*Fast Handovers using Multicast Addressing*), using multicast in order to manage mobility in these types of networks. To make the performance analysis, we decided for simulations and we considerered metrics such as the handover delay and packet loss as the most important criteria for evaluating the effectiveness of our proposal. Simulation results have shown that FaHMA achieve better results than FMIPv6, including factors that determine the quality of operation in networked multimedia applications.

Keywords: Multi-interface Devices. Mobility. Service Continuity. Handover. Vertical Handover. Seamless Handover. Multicast Addressing.

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

AR	*Access Router*
BU	*Binding Update*
CN	*Correspondent Node*
CoA	*Care-of-Address*
DAD	*Duplication Address Detection*
DVMRP	*Distance Vector Multicast Routing Protocol*
FaHMA	*Fast Handovers using Multicast Addressing*
FBack	*Fast Binding Acknowledgment*
FBU	*Fast Binding Update*
FMIPv6	*Fast Handovers Mobile For IPv6*
FNA	*Fast Neighbor Advertisement*
IEEE	*Institute of Electrical and Electronics Engineers*
IGMP	*Internet Group Management Protocol*
IP	*Internet Protocol*
IPv4	*Internet Protocol Version 4*
IPv6	*Internet Protocol Version 6*
HAck	*Handover Acknowledgement*
HI	*Handover Initiate*
HMM	*Handover Management Module*
MFMIPv6	*Multicast-Suported FMIPv6*
MICS	*Media Independent Command Service*
MIES	*Media Independent Event Service*
MIIS	*Media Independent Information Service*
MIH	*Mobile Independent Handover*
MIHF	*Mobile Independent Handover Function*
MIPv6	*Mobile For IPv6*
MLD	*Multicast Listener Discovery*
MN	*Mobile Node*
MPD User	*Multicast Proxy Device User*
NAR	*New Access Router*

NF	*Network Filter*
NIST	*National Institute of Standards and Technology*
NS-2	*Network Simulator 2*
PAR	*Previous Access Router*
PD	*Proxy Device*
PIMSM	*Protocol Independent Multicast-Sparse Mode*
PMM	*Processing and Monitoring Module*
PrRtAdv	*Proxy Router Advertisement*
QoS	*Quality of Service*
RFC	*Request For Comments*
RtSolPr	*Router Solicitation Proxy*
SAP	*Service Access Point*
SCTP	*Stream Control Transmission Protocol*
SIP	*Session Initiation Protocol*
WIMAX	*Worldwide Interoperability for Microwave Access*
VoIP	*Voice-over-IP*
3GPP	*Third Generation Partnership Project*
3GPP2	*Third Generation Partnership Project 2*

Sumário

1 Introdução

Esta dissertação apresenta uma proposta de melhoria para *Handovers* Verticais. Propõe-se uma extensão do FMIPv6 (*Fast Handovers for Mobile IPv6*) (KOODLI, 2005), fazendo uso de endereçamento *multicast*, com o objetivo de gerenciar a mobilidade em nível de rede. Além disso, utilizamos os serviços do *framework* MIH (*Media Independent Handover*), definido pelo padrão IEEE 802.21 (IEEE, 2009), para viabilizar o *handover* vertical em redes heterogêneas.

Na Seção 1.1, são apresentadas a contextualização e a motivação que impulsionou o desenvolvimento deste trabalho. Em seguida, na Seção 1.2, são expostos os objetivos a serem alcançados com este trabalho, sendo a metodologia utilizada para alcançá-los discutida na Seção 1.3. Por fim, na Seção 1.4, é apresentada a estrutura dos capítulos na qual esta dissertação está organizada.

1.1 Contexto e Motivação

O início do século XXI tem confirmado a relevância da informação como o ativo mais valioso em todos os níveis da sociedade, configurando a chamada "Idade da Informação" (SHACKEL, 2009). As constantes buscas por informações feitas por usuários incentivam o desenvolvimento de novos produtos e serviços, estendendo os horizontes da conectividade para o dia a dia.

Neste cenário contemporâneo, o uso de dispositivos multiacesso, tal como *smart phones* e *tables PCs*, cresce, ao mesmo tempo em que aumentam as demandas por melhores serviços de mobilidade em redes heterogêneas. Essas redes incluem diferentes tipo de tecnologias de acesso via rádio, tal como WiFi (IEEE 802.11) (BIJU, 2009), WiMAX (IEEE 802.16) (ANDREWS; GHOSH; MUHAMED, 2007), tecnologias de rede de terceira geração (3GPP e 3GPP2) (STOCKHAMMER; LIEBL, 2007) (BRADNER et al., 2001) e outros sistemas sem fio. Para lidar com essas demandas, esquemas de *handover* eficientes devem ser desenvolvidos com o objetivo de alcançar o desejável *Handover* Transparente. Este tipo de

handover está associado com a troca de canal entre ponto de acesso, onde o Nó Móvel (*Mobile Node* - MN) ou não sofre nenhuma degradação na qualidade dos serviços, segurança e capacidades; ou sofre com alguma degradação, mas dentro de um intervalo aceitável pelo usuário final (IEEE, 2009).

Atualmente, há diferentes soluções de esquemas de *handover* e algumas delas podem ser usadas entre diferentes tecnologias de acesso, o que configura o *Handover* Vertical. Entretanto, os principais problemas desses esquemas como, a perda de pacotes e o atraso provocado pelo *handover*, que podem alcançar níveis inaceitáveis para aplicações de fluxo contínuo de dados, como áudio e vídeo, e aplicações de tempo real, como aplicações de videoconferência e Voz Sobre IP (VoIP), são ainda problemas a serem resolvidos.

Uma dessas soluções é o protocolo *Fast Handovers for Mobile IPv6* (FMIPv6) (GUNDAVELLI; LEUNG; DEVARAPALLI, 2008). Este é responsável por gerenciar as conexões de dados em nível da camada de rede (camada 3), implementando a chamada mobilidade IP. Dentre os objetivos do FMIPv6, destacamos a redução do atraso do *handover* e da perda de pacotes. Entretanto, por depender de informações da camada 2 e por apenas gerenciar a troca de conexões a nível da camada 3, o FMIPv6 não pode ser aplicado diretamente para gerenciar *Handovers* Verticais. Portanto, deve existir alguma outra tecnologia implementada nas camadas mais baixas (camada 2, por exemplo) que realize o trabalho de gerenciar a troca de conexões entre diferentes tecnologias de acesso, de uma forma transparente para este protocolo.

Além disso, a especificação do FMIPv6 requisita a implementação de mecanismos de armazenamento em *buffer* nos Roteadores de Acesso (*Access Router* - AR) , que serão utilizados durante a tarefa de Detecção de Endereços Duplicados (*Duplication Address Detection* - DAD) para armazenar os pacotes destinados a um determinado MN, provocando a chamada interrupção de serviços, que é o atraso de resposta percebido pelo usuário. O DAD consiste no processo de gerar um endereço IP temporário para o MN, com a garantia de que este endereço não seja duplicado. Contudo, esta tarefa é a que requer a maior quantidade de tempo durante o *handover*, podendo provocar níveis de interrupção inaceitáveis para aplicações de fluxo contínuo de dados e de tempo real (LAI; SHIEH; CHOU, 2009).

A proposta deste trabalho é uma melhoria de *handovers* em redes heterogêneas (*Handovers* Verticais) através de um esquema que considera mobilidade IP como a principal tecnologia, por ser amplamente aceito que as próximas arquiteturas da chamada Quarta Geração (4G) serão baseadas na tecnologia IP (DIAB; MITSCHELE-THIEL, 2009). Neste trabalho, empregamos o FMIPv6 como o protocolo base para gerenciar as conexões de dados. De acordo com os trabalhos de Pérez-Costa et al. (2003) e Diab et al. (2009), foi possível concluir que o FMIPv6

apresenta melhor desempenho dentre os protocolos baseados no *Mobile* IPv6 (MIPv6) (JOHNSON; PERKINS; ARKKO, 2011): *Hierchical Mobile IPv6* (HMIPv6) (SOLIMAN et al., 2008) e o *Proxy Mobile for IPv6* (PMIPv6) (GUNDAVELLI; LEUNG; DEVARAPALLI, 2008).

De forma a melhorar o desempenho do *handover*, propomos um módulo em que o padrão IEEE 802.21, por meio do seu *framework* Mobile Independent Handover (MIH), é usado para adaptar o FMIPv6 para trabalhar em ambientes de redes heterogêneas. Além disso, a interrupção de serviços provocada pelo FMIPv6 durante o período do *handover* será superado pelo uso de endereçamento *multicast*. A ideia básica de se utilizar *multicast* consiste em permitir que o MN mude o seu modo de endereçamento *unicast* para um modo de endereçamento *multicast* durante o *handover*, e então voltar ao seu estado anterior, depois que todas as operações necessárias para a realização do *handover* estiverem concluídas. Como consequência, o MN continuará recebendo os pacotes durante a tarefa de DAD, a tarefa que requer maior quantidade de tempo durante o *handover* (LAI; SHIEH; CHOU, 2009).

Com o objetivo de avaliar e efetividade do nosso esquema, escolhemos trabalhar com simulações devido à impossibilidade de trabalhar com medições, pela falta de recursos disponíveis para a implantação de um *testbed*, que consistiria em um ambiente real; e simulações são mais precisas do que uma modelagem analítica (Precisão) consideramos as métricas de atraso do *handover*. Além disso, consideramos o atraso do *handover* e perda de pacotes como os critérios mais importantes (LAI; SHIEH; CHOU, 2009).

1.2 Objetivos

1.2.1 Objetivos Gerais

O objetivo deste trabalho é propor um mecanismo de melhoria para o processo do *handover* em redes heterogêneas, também chamado de *Handover* Vertical. Nossa solução consiste em um esquema que considera a mobilidade IP como a principal tecnologia para garantir o chamado *Handover* Transparente. Para tanto, fazemos uso do *framework* MIH fornecido pelo padrão IEEE 802.21 para habilitar o *handover* em redes heterogêneas. Além disso, propomos uma extensão do FMIPv6, o FaHMA (*Fast Handovers using Multicast Addressing*), a partir da utilização de endereçamento *multicast* com o intuito de gerenciar a mobilidade nesses tipos de rede. Com essa solução, objetivamos reduzir o atraso de *handover* e a perda de pacotes, critérios mais importantes para avaliar a efetividade de uma solução de *handover*; e reduzir, sem grandes impactos na escalabilidade da rede, a interrupção dos serviços provocada pela atividade DAD, que consiste no atraso da resposta dos serviços percebido pelo usuário final.

1.2.2 Objetivos Específicos

Para atingir os objetivos deste trabalho, as seguintes atividades específicas foram realizadas e os seus resultados alcançados:

- definir de uma camada lógica, localizada entre a camada 2 (enlace) e a 3 (rede), que utiliza os serviços do *framework* MIH descrito no padrão IEEE 802.21 e que, ao mesmo tempo, disponibiliza serviços para as camadas superiores, com o intuito de habilitar o *Handover* Vertical; e
- descrever e implementar uma extensão do protocolo FMIPv6, mediante a utilização de endereçamento *multicast*. Tal protocolo será responsável por gerenciar a conexão de pacotes de dados, garantindo uma redução do atraso de *handover* e da perda de pacotes e, além disso, reduzindo a interrupção dos serviços provocada pela atividade DAD, sem grandes impactos sobre a escalabilidade da rede.
- simular o esquema completo de forma a validar e avaliar sua efetividade com relação ao atraso do *handover* e as perda de pacotes.

1.3 Metodologia

A metodologia científica utilizada para o desenvolvimento desta dissertação pode ser descrita de forma resumida a seguir.

1. Levantamento Bibliográfico

   Inicialmente, foi realizada uma revisão bibliográfica sobre os conceitos e os desafios relacionados ao *Handover* Vertical. Um estudo foi feito sobre os principais protocolos que implementam a mobilidade IP e que são utilizados para gerenciar a conexão dos pacotes de dados, e como o endereçamento *multicast* pode ser utilizado para tornar essa tarefa mais eficiente. Foi realizada também uma revisão bibliográfica sobre o padrão IEEE 802.21 (IEEE, 2009), e como o seu *framework* MIH pode habilitar e facilitar a realização do *Handover* Vertical.

2. Desenvolvimento de uma Técnica para *Handovers* Verticais

   Em outra fase, foi definida neste trabalho uma camada lógica, localizada entre a camada 2 (enlace) e a 3 (rede), o HMM (*Handover Management Module*). Tal camada utiliza os serviços do *framework* MIH descrito no padrão IEEE 802.21 (IEEE, 2009) e, ao mesmo

tempo, disponibiliza serviços para as camadas superiores, com o intuito de habilitar e facilitar o *Handover* Vertical. Dentre os serviços providos para essas camadas superiores, podemos citar: a comunicação transparente entre o MIHF (MIH *Function*) e essas camadas; a responsabilidade pela decisão do *handover*, ou seja, decidir se o *handover* é necessário ou não (detecção do *handover*) e a seleção da próxima rede com a qual será estabelecida uma nova conexão.

3. Definição do Protocolo de *Handover*

   Nessa etapa foi proposto e especificado o protocolo FaHMA (*Fast Handovers using Multicast Addressing*), uma extensão do protocolo FMIPv6, que realiza sua integração com o endereçamento *multicast*. Tal integração consiste em obter as vantagens existentes em ambas as abordagens, tais como: reduzir o atraso de *handover* e a perda de pacotes (FMIPv6) e, além disso, reduzir, sem grandes impactos sobre a escalabilidade da rede, a interrupção dos serviços, que consiste no atraso na resposta dos serviços percebido pelo usuário final.

4. Experimentação e Análise de Desempenho

   Para finalizar, a utilização de simulações foi escolhida para verificar e analisar a nossa proposta. Uma vez definidos e justificados os modelos de mobilidade e o simulador de rede a serem utilizados, o protocolo FaHMA, juntamente com o HMM, foram implementados e avaliados. Durante a análise, consideramos verificar se o FaHMA realmente alcança os objetivos que foram definidos para ele e, para auxiliar nessa análise, comparações foram realizadas com protocolo FMIPv6, que também foi utilizado.

1.4 Estrutura da Dissertação

Além deste capítulo de introdução (Capítulo 1), que descreve o contexto, a motivação, os objetivos e a metodologia deste trabalho, esta dissertação está organizada conforme os seguintes capítulos, seguidos de suas descrições.

- Capítulo 2

  No Capítulo 2, são definidos os conceitos de *handover* e seus tipos, em determinadas categorias. Além disso, discutiremos sobre mobilidade IP, definindo dois protocolos que são bastante utilizados para realizar o *handover* na camada 3 (rede) do modelo ISO/OSI, o MIPv6 (JOHNSON; PERKINS; ARKKO, 2011) e o FMIPv6 (KOODLI, 2005), responsáveis por gerenciar as conexões de dados. Além disso, são apresentados os conceitos

sobre a tecnologia *multicast* e, mediante a descrição do trabalho de Lai et al. (2009), mostraremos como um mecanismo de endereçamento *multicast* pode ser utilizado para reduzir a interrupção dos serviços durante os processos de *handover*.

- Capítulo 3

  No Capítulo 3, é apresentado o padrão IEEE 802.21, que tem os objetivos de habilitar e facilitar o *Handover* Vertical. Definiremos o seu *framework* MIH, que visa a atingir esses objetivos fornecendo alguns serviços para as camadas superiores. Mostraremos os seus principais componentes e definiremos esses serviços disponíveis.

- Capítulo 4

  No Capítulo 4, a nossa proposta é descrita. O detalhamento se divide em duas fases: Na primeira, definiremos uma camada lógica, o HMM. Tal camada utiliza os serviços do *framework* MIH descrito no padrão IEEE 802.21 e, ao mesmo tempo, disponibilizá serviços para o protocolo FMIPv6 com o intuito de criar um esquema de *Handover* Vertical que reduz a perda de pacotes e o atraso de *handover*. A última fase consiste em definir o FaHMA, uma extensão do FMIPv6. Esse protocolo visa a melhorar o esquema anterior, reduzindo a interrupção dos serviços, provocada pela atividade DAD, por meio da integração desse protocolo com um mecanismo de endereçamento *multicast*.

- Capítulo 5

  No Capítulo 5, é descrita as implementações dos protocolos FMIPv6 e FaHMA, juntamente com o HMM, destinados as simulações. Descreve-se também as simulações que foram realizadas. Mostraremos a análise dos resultados obtidos dessas simulações, verificando se o FaHMA alcança seus objetivos a partir de sua comparação com o FMIPv6.

- Capítulo 6

  Por fim, no Capítulo 6, apresentamos as conclusões desta dissertação, mostrando as contribuições da pesquisa com base nos resultados alcançados e discutindo sobre os trabalhos futuros em decorrência de componentes não implementados neste trabalho, e de outras perspectivas surgidas desta pesquisa.

2 *Gerenciamento de Mobilidade*

Neste capítulo, são apresentados os meios de se implementar suporte para à Mobilidade. Com a proliferação dos dispositivos móveis, cresce a expectativa de seus usuários de estarem sempre conectados a qualquer hora e em qualquer lugar (PERERA; SIVARAMAN; SENEVIRATNE, 2004). Tal suporte a mobilidade é implementado através dos chamados *Handovers*.

Na Seção 2.1, apresentaremos a definição de *Handover* e os tipos existentes sobre diferentes ópticas. Na Subseção 2.2, discutiremos sobre Mobilidade IP, apresentando os protocolos que suportam tal serviço. Daremos mais atenção ao protocolo FMIPv6, que utilizaremos em nosso trabalho, descrevendo o seu funcionamento, as vantagens e as desvantagens de sua utilização. Na Seção 2.3, mostraremos como um mecanismo de endereçamento *multicast* pode ser utilizado para evitar a interrupção dos serviços durante a tarefa do DAD (*Duplication Address Detection*). Além disso, definiremos uma forma de reduzir o sobrecarga, provocado pela utilização desse tipo de mecanismo, e consequentemente aumentar a sua escalabilidade. Finalmente, na Seção 2.4, apresentaremos uma conclusão sobre o que foi discutido neste capítulo.

2.1 *Handovers*

O processo de transferir o controle de um serviço em andamento de uma rede para outra, dentro de um mesmo sistema ou entre sistemas diferentes, é denominado *handover* (NOGUEIRA, 2007). O *handover* é realizado quando um dispositivo móvel se desloca em direção a uma outra rede, tornando a comunicação com a sua rede de origem cada vez mais inviável.

Definem-se como redes homogêneas, as redes que englobam apenas pontos de acesso que implementam a mesma tecnologia de acesso via rádio, ou seja, o movimento dos usuários dentro deste tipo de rede será realizado utilizando uma única interface de acesso sem fio. Dentre essas tecnologias de acesso, podemos citar: WiFi (IEEE 802.11) (BIJU, 2009), WiMAX (IEEE 802.16) (ANDREWS; GHOSH; MUHAMED, 2007) e as tecnologias de terceira geração (3GPP e 3GPP2) (STOCKHAMMER; LIEBL, 2007) (BRADNER et al., 2001). Ao realizar um *han-*

dover em ambientes desse tipo, diz-se que foi realizado um *Handover* Horizontal, pois não há mudança de tecnologia. Podemos visualizar essa situação na figura 2.1, onde o MN 1 realiza um *handover* horizontal entre duas redes WiMAX.

Entretanto, o uso de dispositivos multi-acesso, tal como *smart phones* e *tablets PCs*, têm crescido ao mesmo tempo que as demandas por melhores serviços de mobilidade nas chamadas redes heterogêneas. Essas redes incluem diferentes tipo de tecnologias de acesso e, nesse caso, o *handover* é denominado como *Handover* Vertical (IEEE, 2009). Na figura 2.1, exemplificamos essa abordagem realizando um handover vertical do MN 2 de uma célula WiFi para uma célula com uma tecnologia de terceira geração.

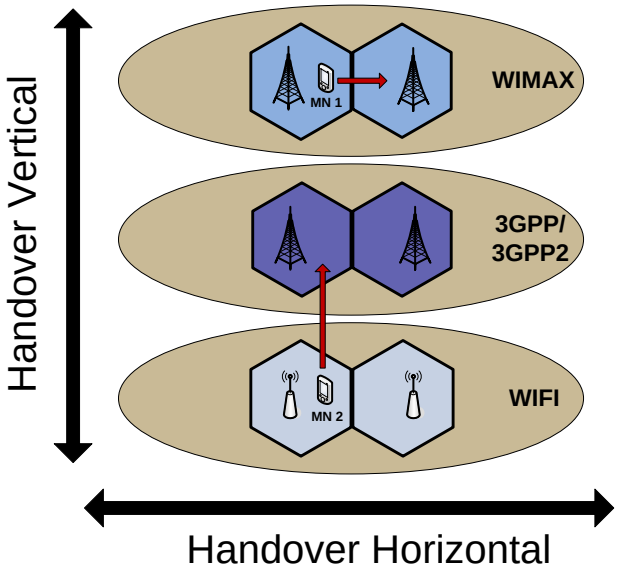


Figura 2.1: Tipos de *Handover*.

Além disso, podemos ainda classificar o *handover* de acordo com à camada onde ele é implementado. Para cada camada, o *handover* executa um processo diferente. Entende-se como *handover* na camada 2 o processo da troca de enlaces, ou seja, da desabilitação do enlace físico com a rede de origem e a criação de outro com a rede de destino. Não obstante, existe também um processo de *handover* implementado nas camadas superiores. Tal processo consiste em gerenciar a comunicação de um dispositivo móvel com os seus serviços. Quando um dispositivo estabelece um enlace físico com uma nova rede, é de seu interesse que os dados trocados com os correspondentes passem a ser enviados por esse novo enlace, ou seja, juntamente com a troca de enlace também deve haver um remanejamento na conexão dos dados.

Atualmente, existem diferentes esquemas de *handover* e alguns deles podem envolver diferentes tecnologias de acesso (*Handovers* Verticais). Esses esquemas tem o objetivo de alcançar o desejável *Handover* Transparente (LEOLEIS; VENIERIS, 2007). Este tipo de *handover* está

associado com a troca de canal entre ponto de acesso, onde o MN ou não sofre nenhuma degradação na qualidade dos serviços, segurança e capacidades; ou sofre com alguma degradação, mas dentro de um intervalo aceitável pelo usuário final (IEEE, 2009).

Como citado anteriormente, os esquemas de *handovers* podem ser implementados em diferentes camadas (LAI; SHIEH; CHOU, 2009). Com relação à camada de aplicação, o SIP (*Session Initiation Protocol*) (ROSENBERG et al., 2002) é usado para suporte de mobilidade (BANERJEE; ACHARYA; DAS, 2006). O SCTP (*Stream Control Transmission Protocol*) (STEWART et al., 2006) é um protocolo da camada de transporte que fornece um bom suporte de mobilidade para essa camada (MA et al., 2004). O alvo de pesquisas, no entanto, em razão da sua ampla transparência para as camadas superiores, são os esquemas de *handovers* implementados na camada de rede, que realizam o chamado *handover* na camada 3 (LAI; SHIEH; CHOU, 2009). O *Mobile* IP (MIP) (PERKINS, 1996) é o protocolo padrão para o gerenciamento da mobilidade na camada de rede (ALNAS; AWAN; HOLTON, 2009)(DO; ONOZATO, 2007). A seguir, são apresentadas duas das implementações deste protocolo.

2.2 Mobilidade IP

O *Mobile* IPv6 (MIPv6) é um exemplo de suporte à mobilidade na camada de rede. Definido na RFC 3775 (JOHNSON; PERKINS; ARKKO, 2011), a principal proposta desse protocolo é garantir a mobilidade em ambientes IPv6 (DEERING; HINDEN, 1998), permitindo que um Nó Móvel (*Mobile Node* - MN) continue alcançável enquanto esteja se deslocando por entre diferentes redes. Para tanto, o MN adquire um endereço temporário (*Care-of-Address* - CoA) quando está em um novo domínio de rede, sem perder o seu endereço original (*Home Address* - HA), ou seja, o MN dispões de dois endereços IP fora do seu domínio de origem. O MIPv6, quando comparado ao seu antecessor, ao MIPv4 (PERKINS, 2002), apresenta algumas otimizações em relação ao roteamento, pois permite que o Nó Correspondente (*Correspondent Node* - CN) armazene uma associação do HA de um determinado MN com o seu respectivo CoA, com o objetivo de estabelecer uma comunicação direta.

Contudo, mesmo com a garantia da mobilidade, a utilização do MIPv6 pode provocar sérios problemas com relação ao provimento de serviços. Dentre esses problemas, os principais são: o alto atraso e a perda de pacotes provocados durante o *handover*. Para solucionar esses problemas, extensões do MIPv6 foram propostas e uma visão geral das principais soluções será apresentada abaixo.

Definido na RFC 5380 (SOLIMAN et al., 2008), *Hierchical Mobile IPv6* (HMIPv6) foi

proposto para ambientes de micro mobilidade, onde os MNs realizam *handovers* frequentemente e onde a utilização do MIPv6 poderia resultar em uma alta carga de sinalização e perda de pacotes, e uma alto atraso do *handover*. No HMIPv6, gerenciamento da mobilidade dentro de um domínio local é assegurado por uma entidade, o *Mobility Anchor Point*, enquanto que a mobilidade entre diferentes domínios MAP é realizado pelo MIPv6. O HMIPv6, através do MAP, reduz a carga de sinalização em domínios locais, reduzindo o atraso do *handover* e a perda de pacotes. Entretanto, esse ganho só será alcançado em *handovers* realizados em um domínio local (PÉREZ-COSTA; TORRENT-MORENO; HARTENSTEIN, 2003).

O *Proxy Mobile for IPv6* (PMIPv6) (GUNDAVELLI; LEUNG; DEVARAPALLI, 2008) foi proposto pelo Network-based Localized Mobility (NETLMM), um grupo de de trabalho do IETF. O PMIPv6 foi proposto para realizar o handover sem o envolvimento do MN, ou seja, todo o processo é realizado na rede, gerando uma arquitetura que melhor se adapta as mudanças nas tecnologias e aos requisitos do mercado. Cada domínio do PMIPv6 é controlado pelo *Localized Mobility Anchor* (LMA). Por outro lado, o suporte a conectividade com o núcleo da rede é realizado pelos *Mobile Access Gateways* (MAGs). Toda vez que um MN iniciar um *handover* para um novo domínio do PMIPv6, o MAG desta rede ficará responsável por identificar o MN e checar se ele está autorizado a utilizar o serviço de gerenciamento de mobilidade. Ao propor uma arquitetura de de gerenciamento de mobilidade sem requerer a implantação de software no MN, o PMIPv6 reduz a carga de sinalização e, consequentemente, reduz o atraso do *handover* e a perda de pacotes (DIAB; MITSCHELE-THIEL, 2009).

Tabela 2.1: Tempo das operações do MIPv6 durante o *handover* (LAI; SHIEH; CHOU, 2009).

Operação	**Tempo Médio**
Handover na Camada 2	50 *ms*
Descoberta de Rotas	100 *ms*
Detecção de Endereços Duplicados (DAD)	1000 *ms*
Enviar *Binding Update*	70 *ms*
Receber *Binding ACK*	70 *ms*

A tabela 1 (LAI; SHIEH; CHOU, 2009) mostra os tempos requeridos pelas operações do MIPv6 que compõe o *handover*. É fácil ver que a Detecção de Endereços Duplicados (*Duplication Address Detection* - DAD) é a atividade que requer a maior quantidade de tempo para ser realizada, atingindo uma média de tempo de 1 segundo, considerada alta em termos computacionais. Esse tempo de duração depende da quantidade de MNs conectados a mesma rede e pode atingir valores bem mais altos que a média. O DAD consiste no processo de gerar um endereço IP temporário para o MN, com a garantia de que este endereço não seja duplicado (LAI; SHIEH; CHOU, 2009). Com o objetivo de reduzir o impacto desta atividade e, consequentemente re-

duzir o atraso do *handover* e a perda de pacote foi proposto o *Fast Handovers for Mobile IPv6* (FMIPv6) (GUNDAVELLI; LEUNG; DEVARAPALLI, 2008).

Estudos foram realizados como forma de comparar o desempenho destes protocolos. Em Pérez-Costa et al. (2003), o FMIPv6 é comparado com HMIPv6 e mostrou-se melhor considerando o atraso do handover e a perda de pacotes, avaliados conforme os impactos do número de nós móveis e do atraso no enlace cabeado. Em Diab et al. (2009), o PMIPv6 foi avalidado em comparação com o FMIPv6 em diferentes tecnologias de acesso via radio (WLAN, GSM e UMTS). Os resultados deste trabalho chegaram a seguinte conclusão: O FMIPv6 executando no modo preditivo sempre é melhor, enquanto que, considerando a atraso do handover e a perda de pacotes, o PMIPv6 alcança resultados comparáveis com o FMIPv6 executando no modo reativo para tecnologias de acesso via rádio consideradas rápidas. Portanto, é possível concluir que o FMIPv6 apresenta melhor desempenho dentre os protocolos acima descritos e, por este motivo, será utilizado neste trabalho.

Na próxima subseção, descrevemos de forma detalhada o funcionamento do FMIPv6. Veremos que ele reduz o atraso do *handover* e a perda de pacotes, utilizando o chamado "*trigger*"da camada 2, que consiste de um evento que ocorre em tempos predeterminados ou não, e que são enviados da camada 2 em direção à camada 3, fornecendo algum tipo de informação de estado do enlace. Utilizando-se desses eventos, o FMIPv6 pode antecipar a atividade DAD antes que ocorra a troca de conexões na camada inferior.

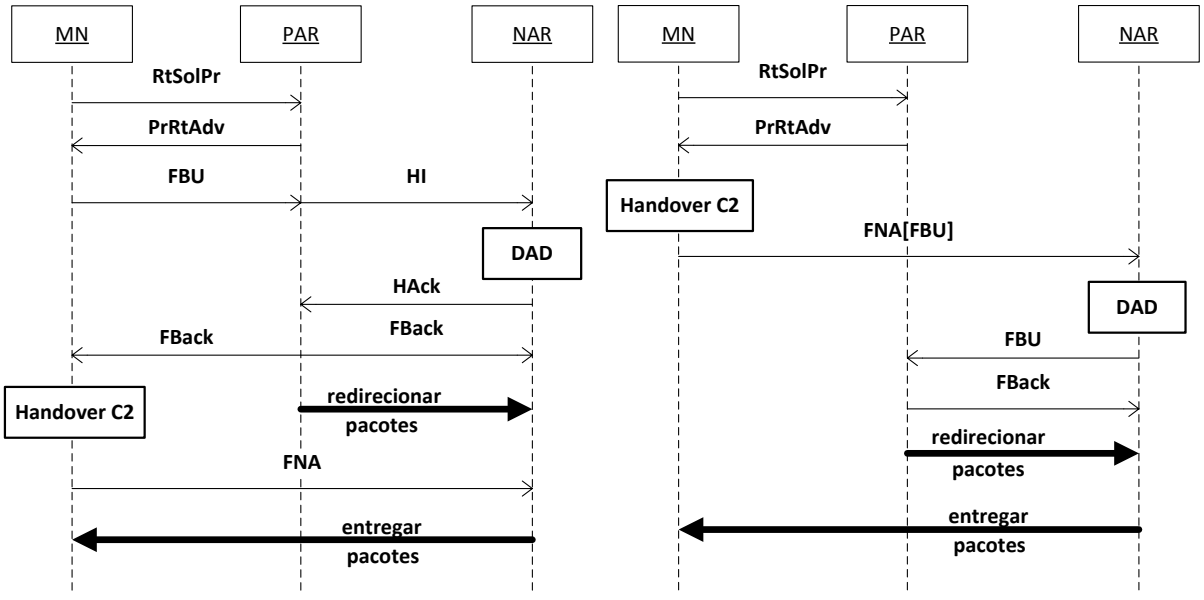
2.2.1 FMIPv6

A proposta principal do MIPv6 é manter a conectividade dos MNs enquanto eles estão se deslocando de um *Access Router* (AR) para outro. Esse protocolo, todavia, sofre pelo seu alto atraso durante o *handover* e alta taxa de perda de pacotes em virtude do atraso na troca de enlaces e as operações do protocolo IP (LAI; SHIEH; CHOU, 2009). Tanto o atraso do handover quanto a perda de pacotes podem atingir níveis que os classifiquem como inaceitáveis para aplicações de tempo real e até mesmo algumas aplicações não sensíveis ao tempo. Anteriormente, citou-se que, dentre as atividades realizados pelo MIPv6, o *Duplication Address Detection* (DAD) é o fator dominante que contribui com o atraso do *handover*, atingindo um tempo médio de 1 segundo, valor bem superior às durações das outras atividades. Por esse motivo, o DAD tem sido o grande alvo de pesquisas que visam a reduzir os problemas provocados pelo MIPv6 (JOHNSON; PERKINS; ARKKO, 2011). Tais pesquisas resultaram na proposta do *Fast Handovers for Mobile IPv6* (FMIPv6). Descrito na RFC 4068 (KOODLI, 2005), esse protocolo estende o MIPv6, com o objetivo de otimizar os procedimentos de *handover*.

O FMIPv6 reduz o atraso do *handover* e a perda de pacotes fazendo uso dos chamados *triggers* da camada 2, que consistem em eventos ocorrentes em tempos predeterminados ou não, e que são enviados da camada 2 em direção à camada 3, fornecendo algum tipo de informação de estado do enlace. Utilizando-se desses eventos, o FMIPv6 pode antecipar a atividade DAD antes que ocorra a troca de conexões. Com o objetivo de reduzir a perda de pacotes, o AR também implementa mecanismos de persistência de pacotes em *buffer*. Nem sempre, entretanto, o DAD poderá ser realizado de forma antecipada e, por isso, o FMIPv6 pode funcionar em dois modos.

- **Modo Preditivo (Figura 2.2a):** executa em uma abordagem *"make-before-break"* (IEEE, 2009). Nesse modo de execução, o DAD é executado antes do *handover* na camada 2. Para tanto, o tempo entre o momento em que o *trigger* da camada 2 ocorre e a troca de conexões deve ser maior do que o tempo requerido para o DAD, isto é, o tempo desde envio do *Fast Binding Update* (FBU) até o recebimento do *Fast Binding Acknowledgement* (FBack);

- **Modo Reativo (Figura 2.2b):** executa em uma abordagem *"break-before-make"* (IEEE, 2009). Nesse caso, o DAD é executado após a realização do *handover* na camada 2. Para tanto, o tempo entre o momento em que o *trigger* da camada 2 ocorre e a troca de conexões deve ser menor do que o tempo requerido para o DAD. Funciona como o MIPv6 em termos do atraso do *handover* e perda de pacotes.



(a) Modo Preditivo (b) Modo Reativo

Figura 2.2: Diagrama de sequência representando o funcionamento do FMIPv6.

Os diagramas de sequência do FMIPv6 em seus dois modos são mostrados na figura 2.2. Inicialmente, em ambos os modos, através das mensagens *Router Solicitation Proxy* (RtSolPr) e *Proxy Router Advertisement* (PrRtAdv), o MN pode obter informações sobre as redes alcançáveis na sua área de cobertura. Essas informações permitem ao MN formular um novo provável CoA enquanto ele ainda está presente na rede do antigo AR (*Previous Access Router* - PAR). Com isso, o atraso decorrente da descoberta do novo endereço IP, que ocorria após o *handover* no MIPv6, é eliminado.

Quando realiza o *handover* no modo preditivo, o MN adquire um novo provável CoA (NCoA) do novo AR (*New Access Router* - NAR), enquanto ainda está conectado ao seu PAR, isto é, o DAD acontece antes da troca de conexões. Os passos realizados na execução do FMIPv6 nesse modo são descritos abaixo (Figura 2.2a):

- Uma vez que o MN detecta que o *handover* é necessário através do *trigger* na camada 2, ele envia um *Fast Binding Update* (FBU), incluindo o NCoA, em direção ao seu PAR, para requisitar a realização do DAD e instruir o PAR a redirecionar o tráfego de dados do MN para o NAR;

- O PAR inicia o armazenamento de pacotes endereçados ao MN em seu *buffer*, após receber uma mensagem FBU desse MN. Em seguida, o PAR envia a mensagem *Handover Initiate* (HI) para o NAR requisitando a realização do DAD e o estabelecimento de um túnel para encaminhar os pacotes de dados armazenados e os novos que chegam destinados ao MN;

- Ao final do DAD, o NAR responde o PAR, enviando-lhe a mensagem *Handover Acknowledgement* (HAck). Se o NCoA não tiver sido aceito pelo NAR, este incluirá um novo NCoA nesta mensagem, ao qual o MN deverá utilizar de agora em diante;

- Depois que o HAck é recebido, o PAR estabelecerá um túnel com NAR, pelo qual encaminhará os pacotes endereçados ao MN. O NAR deve armazenar esses pacotes em seu *buffer* para evitar a perda de pacotes até a chegada do MN. O PAR também envia um *Fast Binding Acknowledgment* (FBack) para o MN e o NAR. Se o FBack inclui um novo endereço, o MN deve utilizá-lo quando associar-se ao NAR; e

- Ao associar-se ao NAR, o MN deve enviá-lo um *Fast Neighbor Advertisement* (FNA). Deste modo, o NAR pode considerar o MN alcançável e, consequentemente, os pacotes armazenados e os novos que chegarem serão reencaminhados para o MN até ele completar o processo do *Binding Update* (BU).

O modo preditivo evita a perda de pacotes. Entretanto, o atraso do *handover* é sensível ao procedimento do DAD, ou seja, quando maior o DAD, maior será o atraso provocado durante o *handover*. Isso ocorre porque os pacotes armazenados no PAR experimentam considerável atraso antes de serem encaminhados para o NAR (LAI; SHIEH; CHOU, 2009).

Por outro lado, se o MN realiza o *handover* na camada 2 antes de receber um FBack, o DAD ocorrerá apenas quando o MN já estiver conectado ao NAR. Nesse caso, o FMIPv6 executa no modo reativo (Figura 2.2b). Nesse modo, o FMIPv6 trabalha como o básico MIPv6 e herda os seus problemas de alto atraso de *handover* e alta perda de pacotes. Se, contudo, o PAR inicia o armazenamento de pacotes (após receber um FBU) antes do MN associar-se a outro domínio, o problema dos pacotes perdidos pode ser evitado. Isso ocorre porque o NAR, ao receber uma mensagem FNA de algum MN funcionando no modo reativo, envia um FBU para o PAR em busca de pacotes que podem ter sido armazenados no *buffer* e que sejam endereçados ao MN.

Neste trabalho, adotamos como base o FMIPv6 de forma a reduzir o atraso de *handover* e a perda de pacotes. Existem, porém, alguns problemas nesse protocolo (LAI; SHIEH; CHOU, 2009) (HUANG; WU, 2009) e outros que surgem quando o utilizamos em ambientes de redes heterogêneas. Esse problemas estão delineados a seguir.

- O FMIPv6 não pode ser aplicado diretamente para gerenciar *Handovers* Verticais, pois, de acordo com a separação em camadas definido no modelo ISO/OSI (TANENBAUM, 2003), os protocolos das camadas superiores utilizam apenas os serviços disponibilizados pelas camadas mais baixas, que trabalham de forma transparente. Portanto, deve existir alguma outra tecnologia implementada nas camadas mais baixas (camada 2, por exemplo) que realize este trabalho de forma transparente para o FMIPv6;

- O *handover* realizado no modo reativo é igual ao MIPv6 em termos de atrasos e perda de pacotes, ou seja, mesmo com operações diferentes, esses dois protocolos são extremamente sensíveis à duração do DAD;

- Em razão do armazenamento de pacotes em *buffer*, pode ocorrer uma profunda interrupção dos serviços, que pode ser inaceitável para aplicações de fluxo contínuo de dados, como áudio e vídeo, e aplicações de tempo real, como aplicações de videoconferência e Voz Sobre IP (VoIP).

2.3 Endereçamento *Multicast* para *Handovers*

Nesta seção, mostraremos como um mecanismo de endereçamento *multicast* pode ser utilizado para evitar a interrupção dos serviços durante a tarefa do DAD (*Duplication Address Detection*). Além disso, definiremos uma forma de reduzir o sobrecarga, provocado pela utilização desse tipo de mecanismo e, consequentemente, aumentar a escalabilidade.

Na Subseção 2.3.1, apresentaremos o conceito de *multicast* e citaremos alguns dos protocolos de roteamento utilizados para prover esse tipo de serviço. Na Subseção 2.3.2, descrevemos o protocolo MFMIPv6, definido em Lai et al. (2009), que propõe um mecanismo de endereçamento *multicast* para evitar a interrupção dos serviços durante o DAD. Mostraremos também que esse mecanismo insere algum sobrecarga na rede. Portanto, para resolver esse problema, na Seção 2.3.3 definiremos o *Proxy Device*, que têm o objetivo de criar um ambiente *multicast*, sem a necessidade do uso de um protocolo de roteamento.

2.3.1 *Multicast*

Ao contrário de um serviço *unicast*, que consiste na entrega de informações de uma única fonte para um destinatário único, e do *broadcast*, onde uma fonte envia mensagens para todos os integrantes de uma rede, o serviço *multicast* permite a entrega de uma mensagem para um subgrupo de nós da rede. Existe uma série de demandas por esses serviços, com aplicações que vão desde atualizações de *software*, fornecidas pelo desenvolvedor para os seus usuários, até aplicações de videoconferência, compartilhadas por diferentes participantes (KUROSE; ROSS, 2010).

Para que seja possível prover um serviço *multicast*, no entanto, é preciso saber como endereçar um pacote *multicast* no momento do envio. Para solucionar esse problema, utiliza-se apenas um endereço para identificar os destinatários de determinado pacote. Tal grupo de destinatários é chamado de grupo *multicast*. Portanto, um pacote identificado com o endereço de um grupo será entregue a todos os destinatários associados a esse grupo. Contudo, nem todo endereço serve para identificar um grupo *multicast*. Nesse caso, uma faixa de endereços IP é reservada única e exclusivamente para o uso de serviços *multicast*. A faixas de endereços para redes IPv4 e IPv6 podem ser consultadas, respectivamente, em Cotton et al. (2010) e na referência da IANA (GROUP, 2011). Na figura 2.3, podemos visualizar como funciona o fluxo de mensagens dentro de um determinado grupo. Nesse caso, o grupo *multicast* é composto por 3 terminais (T1, T2 e T3), configurados para um mesmo endereço IPv6 *multicast* (FF02:0:0:0:0:0:1:5). No cenário exposto na figura 2.3, o T1 envia mensagens para um único endereço de destino, o

FF02:0:0:0:0:0:1:5. Podemos visualizar que o fluxo dos pacotes segue para dois destinos, o T2 e o T3, pois estes estão configurados como participante do grupo *multicast*.

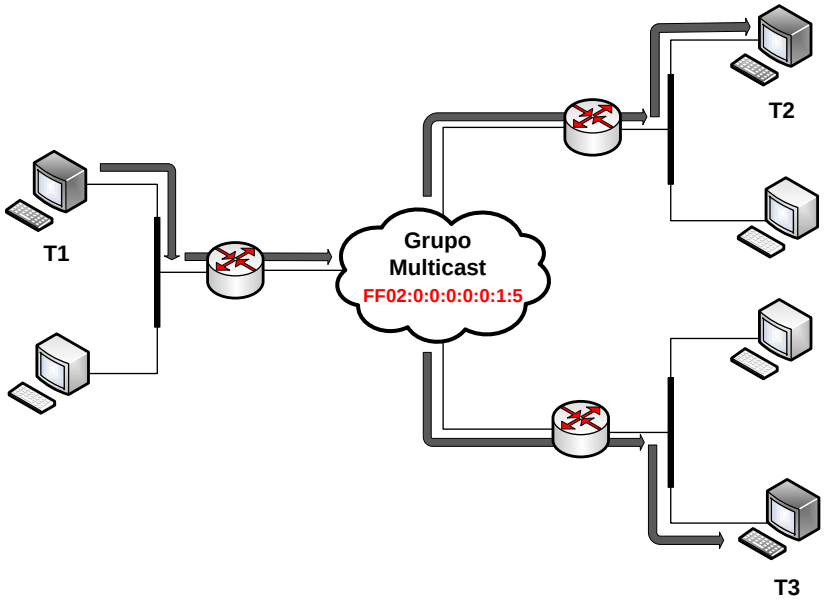


Figura 2.3: Fluxo de Mensagens *Multicast*.

Outro problema para implementar esses tipos de serviços é como identificar os destinatários de um determinado pacote *multicast*. Para isso, faz-se uso do *Internet Group Management Protocol* (IGMP) versão 2 (FENNER, 1997) e 3 (CAIN et al., 2002), para redes IPv4; e do *Multicast Listener Discovery* (MLD) versão 1 (DEERING; FENNER; HABERMAN, 1999) e 2 (VIDA; COSTA, 2004), para redes IPv6. O IGMP/MLD opera entre um *host* e o seu roteador diretamente conectado. Sua função consiste em disponibilizar meios para um *host* informar ao seu roteador que deseja se juntar a um determinado grupo *multicast* e, consequentemente, receber os pacotes endereçados com o identificador desse grupo.

Em determinado ambiente, podemos encontrar vários roteadores *multicast*. Tais roteadores cooperam entre si com o objetivo de encaminhar os pacotes *multicast* para os seus respectivos destinatários. Por outro lado, o escopo do IGMP/MLD inclui apenas a comunicação entre um *host* e roteador ao qual ele está diretamente conectado. Portanto, outro protocolo é necessário para coordenar esses diferentes roteadores. Tais protocolos são conhecidos como os protocolos de roteamento *multicast*.

Na figura 2.4, podemos verificar que os protocolos de roteamento *multicast* se localizam no núcleo da rede interconectando todos os roteadores *multicast*. Dentre estes protocolos, podemos citar: o *Distance Vector Multicast Routing Protocol* (DVMRP) (WAITZMAN; PARTRIDGE, 1998) e o *Protocol Independent Multicast-Sparse Mode* (PIM-SM) (ESTRIN; FARINACCI; HELMY, 1998). Ambos protocolos buscam determinar caminhos para distribuir as mensagens

multicast de um determinado grupo. Para tal, criam a chamada árvore *multicast*, que pode ser compartilhada para todas as fontes de um grupo ou ser específica para cada fonte, com objetivo de distribuir os pacotes de forma a evitar problemas de duplicação de pacotes e ciclos.

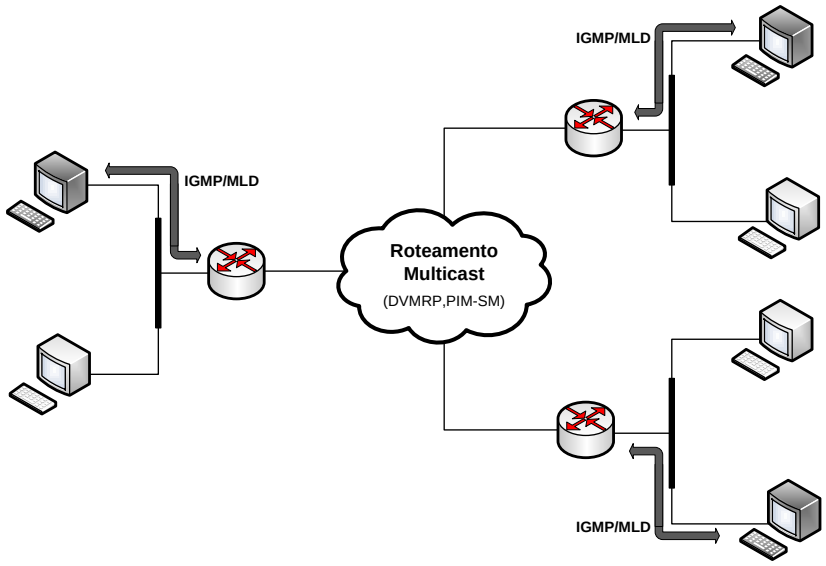


Figura 2.4: Roteamento *Multicast*.

2.3.2 *Multicast-Supported* FMIPv6 (MFMIPv6)

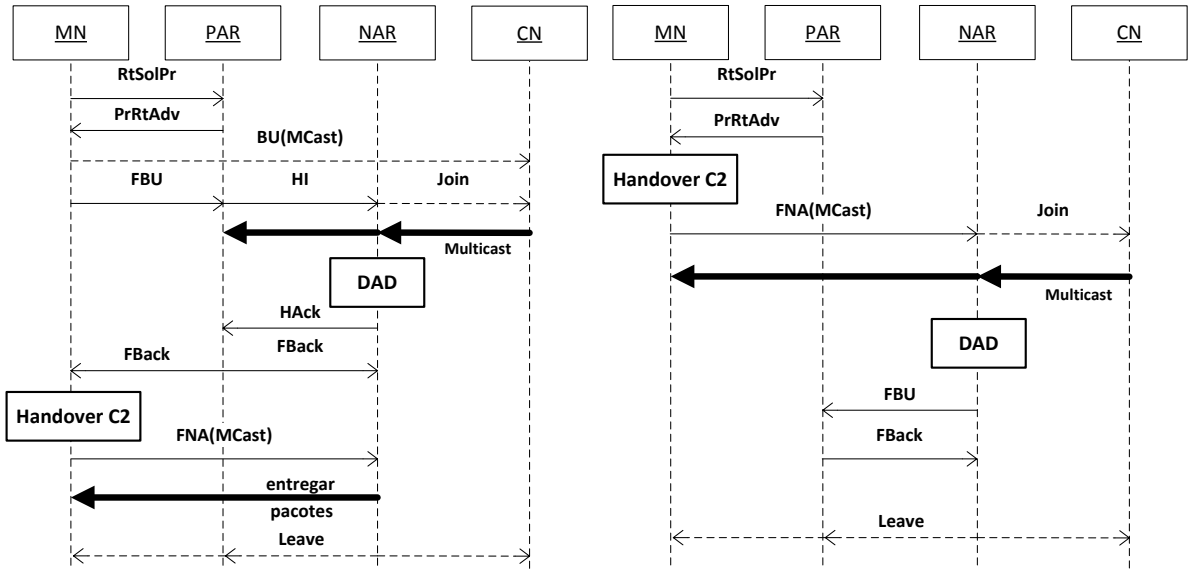
Como discutido no Capítulo 2, o FMIPv6 é um protocolo que provê suporte à mobilidade IP. Esse protocolo é uma extensão do MIPv6 e tem o objetivo de reduzir problemas encontrados neste, como atraso no *handover* e perda de pacotes. Sua execução, todavia, pode resultar em uma profunda interrupção dos serviços, que consiste em um atraso na resposta dos serviços percebido pelo usuário final, durante o processo de *handover*. Tal atraso ocorre devido aos mecanismo de armazenamento em *buffer*, ou seja, os pacotes armazenados no PAR (*Previous Access Router*) podem ficar armazenados por um longo período, antes de serem enviados para o NAR (*Previous Access Router*). Esses mecanismos são utilizados pelo FMIPv6 para reduzir a perda de pacotes. Para resolver esse problema, um esquema de endereçamento *multicast* pode ser usado para evitar essa interrupção durante o DAD (*Duplicate Address Detection*).

Helmy el al. (2004) foram os pioneiros na ideia de utilizar *multicast* para melhorar a eficiência do *handover* para micromobilidade. Inspirado neste trabalho, Lai et al. (2009) propuseram uma extensão do FMIPv6, chamado *Multicast-Supported* FMIPv6 (MFMIPv6). Neste trabalho, um esquema de endereçamento *multicast* foi adotado para reduzir a interrupção dos serviços durante o DAD.

O objetivo central do MFMIPv6 é omitir a fase do *handover* na camada 3 para as aplicações

das camadas superiores, a partir da utilização de um mecanismo de endereçamento *multicast*. A ideia fundamental por trás desse esquema é permitir que o MN (*Mobile Node*) mude o seu modo de endereçamento *unicast* para trabalhar com endereçamento *multicast* durante as operações de *handover*. Essa mudança se dá da seguinte forma: uma vez que o MN detecta a necessidade de realizar um *handover*, iniciará a criação de um grupo *multicast*, o qual fará parte e, ao mesmo tempo, requisitará aos CNs (*Correspondent Nodes*) para que se tornem fontes nesse grupo.

Portanto, após essa mudança para o modo *multicast*, o MN continuará a receber os pacotes necessários para a continuidade dos serviços, agora a partir de um endereço *multicast*. Ao fim das operações de *handover*, o MN retornará ao seu estado anterior, ou seja, voltará para o modo *unicast*, passando a utilizar o seu novo endereço, validado pelo NAR por intermédio do DAD. Abaixo, enumeramos os passos da execução desse protocolo, quando executado no modo preditivo (Figura 2.5(a)).



(a) Modo Preditivo (b) Modo Reativo

Figura 2.5: Diagrama de sequência representando o funcionamento do MFMIPv6 (LAI; SHIEH; CHOU, 2009).

- O MN envia periodicamente uma mensagem RtSolPr (*Router Solicitation Proxy*) e recebe um PrRtAdv (*Proxy Router Advertisement*), com o objetivo de obter informações sobre as redes disponíveis em sua área de cobertura;

- Um vez detectada a necessidade de realização do *handover*, o MN envia um BU (*Binding Update*) aumentado para o CN, incluindo o endereço do grupo *multicast*. Como consequência, ambos mudam para o modo *multicast*, sendo o CN uma fonte do grupo;

- O MN envia um FBU (*Fast Binding Update*) aumentado para o PAR, incluindo o endereço do grupo *multicast*. Ao receber essa mensagem, o PAR muda para o modo *multicast*. Além disso, em vez de iniciar o armazenamento dos pacotes (destinados ao MN) em *buffer*, como ocorre no FMIPv6, o PAR continuará a enviar esses pacotes, utilizando agora o endereço *multicast*;

- Em seguida, o PAR envia um HI (*Handover Initiate*) aumentado para o NAR, também incluindo o endereço do grupo *multicast*. Consequentemente, o NAR também mudará para o modo *multicast*. Uma vez feito isso, ele também passa a receber os pacotes enviados pelo CN e irá armazená-los até que o MN o notifique de sua chegada;

- Ao fim do DAD, o NAR responde o PAR por meio de um HAck (*Handover Acknowledgement*). Ao contrário do que ocorre no FMIPv6, não ocorrerá o estabelecimento do túnel entre essas duas entidades, pois esses pacotes estão sendo entregues agora via rede *multicast*;

- Uma vez recebido o HAck, o PAR envia um FBAck (*Fast Binding Acknowledgement*) para o MN e o NAR. Deste modo, o *handover* na camada 2 será realizado;

- Após o término do *handover* na camada 2, o MN notificará a sua chegada ao NAR, com o envio de uma mensagem FNA (*Fast Neighbor Advertisement*);

- Por fim, o NAR envia uma mensagem LEAVE, indicando a todos os membros do grupo a retornarem para o modo *unicast*.

Uma questão importante a se destacar é o tempo necessário para estabelecer esse grupo *multicast*. Lai et al. (2009) utilizam o trabalho de Cheng et al. (2005) para mostrar que a média de tempo necessário para que um grupo de 50 membros se junte, formando um grupo *multicast*, é de menos de 300 milissegundos. Esse tempo é inferior ao tempo de duração definido para o processo de DAD. Vale ressaltar também que, em Lai et al. (2009), o número de nós participantes do grupo é de apenas 3: o NAR, o PAR e o MN. Outro ponto importante, é como os MNs estabelecem que endereço *multicast* utilizar para estabelecer um grupo. Em Lai et al. (2009), a faixa de endereços IPv6 (FF3X::8000:0-FF3X::FFFF:FFFF), reservados apenas para serviços *multicast* (IANA, 2011), pode ser alocada dinamicamente aos MNs quando necessário.

O emprego desse esquema de endereçamento *multicast*, no entanto, vem acompanhado de problemas de escalabilidade e um sobrecarga operacional, causado pela inclusão de novas mensagens. Esses problemas podem ser verificados em duas situações: a primeira consiste quando, em uma rede, podemos ter centenas de MNs em processo de saída dessa rede, cada MN deve

estabelecer um grupo *multicast* com os seus respectivos PAR, NAR e CN. A segunda situação ocorre quando, em uma rede, podemos ter centenas ou milhares de MNs em processo de entrada; o AR dessa rede deve armazenar pacotes em um *buffer* de todos esses MNs e, consequentemente, uma grande quantidade desses pacotes pode ser descartada, caso a quantidade máxima de armazenamento seja atingida. Outra desvantagem é o sobrecarga causado pelo uso de um protocolo de roteamento *multicast* dentro da rede. Em Lai et al. (2009), considera-se o PIM-SM (ESTRIN; FARINACCI; HELMY, 1998) como o protocolo de roteamento utilizado para manter os grupos *multicast* estabelecidos.

Como forma de reduzir esses problemas, na próxima seção veremos uma abordagem que tem o objetivo de reduzir o sobrecarga operacional, causado pelo uso de protocolos de roteamento *multicast*.

2.3.3 *Proxy Device*

Neste trabalho pretendemos desenvolver um esquema de endereçamento *multicast* para tornar o processo de *handover* mais eficiente, não só em relação a atraso e perda de pacotes, como também considerando o crescimento da rede, ou seja, a escalabilidade. Para tal, pretendemos utilizar a ideia do *Proxy Device* (PD), definido na RFC 4605 (FENNER; HE; HABERMAN, 2006). Detalharemos o nosso esquema no Capítulo 4. Por enquanto, definiremos apenas os objetivos, a infraestrutura e as vantagens do PD.

O PD é definido na RFC 4605 (FENNER; HE; HABERMAN, 2006) e permite criar um ambiente *multicast* sem a necessidade do uso de um protocolo de roteamento *multicast* (DVMRP, PIM-SM) (WAITZMAN; PARTRIDGE, 1998)(ESTRIN; FARINACCI; HELMY, 1998) e, consequentemente, com ganhos em relação ao custo de processamento, sobrecarga e escalabilidade. O PD considera apenas o uso do protocolo auxiliar ao roteamento, o *Multicast Listener Discovery* (MLD) versão 1 e/ou 2, definidos respectivamente nas RFCs 2710 (DEERING; FENNER; HABERMAN, 1999) e 3810 (VIDA; COSTA, 2004).

A principal motivação dessa RFC é a possibilidade de dispensar um protocolo de roteamento *multicast* em ambientes com topologia de árvore, considerada uma topologia simples. Nesse caso, faz-se necessário apenas armazenar informações sobre os membros, utilizando serviços *multicast* inseridos nessa árvore. Com origem nessas informações, é possível replicar os pacotes *multicast* com o objetivo de redirecioná-los para os seus respectivos destinatários.

O PD é a principal entidade desse tipo de ambiente. Ele é responsável por armazenar as informações de *membership*, que são informações sobre os membros gerenciados por uma PD.

Além disso, com amparo nessas informações, um PD realizará o processo de redirecionamento de pacotes *multicast* para os seus membros. Para isto, destaca-se dois tipos de interface implementadas em um PD: uma ou mais interfaces *downstreams*, que gerenciam a comunicação dos membros com o PD; e uma só interface *upstream*, responsável pela comunicação do PD com as redes externas. T

Na figura 2.6, pode ser visualizado onde PD é posicionado em uma infraestrutura de rede, localizando-se entre a rede externa (núcleo da rede) e os seus respectivos membros. Além disso, também pode ser visualizado as duas partes que compõe esta entidade.

Na parte 1, encontramos as interfaces *downstreams* e, além disso, é onde o PD implementa a porção *router* do MLD (versão 1 ou 2), que será responsável pelo recebimento dos *Unsolicited Reports* , enviados pelos seus membros. Tais mensagens, são enviadas quando os membros do PD desejam ingressar em um determinado grupo *multicast*. Ao receber um *Unsolicited Report*, a porção *router* do MLD criará uma *subscription* para o membro requisitante em um banco local encontrado na interface downstream, ao qual o membro está associado. Nessa RFC, os autores definem como *subscriptions*, as informações armazenadas de um determinado membro e, nesse caso, são informações que associam o membro a um grupo *multicast*. Por outro lado, na parte 2, encontramos uma única interface *upstream*, sendo também onde o PD implementa a porção *host* do MLD, que tem por objetivo estabelecer e gerenciar conexões com grupos *multicast*, enviando *Unsolicited Reports* para a rede externa e respondendo as consultas MLD vindas desta.

Finalmente, os membros do PD implementam a porção *host* do MLD (versão 1 ou 2). Entretanto, existem diferenças quando comparado a porção *host* do PD. Nos membros, os *Unsolicited Reports* são enviados para o PD, que será o responsável por gerenciar as suas associações com grupos *multicast*. Além disso, responderá apenas as consultas MLD vindas da porção *router*, implementada no PD.

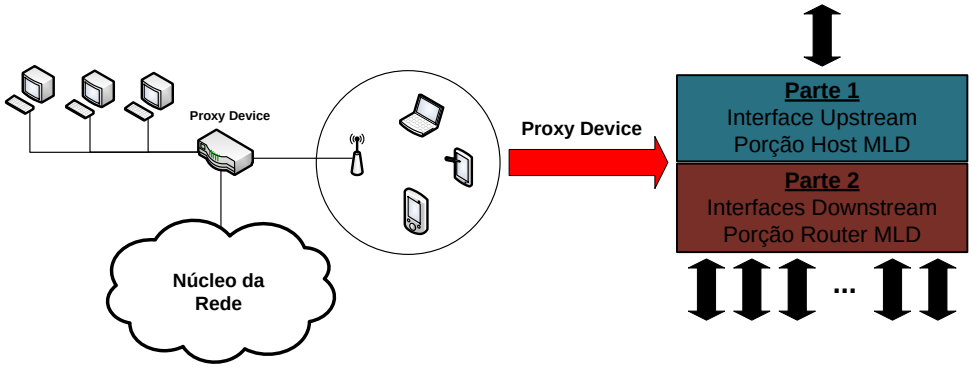


Figura 2.6: Arquitetura do *Proxy Device*.

Para gerenciar todas as conexões dos seus membros com grupos *multicast*, o PD deve ar-

mazenar as chamadas informações *membership*. Tais informações, são armazenadas em um banco de dados e são geradas por um algoritmo de intercalação, definido na própria RFC. Na figura 2.7, podemos visualizar que este algoritmo é responsável por receber informações da parte 2 do PD e transformar em informações que serão armazenadas na parte 1.

O algoritmo de intercalação recebe como entrada todas as *subscriptions* armazenadas nas interfaces *downstreams* do PD e, durante a sua execução, todas essas *subscriptions* são intercaladas, alteradas e resumidas, gerando um único conjunto de informações *membership*. Estas, por outro lado, serão armazenadas em um banco de dados e serão utilizadas para definir o PD como um único nó da rede e o único participante de grupos *multicast* visível pelas rede externa, ou seja, ocultando os seus membros. Resumindo, um PD torna-se o único representante da rede, respondendo por todos os seus membros e, além disso, mantendo os serviços *multicast* disponíveis por meio da replicação e o redirecionamento de pacotes.

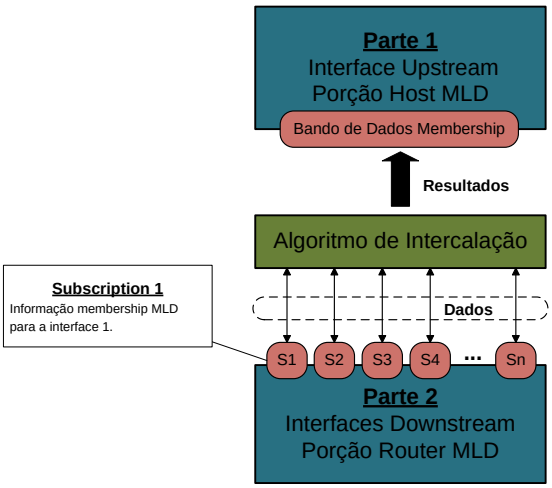


Figura 2.7: Banco de Dados de Informações *Membership*.

O principal requisito para trabalhar com PDs é que a topologia da rede seja limitada por uma árvore. Existem, contudo, algumas vantagens para trabalhar dessa forma. Tais como:

- Simplifica o projeto e a implementação de dispositivos móveis, pois não é necessário implementar um protocolo de roteamento *multicast* no ambiente;
- Por não suportar tais protocolos, reduz o custo de processamento nesses dispositivos e o sobrecarga operacional;
- O ambiente se torna independente do protocolo de roteamento *multicast* utilizado fora da rede, diferente do MFMIPv6, que se amarra ao PIM-SM (ESTRIN; FARINACCI; HELMY, 1998).

2.4 Conclusão

Neste capítulo, apresentamos alguns meios de se implementar um suporte para mobilidade. Inicialmente, definimos o conceito de *Handover* sobre duas diferentes ópticas. Podemos classificá-lo de acordo com o ambiente em que ele foi realizado, nesse caso, podendo ser chamado de *Handover* Vertical ou Horizontal. Além disso, podemos classificá-lo de acordo com à camada onde é implementado. Nesse caso, citamos o *handover* na camada 2, que realiza o processo da troca de enlaces, e o *handover* da camadas 3 e das superiores, responsável por gerenciar a comunicação de um dispositivo móvel com os seus serviços.

Discutimos rapidamente sobre o MIPv6 que, mesmo com a garantia da mobilidade, pode provocar sérios problemas com relação ao alto atraso e a perda de pacotes provocados durante o *handover*. Vimos também que, para solucionar esses problemas, extensões do MIPv6 foram propostas e que o FMIPv6 é a que apresenta um melhor desempenho quando comparado com as principais implementações: o PMIPv6 e HMIPv6. Descrevemos o funcionamento do FMIPv6 em ambos os modos, preditivo e reativo. Apresentamos também alguns dos problemas desse protocolo. Dentre esse problemas, temos: o FMIPv6 não pode ser aplicado diretamente para gerenciar *Handovers* Verticais, o *handover* realizado no modo reativo é igual ao MIPv6 em termos de atrasos e perda de pacotes e, em razão do armazenamento de pacote em *buffer*, pode ocorrer uma profunda interrupção dos serviços, que pode ser inaceitável para aplicações de fluxo contínuo de dados.

Em seguida, definimos e mostramos como o endereçamento *multicast* pode ser utilizado para melhorar o FMIPv6, reduzindo a interrupção dos serviços durante o *handover*. Para isso, apresentamos o MFMIPv6. Entretanto, tal solução apresenta problemas de escalabilidade e um sobrecarga operacional, causado pela inclusão de novas mensagens. Além disso, funciona apenas quando o protocolo de roteamento *multicast* é o PIM-SM. Como forma de minimizar esses problemas, definimos os PDs (*Proxy Devices*). A ideia por trás dos PDs consiste em criar um ambiente *multicast*, sem a necessidade do uso de um protocolo de roteamento, como o PIM-SM.

Vimos que, o FMIPv6 não pode ser aplicado diretamente para gerenciar *Handovers* Verticais. Nesse caso, deve existir alguma outra tecnologia implementada nas camadas mais baixas que realize este trabalho de forma transparente. No próximo capítulo, discutiremos sobre o padrão IEEE 802.21 que define um *framework* que provê tal mecanismo.

3 *Padrão IEEE 802.21*

O uso de dispositivos multiacesso, tais como *smart phones* e *tablets PCs*, cresce ao mesmo tempo que as demandas por melhores serviços de mobilidade em redes heterogêneas. Essas redes incluem diferentes tipos de tecnologias de acesso via rádio, tal como WiFi (IEEE 802.11) (BIJU, 2009), WiMAX (IEEE 802.16) (ANDREWS; GHOSH; MUHAMED, 2007), tecnologias de terceira geração (3GPP e 3GPP2) (STOCKHAMMER; LIEBL, 2007) (BRADNER et al., 2001) e outros sistemas celulares. Para lidar com essas demandas, esquemas de *handover* eficientes devem ser desenvolvidos com o objetivo de alcançar o desejável *handover* transparente.

Com o objetivo de facilitar o *handover* entre diferentes tecnologias de acesso via rádio, também conhecido como *Handover* Vertical, o IEEE desenvolveu um novo padrão, o IEEE 802.21 *Media Independent Handover* (MIH - Handover Independente da Mídia) (IEEE, 2009), que consiste em diferentes dispositivos de acesso via rádio, tais como: *Third Generation* (3G) *Partnership Project* (3GPP), 3GPP2, as tecnologias da família 802 e mídias cabeadas.

De acordo com De La Oliva et al. (2008), o IEEE 802.21 possui os seguintes objetivos:

- Evitar o necessidade de reiniciar a sessão de comunicação após o processo de *handover*, garantindo a chamada continuidade dos serviços, que consiste em manter a disponibilidade dos serviços em curso;
- Permitir o desenvolvimento de aplicações sensíveis à *handover*, ou seja, aplicações com funções para participar das decisões de *handover*;
- Implementar esquemas de *handovers* sensíveis à Qualidade de Serviço (QoS), desde o provimento de funções que permitem realizar decisões de*handover* baseados em critérios de QoS;
- Disponibilizar uma ferramenta para facilitar a descoberta de redes candidatas ao *handover*;

- Prover assistência para a seleção de redes, baseado em vários critérios, tais como: QoS, vazão, políticas, custo etc uma vez que o 802.21 não realiza decisões de *handovers*, atividade reservada para as camadas superiores. Apenas disponibiliza funções para assistir tal atividade; e

- Reduzir o consumo de energia, fornecendo informações que facilitam o execução de *handovers*, reduzindo o processamento operacional. Como exemplo de tais informações, podemos citar: informações de redes na área de cobertura de um Nó Móvel (*Mobile Node* - MN), parâmetros de enlace, etc.

Esses, no entanto, são apenas objetivos secundários, pois o objetivo principal do padrão IEEE 802.21 é o de habilitar e facilitar o *Handover* Vertical com a garantia da continuidade dos serviços. Para alcançar essa funcionalidade, o MIH fornece um mecanismo "inteligente"na camada 2, que disponibiliza alguns serviços e outras informações da rede para as camadas superiores. Basicamente, o padrão IEEE 802.21 é composto pelos seguintes elementos:

- O *framework Media Independent Handover* (MIH), que permite executar *Handovers* Verticais com a garantia da continuidade dos serviços. Esse *framework* não fornece um protocolo de gerenciamento de mobilidade (MIPv6, FMIPv6, SIP) que faça uso de seus serviços. Portanto, o MIH apenas confia na existência de tal protocolo;

- Um conjunto de funções que permitem o controle total sobre os procedimentos de *handover*, funções essas que são disponibilizadas na forma de serviços por uma entidade lógica, chamada de Função MIH (MIH *Function* - MIHF);

- Uma interface de comunicação entre os Usuários MIH (MIPv6, FMIPv6, SIP) e o MIHF, chamada de Ponto de Acesso de Serviços MIH (MIH *Service Access Point* - MIH_SAP), e suas primitivas associadas. Essa interface permite aos Usuários MIH utilizarem os serviços disponibilizados pelo MIHF. Define-se como Usuário MIH os protocolos que funcionam sobre o MIH; e

- Uma interface específica para cada tecnologia de acesso, chamadas *Service Access Points* (SAPs) e primitivas associadas. Essas interfaces permitem a comunicação do MIHF com as tecnologias de acesso, permitindo a recuperação de informações de enlaces e o controle dos procedimentos do *handover* por meio de chamadas de funções.

A Seção 3.1 apresenta o *framework* MIH, mostrando as suas funcionalidades e detalhando a sua principal entidade, o MIHF. Na Seção 3.2, descrevemos os serviços fornecidos pelo MIHF,

importantes para que o IEEE 802.21 alcance os seus objetivos. Na Seção 3.3, apresentamos as primitivas, implementadas pelas interfaces SAPs, com o objetivo de provê os serviços do MIHF. Por fim, na Seção 3.4, uma conclusão do capítulo é feita, resumindo os seus principais tópicos.

3.1 *Framework* MIH

A especificação do padrão 802.21 busca atingir os seus objetivos pela definição de um *framework*, chamado de *Media Independent Handovers* (MIH). O MIH permite executar *Handovers* Verticais com a garantia da continuidade dos serviços. Para tal, define uma entidade lógica, chamada de MIH *Function* (MIHF).

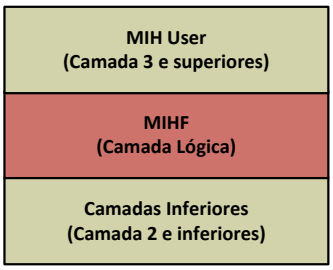


Figura 3.1: MIH *Function*.

O MIHF age como uma camada intermediária entre a camada 2 e as camadas superiores (Figura 3.1). Sua principal função é coordenar a troca de informações e comandos entre os diferentes dispositivos envolvidos na realização da decisão e execução do *handover*. Para isso, provê alguns serviços abstratos para as camadas superiores. Tais serviços são estabelecidos através de primitivas que são definidas para cada tipo de tecnologia de acesso via rádio.

Podemos definir esses serviços de três formas:

- o *Media Independent Event Service* (MIES), que fornece um mecanismo de geração de eventos correspondentes a mudanças nas características, status e qualidade do enlace;
- o *Media Independent Command Service* (MICS), que habilita um usuário do MIHF a gerenciar o *handover* e controlar o comportamento do enlace;
- o *Media Independent Information Service* (MIIS), que disponibiliza informações necessárias para realizar um *handover* com sucesso.

Na próxima seção, descreveremos esses serviços de forma detalhada, apontando as suas funcionalidades, características e formas de comunicação.

3.2 Serviços do MIHF

3.2.1 *Media Independent Event Service* (MIES)

O MIES (Figura 3.2) fornece um mecanismo de geração de eventos correspondente a mudanças nas características, *status* e qualidade do enlace. Ele detecta, filtra, classifica e reporta as mudanças dinâmicas de parâmetros do enlace. Além disso, pode também predizer mudanças nessa camada ou indicar ações de gerenciamento.

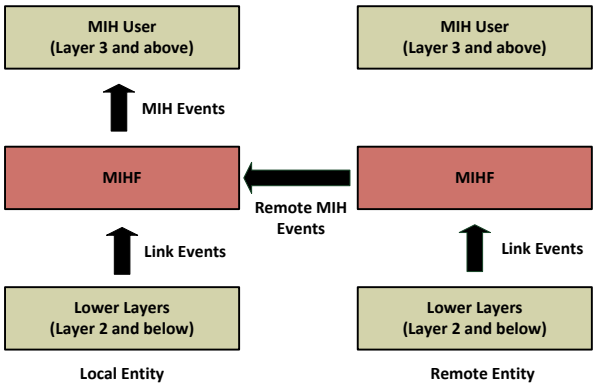


Figura 3.2: *Media Independent Event Service*.

Os eventos podem ser originados do MIHF (MIH *Events* - MIHE) ou de algum meio de acesso através de sua camada 2 (*Link Events*). Esses eventos também podem ser locais, ou seja, são produzidos por meios de acesso do próprio MN, ou podem ser remotos, quando originados por meios de acesso de outros dispositivos. Quando remotos, os eventos são transmitidos utilizando-se o protocolo MIH.

3.2.2 *Media Independent Command Service* (MICS)

O MICS (Figura 3.3) habilita um usuário do MIHF a gerenciar o *handover* e controlar o comportamento do enlace. Para isto, fornece um conjunto de comandos para que esses usuários possam controlar os variados meios de acesso. Através do MICS, o usuário MIH pode determinar o estado do enlace e usar os comandos de *handover* para controlar a reconfiguração e seleção de um apropriado enlace, se requerido.

Os comandos podem ser originados do usuários MIH (MIH *Commands* - MIHC) ou pelo MIHF (*Link Commands*). Esse comandos também podem ser locais, ou seja, gerados pelo usuário do próprio MN, ou podem ser remotos, quando originados por usuários de outros dispositivos. Quando remotos, os comandos são transmitidos utilizando-se o protocolo MIH.

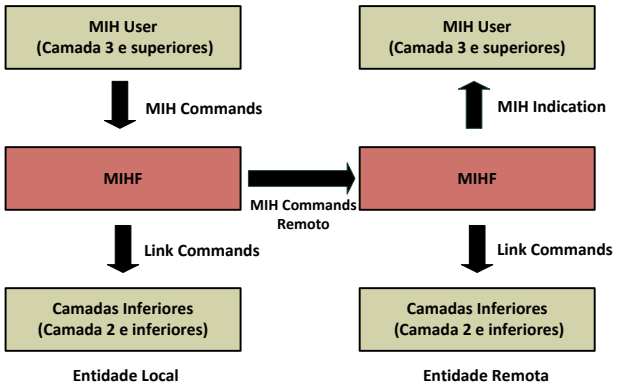


Figura 3.3: *Media Independent Command Service*.

3.2.3 *Media Independent Information Service* (MIIS)

O MIIS (Figura 3.4) descreve um *framework* pelo qual um MIHF remoto obtém informações sobre a disponibilidade de redes de acesso na área de cobertura do MN, habilitando decisões de *handover* mais efetivos, em termos de seleção da nova rede.

O MIHF disponibiliza um mecanismo de consulta (*query/response*) que permite que um provedor de serviços e um MIHF troquem informações. Esse serviço é baseado em *Information Elements* (IE), que fornecem informações essenciais para que um algoritmo de seleção de rede possa realizar um *handover* com sucesso através de redes heterogêneas. Tais informações consistem em:

- Disponibilidade de redes de acesso na área de cobertura do MN;
- Informações de parâmetros estáticos dos meios de acesso que ajudam o MN na seleção de uma rede acesso apropriada;
- Informações sobre as capacidade de diferentes redes de acesso;
- Serviços das camadas superiores disponíveis por diferentes redes de acesso.

3.3 Interfaces SAP

Com o objetivo de habilitar o acesso aos serviços definidos anteriormente, o MIH define primitivas de serviços que são agrupadas em Pontos de Acesso de Serviços (*Service Access Points* - SAPs) (Figura 3.5). Nas próximas subseções, apresentaremos os SAPs definidos pelo

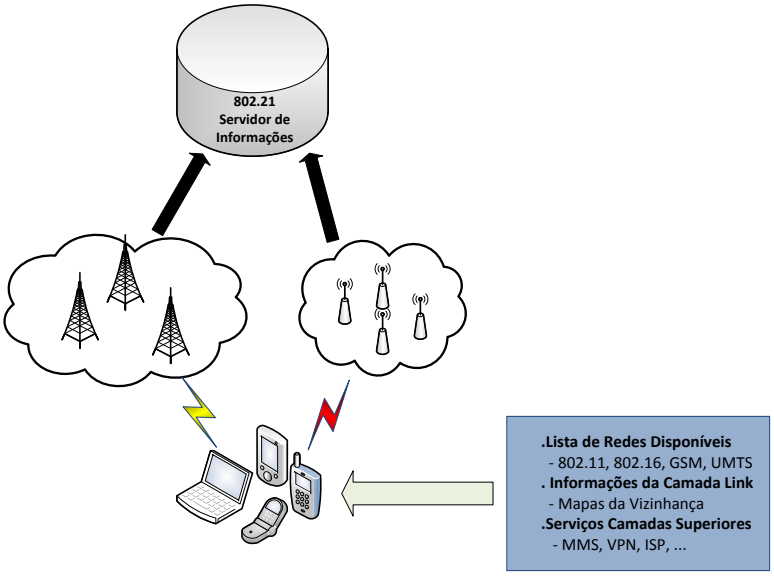


Figura 3.4: *Information Services*.

padrão IEEE 802.21, seguidos da descrição de seus respectivos comandos e eventos, quando estes existirem.

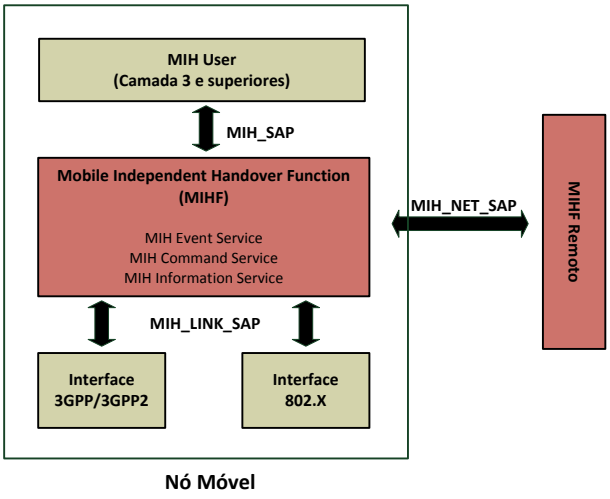


Figura 3.5: Esquema de interfaces SAP.

3.3.1 MIH_SAP

A interface MIH_SAP é responsável por estabelecer um meio de comunicação entre o MIHF e os Usuários MIH. Com essa interface, esses usuários podem requisitar todos os serviços definidos na seção anterior que cabem a ele, tais como: MIHE, MIHC e o MIIS.

As Tabelas 3.1 e 3.2 listam e descrevem as funcionalidades dos respectivos eventos e comandos providos por essa interface:

Tabela 3.1: MIH *Events* (IEEE, 2009).

Evento	**(L)ocal/(R)emoto**	**Descrição**
MIH_Link_Detected	L,R	Informa que um enlace de uma nova rede de acesso foi detectada.
MIH_Link_Up	L,R	Informa que uma conexão na camada 2 foi estabelecida e que o nova enlace está disponível para uso.
MIH_Link_Down	L,R	Informa que uma conexão na camada 2 foi quebrada e que o enlace não está mais disponível para uso.
MIH_Link_Parameters_Report	L,R	Informa que o valor de um parâmetro do enlace ultrapassou o *threshold* pré-definido.
MIH_Link_Going_Down	L,R	Informa que as condições do enlace estão degradando e que a perda de conexão será iminente.
MIH_Link_Handover_Imminent	L,R	O *handover* na camada 2 é iminente baseada nas mudanças das condições do enlace.
MIH_Link_Handover_Complete	L,R	O *handover* na camada 2 está completa, ou seja, um enlace com um novo ponto de acesso foi estabelecido.
MIH_Link_PDU_Transmit_Status	L	Indica o status de transmissão de um PDU.

3.3.2 MIH_LINK_SAP

O MIH_LINK_SAP especifica uma interface abstrata, dependente da mídia utilizada, que permite a comunicação entre a camada 2, pertencente a um meio de acesso qualquer, e o MIHF. Com essa interface, o MIHF pode requisitar todos os serviços definidos na seção anterior que cabem a ele, tais como: *Link Events* e *Link Commands*.

Cada tecnologia de acesso possui um SAP específico que provê a funcionalidade do MIH _LINK_SAP. Portanto, uma interface WiFi possui a sua própria SAP, assim como uma interface WiMAX também possui sua. Todas essas SAP específicas para cada tecnologia são gerenciadas por uma entidade responsável por distribuir os eventos e comandos gerados por ou para uma interface específica.

As Tabelas 3.3 e 3.4 listam e descrevem as funcionalidades dos respectivos eventos e comandos providos por essa interface:

3.3.3 MIH_NET_SAP

A MIH_NET_SAP especifica uma interface abstrata, dependente da mídia utilizada, que será utilizada para suportar a comunicação entre diferente entidades MIHF.

Ela é uma interface que não provê comandos ou eventos, entretanto, apresenta apenas uma primitiva, o MIH_TP_Data, responsável por transferir os dados na rede, trocados por diferentes

Tabela 3.2: MIH *Commands* (IEEE, 2009).

Comando	**(L)ocal/(R)emoto**	**Descrição**
MIH_Link_Get_Parameters	L,R	Captura o status do enlace.
MIH_Link_Configure_Thresholds	L,R	Configura *thresholds* para os parâmetros do enlace.
MIH_Link_Action	L,R	Controla o comportamento de um conjunto de enlaces.
MIH_Net_HO_Candidate_Query	R	A rede inicia o *handover* e envia uma lista de redes sugeridas com os seus respectivos pontos de acesso.
MIH_MN_HO_Candidate_Query	R	Usado pelo MN para consultar e obter informações de*handover* sobre possíveis redes candidatas.
MIH_N2N_HO_Query_Resources	R	Enviado de um MIHF para outro como forma de realizar uma consulta de recursos.
MIH_MN_HO_Commit	R	Comando utilizado pelo MN para notificar a rede atual com informações sobre a nova rede que ele pretende conectar-se.
MIH_Net_HO_Commit	R	Comando utilizado pela rede atual para notificar o MN com informações sobre a nova rede que ele deverá conectar-se.
MIH_N2N_HO_Commit	R	Utilizado pela rede atual para informar a outra rede que um de seus MNs está nesse momento movendo-se em direção a esta. Serve também para iniciar a transferência do contexto (quando aplicável) e preparar para o *handover*.
MIH_MN_HO_Complete	R	Notificação do MIHF do MN para a sua nova rede ou da anterior, indicando o status de término dos processos do *handover*.
MIH_N2N_HO_Complete	R	Notificação da nova rede ou da anterior para o MIHF do MN, indicando o status de término dos processos do *handover*.

MIHFs, e que visam à realização das diversas operações definidas nas seções anteriores.

3.4 Conclusão

Neste capítulo, definimos o padrão IEEE 802.21, um esforço da Comunidade IEEE para desenvolver um padrão para o processo de *Handovers* Verticais. Esse padrão define o *framework* MIH, que tem o objetivo de facilitar o *handover* entre diferentes tecnologias de acesso via rádio.

Para alcançar esse objetivo, utiliza-se do MIHF, uma entidade que é gerada a partir desse *framework*. O MIHF age como uma camada intermediária da camada 2 com as camadas superiores. Sua principal função é coordenar a troca de informações e comandos entre os diferentes dispositivos envolvidos na realização da decisão e execução do *handover*. Para isso, provê alguns serviços abstratos para as camadas superiores, tais como: MIH *Event Service*, MIH *Command Service* e MIH *Information Service*.

Tabela 3.3: *Link Events* (IEEE, 2009).

Evento	**Descrição**
Link_Detected	Informa que um enlace de uma nova rede de acesso foi detectada.
Link_Up	Informa que uma conexão na camada 2 foi estabelecida e que o nova enlace está disponível para uso.
Link_Down	Informa que uma conexão na camada 2 foi quebrada e que o enlace não está mais disponível para uso.
Link_Parameters_Report	Informa que o valor de um parâmetro do enlace ultrapassou o threshold pré-definido.
Link_Going_Down	Informa que as condições do enlace estão degradando e que a perda de conexão será iminente.
Link_Handover_Imminent	O *handover* na camada 2 é iminente baseada nas mudanças das condições do enlace.
Link_Handover_Complete	O *handover* na camada 2 está completa, ou seja, um enlace com um novo ponto de acesso foi estabelecido.
Link_PDU_Transmit_Status	Indica o status de transmissão de um PDU.

Tabela 3.4: *Link Commands* (IEEE, 2009).

Comando	**Descrição**
Link_Capability_Discovery	Consultar e descobrir a lista de serviços.
Link_Event_Subscribe	Registra os tipos de eventos a serem aceitos.
Link_Event_Unsubscribe	Remove registros para um conjunto de eventos.
Link_Get_Parameters	Acessa o valor medido de parâmetros do enlace ativo.
Link_Configure_Thresholds	Configura thresholds para o evento Link_Parameters_Report.
Link_Action	Requisita uma ação para a conexão ativa na camada 2.

Finalmente, mostramos que, para habilitar o acesso a esses serviços, o MIH definem primitivas de serviços que são agrupados em três SAPs: MIH_SAP, MIH_Link_SAP e MIH_NET_SAP.

No próximo capítulo, definiremos a proposta deste trabalho.

4 Proposta

Neste capítulo, descreveremos nosso mecanismo de *Handover* Vertical. Neste trabalho, empregamos o FMIPv6 como o protocolo base para gerenciar as conexões de dados. De acordo com os trabalhos de Pérez-Costa et al. (2003) e Diab et al. (2009), foi possível concluir que o FMIPv6 apresenta melhor desempenho dentre os protocolos baseados no MIPv6 (JOHNSON; PERKINS; ARKKO, 2011): HMIPv6 (SOLIMAN et al., 2008) e o PMIPv6 (GUNDAVELLI; LEUNG; DEVARAPALLI, 2008). Por outro lado, de forma a melhorar o desempenho do *handover*, propomos um módulo em que o padrão IEEE 802.21, por meio do seu *framework* Mobile Independent Handover (MIH), é usado para adaptar o FMIPv6 para trabalhar em ambientes de redes heterogêneas.

Nossa proposta é baseada na seguinte hipótese: se integrarmos o protocolo FMIPv6 com o *framework* MIH definido pelo padrão IEEE 802.21, com o objetivo de habilitar a mobilidade IP em redes heterogêneas, e estendermos o FMIPv6 aplicando um esquema de endereçamento *multicast*, podemos obter um esquema de *handover* transparente sobre redes heterogêneas que reduz a perda de pacotes, o atraso do *handover* e, além disso, a interrupção dos serviços (atraso percebido pelo usuário final) causada pela atividade Detecção de Endereços Duplicados (*Duplication Address Detection* - DAD), sem grandes impactos sobre a escalabilidade da rede.

Inicialmente, na Seção 4.1, discutiremos os trabalhos relacionados, apresentando uma análise crítica. Na Seção 4.2, apresentaremos o Módulo de Gerenciamento de *Handover* (*Handover Management Module* - HMM), responsável por integrar o FMIPv6 e o MIH com o objetivo de habilitar *Handovers* Verticais e prover suporte à mobilidade IP. Também mostraremos que o HMM será responsável por detectar a necessidade da realização do *handover* e pela seleção da próxima rede a se conectar. Sendo neste caso, utilizado o critério de sempre realizar o *handover* no modo preditivo. Na Seção 4.3, apresentaremos a proposta de extensão do FMIPv6, o FaHMA (*Fast Handovers using Multicast Addressing*), que utiliza um esquema de endereçamento *multicast* com o objetivo de reduzir a interrupção dos serviços provocado pelo FMIPv6. Finalmente, na Seção 4.4, apresentaremos uma conclusão sobre o que foi discutido neste capítulo.

4.1 Trabalhos Relacionados

Nesta seção, discutiremos os trabalhos relacionados encontrados na literatura. Tais trabalhos buscam integrar o FMIPv6 com o *framework Media Independent Handover* (MIH) de forma a implementar um mecanismo de *Handover* Vertical. Apresentaremos também uma análise crítica destes.

Em Mussabbir et al. (2007), os autores otimizaram os procedimentos de *handover* do protocolo FMIPv6 em ambientes veiculares, utilizando os serviços do padrão IEEE 802.21, a partir do *framework* MIH. Para isso, eles desenvolveram o chamado *Information Element Container* para armazenar informações estáticas e dinâmicas da camadas 2 e 3 pertencentes às redes de acesso vizinhas. Eles também propõem o uso de um *cache* especial para reduzir o tempo de antecipação no FMIPv6 durante o processo de *handover*, deste modo, incrementando a probabilidade de realizar o *handover* no modo preditivo. Entretanto, os autores não incluem soluções para reduzir a interrupção dos serviços provocados pelo FMIPv6 que podem atingir níveis inaceitáveis para aplicações fluxo contínuo e de tempo real. Além disso, não consideram a possibilidade da ocorrência do *handover* no modo reativo, trabalhando somente com a abordagem que alcança o melhor desempenho, o modo preditivo. Por fim, os resultados dos experimentos apresentados não demonstram significância estatística, não considerando, desta forma, a aleatoriedade do ambiente, necessária por aproximar os resultados obtidos aos de um ambiente real.

Uma solução *cross-layer* foi apresentada em Huang et al. (2009). Neste trabalho, o *framework* MIH foi utilizado para combinar o 802.16e (WANG et al., 2005) e a camada 2 com o objetivo de aumentar o desempenho do *handover* na camada 3. Com o *pre-binding update* e a indicação de *handover*, fornecidos pelo 802.16e, um mecanismo rápido de controle de *handover* foi desenvolvido para reduzir a perda de pacotes e suportar MNs que se movem rápido. Por meio dos comandos do MICS e informações recuperadas pelo MIIS, o processo antecipação do *handover* pode ser evitado, aumentando a precisão do *handover* na camada 3. Entretanto, este trabalho engloba os mesmo problemas do trabalho anterior, ou seja, os autores não consideram a interrupção dos serviços, a possibilidade do *handover* no modo reativo e a significância estatística na análise dos resultados.

Em Kim el al. (2008), foi proposto um esquema de *handover* utilizando FMIPv4 e 802.21. Neste caso, o funcionamento do FMIPv4 é o mesmo do FMIPv6, a única diferença é o uso de endereços IPv4, ao invés do IPv6. Neste trabalho, foi proposto um algoritmo de decisão para melhorar o processo de *handover* na camada 3, em outras palavras, realizar o *handover*

no modo preditivo. Esse algoritmo é baseado em informações fornecidas pelo MIH, tais como: largura de banda disponível e distância física, em relação ao ponto de acesso. Outra vantagem é a redução do atraso provocado durante o *handover* na camada 2, em virtude da redução no tempo de busca de informações de Pontos de Acesso (*Access Point* - AP) candidatos. Não obstante, este trabalho também apresenta os mesmos problemas identificados para os trabalhos anteriores.

Yoon Young An et al (2006) integraram o FMIPv6 com o MIH e propuseram um mecanismo de *handover* eficiente com a definição de novas primitivas e parâmetros para os serviços do MIH definidos no IEEE 802.21. Esse mecanismo reduz o atraso do *handover* quando integrado ao MIPv6, removendo o tempo de descoberta de *Access Routers* (AR). Além disso, quando aplicado ao FMIPv6, incrementa a probabilidade do *handover* no modo preditivo, melhorando o desempenho do *handover*. Entretanto, este trabalho apresenta os mesmos problemas apresentados anteriormente nos outros trabalhos: interrupção dos serviços, a possibilidade do *handover* no modo reativo e a significância estatística na análise dos resultados.

Tabela 4.1: Comparativo dos Trabalhos Relacionados.

	Solução			
Realizam	**Mussabbir et al (2007)**	**Huang et al (2009)**	**Kim el al (2008)**	**An et al (2006)**
FMIPv6 + MIH	Sim	Sim	Sim	Sim
Facilita o Modo Preditivo	Sim	Sim	Sim	Sim
Considera Modo Reativo	Não	Não	Não	Não
Interrupção dos Serviços	Não	Não	Não	Não
Significância Estatística	Não	Não	Não	Não

O padrão 802.21 foi inciado por um grupo de trabalho IEEE no início de 2004, tendo o seu primeiro *draft* lançado em 2005. Entretanto, o padrão IEEE 802.21 foi publicado, na sua forma completa, apenas em janeiro de 2009. Portanto, ainda existem poucos estudos relacionados ao uso desta tecnologia. Os trabalhos citados anteriormente integram o FMIP com o MIH, propondo soluções que podem evitar o *handover* no modo reativo. Eles, porém, não consideram a interrupção dos serviços, a possibilidade do *handover* no modo reativo e os intervalos de confiança. Podemos visualizar estas afirmações na Tabela 4.1. Neste trabalho, todos esses pontos serão considerados.

4.2 Módulo de Gerenciamento de *Handover*

O padrão IEEE 802.21 (IEEE, 2009) surgiu com o objetivo de melhorar o desempenho dos *handovers* verticais e horizontais. A principal proposta desse padrão é habilitar e facilitar o *handover* entre redes que podem ou não incluir diferentes tecnologias de acesso via rádio (redes

heterogêneas). Para isto, define o *framework* MIH, cuja principal entidade é o MIHF, uma camada lógica que age como uma camada intermediária das camadas 2 e 3. Com base numa implementação do MIHF, podemos desenvolver soluções para o gerenciamento de *handovers*, implementando funcionalidades que visam alcançar a continuidade dos serviços.

Por outro lado, o FMIPv6 (GUNDAVELLI; LEUNG; DEVARAPALLI, 2008) é um protocolo que gerencia as conexões de dados a nível da camada de rede. Como citado anteriormente, esse protocolo estende o MIPv6 (JOHNSON; PERKINS; ARKKO, 2011) com o objetivo de reduzir o atraso do *handover* e a perda de pacotes. Entretanto, o FMIPv6 depende diretamente de informações da camada 2 para alcançar o seu melhor funcionamento durante a fase do *handover* (modo preditivo), ou seja, ele não pode ser aplicado diretamente para gerenciar *Handovers* Verticais. Portanto, neste trabalho, integramos o FMIPv6 com o *framework* MIH para criar a proposta-base de um esquema de *Handover* Vertical. Na próxima seção, apresentaremos uma extensão do FMIPv6, onde adicionamos um esquema de endereçamento *multicast* ao protocolo, com o objetivo de obter um esquema mais eficiente.

Como resultado dessa integração (FMIPv6 e o MIH), um novo módulo foi criado: o Módulo de Gerenciamento de *Handover* (*Handover Management Module* - HMM). Esse módulo se comporta como uma camada lógica intermediária, pois situa-se logicamente entre a camada 3 e o MIHF (Figura 4.1), gerenciando a troca de mensagens entre eles. O HMM tem três funções:

- Age como um Usuário MIH, intermediando a camada 3 e o MIHF, com o objetivo de gerenciar a troca de mensagens durante o *handover*, permitindo assim uma comunicação transparente entre o FMIPv6 e o MIHF. O HMM é o responsável por utilizar os serviços do MIHF (MIES, MICS, MIIS) e, ao mesmo tempo, fornecer serviços (comandos e eventos) para o FMIPv6, importantes para realizar o *handover* na camada 3 de forma eficiente;
- É responsável pela decisão do *handover*, ou seja, decidir se o *handover* é necessário ou não; e
- O HMM realiza a seleção da próxima rede a se conectar. A partir de uma lista de redes, sendo que estas se encontram na área de cobertura do nó móvel (*Mobile Node* - MN), o HMM seleciona a melhor delas para realizar o *handover* no modo preditivo. No caso de não existir tal rede, ou seja, não existir uma rede que a qual seja possível realizar o *handover* no modo preditivo, o HMM seleciona a melhor rede para realizar o *handover* no modo reativo.

A Figura 4.2 ilustra a arquitetura do HMM. Ele é composto por duas entidades: o Módulo

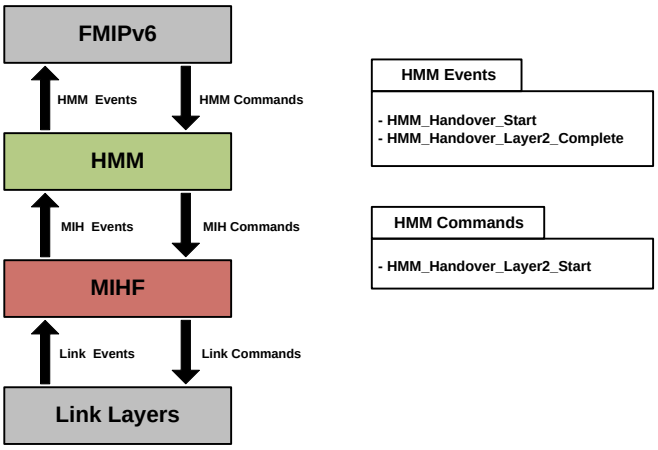


Figura 4.1: Esquema do Módulo de Gerenciamento de *Handover*.

de Monitoramento e Processamento (*Processing and Monitoring Module* - PMM) e o Filtro de Rede (*Network Filter* - NF). Sobre essas entidades, implementamos as três funcionalidades descritas anteriormente, com o objetivo de realizar uma partição de tais responsabilidades. Nas próximas subseções, 4.2.1 e 4.2.2, definiremos essas duas entidades.

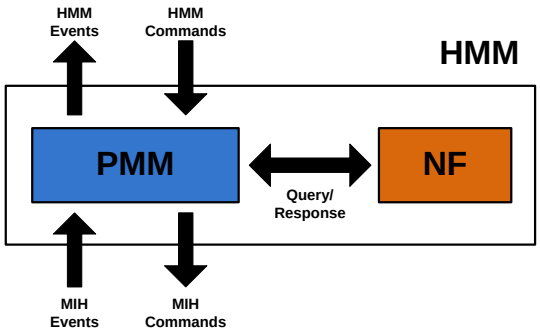


Figura 4.2: Arquitetura do HMM.

4.2.1 Módulo de Monitoramento e Processamento

O Módulo de Monitoramento e Processamento (*Processing and Monitoring Module* - PMM) é a entidade do HMM responsável pelo gerenciamento da troca de mensagens entre o FMIPv6 a o MIHF, intermediando a comunicação dessas duas camadas. Esse módulo permite uma comunicação transparente entre o FMIPv6 e o MIHF, evitando assim uma grande quantidade de mudanças na arquitetura do FMIPv6. O PMM utiliza os serviços do MIHF (MIES, MICS, MIIS) e fornece alguns serviços para o FMIPv6, ou seja, novos eventos e comandos. Tais serviços são chamados de HMM *Commands* e HMM *Events* (Figura 4.1).

Os HMM *Commands* são comandos utilizados pelo FMIPv6 para requisitar algum serviço

ao PMM. O PMM disponibiliza apenas um comando, chamado de *HMM_Handover_Layer2 _Start*. Por outro lado, os HMM *Events* são eventos disparados pelo PMM e recebidos pelo FMIPv6. Esses serviços incluem informações necessárias para a correta realização do *handover* nas camadas 2 e 3. O PMM implementa dois tipos de eventos: o *HMM_Handover_Start* e o *HMM_Handover_Layer2_Complete*. As descrições e as funcionalidades desses comandos e eventos são apresentados na Tabela 4.2.

Tabela 4.2: Mensagens implementadas no PMM.

Nome	**Tipo**	**Funcionalidades**
HMM_Handover_Start	HMM Event	Requisita que o FMIPv6 inicie o *handover* na camada 3.
HMM_Handover_Layer2_Start	HMM Command	Orienta o HMM para iniciar o *handover* na camada 2 a partir do uso dos serviços do MIHF.
HMM_Handover_Layer2_Complete	HMM Event	Requisita que o FMIPv6 envie um *Fast Neighbor Advertisement* (FNA) para *New Access Router* (NAR).

O evento *HMM_Handover_Start* requisita que o FMIPv6 inicie o *handover* na camada 3. Ele é disparado quando o PMM detecta a necessidade de realizar o *handover*, que é sua outra funcionalidade. Em seu corpo, incluirá informações da próxima rede na qual o MN estabelecerá uma nova conexão, permitindo que o FMIPv6 antecipe a tarefa do *Duplication Address Detection* (DAD). Já o comando *HMM_Handover_Layer2_Start* é invocado pelo FMIPv6 quando a tarefa de antecipar o DAD é realizada com sucesso, ou seja, quando o MN realiza o *handover* no modo preditivo. Ele orienta o PMM a iniciar o *handover* na camada 2, utilizando os serviços do MIHF.

Finalmente, o evento *HMM_Handover_Layer2_Complete* requisita que o FMIPv6 envie um FNA (*Fast Neighbor Advertisement*) para NAR (New Access Router), com o objetivo de reativar o recebimento dos pacotes de dados, agora com o MN conectado a uma nova rede. Ele é disparado quando os procedimentos do *handover* na camada 2 são realizados com sucesso. Se esse evento for disparado sem o PMM ter recebido previamente um comando *HMM_Handover_Layer2_Start*, quer dizer que o *handover* foi realizado no modo reativo, ou seja, o *handover* na camada 2 foi realizado antes da tarefa de antecipar o DAD ser concretizada. Além disso, nessa situação, o corpo desse evento incluirá informações sobre a nova rede, necessárias para a realização do *handover* na camada 3.

Além de gerenciar a troca de mensagens entre o FMIPv6 e o MIHF, o PMM também é responsável pelo monitoramento do enlace como forma de detectar a necessidade de realização do *handover*. Entretanto, para implementar um mecanismo de detecção, primeiramente devemos decidir que critérios de decisão serão utilizado para fazer decisões sobre o *handover*.

Em Kashihara et al. (2007), os autores mostraram que o número de retransmissão de quadros é um bom critério de decisão, pois detecta tanto a redução da potência do sinal quanto a interferência de rádio provocada pelo intersecção de várias redes, parâmetros esses decisivos para a detecção da degradação da qualidade da comunicação. Os autores mostraram também que esse critério tem a vantagem de ser de fácil implementação, devendo-se apenas configurar um limite de retransmissões e que, além disso, tem potencial para funcionar de forma otimizada para todos os tipos de aplicações, incluindo as de fluxo contínuo e de tempo real. Portanto, por esse motivos, escolhemos o número de retransmissões de quadros como o nosso critério de decisão para implementar nosso mecanismo de detecção de *handover*.

A implementação desse mecanismo no PMM foi realizada com a utilização de alguns eventos e comandos disponibilizados pelo MIHF. Inicialmente, implementamos o nosso critério de decisão, utilizando um comando do MIH, o *MIH_Link_Configure_Thresholds*. Esse comando permite a configuração de *thresholds* , ou seja, valore-limites mínimos ou máximos para parâmetros dinâmicos do enlace de comunicação, tais como: potência do sinal, velocidade de transmissão, atrasos etc. No nosso caso, utilizamos esse comando para configurar o número de retransmissão de quadros, ou seja, definimos um limite de retransmissões. Como entendido em Kashihara et al. (2007), esse valor-limite deve ser definido de forma empírica, pois, dependendo do ambiente em que será utilizado, poderá assumir diferentes valores. Portanto, não faz parte do escopo deste trabalho definir o melhor valor para este parâmetro.

Ao utilizar o *MIH_Link_Configure_Thresholds*, definimos apenas o limite de retransmissões. O nosso mecanismo de detecção também necessita ser notificado quando esse limite for atingido. Para isso, um evento do MIH, o *MIH_Link_Parameters_Report*, é disparado sempre que algum valor-limite, definido por aquele comando, for ultrapassado. Com o recebimento desse evento, o PMM pode então decidir sobre a realização do *handover*.

O PMM sozinho, porém, não garante que o *handover* no modo preditivo será sempre realizado. Para isso, criamos o segundo componente do HMM, o Filtro de Rede (*Network Filter* - NF). O NF é responsável por selecionar a melhor rede, dentre uma lista de redes identificadas na área de cobertura do MN, que torna possível realizar o *handover* no modo preditivo. Definiremos o funcionamento do NF mais tarde e, por enquanto, utilizaremos essa definição para abordar a última funcionalidade do PMM.

Para finalizar, o PMM também é responsável pela busca e recuperação de uma lista de redes presentes na área de cobertura do MN, que será enviada posteriormente para o NF. Para criar essa lista, o PMM utiliza um comando do MIH, o *MIH_Get_Information*. Esse comando tem a função de requisitar informações disponibilizadas pelo *Media Independent Information*

Service (MIIS) e, dentre essas informações, temos as redes presentes na área de cobertura do MN. Como citado no capítulo 3, a implementação do MIIS não é definida na especificação do padrão IEEE 802.21 (IEEE, 2009) e é deixada a cargo do desenvolvedor. Apenas a definição dos serviços e do mecanismo *query/response* é apresentado. Neste trabalho, tal implementação não foi realizada. Deixaremos como um trabalho futuro, em razão do nível de complexidade de tal serviço. Portanto, assumiremos apenas a ideia de que ele está disponível juntamente com os seus serviços.

4.2.2 Filtro de Rede

Como definimos anteriormente, o Filtro de Rede (*Network Filter* - NF) é responsável por selecionar a próxima rede a se conectar. Esse componente seleciona uma nova rede baseado no critério de ser possível realizar os procedimentos de *handover* com essa rede no modo preditivo do FMIPv6. Para este trabalho, descreveremos apenas o mecanismo *query/response* do NF e uma especificação geral de como implementar o mecanismo de seleção. A especificação detalhada da implementação desse mecanismo não faz parte do escopo deste trabalho e, portanto, será deixada como parte dos trabalhos futuros.

Ao detectar a necessidade de realização do *handover* (limite de retransmissões de quadros ultrapassados), o PMM requisita ao MIIS uma lista de redes disponíveis na área de cobertura do MN. Ao receber essa lista, o PMM envia uma mensagem para o NF, chamada de *HMM_Request_Networks*, com os seguintes parâmetros de entrada (*Input*):

- Identificação do Nó Móvel;
- O instante de tempo em que a mensagem foi enviada;
- A velocidade do Nó Móvel;
- Raio de cobertura das redes na área de cobertura; e
- Lista das redes na área de cobertura.

Os dois primeiros parâmetros (identificação e tempo) são recuperados localmente e serão utilizados no processo de seleção caso seja utilizado algum mecanismo de inferência sobre um histórico de movimentações. Essa parte do mecanismo de seleção será utilizada apenas como apoio ao verdadeiro critério de seleção, que deve ser implementado utilizando os três últimos parâmetros (velocidade, raio de cobertura e a lista de redes), sendo a velocidade do MN recuperada localmente e outros serão requisitados ao MIIS.

Esse critério é baseado no que é definido em Ryu et al. (2009). Neste trabalho, os autores mostram que se pode obter uma probabilidade de falha na realização do *handover* no modo preditivo quando um MN decide mudar de uma rede para outra. Essa probabilidade é afetada por três parâmetros: raio das células, velocidade do MN e o tempo em que foi decidida a realização do *handover*. Esse último parâmetro é preenchido pelo nosso mecanismo de decisão utilizando o número de retransmissão de quadros, que foi caracterizado como sendo uma abordagem eficiente (KASHIHARA; TSUKAMOTO; OIE, 2007). Um modelo do NF com suas mensagens e parâmetros pode ser visualizado na Figura 4.3.

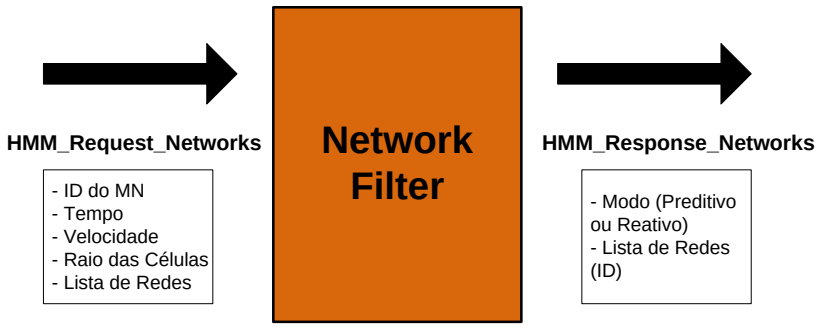


Figura 4.3: Filtro de Rede

Ao receber um *HMM_Request_Networks*, o NF deverá calcular a probabilidade de falha do *handover* no modo preditivo, para cada uma das redes presentes na lista, de acordo com o método definido em Ryu et al. (2009). Para este trabalho, as redes selecionadas para realizar o *handover* no modo preditivo deverão ter esta probabilidade com o valor máximo aceitável de 0%. Entretanto, uma avaliação no ambiente de execução pode ser realizada para determinar um valor máximo aceitável dessa probabilidade, como forma de melhorar o processo de seleção. Entretanto, como está investigação pode ser realizada não faz parte do escopo deste trabalho. Para as redes cujo o valor probabilidade é menor ou igual ao valor máximo definido, um filtro com novas políticas de seleção pode ser utilizado, entretanto, esse filtro não faz parte do escopo deste trabalho. Caso nenhuma das redes tenha atingindo uma probabilidade até o valor máximo definido, será retornada para o PMM a própria lista recebida, como forma de selecionar apenas uma rede para realizar o *handover* no modo reativo. Essa lista também pode passar ou não por um filtro.

O NF responde ao PMM através da mensagem *HMM_Response_Networks*. Essa mensagem incluirá dois parâmetros de retorno, um incluindo o modo do *handover* e o outro, a lista de redes selecionadas. O modo do *handover* indica qual tipo (preditivo ou reativo) deverá ser realizado para a lista de redes retornada. Esse modo de *handover* retorna inclui um valor inteiro, onde o

valor 1 indica que o *handover* deverá ser realizado no modo preditivo e o valor 0 indica que o *handover* deverá ser do tipo reativo. Para a lista de redes retornada, o MN escolherá a primeira delas. A Tabela 4.3 resume as etapas, de forma ordenada, que uma implementação do NF deverá realizar no processo de seleção.

Tabela 4.3: Etapas do processo de seleção do Filtro de Rede.

Etapa	**Descrição**
1	Calcular as probabilidades de falha do *handover* no modo preditivo para cada uma das redes recebidas como entrada.
2	Selecionar as redes com uma probabilidade igual ou inferior ao valor máximo determinado (inicialmente 0%) e indicar que o *handover* deverá ser realizado no modo preditivo.
3	Caso nenhuma rede seja selecionada, todas as redes serão selecionados e deve-se indicar que o *handover* deverá ser realizado no modo reativo.
4	Aplicar um filtro de políticas de seleção (histórico, largura de banda, etc) as redes selecionadas.
5	Retornar as redes selecionadas em todas as etapas, com o respectivo modo de *handover* (preditivo ou reativo).

Na Subseção 4.2.3 mostraremos o passo a passo do funcionamento do HMM durante o processo de *handover* tanto no modo preditivo, quanto no reativo.

4.2.3 Funcionamento Passo a Passo

Nas subseções anteriores, mostramos como foi feita a integração do FMIPv6 como o *framework* MIH, que tem por objetivo criar uma proposta-base de um esquema de *Handover* Vertical que engloba as vantagens fornecidas por essas duas abordagens. Definimos o HMM, descrevendo a implementação das funcionalidades incluídas em suas entidades (PMM e NF). Não houve, porém, uma preocupação em seguir uma ordem temporal na definição dos seus componentes. Nessa subseção, apresentaremos o funcionamento passa a passo do HMM durante o processo de *handover*, exemplificando situações em que ele ocorre, tanto no modo preditivo quanto no reativo.

Iniciaremos mostrando a sequência de passos realizados para executar um *handover* no modo preditivo. Essa sequência poderá ser visualizada no diagrama da Figura 4.4. Seguem os passos necessários.

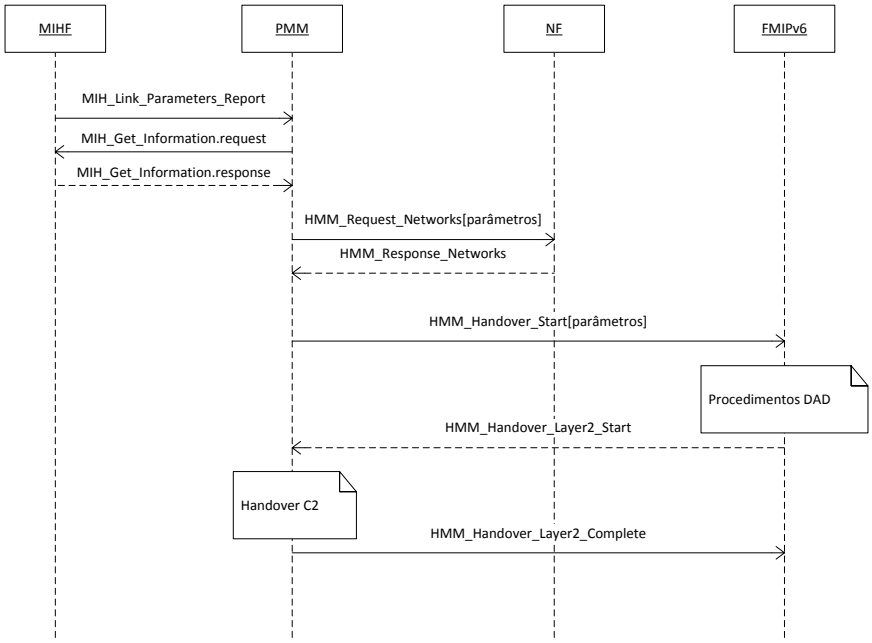


Figura 4.4: Diagrama representando o funcionamento do HMM no modo preditivo.

- Ao identificar que o número limite (*threshold*) de retransmissões de pacotes foi ultrapassado, o MIHF dispara o evento *MIH_Link_Parameters_Report* que será recebido pelo PMM no HMM;

- Ao receber esse evento, o PMM decide pela realização do *handover* e executa o comando *MIH_Get_Information*, com o objetivo requisitar ao MIHF uma lista de redes, com seus respectivos parâmetros, presentes na área de cobertura do MN;

- Com o recebimento dessas informações, o PMM utiliza o mecanismo *query/response* do NF, ou seja, envia a mensagem *HMM_Request_Networks* para o NF, incluindo os parâmetros definidos na Subseção 4.2.2, com o objetivo de selecionar uma rede apropriada para conseguir realizar o *handover* da camada 3 no modo preditivo;

- Ao finalizar os processos para a seleção das redes, o NF envia a mensagem de resposta *HMM_Response_Networks* para o PMM. Essa resposta inclui uma lista de redes e o valor do modo de *handover* preenchido com 1 (preditivo);

- De posse dessas informações, o PMM seleciona a primeira opção das redes da lista e, verificando que o *handover* deverá ser realizado no modo preditivo (modo de *handover* igual a 1), o PMM dispara o evento *HMM_Handover_Start* que será recebido pelo FMIPv6, incluído como parâmetro o identificador da rede previamente selecionada;

- Ao receber tal evento, o FMIPv6 inicia o *handover* na camada 3, ou seja, o processo de antecipação da tarefa DAD, que consiste nos procedimentos de troca de mensagens que

iniciam com o envio de um *Fast Binding Update* (FBU) pelo MN, e terminam com o recebimento de um *Fast Binding Acknowledgment* (FBack) por parte do mesmo;

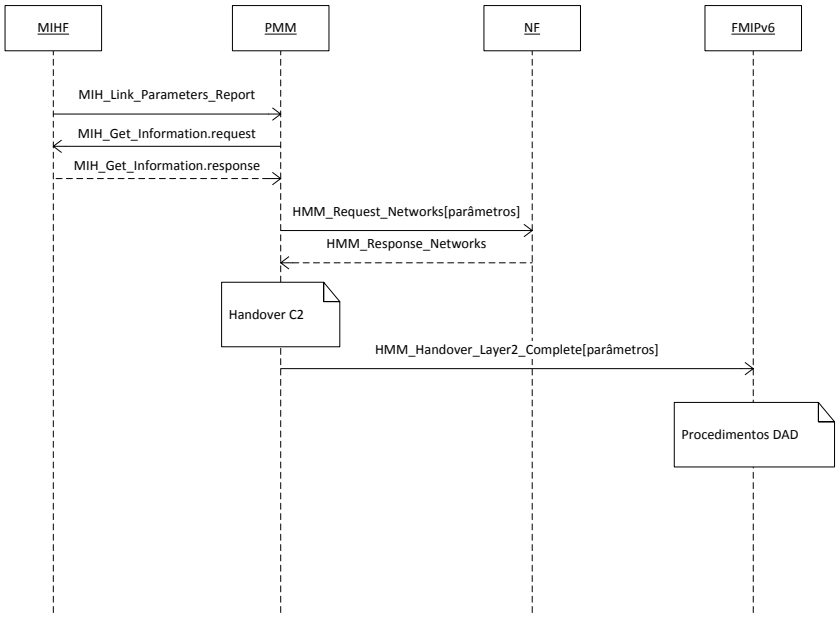
- Ao fim desses procedimentos, o FMIPv6 executa o comando *HMM_Handover_Layer2 _Start*, indicando ao PMM para iniciar os procedimentos para realizar o *handover* na camada 2, ou seja, estabelecer uma conexão com a nova rede e finalizar a conexão com a antiga; e

- Utilizando as primitivas fornecidas pelo MIHF, o PMM gerencia o *handover* na camada 2 e, ao fim desse processo, dispara o evento *HMM_Handover_Layer2_Complete*, indicando ao FMIPv6 para enviar a mensagem *Fast Neighbor Advertisement* (FNA) para o nova rede (NAR), requisitando a continuação do envio de pacotes de dados, agora fornecidos por essa nova rede.

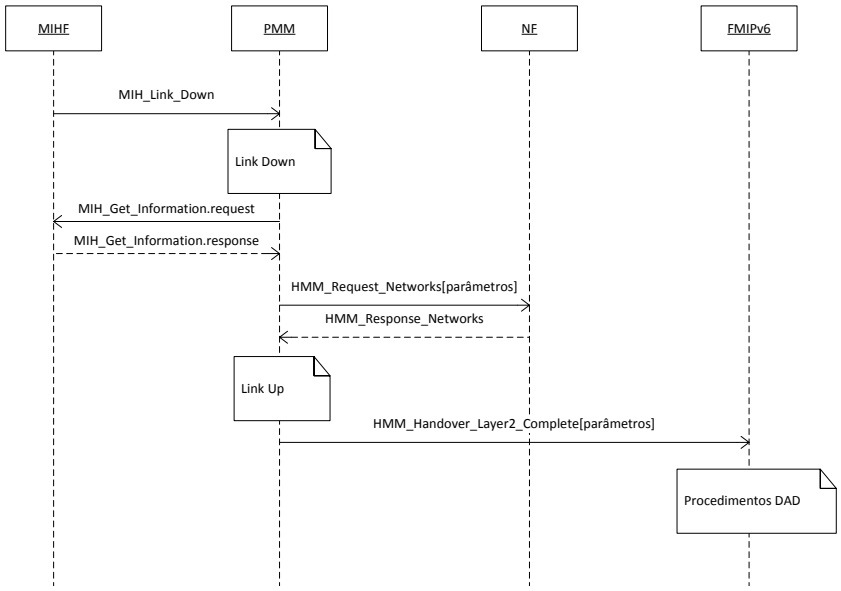
O *handover* no modo reativo ocorre quando o tempo para realizar a antecipação do DAD é insuficiente, forçando o *handover* na camada 3 ocorrer após o da camada 2. O modo reativo pode suceder de duas formas em nossa solução:

- Quando o MIHF dispara o evento *MIH_Link_Parameters_Report* e o NF retorna, através da mensagem *HMM_Response_Networks*, uma lista de redes e o modo de *handover* preenchido com o valor 0; e

- Quando o PMM recebe um evento *MIH_Link_Down* gerados pelo MIHF. Esse evento é disparado quando a conexão com a rede atual está na iminência de ser perdida. Essa perda de conexão pode ser causada por diversos fatores, dentre as quais a redução do nível da potência de sinal, sobrecarga na rede, interferências etc.



(a) Situação 1



(b) Situação 2

Figura 4.5: Diagrama representando o funcionamento do HMM no modo reativo.

Quando o *handover* ocorrer na primeira opção descrita acima, teremos a seguinte sequência de passos na execução do HMM (Figura 4.5(a)):

- Ao identificar que o número limite (*threshold*) de retransmissões de pacotes foi ultrapassado, o MIHF dispara o evento *MIH_Link_Parameters_Report* que será recebido pelo PMM no HMM;
- Ao receber esse evento, o PMM decide pela realização do *handover* e executa o comando *MIH_Get_Information*, com o objetivo requisitar ao MIHF uma lista de redes, com seus

respectivos parâmetros, presentes na área de cobertura do MN;

- A partir do recebimento dessas informações, o PMM utiliza o mecanismo *query/response* do NF, enviando a mensagem *HMM_Request_Networks* para o NF, incluindo os parâmetros definidos na Subseção 4.2.2, com o objetivo de selecionar uma rede apropriada para conseguir realizar o *handover* da camada 3 no modo preditivo;

- Ao finalizar os processos para a seleção das redes, o NF envia a mensagem de resposta *HMM_Response_Networks* para o PMM. Essa resposta inclui uma lista de redes e o valor modo de *handover* preenchido como 0 (reativo);

- Uma vez recebidas essas informações, o PMM seleciona uma das redes da lista (preferencialmente, a primeira) e, verificando que o *handover* deverá ser realizado no modo reativo (modo de *handover* diferente de 1), inicia os procedimentos para realizar o *handover* na camada 2, utilizando as primitivas fornecidas pelo MIHF; e

- Com o fim desse processo, o PMM dispara o evento *HMM_Handover_Layer2_Complete*, incluindo o identificador da rede selecionada. Esse evento indica ao FMIPv6 para enviar uma mensagem FNA com uma mensagem FBU encapsulada, com o objetivo de realizar *handover* na camada 3 e para continuar o recebimento dos pacotes de dados, agora pela nova rede.

Finalmente, a ocorrência do *handover* na segunda forma é realizada pela seguinte sequência de passos (Figura 4.5(b)):

- Quando o MN estiver no limite de perder a sua conexão com a rede atual, o MIHF dispara o evento *MIH_Link_Down*;

- Ao receber esse evento, o PMM decide pela realização do *handover* e executa o comando *MIH_Get_Information*, com o objetivo requisitar ao MIHF uma lista de redes, com seus respectivos parâmetros, presentes na área de cobertura do MN;

- Recebidas essas informações, o PMM utiliza o mecanismo *query/response* do NF, enviando a mensagem *HMM_Request_Networks* para o NF, incluindo os parâmetros definidos na Subseção 4.2.2, com o objetivo de selecionar uma rede apropriada para conseguir realizar o *handover* da camada 3;

- Ao finalizar os processos para a seleção das redes, o NF envia a mensagem de resposta *HMM_Response_Networks* para o PMM. Essa resposta inclui uma lista de redes e o valor modo de *handover* preenchido como 0 (reativo);

- De posse dessas informações, o PMM seleciona uma das redes da lista (preferencialmente, a primeira) e, verificando que o *handover* deverá ser realizado no modo reativo (modo de *handover* diferente de 1), inicia os procedimentos para realizar o *handover* na camada 2, utilizando as primitivas fornecidas pelo MIHF; e

- Com o fim desse processo, o PMM dispara o evento *HMM_Handover_Layer2_Complete*, incluindo o identificador da rede selecionada. Esse evento indica ao FMIPv6 para enviar uma mensagem FNA com uma mensagem FBU encapsulada, com o objetivo de realizar *handover* na camada 3 e para continuar o recebimento dos pacotes de dados, a partir disto pela nova rede.

4.3 FaHMA

Há, na literatura, trabalhos que buscam realizar o *handover* sempre no modo preditivo (MUSSABBIR et al., 2007) (HUANG; WU, 2009) (KIM et al., 2008) (AN et al., 2006). Entretanto, estes não incluem soluções para reduzir a interrupção dos serviços provocada pela operação do FMIPv6, que podem ser prejudiciais para aplicações de fluxo contínuo e em tempo-real. Um esquema de endereçamento *multicast* pode ser usado para evitar essa interrupção. Esse tipo de esquema consiste em manter o recebimento dos pacotes de dados por parte do MN durante o *handover* na camada 3, que antes era interrompido em decorrência do processo de armazenamento de pacotes em *buffer*, ocasionado pela atividade DAD e implementado nos ARs.

Como citado no capítulo 2, porém, o uso do esquema de endereçamento *multicast* definido em Lai et al. (2009) vem acompanhado de problemas de escalabilidade e um sobrecarga operacional, causado pela inclusão de novas mensagens. Ademais, tal protocolo tem o pré-requisito de que o protocolo de roteamento *multicast* utilizado na rede seja o PIM-SM (ESTRIN; FARINACCI; HELMY, 1998). Para solucionar tais problemas, a partir do uso *Proxy Device* (PD) (FENNER; HE; HABERMAN, 2006), podemos desenvolver redes de comunicação *multicast* sem a necessidade de um protocolo de roteamento *multicast* (DVMRP, PIM-SM) (WAITZMAN; PARTRIDGE, 1998) (ESTRIN; FARINACCI; HELMY, 1998) e, consequentemente, ter ganhos em relação ao custo de processamento, sobrecarga e escalabilidade. Além disso, por funcionar isolando os nós das outras redes, o PD têm a flexibilidade de trabalhar com qualquer protocolo de roteamento *multicast* que esteja executando na rede externa.

Neste trabalho, estendemos o FMIPv6 através de sua integração com um mecanismo de endereçamento *multicast*. Para desenvolver tal mecanismo, partimos da ideia apresentada por Lai et al. (2009) e a adaptamos para trabalhar sobre a arquitetura do PD. Com isso, exploramos

as vantagens das duas tecnologias, criando uma extensão do protocolo FMIPv6 que, além de reduzir a interrupção dos serviços, reduz o impacto sobre a escalabilidade da rede através de uma menor sobrecarga de mensagens.

Dessa integração (FMIPv6 e *multicast*), propomos um novo protocolo, chamado de FaHMA (*Fast Handovers using Multicast Addressing*). A descrição da sua implementação e do seu funcionamento será mostrada a seguir.

Inicialmente, buscamos inserir o PD no nosso ambiente. Temos que nosso cenário é composto por várias redes compostas por diferentes tecnologias de acesso. Uma rede neste cenário é composta por um Roteador de Acesso (*Access Router* - AR) e os seus respectivos Nós Móveis (Mobile Nodes - MN), o que configura a sua topologia como a de uma árvore de profundidade 1. Portanto, como o principal requisito para se trabalhar com PD é que a topologia da rede seja limitada por uma árvore (FENNER; HE; HABERMAN, 2006), cada rede pode incluir um PD em seu respectivo AR (Figura 4.6).

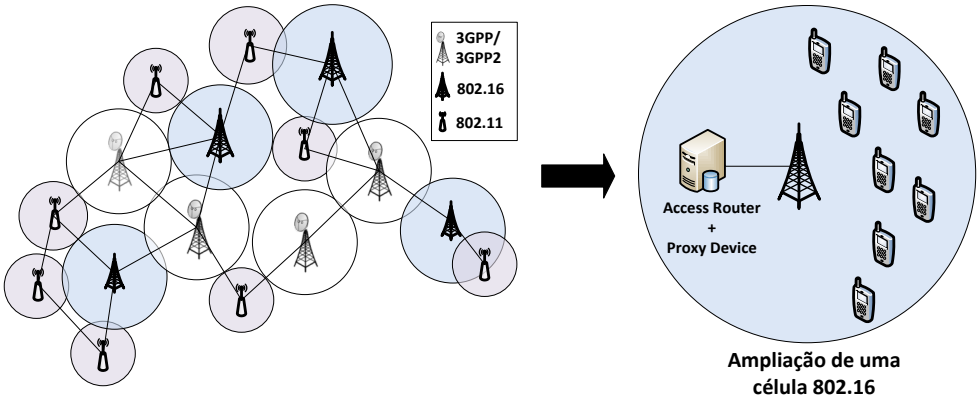


Figura 4.6: Exemplo de um cenário incluindo o *Proxy Device*.

Por definição, um PD pode trabalhar com ambas as versões 1 e 2 do protocolo *Multicast Listener Discovery* (MLD) (FENNER; HE; HABERMAN, 2006) (DEERING; FENNER; HABERMAN, 1999) (VIDA; COSTA, 2004). Para o nosso PD, decidimos trabalhar apenas com a versão 2 (MLDv2) desse protocolo, com o objetivo de habilitar, quando necessário, a criação de um grupo *multicast* com múltiplas fontes, ou seja, quando mudar para o modo *multicast*, o MN continuará dispondo dos seus serviços, mesmo que os tenha requisitado de múltiplos Nós Correspondentes (*Correspondent Nodes* - CN).

No PD, destacamos dois tipos de interface: uma ou mais interfaces *downstreams*, que gerencia a comunicação dos MNs com o PD ; e uma interface *upstream* única, responsável pela comunicação do PD com a Internet. Sabendo disso, implementamos a porção *router* do MLDv2 nas interfaces *downstreams*, que ficará responsável pela recepção dos *Unsolicited Reports* dos

MNs; e a porção *host* na interface *upstream*, que tem por objetivo estabelecer conexões com grupos *multicast*.

Considerando o MN, implementamos e integramos um módulo ao FMIPv6, o *Multicast Proxy Device User* (MPD *User*) (Figura 4.7). Esse módulo foi configurado para suportar o porção *host* do MLDv2 e é responsável por habilitar e desabilitar a recepção de pacotes de dados, enviados para um grupo *multicast* específico. Basicamente, o MPD *User* tem duas funções:

- Enviar para todos os CNs uma mensagem IGMP, incluindo o endereço do grupo *multicast* e com o objetivo de indicar que estes ingressem no grupo, funcionando como fontes; e
- Enviar uma mensagem *Unsolicited Report* para o PD, para que este habilite o recebimento de pacotes de dados destinado ao grupo *multicast* previamente estabelecido.

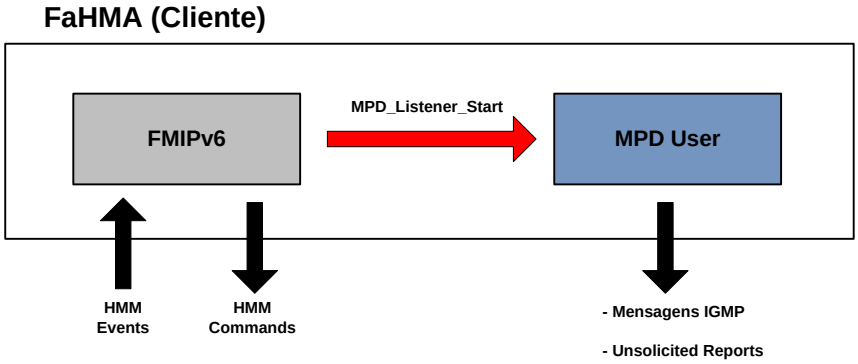


Figura 4.7: MPD *User*.

Quando o FMIPv6 decide mudar para modo *multicast*, ele envia uma mensagem para o MPD *User*: o *MPD_Listener_Start*. Uma vez recebida essa mensagem, o MPD User envia uma mensagem IGMP para todos os CNs em que o MN esteja utilizando algum serviço. Essa mensagem, que incluirá o endereço do grupo *multicast* a ser estabelecido, serve para indicar aos CNs para ingressar no grupo e funcionar como fontes, ou seja, os pacotes que antes eram enviados para o endereço do MN, agora serão enviados para o endereço *multicast* do grupo.

Ao fim do procedimento anterior, o MPD *User* usará a porção *host* do MLDv2 para enviar uma mensagem *Unsolicited Report* para o AR ao qual o MN está conectado, que será recebida por uma implementação do PD em execução. O PD em destaque, após receber essa mensagem, atualizará a *subscription* da interface do MN e, utilizando o algoritmo de fusão do MLDv2, também atualizará o *membership information database* (MID) (VIDA; COSTA, 2004). Em seguida, o PD ingressará no grupo *multicast*, tornando-se apto a replicar todos os pacotes direcionados para esse grupo e, em seguida, redirecioná-los para o MN.

Na seção anterior, definimos o Filtro de Rede (*Network Filter* - NF). Com base no NF, podemos prever se o MN pode ou não realizar o *handover* no modo preditivo. Portanto, considerando essa funcionalidade de decisão, podemos desenvolver, para cada modo (preditivo ou reativo), diferentes abordagens para tratar os procedimentos do *handover* na camada 3.

Neste trabalho, propomos duas formas de se implementar o FaHMA: o FaHMAv1 (FaHMA versão 1) e FaHMAv2 (FaHMA versão 2). Tais soluções se resumem em quando utilizar ou não o mecanismo de endereçamento *multicast*. Ambos, FaHMAv1 e FaHMAv2, apresentam o mesmo funcionamento quando executados no modo reativo. No modo preditivo, entretanto, apresentam abordagens bem diferentes para tratar o *handover* na camada 3.

O FaHMAv1 consiste em utilizar o *multicast* apenas no modo reativo. Com isso, teremos uma menor quantidade de mensagens trocadas quando o *handover* for realizado no modo preditivo, reduzindo o sobrecarga na rede e, consequentemente, aumentando a escalabilidade. No modo preditivo, o FaHMAv1 se utiliza da uma execução básica do FMIPv6, ou seja, sem utilizar os processos de armazenamento em *buffer* no PAR (*Previous Access Router*), que são os principais causadores da interrupção dos serviços. Entretanto, nesse caso, por estabelecer um túnel para o redirecionamento dos pacotes, essa solução fica susceptível aos atrasos neste.

Por outro lado, o FaHMAv2 propõe aplicar tal mecanismo nos dois modos. Este protocolo se baseia na ideia do MFMIPv6 (LAI; SHIEH; CHOU, 2009), trabalhando com *multicast* de forma a reduzir a perda de pacotes e a interrupção dos serviços. Entretanto, no modo preditivo, esse protocolo exige que mais mensagens sejam trocadas durante o *handover* em relação ao FaHMAv1, o que gera um maior sobrecarga na rede.

Nas próximas subseções, 4.3.1 e 4.3.2, mostraremos, para cada modo, a execução dessas soluções no momento do *handover* na camada 3 e apontaremos suas diferenças.

4.3.1 Modo Preditivo

No capítulo 2, mostramos que, durante a execução do FMIPv6 em meio a um processo de *handover*, ele utiliza mecanismos de armazenamento de pacotes em *buffer* tanto no NAR quanto no PAR. No NAR, esse processo de armazenamento é realizada para evitar a perda de pacotes durante o *handover* na camada 2. Por outro lado, o armazenamento realizado no PAR é implementado como um mecanismo de segurança, com o objetivo de reduzir a perda de pacotes caso não haja tempo de realizar o *handover* no modo preditivo.

Neste trabalho, entretanto, temos um mecanismo que é capaz de prever se o MN pode ou não realizar o *handover* no modo preditivo, chamado de NF (*Network Filter*). Logo, a motivação

por trás de se utilizar um mecanismo de armazenamento em *buffer* no PAR deixa de existir. Essa vantagem será utilizada pelo FaHMAv1, quando estiver realizando um *handover* na camada 3 no modo preditivo. Com isso, em vez de armazenar os pacotes, o PAR continuará os enviando para o MN, até o momento em que for estabelecido o túnel com o NAR, ou seja, quando esses pacotes forem redirecionados para este. O funcionamento FaHMAv1 no modo preditivo será descrito em seguida (Figura 4.8).

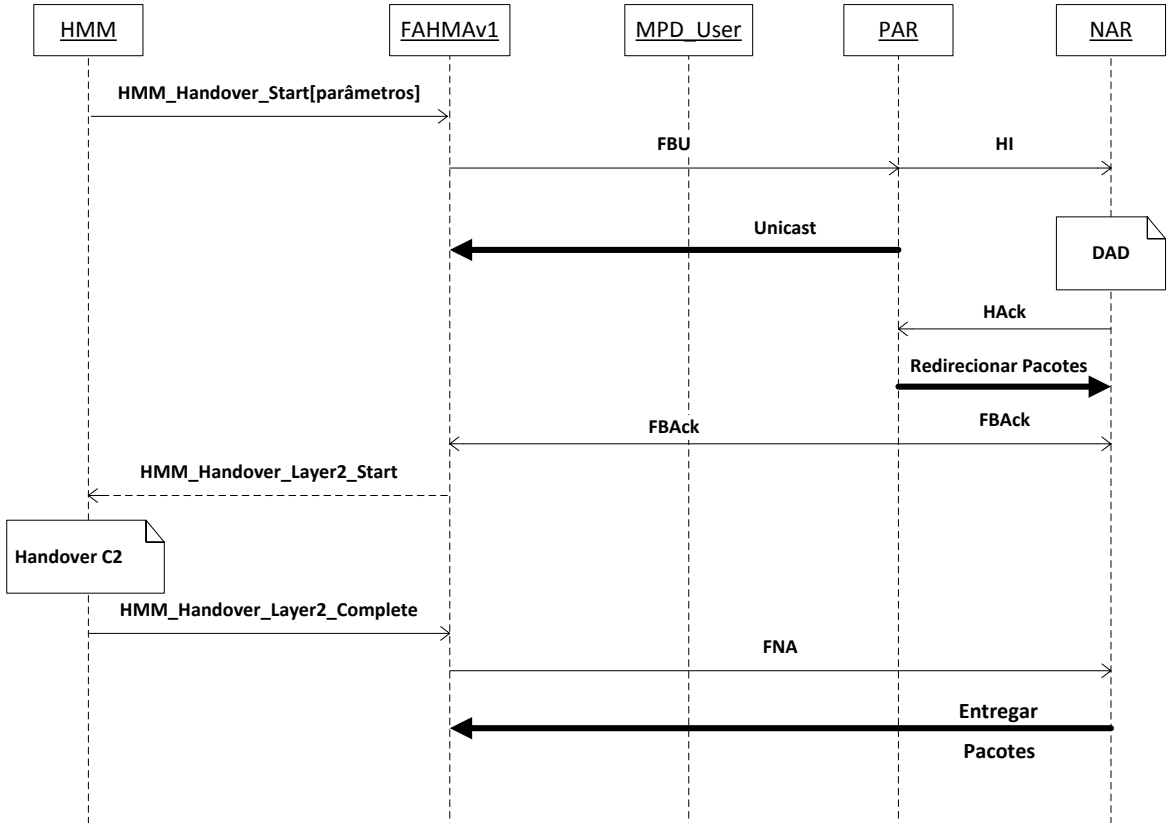


Figura 4.8: FaHMAv1: Modo Preditivo.

Ao receber o *HMM_Handover_Start*, o FaHMAv1 inicia o processo de *handover* no modo preditivo enviando um FBU para o PAR, como forma de antecipar o DAD. Ao receber um FBU, no lugar de interromper o envio de pacotes para o MN e iniciar o armazenamento em *buffer* (funcionamento normal do FMIPv6), o PAR continuará enviando-os. A partir daí, toda a execução se dará conforme a especificação do FMIPv6 (GUNDAVELLI; LEUNG; DEVARAPALLI, 2008) até o PAR receber um HAck do NAR. Nesse momento, o PAR irá interromper o envio de pacotes para o MN e passará a redirecioná-los para o túnel estabelecido com o NAR. Ao final, o PAR enviará as mensagens FBack e o restante do processo de *handover* (camada 2 e 3) poderá ser realizado como especificado na subseção 4.2.3.

Por outro lado, o FaHMAv2 baseia-se na ideia do MFMIPv6, definido em Lai et al. (2009), e propõe utilizar o mecanismo de endereçamento *multicast* para executar no modo preditivo.

Entretanto, o FaHMAv2 apresenta vantagens quando essas duas soluções são comparadas. Tais vantagens são apresentadas abaixo:

O MFMIPv6 utiliza o *Protocol Independent Multicast-Sparse Mode* (PIM-SM) (ESTRIN; FARINACCI; HELMY, 1998) como protocolo de roteamento *multicast*, gerando a necessidade que toda a região externa à rede também utilize esse protocolo. Ao contrário disso, o FaHMAv2 não inclui tal restrição por trabalhar com a ideia dos *Proxy Devices* (PDs). O uso de PDs tem o objetivo de permitir uma rede utilizar-se de serviços *multicast*, sem a necessidade de um protocolo de roteamento funcionando dentro dessa rede. Logo, FaHMAv2 pode trabalhar com quaisquer desses protocolos implementados na região externa a rede.

Outra vantagem consiste na redução do sobrecarga de mensagens trafegando na rede e o aumento da escalabilidade, em relação ao MFMIPv6. Por utilizar PDs em sua arquitetura, FaHMAv2 reduz o sobrecarga na rede, eliminando as mensagens que seriam trocadas pelo protocolo de roteamento *multicast*. Consequentemente, como a carga sobre o AR diminui, teremos um aumento de escalabilidade.

Por último, ambas as soluções, MFMIPv6 e FaHMAv2, armazenarão pacotes em *buffer* no NAR quando executam no modo preditivo; entretanto, no FaHMAv2, o NAR inicia o armazenamento de pacotes após o DAD (após recebimento da mensagem FBAck), diferente do que ocorre no MFMIPv6, onde o NAR inicia esse armazenamento antes do início do DAD (após receber uma mensagem HI). Portanto, o FaHMAv2 armazenará menos pacotes do que a outra solução, reduzindo assim o uso de espaço de armazenamento no NAR e, consequentemente, aumentando a escalabilidade, ao mesmo tempo em que também evita a perda de pacotes e diminui o *jitter*, pois reduz o enfileiramento de pacotes no NAR.

O funcionamento FaHMAv2 no modo preditivo será descrito a seguir (Figura 4.9).

- Ao receber o *HMM_Handover_Start*, o FaHMAv2 envia a mensagem *MPD_Listener_Start* para o MPD *User*;
- Em seguida, o MPD *User* enviará para todos os CNs uma mensagem IGMP, incluindo o endereço do grupo *multicast* e com o objetivo de indicar que estes ingressem no grupo, funcionando como fontes. Ele também enviará uma mensagem *Unsolicited Report* para o PD executando no PAR, para que aquele habilite o recebimento de pacotes de dados destinados ao grupo *multicast* previamente estabelecido. Nesse momento o MN muda para o modo *multicast* e passa a receber pacotes do grupo redirecionados pelo seu PAR;
- Paralelo ao passo anterior, o procedimento básico do FMIPv6 é realizado, como forma de antecipar o DAD. Ao final desse processo, ao receber um FBAck, o PD do NAR também

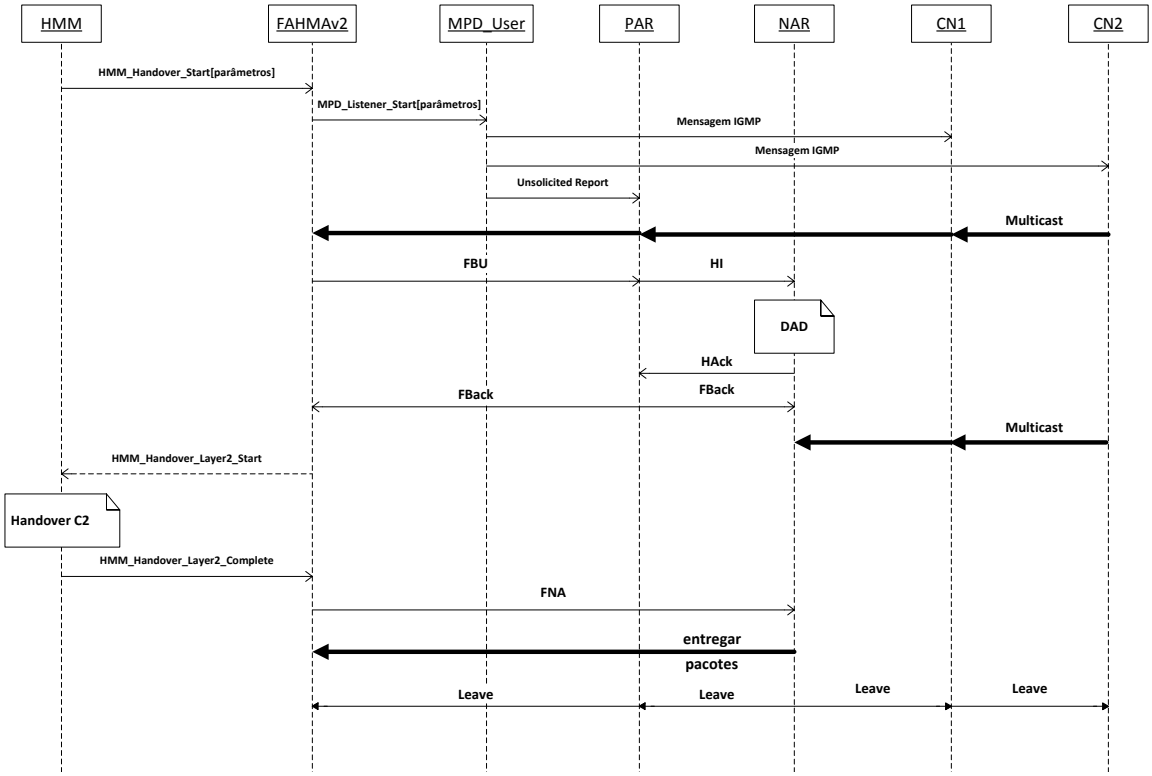


Figura 4.9: FaHMAv2: Modo Preditivo

habilitará o recebimento de pacotes de dados destinados ao grupo *multicast* previamente estabelecido; entretanto, em vez de redirecionar esses pacotes para o MN (que ainda não realizou o *handover* na camada 2), o NAR os armazena em *buffer*; e

- Ao fim do *handover* na camada 2 e do envio do FNA, o NAR enviará os pacotes armazenados para o MN (neste momento conectado) e iniciará o redirecionamento do fluxo dos dados (*Binding Update*). Por fim, o seu PD sinalizará a sua saída do grupo *multicast*, juntamente com o PAR.

4.3.2 Modo Reativo

Tanto o FaHMAv1 quanto o FaHMAv2 realizam o mesmo procedimento quando executados no modo reativo. Ambos utilizam o mecanismo de endereçamento *multicast* definido nas subseções anteriores. Portanto, nessa subseção, os trataremos apenas como o protocolo FaHMA.

Vimos anteriormente que o modo reativo ocorre quando o DAD é executado após o *handover* na camada 2. Além disso, o FMIPv6, quando realiza um *handover* no modo reativo, funciona como MIPv6 em termos da perda de pacotes e do atraso, este último podendo causar uma profunda interrupção dos serviços.

O FaHMA busca reduzir esses problemas, quando executado no modo reativo. Para tal, habilita o MN a continuar o recebimento de pacotes de dados, agora via *multicast*, logo após o *handover* na camada 2 e durante todo o processo de *handover* na camada 3. Com isso, teremos redução da perda de pacotes, pois continuaremos recebendo-os via *multicast*. O atraso do *handover* continuará o mesmo, mas a interrupção dos serviços, que é o atraso percebido pelo usuário, será reduzida.

A seguir, mostraremos o funcionamento do FaHMA no modo reativo (Figura 4.10).

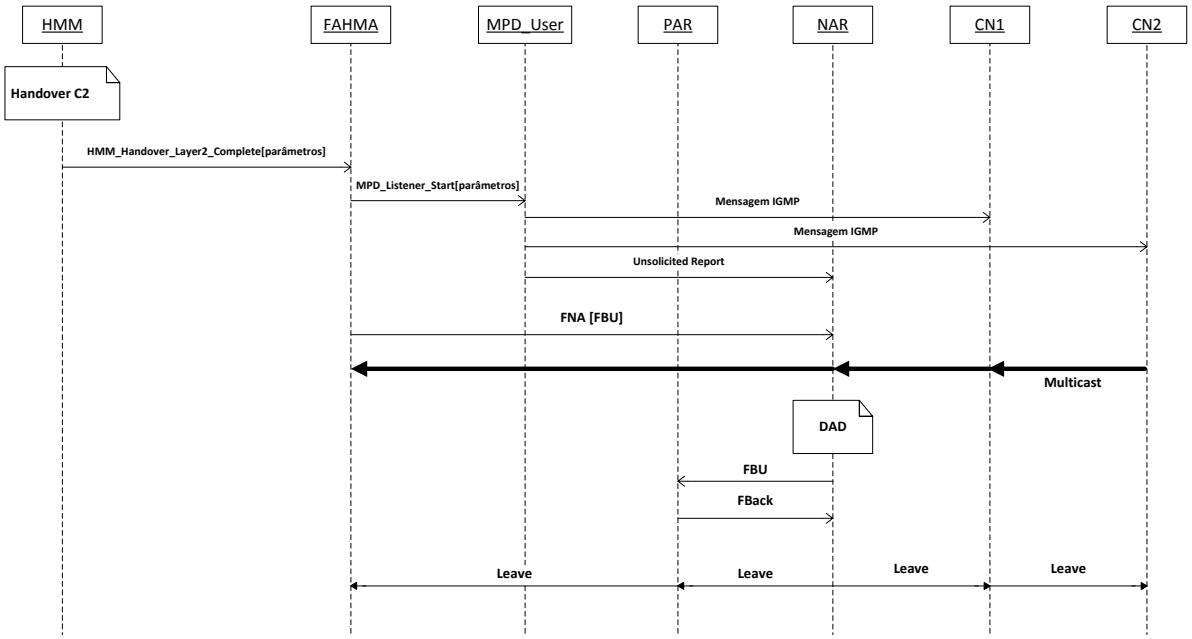


Figura 4.10: FaHMAv1 e FaHMAv2: Modo Reativo.

- Após o *handover* na camada 2, o HMM envia um *HMM_Handover_Layer2_Complete* para o FaHMA;
- Ao receber a mensagem anterior, o FaHMA envia a mensagem *MPD_Listener_Start* para o MPD *User*;
- Em seguida, o MPD *User* enviará a todos os CNs uma mensagem IGMP, incluindo o endereço do grupo *multicast* e com o objetivo de indicar que estes ingressem no grupo, funcionando como fontes. Ele também enviará uma mensagem *Unsolicited Report* para o PD executando no NAR, para que habilite o recebimento de pacotes de dados destinados ao grupo *multicast* previamente estabelecido. Nesse momento, o MN muda para o modo *multicast* e passa a receber pacotes do grupo redirecionados pelo seu NAR; e
- Paralelo ao passo anterior, o procedimento básico do FMIPv6 no modo reativo será realizado. Ao final desse processo, ao receber um FBAck, o NAR iniciará o processo de

redirecionamento do fluxo dos dados (*Binding Update*). Por fim, o seu PD sinalizará a sua saída do grupo *multicast*.

4.4 Conclusão

Neste capítulo, descrevemos a nossa proposta, que consiste em uma melhoria do processo de *Handover* Vertical, utilizando mobilidade IP e serviços definido no padrão IEEE 802.21. Nossa proposta se divide em duas partes: o módulo HMM e o protocolo FaHMA.

Inicialmente apresentamos o HMM, uma camada lógica que permite integrar o FMIPv6 com o *framework* MIH do IEEE 802.21 para criarmos a proposta-base de um esquema de *Handover* Vertical. Apresentamos também os seus subcomponentes: o PMM e NF. O PMM é a entidade do HMM responsável por intermediar a comunicação entre o FMIPv6 a o MIHF. Além disso, evita uma grande quantidade de mudanças na arquitetura do FMIPv6. Por outro lado, o NF é responsável por selecionar a melhor rede, dentre uma lista de redes identificadas na área de cobertura do MN, que torna possível realizar o *handover* no modo preditivo.

Em seguida, definimos um protocolo que será responsável por melhorar o *handover* na camada 3. O FaHMA estende o FMIPv6 a partir de sua integração com um mecanismo de endereçamento *multicast*. Para desenvolver tal mecanismo, partimos da ideia apresentada em Lai et al (2009) e a adaptamos para trabalhar sobre a arquitetura dos PDs. Com isso, exploramos as vantagens das duas técnicas, criando um outro mecanismo integrado ao FMIPv6, que reduz a interrupção de serviços e o impacto sobre a escalabilidade da rede, através de uma menor sobrecarga de mensagens.

Utilizando-se dessas soluções, criamos um esquema de *Handover* Vertical que evita a perda de pacotes e a interrupção dos serviços. No próximo capítulo, implementaremos nossa proposta, como forma de realizar simulações. Com base nos resultados obtidos, faremos uma análise de desempenho para mostrar que a nossa proposta alcança o seu objetivo.

5 *Análise de Desempenho*

5.1 Simulações

De acordo com Jain (1991), existem três técnicas para realizar uma análise de desempenho de algum sistema: a simulação, que permite a avaliação de um sistema computacional em um ambiente virtual, com o objetivo de alcançar resultados que se aproximem ao máximo aos obtidos em um ambiente real; a modelagem analítica, que consiste em um modelo matemático com algumas equações, incluindo informações que devem ser analisadas; e, por último, a medição, que visa a avaliar um sistema em um ambiente real propriamente dito. Existem, porém, algumas considerações que ajudam a decidir pela escolha da melhor técnica para determinados sistemas. Tais considerações podem ser visualizadas na Tabela 5.1.

Tabela 5.1: Critérios para selecionar técnicas de análise de desempenho (JAIN, 1991).

Critério	**Simulação**	**Modelo Analítico**	**Medição**
1. Estágio do Trabalho	Qualquer	Qualquer	A partir de protótipo construído
2. Tempo Requerido	Médio	Pequeno	Variável
3. Ferramentas	Linguagem de Programação	Analistas	Instrumentação
4. Precisão	Moderada	Baixa	Variável
5. Avaliação de Trade-off	Moderado	Fácil	Difícil
6. Custo	Médio	Baixo	Alto
7. Negociabilidade	Médio	Baixo	Alto

Escolhemos as simulações ao invés das outras como a técnica para avaliar a nossa proposta. Consideramos os seguintes pontos citados na Tabela 5.1 para essa decisão: a impossibilidade de trabalhar com medições, pela falta de recursos disponíveis (Custo) para a implantação de um *testbed*, que consistiria em um ambiente real; e simulações são mais precisas do que uma modelagem analítica (Precisão). Em geral, a modelagem analítica requer muitas simplificações e suposições, enquanto que, nas simulações podemos incorporar mais detalhes e fazer menos suposições, tornando-se geralmente uma abordagem mais próxima da realidade (JAIN, 1991).

De todos os trabalhos relacionados estudados (MUSSABBIR et al., 2007)(HUANG; WU, 2009)(KIM et al., 2008)(AN et al., 2006), nenhum deles trabalha com simulações que verifi-

quem a significância estatística (JAIN, 1991). Ao invés disso, os autores realizam a análise de desempenho utilizando dois dos métodos citados: simulações e modelagem analítica. Tais simulações são realizadas sem a devida repetição dos experimentos e, consequentemente, sem o cálculo dos intervalos de confiança. Contudo, como forma de suprimir essa abordagem não adequada do método e validar os trabalhos, uma análise matemática é utilizada como complementação, que consiste em uma abordagem mais simples e menos precisa que a simulação.

Como apresentado no capítulo 4, neste trabalho decidimos utilizar simulações para realizar a análise de desempenho. Além disso, pretendemos incorporar a abordagem das repetições dos experimentos de forma a considerar a aleatoriedade do ambiente e, consequentemente, produzir gráficos que incluam intervalos de confiança em sua constituição. Desta forma, podemos utilizar apenas as simulações para garantir a corretude dos resultados que esperamos alcançar.

5.2 Caracterização das Simulações

5.2.1 Ferramentas e Implementações

Após a escolha da simulação como nossa técnica de avaliação da proposta, o próximo passo foi a escolha da ferramenta de simulação. Considerando que os melhores simuladores do mercado são pagos e que os preços das licenças estão fora da realidade financeira deste trabalho, foi feita a opção por um simulador gratuito e de código aberto, o Network Simulator 2 (NS-2) (MCCANNE; FLOYD, 2006). Esse simulador é bastante utilizado no meio acadêmico e surge em muitas das melhores publicações.

Utilizamos a versão 2.29 do NS-2, pois é a versão para a qual o módulo NIST é suportado. O pacote de mobilidade NIST (NIST, 2007), fornece um suporte para *handovers* em redes heterogêneas, desde uma implementação do framework MIH definido no padrão IEEE 802.21. Esse pacote não possui, porém, uma implementação completa do MIH, considerando apenas uma versão antiga deste, proposta em um *draft* de dezembro de 2006 (IEEE, 2006). Esse pacote também fornece uma implementação genérica do MIPv6, como forma de realizar o gerenciamento da mobilidade na camada 3.

Escolhidas as ferramentas, devemos executar as implementações necessárias para conseguir realizar as simulações do nosso trabalho. Portanto, organizamos o nosso processo de implementação em quatro fases.

- Atualizar o *framework* MIH para a versão mais atual (IEEE, 2009);

- Implementar o HMM (*Handover Management Module*);
- Estender a implementação genérica do MIPv6 para o FMIPv6; e
- Implementar as duas versões do FaHMA.

Vale ressaltar que a implementação do HMM considera apenas uma versão reduzida deste, pois ainda não foi definida uma arquitetura para o módulo NF (relacionada como um dos trabalhos futuros), responsável pela seleção da próxima rede na qual será estabelecida uma conexão. Como veremos mais a frente, nosso cenário incluirá apenas duas redes, portanto, o MN já terá disponíveis as informações de sua nova rede.

Ao fim dessas fases, podemos então iniciar a simulação, iniciando com a implementação e configuração do cenário onde serão rodados os experimentos e, ao final, realizar a análise de desempenho a partir dos resultados obtidos.

5.2.2 Cenário

O ambiente de simulação pode ser visualizado na Figura 5.1.

Temos uma rede simplificada com os seguintes componentes:

- 2 Roteadores Ethernet: router0 e router1;
- 2 Estações Base (*Access Router* - AR): AR1 (WiFi) e AR2 (WiMAX); e
- 1 Nó Móvel (*Mobile Node* - MN) com duas interfaces: iface1 (WiFi) e iface2 (WiMAX).

O router0 é um *host* fixo que fará o papel do CN (*Correspondent Node*) em nossas simulações, ou seja, esse nó ficará responsável por enviar os pacotes de dados em direção ao MN. O router1 ficará responsável por mediar as comunicações entre o CN e os ARs das redes sem fio, tendo o papel de simular o núcleo da rede (Internet), gerando as suas variações de estado. As estações base (AR1 e AR2) realizarão, além do encaminhamento das mensagens para o MN, a troca de pacotes de controle para viabilizar o *handover* nas diferentes soluções apresentadas. Além disso, o AR1 funcionará como um ponto de acesso WiFi, enquanto o AR2 funcionará como um ponto de acesso WiMAX. Por fim, o MN terá o trabalho de receber os pacotes de dados do CN e realizar todas os procedimentos necessários para o sucesso do *handover*.

A ideia por trás deste cenário é fazer com que o MN se desloque da região de cobertura do AR2 para a cobertura do AR1, forçando-o a realizar um *Handover* Vertical. Todos os experimentos são realizados considerando apenas o *handover* realizado entre essas duas tecnologias

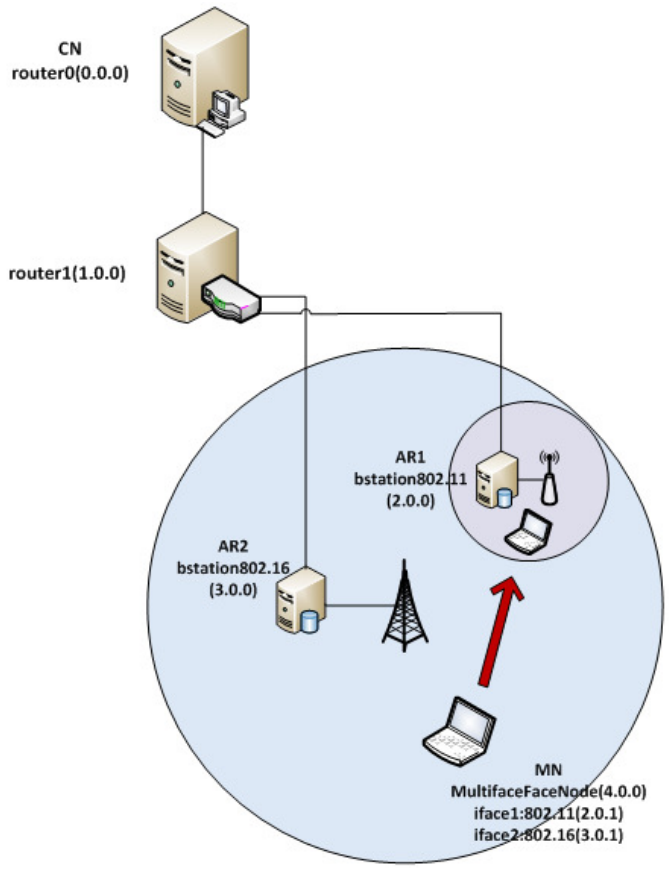


Figura 5.1: Cenário de Simulações.

e sempre na mesma ordem: da rede WiMAX (AR2) para a WiFi (AR1). Isso é necessário, pelo seguinte motivo: Se realizássemos experimentos considerando outras tecnologias, ou se apenas invertêssemos o sentido do deslocamento, obteríamos resultados diferentes, que seriam provocados pelas características dos meios físicos envolvidos, e não pelo comportamento da nossa proposta, o que dificultaria as comparações.

5.2.3 Modelo de Mobilidade

Um modelo de simulação provê uma forma fácil de predizer o desempenho ou comparar diversas alternativas (JAIN, 1991). Entretanto, esse modelo geralmente falha, ou seja, não produz resultados úteis ou produz resultados que não representam a realidade. Uma das principais causas é falta da repetições dos experimentos, onde para cada repetição, valores aleatórios são aplicados como forma de aproximar tais experimentos aos feitos em ambientes reais.

Como citado na seção 5.1, neste trabalho foi realizado simulações considerando a repetição

dos experimentos e, consequentemente, abordadando a verificação da significância estatística. Entretanto, uma questão a ser resolvida foi a de como inserir valores aleatórios ao ambiente de simulação, de forma a gerar resultados diferentes durante as repetições, possibilitando o cálculo dos intervalos de confiança. De forma a solucionar o problema da aleatoriedade, decidimos trabalhar com o modelo de mobilidade *Random WayPoint* (HYYTIä; VIRTAMO, 2007). Entretanto, em nosso caso, nos reservamos a considerar apenas uma abordagem reduzida deste.

O cenário, no qual se insere o *Random WayPoint*, consiste em vários MNs, movendo-se aleatoriamente por entre vários pontos de acesso. Em nossa abordagem, temos apenas dois pontos de acesso, sendo que um implementa a tecnologia WiMAX e o outro a WiFi (como mostrado na subseção anterior); e vários MNs movendo-se apenas no sentido do ponto WiMAX para o WiFi, ou seja, cada MN estará inicialmente na área de cobertura do WiMAX e, em um determinado momento, se deslocará para a área de cobertura do WiFi. Apesar disso, mesmo o nosso modelo sendo reduzido em relação ao *Random WayPoint*, consideramos as mesmas variáveis presentes na abordagem completa:

- Ponto de Chegada;
- Tempo de Parada (duração da permanência na área de cobertura); e
- Velocidade do Nó.

Portanto, em nosso modelo, teremos um MN partindo de um determinado ponto do cenário (dentro da rede WiMAX e fora da WiFi), em um determinado instante tempo, com uma determinada velocidade (velocidade do nó) e com um destino aleatório (ponto de chegada), dentro do raio de cobertura da rede WiFi. O tempo de parada será dado pela quantidade de tempo que um MN espera para se deslocar de um rede para a outra, a partir do início da simulação. Podemos visualizar essa dinâmica na Figura 5.2.

5.2.4 Parâmetros e Métricas Analisados

Na Tabela 5.2, descrevemos os parâmetros com seus respectivos valores, que são considerados em nosso ambiente de simulação. Além disso, também são especificados os parâmetros que variam aleatoriamente a cada rodada dos experimentos; e os fatores, com seus respectivos conjuntos de níveis, seguidos do seu valor *default* delimitado por colchetes.

Primeiramente, justificaremos os parâmetros de valor fixo. Em relação aos parâmetros que descrevem as tecnologias de acesso (tipo, dimensão, posição e raio), buscamos definir valores

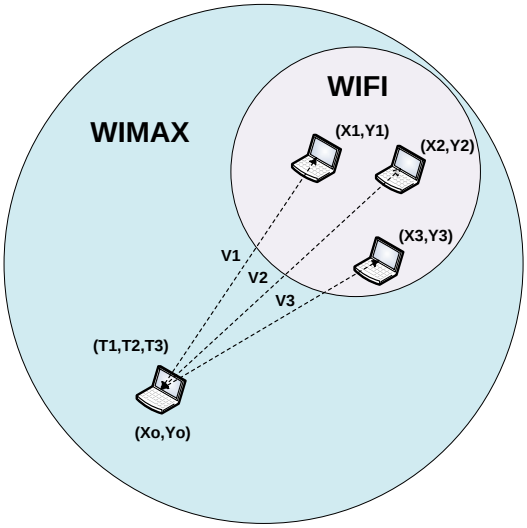


Figura 5.2: Modelo de Mobilidade.

que representem fielmente as características destas, como forma de tornar os cenários simulados mais realistas. Além disso, em nossos experimentos, o tipo e a vazão dos dados trafegados na rede são fixos, pois não influenciam no resultado das análises. Nesse caso, as aplicações utilizadas pelos CNs serão apenas simples geradores de tráfego CBR. Esses pacotes serão transmitidos em intervalos de tempo constantes, 1 pacote de 200 *bytes* a cada 0.03 s. Além disso, consideramos um tempo de simulação necessário para observarmos todo o processo do *handover*, 30 segundos. Finalmente, como forma de avaliar algumas situações de perda de pacotes, limitamos o tamanho do *buffer* nos ARs para que possam armazenar apenas 50 pacotes por MN.

Foram variados de forma aleatória tanto os parâmetros necessários para o nosso modelo de mobilidade (ponto de chegada, início do deslocamento e velocidade do MN) quanto os valores de atraso e largura de banda dos enlaces entre os roteadores (router0 e router1) e entre o router1 e as estações base (AR1 e AR2), como forma de simular as variações existentes desses parâmetros em ambientes reais.

Os seguintes fatores: número de nós móveis, atraso DAD e atraso no enlace entre os ARs, tiveram os seus níveis definidos de acordo com as experiências obtidas dos trabalhos estudados (LAI; SHIEH; CHOU, 2009)(MUSSABBIR et al., 2007)(HUANG; WU, 2009)(KIM et al., 2008)(AN et al., 2006) e, além disso, ainda foram adaptados (valores mais realísticos) para permitir uma melhor análise. Finalmente, o fator modo de operação dos protocolos, define o tipo execução que os protocolos utilizados na simulação realizaram. Tais fatores têm o objetivo de realizar experimentos sob diferentes condições de operação.

Após a realização das simulações, obtemos os resultados e, a partir destes, avaliamos o impacto dos seguintes fatores:

Tabela 5.2: Lista de Parâmetros.

Parâmetro	**Valor ou Níveis**
Tecnologia AR1	WLAN 802.11
Tecnologia AR2	WiMAX 802.16
Dimensão Topográfica	x=2000, y=2000
Posição AR1	x=500, y=1000
Posição AR2	x=1000,y=1000
Raio AR1	20 m
Raio AR2	1000 m
Ponto do Saída do Nó Móvel	x=500 y=1000
Ponto do Chegada do Nó Móvel	Aleatório
Início do deslocamento do Nó Móvel	Aleatório
Fonte de Dados	CBR sobre UDP
Tamanho do Pacote de Dados	200 bytes
Vazão	1 pct. (200 bytes) a cada 0.03 s (6,67 Mbps)
Tempo de Simulação	30 s
Capacidade do *buffer* nos ARs por MN	50 pacotes
Número de Nós Móveis	1 - 20 [1] (Passo - 1)
Atraso DAD	0.25 - 3 s [1 s] (Passo - 0,25 s)
Atraso no enlace entre os ARs	50 - 500 ms [100 ms] (Passo - 25 ms)
Modo de Operação dos Protocolos	Modo preditivo e reativo
Atraso no enlace entre os roteadores	Aleatório
Largura de Banda em todos os enlaces cabeados	Aleatório
Velocidade do Nó Móvel	Aleatório

- Atraso do DAD;
- Atraso no Enlace entre os ARs; e
- Número de Nós Móveis.

Sobre as seguintes métricas:

- Atraso do *Handover*;
- Taxa de Perda de Pacotes;
- Atraso Fim a Fim;
- *Jitter*; e
- Número de Pacotes Recebidos.

Dentre os fatores, o atraso do DAD consiste no tempo de duração do DAD, que é o fator de maior impacto no atraso do *handover* e, consequentemente, na interrupção dos serviços. O atraso no enlace entre os ARs é considerado em razão do túnel estabelecido entre estes durante

o processo de *handover*. Foi citado anteriormente que, no FaHMAv2, esse túnel não é estabelecido, o que pode ser um ganho no caso em que esse atraso é muito alto. Finalmente, a variação do número de nós móveis nos permite avaliar o impacto da nossa solução sobre o número de pacotes de controle trocados na rede e, consequentemente, mensurar o impact da nossa solução sobre a escalabilidade (se piora ou não).

Em relação às métricas, consideramos as variáveis de maior impacto para confirmar a efetividade de um esquema de *handover* de acordo com Lai et al (2009): o atraso do *handover* e a perda de pacotes. No nosso caso, consideramos o atraso do *handover* como a própria interrupção do serviços, pois o que nos interessa é o atraso percebido pelo usuário e não a duração de todo o processo de *handover*. Esse atraso consiste na diferença entre o instante de tempo em que o MN recebe o primeiro pacote do NAR e o instante de tempo em que recebe o último pacote do PAR. Além disso, a taxa de perda de pacotes consiste na quantidade de pacotes perdidos dividido pelo número de pacotes enviados, durante o *handover*.

Adicionalmente, também avaliamos o impacto dos fatores sobre o atraso fim a fim e o *jitter*, como forma de avaliar a qualidade com que os dados são consumidos pelo usuário final. O atraso fim a fim indica a quantidade de tempo que um pacote leva para chegar ao MN, desde o seu envio pelo CN. O *jitter*, que também é conhecido como variação do atraso, indica a variação do atraso de pacotes de dados sucessivos. Vale ressaltar que quanto maior essa variação, menor será a regularidade na entrega dos pacotes de dados, o que aumenta o retardo perceptível pelo usuário final.

O atraso fim a fim e o *jitter* serão utilizados para avaliar o comportamento das nossas soluções, quando utilizadas com aplicações multimídias. Diferente das aplicações elásticas, como *e-mail* e navegação *web*, que são tolerantes ao atraso, as aplicações multimídia são sensíveis ao atraso fim a fim e ao *jitter*. Essas aplicações, todavia, são tolerantes às perdas de dados, suportando pequenas perturbações na recepção destes.

Por fim, o número de pacotes recebidos corresponde à contabilização do número médio de pacotes de controle utilizados para manter o funcionamento do *handover*. Vale ressaltar que essa métrica será avaliada apenas quando variar o número de nós móveis e serve, exclusivamente, para avaliar a nossa solução de forma a mensurar o impacto desta sobre a escalabilidade da rede (se piora ou não).

5.2.5 Caracterização dos Experimentos

No início do Capítulo 4, mostramos que a nossa proposta é baseada na seguinte hipótese: se integrarmos o protocolo FMIPv6 com o *framework* MIH definido pelo padrão IEEE 802.21, com o objetivo de habilitar a mobilidade IP em redes heterogêneas, e estendermos o FMIPv6 aplicando um esquema de endereçamento *multicast*, podemos obter um esquema de *handover* transparente sobre redes heterogêneas que reduz a perda de pacotes, o atraso do *handover* e, além disso, a interrupção dos serviços (atraso percebido pelo usuário final) causada pela atividade Detecção de Endereços Duplicados (*Duplication Address Detection* - DAD), sem grandes impactos sobre a escalabilidade da rede. Portanto, em nossos experimentos, esperamos verificar se essa hipótese é alcançada. Para tal, comparamos as nossas soluções (FaHMAv1 e FaHMAv2) com o protocolo FMIPv6.

A análise de desempenho do HMM não será estudada neste trabalho, entretanto, devemos utilizá-lo para viabilizar os nossos experimentos em ambientes simulados heterogêneos.

Durante as simulações, avaliaremos cada combinação (fator/métrica) para cada um dos cenários abaixo:

1. FMIPv6 + HMM - A implementação do FMIPv6 básico, sobre a camada lógica HMM;
2. FaHMAv1 + HMM - A implementação do protocolo FaHMA de acordo com a proposta 1, sobre a camada lógica HMM;
3. FaHMAv2 + HMM - A implementação do protocolo FaHMA de acordo com a proposta 2, sobre a camada lógica HMM.

Ressaltamos que, neste trabalho, não incluímos o MFMIPv6 (LAI; SHIEH; CHOU, 2009) em nossos experimentos. Tal escolha foi realizada em razão do cenário para o qual foi proposto este protocolo. O MFMIPv6 foi projetado para operar sobre redes homogêneas, redes que incluem a mesma tecnologia de acesso em seus Pontos de Acesso (*Access Point* - AP), enquanto que, os outros protocolos (FMIPv6 e FaHMA) foram adaptados ao HMM como forma de operarem em redes heterogêneas. Para adaptarmos o MFMIPv6 ao HMM devemos realizar modificações em sua arquitetura, estabelecendo uma comunicação e uma sincronização entre essas duas entidades, o que gerariam resultados diferentes da proposta básica em possíveis experimentos.

Cada um destes cenários será avaliado tanto no seu funcionamento no modo preditivo quanto no reativo. A seguir, apresentaremos os gráficos gerados após a realização das simula-

ções, organizados pelos fatores escolhidos e, para cada gráfico, discutiremos sobre os resultados alcançados.

Os valores ilustrados nos gráficos são valores médios de 200 amostras obtidas por meio de repetições das simulações, com 95% de confiança. De acordo com Jain (1991), o procedimento utilizado para encontrar o intervalo de confiança de um experimento serve apenas quando a quantidade de amostras é grande, ou seja, maior ou igual a 30. Além disso, quanto maior for quantidade amostras, menor será o intervalo de confiança. Portanto, a escolha de 200 amostras justifica-se pela qualidade dos intervalos de confiança obtidos, permitindo uma melhor análise dos resultados.

Uma última consideração consiste na representação do FaHMAv1 e FaHMAv2 nos gráficos que mostram os resultados do funcionamento dessas soluções no modo reativo. Como ambas as soluções possuem o mesmo funcionamento neste modo, preferimos tratá-las apenas por um protocolo, o FaHMA.

5.3 Análise dos Resultados

5.3.1 Atraso do DAD

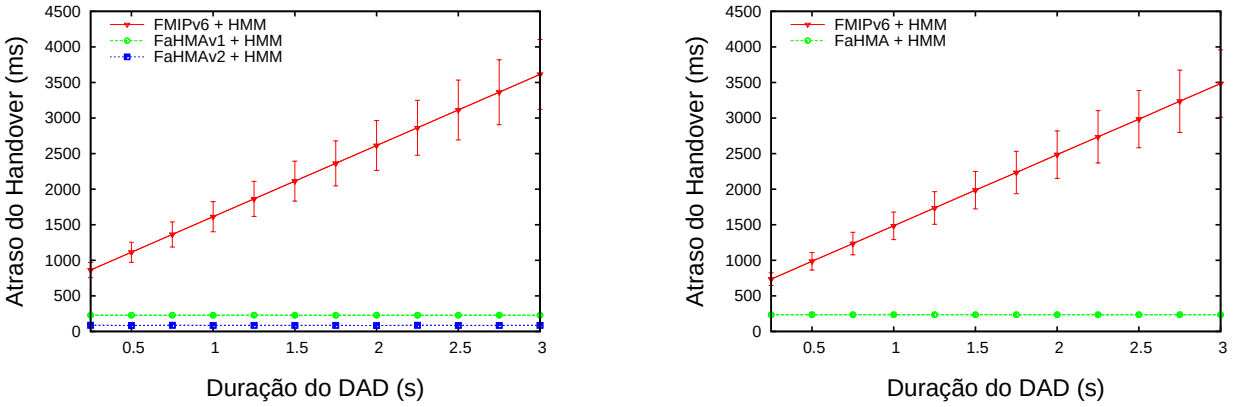
Nesta subseção, avaliaremos o impacto da variação do atraso DAD sobre todas a métricas definidas na seção anterior: Atraso do *Handover*, Perda de Pacotes, Atraso Fim a Fim e o *Jitter*. Fizemos o atraso do DAD assumir os seguintes valores: 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2, 2.25, 2.5, 2.75 e 3 (tempos em segundos).

Atraso do *Handover*

Nesse tópico, avaliamos o impacto do valor do atraso do DAD sobre o atraso do *handover*. Nesse caso, também considerado como o tempo da interrupção dos serviços.

No modo preditivo, vimos que o FaHMAv1 mantém a entrega de pacotes de dados para o MN, via *unicast*, durante o DAD, em vez de armazená-los em *buffer*, como ocorre no FMIPv6. Por outro lado, o FaHMAv2 estabelece um grupo *multicast* para que o MN também continue recebendo esses pacote durante o DAD, mas via *multicast*. Portanto, ambas as soluções atingem o mesmo objetivo, que consiste em manter o recebimento desses pacotes durante a antecipação do DAD, atenuando, assim, a sua influência negativa no atraso do *handover*.

Na Figura 5.3a, verificamos a situação descrita no parágrafo anterior. Com o aumento



(a) Modo Preditivo

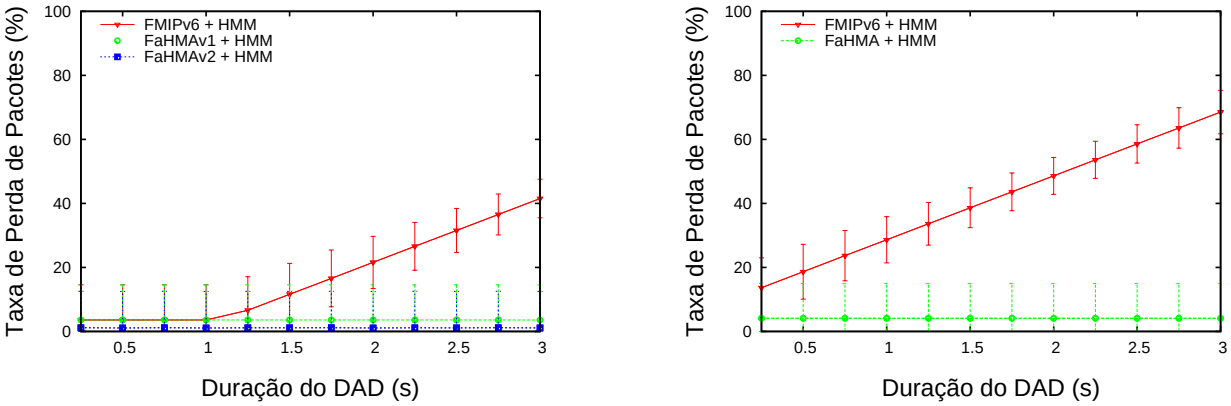
(b) Modo Reativo

Figura 5.3: Impacto do Atraso do DAD sobre o Atraso do *Handover*.

do atraso do DAD, ambas as versões do FaHMA mantêm aproximadamente sempre o mesma média do atraso do *handover*: uma média de 228 ms no FaHMAv1 e 86 ms para o FaHMAv2. Por outro lado, no FMIPv6, o atraso do *handover* cresce linearmente com o aumento do tempo DAD, iniciando com aproximadamente 863 ms, valor já superior ao atraso das duas versões do FaHMA, e terminando com aproximadamente 3613 ms.

Também vale ressaltar que, no modo preditivo, o FaHMAv2 possui um desempenho melhor do que o FaHMAv1. Isso ocorre em razão do estabelecimento do túnel entre o PAR e o NAR durante a execução do FaHMAv1. Ao receber um HAck do NAR, o PAR irá interromper o envio de pacotes para o MN e passará a redirecioná-los para o túnel estabelecido com o NAR. No FaHMAv2, o MN continuará a receber esses pacotes até o momento em que ocorrer o *handover* na camada 2, pois não existe o estabelecimento de um túnel entre os ARs, sendo os dados enviados diretamente para o NAR pelo CN, via *multicast*. Portanto, na versão 2, o MN receberá dados do PAR durante mais tempo do que na versão 1.

No modo reativo (Figura 5.3b), temos que o FaHMA mantém o objetivo de evitar que o atraso DAD influencie negativamente no atraso do *handover*, fazendo com que o MN receba o seus pacotes de dados durante a configuração de sua conexão com o NAR, com base no estabelecimento de um grupo *multicast*. Nesse caso, podemos visualizar na Figura 5.3b que, na execução do FaHMA, o atraso do *handover* se mantém aproximadamente sempre com a mesma média: 235 ms. Em contrapartida, o atraso de *handover* provocado pelo FMIPv6 tende a aumentar linearmente com o aumento do atraso DAD, iniciando com aproximadamente 734 ms, valor já superior ao atraso do FaHMA, e terminando com aproximadamente 3485 ms.



(a) Modo Preditivo

(b) Modo Reativo

Figura 5.4: Impacto do Atraso do DAD sobre a Perda de Pacotes.

Taxa de Perda de Pacotes

Neste tópico, avaliamos o impacto do valor do atraso do DAD sobre a perda de pacotes.

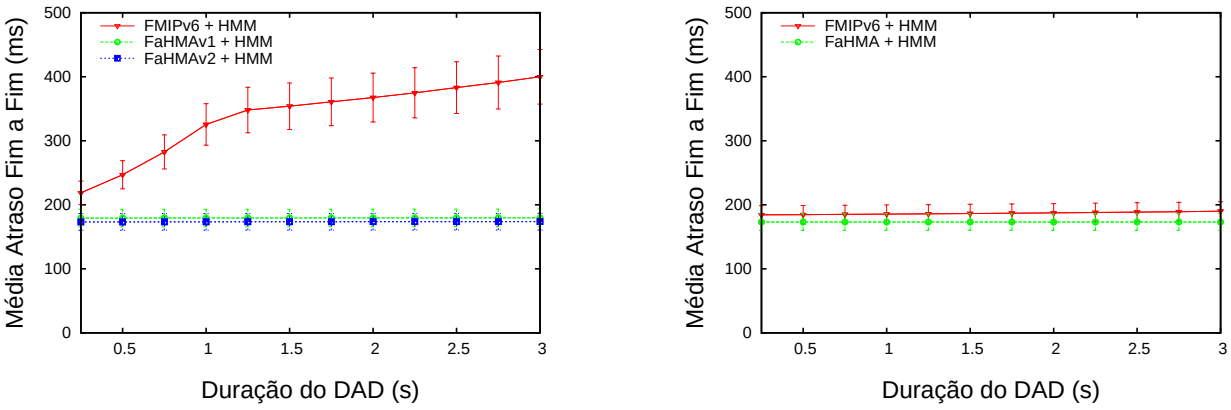
Considera-se que, quando executadas no modo preditivo, as três soluções utilizam mecanismos de armazenamento em *buffer*, com o objetivo de reduzir a perda de pacotes durante o processo de *handover*. Entretanto, as duas versões do FaHMA implementam esse mecanismo apenas no NAR, enquanto o FMIPv6 implementa tanto no NAR quanto no PAR, pois inicia o armazenamento de pacotes desde a indicação do MN de realizar o *handover* na camada 3. No FaHMA, o armazenamento no NAR inicia apenas após a realização do DAD. Portanto, a quantidade de pacotes armazenados no NAR será maior quando for utilizado o FMIPv6 do que quando usadas as duas versões do FaHMA, pois no NAR serão armazenados tanto os novos pacotes que forem chegando quanto os armazenados no PAR. Como forma de analisar essa situação, decidiu-se limitar o tamanho do *buffer* no NAR para esse experimento (50 pacotes por MN).

Na Figura 5.4a, verificamos a situação anterior. No intervalo de 0,25 a 2 s, nenhuma das três soluções se mostra significantemente superior a outra, pois os seus intervalos de confiança se cruzam. Entretanto, no intervalo 1 a 2 s, verificamos que FMIPv6 já apresenta situações em que a perda de pacotes é maior em relação às nossas soluções e isso decorre da situação apresentada anteriormente. No FMIPv6, quanto maior o atraso DAD, maior a quantidade de pacotes armazenados no NAR. Por isso, a partir de 1 s de atraso, a quantidade de pacotes armazenados no NAR ultrapassa o tamanho do *buffer*, provocando a perda de alguns deles. Em contrapartida, em nossas soluções, a quantidade de pacotes armazenados no NAR não depende

do atraso DAD. Logo, a quantidade de pacotes perdidos se manterá, em média, sempre constante com o aumento desse atraso. Com isso, para um tempo de DAD acima de 2 s, os intervalos de confiança entre o FMIPv6 e das duas versões do FaHMA não se cruzam mais, tornando as nossas soluções significativamente melhores do que o FMIPv6.

No modo reativo, nenhuma das soluções utiliza esses mecanismos de armazenamento em *buffer*. Entretanto, no FMIPv6, durante a realização do DAD, o MN não receberá pacotes, pois a troca de conexões ainda não terá sido realizada completamente. Logo, quanto maior o atraso do DAD, maior será a perda de pacotes do dispositivo. Por outro lado, o FaHMA estabelece um grupo *multicast* para que o MN continue a receber os seus pacotes de dados durante todo o processo de *handover*, via *multicast*. Na Figura 5.4b, podemos verificar a situação anterior. No FaHMA, a taxa da perda de pacotes se mantém aproximadamente constante: 4%. Por outro lado, o FMIPv6 inicia com taxa de 13,67% em 0.25 s de DAD, valor já superior ao do FaHMA, e cresce linearmente, chegando a uma taxa de 68,54% com 3 s de DAD. Entretando, mesmo sendo superior, o FaHMA somente se torna significativamente melhor do que o FMIPv6 a partir do tempo de 0,75s do atraso DAD, onde os intervalos de confiança não mais se cruzam.

Atraso Fim a Fim



(a) Modo Preditivo

(b) Modo Reativo

Figura 5.5: Impacto do Atraso do DAD sobre o Atraso Fim a Fim.

Nesse tópico, avaliamos o impacto do valor do atraso do DAD sobre o atraso fim a fim. Daqui em diante, pretendemos avaliar o impacto do atraso do DAD no comportamento das nossas soluções, quando utilizadas para aplicações multimídias. No próximo tópico, analisaremos o *jitter*.

Vimos no tópico anterior que, quando executadas no modo preditivo, as três soluções utilizam mecanismos de armazenamento em *buffer* com o objetivo de reduzir a perda de pacotes durante o processo de *handover*. Quanto maior o tempo que um pacote passar armazenado em um *buffer*, maior será o seu atraso fim a fim. A quantidade de pacotes armazenados também influencia na média final dos atrasos fim a fim, pois quanto maior o número de pacote armazenados, maior será essa média em virtude do maior atraso sofrido.

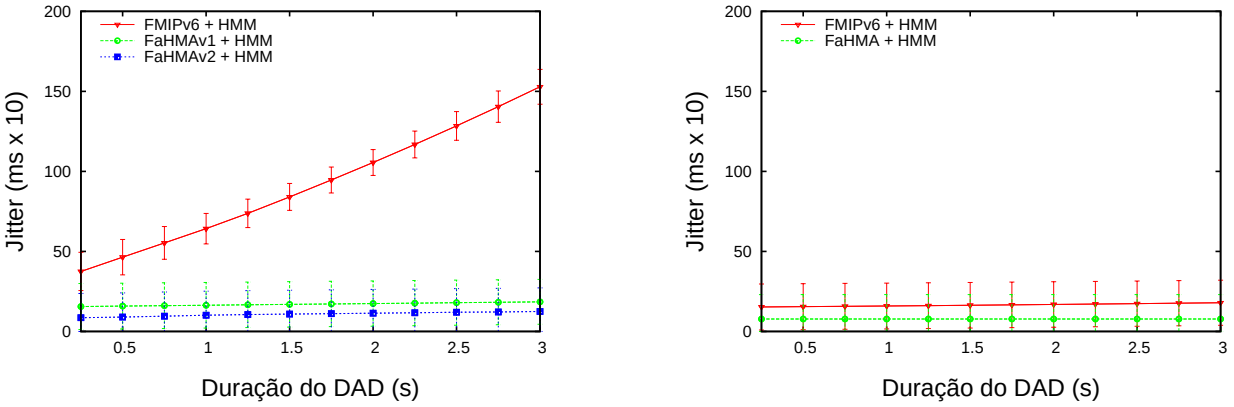
No modo preditivo, temos a situação visualizada na Figura 5.5a. No FMIPv6, quanto maior o atraso DAD, maior a quantidade de pacotes armazenados no NAR. No gráfico, vemos que o FMIPv6 inicia com a média de aproximadamente 219 ms e, a cada incremento do tempo de DAD, essa média também aumenta, chegando ao nível de 400 ms, quando esse tempo chega aos 3 s. Por outro lado, as duas versões do FaHMA iniciam o armazenamento de pacotes no NAR somente após a realização do DAD, não recebendo influências dessa variável. Pelo gráfico, visualizamos um valor médio aproximadamente constante do atraso fim a fim, durante toda variação do tempo do DAD: 179 ms no FaHMAv1 e 173 ms no FaHMAv2, valores inferiores a toda faixa assumida pelo FMIPv6.

Ainda na Figura 5.5a, em relação ao FMIPv6, podemos visualizar uma redução na ordem de crescimento da média do atraso fim a fim a partir do tempo de 1 s. Isso ocorre em virtude do aumento da perda de pacotes (Figura 5.4a), provocadas pela sobrecarga no *buffer* presente no NAR. Esses pacotes perdidos geram uma redução na média do atraso fim a fim, pois, além de não fornecerem valores para este cálculo, seriam pacotes que experimentariam os maiores atrasos.

No modo reativo, nenhuma das soluções trabalha com *buffer*. Portanto, o maior fator de impacto na média do atraso fim a fim deixa de existir neste modo de execução. Na Figura 5.5b, vemos que em todas as soluções as médias do atraso fim a fim se mantêm sempre constantes. O FMIPv6 possui uma média de atraso maior do que o FaHMA: 187 para 173 ms. Entretanto, como seus intervalos de confiança se cruzam, não podemos afirmar que uma solução é significativamente melhor do que a outra.

Jitter

Nesse tópico, finalizamos a avaliação do impacto do atraso do DAD no comportamento das nossas soluções, quando utilizadas para aplicações multimídias, investigando o seu impacto sobre o *jitter*. Lembramos que, quanto maior o *jitter*, pior a qualidade de uma recepção e consumo de dados.



(a) Modo Preditivo

(b) Modo Reativo

Figura 5.6: Impacto do Atraso do DAD sobre o *Jitter*.

No modo preditivo, o FMIPv6 armazena pacotes tanto no PAR quanto no NAR. Inicialmente, os pacotes são armazenados no PAR no momento em que este recebe um FBU do MN. Em seguida, esses pacotes são enviados para o NAR após o PAR receber um HAck daquele. Além disso, no NAR, teremos o armazenamento de novos pacotes que chegam ao PAR e são redirecionados pelo túnel após o DAD. Assim, identificamos três classes de atraso fim a fim de pacotes. O atraso dos pacotes enviados para o MN antes e após o processo de *handover* (A1), atraso dos pacotes que são armazenados apenas no NAR (A2) e o atraso dos pacotes que são armazenados tanto no PAR quanto no NAR (A3). Temos que, em média, A1<A2<A3. O média do A3 varia de acordo com o atraso do DAD, pois os pacotes só serão encaminhados para o *buffer* do NAR após a DAD. Logo, temos que a variação do atraso de A3 em relação a A1 e A2 tende a aumentar com o aumento do atraso do DAD. Pode-se visualizar essa situação na Figura 5.6a. O FMIPv6 inicia com um *jitter* médio de 3,7 ms, com um atraso de 0,25 s; e cresce com o aumento do atraso, chegando à média de 15,2 ms com 3 s de atraso, ou seja quase o triplo.

Com relação às soluções FaHMA, facilmente identificamos a ocorrência de apenas duas classes de atraso fim a fim: A1 e A2, pois não existem pacotes armazenados no PAR. Ambas as classes não são influenciadas pelo aumento do atraso DAD e, portanto, conservam aproximadamente a mesma média de *jitter*: o FaHMAv1 com uma média de aproximadamente 1,7 ms e o FaHMAv2 com 1,06 ms, valor inferior ao do primeiro. Essa diferença ocorre pois o A2 surge de processos diferentes de execução desses dois protocolos. No caso do FaHMAv1, temos que os pacotes são recebidos pelo PAR e reencaminhados para o NAR, onde serão armazenados em *buffer*. Por outro lado, no FaHMAv2, os pacotes são enviados diretamente para o NAR,

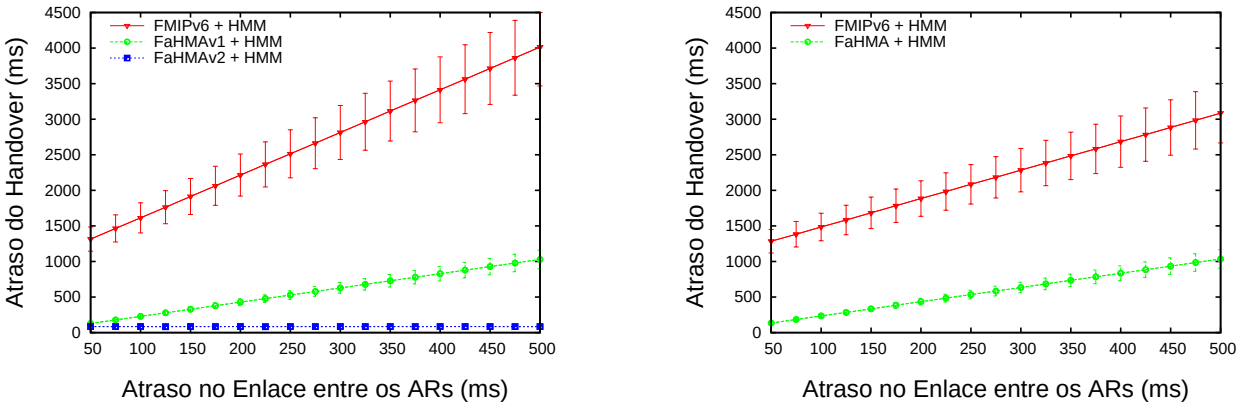
via *multicast*, justificando um valor médio inferior de *jitter*. Entretanto, não podemos afirmar que uma solução é significativamente melhor do que a outra, pois seus intervalos de confiança sempre se cruzam. A partir de 0.5 s de atraso, porém, é possível verificar que nossas soluções se tornam significativamente melhores do que o FMIPv6.

No modo reativo, nenhuma das soluções trabalha com *buffer*. Portanto, o maior fator de impacto no *jitter* deixa de existir nesse modo de execução. Na Figura 5.6b vemos que, em todas soluções, a média do *jitter* se mantém constante. O FMIPv6 possui média de atraso maior do que o FaHMA: 1,65 para 0,78 ms. Entretanto, como o seus intervalos de confiança se cruzam, não podemos afirmar que uma solução é significativamente melhor do que a outra.

5.3.2 Atraso no Enlace entre ARs

Nesta subseção, avaliaremos o impacto da variação do atraso no enlace entre os ARs sobre todas a métricas definidas na seção anterior: Atraso do *Handover*, Perda de Pacotes, Atraso Fim a Fim e o Jitter. Fizemos esse atraso assumir os seguintes valores: 50, 75, 100, 125, 150, 175, 200, 225, 250, 275, 300, 325, 350, 375, 400, 425, 450, 475, 500 (tempos em milissegundos).

Atraso do Handover



(a) Modo Preditivo

(b) Modo Reativo

Figura 5.7: Impacto do Atraso no Enlace entre ARs sobre o Atraso do *Handover*.

Nesse tópico, avaliamos o impacto da variação do atraso no enlace entre os ARs sobre o atraso do *handover*. Nesse caso, também considerado como o tempo da interrupção dos serviços.

No modo preditivo, o FMIPv6 e o FaHMAv1 estabelecem um túnel entre o PAR e o NAR para encaminhar os pacotes de dados destinados a um MN específico, durante o *handover*. Esse túnel é criado ao final da realização do DAD, ou seja, quando o PAR receber um HAck do NAR. Entretanto, no FMIPv6, após receber um FBU do MN, o PAR interrompe o envio de pacotes para o MN e passa a armazená-los em um *buffer*, até que esse túnel seja estabelecido, quando, então, esses pacotes são enviados para o NAR e armazenados também em um *buffer*, até que o MN possa recebê-los. Por outro lado, o FaHMAv1 não armazena tais pacotes; em vez disso, continua enviando-os para o MN até que PAR receba o HAck e o túnel seja criado e, a partir daí, o seu funcionamento passa a ser igual ao do FMIPv6. Com isso, no FMIPv6, o MN recebe o seu último pacote do PAR mais cedo do que no FaHMAv1. Quanto ao recebimento do primeiro pacote do NAR, ambos os protocolos funcionam da mesma forma e iniciam o recebimento dos pacotes armazenados no NAR após o envio da mensagem FNA. Portanto, com o aumento do atraso no enlace entre os ARs, essas duas soluções aumentam o atraso do *handover*, pois estabelecem o túnel entre os ARs. No FMIPv6, porém, esse atraso será maior em razão do momento em que se inicia o armazenamento dos pacotes.

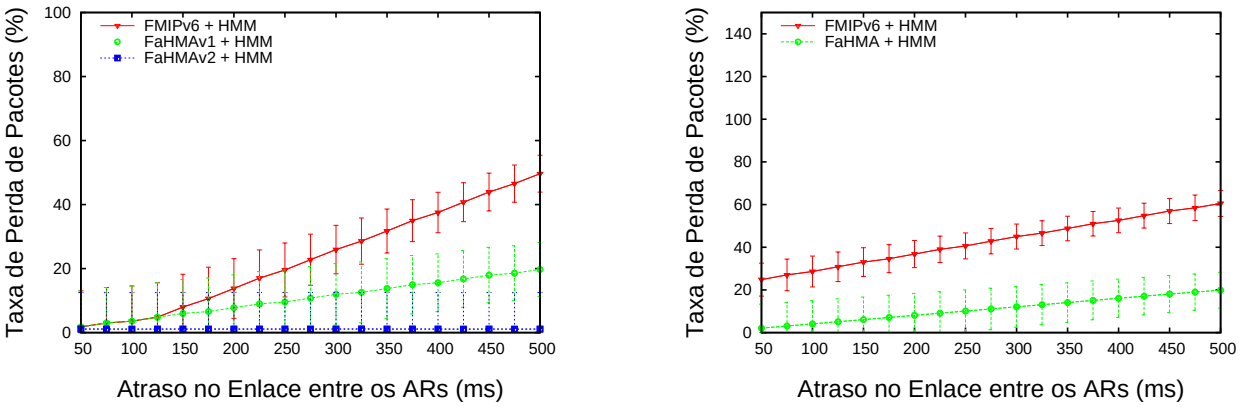
Através da Figura 5.7a, podemos visualizar a situação descrita anteriormente. Podemos verificar que o FMIPv6 inicia com um atraso médio de *handover* de 1314 ms contra 129 ms do FaHMAv1, para um atraso no enlace de 50 ms. Essas variáveis aumentam de forma proporcional e, ao final, com o atraso no enlace de 500 ms, assumem o valor de 4014 e 1028 ms, respectivamente. Como os intervalos de confiança não se cruzam, temos que o FaHMAv1 é significativamente melhor do que o FMIPv6, nesse caso.

O FaHMAv2, no modo preditivo, não estabelece o túnel entre os ARs, como descrito anteriormente. Ele utiliza o nosso mecanismo de endereçamento *multicast* para manter a entrega dos pacotes de dados para o MN durante todo o processo de *handover*. Logo, ele não é influenciado pelo aumento no atraso do enlace entre os ARs, o que o permite atingir um melhor desempenho do que o FaHMAv1 e, consequentemente, que o FMIPv6. Pela Figura 5.7a, podemos observar que o atraso do *handover* se mantém constante durante todo o experimento, assumindo um valor médio de 84 ms, valor já inferior ao assumido pelas outras soluções durante todo o intervalo. Como os intervalos de confiança desse protocolo e do FaHMAv1 não se cruzam, temos que o primeiro é significativamente melhor do que o segundo, nesse caso.

No modo reativo, não existe o estabelecimento de túnel em nenhuma das soluções. Portanto, não há dependência em relação ao crescimento do atraso no enlace entre os ARs, a não ser que parte desse atraso seja atribuído ao enlace que liga o router1 ao NAR, o que ocorre em nossos experimentos. Na Figura 5.7b, notamos que o FMIPv6 possui um atraso de *handover* maior que

o FaHMA, durante todo o experimento, iniciando com uma média de aproximadamente 1285 e terminando com 3085 ms, enquanto o FaHMA inicia com uma média de aproximadamente 135 e terminando com 1035 ms, tornando a nossa solução significativamente melhor do que o FMIPv6. Isso ocorre pois, além do aumento do atraso no enlace, existe ainda a dependência do FMIPv6 em relação ao atraso do DAD, que nesse caso assume o valor padrão de 1 s.

Taxa de Perda de Pacotes



(a) Modo Preditivo

(b) Modo Reativo

Figura 5.8: Impacto do Atraso no Enlace entre ARs sobre a Perda de Pacotes.

Nesse tópico, avaliamos o impacto da variação do atraso no enlace entre os ARs sobre a perda de pacotes.

Já foi mostrado que, no modo preditivo, tanto o FMIPv6 quanto o FaHMAv1 estabelecem um túnel entre o PAR e o NAR com o objetivo de encaminhar os pacotes de dados destinados a um MN específico, durante o processo de *handover*. O aumento do atraso no enlace entre os ARs causa um aumento da perda de pacotes nessas duas soluções, pois, com o aumento desse atraso, o tempo em que os pacotes trafegam no túnel aumenta, o que reduz a largura de banda disponível para o PAR e, consequentemente, causará um atraso no envio de novos pacotes. No FMIPv6, vimos que, ao receber um FBU do MN, ele interrompe o envio de pacotes e passa a armazená-los em um *buffer* até o momento de enviá-los para o NAR, através do túnel. No momento desse envio, com o aumento do atraso no enlace entre os ARs, a taxa de envio dos pacotes armazenados é reduzida, o que pode acarretar uma sobrecarga no *buffer* do PAR com a chegada dos novos pacotes e, consequentemente, a perda de alguns destes.

Podemos visualizar esta situação na Figura 5.8a. Nesse gráfico, no inicio do experimento, FMIPv6 é alvo de uma taxa perda de pacotes média de 1,8%, valor bem próximo ao das nossas

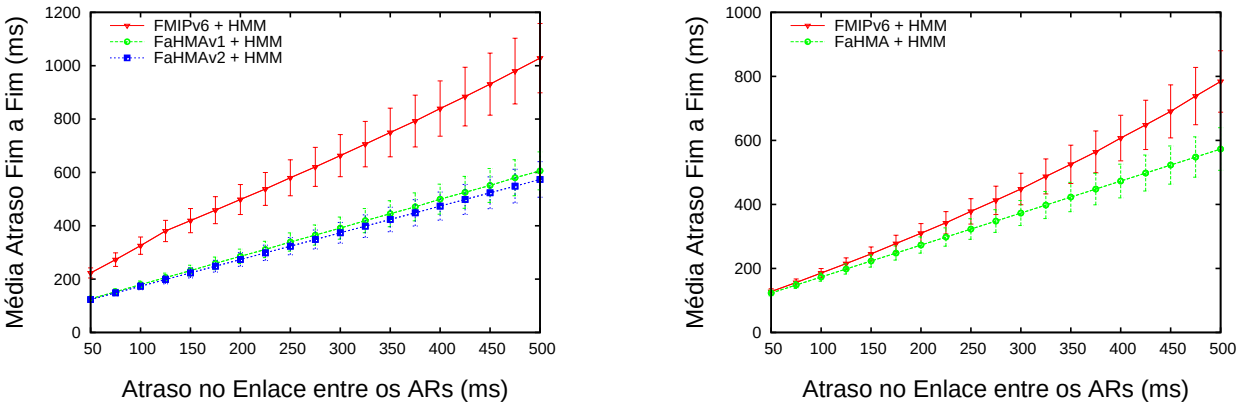
soluções. Durante o aumento do atraso, o PAR reduz cada vez mais a sua taxa de envio de pacotes, acarretando maior perda e, com o fim do experimento, esse protocolo chega à taxa de 49,64% de pacotes perdidos.

Considerando agora o FaHMAv1 no modo preditivo, temos que esse protocolo não armazena dados no *buffer* do PAR, no lugar disso, continua o envio dos pacotes para o MN até o recebimento do HAck enviado do NAR para o PAR. Nesse momento, o túnel é estabelecido e os pacotes são redirecionados por ele. No parágrafo anterior, vimos que, com o aumento do atraso no enlace entre os ARs, a taxa de envio dos pacotes por parte do PAR é reduzida. Por isso, novos pacotes que chegam ao PAR podem ser descartados pela indisponibilidade de largura de banda disponível para o envio destes. Na Figura 5.8a, esse protocolo inicia com uma taxa de 1,8%, valor idêntico ao do FMIPv6. Entretanto, durante o aumento do atraso, essa perda aumenta e, ao final, atinge a taxa média 19,74% de pacotes perdidos, um crescimento inferior ao do FMIPv6. Isso ocorre pelo fato de o PAR no FMIPv6 antecipar a interrupção da entrega dos pacotes para o MN, aumentando o número de pacotes que devem ser enviados pelo túnel. Podemos visualizar, também, que, a partir do atraso de 350 ms, o FaHMAv1 se torna uma solução significativamente melhor do que o FMIPv6, pois os seus intervalos de confiança não mais se cruzam.

A execução do FaHMAv2 no modo preditivo mantém constante a taxa média da perda, 1,11% apenas. Isso ocorre pelo fato de esse protocolo não ser influenciado por esse tipo de atraso. Podemos verificar também que, a partir do atraso de 500 ms, o FaHMAv2 se torna significativamente melhor do que o FaHMAv1. A cada aumento desse atraso, os intervalos de confiança desses dois protocolos cada vez ficam mais distantes, o que nos permite predizer gradual melhora do primeiro em relação ao segundo.

No modo reativo, não há o estabelecimento do túnel em nenhuma das soluções. Portanto, não existe dependência em relação ao crescimento do atraso no enlace entre os ARs, a não ser que parte desse atraso seja atribuído ao enlace que liga o router1 ao NAR, o que ocorre em nossos experimentos. Na Figura 5.8b, notamos que o FMIPv6 apresenta uma perda de pacotes maior do que FaHMA, durante todo o experimento, iniciando com uma taxa média de 24,84% e terminando com 60,53% de pacotes perdidos, enquanto o FaHMA inicia com uma taxa média de 2,1% e terminando com 19,87% de pacotes perdidos, tornando a nossa solução significativamente melhor que do o FMIPv6. Isso ocorre porque, além do aumento do atraso no enlace, existe ainda a dependência do FMIPv6 em relação ao atraso do DAD, que nesse caso assume o valor padrão de 1 s.

Atraso Fim a Fim



(a) Modo Preditivo

(b) Modo Reativo

Figura 5.9: Impacto do Atraso no Enlace entre ARs sobre o Atraso Fim a Fim.

Nesse tópico, avaliamos o impacto da variação do atraso no enlace entre os ARs sobre o atraso fim a fim. Pretendemos avaliar o impacto do atraso no enlace entre os ARs no comportamento das nossas soluções, quando utilizadas para aplicações multimídias. No próximo tópico, analisaremos o *jitter*.

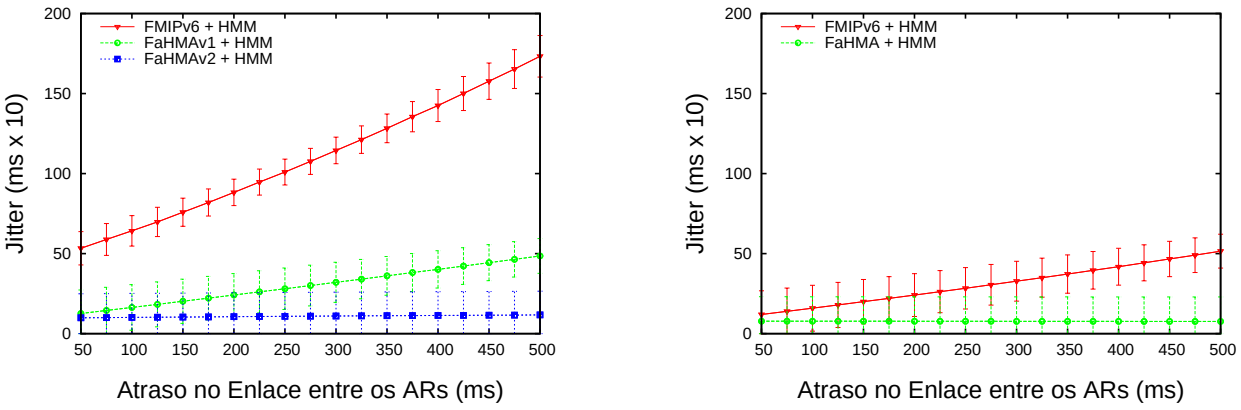
No FMIPv6 e no FaHMAv1, com relação ao modo preditivo, a média do atraso fim a fim é diretamente proporcional à quantidade de tempo que os pacotes ficam armazenado no *buffer*, o número de pacotes que ficam armazenados e o atraso no túnel estabelecido. Quando comparamos FMIPv6 com FaHMAv1, temos uma quantidade maior de pacotes armazenados e por uma duração de tempo maior no primeiro, pois depende diretamente da duração do DAD que, nesse caso, assume um valor padrão de 1 s. Em razão dessa característica, podemos visualizar pela Figura 5.9a, que com o aumento do atraso no enlace entre os ARs, a média do atraso fim a fim aumentará nos dois protocolos, entretanto, no FMIPv6, esse atraso se manterá maior do que o FaHMAv1 durante todo o experimento. Além disso, o FaHMAv1 é uma solução significativamente melhor do que o primeiro, em virtude do não cruzamento dos intervalos de confiança. O FMIPv6 inicia com um atraso fim a fim médio de 223 ms e termina com 1028 ms, enquanto o FaHMAv1 inicia com 126 e termina com 605 ms.

Mesmo o FaHMAv2 não sendo influenciado por esse tipo de atraso, por não estabelecer o túnel durante o *handover*, esse protocolo recebe alguma influência, provocada por parte desse atraso atribuído ao enlace que liga o router1 ao NAR. Por esse enlace, passam os pacotes *multicast* destinados ao MN. Por esse motivo, nesse protocolo, temos um aumento na média do

atraso fim a fim com o crescimento do atraso do enlace entre os ARs, que com o tempo, tende a ser menor do que o do FaHMAv1. O atraso fim a fim inicia com uma média de 123 ms e chega à 574 ms. Como os intervalos de confiança se cruzam, não podemos dizer que uma solução é melhor do que a outra.

No modo reativo, não há o estabelecimento do túnel em nenhuma das soluções. Portanto, não há dependência em relação ao crescimento do atraso no enlace entre os ARs, a não ser que parte desse atraso seja atribuído ao enlace que liga o router1 ao NAR, o que ocorre em nossos experimentos. Na Figura 5.9b, notamos que, no FMIPv6, ocorre um aumento mais significante no atraso fim a fim do que no FaHMA, iniciando com uma média de aproximadamente 128 ms e terminando com 784 ms, enquanto o FaHMA inicia com uma média de aproximadamente 123 ms e terminando com 573 ms. Esse aumento decorre do motivo descrito no começo desse parágrafo. Nesse caso, a inclusão desse atraso provoca maior tempo de chegada dos pacotes no NAR. Finalmente, a diferença entre esses aumentos ocorre em razão da dependência do FMIPv6 com relação à duração do DAD. Podemos verificar que para um atraso nos enlaces de 375 ms, o FaHMA torna-se significativamente melhor do que o FMIPv6.

Jitter



(a) Modo Preditivo

(b) Modo Reativo

Figura 5.10: Impacto do Atraso do DAD sobre o *Jitter*.

Nesse tópico, finalizamos a avaliação do impacto do atraso do DAD no comportamento das nossas soluções, quando utilizadas para aplicações multimídias, investigando o seu impacto sobre o *jitter*. Lembramos que, quanto maior o *jitter*, pior a qualidade de uma comunicação de dados.

Quando discutimos sobre o impacto do atraso do DAD no *jitter*, mostramos que, no modo preditivo do FMIPv6, podemos identificar três classes de atraso fim a fim de pacotes. O atraso dos pacotes enviados para o MN antes e após o processo de *handover* (A1), o atraso dos pacotes armazenados apenas no NAR (A2) e o atraso dos pacotes que são armazenados tanto no PAR quanto no NAR (A3). Além disso, também mostramos que, em média, A1<A2<A3. Explicamos também que, em nossas soluções, identificamos a ocorrência de apenas duas classes de atraso fim a fim - A1 e A2 - pois não há pacotes armazenados no PAR. Entretanto, o A2 surge de processos diferentes de execução desses dois protocolos. No caso do FMIMASv1, temos que os pacotes são recebidos pelo PAR e reencaminhados para o NAR utilizando o túnel, onde serão armazenados em *buffer*. Por outro lado, no FaHMAv2, os pacotes são enviados diretamente para o NAR, via *multicast*. Convencionamos tratar o A2 do FaHMAv1 por A2.1 e o do FaHMAv2 por A2.2.

Já foi citado que, no modo preditivo, tanto o FMIPv6 quanto o FaHMAv1 estabelecem um túnel entre o PAR e o NAR com o objetivo de encaminhar os pacotes de dados destinados a um MN específico, durante o processo de *handover*. Portanto, no FMIPv6, temos que a variação do atraso de A3 e A2 em relação a A1 tende a crescer com o aumento desse atraso nos enlaces. No FaHMAv1, teremos que a variação do atraso de A2.1 em relação a A1 também tende a crescer com o aumento desse atraso. Como no FaHMAv1, temos apenas o A2.1 recebendo influência, afirmamos que o *jitter* provocado por esse protocolo tende a ser inferior ao provocado pelo FMIPv6. Podemos visualizar essa situação na figura 5.10a. Pelo gráfico, podemos verificar que o FaHMAv1 é significativamente melhor do que o FMIPv6 durante todo o experimento, iniciando com um *jitter* médio de 1,26 ms, para um atraso de 0,25 s; e chegando à 4,85 ms com 3 s de atraso. No FMIPv6 temos uma média de *jitter* que vai de 5,33 à 17,73 ms.

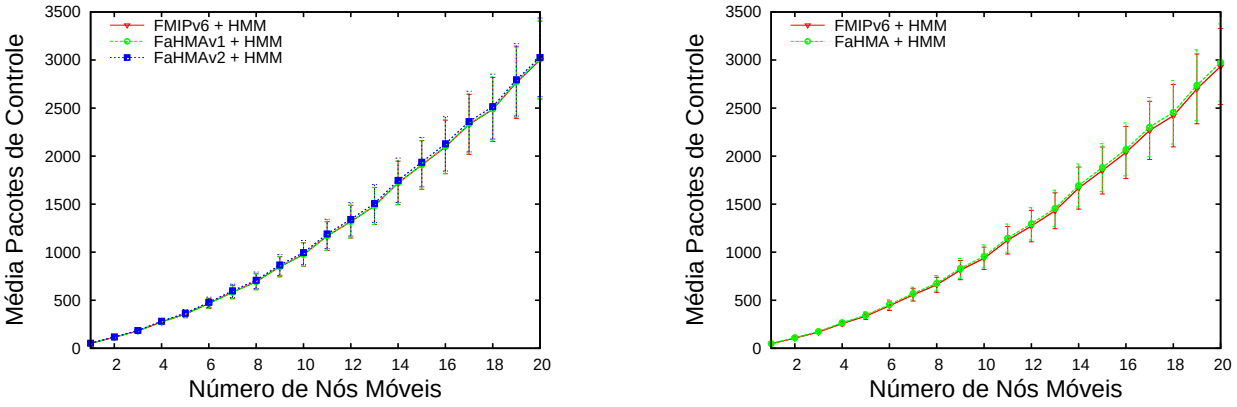
O FaHMAv2, no modo preditivo, possui apenas duas classes de atraso fim a fim - A1 e A2.2 - pois não existem pacotes armazenados no PAR. Ambas as classes não são influenciadas pelo aumento do atraso no enlace entre os ARs e, portanto, conservam a mesma média de *jitter*: 1.1 ms. Pela Figura 5.10a, podemos verificar que, a partir de um atraso de 400 ms, o FaHMAv2 se torna uma solução significativamente melhor do que o FaHMAv1, pois os seus intervalos de confiança passam a não se cruzar.

No modo reativo, nenhuma das soluções trabalha com o túnel. Portanto, o maior fator de impacto no *jitter* deixa de existir nesse modo de execução. Na Figura 5.10b, observamos que em todas soluções a média do *jitter* se mantém constante. O FMIPv6 possui média de atraso maior do que o FaHMA: 1,65 ms para 0,78 ms. Como, porém, o seus intervalos de confiança se cruzam, não podemos afirmar que uma solução é significativamente melhor do que a outra.

5.3.3 Número de Nós Móveis

Nesta subseção, avaliaremos o impacto da variação do número de nós móveis sobre uma única métrica definida na seção anterior: Número de Pacotes de Controle. Essa avaliação é feita como forma de avaliar o impacto deste fator sobre a escalabilidade da rede. Fizemos o número de nós móveis variar a sua quantidade de 1 a 20 nós.

Número de Pacotes de Controle



(a) Modo Preditivo

(b) Modo Reativo

Figura 5.11: Impacto do Número de Nós Móveis sobre o Número de Pacotes de Controle.

Podemos verificar na Tabela 5.3 que, no modo preditivo, o FMIPv6 e FaHMAv1, mesmo com diferentes formas de execução, apresentam basicamente a mesma média de quantidade de pacotes de controle entregues durante o processo de *handover*. Isso ocorre porque esses dois protocolos enviam as mesmas mensagens de controle, realizando basicamente o mesmo processo para realizar o *handover*. A única diferença consiste no armazenamento dos pacotes de dados em *buffer* no PAR por parte do FMIPv6, o que não influencia na quantidade de pacotes de controle trocados.

O FaHMAv2, no modo preditivo, por sua vez se utiliza de endereçamento *multicast* para manter o recebimento dos pacotes de dados durante a atividade DAD. Nesse caso, é necessária a troca de novos pacotes de controle para estabelecer o grupo *multicast*. Portanto, comparando com os outros dois protocolos, o FaHMAv2 utilizará mais pacotes de controle durante o processo de *handover*. Na Tabela 5.3 podemos comprovar essa afirmação. Por exemplo, para uma quantidade de 8 MNs, tem-se uma média de 708 pacotes de controles trocados na execução do FaHMAv2, para uma média aproximada de 692 pacotes nos outros dois protocolos.

Tabela 5.3: Número de Nós Móveis x Número de Pacotes de Controle.

	Modo Preditivo			Modo Reativo		
Qtd. de Nós	**FMIPv6**	**FaHMAv1**	**FaHMAv2**	**FMIPv6**	**FaHMAv1**	**FaHMAv2**
1	51	51	53	47	49	49
2	114.12	114.12	118.12	106.12	110.12	110.12
3	180.255	180.255	186.27	168.27	174.27	174.27
4	273.705	273.705	281.66	257.72	265.72	265.72
5	354.885	354.995	365.035	335.19	345.19	345.19
6	466.42	466.32	478.38	442.62	454.62	454.62
7	583.665	583.28	597.505	555.905	569.905	569.905
8	692.08	692.18	708.145	660.715	676.33	676.33
9	848.525	848.555	866.505	813.315	831.2	831.2
10	975.62	975.83	995.735	936.395	956.615	956.615
11	1167.78	1168.41	1189.99	1124.67	1146.83	1146.83
12	1317.86	1318.43	1341.92	1271.73	1295.49	1295.49
13	1480.92	1480.92	1507.17	1431.55	1456.31	1456.31
14	1721.24	1722.01	1748.26	1666.94	1694.65	1694.65
15	1907.58	1908.84	1936.98	1850.04	1880.16	1880.16
16	2095.61	2099.83	2127.41	2037.42	2070.09	2070.09
17	2330.03	2334.72	2359.68	2267.32	2302.14	2302.14
18	2485.82	2490.43	2514.83	2419.99	2455.53	2455.53
19	2765.18	2773.8	2794.03	2699.72	2737.76	2737.76
20	2999.86	3009.31	3028.35	2932.77	2974.13	2974.13

Entretanto, de acordo com a Figura 5.11a, considerando a quantidade de pacotes de controle trocados, não podemos afirmar que uma solução é melhor do que a outra no modo preditivo, mesmo com a diferença na média dessa quantidade, pois os intervalos de confiança se cruzam durante todo o experimento.

Finalmente, no modo reativo, o FaHMA utiliza-se de mais pacotes de controle como forma de estabelecer o grupo *multicast*, necessário para manter o recebimento de pacotes de dados por parte do MN, durante o DAD. Na Tabela 5.3, por exemplo, para os mesmos 8 MNs, temos uma média aproximada de 676 pacotes de controles trocados na execução do FaHMA, para uma média aproximada de 660 pacotes no FMIPv6. Entretanto, de acordo com a Figura 5.11b, também não podemos afirmar que uma solução é melhor do que a outra no modo reativo, mesmo com a diferença na média dessa quantidade, pois os intervalos de confiança se cruzam durante todo o experimento.

5.4 Conclusão

Neste capítulo, apresentamos uma análise de desempenho dos nossos protocolos propostos. Os resultados para essa análise foram obtidos por meio de simulações, abordagem escolhida em razão do seu grau de precisão, superior ao de um modelo matemático; e do seu baixo custo em relação às medições em ambientes reais. Em nossa análise, comparamos o comportamento

das duas versões do FaHMA com os resultados obtidos por meio da simulação do protocolo FMIPv6, pois quisemos apenas verificar se os objetivos das nossas soluções são alcançados.

Como visto nas análises discutidas neste capítulo, para os fatores de impacto analisados, quando executados no modo preditivo, ambos FaHMAv1 e FaHMAv2, obtêm melhores resultados do que o FMIPv6, reduzindo o atraso do *handover* (nesse caso, representando a interrupção dos serviços) e a perda de pacotes. Além disso, quando avaliados para as duas métricas mais importantes para determinar a qualidade do funcionamento de aplicações multimídias em rede, o atraso fim a fim e o *jitter*, nossas soluções apresentam os melhores resultados, mantendo as médias dessas duas métricas constantes e menores do que as obtidas dos resultados do FMIPv6.

No modo reativo, para os fatores de impacto analisados, temos que o FaHMA (nesse caso, considerando as duas soluções) reduz o atraso do *handover* e a perda de pacotes, em relação ao FMIPv6. Quanto ao atraso fim a fim e ao *jitter*, quando o fator analisado é o atraso no enlace entre os ARs, o FaHMA apresenta-se como a melhor solução, pois, embora não sendo constantes, as médias nessas duas métricas estão sempre abaixo das do FMIPv6, mesmo que para valores menores desse fator, os intervalos de confiança se cruzem. Por outro lado, quando o fator analisado é o atraso do DAD, temos que as médias obtidas dessas métricas, para ambos os protocolos (FaHMA e FMIPv6), se mantêm constantes durante todo o experimento e, embora as médias do FaHMA sejam menores, não podemos determinar quais das duas soluções é a melhor, pois os seus intervalos de confiança sempre se cruzam.

Finalmente, tanto no modo preditivo quanto no reativo, não podemos afirmar qual das soluções é a melhor, considerando a quantidade de pacotes de controle trocados, pois os intervalos de confiança sempre se cruzam. Logo, não teremos um grande impacto na escalabilidade da rede, ao utilizar as duas versões do FaHMA para gerenciar o *handover*.

6 *Considerações Finais*

Esta dissertação descreve a proposta de um mecanismo de melhoria para o processo de *handover* em redes heterogêneas, também chamado de *Handovers* Verticais. Na Seção 6.1, são apresentadas as conclusões sobre esta dissertação, sendo discutidos os resultados alcançados com a utilização das duas versões do protocolo FaHMA. Na Seção 6.2, são exibidas as principais contribuições alcançadas com a realização deste trabalho. Finalmente, na Seção 6.3, são discutidas possíveis perspectivas de trabalhos futuros, que foram apontados durante o texto e que precisam de um maior detalhamento.

6.1 Conclusões

O objetivo central deste trabalho foi propor um mecanismo de melhoria para o processo de *handover* em redes heterogêneas, também chamado de *Handovers* Verticais. Nossa solução consiste em um esquema que considera a mobilidade IP como a principal abordagem para garantir o chamado *Handover* Transparente. Com essa solução, objetivamos reduzir o atraso de *handover* e a perda de pacotes, critérios mais importantes para avaliar a efetividade de uma solução de *handover* e reduzir, sem grandes impactos sobre a escalabilidade da rede , a interrupção dos serviços, que consiste no atraso na resposta dos serviços, percebido pelo usuário final.

Para tanto, primeiramente, foi definida uma camada lógica, o HMM (*Handover Management Module*). Tal camada utiliza os serviços do *framework* MIH descrito no padrão IEEE 802.21 e, ao mesmo tempo, disponibiliza serviços para as camadas superiores, com o intuito de habilitar e facilitar o *Handover* Vertical. Dentre os serviços providos para essas camadas superiores, podemos citar: permite uma comunicação transparente entre o MIHF e essas camadas; é responsável pela decisão do *handover*, ou seja, decidir se o *handover* é necessário ou não (detecção do *handover*) e realiza a seleção da próxima rede a ser estabelecida uma nova conexão.

Por fim, definimos o protocolo FaHMA (*Fast Handover using Multicast Addressing*), que dividimos em duas versões: FaHMAv1 e FaHMAv2. Tal protocolo consiste em uma extensão do protocolo FMIPv6, que propõe a integração deste com um mecanismo de endereçamento *multicast*. Tal integração consiste em obter as vantagens de ambas as abordagens. Tais vantagens consistem em: reduzir o atraso de *handover* e a perda de pacotes (FMIPv6) e, além disso, reduzir, sem grandes impactos sobre a escalabilidade, a interrupção dos serviços (*multicast*), que consiste no atraso na resposta dos serviços, percebido pelo usuário final.

Para a análise de desempenho do FaHMA, realizamos experimentos de simulação mostrando que este protocolo alcança os seus objetivos, que consiste em reduzir a interrupção dos serviços, causado pelo atraso do *handover*; e a perda de pacotes. A sua eficiência foi avaliada por meio da comparação de seu desempenho com o desempenho alcançado pelo protocolo FMIPv6, tanto executando no modo preditivo quanto no reativo. Além disso, o atraso fim a fim e o *jitter* foram utilizados para avaliar o comportamento das nossas soluções, quando utilizadas com aplicações multimídias.

Dentre as dependências existentes para a implantação deste trabalho em ambiente reais, temos:

- O núcleo da rede (rede externa aos ARs) deve disponibilizar um suporte a *multicast*, ou seja, deve permitir o estabelecimento de grupos *multicast* através de um protocolo de roteamento qualquer; e
- Deve existir uma implementação do *framework* MIH nos MNs, como forma de habilitar o uso do HMM; e nos Access Routers, para dar suporte as requisições inciadas por aquele.

Respeitada tais dependências, são necessários os seguintes procedimentos para que está solução possa ser utilizada.

- Nos MNs (*Mobile Nodes*) deve ser implantado o conjunto FaHMA (cliente) + HMM; e
- Para cada AR (*Access Router*) deve ser implantado o FaHMA (servidor) e uma entidade do PD (*Proxy Device*), descrita na proposta deste trabalho.

Em relação a originalidade, diferentemente dos trabalhos relacionados abordados neste trabalho (MUSSABBIR et al., 2007)(HUANG; WU, 2009)(KIM et al., 2008)(AN et al., 2006), a nossa proposta visa reduzir a interrupção dos serviços provocados pelo FMIPv6 que podem atingir níveis inaceitáveis por aplicações fluxo contínuo e em tempo real. Além disso, é considerada a probabilidade da ocorrência do *handover* no modo reativo, através de um esquema

de operação que busca reduzir o seu impacto no atraso do *handover* (interrupção dos serviços) e na perda de pacotes. Finalmente, a análise de desempenho deste trabalho foi realizada através da repetição dos experimentos, ou seja, os resultados são gerados com intervalos de confiança. Deste modo, é considerada a aleatoriedade do ambiente, necessária para aproximar os resultados obtidos aos de um ambiente real.

6.2 Contribuições

A principal contribuição desta pesquisa consiste no projeto e no desenvolvimento de um mecanismo de *Handover* Vertical que reduz a perda de pacotes e o atraso desta operação, principais fatores para determinar a efetividade de uma solução deste tipo. Além disso, a proposta reduz a interrupção de serviços, provocada pela atividade DAD e que representa o atraso percebido pelo usuário final de serviços de mídias contínuas (*streaming* e tempo real), visando alcançar o chamado *Handover* Transparente.

Há, no entanto, contribuições secundárias que foram alcançadas com a descrição deste mecanismo de *Handover* Vertical e que constituem sua base de operação, conforme descrito a seguir.

- Definição do HMM (*Handover Management Module*), uma camada lógica que utiliza os serviços do *framework* MIH descrito no padrão IEEE 802.21 e que, ao mesmo tempo, disponibiliza serviços para as camadas superiores, com o intuito de habilitar o *Handover* Vertical;

- Realização da integração entre o protocolo FMIPv6 e o HMM, estabelecendo comunicação e sincronização destes através da troca de mensagens. Com essa integração, criamos a proposta-base de um mecanismo que habilita a mobilidade IP em redes heterogêneas;

- Descrição e implementação de uma extensão do protocolo FMIPv6, o FaHMA (*Fast Handover using Multicast Addressing*), mediante a utilização de endereçamento *multicast*. Este protocolo é responsável por gerenciar a conexão de pacotes de dados, garantindo uma redução do atraso de *handover* e da perda de pacotes e, além disso, reduzindo a interrupção dos serviços.

- Descrição da execução do FaHMA, integrado ao HMM, durante o *handover*. Neste ponto, descrevemos o MPD *User*, entidade pertencente ao FaHMA que é responsável por implementar o mecanismo *multicast* necessário para reduzir a interrupção dos serviços provocado pelo *handover*; e

6.3 Trabalhos Futuros

No decorrer desta dissertação, foram identificados alguns pontos que ainda necessitam ser aprofundados e/ou solucionados. Portanto, a seguir serão listados alguns direcionamentos para possíveis trabalhos futuros decorrentes desta pesquisa.

- Apresentar uma especificação detalhada da implementação da entidade NF (*Network Filter*), componente do HMM responsável por selecionar a próxima rede a se conectar. Para este trabalho, descreveremos apenas o mecanismo *query/response* do NF e uma especificação geral de como implementar o mecanismo de seleção. Portanto, ainda carece da implementação de um mecanismo que será responsável pela realização da seleção da nova rede, com base nos parâmetros de entrada;

- Considerar, além do critério atual utilizado no mecanismo de seleção da próxima rede (probabilidade de realizar o *handover* no modo preditivo), outros critérios, como: largura de banda, congestionamento etc;

- Nesta proposta não foi considerada a questão da segurança das operações. Portanto, é necessário implementar mecanismos de segurança que garantam a confidencialidade, a integridade e a confiabilidade das mensagens trocadas durante o processo de *handover*.

- Realizar a análise de desempenho do HMM, de forma a demonstrar a sua eficiência em relação a realização de *Handovers* Verticais;

- Realizar uma análise de desempenho que compara as duas versões do FaHMA, com o objetivo de mostrar qual das duas soluções se adapta melhor para determinados cenários; e

- Implementar e implantar a proposta com o objetivo de analisar e avaliar a sua viabilidade em ambientes reais.

Referências Bibliográficas

ALNAS, M.; AWAN, I.; HOLTON, D. R. W. Enhanced mobile ip handoff using link layer information. In: *Proceedings of the 11th International Conference on Information Integration and Web-based Applications & Services*. New York, NY, USA: ACM, 2009. (iiWAS '09), p. 344–349. ISBN 978-1-60558-660-1. Disponível em: <http://doi.acm.org/10.1145/1806338-.1806401>.

AN, Y. Y. et al. Reduction of Handover Latency Using MIH Services in MIPv6. In: *Advanced Information Networking and Applications, 2006. AINA 2006. 20th International Conference on*. [S.l.: s.n.], 2006. v. 2, p. 229 –234. ISSN 1550-445X.

ANDREWS, J. G.; GHOSH, A.; MUHAMED, R. *Fundamentals of WiMAX: Understanding Broadband Wireless Networking (Prentice Hall Communications Engineering and Emerging Technologies Series)*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2007. ISBN 0132225522.

BANERJEE, N.; ACHARYA, A.; DAS, S. Seamless SIP-Based Mobility for Multimedia Applications. *Network, IEEE*, v. 20, n. 2, p. 6 –13, march-april 2006. ISSN 0890-8044.

BIJU, I. *IEEE 802.11 Wireless Networks: Basic Concepts, Mobility Management and Security Enhancements*. Saarbrücken, Germany, Germany: VDM Verlag, 2009. ISBN 3639186087, 9783639186086.

BRADNER, S. et al. *3GPP2-IETF Standardization Collaboration*. United States: RFC Editor, 2001.

CAIN, B. et al. *Internet Group Management Protocol, Version 3*. IETF, October 2002. RFC 3376 (Proposed Standard). (Request for Comments, 3376). Disponível em: <http://www.ietf-.org/rfc/rfc3376.txt>.

DEERING, S.; FENNER, W.; HABERMAN, B. *Multicast Listener Discovery (MLD) for IPv6*. IETF, oct 1999. RFC 2710. (Request for Comments). Disponível em: <http://www.ietf.org-/rfc/rfc2710.txt>.

DEERING, S.; HINDEN, R. *RFC 2460 Internet Protocol, Version 6 (IPv6) Specification*. [S.l.], December 1998. Disponível em: <http://tools.ietf.org/html/rfc2460>.

DIAB, A.; MITSCHELE-THIEL, A. Comparative Analysis of Proxy MIPv6 and Fast MIPv6. In: *Proceedings of the 7th ACM international symposium on Mobility management and wireless access*. New York, NY, USA: ACM, 2009. (MobiWAC '09), p. 26–33. ISBN 978-1-60558-617-5. Disponível em: <http://doi.acm.org/10.1145/1641776.1641781>.

DO, H. T.; ONOZATO, Y. A Comparison of Different Paging Mechanisms for Mobile IP. *Wirel. Netw.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 13, p. 379–395, June 2007. ISSN 1022-0038. Disponível em: <http://dx.doi.org/10.1007/s11276-006-6467-8>.

ESTRIN, D.; FARINACCI, D.; HELMY, A. *Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*. IETF, jun 1998. RFC 2362. (Request for Comments). Disponível em: <http://www.ietf.org/rfc/rfc2362.txt>.

FENNER, B.; HE, H.; HABERMAN, B. *Internet Group Management Protocol (IGMP)/Multicast Listener Discovery (MLD)-Based Multicast Forwarding (IGMP/MLD Proxying)*. IETF, aug 2006. RFC 4605. (Request for Comments). Disponível em: <http://www.ietf.org/rfc/rfc4605.txt>.

FENNER, W. *Internet Group Management Protocol, Version 2*. [S.l.], nov. 1997. [Standards Track RFC 2236]. Disponível em: <http://www.ietf.org/rfc/rfc2236.txt>.

GROUP, I. *IPv6 Multicast Address Space Registry*. United States: IANA, 2011.

GUNDAVELLI, E. S.; LEUNG, K.; DEVARAPALLI, V. *Proxy Mobile IPv6*. IETF, aug 2008. RFC 5213. (Request for Comments). Disponível em: <http://www.ietf.org/rfc/rfc5213.txt>.

HUANG, H.-H.; WU, J.-S. A Pre-Binding Update Fast Handover Control Using IEEE 802.21 MIH over 802.16e Networks. In: *Communications and Mobile Computing, 2009. CMC '09. WRI International Conference on*. [S.l.: s.n.], 2009. v. 2, p. 417 –421.

HYYTIä, E.; VIRTAMO, J. Random Waypoint Mobility Model in Cellular Networks. *Wirel. Netw.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 13, p. 177–188, April 2007. ISSN 1022-0038. Disponível em: <http://dx.doi.org/10.1007/s11276-006-4600-3>.

IANA. *Internet Protocol Version 6 Multicast Addresses*. maio 2011. Disponível em: <http://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xml>.

IEEE. IEEE Standard for Local and Metropolitan Area Networks: Media Independent Handoff, Draft IEEE P802.21/D03.00. *IEEE Std 802.21-2006*, dec 2006.

IEEE. IEEE Standard for Local and Metropolitan Area Networks- Part 21: Media Independent Handover. *IEEE Std 802.21-2008*, p. c1 –301, 21 2009.

JAIN, R. Book. *The Art of Computer Systems Performance Analysis : Techniques for Experimental Design, Measurement, Simulation, and Modeling / Raj Jain*. Wiley, New York :, 1991. xxvii, 685 p. : p. ISBN 0471503363. Disponível em: <http://www.loc.gov/catdir/toc-/onix04/90045479.html>.

JOHNSON, D.; PERKINS, C.; ARKKO, J. *Mobility Support in IPv6*. IETF, jul 2011. RFC 6275. (Request for Comments). Disponível em: <http://www.ietf.org/rfc/rfc6275.txt>.

KASHIHARA, S.; TSUKAMOTO, K.; OIE, Y. Service-Oriented Mobility Management Architecture for Seamless Handover in Ubiquitous Networks. *Wireless Communications, IEEE*, v. 14, n. 2, p. 28 –34, april 2007. ISSN 1536-1284.

KIM, B.-K. et al. Enhanced FMIPv4 Horizontal Handover with Minimized Channel Scanning Time Based on Media Independent Handover (MIH). In: *Network Operations and Management Symposium Workshops, 2008. NOMS Workshops 2008. IEEE*. [S.l.: s.n.], 2008. p. 52 –55.

KOODLI, E. R. *Fast Handovers for Mobile IPv6*. IETF, jul 2005. RFC 4068. (Request for Comments). Disponível em: <http://www.ietf.org/rfc/rfc4068.txt>.

KUROSE, J. F.; ROSS, K. W. *Redes de Computadores e a Internet: Uma Abordagem Top-Down*. Trad. 5 ed. São Paulo: Addison Wesley, 2010.

LAI, W. K.; SHIEH, C.-S.; CHOU, K.-P. Improving Handover Performance by Switching Between Unicast and Multicast Addressing. *Wireless Communications, IEEE Transactions on*, v. 8, n. 3, p. 1238 –1246, march 2009. ISSN 1536-1276.

LEOLEIS, G. A.; VENIERIS, I. S. Fast MIPv6 Extensions Supporting Seamless Multicast Handovers. *Comput. Netw.*, Elsevier North-Holland, Inc., New York, NY, USA, v. 51, p. 2379–2396, June 2007. ISSN 1389-1286. Disponível em: <http://dl.acm.org/citation-.cfm?id=1241112% -.1241367>.

MA, L. et al. A New Method to Support UMTS/WLAN Vertical Handover Using SCTP. *Wireless Communications, IEEE*, v. 11, n. 4, p. 44 – 51, aug. 2004. ISSN 1536-1284.

MCCANNE, S.; FLOYD, S. *The Network Simulator - NS2*. 2006. Develop by University of Southern California.

MUSSABBIR, Q. et al. Optimized FMIPv6 Using IEEE 802.21 MIH Services in Vehicular Networks. *Vehicular Technology, IEEE Transactions on*, v. 56, n. 6, p. 3397 –3407, nov. 2007. ISSN 0018-9545.

NIST. *The Network Simulator NS-2 NIST add-on IEEE 802.21 Model*. jan 2007. Develop by The National Institute of Standards and Technology (NIST). Disponível em: <http://w3.antd-.nist.gov/seamlessandsecure% -/pubtool.shtml>.

NOGUEIRA, A. D. B. *Uma Proposta de Integração das Redes UMTS e IEEE 802.11 com Suporte a Mobilidade*. Dissertação (Mestrado) — Universidade Federal do Ceará, Fortaleza, CE, Brasil, 2007.

PERERA, E.; SIVARAMAN, V.; SENEVIRATNE, A. Survey on Network Mobility Support. *SIGMOBILE Mob. Comput. Commun. Rev.*, ACM, New York, NY, USA, v. 8, p. 7–19, April 2004. ISSN 1559-1662. Disponível em: <http://doi.acm.org/10.1145/997122.997127>.

PÉREZ-COSTA, X.; TORRENT-MORENO, M.; HARTENSTEIN, H. A Performance Comparison of Mobile IPv6, Hierarchical Mobile IPv6, Fast Handovers for Mobile IPv6 and their Combination. *SIGMOBILE Mob. Comput. Commun. Rev.*, ACM, New York, NY, USA, v. 7, n. 4, p. 5–19, out. 2003. ISSN 1559-1662. Disponível em: <http://doi.acm.org/10.1145-/965732.965736>.

PERKINS, C. *RFC 2002: IP Mobility Support*. [S.l.], out. 1996. `http://www.ietf.org/rfc/rfc2002.txt`.

PERKINS, C. E. Ip mobility support for ipv4. *RFC 3344*, August 2002.

ROSENBERG, J. et al. *SIP: Session Initiation Protocol*. IETF, June 2002. RFC 3261 (Proposed Standard). (Request for Comments, 3261). Updated by RFCs 3265, 3853, 4320, 4916, 5393, 5621. Disponível em: <http://www.ietf.org/rfc/rfc3261.txt>.

RYU, S.; MUN, Y. Performance Analysis for FMIPv6 Considering Probability of Predictive Mode Failure. In: . [S.l.: s.n.], 2009. p. 34 –38.

SHACKEL, B. Designing for People in the Age of Information. *Interact. Comput.*, Elsevier Science Inc., New York, NY, USA, v. 21, p. 325–330, December 2009. ISSN 0953-5438. Disponível em: <http://dl.acm.org/citation.cfm?id=1640535% -.1640741>.

SOLIMAN, H. et al. *Hierarchical Mobile IPv6 (HMIPv6) Mobility Management*. IETF, October 2008. RFC 5380 (Proposed Standard). (Request for Comments, 5380). Disponível em: <http://www.ietf.org/rfc/rfc5380.txt>.

STEWART, R. et al. *Stream Control Transmission Protocol (SCTP) Specification Errata and Issues*. IETF, April 2006. RFC 4460 (Informational). (Request for Comments, 4460). Disponível em: <http://www.ietf.org/rfc/rfc4460.txt>.

STOCKHAMMER, T.; LIEBL, G. On Practical Crosslayer Aspects in 3GPP Video Services. In: *Proceedings of the international workshop on Workshop on mobile video*. New York, NY, USA: ACM, 2007. (MV '07), p. 7–12. ISBN 978-1-59593-779-7. Disponível em: <http://doi.acm.org/10.1145/1290050.1290053>.

TANENBAUM, A. S. *Redes de Computadores*. trad. 4 ed. Rio de Janeiro: Elsevier, 2003.

VIDA, E. R.; COSTA, E. L. *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*. IETF, jun 2004. RFC 3810. (Request for Comments). Disponível em: <http://www.ietf.org/rfc-/rfc3810.txt>.

WAITZMAN, D.; PARTRIDGE, C. *Distance Vector Multicast Routing Protocol*. IETF, nov 1998. RFC 1075. (Request for Comments). Disponível em: <http://www.ietf.org/rfc/rfc1075-.txt>.

WANG, F. et al. IEEE 802.16e System Performance: Analysis and Simulations. In: *Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on*. [S.l.: s.n.], 2005. v. 2, p. 900 –904 Vol. 2.