



UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Warley Murici Valente Junior

**Arcabouço para Aprovisionamento de QoS e QoE em Redes
Sem Fio Heterogêneas WiMAX/Wi-Fi com Garantia de
Equidade entre Vazões**

DM: 11/2011

UFPA – ITEC – PPGEE

Belém – Pará – Brasil

2011

WARLEY MURICI VALENTE JUNIOR

**Arcabouço para Aprovisionamento de QoS e QoE em Redes
Sem Fio Heterogêneas WiMAX/Wi-Fi com Garantia de
Equidade entre Vazões**

Dissertação submetida à Banca Examinadora do Programa de Pós-Graduação em Engenharia Elétrica da UFPA para a obtenção do Grau de Mestre em Engenharia Elétrica com Ênfase em Computação Aplicada.

Orientador: Prof. Dr. Eduardo Coelho Cerqueira

DM: 11/2011

UFPA – ITEC – PPGEE

Belém – Pará – Brasil

2011

WARLEY MURICI VALENTE JUNIOR

**Arcabouço para Aprovisionamento de QoS e QoE em Redes
Sem Fio Heterogêneas WiMAX/Wi-Fi com Garantia de
Equidade entre Vazões**

Dissertação submetida à avaliação da banca examinadora aprovada pelo colegiado do programa de pós-graduação em engenharia elétrica da Universidade Federal do Pará e julgada adequada para obtenção do grau de mestre em engenharia elétrica com ênfase em computação aplicada.

Aprovada em: 17 de Agosto de 2011

Prof. Dr. Eduardo Coelho Cerqueira
(ORIENTADOR – UFPA)

Prof. Dr. João Crisóstomo W. A. Costa
(PPGEE – UFPA)

Prof. Dr. Antônio Jorge Gomes Abelém
(PPGCC – UFPA)

VISTO:

Prof. Dr. Marcus Vinícius Alves Nunes
(COORDENADOR – UFPA)

Aos meus pais pelo apoio,
incentivo e motivação durante todas
as fases da minha vida, me dando
forças incondicionais e
constantemente me mostrando o
caminho correto a ser seguido.

AGRADECIMENTOS

Primeiramente a Deus, por me possibilitar a existência e a vida, sem a qual nada disso seria possível. Graças a ele e a fé depositada conseguimos encontrar forças quando precisamos podendo sobrepujar caminhos difíceis.

Agradeço aos meus pais Warley Murici Valente e Maria José Dias Valente pelo amor incondicional, pela amizade, pelo carinho, apoio, incentivo e força que sempre me deram nos bons e maus momentos e aos meus irmãos Tiago Dias Valente e Glenda Dias Valente pelas palavras de carinho e pela motivação.

Agradeço também ao meu orientador Prof. Dr. Eduardo Coelho Cerqueira, pela confiança depositada, aceitando-me como orientando junto ao PPGEE e contribuindo assim, com importantes sugestões na escrita desta dissertação e finalização deste projeto.

Agradeço ao meu amigo Prof. Dr. Kelvin Lopes Dias, por ter acreditado no meu potencial, pela oportunidade de realizar o mestrado e pela amizade.

Aos meus amigos do GERCOM (Grupo de Estudos em Redes de Computadores e Comunicação Multimídia), em especial ao Jailton, Tássio e Thiago pela amizade e companheirismo.

Aos bons amigos que sempre me apoiaram e me ajudaram seja com conselhos, dicas ou palavras para que tudo se tornasse possível, em especial ao Jailton, Tássio, Ericson, meu primo Kevin e minha namorada Aline que sempre me motivaram nos momentos mais difíceis.

RESUMO

Recentemente a sociedade está vivendo em meio a um aglomerado de dispositivos móveis com múltiplas interfaces sem fio que permitem que clientes móveis domésticos e corporativos solucionem problemas rotineiros no menor espaço de tempo e em diversas localizações. Por outro lado, a disponibilidade de diversas tecnologias de acesso sem fio como: Wi-Fi (*Wireless Fidelity*), WiMAX (*Worldwide Interoperability for Microwave Access*) e LTE (*Long Term Evolution*), proporcionam um cenário heterogêneo com diversas oportunidades de conectividade para o usuário de dispositivos com múltiplas interfaces. Alguns dos desafios atuais no contexto da concepção da então denominada NGN (*Next Generation Networks*) são o desenvolvimento de arcabouços e mecanismos que viabilizem tanto o suporte adequado à Qualidade de Serviço (QoS – *Quality of Service*) e à Qualidade de Experiência (QoE – *Quality of Experience*) para aplicações multimídia, quanto a garantia de equidade entre vazões de diferentes aplicações em ambiente heterogêneo. Esta dissertação propõe um arcabouço para o provisionamento de QoS e QoE em redes sem fio heterogêneas formada por redes WiMAX e Wi-Fi. Especificamente, nossa solução provê mapeamento estático de QoS entre as classes de serviços WiMAX e categorias de acesso Wi-Fi. Além disso, nossa proposta também combina funcionalidades de equidade entre vazões com a solução de mapeamento, a fim de alcançar um bom compromisso tanto para o operador da rede, quanto para o usuário, através de uma nova metodologia de coleta e cálculo de vazão agregada e algoritmo de decisão de handover vertical. A avaliação quantitativa da proposta foi realizada através de modelos de simulação no ns-2 (*Network Simulator*). Os resultados de desempenho demonstram a eficiência da arquitetura em termos de métricas de QoS (vazão, vazão média e atraso) e QoE (PSRN - *Peak Signal to Noise Ratio*, SSIM - *Structural Similarity Index* e VQM - *Video Quality Metric*)

Palavras – chaves: MIH, WiMAX, Wi-Fi, Qualidade de Serviço (QoS), Qualidade de Experiência (QoE), Equidade.

ABSTRACT

Recently, the society is living amid an agglomerate of mobile devices with multiple wireless interfaces that allow mobile customers home and business solve routine problems in the shortest time and in several locations. On the other hand, the availability of various wireless access technologies such as Wireless Fidelity (Wi-Fi), Worldwide Interoperability for Microwave Access (WiMAX) and Long Term Evolution (LTE), provide a heterogeneous scenario with several connectivity opportunities for the user of devices with multiple interfaces. Some of the current challenges in framing the so called Next Generation Networks (NGN) are the development of frameworks and mechanisms that enable both adequate support for Quality of Service (QoS) and Quality of Experience (QoE) for multimedia applications, and assurance equity among throughputs of different applications in a heterogeneous environment. This dissertation proposes a framework for provisioning of QoS and QoE in heterogeneous wireless networks consisting of WiMAX and Wi-Fi networks. Specifically, our solution provides static mapping of QoS between classes of service WiMAX and access categories Wi-Fi. In addition, our proposal also combines features of equity among throughputs with the mapping solution in order to achieve a good tradeoff for both, for the network operator and for the user, through a new methodology for collection and calculation of aggregate throughput and algorithm for Vertical Handover Decision (VHD). The quantitative evaluation of the proposal was carried out using simulation models in Network Simulator (NS-2). The performance results demonstrate the efficiency of the framework in terms of QoS (throughput, average throughput and delay) and QoE (Peak Signal to Noise Ratio, Structural Similarity Index and Video Quality Metric) metrics.

Keywords: MIH, WiMAX, Wi-Fi, Quality of Service (QoS), Quality of Experience (QoE), Equity.

LISTA DE ABREVIATURAS E SIGLAS

AAA	Authentication, Authorization and Accounting
AC	Access Categories
AC_BE	Access Categories Best Effort
AC_BK	Access Categories Background
AC_VI	Access Categories Video
AC_VO	Access Categories Voice
ACK	Acknowledgment
AIFS	Arbitrary Inter-Frame Spacing
ANSI	American National Standards Institute
AP	Access Point
API	Application Programming Interfaces
ASN	Access Service Network
BE	Best Effort
BS	Base Station
BSS	Basic Service Set
CAC	Connection Admission Control
CBR	Constant Bit Rate
CCA	Clear Channel Assessment
CGU	Chang Gung University
CID	Connection Identifier
CoS	Class of Service
CPE	Customer Premises Equipments
CSF	Contrast Sensitivity Function
CSN	Connectivity Service Network
CTS	Clear To Send
CW	Contention Window
DCT	Discrete Cosine Transform
DLC	Data Link Control
DL-MAP	Downlink Bandwidth Allocation Map
DSA-REQ	Dynamic Service Addition Request
DSA-RSP	Dynamic Service Addition Reponse

DSCP	Differentiated Services Code Point
DSSS	Direct Sequence Spread Spectrum
EDCA	Enhanced Distributed Coordinated Access
ertPS	extended real-time Polling Service
ESS	Extended Service Set
ESSID	Extended Service Set Identifier
ETSI	European Telecommunications Standards Institute
FIFO	First In First Out
FTP	File Transfer Protocol
GTH	Get Throughput
HCCA	HCF Controlled Channel Access
HCF	Hybrid Coordination Function
HR-DSSS	High Rate Direct Sequence Spread Spectrum
HTTP	Hypertext Transfer Protocol
IBSS	Independent BSS
IEEE	Institute of Electrical and Electronics Engineers
IME	Integration Management Entity
IP	Internet Protocol
ISM	Industrial, Scientific, Medical
ISO	International Organization for Standardization
ITU	International Telecommunication Union
LAN	Local Area Network
LC	Local Contrast
LLC	Logical Link Control
LOS	Line-of-Sight
LTE	Long Term Evolution
MAC CPS	Common Part Sublayer
MAC CS	Service-Specific Convergence Sublayer
MAC SS	Security Sublayer
MAC	Media Access Control
MIB	Management Information Base
MICS	Media Independent Command Service
MIES	Media Independent Event Service
MIH SAP	MIH Service Access Point

MIHF	Media Independent Handover Function
MIHS	Media Independent Handover Services
MIHU	MIH users
MIIS	Media Independent Information Service
MIMO	Multiple-Input Multiple-Output
MN	Mobile Node
MOS	Mean Opinion Score
MS	Mobile Station
mSIR	maximum Signal-to-Interference Ratio
NAP	Network Access Provider
NGN	Next Generation Networks
NIST	National Institute of Standards and Technology
NLOS	No Line-Of-Sight
nrtPS	non real-time Polling Service
NSP	Network Service Provider
O-DRR	Opportunistic Deficit Round Robin
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
PDP	Policy Decision Point
PDU	Protocol Data Units
PEP	Policy Enforcement Point
PF	Persistence Factor
PHB	Per-Hop Behavior
PHY	Physical
PKM	Privacy Key Management
PLCP	Physical Layer Convergence Protocol
PMD	Physical Medium Dependent
PMP	Point-to-Multipoint
PoA	Point of Attachment
PoS	Point of Service
PSNR	Peak Signal to Noise Ratio
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network

RAT	Radio Access Technology
RR	Round Robin
RTH	Repository Throughput
rtPS	real-time Polling Service
RTS	Request to Send
SAP	Service Access Point
SFID	Service Flow Identifier
SS	Subscriber Station
SSIM	Structural Similarity Index
STA	Stations
TKN	Telecommunication Networks Group
TRS	Temporary Removal Scheduler
TxOP	Transmission Opportunity
UGS	Unsolicited Grant Service
UL-MAP	Uplink Bandwidth Allocation Map
UMTS	Universal Mobile Telecommunication System
UP	User Priorities
VDA	Virtual Device Agent
VHD	Vertical Handover Decision
VHTC	Vertical Handoff Translation Center Architecture
VQM	Video Quality Metric
WEP	Wired Equivalent Privacy
WFQ	Weighted Fair Queuing
Wi-Fi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WPA	WiFi Protected Access
WRR	Weighted Round Robin
xDSL	Digital Subscriber Line

SUMÁRIO

1. Introdução.....	17
1.1 Motivação.....	17
1.2 Objetivos.....	20
1.3 Estrutura da Dissertação.....	20
2. Tecnologias de Acesso Sem Fio Wi-Fi e WiMAX e o <i>Framework</i> MIH.....	22
2.1 Padrão IEEE 802.11.....	22
2.1.1 O Padrão e suas Emendas.....	23
2.1.2 Arquitetura e Protocolos Wi-Fi.....	25
2.1.3 QoS na Rede Wi-Fi.....	28
2.2 Padrão IEEE 802.16.....	31
2.2.1 O Padrão e suas Emendas.....	32
2.2.2 Arquitetura e Protocolos WiMAX.....	34
2.2.3 QoS na Rede WiMAX.....	38
2.3 Padrão IEEE 802.21.....	43
2.4 Resumo do Capítulo.....	48
3. Trabalhos Relacionados.....	49
3.1 Mapeamento de QoS em Redes Sem Fio Heterogêneas.....	49
3.2 Métodos para o Suporte a Balanceamento de Carga.....	52
3.3 Resumo do Capítulo.....	54
4. Aprovisionamento de QoS/QoE e Equidade entre Vazões.....	55
4.1 Qualidade de Experiência (QoE).....	55
4.1.1 Métricas Objetivas.....	56
4.1.2 Métricas Subjetivas.....	58
4.2 Arcabouço Proposto.....	59
4.2.1 Arquitetura para Mapeamento de QoS.....	59
4.2.2 Algoritmo VHD.....	61
4.2.3 Metodologia para Obtenção de Vazão Agregada.....	64
4.3 Resumo do Capítulo.....	68
5. Avaliação do Arcabouço.....	69

5.1 Parâmetros de Simulação e Topologia da rede.....	69
5.2 Análise das Simulações	72
5.2.1 Cenário 1	73
5.2.2 Cenário 2	79
5.3 Resumo do Capítulo	84
6. Conclusão	85
7. Referências Bibliográficas.....	87
Anexos.....	93

LISTA DE FIGURAS

Figura 1 – Cenário de mobilidade de Nodos Móveis (MNs) entre a cobertura da Estação Base (BS) e Pontos de Acessos (APs).....	19
Figura 2 – Arquitetura de uma rede infraestruturada IEEE 802.11.....	25
Figura 3 – Arquitetura de uma rede ad-hoc IEEE 802.11.	26
Figura 4 – Arquitetura de protocolo e gerenciamento do IEEE 802.11.	27
Figura 5 – Filas de prioridades proposto pelo EDCA 802.11e.....	29
Figura 6 – Acesso ao meio através do método EDCA.	30
Figura 7 – Conectividade PMP para usuários fixos e móveis.	31
Figura 8 – Arquitetura de rede proposta pelo WiMAX Fórum.	35
Figura 9 – Arquitetura da pilha de protocolos do WiMAX.....	37
Figura 10 – Arquitetura completa de QoS em redes WiMAX.	40
Figura 11 – Estrutura de um quadro de gerenciamento WiMAX.....	41
Figura 12 – Modelo de referência do framework MIH.	44
Figura 13 – Arquitetura geral do IEEE 802.21.....	45
Figura 14 – Escopo do IEEE 802.21 durante o handover.	47
Figura 15 – Arquitetura geral para mapeamento de QoS.	60
Figura 16 – Pseudocódigo do Algoritmo VHD.....	62
Figura 17 – Pseudocódigo da Metodologia GTH.....	64
Figura 18 – Pseudocódigo da Metodologia RTH.	65
Figura 19 – Sinalização durante o <i>handover</i> vertical.	66
Figura 20 – Topologia genérica da rede para simulação.	72
Figura 21 – Representação gráfica do Cenário 1.....	73
Figura 22 – Vazão média do três fluxos sem mapeamento de QoS.	74
Figura 23 – Vazão média do três fluxos com mapeamento de QoS.....	75
Figura 24 – PSNR do vídeo transmitido com e sem o mapeamento de QoS.....	77
Figura 25 – SSIM do vídeo transmitido com e sem o mapeamento de QoS.....	77
Figura 26 – VQM do vídeo transmitido com e sem o mapeamento de QoS.....	78
Figura 27 – Representação gráfica do Cenário 2.....	80
Figura 28 – Vazão média sem o algoritmo VHD e metodologia.	81
Figura 29 – Vazão média com o algoritmo VHD e metodologia.....	82
Figura 30 – Média de atraso de pacotes sem o algoritmo VHD e metodologia.....	83
Figura 31 – Média de atraso de pacotes com o algoritmo VHD e metodologia.....	84

LISTA DE TABELAS

Tabela 1 – Valores de Classificação do PSNR.....	56
Tabela 2 – Escala de Valores MOS.....	59
Tabela 3 – Parâmetros usados nas simulações.....	70
Tabela 4 – Configuração de parâmetros de ACs.....	71
Tabela 5 – Configuração de parâmetros de CoSs.....	71
Tabela 6 – Resultados para vazão média e respectivo ganho para o Cenário 1.....	75
Tabela 7 – Parâmetros de simulação de tráfego de vídeo.....	76
Tabela 8 – Resultados das métricas de QoE para o tráfego de vídeo.....	78
Tabela 9 – Resultados para vazão média e respectivo ganho para o Cenário 2.....	82

LISTA DE QUADROS

Quadro 1 - Principais Emendas do IEEE 802.11.....	23
Quadro 2 – Mapeamento entre prioridades de usuário e categorias de acesso.....	29
Quadro 3 – Principais Emendas da família IEEE 802.16.....	34
Quadro 4 – Classes de serviços WiMAX e parâmetros de QoS associados.	39
Quadro 5 – Relação dos trabalhos relacionados de acordo com seus assuntos chave. ..	53
Quadro 6 – Mapeamento de QoS entre as redes WiMAX e Wi-Fi.	61
Quadro 7 – <i>Frames</i> do vídeo “ <i>Highway</i> ”.	79

1. Introdução

Este Capítulo descreve a motivação, os objetivos e a organização da dissertação, explanando sobre os problemas encontrados na manutenção da qualidade de serviço prestada para os usuários móveis em redes heterogêneas, uma breve introdução sobre o trabalho e a proposta de um arcabouço para solucionar estes problemas. Introduce assuntos relacionados à mobilidade, *handover*, qualidade de serviço, qualidade de experiência e equidade entre vazões, assim como descreve a proposta desta dissertação.

1.1 Motivação

Recentemente a sociedade está vivendo em meio a um aglomerado de dispositivos portáteis com múltiplas interfaces sem fio que permitem que clientes móveis domésticos e corporativos solucionem problemas rotineiros no menor espaço de tempo e em diversas localizações. Além disso, a demanda por estes dispositivos tais como smartphones, tablets, laptops e notebooks estão com um índice de crescimento nas vendas acima do esperado pelas indústrias. Conseqüentemente, a exigência por serviços de qualidade prestados por operadores de rede também está aumentando.

Por outro lado, a disponibilidade de diversas tecnologias de acesso sem fio como: Wi-Fi (*Wireless Fidelity*), WiMAX (*Worldwide Interoperability for Microwave Access*), UMTS (*Universal Mobile Telecommunication System*) e LTE (*Long Term Evolution*), proporcionam um cenário heterogêneo com diversas oportunidades de conectividade para o usuário de dispositivos com múltiplas interfaces.

Particularmente, duas tecnologias de acesso sem fio, amplamente difundidas, podem prover suporte à qualidade de serviço (QoS - *Quality of Service*) para o acesso à internet móvel. O IEEE 802.11e (IEEE 802.11e 2005), um padrão para garantia de QoS em redes Wi-Fi, permite cobertura sem fio local. Por outro lado, o IEEE 802.16e (IEEE 802.16e 2005), suporta QoS no âmbito de redes metropolitanas WiMAX.

A emenda IEEE 802.11e, inclui o HCF (*Hybrid Coordination Function*) que introduz dois modos de operação, o EDCA (*Enhanced Distributed Coordinated Access*), um mecanismo baseado em contenção e o HCCA (*HCF Controlled Channel Access*), um mecanismo livre de contenção. O EDCA (IEEE draft p802.11e d13.0 2005) define basicamente quatro categorias de acesso AC (*Access Categories*) na camada

MAC (*Media Access Control*), conhecidos como (AC_VO) para tráfego de voz, (AC_VI) para tráfego de vídeo, (AC_BE) para tráfego de melhor esforço, tal como HTTP (*Hypertext Transfer Protocol*) e (AC_BK) para tráfego de fundo, tal como FTP (*File Transfer Protocol*), classificados da maior para menor prioridade, respectivamente. Cada categoria de acesso possui uma única fila de transmissão e parâmetros particulares, tais como limiares superiores e inferiores da janela de contenção, (CW_{Max} e CW_{Min} , respectivamente), espaçamento inter-quadros arbitrário AIFS (*Arbitrary Inter-Frame Spacing*) e oportunidade de transmissão TxOP (*Transmission Opportunity*), que são configurados para dar diferentes prioridades de acesso ao meio. Tráfegos com ACs de alta prioridade possuem valores menores de CW_{Max} , CW_{Min} e AIFS em relação a tráfegos de baixa prioridade.

O WiMAX, implementa suporte à QoS na camada MAC para facilitar a interação com o gerenciamento de recursos de rádio e camada física. Seu *framework* de QoS adota cinco classes de serviços ou CoS (*Class of Service*): UGS (*Unsolicited Grant Service*), rtPS (*real-time Polling Service*), ertPS (*extended real-time Polling Service*), nrtPS (*non real-time Polling Service*) e BE (*Best Effort*). Cada CoS possui um conjunto de parâmetros de QoS que devem ser inclusos na definição do fluxo de serviço quando a classe de serviço está habilitada para um fluxo de serviço. Os principais parâmetros são: prioridade de tráfego, latência máxima, jitter, máxima e mínima taxa de dados e máximo atraso. Após ser admitido em uma conexão, o fluxo de serviço receberá da estação base um CID (*Connection Identifier*) para sua devida classificação (Sekercioglu et al 2009).

Mesmo com suporte à QoS presente em cada uma dessas tecnologias, não há garantias que em um ambiente integrado e heterogêneo seja possível manter a qualidade e continuidade dos serviços à medida que o usuário muda, por exemplo, de uma estação base (BS - *Base Station*) WiMAX para um ponto de acesso (AP - *Access Point*) Wi-Fi, bem como quanto a visão centrada no usuário através do suporte adequado à Qualidade de Experiência (QoE - *Quality of Experience*) para aplicações multimídia. Além de que nessas tecnologias, não há um controle de distribuição de tráfego diretamente relacionada às prioridades de serviços entre dispositivos móveis em momentos de congestionamento no acesso ao meio sem fio.

Alinhado com a necessidade de permitir conectividade transparente para usuários em movimento, servidos por diversas tecnologias sem fio, o IEEE (*Institute of Electrical and Electronics Engineers*) desenvolveu e aprovou o novo padrão IEEE

802.21 ou MIHS (*Media Independent Handover Services*) (IEEE draft p802.21 d11.0 2008). O MIHS ou simplesmente MIH, foi projetado para aperfeiçoar a integração e mobilidade entre redes sem fio de diferentes tecnologias, bem como para permitir o *handover* horizontal e vertical, isto é, a troca de PoA (*Point of Attachment*) entre tecnologias similares e distintas, respectivamente. O ambiente heterogêneo e a mobilidade dos Nodos Móveis (MNs – *Mobile Nodes*) é ilustrada na Figura 1.

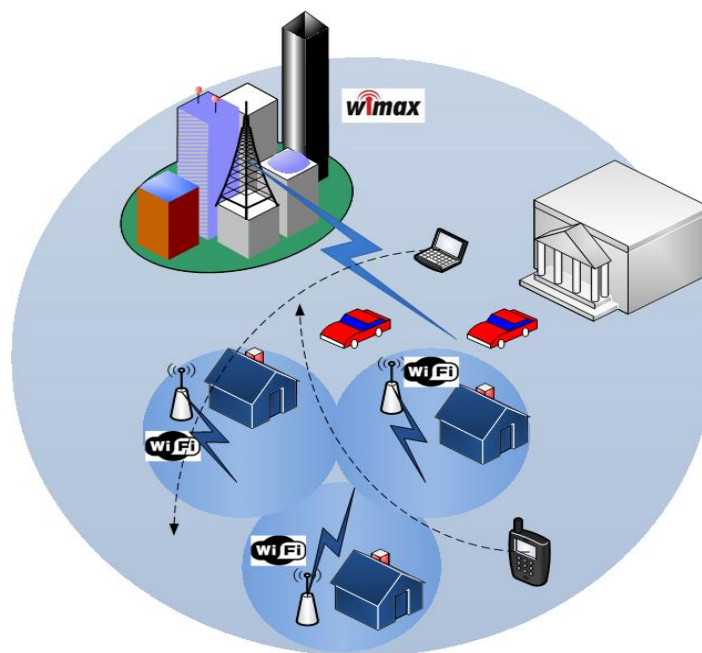


Figura 1 – Cenário de mobilidade de Nodos Móveis (MNs) entre a cobertura da Estação Base (BS) e Pontos de Acessos (APs).

Para realizar estes objetivos, o MIH conta com um conjunto de eventos de sinalização, gatilhos e serviços, unificados para qualquer tecnologia, que disponibilizam informação de camadas inferiores (MAC e Física) para as camadas superiores (Camada de Aplicação) da pilha de protocolos.

O MIH também inclui em seu framework, um mecanismo de política de handover na qual, faz o uso de informações de redes vizinhas para tomar a decisão de handover. Com a inclusão de um algoritmo inteligente, o MN equipado com MIH poderá então ter mais controle na decisão de realizar handover ou não para uma rede alvo.

1.2 Objetivos

Esta dissertação propõe um arcabouço para o provisionamento de QoS e QoE em redes sem fio heterogêneas formada por redes WiMAX e Wi-Fi. Especificamente, nossa solução provê mapeamento estático de QoS entre as classes de serviços WiMAX e categorias de acesso Wi-Fi.

Além disso, nossa proposta também combina funcionalidades que visa garantir a equidade entre vazões de diferentes aplicações com a solução de mapeamento, a fim de alcançar um bom compromisso tanto para o operador da rede, quanto para o usuário, através de um novo algoritmo de decisão de *handover* vertical (VHD - *Vertical Handover Decision*).

Além de utilizarmos o MIH, como tradicionalmente usado na literatura, para facilitar o *handover* e descoberta de novas PoAs, também desenvolvemos nesta dissertação, uma nova metodologia baseada no MIH para a obtenção de vazão agregada tanto da rede atual, quanto da rede alvo, para auxiliar o algoritmo VHD na comparação com limiares no processo de decisão de *handover*.

O arcabouço viabiliza o *handover* vertical transparente entre tecnologias distintas, a continuidade da garantia de QoS e QoE durante a mobilidade do MN entre PoAs e a equidade entre vazões de diferentes aplicações em ambiente heterogêneo de modo que serviços de maior e menor prioridade, como vídeo e HTTP respectivamente, tenham a qualidade de serviço garantida mesmo em situações de congestionamento na rede em que estão associados.

1.3 Estrutura da Dissertação

Este Capítulo apresentou uma breve introdução aos aspectos de mobilidade e garantia na qualidade de serviço encontrados em redes WiMAX e Wi-Fi em integração com o MIH, e das propostas desta dissertação, trazendo consigo uma pequena iniciação dos assuntos que serão discutidos posteriormente.

O Capítulo 2 trará conceitos sobre redes Wi-Fi e WiMAX, descrevendo suas definições, seus principais padrões homologados, protocolos e arquiteturas, e aprofundará o conceito de qualidade de serviço em ambas as tecnologias. Além do mais, abordamos os principais conceitos e objetivos do framework MIH, sua arquitetura e serviços MIH.

O Capítulo 3 descreverá alguns trabalhos relacionados com o tema proposto por esta dissertação descrevendo resumidamente seus pontos fortes e criticando seus pontos fracos.

O Capítulo 4 descreve o arcabouço proposto pela dissertação, os módulos de QoS WiMAX/Wi-Fi integrados ao módulo NIST Mobility para prover o processo de mapeamento estático de QoS entre as tecnologias WiMAX e Wi-Fi em conjunto com o MIH, o mecanismo baseado no MIH para a medição e coleta das vazões agregadas das redes atual e alvo e por fim, o algoritmo VHD que com o auxílio do MIH e deste mecanismo é capaz de prover a equidade entre vazões de diferentes aplicações.

O Capítulo 5 descreve a avaliação da proposta analisando os cenários e os resultados deste trabalho, explicando os resultados obtidos com a utilização do arcabouço proposto.

Posteriormente se tem as conclusões finais desta dissertação assim como as referências utilizadas e os anexos correspondentes ao código utilizado no simulador.

2. Tecnologias de Acesso Sem Fio Wi-Fi e WiMAX e o Framework MIH

Este capítulo descreve conceitos dos dois principais padrões para tecnologia de acesso sem fio, adotados no arcabouço proposto: o IEEE 802.11 (Wi-Fi) e IEEE 802.16 (WiMAX). Discutimos suas emendas, seus protocolos, arquitetura e suporte a qualidade de serviço. Discutimos também, os principais conceitos, objetivos, arquitetura e serviços do framework MIH, a qual é à base do desenvolvimento desta dissertação.

2.1 Padrão IEEE 802.11

Para definir um padrão para as redes locais sem fio (WLANs - *Wireless Local Area Network*), o IEEE criou o “*Wireless Local-Area Networks Standard Working Group, IEEE Project 802.11*”. Este grupo tinha como objetivo definir especificações e padrões para as redes que possuíam como meio de transmissão as ondas de rádio ou infravermelho.

Mais tarde, uma organização sem fins lucrativos foi formada com o objetivo de certificar a interoperabilidade de produtos baseados no padrão IEEE 802.11, a *Wi-Fi Alliance* (Wi-Fi Alliance, 2011). Através de um severo e abrangente teste de interoperabilidade, o programa de certificação da *Wi-Fi Alliance* assegura que produtos de diferentes fabricantes funcionam sem problemas.

O padrão IEEE 802.11 (IEEE 802.11, 1999) é caracterizada como a família mais famosa entre WLANs. O principal objetivo deste padrão é a especificação de uma WLAN simples e robusta que oferece serviços em tempo limitado e assíncrono. A camada MAC deve ser capaz de operar com múltiplas camadas físicas, cada uma das quais apresenta uma diferente sensibilidade ao meio e característica de transmissão.

Os dispositivos do padrão operam na faixa não licenciada do espectro eletromagnético, ou seja, nas bandas ISM (*Industrial, Scientific, Medical* - Médica, Científica, Industrial). Existem três bandas ISM em uso atualmente: a banda ISM de 902-908 MHz, a banda ISM de 2,4-2,485 GHz e a banda ISM de 5,734 GHz a 5,860 GHz, sendo que a banda ISM de 2,4 GHz é a mais comumente usada, não só nas WLANs, mas também em muitos outros dispositivos que usam a faixa não licenciada do espectro. Geralmente, os aparelhos que funcionam nas bandas ISM têm

transmissores de baixa potência e não são capazes de cobrir mais do que algumas dezenas ou centenas de metros, como é o caso das WLANs (Silva & Dias, 2011).

2.1.1 O Padrão e suas Emendas

O IEEE publicou inicialmente em 1997 o padrão 802.11 e em 1999 este padrão foi atualizado passando a ser aceito também pelo ANSI (*American National Standards Institute*) e pela ISO (*International Organization for Standardization*). Este padrão define as funções e os serviços requeridos pelos dispositivos que operam de acordo com o IEEE 802.11. Entre estas funções e serviços destacam-se os aspectos referentes ao acesso ao meio, à mobilidade das estações, ao processo de *handoff*, à camada física e à privacidade e segurança nas transmissões dos dispositivos de rede local sem fio. O Quadro 1 sumariza as principais emendas ao padrão existentes atualmente.

Quadro 1 - Principais Emendas do IEEE 802.11

Padrão	Data Final de Homologação	Propósito	Faixa de Frequência
802.11-1999	18-03-1999	Padrão IEEE para Especificação Wireless LAN do Controle de Acesso ao Meio (MAC) e Camada Física (PHY)	2 Mbps DSSS PHY @2.4 GHz
802.11a	16-09-1999	Maior Velocidade pela Extensão da Camada Física (PHY)	54 Mbps OFDM PHY @5.0 GHz
802.11d	14-06-2001	Operação em Domínio Complementar de Regulamentação. Permite que Dispositivos Estejam em Conformidade com Exigências Regionais	11-54 Mbps @5.0 GHz
802.11e	22-09-2005	Aperfeiçoamento da Camada MAC para Oferecer Suporte a QoS	11-54 Mbps @2.4 GHz
802.11g	12-06-2003	Extensão para Maior Taxa de Dados	54 Mbps OFDM PHY @2.4GHz
802.11i	24-06-2004	Aperfeiçoamento de Segurança na Camada MAC	54 Mbps OFDM PHY @2.4GHz
802.11n	11-09-2009	Aperfeiçoamento para Altas Taxas de Transmissão	600 Mbps MIMO PHY @2.4GHz e 5.0GHz

Originalmente, o padrão IEEE 802.11 estabeleceu transmissões a taxas de 1 Mbps a 2 Mbps na faixa de 2.4 GHz. Posteriormente, este padrão foi revisado e no mesmo ano de 1999 o IEEE publicou o padrão 802.11a e o padrão 802.11b.

A primeira extensão do projeto para o padrão 802.11a, foi iniciada em Setembro de 1997. Ela adiciona a codificação OFDM (*Orthogonal Frequency Division Multiplexing*) que suporta taxa de dados de até 54 Mbps. O padrão 802.11b opera em frequência de 2.4 GHz e suporta taxas de transmissão de até 11Mbps. Este padrão utiliza codificação DSSS (*Direct Sequence Spread Spectrum*) em três canais não sobrepostos ou HR-DSSS (*High Rate Direct Sequence Spread Spectrum*) (Eiras & Zucchi, 2009).

Uma vez que o 802.11a opera com frequência de transmissão de 5 GHz, a comunicação com dispositivos 802.11 original se tornou impossível. Esta falta de interoperabilidade levou a formação do padrão 802.11g, que introduz os benefícios do OFDM para a faixa de 2.4 GHz. Como a frequência estendida do 802.11g provê compatibilidade de sinalização com DSSS, uma fácil migração de dispositivos 802.11 para 802.11g tornou-se possível.

O padrão 802.11d, teve seu último aperfeiçoamento aprovado em 2001, a qual introduz suporte às questões regulatórias internacionais, permitindo que produtos 802.11 operem de forma compatível às diversas normas nacionais. Bem antes da homologação final do 802.11d, foi aprovado em Março de 2000 a primeira emenda do 802.11e. O objetivo do projeto original 802.11e, previa melhorias gerais do padrão WLAN. Melhorias de rendimento, suporte para qualidade de serviço (QoS) e aperfeiçoamento de segurança foram seus elementos principais. No entanto, já em 2001, o algoritmo de criptografia WEP (*Wired Equivalent Privacy*) foi quebrado por um ataque. Assim, melhorias de segurança foram deslocadas para o então padrão 802.11i.

Após intensos debates, o padrão final 802.11e finalmente foi aprovado em 2005, para dar suporte a QoS. Com um novo esquema de acesso ao meio, o 802.11e provê a função de coordenação híbrida (HCF - *Hybrid Coordination Function*), onde o termo híbrido refere-se a duas versões de protocolo MAC com controle centralizado, implementado pelo HCCA (*HCF Controlled Channel Access*) e distribuído, implementado pelo EDCA (*Enhanced Distributed Channel Access*) (Hiertz et. al., 2010).

Como abordado antes, o método de criptografia WEP apresentava diversos problemas de segurança, e devido a este fato, a *WiFi Alliance* desenvolveu outro

método de criptografia conhecido como WPA (*WiFi Protected Access*). E com a finalização do IEEE 802.11i, uma versão melhorada do WPA, (WPA2), foi integrada no padrão. O 802.11i trata também de métodos de autenticação considerados inseguros do padrão original, sistemas abertos e chave de autenticação compartilhada, bem como, tem incorporado o método de autenticação 802.1X que é usado em todos os padrões da família IEEE 802.

Ao contrário dos aprimoramentos anteriores, o padrão 802.11n visa aumentar a vazão da camada MAC do 802.11, ao invés de simplesmente aumentar a taxa de dados da camada Física (PHY). Desta forma o 802.11n oferece vazão máxima de até 600 Mbps. Este objetivo pode ser alcançado, principalmente devido à implementação da codificação MIMO (*Multiple-Input Multiple-Output*). Esta codificação permite que matrizes de até quatro antenas possibilitem a multiplexação espacial e formação de feixe (Kuran & Tugcu, 2007).

2.1.2 Arquitetura e Protocolos Wi-Fi

Redes Wi-Fi podem apresentar dois diferentes sistemas de arquitetura básica: infraestruturada ou ad-hoc. A Figura 2 ilustra os componentes da parte infraestruturada e sem fio conforme especificado no IEEE 802.11.

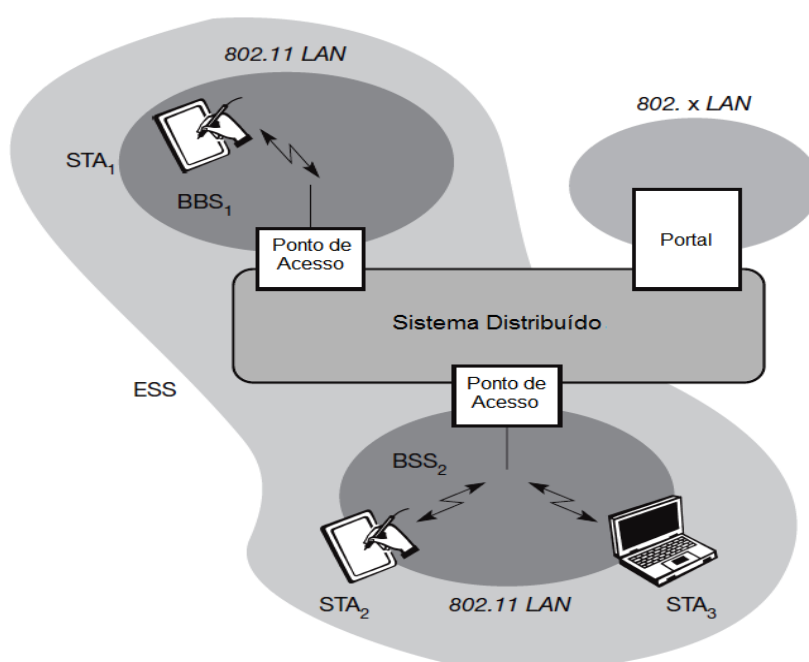


Figura 2 – Arquitetura de uma rede infraestruturada IEEE 802.11.

Vários nodos, chamados de Estações (STA_i - *Stations*), estão conectados a **pontos de acesso** (AP). Estações são terminais com mecanismos de acesso ao meio sem fio e comunicação via rádio com o AP. Estações e AP que estejam dentro da mesma cobertura de rádio forma um **conjunto básico de serviços** (BSS_i - *Basic Service Set*). A Figura 2 mostra como exemplo duas $BSSs$ (BSS_1 e BSS_2) que estão conectadas através de um **sistema distribuído**. Um sistema distribuído conecta várias $BSSs$ via AP para formar uma única rede e, portanto estende a área de cobertura sem fio. Esta rede passa então a ser chamada de **conjunto estendido de serviços** (ESS - *Extended Service Set*) e tem seu próprio identificador, o ESSID. O ESSID é o “nome” da rede e é usado para separar diferentes redes. O sistema distribuído conecta as redes sem fio através de APs com um **portal**, que constitui a unidade de interoperabilidade com outras LANs.

Estações podem selecionar um AP e associar a eles. Os APs suportam *roaming*¹, o sistema distribuído trata da transferência de dados entre diferentes APs. Os APs fornecem sincronização dentro de uma BSS , suporte a gerenciamento de energia, e podem controlar o acesso ao meio para suporte a serviços de tempo-limitado.

Além de redes infraestruturadas, o IEEE 802.11 permite a construção de redes ad-hoc entre estações, formando uma ou mais **$BSSs$ independentes** ($IBSS$ – *Independent BSS*) conforme mostrado na Figura 3.

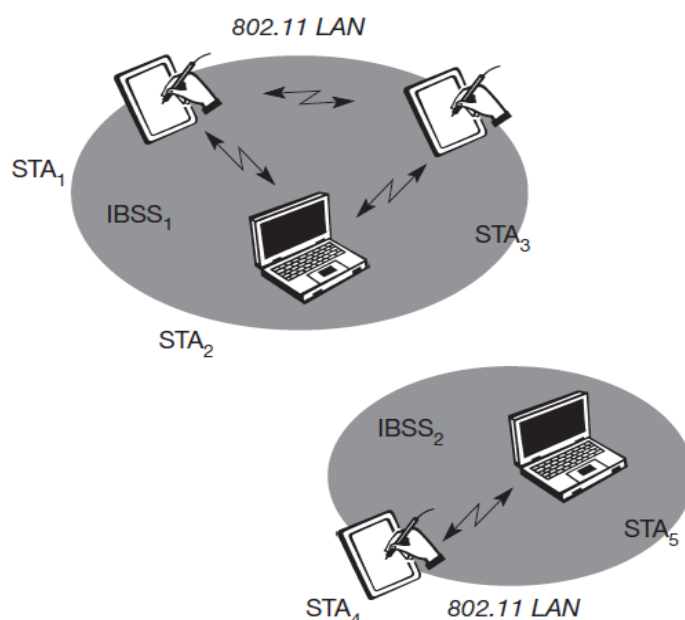


Figura 3 – Arquitetura de uma rede ad-hoc IEEE 802.11.

¹ *Roaming* é um termo empregado em tecnologias de rede sem fio, no que se refere à capacidade de um usuário obter conectividade em áreas fora da localidade geográfica onde está registrado, ou seja, conectividade através de outra rede onde é visitante.

Neste caso, uma IBSS compreende um grupo de estações usando a mesma frequência de rádio. Estações STA_1 , STA_2 e STA_3 estão na $IBSS_1$, STA_4 e STA_5 na $IBSS_2$. Isto significa, por exemplo, que a STA_3 pode se comunicar diretamente com a STA_2 mas não com a STA_5 . Várias IBSSs podem ser formadas pelas distâncias entre IBSSs ou utilizando diferentes portadoras de frequências (assim as IBSSs podem se sobrepor fisicamente).

Da mesma forma que LANs 802.x, o padrão IEEE 802.11 abrange apenas a camada física (PHY) e camada de controle de acesso ao meio (MAC). Conforme pode ser visto na Figura 4, a camada de controle de enlace de dados (DLC - *Data Link Control*) é subdividida pelo controle de enlace lógico (LLC - *Logical Link Control*) e subcamada de controle de acesso ao meio (MAC) ou também denominada camada MAC. O LLC cobre as diferenças das camadas de controle de acesso ao meio, necessárias para diferentes meios de comunicação. As tarefas básicas da camada MAC inclui acesso ao meio, fragmentação de dados e criptografia.

A camada física é subdividida pelo protocolo de convergência da camada física (PLCP - *Physical Layer Convergence Protocol*) e a subcamada dependente de meio físico (PMD - *Physical Medium Dependent*).

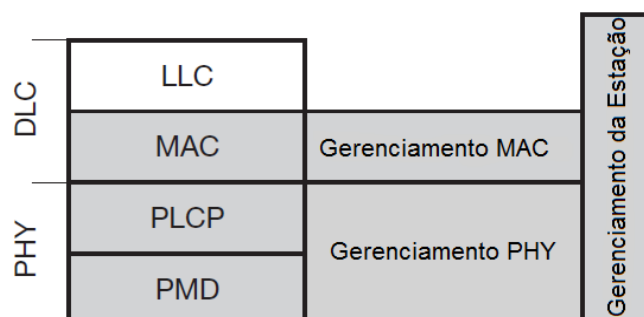


Figura 4 – Arquitetura de protocolo e gerenciamento do IEEE 802.11.

A subcamada PLCP provê sinal de detecção de portadora, chamada de avaliação de canal livre (CCA - *Clear Channel Assessment*), e fornece um ponto de acesso de serviço comum PHY (SAP - *Service Access Point*) independente da tecnologia de transmissão. A subcamada PMD trata da modulação e codificação/decodificação de sinais.

Além das subcamadas de protocolos, o padrão especifica as camadas de gerenciamento e o gerenciamento da estação. O gerenciamento MAC suporta a associação e reassociação de uma estação para um ponto de acesso e *roaming* entre

diferente diferentes pontos de acesso. Ele também controla os mecanismos de autenticação, criptografia, sincronização de uma estação em relação a um ponto de acesso e gerenciamento de energia para economizar bateria. Os principais objetivos do gerenciamento PHY incluem a sintonização de canais e manutenção PHY. E finalmente, o gerenciamento da estação que interage com ambas as camadas de gerenciamento e é responsável por funções adicionais da camada superior (Schiller, 2003).

2.1.3 QoS na Rede Wi-Fi

Conforme apresentado anteriormente na seção 2.1.1, o padrão IEEE 802.11e (IEEE 802.11e, 2005) trata de melhorias da camada MAC, a fim de oferecer suporte a QoS. Desta forma, o padrão define o HCF (*Hybrid Coordination Function*) que introduz dois modos de operação, o EDCA (*Enhanced Distributed Coordinated Access*), um mecanismo baseado em contenção e o HCCA (*HCF Controlled Channel Access*), um mecanismo livre de contenção. Nesta seção, estudaremos com mais detalhe o EDCA (IEEE draft p802.11e d13.0, 2005), pois este é atualmente a solução mais utilizada por operadoras de redes, instituições acadêmicas, além de ser mais eficaz e eficiente, na concessão de prioridades de acesso ao meio.

A fim de permitir o tratamento do tráfego de modo a prover prioridades no acesso ao meio, o EDCA 802.11e, utiliza oito prioridades de usuário (UPs - *User Priorities*), as quais estão ilustradas na Figura 5. Tráfegos de dados de oito UPs provenientes das camadas superiores são mapeados para quatro diferentes categorias de acesso (ACs): para voz (AC_VO), vídeo (AC_VI), melhor esforço (AC_BE) e tráfego de fundo (AC_BK), classificados da maior para menor prioridade, respectivamente. O Quadro 2 descreve resumidamente este mapeamento. Cada categoria de acesso é equipada com uma única fila de transmissão, cuja estratégia utilizada de fila é a de “primeira a entrar, primeiro a sair” (FIFO – *First In First Out*) (Mangold et. al., 2002).

Cada AC possui parâmetros específicos, tais como limiares superiores e inferiores da janela de contenção (CW - *Contention Window*), CW_{Max} e CW_{Min} respectivamente, espaçamento inter-quadros arbitrário (AIFS - *Arbitrary Inter-Frame Spacing*), oportunidade de transmissão (TxOP - *Transmission Opportunity*) e fator de persistência (PF - *Persistence Factor*) (Junior et. al., 2011).

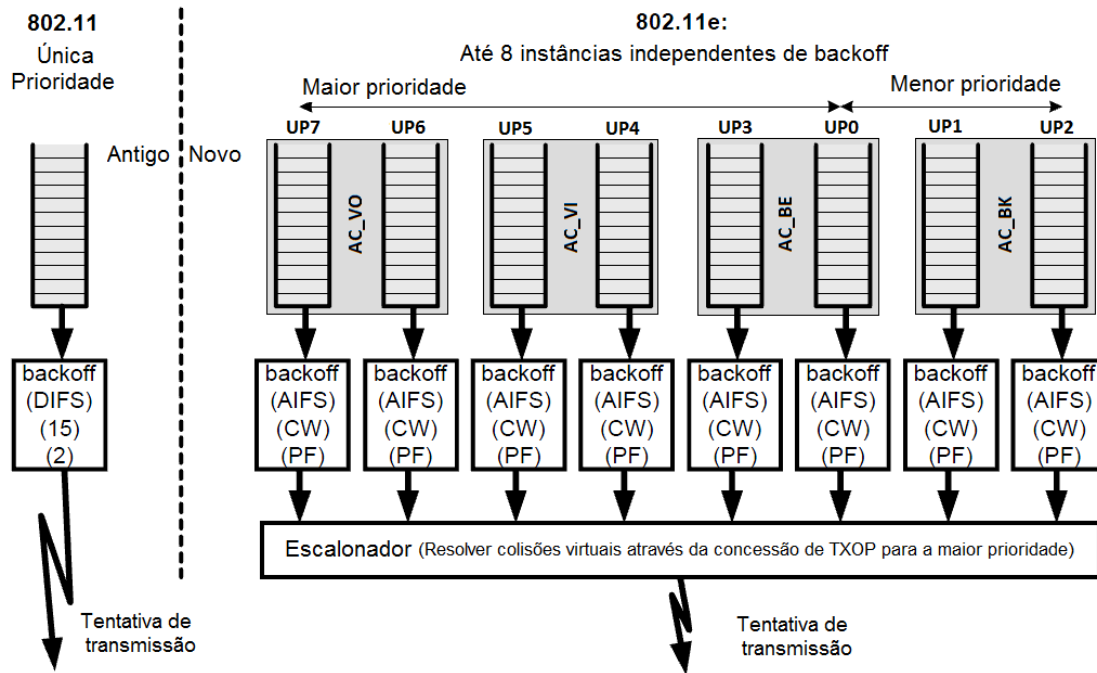


Figura 5 – Filas de prioridades proposto pelo EDCA 802.11e.

Um exemplo de mapeamento no EDCA seria um quadro de voz direcionado para AC_VO, que possui valores de CW e AIFS menores, aumentando a probabilidade de acesso ao meio, antes de um quadro HTTP direcionado para AC_BE.

Quadro 2 – Mapeamento entre prioridades de usuário e categorias de acesso.

Prioridade do Usuário	Descrição	Exemplo	Categoria de Acesso
1	BK (Tráfego de Fundo)	FTP	AC_BK
2	BK (Tráfego de Fundo)		AC_BK
0	BE (Melhor Esforço)	HTTP e e-mails	AC_BE
3	EE (Vídeo/Excelente Esforço)		AC_BE
4	CC (Vídeo/Carga Controlada)	Streaming de Vídeo	AC_VI
5	VI (Vídeo)		AC_VI
6	VO (Voz)	VoIP	AC_VO
7	RC (Rede de Controle)		AC_VO

O parâmetro de janela de contenção define um tempo aleatório que uma estação aguarda antes de acessar o meio. A estação que deseja transmitir, aguarda um tempo aleatório que está entre 0 e CW valores de *TimeSlot*, escolhido aleatoriamente no intervalo. O intervalo é definido por CW_{Min} e CW_{Max} , em que o CW_{Min} é o valor mínimo escolhido na primeira tentativa de transmissão. Caso corra falha na transmissão (não recebimento do frame ACK) o valor de CW é dobrado e uma nova tentativa de

transmissão é feita. Caso ocorram falhas sucessivas na transmissão, o valor de CW é dobrado até o limite definido pelo valor CW_{Max} . Em caso de sucesso na transmissão o próximo valor de CW volta a ser o definido pelo valor de CW_{Min} .

O parâmetro AIFS é um intervalo de tempo em que a estação aguarda antes de começar a transmitir. Diferente da CW, mesmo em caso de falha ou sucesso na transmissão, uma estação que deseje transmitir é obrigada a aguardar o AIFS. Esse intervalo de tempo permite que outros quadros de controle, como o ACK (*Acknowledge*), RTS (*Request to Send*) e CTS (*Clear To Send*), possam ser transmitidos e recebidos por todas as estações da rede, uma vez que os mesmos não necessitam aguardar o tempo AIFS.

O parâmetro oportunidade de transmissão TxOP, é um intervalo de tempo reservado para a transmissão de vários quadros sequencialmente de uma única estação, podendo atuar como um fator de priorização. Ele é especificado como o período máximo de ocupação do canal sem fio e é dependente da categoria de acesso. Caso duas ou mais UPs pertencente à mesma AC em uma única estação tentem transmitir ao mesmo tempo, um escalonador dentro da estação irá prevenir a colisão virtual através da concessão de TxOP para a UP com maior prioridade.

O fator de persistência é um valor fixo que irá ajudar no cálculo do novo valor de CW depois de uma tentativa de transmissão mal sucedida, afim de, reduzir a probabilidade de uma nova colisão. O fator de persistência então aumenta o valor de CW que é diferente para cada categoria de acesso (Guirardello & Branquinho, 2008).

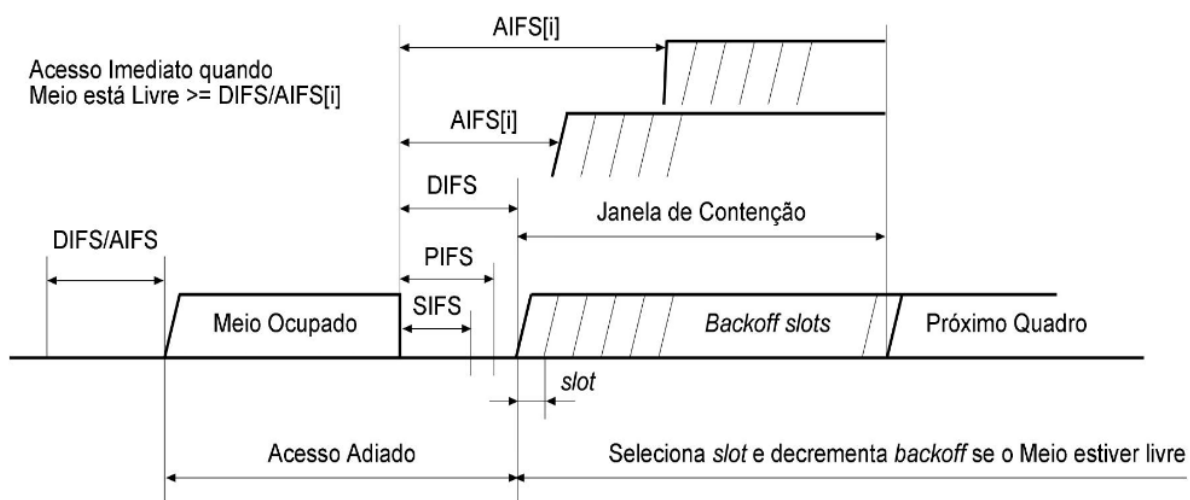


Figura 6 – Acesso ao meio através do método EDCA.

A Figura 6 acima, nos mostra a relação do AIFS com os demais intervalos. Quanto menor é a prioridade de uma categoria de acesso, maior será seu valor AIFS e consequentemente maior será o tempo de espera para transmitir dados. Assim, prioridades baixas têm de diferir por mais tempo o início da transmissão ou decrementar o *backoff*² (Sehrawat et. al., 2007).

2.2 Padrão IEEE 802.16

O padrão IEEE 802.16 (IEEE 802.16, 2004), também conhecido como WiMAX, é uma das mais atrativas soluções desenvolvidas nos últimos anos para a adoção generalizada de acesso a banda larga sem fio para áreas metropolitanas. Este padrão especifica as camadas MAC e PHY da pilha de protocolos. Teoricamente, uma estação base WiMAX pode prover acesso sem fio de banda larga na faixa de até 50 Km para estações assinantes (SS - *Subscriber Station*) e de 5 a 15 Km para estações móveis (MS - *Mobile Station*) com taxa máxima de dados de até 70 Mbps.

O WiMAX oferece dois modos de conectividade: Ponto-Multiponto (PMP - *Point-to-Multipoint*) e Mesh. A Figura 7 ilustra o modo PMP, a BS servindo órgãos governamentais, empresas, estações móveis, residências e cidades vizinhas.

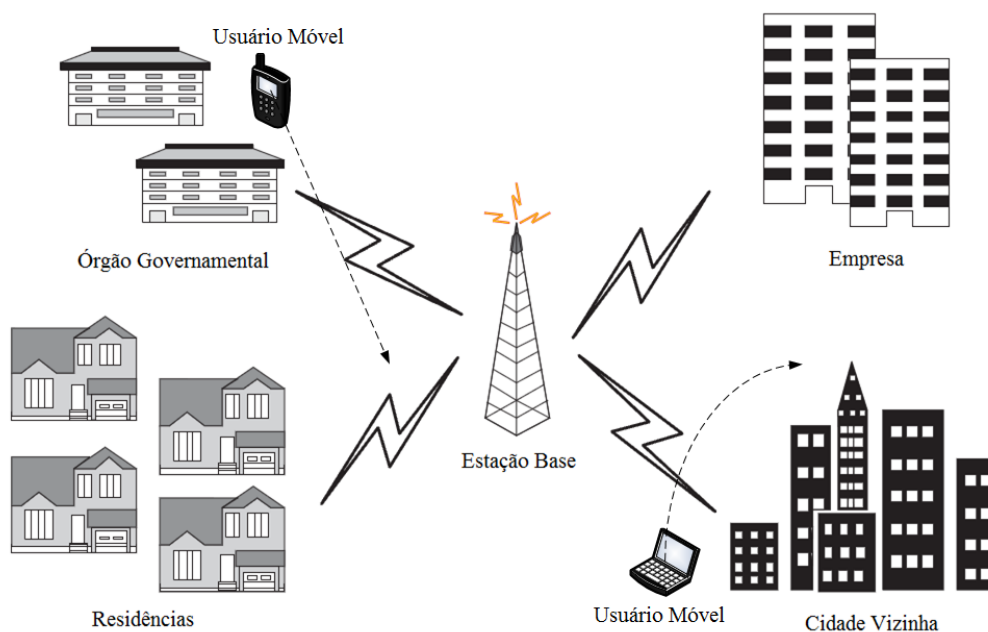


Figura 7 – Conectividade PMP para usuários fixos e móveis.

² *Backoff* é um intervalo de tempo determinado de forma aleatória, executado antes de uma estação iniciar efetivamente a transmissão de dados.

No modo PMP, dados são trocados entre BS e todas as SSs por meio de mensagens *broadcast*. A BS está centralizada e coordena todos os tráfegos de dados, estabelecendo para cada SS um único canal de uplink para seus dados e solicitações de transmissão. No modo Mesh, o tráfego é gerenciado e trocado diretamente entre SSs, bem como com uma BS. Ele contempla duas alternativas: centralizado (a BS gerencia requisitos e concessão de largura de banda para SSs) e distribuído (SSs podem se comunicar entre elas com ou sem concordância preventiva).

O IEEE 802.16 suporta também uma variedade de camadas físicas. Para ambientes de linha de visada (LOS - *Line-of-Sight*), frequências na faixa de 10 a 66 GHz são aplicáveis. O, porém é que nessas faixas de frequências há pouca propagação multipercurso, portanto faixa de 2 a 11 GHz é mais adequado. Além disso, para estas faixas de baixa frequência, dois modos de operação da camada física são aplicados. O primeiro é o OFDM (*Orthogonal Frequency Division Multiplexing*), modulação que transmite centenas de portadoras, simultaneamente, em diferentes frequências com espaçamento ortogonal, a fim de evitar interferências. O segundo é o OFDMA (*Orthogonal Frequency Division Multiple Access*), que combina o acesso múltiplo por divisão de frequência com acesso múltiplo por divisão de tempo (Lima et. al., 2006).

Estes dois tipos de modulação melhora muito a propagação de onda de rádio em condições sem linha de visada (NLOS - *No Line-Of-Sight*) e são amplamente aplicadas soluções par atenuar o efeito multipercurso.

Com relação à mobilidade, a emenda 802.16e suporta *handoff* transparente de modo a permitir que a MS troque de uma estação base para outra com velocidades veiculares (120 Km/h), a fim de garantir que aplicações de tempo real não sofram degradação de serviços (Andreadis et. al., 2011).

2.2.1 O Padrão e suas Emendas

O padrão IEEE 802.16 é conhecido por ser um padrão global, pois foi desenvolvido de modo a ter compatibilidade com os padrões do ITU (*International Telecommunication Union*) e do ETSI (*European Telecommunications Standards Institute*) (Viscaíno, 2008).

O IEEE 802.16-2001 foi o primeiro padrão da família. Foi aprovado em Dezembro de 2001 e publicado em 2002. É resultado do trabalho de centenas de participantes em todo o mundo. Este padrão fornece acesso à rede para edifícios através

da comunicação de antenas externas com uma estação base utilizando topologia PMP infraestruturado operando na frequência de rádio entre 10 e 66 GHz com uma taxa de transmissão média de 70 Mbps e uma taxa de pico de até 268 Mbps.

Em seguida foi aprovado em Abril de 2003 o padrão 802.16a. Este padrão foi projetado para operar em frequências mais baixas (2–11 GHz) e com suporte para ambientes NLOS. Sua especificação visa competir com tecnologias que oferecem acesso à última milha, como a XDSL e *cable modems*. A taxa de transmissão é de até 75 Mbps com alcance máximo de 50 km. Posteriormente foi aprovado o IEEE 802.16b, para tratar de aspectos relativos à qualidade de serviço e logo em seguida o IEEE 802.16c, desenvolvido para prover interoperabilidade, protocolos e especificação de testes de conformação.

Depois de inúmeras revisões nas emendas anteriores e das inclusões feitas por estas, surge em Junho de 2004 um novo padrão. O IEEE 802.16-REVd consolida as emendas 802.16a e 802.16c em um único padrão, substituindo o 802.16a como o padrão base. Entre as alterações pode-se destacar a provisão de suporte para antenas MIMO, o que aumenta a confiabilidade do alcance com multipercurso.

A fim de padronizar mecanismos para suporte à mobilidade, o grupo de trabalho IEEE 802.16 finalizou o desenvolvimento do novo padrão em Dezembro de 2005. O padrão 802.16e permite estações móveis trabalharem na faixa de frequência de 2–6 GHz com velocidade veicular de até 150 km/h e taxa de dados até 30 Mbps. Em frequências inferiores a 3.5 GHz pode oferecer concorrência à tecnologia celular com alcance de 2 a 5 km (Ahson, 2008).

Algumas emendas encontram-se ativas assim como outras ainda estão em fase de desenvolvimento e em estágio de pré-seleção. Entre elas podemos destacar: o IEEE 802.16f que proporciona um modelo de referência para gerenciamento de redes baseadas no padrão 802.16d, cujo modelo consiste em um sistema de gerenciamento de rede, nodos gerenciados e banco de dados de fluxo de serviço; o escopo do IEEE 802.16i fornece melhorias na mobilidade para o 802.16 MIB (*Management Information Base*) na camada MAC, camada PHY e procedimentos de gestão associados; o escopo do IEEE 802.16g produz procedimentos e alterações de serviços para os padrões 802.16d e 802.16e, provê esquemas de gestão de rede para permitir gerenciamento eficiente e interoperável dos recursos de rede, mobilidade, espectro e padronizar o comportamento do plano de gerenciamento em dispositivos 802.16 fixos e móveis.

O IEEE 802.16k defini procedimentos e melhoramentos na camada MAC de modo a permitir o 802.16d assegurar a funcionalidade *bridge* definida no 802.1D; o IEEE 802.16h visa melhorar mecanismos MAC para permitir coexistência entre dispositivos 802.16d isentos de licença e facilitar a coexistência com outros sistemas utilizando a mesma banda; o IEEE 802.16j é responsável pelo desenvolvimento de emendas para ampliar o IEEE 802.16e para então suportar operações *multihop relay*. Deste modo, este padrão se destina a melhorar a cobertura de rede do legado 802.16, vazão e capacidade do sistema (Ilyas, 2008).

O Quadro 3 mostra algumas informações das principais emendas do padrão IEEE 802.16.

Quadro 3 – Principais Emendas da família IEEE 802.16.

	Emendas IEEE 802.16			
	802.16	802.16a	802.16d	802.16e
Homologação	Dezembro de 2001	Abril de 2003	Junho de 2004	Dezembro de 2005
Faixa de Frequência	10 – 66 GHz	2 – 11 GHz	2 – 11 GHz 10-66 GHz	2 – 6 GHz
Condições do Canal para Aplicações	LOS (<i>line-of-sight</i>)	NLOS (<i>non-line-of-sight</i>)	NLOS (<i>non-line-of-sight</i>)	NLOS (<i>non-line-of-sight</i>)
Taxa de Transferência	32 – 134 Mbps	Máximo de 75 Mbps	Máximo de 75 Mbps	Máximo de 15 Mbps
Largura de Banda dos Canais	20, 25 e 28 MHz	1,25 - 28 MHz	1,25 - 28 MHz	1,25 - 20 MHz
Mobilidade	Fixa	Fixa e Nômade	Fixa e Nômade	Mobilidade, <i>Handover</i>
Raio da Célula	2 – 5 km	5 – 10 km (podendo alcançar 50 km dependendo do tamanho da antena e a mercê de parâmetros como o ganho e a potência)		2 – 5 km

2.2.2 Arquitetura e Protocolos WiMAX

Para satisfazer as necessidades de diversos setores da indústria, o padrão IEEE 802.16 tornou-se muito abrangente e complexo, incorporando uma grande variedade de opções. Então, para promover a interoperabilidade de soluções baseadas no padrão 802.16, foi formado em Junho de 2001 o WiMAX Fórum (WiMAX Fórum, 2011). O WiMAX Fórum fornece certificação da conformidade, compatibilidade e

interoperabilidade de produtos baseados no padrão IEEE 802.16. Os principais membros do WiMAX Fórum são os fabricantes de semicondutores, fabricantes de equipamentos, integradores de sistemas e provedores de serviço.

Uma das principais extensões ao padrão IEEE 802.16 é a sugestão de um modelo padronizado de arquitetura de rede, como mostrado na Figura 8. Este modelo é chamado de modelo de referência de rede WiMAX (*WiMAX network reference model*).

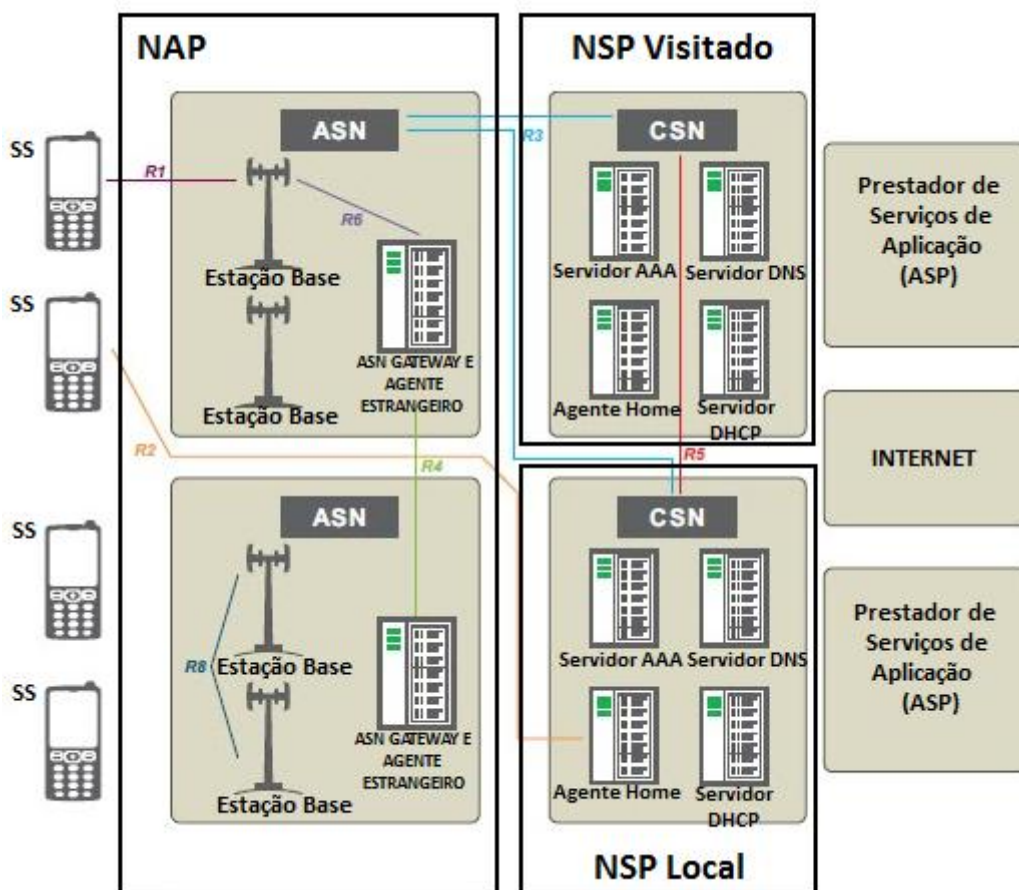


Figura 8 – Arquitetura de rede proposta pelo WiMAX Fórum.

O NAP (*Network Access Provider* – Provedor de Acesso à Rede) é a entidade que fornece o acesso de rádio aos NSPs (*Network Service Providers* – Provedores de Serviço à Rede) e controla os ASNs (*Access Service Network* – Rede para Serviço de Acesso). A conectividade IP e os serviços WiMAX são fornecidos pelos NSPs, que também gerenciam os CSNs (*Connectivity Service Network* – Redes de Serviços de Conectividade). A arquitetura WiMAX suporta dois tipos de mobilidade: *ASN anchored mobility* (Mobilidade Apoiada por ASN), ou micromobilidade ou mobilidade intra-

ASN, e *CSN anchored mobility* (Mobilidade Apoiada por CSN), ou macromobilidade ou mobilidade intra-ASN (Andrews et. al., 2007).

A comunicação entre diferentes entidades é realizada através dos seguintes pontos de referência:

- R1: conexões IEEE 802.16-2009 entre SS/MS e BS;
- R2: autenticação, autorização de serviços, configuração de host IP e gerenciamento de mobilidade;
- R3: protocolos entre ASN e CSN para o suporte de AAA (*Authentication, Authorization and Accounting* – Autenticação, Autorização e Auditoria), aplicações de política e mobilidade;
- R4: protocolos entre ASN-GWs (*ASN Gateways*) para coordenar a mobilidade das MSs e interoperabilidade entre ASNs;
- R5: protocolos para interconectividade entre os CSNs;
- R6: protocolos para comunicação entre BS e ASN-GW;
- R7: ponto de referência opcional entre PDP (*Policy Decision Point* - Ponto de Decisão de Política) e PEP (*Policy Enforcement Point* - Ponto de Aplicação de Política);
- R8: mensagens entre BSs para garantir *handover* rápido e transparente.

Conforme ilustrada na Figura 9, o padrão IEEE 802.16 abrange as duas camadas mais inferiores da pilha de protocolos: a camada MAC e camada PHY.

A camada MAC é responsável por determinar qual SS pode acessar a rede e está subdividida em três subcamadas:

- Subcamada de Convergência de Serviços Específicos (MAC CS - *Service-Specific Convergence Sublayer*);
- Subcamada de Parte Comum (MAC CPS – *Common Part Sublayer*);
- Subcamada de Segurança (MAC SS - *Security Sublayer*).

A MAC CS transforma os dados recebidos do ponto de acesso de serviço (SAP - *Service Access Point*) em pacotes de dados MAC. A transformação mapeia informação da rede externa em informação MAC, tais como fluxo de serviço e identificador de conexão (CID). O padrão atual detalha duas especificações de CS: ATM CS e *Packet*

CS. A CS é também responsável pela manutenção/provisionamento de QoS e permissão de alocação de banda.

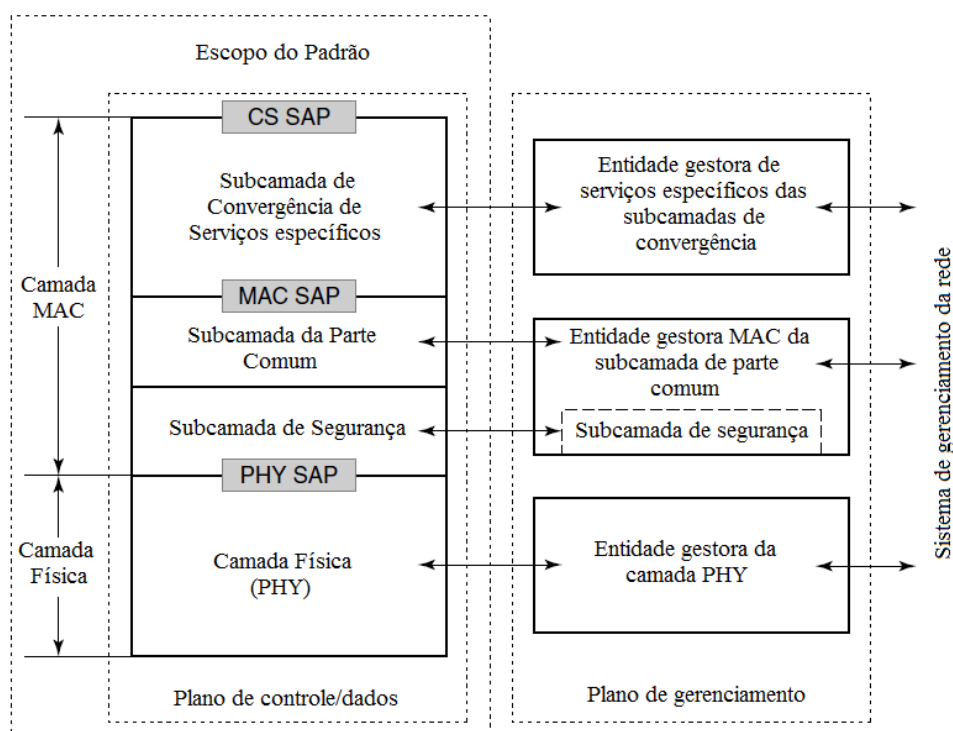


Figura 9 – Arquitetura da pilha de protocolos do WiMAX.

A MAC CPS é responsável pela funcionalidade de controle de acesso, alocação de banda, estabelecimento de conexão e manutenção. Dados, controle PHY e outras informações de gerenciamento são trocadas entre a MAC CPS e PHY através da PHY SAP.

A MAC SS é responsável pela autenticação, troca de chaves seguras e criptografia e integridade do controle de acesso. Ela usa o protocolo de gerenciamento de chaves privadas (PKM - *Privacy Key Management*) para troca segura de chaves entre a SS/MS e a BS.

A camada PHY é responsável pela transmissão e recepção de dados. Ela especifica a faixa de espectro 10-66 GHz considerando um ambiente de linha de visada (LOS) entre BS e SS. A camada PHY suporta ampla largura de banda de 20, 25, ou 28 MHz (Ilyas, 2008).

2.2.3 QoS na Rede WiMAX

O WiMAX, ou mais especificadamente o padrão IEEE 802.16e (IEEE 802.16e, 2006), implementa suporte à QoS na camada MAC para facilitar a interação com o gerenciamento de recursos de rádio e camada física. Seu framework de QoS adota cinco classes de serviços ou CoS (*Class of Service*): UGS (*Unsolicited Grant Service*), rtPS (*real-time Polling Service*), ertPS (*extended real-time Polling Service*), nrtPS (*non real-time Polling Service*) e BE (*Best Effort*).

A CoS UGS foi projetada para suportar tráfegos em tempo real, muitos sensíveis ao atraso; ele é caracterizado por pacotes de dados de tamanho fixo, que requerem alocação fixa de largura de banda. A BS prover concessões de maneira não solicitada.

A CoS rtPS é similar ao UGS, pois trata de tráfegos em tempo real, com o porém de que os dados são mais tolerantes ao atraso e tamanho de pacotes variáveis. Para esta classe de fluxo de serviço, a BS executa periodicamente uma alocação individual para requisições de banda de SSs.

A CoS ertPS é uma classe de serviço recentemente adotada no padrão. Suporta tráfego de tempo real e é representada como uma classe intermediária entre UGS e rtPS. A BS prover concessões de maneira não solicitada (como no UGS) com alocação dinâmica de banda. O ertPS é indicado para aplicações em tempo real de taxa variável que têm requisitos na taxa de dados e atraso.

A CoS nrtPS está associada com tráfegos de dados que não são de tempo real; ela requer um mínimo de reserva de taxa de dados. Novamente, a BS realiza alocação individual para requisitos de banda de SSs.

Por último, temos a CoS BE que foi projetada para aplicações sem garantias em termos de atraso e taxa de dados. As SSs enviam suas requisições de banda através de mecanismo de contenção do canal (Nuaymi, 2007).

Cada CoS possui um conjunto de parâmetros de QoS que devem ser inclusos na definição do fluxo de serviço quando a classe de serviço está habilitada para um fluxo de serviço. Os principais parâmetros são: prioridade de tráfego, latência máxima, jitter, máxima e mínima taxa de dados e máximo atraso.

O Quadro 4 mostra um resumo das cinco classes de serviços WiMAX, aplicações típicas e parâmetros de QoS correspondentes.

Quadro 4 – Classes de serviços WiMAX e parâmetros de QoS associados.

Serviço de Escalonamento	Dados Correspondentes a Prestação de Serviços	Aplicações Típicas	Parâmetros de QoS Obrigatórios
Unsolicited Grant Service (UGS)	Unsolicited grant service (UGS)	VoIP sem supressão de silêncio	Vazão máxima sustentada Latência máxima tolerada Jitter tolerado
Extended Real-Time Polling Service (ertPS)	Extended realtime variable-rate service (ERT-VR)	VoIP com supressão de silêncio	Vazão mínima reservada Vazão máxima sustentada Latência máxima tolerada Prioridade de tráfego Jitter tolerado
Real-Time Polling Service (rtPS)	Real-time variable-rate service (RT-VR)	Streaming de áudio ou vídeo	Vazão máxima reservada Vazão máxima sustentada Latência máxima tolerada Prioridade de tráfego
Non-Real-Time Polling Service (nrtPS)	Non-real-time variable rate service (NRT-VR)	FTP (File Transfers Protocol)	Vazão mínima reservada Vazão máxima sustentada Prioridade de tráfego
Best-Effort Service (BE)	Best-effort service (BE)	Web browsing, e-mail	Vazão máxima sustentada Prioridade de tráfego

A camada MAC do padrão 802.16 é orientada a conexão. Mensagens de sinalização entre BS e SS precisam ser trocadas a fim de se estabelecer um fluxo de serviço³ entre eles. Cada fluxo de serviço é caracterizado por uma série de parâmetros de QoS que indicam latência e jitter necessários e garantia de vazão. Além do mais, cada fluxo de serviço recebe da BS um único identificador de fluxo de serviço (*SFID - Service Flow Identifier*), um valor inteiro longo de 32 bits, a fim de identificar cada fluxo de serviço individualmente. Para todo fluxo de serviço ativo, há um identificador de conexão (*CID - Connection Identifier*), um valor inteiro de 16 bits. Desse modo, para cada conexão⁴ entre BS e SS associada a um fluxo de serviço, será atribuído um CID (IEEE 802.16e, 2005).

A Figura 10 mostra a arquitetura de QoS WiMAX conforme definido no padrão. Podemos observar que escalonadores, classificadores e parâmetros de QoS estão presentes na camada MAC tanto da estação base (BS) quanto da estação assinante (SS).

³ Um fluxo de serviço é um serviço de transporte MAC que provê transporte unidirecional de pacotes uplink transmitidos pela SS ou para transporte de pacotes downlink transmitidos pela BS.

⁴ Uma conexão é um mapeamento unidirecional entre pares MAC de uma BS e uma SS com a finalidade de transportar o tráfego de um fluxo de serviço. Uma conexão é somente um tipo de serviço (por exemplo, voz e e-mail não podem ter a mesma conexão MAC).

A BS é responsável pelo gerenciamento e manutenção de QoS de todos os pacotes transmitidos. Para isto, ela gerencia a distribuição dinâmica de tempo de uso da largura de banda para estações assinantes através informações incorporadas em quadros de gerenciamento conforme ilustrado na Figura 11.

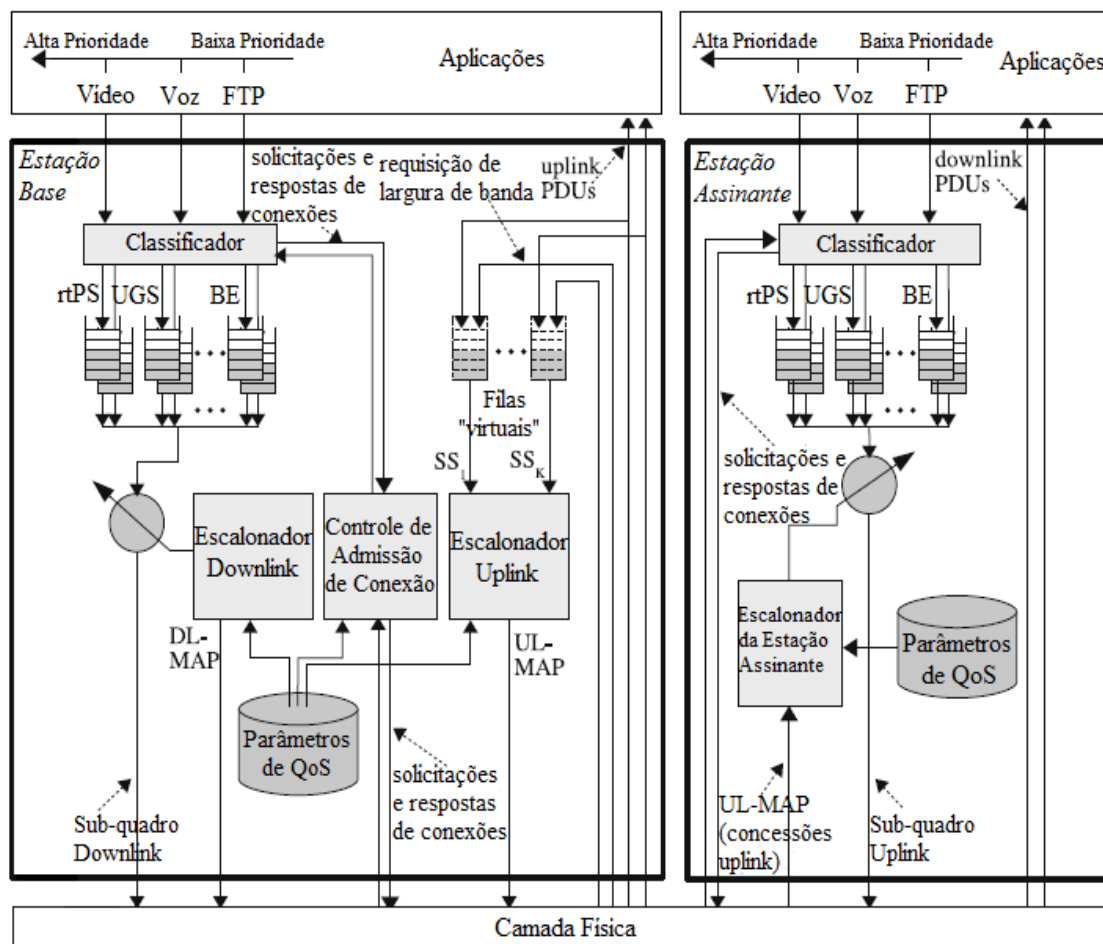


Figura 10 – Arquitetura completa de QoS em redes WiMAX.

A comunicação entre BS e SS, pode ser iniciada pela BS (condição obrigatória) ou iniciada pela SS (condição opcional). Para ambos os casos, é necessário com que haja solicitações de conexão para o controlador de admissão (CAC - *Connection Admission Control*) localizado na BS. O CAC é responsável pela admissão ou rejeição de uma requisição de conexão. Suas decisões são baseadas em parâmetros de QoS contidos nas mensagens de requisição DSA-REQ (*Dynamic Service Addition Request*). Se os parâmetros de QoS estão dentro dos limites dos recursos disponíveis, e se este for o caso, a BS responde com uma mensagem de aceitação DSA-RSP (*Dynamic Service Addition Reponse*) e atribuição de um SFID para o novo fluxo de serviço criado.

O fluxo de serviço então criado é classificado. Conforme definido no padrão, a classificação é o processo pela qual os pacotes de dados são mapeados para uma conexão específica para a transmissão entre pares MAC. O processo de mapeamento associa um pacote de dado com uma conexão, que também cria uma associação com as características do fluxo de serviço desta conexão (Chrost & Brachman, 2010).

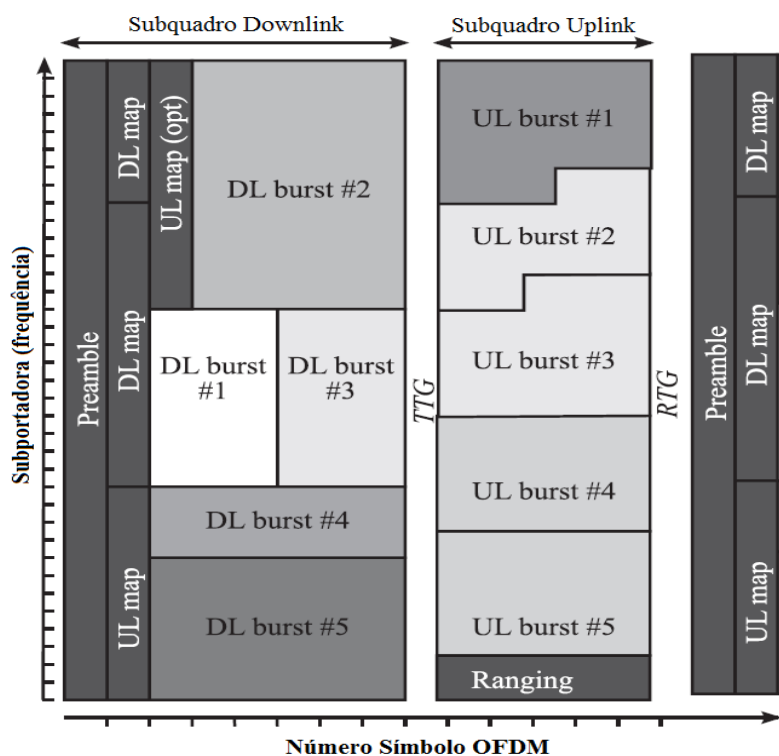


Figura 11 – Estrutura de um quadro de gerenciamento WiMAX.

Após o processo de classificação, o aspecto mais complexo na provisão de QoS para pacotes individuais é realizada por três escalonadores: Escalonadores Downlink e Uplink localizados na BS, responsáveis pelo gerenciamento dos fluxos no sentido downlink ou uplink respectivamente (BS para SS e vice-versa), e escalonador da estação assinante que em conjunto com os demais escalonadores gerenciam fluxos no sentido uplink ou da SS para BS.

Um escalonador de maneira geral tem o objetivo de determinar o perfil de *streaming* de dados (*Burst Profile*) e os períodos de transmissão para cada conexão, levando em consideração os parâmetros de QoS associados ao fluxo serviço, requisitos de largura de banda de estações assinantes e parâmetros de modulação e codificação.

O escalonador downlink tem papel relativamente simples em comparação ao escalonador uplink, uma vez que todas as filas de pacotes downlink residem na BS e seus estados estão localmente acessíveis. As decisões de alocação de tempo de uso da banda são transmitidas usando quadro de gerenciamento, através do subquadro downlink contendo o campo DL-MAP (*Downlink Bandwidth Allocation Map*) conforme a Figura 11. Este campo informa as SSs à tabela de tempo e propriedades da camada física para transmissão subsequente de *streaming* de pacotes.

A tarefa do escalonador uplink é muito mais complexa. Pois as filas de fluxos de pacotes uplink estão distribuídas entre SSs, e seus estados e requisitos de QoS devem ser obtidos através de solicitações de banda de cada SS. As informações obtidas de filas remotas constitui a base operacional do escalonador uplink e é descrita como “filas virtuais” conforme pode ser visto na Figura 1. O escalonador uplink decidirá alocações uplink baseado nas solicitações de banda, parâmetros de QoS e prioridades de classes de serviços. Estas decisões são transmitidas para as SSs por meio do quadro de gerenciamento, através do subquadro downlink contendo o campo UL-MAP (*Uplink Bandwidth Allocation Map*) para regulamentar os direitos de transmissão uplink de cada SS. Ou seja, o UL-MAP controla a quantidade de tempo de acesso ao canal para cada SS em subframe(s) uplink subsequente.

É importante mencionar também o subquadro uplink do quadro de gerenciamento WiMAX. Este subquadro contém basicamente três campos: *Initial ranging*, *slot* de contenção de requisitos de banda e *slots* específicos.

Initial ranging é utilizado por SSs para descobrir a potência de transmissão ideal, bem como o tempo e frequência de deslocamento para se comunicar com a BS.

O *slot* de contenção de requisitos de banda é utilizado por SSs para transmissão de mensagens MAC para solicitações de banda.

Slots especificadamente alocados para SSs individuais para transmissão de dados.

O último escalonador a ser mencionado é o escalonador da SS. Este escalonador visita as filas e seleciona os pacotes para transmissão. Os pacotes selecionados são transmitidos para a BS em *slots* de tempo alocado conforme definido no UL-MAP, que é construído pelo escalonador uplink da BS e difundida via *broadcast* pela BS para as SSs.

O padrão não especifica nenhum algoritmo de escalonamento que deve ser implementado. Alguns dos algoritmos de escalonamento bem conhecidos que podem

ser utilizados são: RR (*Round Robin*), WRR (*Weighted Round Robin*), WFQ (*Weighted Fair Queuing*), mSIR (*maximum Signal-to-Interference Ratio*) e outros projetados especificadamente para o WiMAX tais como, TRS (*Temporary Removal Scheduler*) e O-DRR (*Opportunistic Deficit Round Robin*) (Sekercioglu et. al., 2009).

2.3 Padrão IEEE 802.21

Alinhado com a necessidade de permitir conectividade transparente para usuários em movimento, servidos por diversas tecnologias sem fio, o IEEE desenvolveu e aprovou o novo padrão IEEE 802.21 ou MIHS (*Media Independent Handover Services*) (IEEE draft p802.21 d11.0, 2008). O MIHS ou simplesmente MIH, foi projetado para aperfeiçoar a integração e mobilidade entre redes sem fio de diferentes tecnologias, bem como para permitir o handover horizontal e vertical, isto é, a troca de PoA (*Point of Attachment*) entre tecnologias similares e distintas, respectivamente. Para realizar estes objetivos, o MIH conta com um conjunto de eventos de sinalização, gatilhos e serviços, unificados para qualquer tecnologia, que disponibilizam informação de camadas inferiores (MAC e PHY) para as camadas superiores (Camada de Aplicação) da pilha de protocolos.

A Figura 12 ilustra um ambiente de acesso heterogêneo que serve de suporte para tecnologias de acesso WiMAX, Wi-Fi e 3GPP LTE. Para a troca de sinalização MIH entre pares MAC, funções específicas foram atribuídas a nodos da rede de acordo com o framework IEEE 802.21. A PoA (*Point of Attachment*) é uma tecnologia de acesso sem fio que provê conectividade via enlace sem fio (por exemplo, a Estação Base WiMAX, o Ponto de Acesso Wi-Fi e eNodeB LTE 3GPP), enquanto que a PoS (*Point of Service*) é uma instância MIHF (*Media Independent Handover Function*) do lado da rede, que troca mensagens MIH com um MN multimodo. Inicialmente, antes do procedimento de handover iniciar, a entidade MIHF no MN está diretamente conectada a uma entidade MIHF no lado da rede (MIH PoS). A PoS pode está localizada na PoA da rede, com a qual o MN tem uma conexão L2 ativa (com a PoA servindo), ou mais por dentro da rede (núcleo da rede).

No lado direito da Figura 12, o MIH IS é ilustrado. Ele armazena informações genéricas sobre as Redes de Acesso a Rádio (RANs - *Radio Access Networks*) fornecida pelo operador de rede. O MIH IS também é conhecido como uma entidade non-PoS,

uma vez que não pode trocar mensagens MIH diretamente com MNs (Neves et. al., 2011).

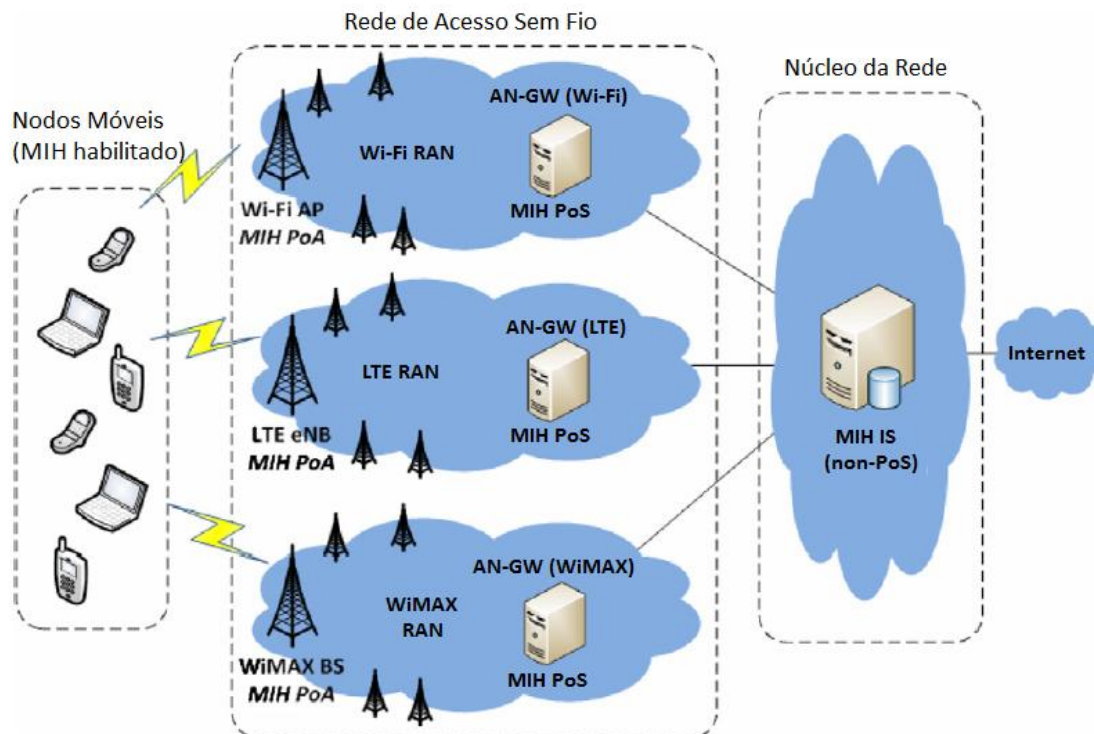


Figura 12 – Modelo de referência do framework MIH.

A Figura 13 mostra um diagrama lógico da arquitetura geral de diferentes nodos em uma rede 802.21. Ela mostra um MN com uma interface 802 e uma 3GPP, atualmente conectado à rede através da interface 802. A figura mostra a arquitetura interna do MN, rede 802, rede 3GPP e do núcleo da rede.

O padrão IEEE 802.21 consiste dos seguintes elementos: uma entidade principal denominada MIH Function (MIHF - *Media Independent Handover Function*); MIH SAP (Ponto de Acesso de Serviço - *Service Access Point*) e os usuários MIH (MIH users).

O MIHF é uma entidade lógica que prover serviços abstratos para as camadas superiores através de uma interface independente de mídia e obtêm informações das camadas inferiores através de interfaces específicas de mídia. Os serviços MIH podem ser local ou remoto, com operação local ocorrendo dentro de uma pilha de protocolos e operação remota ocorrendo entre duas entidades MIHF. Por exemplo, a comunicação remota pode ocorrer entre uma entidade MIHF em um MN e outra entidade MIHF localizado na rede. O MIHF define três tipos de serviços de mobilidade:

- O Serviço de Eventos Independente do Meio (MIES - Media Independent Event Service) proporciona a classificação de eventos, filtragem de eventos e relatórios de eventos, correspondendo a mudanças dinâmicas nas características do enlace, status do enlace e qualidade do enlace;
- O Serviço de Comandos Independente do Meio (MICS - Media Independent Command Service) permite clientes MIH gerenciar e controlar o comportamento do enlace relacionado à handovers e mobilidade. Ele também fornece um conjunto de comandos para camadas superiores em uma pilha de protocolo local ou remoto;
- O Serviço de Informações Independente do Meio (MIIS - Media Independent Information Service) fornece detalhes das características e serviços prestados por redes vizinhas e que estão atualmente servindo acesso ao meio. A informação permite o acesso ao sistema de maneira eficaz e decisões de handover eficiente.

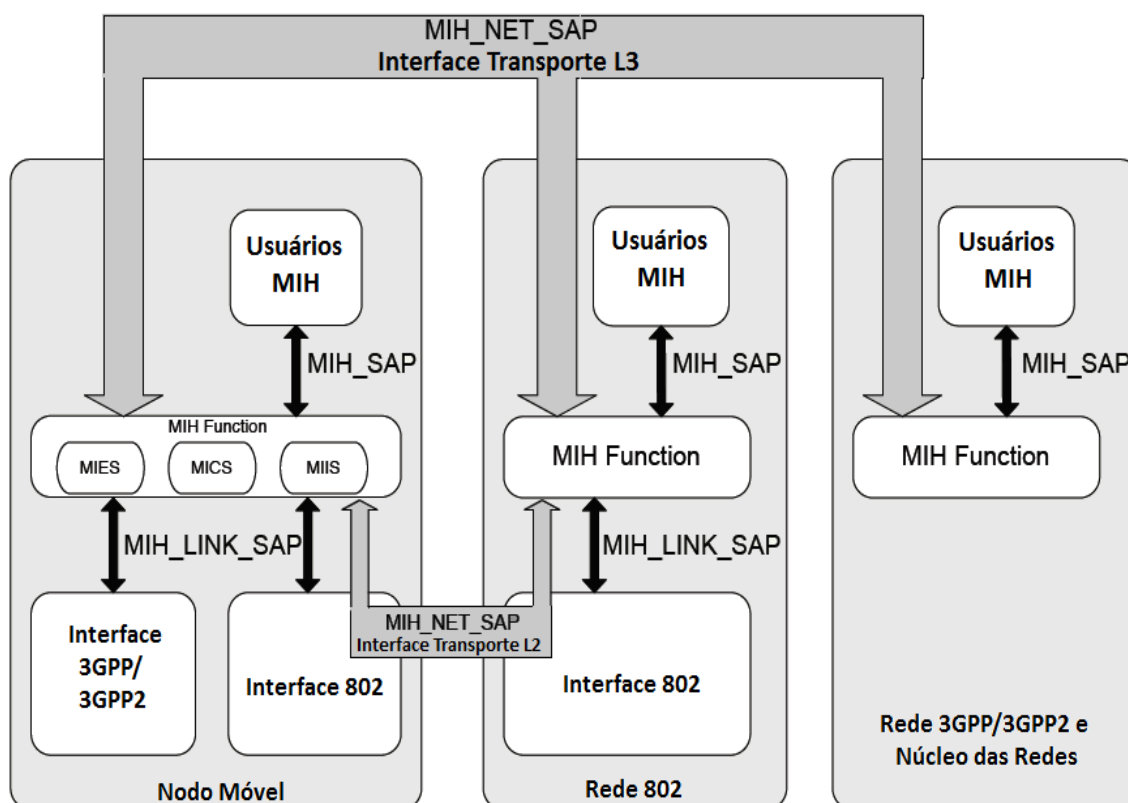


Figura 13 – Arquitetura geral do IEEE 802.21.

Os MIH SAPs são definidos em termos de primitivas na especificação IEEE 802.21, o qual provê informação sobre seus parâmetros e funcionalidades. A

especificação 802.21 não exige uma linguagem de programação específica para representar à primitiva e requer executores do MIHF para definir interfaces de programação de aplicações específicas (APIs - *Application Programming Interfaces*) em termos de linguagem de programação escolhida. A Figura 13 também apresenta os três SAPs mais importantes especificados no IEEE 802.21:

- O MIH_SAP é a interface que permite a comunicação entre o MIHF e as camadas superiores ou usuários MIH;
- O MIH_LINK_SAP é a interface de comunicação entre o MIHF e as camadas inferiores (diferentes tecnologias);
- O MIH_NET_SAP é a interface responsável pela troca de mensagens e informações MIH entre as entidades MIHF remotas ou com MIIS;

Os usuários MIH são abstrações de entidades funcionais que utilizam serviços MIH, ou seja, são consumidores de serviços MIH. Um usuário típico de serviços MIH poderia ser uma aplicação de gerenciamento de mobilidade que usaria estes serviços para aperfeiçoar handovers. Por exemplo, usuários MIH podem se associar com o MIES para ser notificado quando eventos específicos importantes para a decisão e o processo de handover ocorrer (Oliva et. al., 2011) (Taniuchi et. al., 2009).

No que se refere ao handover no escopo IEEE 802.21, o padrão trata com mais clareza da inicialização e preparação de handover, enquanto que a execução de handover é tratada por outros protocolos, tais como protocolos de gerenciamento de mobilidade da camada superior.

Como pode ser visto na Figura 14, o IEEE 802.21 oferece os mecanismos para disparar um handover, bem como preparar o handover para uma nova conexão sem fio. A inicialização de handover envolve os procedimentos de configuração do antigo enlace, relatórios de medição de rádio e nova descoberta de enlace. No IEEE 802.21, isto significa que antigos dispositivos devem ser configurados para registrar medições quando limiares específicos são ultrapassados. Este tipo de registro de medição pode indicar uma solicitação de handover urgente ou apenas uma mensagem informativa periódica. Além disso, a descoberta de uma nova conexão pode ser realizada com gatilhos específicos de camadas de enlace disponível.

Em relação à preparação para o handover, o escaneamento adicional de RATs (*Radio Access Technology*) nas proximidades do MN pode ser realizado com a ajuda de diferentes serviços IEEE 802.21 (ou seja, MIES, MICS e MIIS). Mais especificadamente, a descoberta de um novo enlace pode envolver uma consulta para um servidor MIIS remoto que mantém as informações sobre redes disponíveis na área de um MN específico ou um comando de rede para o MN iniciar o escaneamento. No caso de acesso a novos recursos de rede, o MN deve autenticar-se a rede antes de prosseguir. Se o MN está autenticado na nova rede, o parâmetro de QoS deve ser transferido para uma verificação de disponibilidade de recursos. O resultado de uma verificação de disponibilidade de recursos de rádio e outras informações de rede são entradas para o algoritmo de decisão de handover. Além disso, recursos de rádio devem ser reservados sobre o RAT selecionado. A verificação de segurança e a implementação do algoritmo de decisão estão fora do escopo do IEEE 802.21, enquanto que a reserva de recursos não tem sido especificada na versão atual do projeto IEEE 802.21, mas é esperado para ser concluído no futuro.

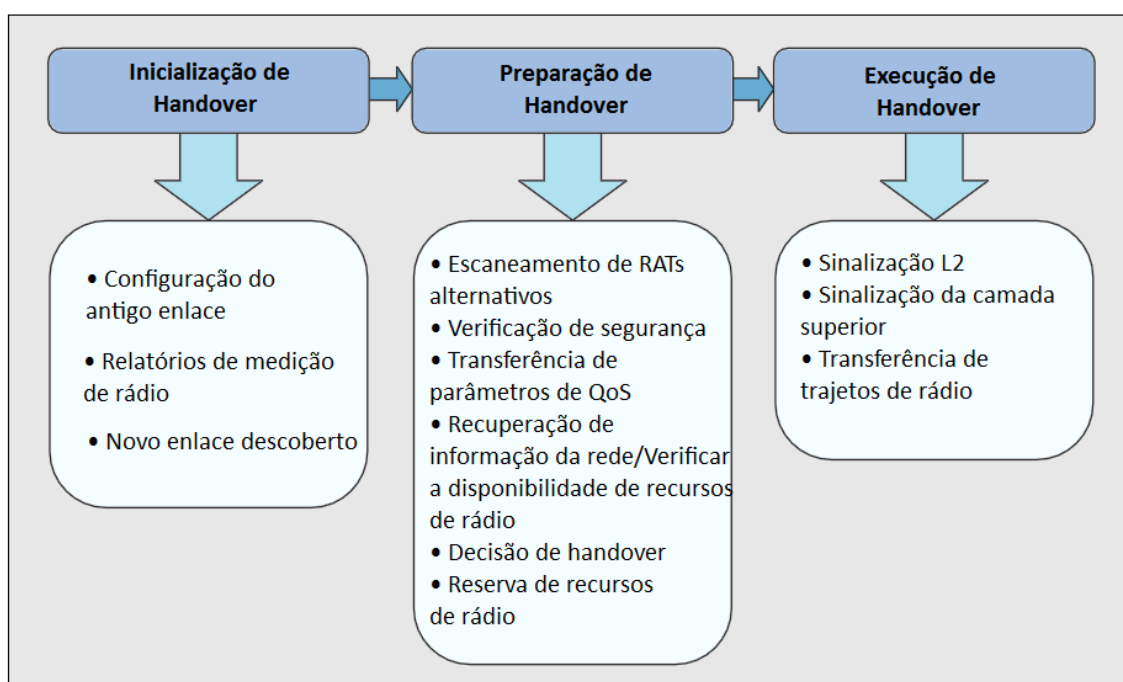


Figura 14 – Escopo do IEEE 802.21 durante o handover.

Por fim, a execução de handover está fora do escopo do padrão IEEE 802.21 como a mobilidade da camada L2 é tratada por procedimentos específicos da rede, e o

re-roteamento do tráfego IP é geralmente realizada com outros protocolos, como o IPv6 móvel. No entanto, o IEEE 802.21 pode desencadear a ativação ou desativação de enlaces de uma maneira que pode preservar os recursos sobre o antigo enlace para menor impacto da interrupção de handover (Lampropoulos et. al., 2008).

2.4 Resumo do Capítulo

Esse capítulo abordou assuntos relacionados a conceitos, padrões, arquitetura, protocolos e respectivos frameworks de QoS das tecnologias de acesso sem fio Wi-Fi e WiMAX. Descreveu também a arquitetura geral do padrão IEEE 802.21 ou MIH, enfatizando com mais detalhes etapas do processo de handover através desta tecnologia.

O Capítulo 3 mostrará alguns trabalhos relacionados ao tema desta dissertação com assuntos pertinentes ao gerenciamento de QoS e QoE, mapeamento de QoS e balanceamento de carga, resumindo-os, criticando seus pontos fracos e enaltecendo seus pontos importantes.

3. Trabalhos Relacionados

Esse capítulo descreve alguns dos trabalhos relacionados encontrados durante a pesquisa desta dissertação. Os trabalhos são agrupados em: Mapeamento de QoS em Redes Sem Fio Heterogêneas e Métodos para Suporte a Balanceamento de Carga. Os trabalhos estão resumidos no Quadro 5 ao término deste capítulo.

Os tópicos fazem menção e resumem um pouco de cada trabalho, mostrando de maneira sucinta e direta as conclusões básicas assim como algumas críticas.

3.1 Mapeamento de QoS em Redes Sem Fio Heterogêneas

Em (Andi et. al., 2010), os autores propõem um *framework* de gerenciamento de interface com suporte a QoS em terminais *multi-interface* sem fio. O *framework* se baseia no MIH e utiliza métricas da camada de enlace para avaliar as condições da rede e assim auxiliar na decisão de handover.

Políticas também foram implementadas para dar suporte às preferências do usuário. Para realizar estes objetivos, o *framework* consiste de três principais blocos funcionais: VDA (*Virtual Device Agent*), serviço MIH, e *Profile Storage*. O VDA visa manter a tabela de conexão, ou seja, o mapeamento entre os fluxos de serviço e a interface de rede, execução das políticas e redirecionamento dos fluxos durante o handover; o MIH fará a unificação de interfaces entre tecnologias distintas e por fim o *Profile Storage* que armazena continuamente parâmetros dinâmicos da camada L2.

O, porém deste artigo é que a prioridade entre os fluxos é feita por reserva de canal, ou seja, é configurada uma taxa de transferência para uma aplicação independente de tecnologia. Diferente da nossa proposta, que utiliza classes de serviços ou categorias de acesso conforme os padrões homologados, de maneira a aperfeiçoar o consumo da banda. O artigo também não faz uso de nenhum esquema de balanceamento de carga associada a métricas indicadoras de congestionamento de ambas as redes.

A proposta apresentada em (Chen et. al., 2008), desenvolveu uma arquitetura denominada VHTC (*Vertical Handoff Translation Center Architecture*), que é constituída por um módulo de mapeamento de QoS, módulo de transladação de pacotes e um módulo de gerenciamento eficiente de uso da largura de banda, para garantir QoS durante a fase de *handover* em redes heterogêneas. No entanto, os autores não simularam ou avaliaram qualquer cenário com mobilidade, tampouco o suporte à QoS

antes, durante e após o procedimento de *handover*. O termo *handover* mencionado no artigo é referente ao tráfego de dados e não a mobilidade de nodos. Os nodos estão estáticos e comunicam entre si via enlace cabeado linear, cujas extremidades da rede cabeada é formada pela Estação Base WiMAX e Ponto de Acesso Wi-Fi.

Este estudo usou uma versão modificada do ns-2. O módulo EDCA do grupo TKN (*Telecommunication Networks Group*) e o módulo WiMAX do NIST (*National Institute of Standards and Technology*), foram modificados para que houvesse integração. O, porém é que o módulo WiMAX não implementa classes de serviços para a tecnologia WiMAX, ou seja, neste artigo não há, verdadeiramente, uma implementação de QoS para o WiMAX.

Em (Tarnng et. al., 2010), os autores propõem um ambiente heterogêneo integrado das redes IEEE 802.11 e IEEE 802.16, bem como o desenvolvimento de um mecanismo de mapeamento de QoS afim de atender os requisitos de aplicações de tempo real por meio da alocação de largura de banda para a estação assinante. Os autores desenvolveram também dois algoritmos de QoS, um para a BS e outro para a SS. O algoritmo da BS realiza a alocação de largura de banda para os serviços de todas as SSs, enquanto o algoritmo da SS realiza alocação de largura de banda para serviços de tempo real na SS.

Para simular e avaliar o ambiente proposto, os autores utilizaram o módulo EDCA do TKN, o módulo WiMAX do CGU (*Chang Gung University*) e o MIH para a troca de mensagens entre ambas as tecnologias. Vale ressaltar que o módulo WiMAX do CGU utilizado neste artigo, não implementa escalonamento de classes de serviços e no cenário avaliado, no cenário de avaliação os nodos não realizam mobilidade e não adotam nenhum mecanismo ou algoritmo que possa garantir balanceamento de carga e QoS em ambas as redes em situações em que uma destas esteja saturada.

Os autores em (Prasath et. al., 2008), propõem um modelo de integração para redes Wi-Fi e WiMAX em equipamentos dentro das instalações do cliente (CPE - *Customer Premises Equipments*). Além disso, desenvolveram um algoritmo de escalonamento adaptativo de modo a garantir QoS tanto para tráfegos de tempo real, quanto para tráfegos de tempo não real sobre a interface WiMAX. A integração Wi-Fi/WiMAX é realizada por meio da Entidade Gestora de Integração (IME - *Integration Management Entity*) localizada no CPE, responsável por prover a nível da camada MAC, o mapeamento de diferentes tráfegos e o gerenciamento de sinalização de conexão.

Apesar de a proposta definir um método eficiente de integração, ela não leva em conta a mobilidade e gerenciamento de QoS. Para a estratégia de escalonamento proposto, a chegada de Unidades de Dados de Protocolo (PDU - *Protocol Data Units*) no CPE, provenientes de nodos Wi-Fi, são mapeados para diferentes filas de classes de serviços (UGS, rtPS, nrtPS e BE). O algoritmo de escalonamento analisa estas filas e prover o menor atraso para conexões rtPS e assegura que não haverá perda de dados para o tráfego nrtPS. Em resumo, o algoritmo garante QoS para a interface WiMAX, mas não para a interface Wi-Fi. Além do mais, não há suporte para mobilidade e gerenciamento de QoS de forma integrada.

No trabalho de (Cerqueira et. al., 2008), os autores desenvolveram um controlador de sessões multi-usuários em redes heterogêneas sem fio e cabeada denominado *QUALITIS*, através da qual coordena o mapeamento de QoS e mecanismos de adaptação de QoS, juntamente com mecanismo de alocação de recurso e mobilidade. A avaliação de desempenho do *QUALITIS* foi validada com base em experiências de simulação via ns-2 para verificar o seu impacto com base na expectativa do usuário (QoE), e protótipo experimental para medir o desempenho da rede (QoS).

Mesmo que a proposta apresente diversos mecanismos e funcionalidades capazes de trabalharem em conjunto para prover o melhor nível de qualidade de uma sessão em andamento, esta somente avalia dispositivos com interfaces Wi-Fi e consequentemente o mapeamento de QoS é entre redes Wi-Fi, além de não assegurar handover transparente. O *QUALITIS* possui um mecanismo de adaptação de QoS utilizado em situação em que a rede está congestionada, contudo, o autores não se preocupam com a possibilidade equilibrar o consumo da banda com a redes vizinhas pela qual o dispositivo móvel pretende realizar o handover.

Os autores em (Lee et. al., 2010), propõem um modelo *cross-layer* da arquitetura *DiffServ* em redes IEEE 802.16 *mesh/relay* de múltiplos-saltos com WLANs IEEE 802.11e, com o objetivo de prover garantia de QoS fim-a-fim, principalmente para comunicação multimídia. A arquitetura consiste de clientes com interface 802.11e, roteadores infraestruturados em um *backbone* sem fio *mesh/relay* e redes *backhaul* IP.

A arquitetura visa manter QoS entre roteadores de borda e clientes, entre roteadores infraestruturados e entre roteadores com funcionalidade *gateway* e redes *backhaul* IP. O mapeamento de QoS é realizado através de uma tecnologia de transição *bridging* IP com *DiffServ* implementada no roteador de borda. Os pacotes provenientes de clientes são roteados pela interface 802.11e do roteador de borda, que através da

camada IP da interface, transmite os pacotes de dados para a interface 802.16 para serem categorizados em classes de serviços conforme as categorias PHB (*Per-Hop Behavior*) associadas com os DSCPs (*Differentiated Services Code Point*) dos pacotes.

Os resultados das simulações validam a garantia da QoS desejada, principalmente para tráfegos multimídia e o mecanismo de mapeamento de QoS é eficiente e flexível. O trabalho comparado com a proposta desta dissertação, não se preocupa com uma avaliação de cenários efetivamente heterogêneos e com mobilidade. Além do mais, o mapeamento é realizado em um único sentido, com tráfegos originados de interface Wi-Fi para interface WiMAX.

3.2 Métodos para o Suporte a Balanceamento de Carga

O trabalho (Andi et. al., 2010) apresenta uma proposta interessante para prover balanceamento de carga através de um framework de gerenciamento de múltiplas interfaces do terminal móvel, considerando que seja mais fácil ter controle para selecionar a rede de acesso desejada de modo a aumentar a qualidade de uma aplicação em andamento. Para inferir na seleção de rede, o terminal móvel faz uso de informações da camada MAC (atraso na camada 2, tempo de ocupação do canal e taxa de perda de requisições de largura de banda) provenientes das interfaces Wi-Fi e WiMAX para então enviar via MIH, para o VDA (*Virtual Device Agent*) localizado nas camadas superiores.

Foi avaliado um cenário, em que as redes Wi-Fi e WiMAX foram gradualmente congestionadas para que então o terminal escolha a melhor rede para sua aplicação. Pelos resultados, podemos notar que os autores não consideram diferentes tipos de aplicações (*Streaming* de vídeo, VoIP, HTTP), bem como a garantia de QoS mesmo em situações de congestionamento.

Em (Ha et. al., 2009), faz o uso do conceito de comunidade de RATs (*Radio Access Technologies*), para realizar o gerenciamento cooperativo de recursos de rádio entre RATs heterogêneos (3GPP LTE, WiBro e WLAN). Os autores também desenvolveram um algoritmo de balanceamento de carga que leva em conta o uso da largura de banda e taxa da relação sinal-ruído recebido.

Após simulações e resultados obtidos, o algoritmo de balanceamento de carga da comunidade de gerenciamento de recursos de rádio, melhora a utilização de recursos de

rádio, reduz a probabilidade de bloqueio de RATs individuais na comunidade, e aumenta ainda mais o desempenho médio de toda a comunidade.

Podemos notar que este trabalho tem como principal foco o gerenciamento de recursos de rádio entre redes sem fio heterogêneas, deixando de lado questões relacionadas ao gerenciamento de QoS em cenários com mobilidade e garantia de qualidade para fluxos de serviços individuais em situações de saturação das redes.

Alguns dos artigos apresentados, implementaram classes de serviços no WiMAX e categorias de acesso no Wi-Fi, mas as análises realizadas foram independentes. Estes trabalhos somente incluem o MIH para unificação de interfaces e coleta de parâmetros da camada de enlace para auxiliar o processo de decisório da rede ou desenvolvem algum mecanismo para realizar mapeamento de tráfego. Assim, até onde sabemos, as propostas não contemplam estudos que integrem a mobilidade em ambiente heterogêneo, o mapeamento entre classes de serviços e categorias de acesso com auxílio do MIH, bem como não propõem decisões de handover inteligentes baseadas em estratégias que visam garantir a equidade entre vazões de diferentes aplicações entre redes Wi-Fi e WiMAX.

Os trabalhos aqui apresentados foram divididos de acordo com tópicos apropriados aos temas-chaves de seu contexto e são resumidos no Quadro 5.

Quadro 5 – Relação dos trabalhos relacionados de acordo com seus assuntos-chave.

Mapeamento de QoS em Redes Sem Fio Heterogêneas	Artigo	Proposta
	(Andi, W. C., Yeo, C. K. and Lee, B. S. 2010)	<i>Framework</i> de gerenciamento de interface com suporte a QoS em terminais <i>multi-interface</i> sem fio.
	(Chen, Y. C., Hsia J. H. and Liao, Y. J. 2008)	Arquitetura denominada VHTC (<i>Vertical Handoff Translation Center Architecture</i>) para garantir QoS durante a fase de <i>handover</i> em redes heterogêneas.
	(Tarng, W., Chen, N. W., Deng, L. Z., Ou, K. L., Hsie, K. R. and Chen, M. 2010)	Ambiente heterogêneo integrado das redes IEEE 802.11 e IEEE 802.16. Mecanismo de mapeamento de QoS afim de atender os requisitos de aplicações de tempo real.
	(Prasath, G. A., Raghu, K. R. and Ma, M. 2008)	Modelo de integração para redes Wi-Fi e WiMAX em equipamentos dentro das instalações do cliente (CPE - <i>Customer Premises Equipments</i>). Algoritmo de escalonamento adaptativo para garantir QoS.

	(Cerqueira, E., Veloso, L., Curado, M., Monteiro, E. and Mendes, P. 2008)	Controlador de sessões multi-usuários em redes sem fio e cabeada (<i>QUALITIS</i>). Coordena o mapeamento de QoS e mecanismos de adaptação de QoS, juntamente com mecanismo de alocação de recurso e mobilidade.
	(Lee, M., Kwon, B. and Copeland, J. A. 2010)	Modelo <i>cross-layer</i> de arquitetura <i>DiffServ</i> em redes IEEE 802.16 <i>mesh/relay</i> de múltiplos-saltos com WLANs IEEE 802.11e, com o objetivo de prover garantia de QoS fim-a-fim, para comunicações multimídia.
Balanceamento de Carga	(Andi, W. C., Xia, Y., Yeo, C. K. and Lee, B. S. 2010)	Framework de gerenciamento de múltiplas interfaces do terminal móvel. Utiliza informações da camada MAC (atraso na camada 2, tempo de ocupação do canal e taxa de perda de requisições de largura de banda) para inferir na seleção da rede.
	(Ha, J., Kim, J. Y., Kim, J. U. and Kim, S. H. 2009)	Comunidade de RATs (<i>Radio Access Technologies</i>), para realizar o gerenciamento cooperativo de recursos de rádio entre RATs heterogêneos (3GPP LTE, WiBro e WLAN).

3.3 Resumo do Capítulo

Este capítulo discutiu alguns dos trabalhos relacionados ao tema desta dissertação e os separou em dois subgrupos de acordo com temas relativos. Houve trabalhos que descreveram diferentes mecanismos de mapeamento de QoS e outros mantiveram foco no gerenciamento de recursos de banda.

O Capítulo 4 descreverá a contribuição da dissertação junto ao arcabouço proposto dissertando suas contribuições, melhorias e sua importância para redes sem fio heterogêneas WiMAX/Wi-Fi e para o usuário final.

4. Aprovisionamento de QoS/QoE e Equidade entre Vazões

Este Capítulo apresenta a proposta de um arcabouço para o provisionamento de QoS e QoE em redes sem fio heterogêneas WiMAX e Wi-Fi. Especificamente, a solução provê mapeamento de QoS entre as classes de serviços WiMAX e categorias de acesso Wi-Fi, bem como garante a qualidade do vídeo recebido pelo usuário. Apresenta uma explicação sobre a qualidade de experiência (QoE), descrevendo suas mais conhecidas métricas com base em fórmulas matemáticas, para análise do tráfego multimídia quanto a qualidade final do vídeo recebido.

O Capítulo apresenta também uma nova metodologia baseada no MIH para a obtenção de vazão agregada tanto da rede atual, quanto da rede alvo, de modo a auxiliar o algoritmo de decisão de handover vertical, a qual levará em conta a classe de serviço ou categoria de acesso da sessão em andamento assim como os valores atuais de vazões agregadas.

4.1 Qualidade de Experiência (QoE)

Tradicionalmente, o desempenho das arquiteturas de redes tem sido avaliadas através das métricas de QoS. A QoS é definida como um conjunto de métricas que representam a capacidade da rede em fornecer garantias de serviço. Aspectos e métricas de QoS (tais como, vazão, jitter, atraso, probabilidade de bloqueio e perda de pacotes) são importantes para avaliar soluções no ponto de vista da rede, mas não em relação a perspectiva do usuário. Vários estudos da área de redes e comunicação multimídia ainda tem suas observações baseadas somente na QoS. Contudo, o foco tem mudado para a satisfação do usuário em relação a um serviço. Mesmo que uma aplicação multimídia esteja com boa vazão, não nos permite afirmar que a qualidade da mesma está satisfazendo a necessidade do usuário. Portanto, abordagens centradas no usuário são necessárias para projetar e avaliar as novas arquiteturas da Internet do Futuro.

O conceito de Qualidade de Experiência está relacionado à avaliação das aplicações multimídia do ponto de vista da percepção do usuário. A utilização de métricas de QoE servem como extensão aos parâmetros de QoS, melhorando o controle de operações nas arquiteturas de redes e protocolos. Consequentemente, proporcionando melhorias para serviços de aplicações de áudio e vídeo (Ferreira, 2009).

As avaliações de QoE tem como foco trazer melhorias para as aplicações multimídias a partir da perspectiva/experiência do usuário. Em linhas gerais, as métricas de QoE, retornam um valor quantitativo que é mapeado para uma faixa de valores qualitativos. Métricas de QoE podem ser objetiva e subjetiva, a qual, serão detalhadas a seguir.

4.1.1 Métricas Objetivas

As métricas objetivas de QoE estimam a qualidade do vídeo recebido pelo usuário através de modelos matemáticos quantitativos, cujos valores computados são mapeados em valores subjetivos de qualidade. A seguir apresentamos as três principais métricas de QoE objetivas que são adotadas nesta dissertação.

4.1.1.1 PSNR

A métrica mais tradicional de QoE é o PSNR (*Peak Signal to Noise Ratio*) (Winkler, 2005), que estima a qualidade do vídeo em decibéis, comparando o vídeo original com o vídeo recebido pelo usuário considerando o aspecto de luminosidade. Para cada faixa de valores de PSNR, há uma qualificação para o vídeo que foi recebido pelo usuário, conforme descrito na Tabela 1.

Tabela 1 – Valores de Classificação do PSNR.

PSNR (dB)	> 37	31 – 37	25 – 31	20 – 25	< 20
Qualidade	Excelente	Bom	Aceitável	Pobre	Péssimo

O valor de PSNR é expresso em dB (decibel). O PSNR é obtido através da seguinte fórmula, na escala logarítmica:

$$\text{PSNR} = 20 \log_{10} \left(\frac{\text{MAX}}{\sqrt{\text{MSE}}} \right) = 20 \log_{10} \left(\frac{255}{\sqrt{\frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \| Y_s(i, j) - Y_d(i, j) \|^2}} \right)$$

Onde:

- MAX = Valor máximo do *pixel* (255 para 8 bits);
- M x N = Quantidade de *pixels* do vídeo;
- MSE = Erro Médio Quadrático para duas imagens M x N monocromáticas Y_s e Y_d , onde uma das imagens é considerada uma medição com ruído da outra;

- $Y_s(i, j)$ = Posição dos *pixels* no vídeo original;
- $Y_d(i, j)$ = Posição dos *pixels* no vídeo recebido pelo usuário.

4.1.1.2 SSIM

A métrica SSIM (*Structural Similarity Index*) (Wang, 2004) faz a comparação quadro a quadro do vídeo original com o vídeo recebido pelo usuário, considerando os seguintes aspectos: contraste, luminosidade e estrutura. O SSIM é expresso como um valor decimal entre 0 e 1. Quanto mais próximo do valor 1, melhor é a qualidade do vídeo. No caso em que o valor é igual a 1, o vídeo recebido pelo usuário corresponde em 100% ao vídeo original. O SSIM é obtido usando a seguinte fórmula:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\text{cov}_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Onde:

- μ_x é a média de x;
- μ_y é a média de y;
- σ_x^2 é a variância de x;
- σ_y^2 é a variância de y;
- COV_{xy} é a covariância de x e y;
- $c_1 = (k_1L)^2$, $c_2 = (k_2L)^2$ são constantes;
- L é a extensão de valores do pixel, ou seja, a diferença entre o valor máximo e o valor mínimo (geralmente: $2^{\text{bits por pixel}} - 1$);
- $k_1 = 0,01$ e $k_2 = 0,03$ por padrão.

Os valores extraídos do frame recebido pelo usuário e do frame original são armazenados em vetores separadamente, um vetor para luminosidade, outro vetor para estrutura e um vetor para cor. Posteriormente, obtém-se a média de cada vetor e a combinação dessas três médias gera o valor do SSIM, indicando a qualidade do vídeo.

4.1.1.3 VQM

A métrica VQM (*Video Quality Metric*) (Xiao, 2000) também compara o vídeo original em relação ao vídeo recebido pelo usuário, considerando os seguintes aspectos: embasamento, ruído, distorção dos frames e cor. Quanto mais próximo o valor for de 0, melhor será a qualidade do vídeo. Esta métrica é baseada na DCT (*Discrete Cosine Transform*). Para a obtenção da métrica, os seguintes passos são realizados:

- 1) Transformação do vídeo para o formato YUV (Y é a luminância, U (R-Y) é o sinal R (*Red*) vermelho e Y somados, V(B-Y) é o sinal do B (*Blue*) azul e Y).
- 2) DCTs de blocos 8x8 são usados para separar as imagens em diferentes frequências.
- 3) Conversão dos coeficientes da DCT para LC (*Local Contrast*) usando a fórmula:

$$LC_i = DCT_{i,j} \frac{(DC/1024)^2}{DC}$$

Onde DC é o coeficiente DCT com índices (posição do pixel) $(i, j) = (0, 0)$.

- 4) Conversão do LC para “a diferença apenas perceptível” (*just-noticeable difference*):

$$JND_{i,j} = LC_{i,j} \cdot CSF_{i,j}$$

Sendo o CSF a Função de Sensitividade do Contraste (*Contrast Sensitivity Function*) (Salomon et. al., 2004).

- 5) Cálculo da seleção ponderada da média e máxima distorções. A diferença absoluta (D) entre o vídeo original e reduzido é calculada para cada um dos coeficientes JND e então temos:

$$VQM = \text{média}(|D|) + 0.005 \cdot \text{máximo}(|D|)$$

4.1.2 Métricas Subjetivas

As métricas subjetivas de QoE visam avaliar a qualidade das aplicações multimídias baseado na opinião do usuário. O usuário qualifica o vídeo de acordo com a sensibilidade visual que tem em relação a vídeo assistido. Essa qualificação é feita através de uma nota (entre 1 a 5) atribuída pelo usuário ao vídeo. As métricas subjetivas

dependem exclusivamente da opinião do usuário final, pois o grande objetivo é justamente melhorar os serviços para os clientes finais. A métrica subjetiva mais utilizada é a MOS (*Mean Option Score*) cuja escala de avaliação do vídeo varia de 1 a 5. Sendo que o valor 5 indica excelente qualidade da aplicação, enquanto o valor 1 indica péssima qualidade. A Tabela 2 mostra a escala de valores de MOS com a sua respectiva qualificação (Ferreira, 2009).

Tabela 2 – Escala de Valores MOS.

MOS	Qualidade
5	Excelente
4	Bom
3	Regular
2	Pobre
1	Péssimo

4.2 Arcabouço Proposto

O Arcabouço proposto é composto basicamente por três entidades funcionais: Arquitetura para Mapeamento de QoS, Algoritmo de Decisão de Handover Vertical (VHD - *Vertical Handover Decision*) e Mecanismo para Obtenção de Vazão Agregada da Rede Atual e Alvo. Este mecanismo é dividido em duas metodologias: o GTH (*Get Throughput*) e o RTH (*Repository Throughput*). Todas estas entidades funcionais trabalham de forma integrada e em conjunto com o MIH e o servidor MIIS. As três entidades serão abordadas com mais detalhes nas seções seguintes.

4.2.1 Arquitetura para Mapeamento de QoS

A Figura 15 apresenta o diagrama lógico da integração do MIH (padrão 802.21) com as tecnologias de acesso WiMAX (padrão 802.16e) e Wi-Fi (padrão 802.11e). A Figura mostra a arquitetura interna do nodo móvel, rede 802, rede 3GPP e da rede núcleo, em interação com a camada MAC das tecnologias WiMAX e Wi-Fi com seus respectivos mecanismos para diferenciação e priorização de tráfego no acesso ao meio.

Para que fosse possível realizar a integração desejada, foi necessário modificar módulos das tecnologias discutidas anteriormente para posterior simulação no ns-2 (Network Simulator NS-2, 2005).

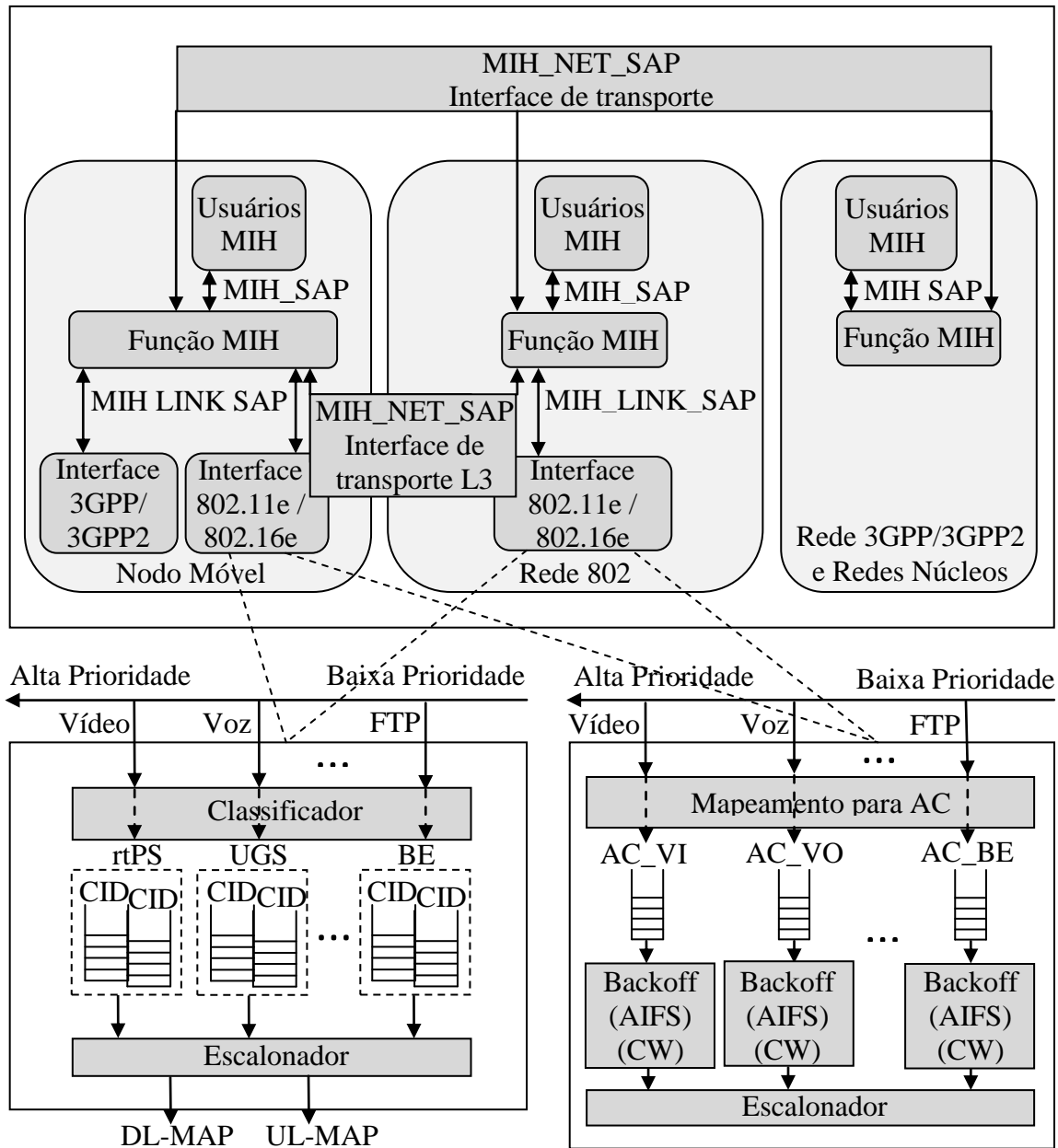


Figura 15 – Arquitetura geral para mapeamento de QoS.

O módulo Nist Mobility (Nist Mobility, 2009) foi utilizado como base para as modificações, pois o mesmo já possui a implementação MAC das tecnologias WiMAX e Wi-Fi. Como este módulo não implementa suporte a QoS em ambas as tecnologias, foi necessário estudar toda a funcionalidade do módulo EDCA do grupo TKN (TKN Group, 2006) e do módulo WiMAX do Aymen (Belghith & Nuaymi, 2009), pois estes são os dois módulos bastante conhecidos pela comunidade científica, que verdadeiramente implementam QoS conforme os padrões 802.11e e 802.16e respectivamente. Então, após estudos de seus *frameworks*, passamos para a fase de

adaptação destas funcionalidades no código do módulo Nist Mobility, e após exaustivas simulações, foi possível garantir a eficiência desta integração.

Com a implementação deste novo módulo, foi possível simular em ambiente heterogêneo, nodos móveis com duas interfaces (WiMAX e Wi-Fi), com suporte a mobilidade vertical transparente e garantia de QoS.

Desta forma, fluxos de serviços (vídeo, voz e dados) provenientes das camadas superiores, sejam de um dispositivo móvel ou de uma rede 802, são classificados para UGS, rtPS ou BE (caso estejam ou tenham feito handover para área de cobertura WiMAX e com a interface WiMAX ativa) na camada MAC. Conforme o tipo de fluxo e seus requisitos de QoS, estes são em seguida escalonados, ou no sentido uplink (UL-MAP), ou no sentido downlink (DL-MAP) para então receberem concessões da BS e assim ter prioridades no acesso ao meio.

Caso estejam ou tenham feito handover para a área de cobertura Wi-Fi e com a interface Wi-Fi ativa, os fluxos são mapeados para ACs (AC_VO, AC_VI ou AC_BE). Como cada AC possui parâmetros particulares de AIFS, CW_{Max} , CW_{Min} e TxOP, fluxos de serviços associados terão diferentes prioridades no acesso ao meio.

Nesta arquitetura o *framework* MIH é responsável pela unificação das interfaces, handover vertical transparente, auxilia o MN na descoberta e seleção da PoA alvo e coleta de informações de recursos das redes vizinhas através do servidor MIIS. O Quadro 6 resume o esquema de mapeamento de QoS entre CoSs e ACs.

Quadro 6 – Mapeamento de QoS entre as redes WiMAX e Wi-Fi.

Aplicação	Exemplo	IEEE 802.16e	IEEE 802.11e
Voz	VoIP sem supressão de silêncio (Tráfego CBR)	UGS	AC_VO
Vídeo	MPEG, VoIP com supressão de silêncio (Tráfego VBR)	rtPS	AC_VI
Melhor Esforço	FTP (<i>File Transfer Protocol</i>)	BE	AC_BE

4.2.2 Algoritmo VHD

Apesar de o MIH auxiliar no processo de handover transparente entre diferentes tecnologias, adoção do protocolo MIPv6, além de outras funcionalidades primárias e

secundárias, o mesmo não possui um sistema ou algoritmo inteligente de decisão de handover, capaz de considerar as condições do meio em que o MN está inserido.

Assim, desenvolvemos um algoritmo VHD localizado no MN, capaz de auxiliar o MIH no processo decisório. O VHD levará em conta CoS e AC combinado com a vazão agregada atual da rede que está servindo e da rede alvo, afim de garantir continuidade de serviço e uma distribuição satisfatória do tráfego dentro de redes heterogêneas.

A Figura 16 ilustra o algoritmo VHD. Os valores dos limiares são baseados em resultados empíricos obtidos através de medições depois da execução de várias simulações (em torno de 25 simulações para cada limiar) para cenários similares aos avaliados em nosso estudo. É importante ressaltar que estes valores podem mudar dependendo do cenário de estudo (diferentes tecnologias, capacidade do enlace cabeado, quantidade de usuários, modelo de tráfego, entre outros). Seja qual for o caso, presume-se que as medições poderão ocorrer a fim de definir novos limiares para diferentes cenários.

AlgoritmoVHD()

```

1: if (RedeAtual = "WiMAX") then
2:   if (CoS = "rtPS") then
3:     if (VazãoAgregadaWiMAX >= THVItotWM) and (VazãoAgregadaWiFi < THVItotWF) then
4:       Inicia handover para rede alvo Wi-Fi
5:     else if (CoS = "UGS") then
6:       if VazãoAgregadaWiMAX (>= THVOtotWM) and (VazãoAgregadaWiFi < THVOtotWF) then
7:         Inicia handover para rede alvo Wi-Fi
8:     else if (CoS = "BE") then
9:       if (VazãoAgregadaWiMAX >= THBEtotWM) and (VazãoAgregadaWiFi < THBEtotWF) then
10:        Inicia handover para rede alvo Wi-Fi
11:      end if
12:   else
13:     if (RedeAtual = "Wi-Fi") then
14:       if (AC = "AC_VI") then
15:         if (VazãoAgregadaWiFi > THVItotWF) and (VazãoAgregadaWiMAX < THVItotWM) then
16:           Inicia handover para rede alvo WiMAX
17:         else if (AC = "AC_VO") then
18:           if (VazãoAgregadaWiFi > THVOtotWF) and (VazãoAgregadaWiMAX <= THVOtotWM) then
19:             Inicia handover para rede alvo WiMAX
20:           else if (AC = "AC_BE") then
21:             if (VazãoAgregadaWiFi > THBEtotWF) and (VazãoAgregadaWiMAX <= THBEtotWM) then
22:               Inicia handover para rede alvo WiMAX
23:             end if
24:           end if

```

Figura 16 – Pseudocódigo do Algoritmo VHD.

$TH_{VI}^{totWM}(10Mbps)$: Limiar máximo ou mínimo de vazão agregada WiMAX para vídeo, se a rede atual é WiMAX ou Wi-Fi respectivamente.

$TH_{VO}^{totWM}(8Mbps)$: Limiar máximo ou mínimo de vazão agregada WiMAX para voz, se a rede atual é WiMAX ou Wi-Fi respectivamente.

$TH_{BE}^{totWM}(6Mbps)$: Limiar máximo ou mínimo de vazão agregada WiMAX para melhor esforço, se a rede atual é WiMAX ou Wi-Fi respectivamente.

$TH_{VI}^{totWF}(6Mbps)$: Limiar máximo ou mínimo de vazão agregada Wi-Fi para vídeo, se a rede atual é Wi-Fi ou WiMAX respectivamente.

$TH_{VO}^{totWF}(4Mbps)$: Limiar máximo ou mínimo de vazão agregada Wi-Fi para voz, se a rede atual é Wi-Fi ou WiMAX respectivamente.

$TH_{BE}^{totWF}(2Mbps)$: Limiar máximo ou mínimo de vazão agregada Wi-Fi para melhor esforço, se a rede atual é Wi-Fi ou WiMAX respectivamente.

Independente de qual seja a rede candidata a ser selecionada para a realização de handover (Wi-Fi ou WiMAX), considera-se que o MN pode iniciar sua sessão em qualquer cobertura WiMAX ou Wi-Fi. Assim, o estudo considera duas direções de mapeamento de QoS. Em primeiro lugar, o algoritmo verifica o AC/CoS do MN, dependendo de sua rede atual e alvo. Em seguida, a comparação da vazão agregada da rede atual com o limiar máximo pré-definido, bem como a vazão agregada da rede alvo em comparação com o limiar mínimo pré-definido, para o mapeamento de QoS correspondente.

Uma questão interessante a notar, é que um MN com fluxo de menor prioridade é mais propício a realizar handover do que um MN de maior prioridade. Isto beneficia a equidade entre vazões de diferentes aplicações, pois permite que em caso de congestionamento, por exemplo, o tráfego BE desocupe a célula atual a fim de melhorar seu desempenho, pois em geral, esta classe será a primeira a ser degradada de acordo com as prioridades dos fluxos. Desta forma, em nosso cenário, uma rede WiMAX com vazão agregada de 6 Mbps, é o suficiente para prejudicar um fluxo BE que esteja competindo pelo acesso ao meio com fluxos de maior prioridade. Em resumo, nossa proposta leva em conta o mapeamento de QoS afim de manter a continuidade do serviço e também evita que fluxos de alta e baixa prioridade sofram degradação em células sobrecarregadas, portanto, promovendo também a equidade entre vazões.

Os valores de vazões agregadas da rede atual e alvo são adquiridos através de mecanismos localizados na BS e AP que trabalham juntamente com o servidor MIIS. Na seção seguinte veremos com mais detalhes como eles funcionam.

4.2.3 Metodologia para Obtenção de Vazão Agregada

Redes heterogêneas sem fio com suporte a mobilidade transparente, geralmente são compostas por MNs que se movimentam a todo instante. Devido a isto, mensagens de gerenciamento de mobilidade são trocadas constantemente entre MNs, PoAs e MIIS. Para que o algoritmo VHD receba valores de vazões agregadas atualizadas periodicamente, as duas metodologias propostas, GTH e RTH, interagem constantemente com as mensagens de sinalização de handover do MIH e com o servidor MIIS.

GTH()

```

1: Get_Vazao (id, vazao) {
2:   armazena[id] ← vazao }
3:
4:
5: Set_nb_wimax (id) {
6:   nb_wimax ← nb_wimax + 1
7:   nd_ativo[id] ← 0
8:   if (nb_wifi > 0) then
9:     nb_wifi ← nb_wifi - 1
10:  end if }
11:
12:
13: Set_nb_wifi (id) {
14:   nb_wifi ← nb_wifi + 1
15:   nd_ativo[id] ← 1
16:   if (nb_wimax > 0) then
17:     nb_wimax ← nb_wimax - 1
18:   end if }

```

Figura 17 – Pseudocódigo da Metodologia GTH.

A Figura 17 mostra o pseudocódigo do GTH. O GTH é uma metodologia que obtém a vazão atual de um nodo específico e armazenar em uma base de dados. Isto, ele

faz através do método *Get_Vazao* que utiliza como argumentos o id do nodo móvel e sua vazão atual. Para que se tenha um controle de quais MNs estão associados a rede WiMAX e quais estão associados a rede Wi-Fi, os métodos *Set_nb_wimax* e *Set_nb_wifi*, atualiza a quantidade de nodos associados entre ambas as tecnologias e atribui um número identificador 0 ou 1 (0 para o caso de estar na rede WiMAX e 1 para o caso de estar na rede Wi-Fi).

```

RTH ( )
1: Calc_VazaoAgregada ( ) {
2:   for id_nodo = 0 to N do
3:     if (nd_ativo[id_nodo] = 0) then
4:       vazao_WM ← vazao_WM + armazena[id_nodo]
5:     else if (nd_ativo[id_nodo] = 1) then
6:       vazao_WF ← vazao_WF + armazena[id_nodo]
7:     end if
8:   end for }
9:
10: Get_VazaoAgregadaWM ( ) {
11:   return vazao_WM }
12:
13: Get_VazaoAgregadaWF ( ) {
14:   return vazao_WF }

```

Figura 18 – Pseudocódigo da Metodologia RTH.

O RTH é uma metodologia que trabalha em conjunto com o GTH. A mesma está localizada no servidor MIIS. A Figura 18 apresenta seu pseudocódigo. É nela que é feita a somatória de vazões de fluxos dos MNs associados ao WiMAX ou Wi-Fi. No método *Calc_VazaoAgregada*, conforme cada id, é feita uma comparação com 0 ou 1, ou seja, se o nodo com *id_nodo* X está associado a uma rede WiMAX ou Wi-Fi, a sua vazão atual e somada com uma vazão previamente armazenada. No final dos cálculos, os resultados finais estão dispostos para serem chamados pelos métodos *Get_VazaoAgregadaWM* e *Get_VazaoAgregadaWF*.

Para exemplificar como estas entidades funcionais trabalham em conjunto, a Figura 19 ilustra o exemplo de um cenário onde o MN com classe de serviço BE está se movendo a partir da área de cobertura WiMAX, que já está saturada por serviços de

maior prioridade, e então decide realizar handover para uma cobertura sobreposta Wi-Fi. A sequência de sinalização é descrita a seguir.

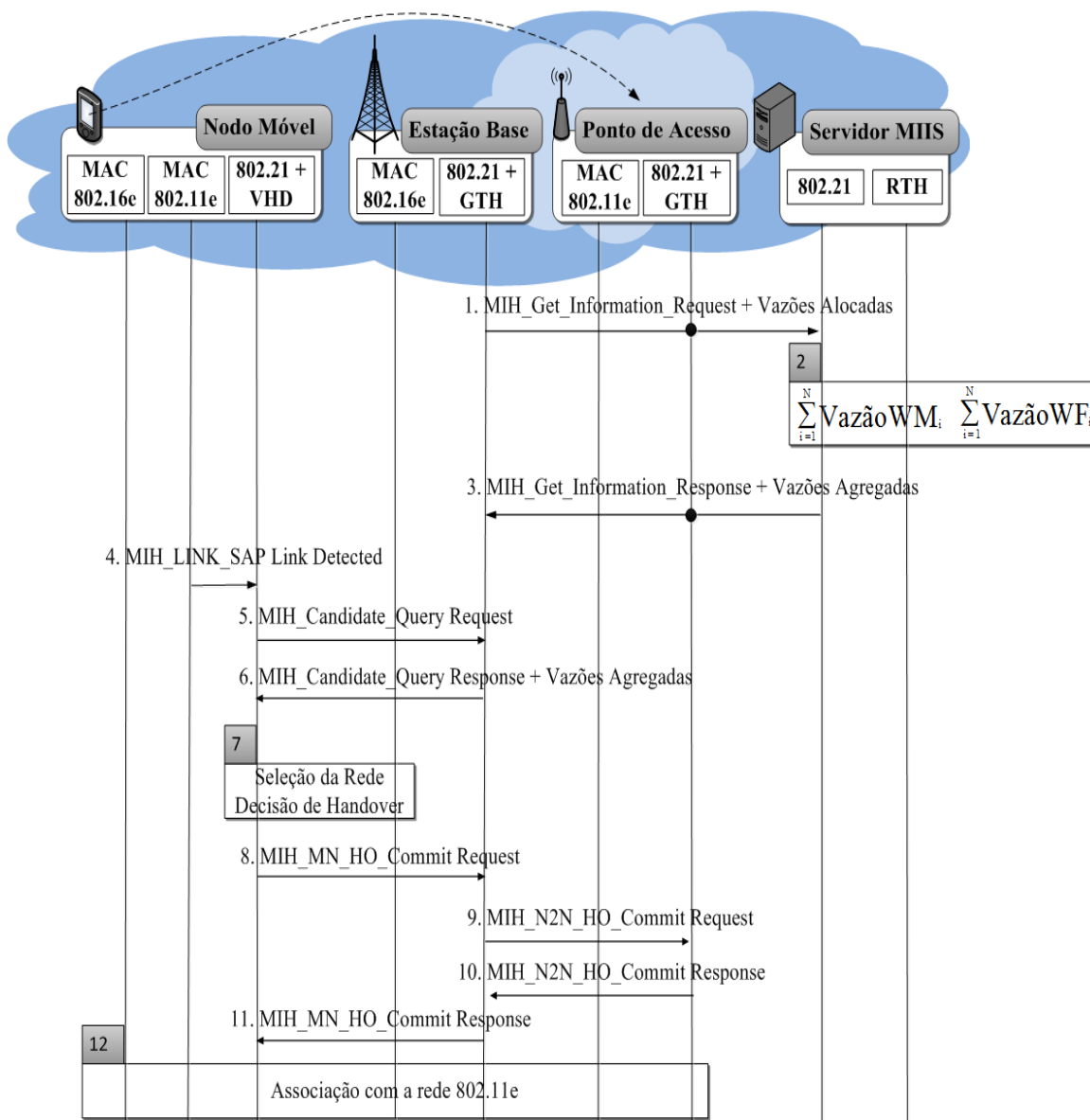


Figura 19 – Sinalização durante o *handover* vertical.

1. Em Primeiro lugar, BS e AP coletam as vazões alocadas de todos os fluxos de MNs associados e armazena em uma base de dados através do GTH. Periodicamente (a cada 45 segundos), tanto a BS WiMAX quanto o AP Wi-Fi, fazem requisição de informações específicas de possíveis redes vizinhas (MIH_GET_Information_Request) para o servidor de informações MIIS. As vazões são enviadas juntamente com estas mensagens.

2. O servidor MIIS ao receber estas requisições, recebe também as vazões de fluxos dos MNs atualmente associados as rede WiMAX e Wi-Fi. No RTH, é feita a somatória

destas vazões separadamente, ou seja, a soma de todas as vazões de MNs na rede WiMAX e a soma de todas as vazões de MNs na rede Wi-Fi. Os resultados (vazão agregada) são armazenados em duas base de dados distintas, uma para WiMAX e outra para Wi-Fi.

3. O servidor MIIS responde as requisições de informação, para a BS e AP (MIH_GET_Information_Response). E juntamente com a mensagem de resposta, as vazões agregadas das redes WiMAX e Wi-Fi também são enviadas. Portanto, a BS, por exemplo, receberá os valores de vazão agregada de sua célula e da célula Wi-Fi vizinha.

4. Como a mensagem anterior contém informações sobre o AP Wi-Fi vizinho, o MN comuta para interface WLAN (802.11e) e começa a “escutar” os *beacons*. Uma vez que um *beacon* é recebido, a interface 802.11e detecta a rede Wi-Fi e gera o evento MIH_LINK_SAP Link_Detected.

5. Apesar de o MN já ter detectado a sua rede alvo Wi-Fi, ele envia para a BS uma requisição de consulta de redes candidatas disponíveis (MIH_Candidate_Query Request). A BS realiza sucessivas trocas de mensagens com o AP e outras PoAs (se caso existam), a fim de requisitar informações de recursos.

6. O resultado da consulta é enviado para o MN (MIH_Candidate_Query Response), juntamente com o resultado da soma da vazão de todos os nodos, tanto dos que estão no WiMAX, quanto dos que estão no Wi-Fi.

7. Neste ponto, o MN tem informação suficiente sobre a rede alvo, para então tomar a decisão de realizar handover ou não. Como o AP é o único disponível, o mesmo é selecionado e a decisão final fica por conta do resultado das vazões agregadas em função das classes de serviço, que neste caso é BE. Como a vazão agregada no WiMAX é maior que o limiar máximo ($\text{Sum_WiMAX} > 6\text{Mbps}$) e a vazão agregada no Wi-Fi é menor que o limiar mínimo ($\text{Sum_Wi-Fi} < 2\text{Mbps}$), então o MN irá iniciar o processo de associação com a rede Wi-Fi.

8. O MN envia uma mensagem de notificação para a BS com informações sobre o AP alvo (MIH_MN_HO_Commit Request).

9. A BS então informa ao AP alvo (MIH_N2N_HO_Commit Request) que o MN irá se mover para sua área de cobertura.

10. O AP alvo responde para a BS, autorizando o início do handover (MIH_N2N_HO_Commit Response).

11. A BS repassa a autorização para o MN (MIH_MN_HO_Commit Response).

12. A interface 802.11e associa-se ao AP alvo.

Com base nas mensagens de sinalização descritas acima, percebemos que é possível integrar a metodologia proposta, o algoritmo VHD e as mensagens de gerenciamento de mobilidade MIH. Toda esta sinalização tem o objetivo principal de promover a equidade entre vazões de diferentes aplicações em ambiente heterogêneo.

4.3 Resumo do Capítulo

Este Capítulo apresentou a proposta desta dissertação descrevendo seus elementos e suas contribuições para a formação do arcabouço para provisionamento de QoS e QoE em redes WiMAX/Wi-Fi com garantia de equidade entre vazões. Descreveu a definição de Qualidade de Experiência e suas principais métricas, a arquitetura para mapeamento de QoS, algoritmo VHD e as metodologias GTH e RTH.

O Capítulo 5 avalia a proposta apresentada neste capítulo com simulações e resultados determinísticos pertinentes a pesquisa realizada nesta dissertação. Ele apresenta os resultados encontrados e os compara aos resultados que não desenvolveram ou avaliaram a integração de novas tecnologias e metodologias inseridas no arcabouço proposto, provando as melhorias e inovações encontradas.

5. Avaliação do Arcabouço

Este Capítulo avalia a proposta apresentada nesta dissertação por meio de módulos de simulação implementados no simulador de redes ns-2 versão 2.29. O capítulo considera dois cenários para os estudos.

São apresentados o impacto causado pela implementação dos módulos de QoS (WiMAX e Wi-Fi) integrados ao módulo *Nist Mobility*, bem como os resultados obtidos com o desenvolvimento do algoritmo VHD e metodologia de obtenção de vazão agregada. Os resultados são avaliados em termos de métricas de QoS (vazão e atraso) e de QoE (PSNR, SSIM e VQM) quando simulados sobre cenários heterogêneos de micro mobilidade.

5.1 Parâmetros de Simulação e Topologia da rede

Como o simulador ns-2 em sua implementação genérica não possuía suporte a mobilidade e QoS para as redes WiMAX e Wi-Fi, fez-se necessário instalar o módulo desenvolvido pelo NIST e posteriormente uma extensão na linguagem de programação C++ para integrar os módulos MAC de QoS para WiMAX (802.16e) e Wi-Fi (802.11e).

Desta forma, tanto o MIHF, usuário MIH, servidor MIIS, *triggers* e eventos MIH, como os escalonadores para BS e MN, mensagens de gerenciamento UL-MAP e DL-MAP, além dos parâmetros AIFS, CW_{Max} , CW_{Min} e TxOP, foram totalmente implementados nesta dissertação.

Para garantir a equidade entre vazões de diferentes aplicações, implementamos também no módulo Nist Mobility, o algoritmo VHD juntamente com duas metodologias para capturar e calcular a vazão agregada atual e da rede alvo. Desta forma o algoritmo de decisão de handover vertical VHD levará em conta não só as classes de serviços ou categorias de acesso, mas também a vazão agregada da rede atual e alvo pelo qual o MN pretende realizar handover.

Para as simulações com tráfego real de vídeo, utilizou-se a ferramenta Evalvid (Evalvid, 2007) para inserir o tráfego de vídeo no simulador ns-2 e assim garantir resultados satisfatórios para análise sobre as métricas de QoE.

Os principais parâmetros utilizados nas simulações são apresentados na Tabela 3 a seguir.

Tabela 3 – Parâmetros usados nas simulações.

Parâmetro de Simulação		Valor
Rede Cabeada	Capacidade do Enlace	10 Mbps
	Atraso do Enlace	40 ms
	Tamanho da Buffer	50
	Tipo de Fila	Drop Tail
WiMAX	Raio de Cobertura	1000 m
	Frequência	3,5GHz
	Padrão	IEEE 802.16e
	Modulação	OFDM
	Escalonador	Round Robin
	Taxa de Transmissão	75 Mbps
Wi-Fi	Raio de Cobertura	50 m
	Frequência	2,4GHz
	Padrão	IEEE 802.11e
	Taxa de Transmissão	11 Mbps
Tempo de Cada Simulação		50s
Número de simulações para cada cenário		50
Intervalo de confiança		95%

Para cada cenário de avaliação, foram feitas 50 simulações pela qual foi obtida a média das vazões e de atraso de pacotes de diferentes tipos de tráfegos (*streaming* de vídeo, voz e FTP).

O intervalo de confiança e o desvio padrão para todos os casos são de 95%. O intervalo de confiança para cada execução é obtido usando a seguinte fórmula:

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$$

$$\text{Intervalo de confiança: } [\bar{x} \pm z_{1-\alpha/2} s / \sqrt{n}]$$

Onde:

s: Desvio padrão;

x_i : Valor obtido;

\bar{x} : Média;

n : Tamanho da amostra;

$z_{1-\alpha/2}$: Quantificador da distribuição normal de unidade. Valores típicos usados são para os intervalos de confiança de 90% (1,645), 95% (1,96) e 99% (2,576).

O tamanho da amostra necessária (n') (linhas do arquivo trace) para uma determinada precisão ou acurácia (r) é obtido através da seguinte fórmula:

$$n' = (100z_{1-\alpha/2}s/r\bar{x})^2$$

A Tabela 4 a seguir, mostra a configuração dos parâmetros específicos para cada AC utilizados nas simulações tanto para o AP como para o MN com interface 802.11e ativa.

Tabela 4 – Configuração de parâmetros de ACs.

Categoria de Acesso (AC)	AIFS	CW_{Min}	CW_{Max}
AC_VI	2	7	15
AC_VO	2	15	31
AC_BE	3	31	1023

Na Tabela 5, é descrito a configuração dos parâmetros específicos para cada CoS utilizados nas simulações tanto para a BS quanto para o MN com interface 802.16e ativa.

Tabela 5 – Configuração de parâmetros de CoSs.

Classes de Serviços (CoS)	Prioridade de Tráfego	Taxa de Tráfego Máximo Sustentado	Taxa de Tráfego Mínima Reservada
rtPS	5	0	0
UGS	5	0	0
BE	1	0	0

Para execução e avaliação das simulações, foi utilizada a topologia que está ilustrada na Figura 20. Para todas as simulações, a rede infraestruturada é formada por um nodo correspondente (origem do tráfego), quatro roteadores, uma BS (interface 802.16e), um AP (interface 802.11e), um servidor MIIS e MNs equipados com MIH e duas interfaces (802.16e/802.11e).

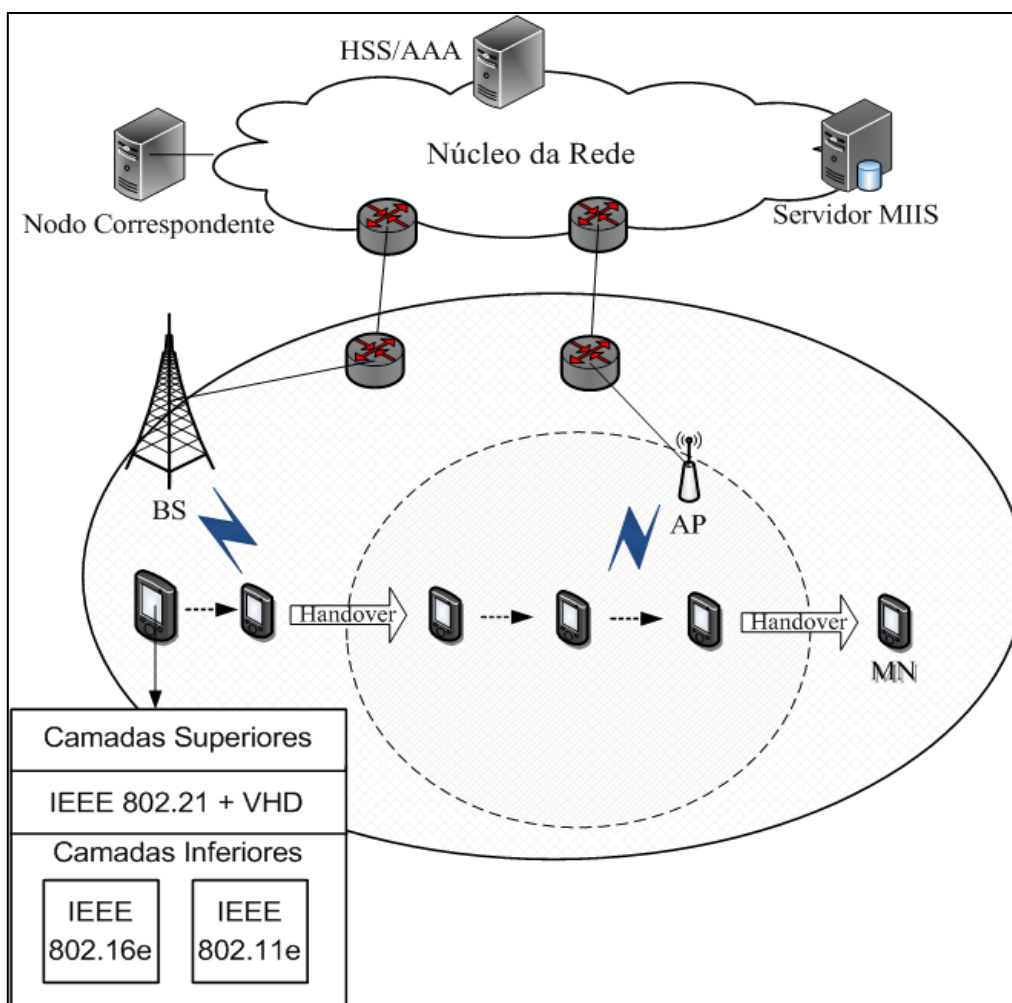


Figura 20 – Topologia genérica da rede para simulação.

Como podemos observar, trata-se de uma única cobertura WiMAX e uma única cobertura Wi-Fi sobreposta. Para o primeiro cenário que será detalhado a seguir, os MNs realizam dois procedimentos de handovers verticais. No segundo cenário os MNs realizam um procedimento de handover vertical.

5.2 Análise das Simulações

Os resultados das simulações estão divididos em dois cenários:

- Um cenário WiMAX/Wi-Fi móvel (Cenário 1), para demonstrar a arquitetura de mapeamento de QoS. Três MNs com suporte a QoS, se movimentam através de diferentes PoAs (WiMAX e Wi-Fi). Neste cenário o objetivo é validar somente a arquitetura de mapeamento de QoS, ou seja, a metodologia e o algoritmo VHD não são considerados. A avaliação é realizada em dois

casos para fins de comparação: com e sem a integração dos módulos de QoS no módulo Nist Mobility.

- Um cenário WiMAX/Wi-Fi móvel (Cenário 2), para demonstrar a eficácia da arquitetura de mapeamento de QoS, algoritmo VHD e metodologia de coleta e cálculo de vazão agregada. Seis MNs com suporte a QoS, se movimentam através de diferentes PoAs (WiMAX e Wi-Fi), e três SSs (*Subscriber Stations*) com CoS UGS estão estáticos gerando tráfego de fundo. Neste cenário o objetivo é validar o arcabouço por completo, considerando a metodologia e o algoritmo VHD. A avaliação é realizada em dois casos para fins de comparação: com e sem as metodologias e algoritmo VHD.

5.2.1 Cenário 1

Neste primeiro cenário, três MNs estão equipados com duas interfaces (WiMAX e Wi-Fi), cada um recebe do nodo correspondente (direção *downlink*) um diferente tipo de tráfego (*streaming* de vídeo, voz e FTP). Todos os fluxos são do tipo CBR (*Constant Bit Rate*) e estão configurados com taxa de 3 Mb/s. Todos os MNs estão, inicialmente, em uma única cobertura WiMAX. Como os MNs se movem a 5 metros por segundo (18 km/h), eles entrarão em uma área sobreposta (cobertura Wi-Fi). À medida que os MNs continuam se movendo, eles retornam para uma única cobertura pertencente à célula WiMAX. Para fim de ilustração, a Figura 21 visa facilitar o entendimento do primeiro cenário.

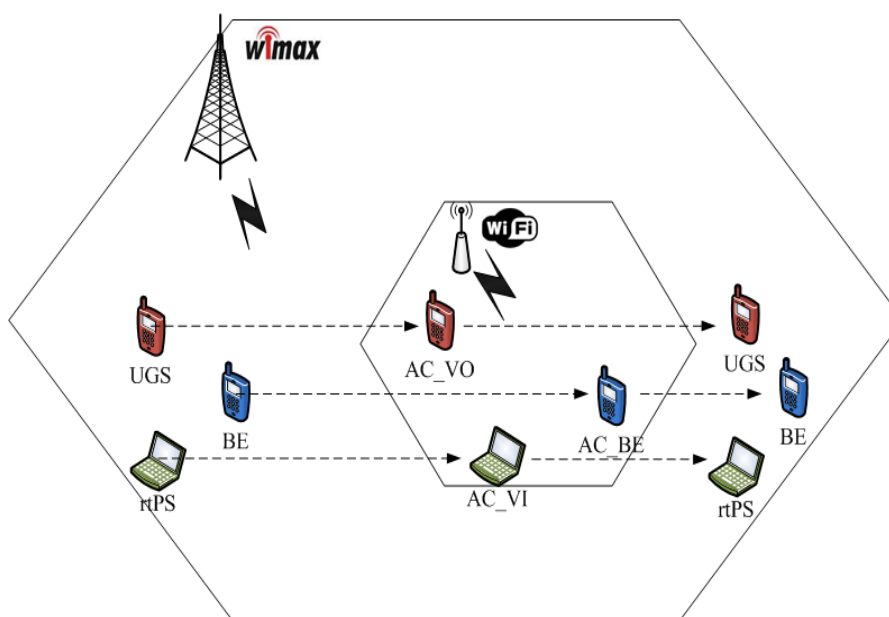


Figura 21 – Representação gráfica do Cenário 1.

Na Figura 22, como não há suporte à QoS para ambas as tecnologias, os fluxos não são classificados ou escalonados para CoSs ou ACs. Durante a simulação, apesar de os três fluxos atingirem a vazão máxima na rede WiMAX, não existe diferenciação de tráfego e, depois do handover para o Wi-Fi, a vazão diminui de forma acentuada para todos os fluxos, prejudicando tráfegos sensíveis ao atraso e perda, como voz e vídeo. No momento em que os MNs retornam para a célula WiMAX, a vazão permanece a mesma para todos os fluxos, ou seja, não existe prioridade entre eles para o acesso ao meio.

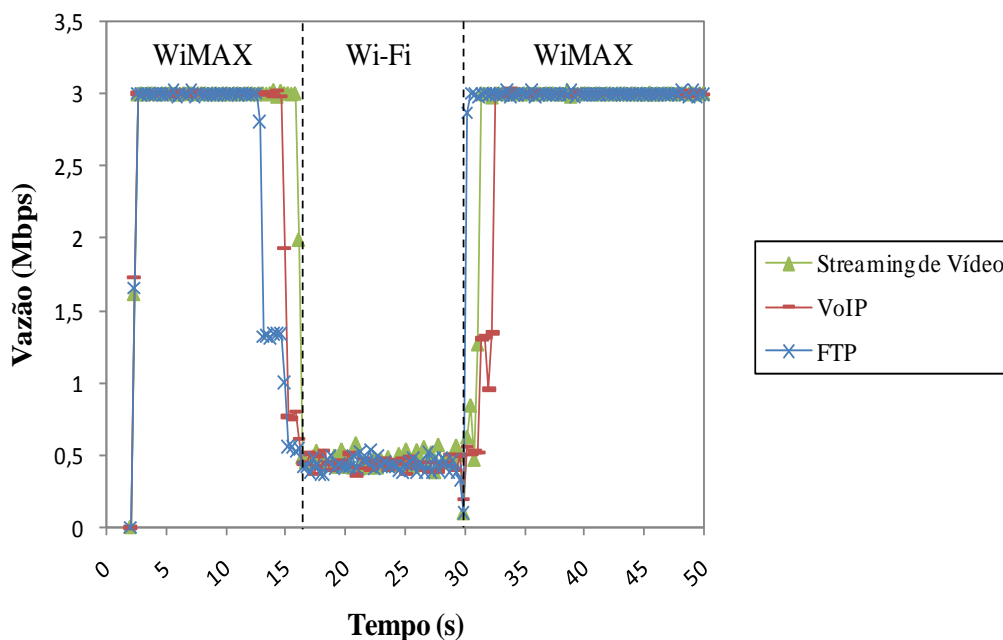


Figura 22 – Vazão média do três fluxos sem mapeamento de QoS.

Conforme a Figura 23, os MNs configurados com fluxos UGS e rtPS tem vazão máxima até o instante 19s quando, então, eles realizam handover e seus fluxos são mapeados para AC_VO e AC_VI, respectivamente na rede Wi-Fi. O MN com fluxo BE na cobertura WiMAX, tem a menor vazão até o instante 13s quando realiza handover e seu fluxo é mapeado para AC_BE na rede Wi-Fi. Como mostrado na Figura 6, o MN com fluxo BE chega a ter um aumento repentino da vazão para 1,8 Mbps, pois encontra o canal livre até o instante 19s. Os MNs realizam mais um handover de volta para a rede WiMAX, de modo que os fluxos com categorias de acesso AC_VO, AC_VI e AC_BE são mapeados para UGS, rtPS e BE, respectivamente.

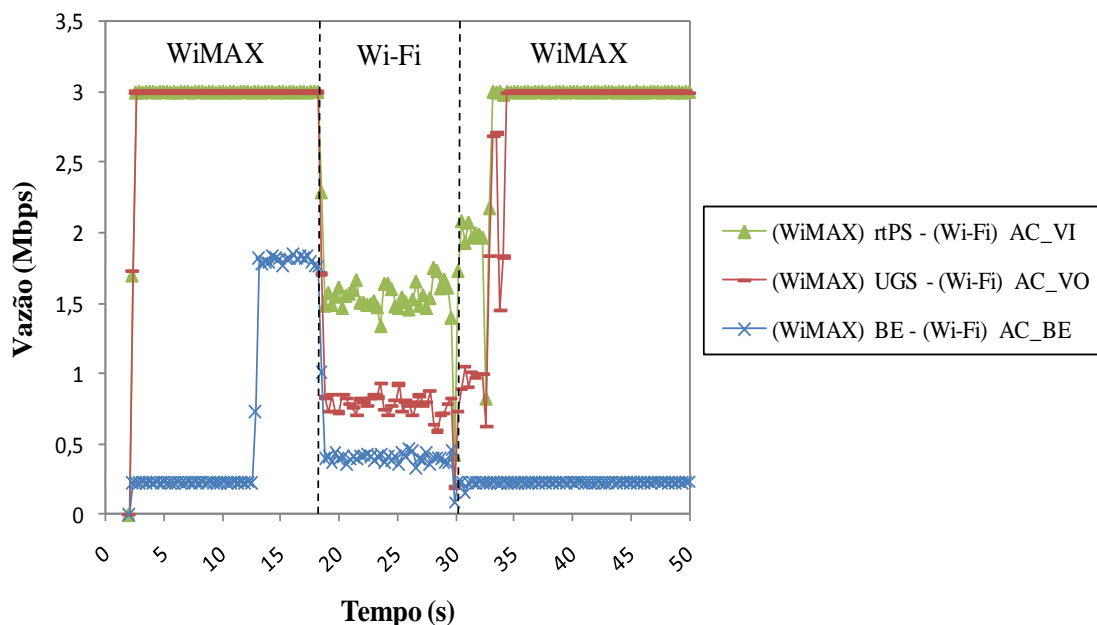


Figura 23 – Vazão média do três fluxos com mapeamento de QoS.

A partir dos resultados representados pelos gráficos, podemos concluir que os fluxos *streaming* de vídeo e VoIP com mapeamento de QoS, obtiveram ganho de 12,10% e 7,35% de vazão respectivamente, comparado aos mesmos fluxos sem o mapeamento de QoS. Um fato interessante a ser analisado é o fluxo FTP, pois sem o mapeamento de QoS obteve ganho de 55,03% em relação ao mesmo fluxo com o mapeamento de QoS. Isso se deve aos parâmetros de QoS da CoS BE, pois leva em conta que fluxos BE não necessitam de muita vazão e não chegam a ser sensíveis a perda e atraso de pacotes. A Tabela 6 mostra os resultados de vazão média dos três tipos de fluxos e respectivos ganhos em porcentagem.

Tabela 6 – Resultados para vazão média e respectivo ganho para o Cenário 1.

	Streaming de Vídeo	VoIP	FTP
Sem Mapeamento de QoS	2,185	2,071	2,099
Com Mapeamento de QoS	2,549	2,292	0,448
Ganho (%)	12,10	7,35	55,03

O cenário para a avaliação do vídeo é similar ao primeiro cenário. Como descrito antes, a ferramenta Evalvid, foi utilizada para avaliar a qualidade do vídeo recebido. Uma sequência de frames do vídeo “*Highway*” (YUF CIF, 2007) foi utilizada nas simulações. O vídeo consiste de 2000 quadros com o formato YUV, amostragem

4:2:0, dimensão de 176x144 pixels, comprimido através de um codec MPEG-4 e enviado a uma taxa de 30 quadros por segundo. O objetivo da avaliação do vídeo, é validar a eficácia do mapeamento de QoS proposto no ponto de vista do usuário. A Tabela 7 resume os principais parâmetros utilizados para avaliação do vídeo.

Tabela 7 – Parâmetros de simulação de tráfego de vídeo.

Vídeo Avaliado	
Vídeo	<i>Highway</i>
Quadros	2000
Amostragem	4:2:0
Resolução	176x144 pixels
Compressão Codec	MPEG-4
Taxa	30 quadros por segundo
Tamanho de pacotes	1052 bytes
Fragmentação máxima de pacotes	1024 bytes
Formato	YUV (Y é a luminância, U(R-Y) é o sinal R (<i>Red</i>) vermelho e Y somados, V(B-Y) é o sinal do B (<i>Blue</i>) azul e Y.)

O vídeo recebido pelo usuário sem qualquer garantia de QoS em ambas a redes WiMAX e Wi-Fi obteve média PSNR igual a 22,17 dB, o que qualifica o vídeo como POBRE. O vídeo recebido com a arquitetura de mapeamento de QoS proposto obteve média PSNR igual a 32,1 dB, qualificando o vídeo como BOM. A Figura 24 deixa clara a diferença de qualidade do vídeo recebido baseado na métrica PSNR.

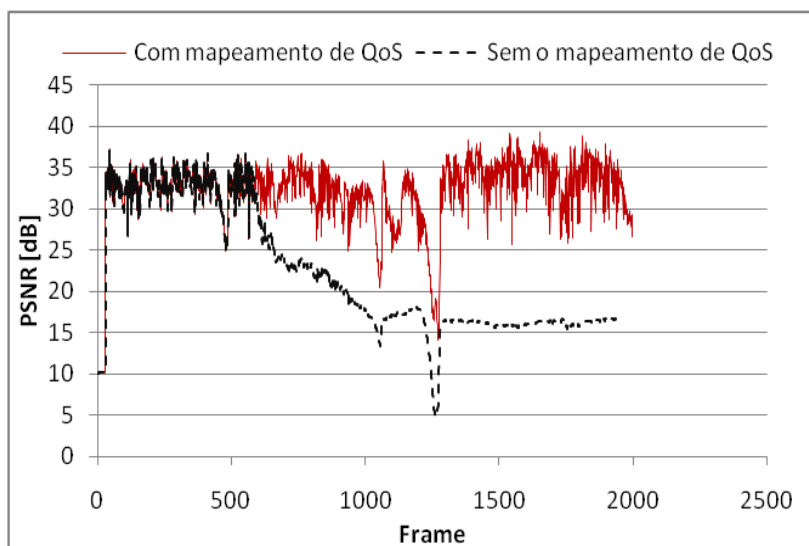


Figura 24 – PSNR do vídeo transmitido com e sem o mapeamento de QoS.

A Figura 25, ilustra a superioridade do vídeo recebido com o mapeamento de QoS em relação ao vídeo sem garantia de QoS através da métrica SSIM. A média do vídeo recebido sem garantia de QoS foi de 0,68, comparado ao vídeo recebido com mapeamento de QoS, a média SSIM foi de 0,91.

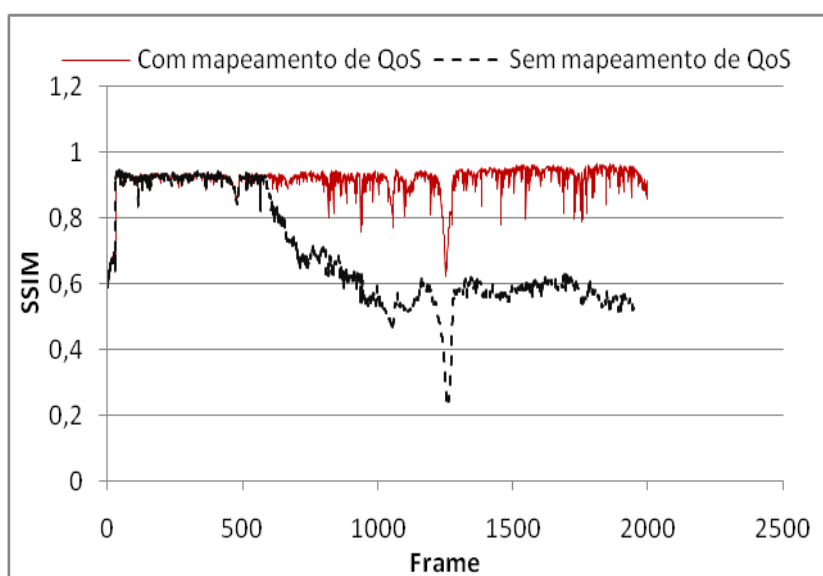


Figura 25 – SSIM do vídeo transmitido com e sem o mapeamento de QoS.

Na Figura 26, os resultados da eficiência do mapeamento de QoS para aplicação de vídeo é confirmado em termos de VQM. A média de VQM do vídeo recebido sem

garantia de QoS foi de 6,47, enquanto que o vídeo recebido com mapeamento de QoS teve média igual a 1,96.

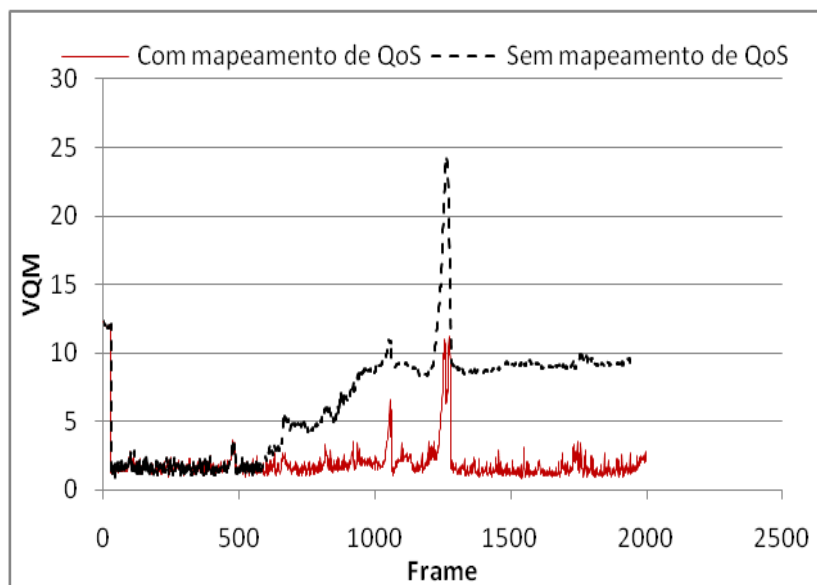


Figura 26 – VQM do vídeo transmitido com e sem o mapeamento de QoS.

Notemos um detalhe de um decremento acentuado entre os *frames* 1000 e 1500 nos três gráficos apresentados. Isso se deve ao segundo handover que o MN faz de volta para a mesma célula WiMAX, pois o padrão MIH não dispara o evento MIH_LINK_SAP Link_Detected para a mesma BS já detectada e assim não realiza o handover vertical transparente. A Tabela 8 resume os resultados apresentados.





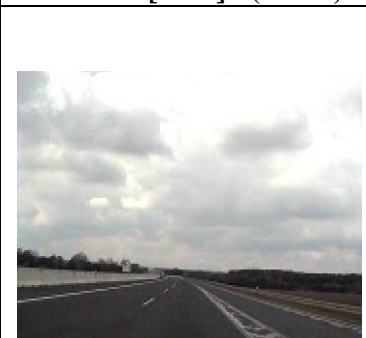

Tabela 8 – Resultados das métricas de QoE para o tráfego de vídeo.

	PSNR [dB]	SSIM	VQM
Sem Mapeamento de QoS	22,17	0,68	6,47
Com Mapeamento de QoS	32,1	0,91	1,96

Os benefícios da solução proposta são ainda mais claros quando analisamos visualmente alguns frames. O Quadro 7 ilustra a sequência de três quadros de *frames* do vídeo “Highway”. Verifica-se que o vídeo recebido sem o mapeamento de QoS foi bastante degradado no momento em que o MN estava na célula WiMAX, e após o

handover para a rede Wi-Fi, a qualidade do vídeo piora ainda mais. O vídeo perde coloração e fica borrado. O mesmo acontece quando o MN retorna para o WiMAX. Este impacto poderia ser ainda maior em vídeos com vários objetos e com mais movimentos. Com o mapeamento de QoS, a continuidade da boa qualidade do vídeo é a mesma independente se o MN está no WiMAX ou Wi-Fi, criando assim uma visão de “transparência” de tecnologia.

Quadro 7 – Frames do vídeo “Highway”.

Qualidade do Vídeo sem Mapeamento de QoS		
FRAME [850] - (WiMAX)	FRAME [1309] - (Wi-Fi)	FRAME [1806] - (WiMAX)
		
Qualidade do Vídeo com Mapeamento de QoS		
FRAME [850] - (WiMAX)	FRAME [1309] - (Wi-Fi)	FRAME [1806] - (WiMAX)
		

5.2.2 Cenário 2

No segundo cenário, nove MNs estão equipados com duas interfaces (WiMAX e Wi-Fi). Todos os MNs recebem do nodo correspondente (direção *downlink*) diferentes tipos de tráfegos, ou seja, um grupo de três MNs recebe *streaming* de vídeo, outro grupo de três MNs recebe voz e os três restantes recebem dados pelo protocolo FTP. Todos os fluxos são do tipo CBR e estão configurados com taxa de 1,5 Mb/s. Nesta análise, os nove MNs estão inicialmente em uma única cobertura WiMAX. Três MNs com CoS BE e três MNs com CoS rtPS, se movem a 5 metros por segundo (18 Km/h) em direção a

uma área sobreposta (cobertura Wi-Fi) e as três SSs com CoS UGS estão estáticos gerando tráfego de fundo. A Figura 27 nos mostra o segundo cenário.

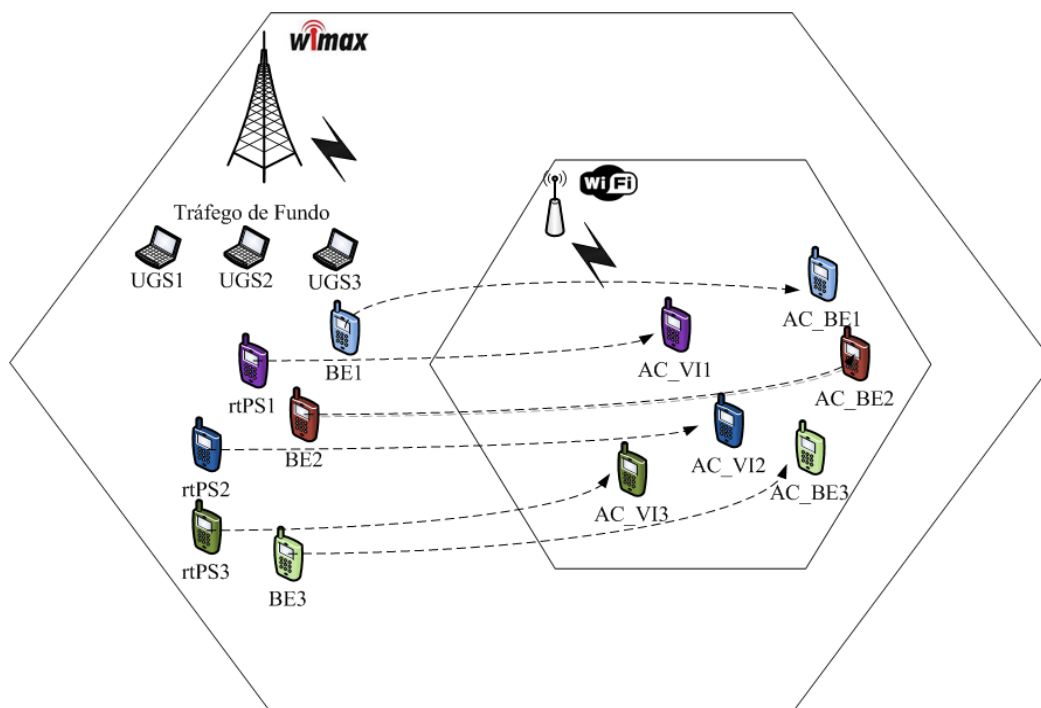


Figura 27 – Representação gráfica do Cenário 2.

A Figura 28 ilustra a vazão média dos seis MNs a serem analisados antes e depois do handover para a rede Wi-Fi. Quando todos os MNs estão na célula WiMAX, os MNs com fluxo BE não possuem vazão suficiente para transmitir dados. Isso acontece porque a rede está saturada por MNs com fluxos rtPS e UGS. Como não existe nenhum controle de handover por um sistema inteligente ou algoritmo, os seis MNs com fluxos BE e rtPS realizam handover para a rede Wi-Fi (pois é a única PoA alvo disponível). Podemos notar que a vazão dos fluxos AC_BE teve uma pequena melhora. Por outro lado, um dos fluxos AC_VI teve um decremento da vazão (1,2 Mbps) comprometendo a qualidade do vídeo.

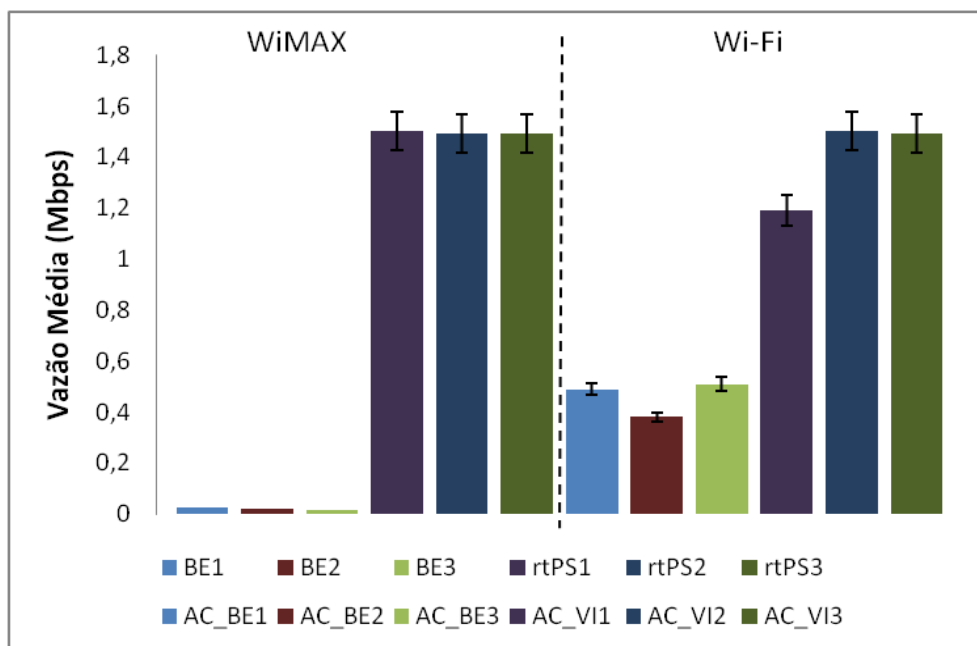


Figura 28 – Vazão média sem o algoritmo VHD e metodologia.

A implementação da metodologia e do algoritmo VHD visa promover uma melhor distribuição de tráfego entre as células da rede heterogênea. Neste cenário, antes mesmo de realizar *handover* para o Wi-Fi, os MNs com fluxo BE, verificam se a vazão agregada da rede WiMAX, cujo o valor é 9,2 Mbps, é maior do que o limiar de 6 Mbps. O valor da vazão agregada da rede Wi-Fi alvo é de 0 Mbps, ou seja, até o momento não há nenhum MN gerando tráfego. Este valor é comparado com o limiar mínimo que é de 2 Mbps. Então, com as duas condições satisfeitas, os três MNs com fluxo BE realizam *handover*. Os MNs com fluxo rtPS, também comparam a vazão agregada atual (9,2 Mbps) com o limiar máximo que é de 10 Mbps, bem como a vazão agregada atual da rede Wi-Fi alvo, cujo o valor é de 4,5 Mbps (os três MNs com AC_BE já estão no Wi-Fi com vazão máxima de 1,5 Mbps), comparado ao limiar mínimo de 6 Mbps. Como a primeira condição não foi satisfeita, os três MNs com fluxo rtPS não realizam *handover* para o Wi-Fi.

O resultado do uso do arcabouço proposto é apresentado na Figura 29. Os três fluxos rtPS não realizam *handover* e, portanto, permanecem com CoS rtPS e vazão máxima de 1,5 Mbps. Os três fluxos BE realizam *handover* e são mapeados para AC_BE na rede Wi-Fi, e assim atingem vazão máxima de 1,5 Mbps.

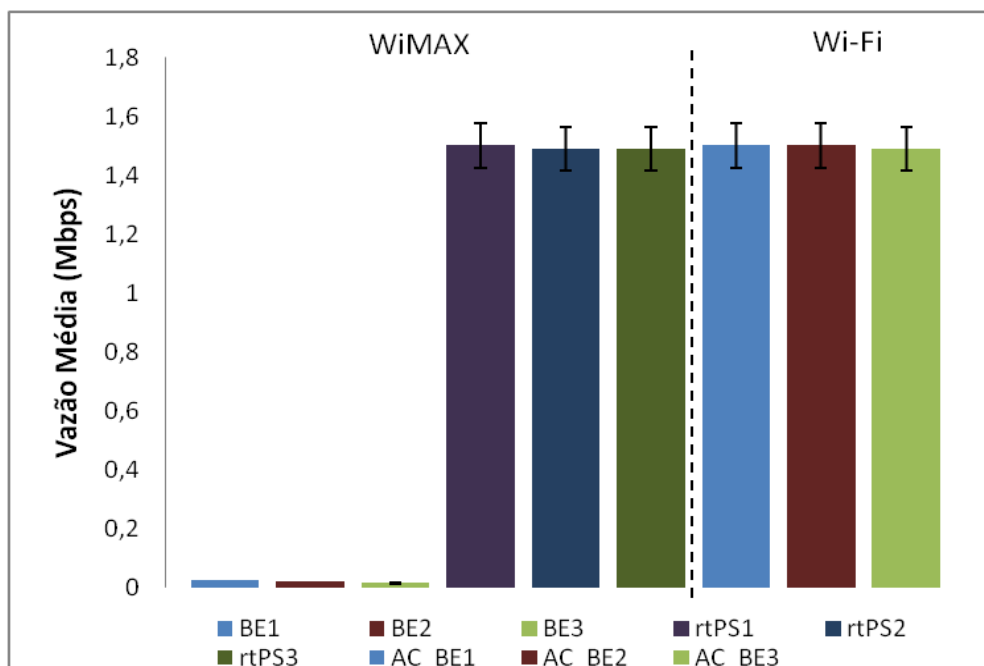


Figura 29 – Vazão média com o algoritmo VHD e metodologia.

A partir dos resultados representados pelos gráficos, podemos concluir que os quatro MNs com CoSs rtPS1, BE1, BE2 e BE3 com garantia de equidade entre vazões, obtiveram ganho de 11,26%, 42,92%, 45,30% e 36,77% de vazão respectivamente, comparado aos mesmos MNs sem a garantia de equidade entre vazões. Os MNs com CoSs rtPS2 e rtPS3 não foram avaliados para fim de comparação, pois no resultado mantêm a mesma vazão com e sem o arcabouço proposto. Em geral, o consumo de banda se tornou equilibrado entre as redes WiMAX e Wi-Fi e fluxos de menor prioridade são beneficiados mesmo em situações de congestionamento. A Tabela 9 mostra os resultados de vazão média dos quatro CoSs/ACs e respectivos ganhos em porcentagem.

Tabela 9 – Resultados para vazão média e respectivo ganho para o Cenário 2.

	rtPS1- AC_VI1	BE1- AC_BE1	BE2- AC_BE2	BE3- AC_BE3
Com Mapeamento de QoS e Sem a Garantia de Equidade entre Vazões	1,201	0,319	0,235	0,294
Com Mapeamento de QoS e Com a Garantia de Equidade entre Vazões	1,377	0,963	0,914	0,846
Ganho (%)	11,26	42,92	45,30	36,77

A Figura 30 mostra o resultado sem o uso da metodologia e do algoritmo VHD em termos de atraso de pacotes. Os primeiros picos de atraso no intervalo de tempo, entre 18s e 24s, é referente ao handover dos três MNs com CoS BE para a rede Wi-Fi. Estes picos acontecem devido o fato de os MNs somente iniciarem tráfego durante o processo de handover, pois na célula WiMAX não havia largura de banda alocada para estes MNs devida a baixa prioridade. Observamos que as maiores atrasos, pertence aos três MNs com AC_BE que estão na rede Wi-Fi. Os três MNs com AC_VI que também realizam handover para o Wi-Fi, não sofrem atraso porque possuem alta prioridade.

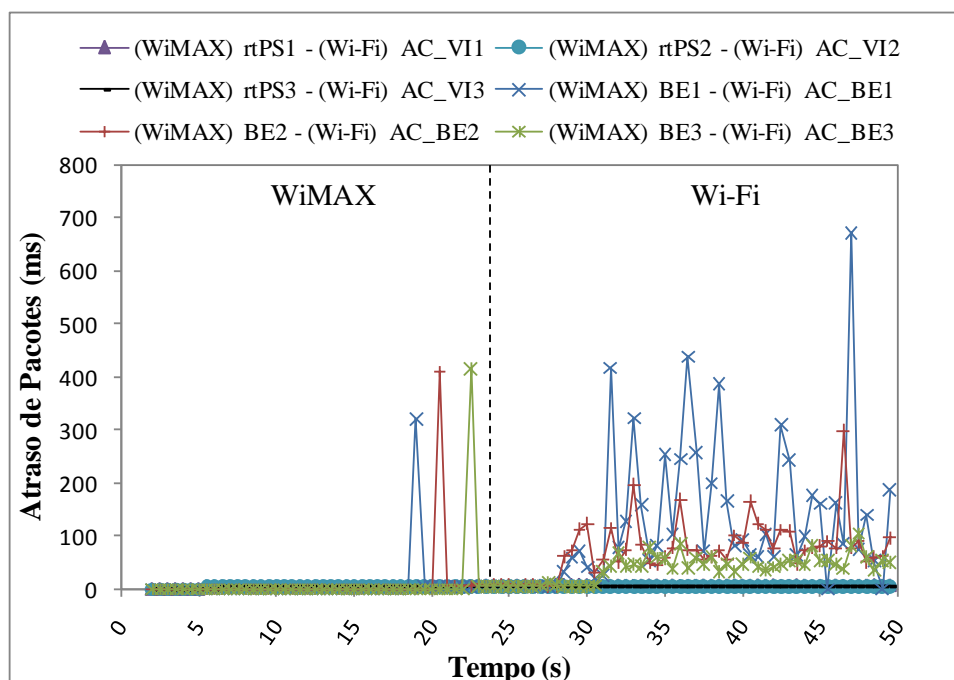


Figura 30 – Média de atraso de pacotes sem o algoritmo VHD e metodologia.

Na Figura 31, os maiores picos de perda de pacotes acontecem somente no momento em que os MNs com CoS BE realizam handover para o Wi-Fi. Com a adoção da metodologia e algoritmo VHD, a perda tem sido praticamente eliminada. Mesmo que fluxos AC_BE não sejam altamente sensíveis ao atraso, o objetivo é comprovar como o arcabouço é capaz de melhorar a alocação de banda em ambiente heterogêneo. Assim podemos comprovar a eficácia do arcabouço proposto também através da métrica de atraso de pacotes.

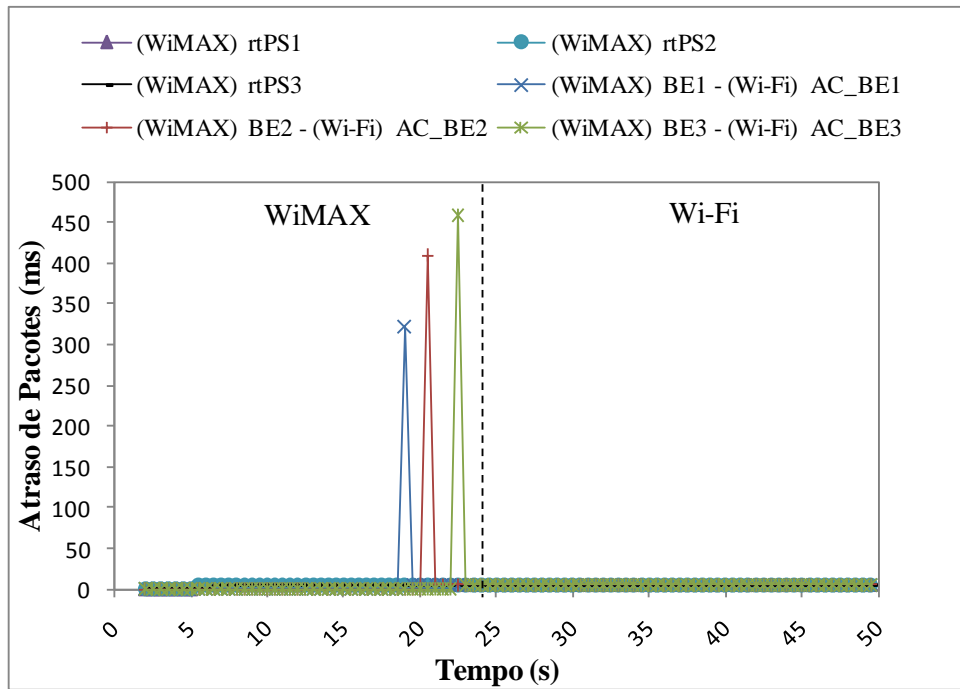


Figura 31 – Média de atraso de pacotes com o algoritmo VHD e metodologia.

5.3 Resumo do Capítulo

Este capítulo descreveu os cenários utilizados por esta dissertação, assim como, as ferramentas envolvidas na execução dos testes para as simulações e para uma verificação completa quanto a qualidade do vídeo, verificando-se a QoS e a QoE. Foram analisados diversos resultados e a partir deles, conclusões parciais a cerca da eficiência do arcabouço proposto.

Com isso, o trabalho encerra suas simulações sobre as propostas da dissertação demonstrando os resultados obtidos e os descrevendo.

6. Conclusão

Este trabalho propõe o desenvolvimento de um novo arcabouço para prover mapeamento de QoS e garantia de equidade entre vazões de diferentes aplicações em redes sem fio heterogêneas WiMAX/Wi-Fi de modo a garantir QoS e QoE, bem como mobilidade transparente em ambiente heterogêneo. Para isto, a escolha de tecnologias sem fio que obedecem ao padrão IEEE 802, metodologias e algoritmos a serem integrados para compor o arcabouço foram de suma importância para que o desenvolvimento tenha sido concretizado. Na rede de acesso sem fio, adotou-se o padrão IEEE 802.16e, popularmente conhecido como WiMAX móvel, e o padrão IEEE 802.11e, também conhecido como Wi-Fi com suporte a QoS. Entre as camadas L2 e L3 dos dispositivos móveis e das PoAs adotou-se o *framework* MIH que obedece o padrão IEEE 802.21. Uma metodologia de coleta e cálculo de vazão agregada e um algoritmo de decisão de handover, baseado na vazão agregada da rede atual e alvo em função de CoSs ou ACs, também foram implementadas para auxiliar o MIH no processo decisório de realizar handover ou não para uma nova PoA disponível.

O arcabouço não foi somente avaliado, como feito tradicionalmente, apenas em termos das métricas de QoS, pelo fato de não expressar a percepção do usuário diante das aplicações multimídias. As novas arquiteturas para redes NGN (*Next Generation Networks*) devem considerar também avaliações baseadas em métricas de QoE, que expressam precisamente a experiência que o usuário obtém diante de uma aplicação de vídeo real.

Conforme os resultados de várias simulações, a QoS e QoE podem ser mantidas para diferentes tipos de tráfegos, durante a mobilidade do MN, independente da tecnologia de acesso ser Wi-Fi ou WiMAX. A mobilidade transparente e controle na decisão de handover têm sido garantidos com o auxílio do MIH. Diferentemente de nossa proposta, muitos artigos da literatura permitem a deterioração de tráfego não prioritário com o intuito de dar garantias ao tráfego de classes/categorias com requisitos estritos de QoS. Nossa proposta permitiu que tanto nodos com fluxos de maior quanto de menor prioridade tivessem seus requisitos de QoS satisfeitos, mesmo em situações em que a rede está congestionada.

As principais contribuições deste trabalho são:

- Arquitetura formada pela integração das tecnologias WiMAX (802.16e) e Wi-Fi (802.11e) ao framework MIH, de modo a possibilitar o mapeamento de QoS entre classes de serviços e categorias de acessos, em cenário heterogêneo móvel.
- Metodologia de coleta de vazão e cálculo de vazão agregada. Esta metodologia inclui o GTH localizado nas PoAs para coleta e armazenamento das vazões atuais dos MNs associados, e o RTH localizado no servidor MIIS para a somatória das vazões.
- Algoritmo VHD localizado nos nodos móveis. A decisão de realizar handover ou não é definida em função de dois elementos. O primeiro é vazão agregada da rede atual, pela qual o MN está associado e vazão agregada da rede alvo em que o MN pretende realizar o handover (valores adquiridos pela metodologia). O segundo são as classes de serviços ou categorias de acesso que são específicos para fluxos de serviços.
- Garantia de equidade entre vazões de diferentes aplicações em ambiente heterogêneo e garantia de QoS para fluxos de menos prioridade mesmo em situações de congestionamento.
- Handover vertical transparente. O framework MIH garante a mobilidade transparente em ambiente heterogêneo.
- Avaliação do arcabouço preocupando-se com o ponto de vista do usuário. É uma nova forma de avaliação introduzida pelo conceito de QoE.

Para trabalhos futuros, propõem-se adicionar a classe de serviço nrtPS e a categoria de acesso AC_BK para as redes WiMAX e Wi-Fi, respectivamente. Além disso, propomos o mapeamento dinâmico, com base nas mudanças dos parâmetros de rede e experiência do usuário. Outra opção, é incluir a tecnologia de acesso sem fio LTE com suporte a QoS, pois trata-se de uma tendência bastante difundida.

7. Referências Bibliográficas

AHSON, Syed. ILYAS, Mohamed. “WiMAX – Standards and Security”. Editora CRC Press. 2008.

ANDI, W. C., YEO, C. K. LEE, B. S. “Environment-aware QoS framework for multi-interface terminal”. Elsevier Journal on Computer Communications, vol. 33, pp. 1049-1055. Janeiro, 2010.

ANDI, Widyo Cahyono. XIA, Yang. YEO, Chai Kiat. LEE, Bu Sung. “An Interface Management Framework for Multihomed Terminal”. 7th International Conference on Information, Communications and Signal Processing, ICICS 2009. Janeiro, 2010.

ANDREADIS, Alessandro. RIZZUTO, Sandro. ZAMBON, Riccardo. “A New NS2 Tool to Investigate QoS Management over Mobile WiMAX”. SIMUTools 2011. Fevereiro, 2011.

ANDREWS, Jeffrey G. GHOSH, Arunabha. MUHAMED, Rias. “Fundamentals of WiMAX – Understanding Broadband Wireless Networking”. Editora Prentice Hall. 2007.

BELGHITH, A. NUAYMI, L. “Design and Implementation of a QoS-included WiMAX Module for NS-2 Simulator”. Disponível em: <http://perso.telecom-bretagne.eu/aymenbelghith/tools/>. Acessado em: Agosto de 2010.

CERQUEIRA, E., VELOSO, L., CURADO, M., MONTEIRO, E. MENDES, P. “Quality Level Control for Multi-user Sessions in Future Generation Networks”. Global Telecommunications Conference, IEEE Globecom 2008. Dezembro, 2008.

CHEN, Y. C., HSIA J. H. LIAO, Y. J. “Advanced seamless vertical handoff architecture for WiMAX and WiFi heterogeneous networks with QoS guarantees”. Elsevier Journal on Computer Communications, vol. 32, pp. 281-293. Novembro, 2008.

CHROST, Lukasz. BRACHMAN, Agnieszka. "Towards a common benchmark in WiMAX environment". Elsevier Computer Communications. Julho, 2010.

EIRAS, Fabio Cocchi da Silva. ZUCCHI, Wagner Luiz. "Benefícios do Padrão IEEE 802.11e para Tráfego de Tempo Real em Redes WLAN não Estruturadas". Escola Politécnica da Universidade de São Paulo. 2009.

Evalvid Tool. "Evalvid Web Site". Disponível em: <http://www.tkn.tu-berlin.de/research/evalvid/>. Acessado em: Janeiro de 2011.

FERREIRA, José Jailton Henrique Júnior. "Handover Transparente para Tráfego Multimídia em uma Arquitetura Integrada WiMAX/MIP/MPLS Móvel". Universidade Federal do Pará. Webmedia, 2009.

GUIRARDELLO, Marcelus. BRANQUINHO, Omar Carvalho. "Política de QoS com Priorização de Acesso ao Meio para Redes IEEE 802.11". Pontifícia Universidade Católica de Campinas, Centro de Ciências Exatas, Ambientais e de Tecnologias. 2008.

HA, Jeounglak. KIM, JiYeon. KIM, Jin-Up. KIM, Sang-Ha. "Dynamic Load Balancing Architecture in Heterogeneous Wireless Network Environment". 9th International Symposium on Communications and Information Technology, ISCIT 2009. Dezembro, 2009.

HIERTZ, Guido R. DENTENEER, Dee. ZANG, Yunpeng. STILBOR, Lothar. COSTA, Xavier Pérez. WALKE, Bernhard. "The IEEE 802.11 Universe". IEEE Communications Magazine. Janeiro, 2010.

IEEE - Institute of Electrical and Electronics Engineers. "Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services". Std P802.21/D11.0-2008.

IEEE - Institute of Electrical and Electronics Engineers. "IEEE Standard for Local and Metropolitan Area Networks – Part 16: Air Interface for Fixed Broadband Wireless Access Systems, Amendment 2: Physical and Medium Access Control Layers for

Combined Fixed and Mobile Operation in Licensed Bands, Corrigendum 1”, Std 802.16eTM-2006. Fevereiro, 2006.

IEEE - Institute of Electrical and Electronics Engineers. “IEEE standard for local and metropolitan area networks Part 16: Air interface for fixed and mobile broadband wireless access systems (amendment and corrigendum to IEEE Std 802.16-2004)”, Std 802.16eTM-2005. Disponível em: <http://standards.ieee.org/getieee802/download/802.16e-2005.pdf>. Acessado em: Fevereiro de 2011.

IEEE - Institute of Electrical and Electronics Engineers. “Medium Access Control (MAC) Quality of Service (QoS) Enhancements”, Std P802.11e/D13.0-2005.

IEEE - Institute of Electrical and Electronics Engineers. “Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems”, Std 802.16TM-2004. October 2004.

IEEE - Institute of Electrical and Electronics Engineers. “Status of Project IEEE 802.11e, MAC Enhancements for Quality of Service”, Std 802.11eTM-2005. Novembro, 2005.

IEEE - Institute of Electrical and Electronics Engineers. “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification”, IEEE Std 802.11TM-1999. Junho, 1999.

ILYAS, Mohamed. AHSON, Syed. “WiMAX – Applications”. Editora CRC Press. 2008.

JUNIOR, Warley M. V. JAILTON, José. CARVALHO, Tassio. DIAS, Kelvin Lopes. “Aprovisionamento de QoS e QoE em Redes Sem Fio Heterogêneas com Suporte a Balanceamento de Carga”. SBRC (WGRS). Junho, 2011.

KURAN, Mehmet S. TUGCU, Tuna. “A survey on emerging broadband wireless access technologies”. Elsevier Computer Networks. Agosto, 2007.

LAMPROPOULOS, George. SALKINTZIS, Apostolis K. PASSAS, Nikos. “Media-Independent Handover for Seamless Service Provision in Heterogeneous Networks”. IEEE Communications Magazine. Janeiro, 2008.

LEE, Myounghwan. KWON, Bongkyung. COPELAND, John A. “Cross-Layer Design of DiffServ Architecture for End-to-End QoS in IEEE 802.16 Multi-hop Mesh/Relay Networks with IEEE 802.11e WLANs”. IEEE Communications Society. Fevereiro, 2010.

LIMA, Luciana dos Santos. SOARES, Luiz Fernando Gomes. ENDLER, Markus. “WiMAX: Padrão IEEE 802.16 para Banda Larga Sem Fio”. Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática. Agosto, 2006.

MANGOLD, Stefan. CHOI, Sunghyun. MAY, Peter. KLEIN, Ole. HIERTZ, Guido. STIBOR, Lothar. “IEEE 802.11e Wireless LAN for Quality of Service”, European Wireless. Fevereiro, 2002.

NEVES, Pedro. SOARES, João. SARGENTO, Susana. PIRES, Hugo. FONTES, Francisco. “Context-aware media independent information server for optimized seamless handover procedures”. Elsevier Computer Networks. Maio, 2011.

Nist Mobility - Seamless and Secure Mobility Tool Suite. Disponível em: http://www.nist.gov/itl/antd/emntg/ssm_seamlessandsecure.cfm. Acessado em: Agosto de 2010.

NUAYMI, Loutfi. “WiMAX - Technology for Broadband Wireless Access”. Editora John Wiley & Sons. 2007.

OLIVA, Antonio. BANCHS, Albert. SOTO, Ignacio. LESSMANN, Johannes. NIEPHAUS, Christian. MELIA, Telemaco. “IEEE 802.21: Media Independence beyond Handover”. Computer Standards & Interfaces. Março, 2011.

PRASATH, G. A., RAGHU, K. R. MA, Maode. "Integration of WLAN and WiMAX with Base Station Assisted QoS". 5th IFIP International Conference on Wireless and Optical Communications Networks, WOCN'08, pp. 1-5, 5-7. Maio, 2008.

SCHILLER, Jochen H. "Mobile Communications". Editora Pearson Education Limited. 2003.

SEHRAWAT, Saurabh. PRASAD, Revoti. HARIHAR, Dheeraj. "Performance Analysis of QoS supported by Enhanced Distributed Channel Access (EDCA) mechanism in IEEE 802.11e". International Journal of Computer Science. Fevereiro, 2007.

SEKERCIOGLU, Y. Ahmet, IVANOVICH, Milosh. and YEGIN, Alper. "A Survey of MAC based QoS implementation for WiMAX networks". Computer Networks Journal. Maio, 2009.

SILVA, Diego dos Passos. DIAS, Kelvin Lopes. "Handover Transparente entre Redes IEEE 802.16 e IEEE 802.11 Usando um Sistema Fuzzy-Genético". Universidade Federal do Pará, Programa de Pós-Graduação em Engenharia Elétrica. 2011.

TANIUCHI, Kenichi. OHBA, Yoshihiro. FAJARDO, Victor. DAS, Subir. TAUIL, Miriam. CHENG, Yuu-Heng. DUTTA, Ashutosh. BAKER, Donald. YAJNIK, Maya. FAMOLARI, David. "IEEE 802.21: Media Independent Handover: Features, Applicability, and Realization". IEEE Communications Magazine. Janeiro, 2009.

TARNG, W., CHEN, N. W., DENG, L. Z., OU, K. L., HSIE, K. R., CHEN, M. "The Integration of Heterogeneous Wireless Networks (IEEE 802.11/IEEE 802.16) and its QoS Analysis". International Journal of Communication Networks and Information Security, vol. 2, No. 3. Dezembro, 2010.

The Network Simulator NS-2. "Ns-2 Web Site". Disponível em: <http://www.isi.edu/nsnam/ns/>. Acessado em: Abril de 2010.

TKN - Technical University Berlin Telecommunication Networks Group. Disponível em: http://www.tkn.tu-berlin.de/research/802.11e_ns2/. Acessado em: Agosto de 2010.

VISCAÍNO, Jorge López. “WiMAX: IEEE 802.16”. Tampere Polytechnic, Telecommunications Engineering. Junho, 2008.

WANG, Z., LU, L. BOVIK, A.C. “Video quality assessment based on structural distortion measurement, Signal Processing: Image Communication”. vol. 19, no. 2. 2004.

Wi-Fi Alliance. “Wi-Fi Alliance Web Site”. Disponível em: <http://www.wi-fi.org/>. Acessado em: Março de 2011.

WiMAX Forum. “WiMAX Forum Web Site”. Disponível em: <http://www.wimaxforum.org/home>. Acessado em: Janeiro de 2011.

WINKLER, S. “Perceptual video quality metrics – a review, in Digital Video Image Quality and Perceptual Coding”. Eds. H. R. Wu, K. R. Rao, cha 5, CRC Press. 2005.

XIAO, F. “DCT-based Video Quality Evaluation MSU Video Quality Metric”. Disponível em: http://compression.ru/video/quality_measure/vqm.pdf. Acessado em: Fevereiro de 2011.

YUV CIF reference videos. “YUV Web Site”. Disponível em: <http://www.tkn.tu-berlin.de/research/evalvid/cif.html>. Acessado em: Janeiro de 2011.

Anexos

Script do Cenário 1:

```

#Author: Warley Junior (warleyjunior@ufpa.br)

#-----#
# MIHS Network Simulation with MIH(802.21), WiMAX(802.16e),
# WiFi(802.11e).
# Network Topology
#       0.0.0           1.0.0
#           Server 0-----Router 1
#                   |
#                   2.0.0
#                   Router 2
#                   /       \
#                   BS 1     AP 1
#           5.0.0           6.0.0
#
#-----#

# Global Parameters

#check input parameters

set max_fragmented_size 1024

#add udp header(8 bytes) and IP header (20bytes)

set packetSize 1052

#check input parameters
if {$argc != 2} {
    puts ""
    puts "Wrong Number of Arguments! No arguments in this
topology"
    puts "ns fuzzylnova.tcl rtPS_scheduler_speed"
    puts ""
    exit (1)
}

global ns

#set debug attributes
Agent/ND set debug_1
Agent/MIH set debug_1
Agent/MIHUser/IFMNGMT/MIPV6 set debug_1
Mac/802_16 set debug_1
Mac/802_11 set debug_1

Mac/802_16 set dcd_interval_ 5 ;#max 10s
Mac/802_16 set ucd_interval_ 5 ;#max 10s
set default_modulation OFDM_16QAM_3_4 ;#OFDM_BPSK_1_2
set contention_size 5 ;#for initial ranging and bw
Mac/802_16 set t21_timeout_ 0.02 ;#max 10s, to replace the
timer for looking at preamble

```

```

Mac/802_16 set client_timeout_          50

#random used in movement of MN
set seed 6
set move [new RandomVariable/Uniform]
$move set min_ 2
$move set max_ 12
for {set j 0} {$j < $seed} {incr j} {
    set departure [$move value]
}

set f0 [open ugs.tr w]
set f1 [open rtps.tr w]
set f2 [open be.tr w]

#$defaultRNG seed [lindex $argv 0]
Mac/802_11 set pr_limit_ 1.1 ;#1.0 for link down only
Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2 set shutdown_on_ack_ 0

# set global variables
set output_dir .
set traffic_start 5
set traffic_stop 70
set simulation_stop 70

#create the simulator
set ns [new Simulator]
#$ns use-newtrace

#open file for trace
set f [open old-video.tr w]
$ns trace-all $f

$ns node-config -addressType hierarchical
$ns node-config -MPLS ON
AddrParams set domain_num_ 18 ;# domain number
AddrParams set cluster_num_ {1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1}
;# cluster number for each domain
AddrParams set nodes_num_ {1 1 1 1 1 1 1 1 1 5 5 1 1 1 1 1 1}
;# number of nodes for each cluster

set router0 [$ns node 0.0.0] ; $router0 set X_ 500.0; $router0 set Y_
500.0; $router0 set Z_ 0.0 ;
$ns add-to-mpls-list $router0
puts "router0: tcl=$router0; id=[$router0 id]; addr=[$router0 node-
addr]"

set router1 [$ns node 1.0.0]; $router1 set X_ 550.0; $router1 set Y_
500.0; $router1 set Z_ 0.0 ;
$ns add-to-mpls-list $router1
puts "router1: tcl=$router1; id=[$router1 id]; addr=[$router1 node-
addr]"

set router2 [$ns node 2.0.0]; $router2 set X_ 650.0; $router2 set Y_
500.0; $router2 set Z_ 0.0 ;
$ns add-to-mpls-list $router2
puts "router2: tcl=$router2; id=[$router2 id]; addr=[$router2 node-
addr]"

```

```

# connect links
$ns duplex-link $router0 $router1 10Mb 40ms CBQ
$ns duplex-link $router1 $router2 10Mb 40ms CBQ

# creation of the MutiFaceNodes. It MUST be done before the 802.11
$ns node-config -multiIf ON ;#to create
MultiFaceNode
set multiFaceNode [$ns node 7.0.0] ;# node id is
6
puts "multiFaceNode: tcl=$multiFaceNode; id=[$multiFaceNode id];
addr=[$multiFaceNode node-addr]"
set multiFaceNode1 [$ns node 8.0.0] ;# node id is
6
puts "multiFaceNode1: tcl=$multiFaceNode1; id=[$multiFaceNode1 id];
addr=[$multiFaceNode1 node-addr]"
set multiFaceNode2 [$ns node 9.0.0] ;# node id is
6
puts "multiFaceNode2: tcl=$multiFaceNode2; id=[$multiFaceNode2 id];
addr=[$multiFaceNode2 node-addr]"
$ns node-config -multiIf OFF

#
# Now we add 802.11 nodes
#

# parameter for wireless nodes
set opt(chan) Channel/WirelessChannel ;# channel type for
802.11
set opt(prop) Propagation/TwoRayGround ;# radio-
propagation model 802.11
set opt(netif) Phy/WirelessPhy ;# network
interface type 802.11
set opt(mac) Mac/802_11 ;# MAC type 802.11
set opt(ifq) Queue/DropTail/PriQueue ;# interface queue
type 802.11
set opt(ll) LL ;# link layer type
802.11
set opt(ant) Antenna/OmniAntenna ;# antenna model
802.11
set opt(ifqlen) 50 ;# max packet in ifq
802.11
set opt(adhocRouting) DSDV ;# routing protocol
802.11
set opt(umtsRouting) "" ;# routing for UMTS
(to reset node config)
set opt(x) 2000 ;# X dimension of the topography
set opt(y) 2000 ;# Y dimension of the topography

set namtrace [open fuzzyl.nam w]
$ns namtrace-all-wireless $namtrace $opt(x) $opt(y)

# configure rate for 802.11
Mac/802_11 set basicRate_ 54Mb
Mac/802_11 set dataRate_ 54Mb
Mac/802_11 set bandwidth_ 54Mb
#Mac/802_11 set RTSThreshold_ 3000

#create the topography
set topo [new Topography]

```

```

$stopo load_flatgrid $opt(x) $opt(y)
#puts "Topology created"
set chan [new $opt(chan)]

# create God
create-god 20 ;# give the number
of nodes

#configure for 20m radius 2.4Ghz
#Phy/WirelessPhy set Pt_ 0.025
#Phy/WirelessPhy set freq_ 2412e+6
#Phy/WirelessPhy set RXThresh_ 6.12277e-09
#Phy/WirelessPhy set CStresh_ [expr 0.9*[Phy/WirelessPhy set
RXThresh_]]

##configure for 50m radius 2.4Ghz
Phy/WirelessPhy set Pt_ 0.0134
Phy/WirelessPhy set freq_ 2412e+6
Phy/WirelessPhy set RXThresh_ 5.25089e-10
Phy/WirelessPhy set CStresh_ [expr 0.9*[Phy/WirelessPhy set
RXThresh_]]

# configure Access Points
$ns node-config -adhocRouting $opt(adhocRouting) \
    -llType $opt(ll) \
    -macType $opt(mac) \
    -channel $chan \
    -ifqType $opt(ifq) \
    -ifqLen $opt(ifqlen) \
    -antType $opt(ant) \
    -propType $opt(prop) \
    -phyType $opt(netif) \
    -topoInstance $stopo \
    -wiredRouting ON \
    -agentTrace ON \
    -routerTrace OFF \
    -macTrace ON \
    -movementTrace OFF

# configure Base station 802.11
set bstation802 [$ns node 6.0.0] ;
$bstation802 set X_ 550.0
$bstation802 set Y_ 350.0
$bstation802 set Z_ 0.0
puts "bstation802: tcl=$bstation802; id=[$bstation802 id];
addr=[$bstation802 node-addr]"
# we need to set the BSS for the base station
set bstationMac [$bstation802 getMac 0]
set AP_ADDR_0 [$bstationMac id]
puts "bss_id for bstation 1=$AP_ADDR_0"
$bstationMac bss_id $AP_ADDR_0
$bstationMac enable-beacon
$ns add-to-mpls-list $bstation802

# creation of the wireless interface 802.11
$ns node-config -wiredRouting OFF \
    -macTrace ON
set ifacel [$ns node 6.0.1] ;#
node id is 8.
$ifacel random-motion 0 ;#
disable random motion

```



```

$iface1 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface1 set X_ 455.0 ;#450.0
$iface1 set Y_ 333.0 ;#330.0
$iface1 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface1 setdest 580.1 333.0 [lindex $argv 1]";#640.0 330.0
#$ns at 5.0 "$iface1 setdest 450.1 330.0 5.0"
puts "iface1: tcl=$iface1; id=[$iface1 id]; addr=[$iface1 node-addr]"

set iface3 [$ns node 6.0.2] ;#
node id is 8.
$iface3 random-motion 0 ;#
disable random motion
$iface3 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface3 set X_ 454.0 ;#450.0
$iface3 set Y_ 332.0 ;#320.0
$iface3 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface3 setdest 590.1 333.0 [lindex $argv 1]";#640.0 320.0
#$ns at 5.0 "$iface3 setdest 450.1 320.0 5.0"
puts "iface3: tcl=$iface3; id=[$iface3 id]; addr=[$iface3 node-addr]"

set iface5 [$ns node 6.0.3] ;#
node id is 8.
$iface5 random-motion 0 ;#
disable random motion
$iface5 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface5 set X_ 460.0 ;#470.0
$iface5 set Y_ 345.0 ;#330.0
$iface5 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface5 setdest 580.1 345.0 [lindex $argv 1]";#660.0 330.0
#$ns at 5.0 "$iface5 setdest 470.1 330.0 5.0"
puts "iface5: tcl=$iface5; id=[$iface5 id]; addr=[$iface5 node-addr]"

# add link to backbone
$ns duplex-link $bstation802 $router2 10Mb 40ms CBQ

# add Wimax nodes
set opt(netif) Phy/WirelessPhy/OFDM ;# network
interface type 802.16
set opt(mac) Mac/802_16 ;# MAC type 802.16

# radius =
Phy/WirelessPhy set Pt_ 0.025
Phy/WirelessPhy set RXThresh_ 7.91016e-15 ;#500m:2.025e-12
#Phy/WirelessPhy set RXThresh_ 1.26562e-13 ;#1000m radius
Phy/WirelessPhy set CStresh_ [expr 0.8*[Phy/WirelessPhy set
RXThresh_]]

# configure Access Points

```

```

$ns node-config -adhocRouting $opt(adhocRouting) \
    -llType $opt(ll) \
    -macType $opt(mac) \
    -channel $chan \
    -ifqType $opt(ifq) \
    -ifqLen $opt(ifqlen) \
    -antType $opt(ant) \
    -propType $opt(prop) \
    -phyType $opt(netif) \
    -topoInstance $topo \
    -wiredRouting ON \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
    -movementTrace OFF

# configure Base station 802.16
set bstation802_16 [$ns node 5.0.0] ;
$bstation802_16 set X_ 450.0
$bstation802_16 set Y_ 350.0
$bstation802_16 set Z_ 0.0
$ns add-to-mpls-list $bstation802_16
puts "bstation802_16: tcl=$bstation802_16; id=[$bstation802_16 id];
addr=[$bstation802_16 node-addr]"
set clas [new SDUClassifier/Dest]
[$bstation802_16 set mac_(0)] add-classifier $clas
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set bs_sched [new WimaxScheduler/BS]
$bs_sched set-default-modulation $default_modulation
[$bstation802_16 set mac_(0)] set-scheduler $bs_sched
[$bstation802_16 set mac_(0)] set-channel 1

# creation of the wireless interface 802.16
$ns node-config -MPLS OFF
$ns node-config -wiredRouting OFF \
    -macTrace ON

set iface2 [$ns node 5.0.1] ;#
node id is 8.
$iface2 random-motion 0 ;#
disable random motion
$iface2 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface2 set X_ 455.0;#450.0
$iface2 set Y_ 333.0;#330.0
$iface2 set Z_ 0.0
set clas2 [new SDUClassifier/Dest]
[$iface2 set mac_(0)] add-classifier $clas2
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched [new WimaxScheduler/SS]
[$iface2 set mac_(0)] set-scheduler $ss_sched
[$iface2 set mac_(0)] set-channel 1
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface2 setdest 580.1 333.0 [lindex $argv 1];#640.0 330.0
#$ns at 5.0 "$iface2 setdest 450.1 330.0 5.0"
puts "iface2: tcl=$iface2; id=[$iface2 id]; addr=[$iface2 node-addr]"

```

```

set iface4 [$ns node 5.0.2] ;#
node id is 8.
$iface4 random-motion 0 ;#
disable random motion
$iface4 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface4 set X_ 454.0;#450.0
$iface4 set Y_ 332.0;#320.0
$iface4 set Z_ 0.0
set clas4 [new SDUClassifier/Dest]
[$iface4 set mac_(0)] add-classifier $clas4
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched4 [new WimaxScheduler/SS]
[$iface4 set mac_(0)] set-scheduler $ss_sched4
[$iface4 set mac_(0)] set-channel 1
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface4 setdest 590.1 333.0 [lindex $argv 1]";#640.0 320.0
#$ns at 5.0 "$iface4 setdest 450.1 320.0 5.0"
puts "iface4: tcl=$iface4; id=[$iface4 id]; addr=[$iface4 node-addr]"

set iface6 [$ns node 5.0.3] ;#
node id is 8.
$iface6 random-motion 0 ;#
disable random motion
$iface6 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface6 set X_ 460.0;#470.0
$iface6 set Y_ 345.0;#330.0
$iface6 set Z_ 0.0
set clas6 [new SDUClassifier/Dest]
[$iface6 set mac_(0)] add-classifier $clas6
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched6 [new WimaxScheduler/SS]
[$iface6 set mac_(0)] set-scheduler $ss_sched6
[$iface6 set mac_(0)] set-channel 1
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface6 setdest 580.1 345.0 [lindex $argv 1]";#660.0 330.0
#$ns at 5.0 "$iface6 setdest 470.1 330.0 5.0"
puts "iface6: tcl=$iface6; id=[$iface6 id]; addr=[$iface6 node-addr]"

$ns duplex-link $bstation802_16 $router2 10Mb 40ms CBQ

$ns duplex-link-op $router0 $router1 orient right
$ns duplex-link-op $router1 $router2 orient down
$ns duplex-link-op $router2 $bstation802_16 orient left-down
$ns duplex-link-op $router2 $bstation802 orient right-down

# add interfaces to MultiFaceNode
$multiFaceNode add-interface-node $iface1
$multiFaceNode add-interface-node $iface2

$multiFaceNode1 add-interface-node $iface3
$multiFaceNode1 add-interface-node $iface4

```

```

$multiFaceNode2 add-interface-node $iface5
$multiFaceNode2 add-interface-node $iface6

# install ND modules

# now WLAN
set nd_bs [$bstation802 install-nd]
$nd_bs set-router TRUE
$nd_bs router-lifetime 1800

set nd_mn [$iface1 install-nd]
set nd_mn3 [$iface3 install-nd]
set nd_mn5 [$iface5 install-nd]

# now WIMAX
set nd_bs2 [$bstation802_16 install-nd]
$nd_bs2 set-router TRUE
$nd_bs2 router-lifetime 20 ;#just enough to expire while we are
connected to wlan.

set nd_mn2 [$iface2 install-nd]
set nd_mn4 [$iface4 install-nd]
set nd_mn6 [$iface6 install-nd]

# install interface manager into multi-interface node and CN
Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2 set debug_ 1

set handover [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode install-ifmanager $handover
$nd_mn set-ifmanager $handover
$handover nd_mac $nd_mn [$iface1 set mac_(0)] ;#to know how to send RS
$nd_mn2 set-ifmanager $handover
$handover nd_mac $nd_mn2 [$iface2 set mac_(0)] ;#to know how to send
RS

set handover1 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode1 install-ifmanager $handover1
$nd_mn3 set-ifmanager $handover1
$handover1 nd_mac $nd_mn3 [$iface3 set mac_(0)] ;#to know how to send
RS
$nd_mn4 set-ifmanager $handover1
$handover1 nd_mac $nd_mn4 [$iface4 set mac_(0)] ;#to know how to send
RS

set handover2 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode2 install-ifmanager $handover2
$nd_mn5 set-ifmanager $handover2
$handover2 nd_mac $nd_mn5 [$iface5 set mac_(0)] ;#to know how to send
RS
$nd_mn6 set-ifmanager $handover2
$handover2 nd_mac $nd_mn6 [$iface6 set mac_(0)] ;#to know how to send
RS

set ifmgmt_cn [$router0 install-default-ifmanager]

# install MIH in multi-interface node
set mih [$multiFaceNode install-mih]
set mih1 [$multiFaceNode1 install-mih]

```

```

set mih2 [$multiFaceNode2 install-mih]

$handover connect-mih $mih ;#create connection between MIH and iface
management
$handover1 connect-mih $mih1 ;#create connection between MIH and iface
management
$handover2 connect-mih $mih2 ;#create connection between MIH and iface
management

# install MIH on AP/BS
set mih_bs [$bstation802 install-mih]
set tmp_bs [$bstation802 set mac_(0)]
$tmp_bs mih $mih_bs
$mih_bs add-mac $tmp_bs

set mih_bs [$bstation802_16 install-mih]
set tmp_bs [$bstation802_16 set mac_(0)]
$tmp_bs mih $mih_bs
$mih_bs add-mac $tmp_bs

# Now we can register the MIH module with all the MACs
set tmp20 [$iface1 set mac_(0)] ;#in 802.11 one interface is created
$tmp20 mih $mih
$mih add-mac $tmp20 ;#inform the MIH about the local MAC
set tmp20 [$iface2 set mac_(0)] ;#in 802.16 one interface is created
$tmp20 mih $mih
$mih add-mac $tmp20 ;#inform the MIH about the local MAC

set tmp21 [$iface3 set mac_(0)] ;#in 802.11 one interface is created
$tmp21 mih $mih1
$mih1 add-mac $tmp21 ;#inform the MIH about the local MAC
set tmp21 [$iface4 set mac_(0)] ;#in 802.16 one interface is created
$tmp21 mih $mih1
$mih1 add-mac $tmp21 ;#inform the MIH about the local MAC

set tmp22 [$iface5 set mac_(0)] ;#in 802.11 one interface is created
$tmp22 mih $mih2
$mih2 add-mac $tmp22 ;#inform the MIH about the local MAC
set tmp22 [$iface6 set mac_(0)] ;#in 802.16 one interface is created
$tmp22 mih $mih2
$mih2 add-mac $tmp22 ;#inform the MIH about the local MAC

# set the starting time for Router Advertisements
$ns at 2 "$nd_bs start-ra"
$ns at 2 "$nd_bs2 start-ra"

#####MPLS
#MPLS Settings
# RSVPTE Settings
$ns PATH-color "purple"
$ns PATHERR-color "red"
$ns PATHTEAR-color "red"
$ns RESV-color "grey"
$ns RESVERR-color "red"
$ns RESVCONF-color "red"
$ns color 1002 green

```

```

Agent/RSVP set noisy_ 255
Agent/RSVP set refresh_ 30
Agent/RSVP set lifetime_factor_ 3
Agent/RSVP set ip6_ 0
Agent/RSVP set nam_ 1

# One RSVP-TE Agent on each MPLS Node
$ns configure-rsvpte-on-all-mpls-nodes

$ns cfg-cbq-for-SBTS 2040 DropTail 990000.000000 0.010000 auto
0.000000
$ns cfg-cbq-for-HBTS 10 DropTail 0.000000 0.000000 auto 0.000000
$ns cfg-cbq-for-STS 10 DropTail 0.000000 0.000000 auto 0.000000
$ns cfg-cbq-for-RTS 2040 DropTail 0.000000 0.990000 auto 0.000000
$ns bind-rsvpte-to-SBTS

$ns enable-control-driven

# Mobile IP reg req message and other things
$ns bind-flowid-to-SBTS 0
$ns bind-flowid-to-SBTS 1
# MPLS test connection

# MPLS test tunnel
$ns compute-routes
#####MPLS

#####
#set the bumber of UGS connections
set nb_UGS 1

#### interval_ of the CBR traffic ####
set interval_ugs(1) 0.15
set interval_ugs(2) 0.2
set interval_ugs(3) 0.25
set interval_ugs(4) 0.27
set interval_ugs(5) 0.3

set interval_ugs(6) 0.04
set interval_ugs(7) 0.05
set interval_ugs(8) 0.1
set interval_ugs(9) 0.1
#####

#### SNR of the UGS connections#####
set SNR_ugs(1) 9.5
set SNR_ugs(2) 12.5
set SNR_ugs(3) 16.5
set SNR_ugs(4) 20.5
set SNR_ugs(5) 22.5

set SNR_ugs(6) 12.5
set SNR_ugs(7) 12.5
set SNR_ugs(8) 12.5
set SNR_ugs(9) 12.5
#####

for {set j 1} {$j < [expr $nb_UGS + 1]} {incr j} {

```

```

#Create a UDP agent and attach it to wl_node$j
set udp_ugs($j) [new Agent/UDP]
#$udp_ugs($j) set packetSize_ 1000
$ns attach-agent $router0 $udp_ugs($j)

# Create a CBR traffic source and attach it to udp_ugs
set cbr_ugs($j) [new Application/Traffic/CBR]
$cbr_ugs($j) set packetSize_ 202 ;#600x8/0.005=960k
$cbr_ugs($j) set interval_ 0.005
$cbr_ugs($j) attach-agent $udp_ugs($j)
$cbr_ugs($j) set rate_ 3.0Mb

# Create the Null agent to sink traffic
set sink_ugs [new Agent/LossMonitor]
$ns attach-agent $multiFaceNode $sink_ugs

# Attach the 2 agents
$ns connect $udp_ugs($j) $sink_ugs

$udp_ugs($j) set prio_ 2
$udp_ugs($j) set fid_ 2
$ns bind-flowid-to-SBTS 2

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0=>SERVICE_UGS, 1=>SERVICE_rtPS,
2=>SERVICE_nrtPS, 3=>SERVICE_BE
$ss_sched add-flow 5 [expr 30 + [$cbr_ugs($j) set packetSize_] *
[Mac/802_16 set frame_duration_] / [$cbr_ugs($j) set interval_]] 0 1

##set-PeerNode-SNR PeerNode SNR
$ns at 1.5 "$bs_sched set-PeerNode-SNR [expr $j] $SNR_ugs($j)"

##set-PeerNode-UGSPeriodicity PeerNode Periodicity (periodicity of the
reservation, every k frames)
$ns at 1.5 "$bs_sched set-PeerNode-UGSPeriodicity [expr $j] 1"

$handover add-flow $sink_ugs $udp_ugs($j) $iface2 1

#Schedule start/stop of traffic
$ns at 2.0 "$cbr_ugs($j) start"

}
#####

#####RTPSSSSSSSSS

# rtPS connections
# The identity of the first rtPS connection is k, the second is k+1,
and so on
set first_rtPS 101

#set the bumber of rtPS connections
set nb_rtPS 1

```

```
#set the number of symbols reserved for unicast request opportunities
$bs_sched set-SymbolNumberForUnicastRequest 3
```

```
set rtPS_scheduler_ [lindex $argv 0]
```

```
#bs_sched set-rtPSscheduling scheduling
$bs_sched set-rtPSscheduling $rtPS_scheduler_
```

```
proc send_next_packet_VBR {udp_size_interval_} {
    global ns traffic_stop
    $udp_send [expr round([$size_value])]
    # $udp_send 1000 # constant if CBR
```

```
if {[ $ns now] < [expr $traffic_stop - $interval_]} {
    $ns at [expr [ $ns now] + $interval_] "send_next_packet_VBR $udp_
    $size_ $interval_"
}
```

```
}
```

```
# seed the default RNG
global defaultRNG
$defaultRNG seed 9999
```

```
#### interval_ #####
set interval_rtPS(1) 0.01
set interval_rtPS(2) 0.04
set interval_rtPS(3) 0.05
set interval_rtPS(4) 0.02
set interval_rtPS(5) 0.05
set interval_rtPS(6) 0.04
set interval_rtPS(7) 0.03
set interval_rtPS(8) 0.02
set interval_rtPS(9) 0.03
#####
```

```
#### SNR #####
set SNR_rtPS(1) 7.0
set SNR_rtPS(2) 7.5
set SNR_rtPS(3) 9.0
set SNR_rtPS(4) 12.0
set SNR_rtPS(5) 17.0
set SNR_rtPS(6) 17.5
set SNR_rtPS(7) 20.0
set SNR_rtPS(8) 24.0
set SNR_rtPS(9) 25.5
#####
```

```
#### WRR #####
# set the weights if using WRR
if { $rtPS_scheduler_ == "WRR" } {
    set WRR_rtPS(1) 1
    set WRR_rtPS(2) 1
}
```



```

set WRR_rtPS(3) 1
set WRR_rtPS(4) 2
set WRR_rtPS(5) 2
set WRR_rtPS(6) 3
set WRR_rtPS(7) 3

set WRR_rtPS(8) 4
set WRR_rtPS(9) 4
}
#####

#####
#set-TRSparameters-SNR-Tr-Tp-L SNRth Tr Tp L
$bs_sched set-TRSparameters-SNR-Tr-Tp-L 8.5 2 3 4
#####

$ns node-config -wiredRouting OFF \
                -macTrace ON                               ;# Mobile nodes
cannot do routing.

for {set j $first_rtPS} {$j < [expr $first_rtPS + $nb_rtPS]} {incr j}
{

#Create a UDP agent and attach it to wl_node$j
set udp_rtPS($j) [new Agent/UDP]
$ns attach-agent $router0 $udp_rtPS($j)

#Traffic
#create video traffic
set cbr_rtps [new Application/Traffic/CBR]
$cbr_rtps set packetSize_ 201 ;#600x8/0.005=960k
$cbr_rtps set interval_ 0.005
$cbr_rtps attach-agent $udp_rtPS($j)
$cbr_rtps set rate_ 3.0Mb

# Create the Null agent to sink traffic
set sink_rtps [new Agent/LossMonitor]
$ns attach-agent $multiFaceNode1 $sink_rtps

# Attach the 2 agents
$ns connect $udp_rtPS($j) $sink_rtps

$udp_rtPS($j) set prio_ 1
$udp_rtPS($j) set fid_ 1
$ns bind-flowid-to-SBTS 1

set interval_rtPS($j) $interval_rtPS([expr $j + 1 - $first_rtPS])

## exponential distribution
#set sizeRNG_rtPS($j) [new RNG]

#set size_rtPS($j) [new RandomVariable/Exponential]
#$size_rtPS($j) set avg_ 1000
#$size_rtPS($j) use-rng $sizeRNG_rtPS($j)

```

```

# uniform distribution
set sizeRNG_rtPS($j) [new RNG]

set size_rtPS($j) [new RandomVariable/Uniform]
$size_rtPS($j) set min_ 500
$size_rtPS($j) set max_ 1500
$size_rtPS($j) use-rng $sizeRNG_rtPS($j)

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0=>SERVICE_UGS, 1=>SERVICE_rtPS,
2=>SERVICE_nrtPS, 3=>SERVICE_BE
$ss_sched4 add-flow 5 0 0 1

##set-PeerNode-SNR PeerNode SNR
$ns at 1.5 "$bs_sched set-PeerNode-SNR [expr $nb_UGS + $j + 1 -
$first_rtPS] $SNR_rtPS([expr $j + 1 - $first_rtPS])"

##set-PeerNode-UnicastRequestPeriodicity PeerNode Periodicity
$ns at 1.5 "$bs_sched set-PeerNode-UnicastRequestPeriodicity [expr
$nb_UGS + $j + 1 - $first_rtPS] 2"

if {$rtPS_scheduler_ == "WRR"} {
  # set-PeerNode-WRRschedulingForrtPS PeerNode Weight
  $ns at 1.5 "$bs_sched set-PeerNode-WRRschedulingForrtPS [expr
$nb_UGS + $j + 1 - $first_rtPS] $WRR_rtPS([expr $j + 1 -
$first_rtPS])"
}

$handover1 add-flow $sink_rtps $udp_rtPS($j) $iface4 1

#$ns at [expr 5.0 + [expr $j + 1 - $first_rtPS] * 0]
"send_next_packet_VBR $udp_rtPS($j) $size_rtPS($j) $interval_rtPS($j)"

$ns at 2.0 "$cbr_rtps start"
}

#####BEEEEEEEEEEEEEEEEEEEE#####
#####
#set first_BE 301

##set-PeerNode-SNR PeerNode SNR
$ns at 1.3 "$bs_sched set-PeerNode-SNR [expr $nb_UGS + $nb_rtPS + 1]
12.3"

##set-BwRequestSendingPeriod BwRequestSendingPeriod_
$ss_sched6 set-BwRequestSendingPeriod 10

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0 => SERVICE_UGS, 1 => SERVICE_rtPS, 2 =>
SERVICE_nrtPS, 3 => SERVICE_BE

$ss_sched6 add-flow 1 0 0 1

#####

                set udp_be [new Agent/UDP]
                #$udp_($i) set packetSize_ 1000

```

```

$udp_be set prio_ 3
$udp_be set fid_ 3
$ns bind-flowid-to-SBTS 3

#create video traffic
set cbr_be [new Application/Traffic/CBR]
$cbr_be set packetSize_ 203 ;#600x8/0.005=960k
$cbr_be set interval_ 0.005
$cbr_be attach-agent $udp_be
$cbr_be set rate_ 3.0Mb

set sink_be [new Agent/LossMonitor]

#sinkNode is transmitter
$ns attach-agent $router0 $udp_be
$ns attach-agent $multiFaceNode2 $sink_be

```

```
#####
```

```

set udp_be1 [new Agent/UDP]
#$udp_($i) set packetSize_ 1000
$udp_be1 set prio_ 3
$udp_be1 set fid_ 4
$ns bind-flowid-to-SBTS 4

#create video traffic
set cbr_be1 [new Application/Traffic/CBR]
$cbr_be1 set packetSize_ 204 ;#600x8/0.005=960k
$cbr_be1 set interval_ 0.005
$cbr_be1 attach-agent $udp_be1
$cbr_be1 set rate_ 3.0Mb

set sink_be1 [new Agent/LossMonitor]

#sinkNode is transmitter
$ns attach-agent $router0 $udp_be1
$ns attach-agent $multiFaceNode2 $sink_be1

```

```
#####
```

```

set udp_be2 [new Agent/UDP]
#$udp_($i) set packetSize_ 1000
$udp_be2 set prio_ 3
$udp_be2 set fid_ 5
$ns bind-flowid-to-SBTS 5

#create video traffic
set cbr_be2 [new Application/Traffic/CBR]
$cbr_be2 set packetSize_ 205 ;#600x8/0.005=960k
$cbr_be2 set interval_ 0.005
$cbr_be2 attach-agent $udp_be2
$cbr_be2 set rate_ 3.0Mb

set sink_be2 [new Agent/LossMonitor]

#sinkNode is transmitter
$ns attach-agent $router0 $udp_be2
$ns attach-agent $multiFaceNode2 $sink_be2

```

```
#####
```

```

$handover2 add-flow $sink_be $udp_be $iface6 1
#$handover2 add-flow $sink_bel $udp_bel $iface6 1
#$handover2 add-flow $sink_be2 $udp_be2 $iface6 1

$ns at 2.0 "$cbr_be start"
#$ns at 2.0 "$cbr_bel start"
#$ns at 2.0 "$cbr_be2 start"

#####BEEEEEEEEEEEEEEEEEEEE#####
#####

#defines function for flushing and closing files
proc finish {} {
    global ns f f0 f1 f2 namtrace
    $ns flush-trace
    close $f
    close $f0
    close $f1
    close $f2
    close $namtrace
    puts " Simulation ended."
    exec xgraph ugs.tr rtps.tr be.tr -geometry 800x400 &
    exit 0
}

proc record {} {
    global sink_ugs sink_rtps sink_be f0 f1 f2 handover handover1
    handover2
    #Get an instance of the simulator
    set ns [Simulator instance]
    #Set the time after which the procedure should be called again
    set time 0.5
    #How many bytes have been received by the traffic sinks?
    set bw0 [$sink_ugs set bytes_]
        set bw1 [$sink_rtps set bytes_]
        set bw2 [$sink_be set bytes_]

    #Get the current time
    set now [$ns now]

    #Calculate the bandwidth (in MBit/s) and write it to the
files
    puts $f0 "$now [expr $bw0/$time*8/1000000]"
    puts $f1 "$now [expr $bw1/$time*8/1000000]"
    puts $f2 "$now [expr $bw2/$time*8/1000000]"

    #Handover UGS [expr $bw0/$time*8/1000000]
    #Handover1 RTPS [expr $bw1/$time*8/1000000]
    #Handover2 BE [expr $bw2/$time*8/1000000]

    #Reset the bytes_ values on the traffic sinks
    $sink_ugs set bytes_ 0
    $sink_rtps set bytes_ 0
    $sink_be set bytes_ 0

    #Re-schedule the procedure
    $ns at [expr $now+$time] "record"
}
#####

$ns at 2.0 "record"

```

```
$ns at 50.0 "finish"
```

```
$ns run
```

```
#####
```

Script do Cenário 2:

```
#Author: Warley Junior (warleyjunior@ufpa.br)
```

```
#-----#
# MIHS Network Simulation with MIH(802.21), WiMAX(802.16e),
# WiFi(802.11e).
# Network Topology
#      0.0.0          1.0.0
#      Server 0-----Router 1
#              |
#              2.0.0
#              Router 2
#              /       \
#          BS 1       AP 1
#          5.0.0       6.0.0
#
#-----#
```

```
# Global Parameters
```

```
#check input parameters
```

```
set max_fragmented_size 1024
```

```
#add udp header(8 bytes) and IP header (20bytes)
```

```
set packetSize 1052
```

```
#global ns
```

```
#check input parameters
```

```
if {$argc != 2} {
```

```
    puts ""
```

```
    puts "Wrong Number of Arguments! No arguments in this
```

```
topology"
```

```
    puts "ns fuzzylnovaTempl.tcl rtPS_scheduler_speed"
```

```
    puts ""
```

```
    exit (1)
```

```
}
```

```
#set debug attributes
```

```
Agent/ND set debug_1
```

```
Agent/MIH set debug_1
```

```
Agent/MIHUser/IFMNGMT/MIPV6 set debug_1
```

```
Mac/802_16 set debug_1
```

```
Mac/802_11 set debug_1
```

```
Mac/802_16 set dcd_interval_          5 ;#max 10s
```

```
Mac/802_16 set ucd_interval_          5 ;#max 10s
```

```
set default_modulation                OFDM_16QAM_3_4 ;#OFDM_BPSK_1_2
```

```
set contention_size                    5 ;#for initial ranging and bw
```

```
Mac/802_16 set t21_timeout_           0.02 ;#max 10s, to replace the
```

```
timer for looking at preamble
```

```
Mac/802_16 set client_timeout_        50
```

```

Mac/802_16 set frame_duration_ 0.020

#random used in movement of MN
#set seed 6
#set move [new RandomVariable/Uniform]
#$move set min_ 2
#$move set max_ 12
#for {set j 0} {$j < $seed} {incr j} {
#   set departure [$move value]
#}

#*** Throughput Trace ***
set f0 [open ugs.tr w]
set f1 [open rtps.tr w]
set f2 [open be.tr w]
set f3 [open ugs1.tr w]
set f4 [open rtps1.tr w]
set f5 [open be1.tr w]
set f6 [open ugs2.tr w]
set f7 [open rtps2.tr w]
set f8 [open be2.tr w]

# *** Packet Loss Trace ***
set f9 [open lostugs.tr w]
set f10 [open lostrtps.tr w]
set f11 [open lostbe.tr w]
set f12 [open lostugs1.tr w]
set f13 [open lostrtps1.tr w]
set f14 [open lostbe1.tr w]
set f15 [open lostugs2.tr w]
set f16 [open lostrtps2.tr w]
set f17 [open lostbe2.tr w]

# *** Packet Delay Trace ***
set f18 [open delayugs.tr w]
set f19 [open delayrtps.tr w]
set f20 [open delaybe.tr w]
set f21 [open delayugs1.tr w]
set f22 [open delayrtps1.tr w]
set f23 [open delaybe1.tr w]
set f24 [open delayugs2.tr w]
set f25 [open delayrtps2.tr w]
set f26 [open delaybe2.tr w]

#$defaultRNG seed [lindex $argv 0]
Mac/802_11 set pr_limit_ 1.1 ;#1.0 for link down only
Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2 set shutdown_on_ack_ 0

# set global variables
set output_dir .
set traffic_start 2
set traffic_stop 100
set simulation_stop 100

#create the simulator
set ns [new Simulator]
$ns use-newtrace

#open file for trace
set f [open old-video.tr w]

```

```

$ns trace-all $f

$ns node-config -addressType hierarchical
$ns node-config -MPLS ON
AddrParams set domain_num_ 15 ;# domain number
AddrParams set cluster_num_ {1 1 1 1 1 1 1 1 1 1 1 1 1 1}
;# cluster number for each domain
AddrParams set nodes_num_ {1 1 1 10 1 1 1 1 1 1 1 1 1 1}
;# number of nodes for each cluster
puts "Configuration of hierarchical addressing done"

set router0 [$ns node 0.0.0] ; $router0 set X_ 500.0; $router0 set Y_
500.0; $router0 set Z_ 0.0 ;
$ns add-to-mpls-list $router0
puts "router0: tcl=$router0; id=[$router0 id]; addr=[$router0 node-
addr]"

set router1 [$ns node 1.0.0]; $router1 set X_ 550.0; $router1 set Y_
500.0; $router1 set Z_ 0.0 ;
$ns add-to-mpls-list $router1
puts "router1: tcl=$router1; id=[$router1 id]; addr=[$router1 node-
addr]"

set router2 [$ns node 2.0.0]; $router2 set X_ 650.0; $router2 set Y_
500.0; $router2 set Z_ 0.0 ;
$ns add-to-mpls-list $router2
puts "router2: tcl=$router2; id=[$router2 id]; addr=[$router2 node-
addr]"

# connect links
$ns duplex-link $router0 $router1 100Mb 40ms CBQ
$ns duplex-link $router1 $router2 100Mb 40ms CBQ

# creation of the MultiFaceNodes. It MUST be done before the 802.11
$ns node-config -multiIf ON ;#to create
MultiFaceNode
set multiFaceNode [$ns node 7.0.0] ;# node id is
6
puts "multiFaceNode: tcl=$multiFaceNode; id=[$multiFaceNode id];
addr=[$multiFaceNode node-addr]"
set multiFaceNode1 [$ns node 8.0.0] ;# node id is
6
puts "multiFaceNode1: tcl=$multiFaceNode1; id=[$multiFaceNode1 id];
addr=[$multiFaceNode1 node-addr]"
set multiFaceNode2 [$ns node 9.0.0] ;# node id is
6
puts "multiFaceNode2: tcl=$multiFaceNode2; id=[$multiFaceNode2 id];
addr=[$multiFaceNode2 node-addr]"
set multiFaceNode3 [$ns node 10.0.0] ;# node id
is 6
puts "multiFaceNode3: tcl=$multiFaceNode3; id=[$multiFaceNode3 id];
addr=[$multiFaceNode3 node-addr]"
set multiFaceNode4 [$ns node 11.0.0] ;# node id
is 6
puts "multiFaceNode4: tcl=$multiFaceNode4; id=[$multiFaceNode4 id];
addr=[$multiFaceNode4 node-addr]"
set multiFaceNode5 [$ns node 12.0.0] ;# node id
is 6

```



```

#configure for 20m radius 2.4Ghz
#Phy/WirelessPhy set Pt_ 0.025
#Phy/WirelessPhy set freq_ 2412e+6
#Phy/WirelessPhy set RXThresh_ 6.12277e-09
#Phy/WirelessPhy set CStresh_ [expr 0.9*[Phy/WirelessPhy set
RXThresh_]]

##configure for 50m radius 2.4Ghz
Phy/WirelessPhy set Pt_ 0.0134
Phy/WirelessPhy set freq_ 2412e+6
Phy/WirelessPhy set RXThresh_ 5.25089e-10
Phy/WirelessPhy set CStresh_ [expr 0.9*[Phy/WirelessPhy set
RXThresh_]]

# configure Access Points
$ns node-config -adhocRouting $opt(adhocRouting) \
    -llType $opt(ll) \
    -macType $opt(mac) \
    -channel $chan \
    -ifqType $opt(ifq) \
    -ifqLen $opt(ifqlen) \
    -antType $opt(ant) \
    -propType $opt(prop) \
    -phyType $opt(netif) \
    -topoInstance $topo \
    -wiredRouting ON \
    -agentTrace ON \
    -routerTrace OFF \
    -macTrace ON \
    -movementTrace OFF

# configure Base station 802.11
set bstation802 [$ns node 6.0.0] ;
$bstation802 set X_ 550.0
$bstation802 set Y_ 350.0
$bstation802 set Z_ 0.0
puts "bstation802: tcl=$bstation802; id=[$bstation802 id];
addr=[$bstation802 node-addr]"
# we need to set the BSS for the base station
set bstationMac [$bstation802 getMac 0]
set AP_ADDR_0 [$bstationMac id]
puts "bss_id for bstation 1=$AP_ADDR_0"
$bstationMac bss_id $AP_ADDR_0
$bstationMac enable-beacon
$ns add-to-mpls-list $bstation802

# creation of the wireless interface 802.11
$ns node-config -wiredRouting OFF \
    -macTrace ON

# UGS
set ifacel [$ns node 6.0.1] ;#
node id is 8. ;#
$ifacel random-motion 0 ;#
disable random motion
$ifacel base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$ifacel set X_ 450.0 ;#460.0
$ifacel set Y_ 330.0 ;#333.0
$ifacel set Z_ 0.0

```

```

# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 15.0 "$iface1 setdest 550.0 330.0 [lindex $argv 1]"
#$ns at 5.0 "$iface1 setdest 450.1 330.0 5.0"
puts "iface1: tcl=$iface1; id=[$iface1 id]; addr=[$iface1 node-addr]"

# UGS1
set iface3 [$ns node 6.0.2] ;#
node id is 8.
$iface3 random-motion 0 ;#
disable random motion
$iface3 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface3 set X_ 450.0 ;#445.0
$iface3 set Y_ 320.0 ;#322.0
$iface3 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 15.0 "$iface3 setdest 550.0 320.0 [lindex $argv 1]"
#$ns at 5.0 "$iface3 setdest 450.1 320.0 5.0"
puts "iface3: tcl=$iface3; id=[$iface3 id]; addr=[$iface3 node-addr]"

# UGS2
set iface5 [$ns node 6.0.3] ;#
node id is 8.
$iface5 random-motion 0 ;#
disable random motion
$iface5 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface5 set X_ 450.0 ;#445.0
$iface5 set Y_ 310.0 ;#322.0
$iface5 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 15.0 "$iface5 setdest 550.0 310.0 [lindex $argv 1]"
#$ns at 5.0 "$iface5 setdest 450.1 310.0 5.0"
puts "iface5: tcl=$iface5; id=[$iface5 id]; addr=[$iface5 node-addr]"

# rtPS
set iface7 [$ns node 6.0.4] ;#
node id is 8.
$iface7 random-motion 0 ;#
disable random motion
$iface7 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface7 set X_ 470.0 ;#470.0
$iface7 set Y_ 330.0 ;#330.0
$iface7 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 14.0 "$iface7 setdest 570.0 330.0 [lindex $argv 1]"
#$ns at 5.0 "$iface7 setdest 470.1 330.0 5.0"
puts "iface7: tcl=$iface7; id=[$iface7 id]; addr=[$iface7 node-addr]"

# rtPS1
set iface9 [$ns node 6.0.5] ;#
node id is 8.
$iface9 random-motion 0 ;#
disable random motion

```

```

$iface9 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface9 set X_ 470.0
$iface9 set Y_ 320.0
$iface9 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 14.0 "$iface9 setdest 570.0 320.0 [lindex $argv 1]"
#$ns at 5.0 "$iface9 setdest 470.1 320.0 5.0"
puts "iface9: tcl=$iface9; id=[$iface9 id]; addr=[$iface9 node-addr]"

# rtPS2
set iface11 [$ns node 6.0.6] ;#
node id is 8.
$iface11 random-motion 0 ;#
disable random motion
$iface11 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface11 set X_ 470.0
$iface11 set Y_ 310.0
$iface11 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 14.0 "$iface11 setdest 570.0 310.0 [lindex $argv 1]"
#$ns at 5.0 "$iface11 setdest 470.1 310.0 5.0"
puts "iface11: tcl=$iface11; id=[$iface11 id]; addr=[$iface11 node-
addr]"

# BE
set iface13 [$ns node 6.0.7] ;#
node id is 8.
$iface13 random-motion 0 ;#
disable random motion
$iface13 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface13 set X_ 440.0
$iface13 set Y_ 330.0
$iface13 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface13 setdest 540.0 330.0 [lindex $argv 1]"
#$ns at 5.0 "$iface13 setdest 440.1 330.0 5.0"
puts "iface13: tcl=$iface13; id=[$iface13 id]; addr=[$iface13 node-
addr]"

# BE1
set iface15 [$ns node 6.0.8] ;#
node id is 8.
$iface15 random-motion 0 ;#
disable random motion
$iface15 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface15 set X_ 440.0
$iface15 set Y_ 320.0
$iface15 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface15 setdest 540.0 320.0 [lindex $argv 1]"
#$ns at 5.0 "$iface15 setdest 440.1 320.0 5.0"

```

```

puts "iface15: tcl=$iface15; id=[$iface15 id]; addr=[$iface15 node-
addr]"

# BE2
set iface17 [$ns node 6.0.9] ;#
node id is 8.
$iface17 random-motion 0 ;#
disable random motion
$iface17 base-station [AddrParams addr2id [$bstation802 node-addr]]
;#attach mn to basestation
$iface17 set X_ 440.0
$iface17 set Y_ 310.0
$iface17 set Z_ 0.0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface17 setdest 540.0 310.0 [lindex $argv 1]"
#$ns at 5.0 "$iface17 setdest 440.1 310.0 5.0"
puts "iface17: tcl=$iface17; id=[$iface17 id]; addr=[$iface17 node-
addr]"

# add link to backbone
$ns duplex-link $bstation802 $router2 100Mb 40ms CBQ

# add Wimax nodes
set opt(netif) Phy/WirelessPhy/OFDM ;# network
interface type 802.16
set opt(mac) Mac/802_16 ;# MAC type 802.16

# radius =
Phy/WirelessPhy/OFDM set g_ 0.25
Phy/WirelessPhy set Pt_ 0.025
Phy/WirelessPhy set RXThresh_ 7.91016e-15 ;#500m:2.025e-12
#Phy/WirelessPhy set RXThresh_ 1.26562e-13 ;#1000m radius
Phy/WirelessPhy set CStresh_ [expr 0.8*[Phy/WirelessPhy set
RXThresh_]]

# configure Access Points
$ns node-config -adhocRouting $opt(adhocRouting) \
    -llType $opt(ll) \
    -macType $opt(mac) \
    -channel $chan \
    -ifqType $opt(ifq) \
    -ifqLen $opt(ifqlen) \
    -antType $opt(ant) \
    -propType $opt(prop) \
    -phyType $opt(netif) \
    -topoInstance $topo \
    -wiredRouting ON \
    -agentTrace OFF \
    -routerTrace OFF \
    -macTrace ON \
    -movementTrace OFF

# configure Base station 802.16
set bstation802_16 [$ns node 5.0.0] ;
$bstation802_16 random-motion 0
$bstation802_16 set X_ 450.0
$bstation802_16 set Y_ 350.0
$bstation802_16 set Z_ 0.0
$ns add-to-mppls-list $bstation802_16

```

```

puts "bstation802_16: tcl=$bstation802_16; id=[$bstation802_16 id];
addr=[$bstation802_16 node-addr]"
set clas [new SDUClassifier/Dest]
[$bstation802_16 set mac_(0)] add-classifier $clas
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set bs_sched [new WimaxScheduler/BS]
$bs_sched set-default-modulation $default_modulation
[$bstation802_16 set mac_(0)] set-scheduler $bs_sched
[$bstation802_16 set mac_(0)] set-channel 0
puts "Base-Station node created"

##### UGS
#set the number of UGS connections
#set nb_UGS 3

#### interval_ of the CBR traffic ####
set interval_ugs(1) 0.15
set interval_ugs(2) 0.2
set interval_ugs(3) 0.25
set interval_ugs(4) 0.27
set interval_ugs(5) 0.3

set interval_ugs(6) 0.04
set interval_ugs(7) 0.05
set interval_ugs(8) 0.1
set interval_ugs(9) 0.1
#####

#### SNR of the UGS connections#####
set SNR_ugs(1) 9.5
set SNR_ugs(2) 12.5
set SNR_ugs(3) 16.5
set SNR_ugs(4) 20.5
set SNR_ugs(5) 22.5

set SNR_ugs(6) 12.5
set SNR_ugs(7) 12.5
set SNR_ugs(8) 12.5
set SNR_ugs(9) 12.5
#####

# creation of the wireless interface 802.16
$ns node-config -MPLS OFF
$ns node-config -wiredRouting OFF \
-macTrace ON

### UGS 1 ###
set iface2 [$ns node 5.0.1] ;#
node id is 8.
$iface2 random-motion 0 ;#
disable random motion
$iface2 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface2 set X_ 450.0
$iface2 set Y_ 330.0
$iface2 set Z_ 0.0
set clas [new SDUClassifier/Dest]
[$iface2 set mac_(0)] add-classifier $clas

```

```

#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched [new WimaxScheduler/SS]
[$iface2 set mac_(0)] set-scheduler $ss_sched
[$iface2 set mac_(0)] set-channel 0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 15.0 "$iface2 setdest 550.0 330.0 [lindex $argv 1]"
#$ns at 5.0 "$iface2 setdest 450.1 330.0 5.0"
puts "iface2: tcl=$iface2; id=[$iface2 id]; addr=[$iface2 node-addr]"

#Create a UDP agent and attach it to wl_node$j
set udp_ugs1 [new Agent/UDP]
$udp_ugs1 set packetSize_ 1000
$ns attach-agent $router0 $udp_ugs1

# Create a CBR traffic source and attach it to udp_ugs
set cbr_ugs1 [new Application/Traffic/CBR]
$cbr_ugs1 set packetSize_ 1000 ;#600x8/0.005=960k
#$cbr_ugs1 set interval_ 0.15
$cbr_ugs1 set rate_ 1.5Mb
$cbr_ugs1 attach-agent $udp_ugs1

# Create the Null agent to sink traffic
set sink_ugs [new Agent/LossMonitor]
$ns attach-agent $multiFaceNode $sink_ugs

# Attach the 2 agents
$ns connect $udp_ugs1 $sink_ugs
$udp_ugs1 set fid_ 1
$udp_ugs1 set prio_ 2

set n3 0.14999999999999999

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0=>SERVICE_UGS, 1=>SERVICE_rtPS,
2=>SERVICE_nrtPS, 3=>SERVICE_BE
$ss_sched add-flow 5 [expr 30 + [$cbr_ugs1 set packetSize_] *
[Mac/802_16 set frame_duration_] / $n3] 0 0

##set-PeerNode-SNR PeerNode SNR
$ns at 1.5 "$bs_sched set-PeerNode-SNR [expr 1] $SNR_ugs(1)"

##set-PeerNode-UGSPeriodicity PeerNode Periodicity (periodicity of the
reservation, every k frames)
$ns at 1.5 "$bs_sched set-PeerNode-UGSPeriodicity [expr 1] 1"

### UGS 2 ###
$ns node-config -wiredRouting OFF \
                -macTrace ON                                ;# Mobile nodes
cannot do routing.

set iface4 [$ns node 5.0.2]                                ;#
node id is 8.
$iface4 random-motion 0                                    ;#
disable random motion
$iface4 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation

```



```

set iface6 [$ns node 5.0.3] ;#
node id is 8.
$iface6 random-motion 0 ;#
disable random motion
$iface6 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface6 set X_ 450.0
$iface6 set Y_ 310.0
$iface6 set Z_ 0.0
set clas [new SDUClassifier/Dest]
[$iface6 set mac_(0)] add-classifier $clas
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched [new WimaxScheduler/SS]
[$iface6 set mac_(0)] set-scheduler $ss_sched
[$iface6 set mac_(0)] set-channel 0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 15.0 "$iface6 setdest 550.0 310.0 [lindex $argv 1]"
#$ns at 5.0 "$iface6 setdest 450.1 310.0 5.0"
puts "iface6: tcl=$iface6; id=[$iface6 id]; addr=[$iface6 node-addr]"

#Create a UDP agent and attach it to wl_node$j
set udp_ugs3 [new Agent/UDP]
$udp_ugs3 set packetSize_ 1000
$ns attach-agent $router0 $udp_ugs3

# Create a CBR traffic source and attach it to udp_ugs
set cbr_ugs3 [new Application/Traffic/CBR]
$cbr_ugs3 set packetSize_ 1000 ;#600x8/0.005=960k
#$cbr_ugs3 set interval_ 0.25
$cbr_ugs3 set rate_ 1.5Mb
$cbr_ugs3 attach-agent $udp_ugs3

# Create the Null agent to sink traffic
set sink_ugs2 [new Agent/LossMonitor]
$ns attach-agent $multiFaceNode2 $sink_ugs2

# Attach the 2 agents
$ns connect $udp_ugs3 $sink_ugs2

$udp_ugs3 set fid_ 3
$udp_ugs3 set prio_ 2

set n5 0.25

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0=>SERVICE_UGS, 1=>SERVICE_rtPS,
2=>SERVICE_nrtPS, 3=>SERVICE_BE
$ss_sched add-flow 5 [expr 30 + [$cbr_ugs3 set packetSize_] *
[Mac/802_16 set frame_duration_] / $n5] 0 0

##set-PeerNode-SNR PeerNode SNR
$ns at 1.5 "$bs_sched set-PeerNode-SNR [expr 3] $SNR_ugs(3)"

```



```

##set-PeerNode-UGSPeriodicity PeerNode Periodicity (periodicity of the
reservation, every k frames)
$ns at 1.5 "$bs_sched set-PeerNode-UGSPeriodicity [expr 3] 1"

##### RTPS
# rtPS connections
# The identity of the first rtPS connection is k, the second is k+1,
and so on
set first_rtPS 101

#set the number of rtPS connections
#set nb_rtPS 3

#set the number of symbols reserved for unicast request opportunities
$bs_sched set-SymbolNumberForUnicastRequest 3

set rtPS_scheduler_ [lindex $argv 0]
$bs_sched set-rtPSscheduling $rtPS_scheduler_

# seed the default RNG
global defaultRNG
$defaultRNG seed 9999

#### interval_ #####
set interval_rtPS(1) 0.01
set interval_rtPS(2) 0.04
set interval_rtPS(3) 0.05
set interval_rtPS(4) 0.02
set interval_rtPS(5) 0.05
set interval_rtPS(6) 0.04
set interval_rtPS(7) 0.03
set interval_rtPS(8) 0.02
set interval_rtPS(9) 0.03
#####

#### SNR #####
set SNR_rtPS(1) 7.0
set SNR_rtPS(2) 7.5
set SNR_rtPS(3) 9.0
set SNR_rtPS(4) 12.0
set SNR_rtPS(5) 17.0
set SNR_rtPS(6) 17.5
set SNR_rtPS(7) 20.0
set SNR_rtPS(8) 24.0
set SNR_rtPS(9) 25.5
#####

#### WRR #####
# set the weights if using WRR
if {$rtPS_scheduler_ == "WRR"} {
set WRR_rtPS(1) 1
set WRR_rtPS(2) 1
set WRR_rtPS(3) 1
set WRR_rtPS(4) 2
set WRR_rtPS(5) 2
set WRR_rtPS(6) 3
set WRR_rtPS(7) 3

set WRR_rtPS(8) 4

```

```

set WRR_rtPS(9) 4
}
#####

#####
#set-TRSPparameters-SNR-Tr-Tp-L SNRth Tr Tp L
$bs_sched set-TRSPparameters-SNR-Tr-Tp-L 8.5 2 3 4
#####

$ns node-config -wiredRouting OFF \
                -macTrace ON                                ;# Mobile nodes
cannot do routing.

### rtPS 1 ###
set iface8 [$ns node 5.0.4]                                ;#
node id is 8.
$iface8 random-motion 0                                    ;#
disable random motion
$iface8 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface8 set X_ 470.0
$iface8 set Y_ 330.0
$iface8 set Z_ 0.0
set clas [new SDUClassifier/Dest]
[$iface8 set mac_(0)] add-classifier $clas
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched [new WimaxScheduler/SS]
[$iface8 set mac_(0)] set-scheduler $ss_sched
[$iface8 set mac_(0)] set-channel 0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 14.0 "$iface8 setdest 570.0 330.0 [lindex $argv 1]"
#$ns at 5.0 "$iface8 setdest 470.1 330.0 5.0"
puts "iface8: tcl=$iface8; id=[$iface8 id]; addr=[$iface8 node-addr]"

#Create a UDP agent and attach it to wl_node$j
set udp_rtPS1 [new Agent/UDP]
$udp_rtPS1 set packetSize_ 1000
$ns attach-agent $router0 $udp_rtPS1

#Traffic
#create video traffic
set cbr_rtps [new Application/Traffic/CBR]
$cbr_rtps set packetSize_ 1000 ;#600x8/0.005=960k
#$cbr_rtps set interval_ 0.01
$cbr_rtps set rate_ 1.5Mb
$cbr_rtps attach-agent $udp_rtPS1

# Create the Null agent to sink traffic
set sink_rtps [new Agent/LossMonitor]
$ns attach-agent $multiFaceNode3 $sink_rtps

# Attach the 2 agents
$ns connect $udp_rtPS1 $sink_rtps

$udp_rtPS1 set fid_ 101
$udp_rtPS1 set prio_ 1

```

```

set interval_rtPS($first_rtPS) $interval_rtPS([expr $first_rtPS + 1 -
$first_rtPS])

# uniform distribution
set sizeRNG_rtPS($first_rtPS) [new RNG]

set size_rtPS($first_rtPS) [new RandomVariable/Uniform]
$size_rtPS($first_rtPS) set min_ 500
$size_rtPS($first_rtPS) set max_ 1500
$size_rtPS($first_rtPS) use-rng $sizeRNG_rtPS($first_rtPS)

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0=>SERVICE_UGS, 1=>SERVICE_rtPS,
2=>SERVICE_nrtPS, 3=>SERVICE_BE
$ss_sched add-flow 5 0 0 1

##set-PeerNode-SNR PeerNode SNR
$ns at 1.5 "$bs_sched set-PeerNode-SNR [expr 3 + $first_rtPS + 1 -
$first_rtPS] $SNR_rtPS([expr $first_rtPS + 1 - $first_rtPS])"

##set-PeerNode-UnicastRequestPeriodicity PeerNode Periodicity
$ns at 1.5 "$bs_sched set-PeerNode-UnicastRequestPeriodicity [expr 3 +
$first_rtPS + 1 - $first_rtPS] 2"

if {$rtPS_scheduler_ == "WRR"} {
    # set-PeerNode-WRRschedulingForrtPS PeerNode Weight
    $ns at 1.5 "$bs_sched set-PeerNode-WRRschedulingForrtPS [expr 3 +
$first_rtPS + 1 - $first_rtPS] $WRR_rtPS([expr $first_rtPS + 1 -
$first_rtPS])"
}

### rtPS 2 ###
$ns node-config -wiredRouting OFF \
                -macTrace ON                ;# Mobile nodes
cannot do routing.

set iface10 [$ns node 5.0.5]                ;#
node id is 8.
$iface10 random-motion 0                    ;#
disable random motion
$iface10 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface10 set X_ 470.0
$iface10 set Y_ 320.0
$iface10 set Z_ 0.0
set clas [new SDUClassifier/Dest]
[$iface10 set mac_(0)] add-classifier $clas
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched [new WimaxScheduler/SS]
[$iface10 set mac_(0)] set-scheduler $ss_sched
[$iface10 set mac_(0)] set-channel 0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 14.0 "$iface10 setdest 570.0 320.0 [lindex $argv 1]"

```

```

# $ns at 5.0 "$iface10 setdest 470.1 320.0 5.0"
puts "iface10: tcl=$iface10; id=[$iface10 id]; addr=[$iface10 node-
addr]"

# Create a UDP agent and attach it to wl_node$j
set udp_rtPS2 [new Agent/UDP]
$udp_rtPS2 set packetSize_ 1000
$ns attach-agent $router0 $udp_rtPS2

# Traffic
# create video traffic
set cbr_rtps1 [new Application/Traffic/CBR]
$cbr_rtps1 set packetSize_ 1000 ;#600x8/0.005=960k
#$cbr_rtps1 set interval_ 0.04
$cbr_rtps1 set rate_ 1.5Mb
$cbr_rtps1 attach-agent $udp_rtPS2

# Create the Null agent to sink traffic
set sink_rtps1 [new Agent/LossMonitor]
$ns attach-agent $multiFaceNode4 $sink_rtps1

# Attach the 2 agents
$ns connect $udp_rtPS2 $sink_rtps1

$udp_rtPS2 set fid_ 102
$udp_rtPS2 set prio_ 1

set interval_rtPS(102) $interval_rtPS([expr 102 + 1 - 101])

# uniform distribution
set sizeRNG_rtPS(102) [new RNG]

set size_rtPS(102) [new RandomVariable/Uniform]
$size_rtPS(102) set min_ 500
$size_rtPS(102) set max_ 1500
$size_rtPS(102) use-rng $sizeRNG_rtPS(102)

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0=>SERVICE_UGS, 1=>SERVICE_rtPS,
2=>SERVICE_nrtPS, 3=>SERVICE_BE
$ss_sched add-flow 5 0 0 1

##set-PeerNode-SNR PeerNode SNR
$ns at 1.5 "$bs_sched set-PeerNode-SNR [expr 3 + 102 + 1 - 101]
$SNR_rtPS([expr 102 + 1 - 101])"

##set-PeerNode-UnicastRequestPeriodicity PeerNode Periodicity
$ns at 1.5 "$bs_sched set-PeerNode-UnicastRequestPeriodicity [expr 3 +
102 + 1 - 101] 2"

if {$rtPS_scheduler_ == "WRR"} {
    # set-PeerNode-WRRschedulingForrtPS PeerNode Weight
    $ns at 1.5 "$bs_sched set-PeerNode-WRRschedulingForrtPS [expr 3 +
102 + 1 - 101] $WRR_rtPS([expr 102 + 1 - 101])"
}

### rtPS 3 ###
$ns node-config -wiredRouting OFF \

```

```

                                -macTrace ON                                ;# Mobile nodes
cannot do routing.

set iface12 [$ns node 5.0.6] ;#
node id is 8.
$iface12 random-motion 0 ;#
disable random motion
$iface12 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface12 set X_ 470.0
$iface12 set Y_ 310.0
$iface12 set Z_ 0.0
set clas [new SDUClassifier/Dest]
[$iface12 set mac_(0)] add-classifier $clas
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched [new WimaxScheduler/SS]
[$iface12 set mac_(0)] set-scheduler $ss_sched
[$iface12 set mac_(0)] set-channel 0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 14.0 "$iface12 setdest 570.0 310.0 [lindex $argv 1]"
#$ns at 5.0 "$iface12 setdest 470.1 310.0 5.0"
puts "iface12: tcl=$iface12; id=[$iface12 id]; addr=[$iface12 node-
addr]"

#Create a UDP agent and attach it to wl_node$j
set udp_rtPS3 [new Agent/UDP]
$udp_rtPS3 set packetSize_ 1000
$ns attach-agent $router0 $udp_rtPS3

#Traffic
#create video traffic
set cbr_rtPS2 [new Application/Traffic/CBR]
$cbr_rtPS2 set packetSize_ 1000 ;#600x8/0.005=960k
#$cbr_rtPS2 set interval_ 0.05
$cbr_rtPS2 set rate_ 1.5Mb
$cbr_rtPS2 attach-agent $udp_rtPS3

# Create the Null agent to sink traffic
set sink_rtPS2 [new Agent/LossMonitor]
$ns attach-agent $multiFaceNode5 $sink_rtPS2

# Attach the 2 agents
$ns connect $udp_rtPS3 $sink_rtPS2

$udp_rtPS3 set fid_ 103
$udp_rtPS3 set prio_ 1

set interval_rtPS(103) $interval_rtPS([expr 103 + 1 - 101])

# uniform distribution
set sizeRNG_rtPS(103) [new RNG]

set size_rtPS(103) [new RandomVariable/Uniform]
$size_rtPS(103) set min_ 500
$size_rtPS(103) set max_ 1500
$size_rtPS(103) use-rng $sizeRNG_rtPS(103)

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType

```

```

##ServiceFlowSchedulingType: 0=>SERVICE_UGS, 1=>SERVICE_rtPS,
2=>SERVICE_nrtPS, 3=>SERVICE_BE
$ss_sched add-flow 5 0 0 1

##set-PeerNode-SNR PeerNode SNR
$ns at 1.5 "$bs_sched set-PeerNode-SNR [expr 3 + 103 + 1 - 101]
$SNR_rtPS([expr 103 + 1 - 101])"

##set-PeerNode-UnicastRequestPeriodicity PeerNode Periodicity
$ns at 1.5 "$bs_sched set-PeerNode-UnicastRequestPeriodicity [expr 3 +
103 + 1 - 101] 2"

if {$rtPS_scheduler_ == "WRR"} {
    # set-PeerNode-WRRschedulingForrtPS PeerNode Weight
    $ns at 1.5 "$bs_sched set-PeerNode-WRRschedulingForrtPS [expr 3 +
103 + 1 - 101] $WRR_rtPS([expr 103 + 1 - 101])"
}

##### BE

$ns node-config -wiredRouting OFF \
                -macTrace ON                                ;# Mobile nodes
cannot do routing.

#set BE_scheduler_ [lindex $argv 1]
#bs_sched set-BEscheduling $BE_scheduler_

### BE 1 ###
set iface14 [$ns node 5.0.7]                                ;#
node id is 8.
$iface14 random-motion 0                                    ;#
disable random motion
$iface14 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface14 set X_ 440.0
$iface14 set Y_ 330.0
$iface14 set Z_ 0.0
set clas [new SDUClassifier/Dest]
[$iface14 set mac_(0)] add-classifier $clas
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched [new WimaxScheduler/SS]
[$iface14 set mac_(0)] set-scheduler $ss_sched
[$iface14 set mac_(0)] set-channel 0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface14 setdest 540.0 330.0 [lindex $argv 1]"
#$ns at 5.0 "$iface14 setdest 440.1 330.0 5.0"
puts "iface14: tcl=$iface14; id=[$iface14 id]; addr=[$iface14 node-
addr]"

##set-PeerNode-SNR PeerNode SNR
$ns at 1.3 "$bs_sched set-PeerNode-SNR [expr 3 + 3 + 1] 12.3"

##set-BwRequestSendingPeriod BwRequestSendingPeriod_
$ss_sched set-BwRequestSendingPeriod 10

```

```

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0 => SERVICE_UGS, 1 => SERVICE_rtPS, 2 =>
SERVICE_nrtPS, 3 => SERVICE_BE

$ss_sched add-flow 1 0 0 3

    set udp_be [new Agent/UDP]
    $udp_be set packetSize_ 1000
    $udp_be set fid_ 301
    $udp_be set prio_ 3

    #create video traffic
    set cbr_be [new Application/Traffic/CBR]
    $cbr_be set packetSize_ 1000 ;#600x8/0.005=960k
    #$cbr_be set interval_ 0.35
    $cbr_be set rate_ 1.5Mb
    $cbr_be attach-agent $udp_be

    set sink_be [new Agent/LossMonitor]
    #sinkNode is transmitter
    $ns attach-agent $router0 $udp_be
    $ns attach-agent $multiFaceNode6 $sink_be

### BE 2 ###

$ns node-config -wiredRouting OFF \
                -macTrace ON                                ;# Mobile nodes
cannot do routing.

set ifacel6 [$ns node 5.0.8]                                ;#
node id is 8.
$ifacel6 random-motion 0                                    ;#
disable random motion
$ifacel6 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$ifacel6 set X_ 440.0
$ifacel6 set Y_ 320.0
$ifacel6 set Z_ 0.0
set clas [new SDUClassifier/Dst]
[$ifacel6 set mac_(0)] add-classifier $clas
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched [new WimaxScheduler/SS]
[$ifacel6 set mac_(0)] set-scheduler $ss_sched
[$ifacel6 set mac_(0)] set-channel 0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$ifacel6 setdest 540.0 320.0 [lindex $argv 1]"
#$ns at 5.0 "$ifacel6 setdest 440.1 320.0 5.0"
puts "ifacel6: tcl=$ifacel6; id=[$ifacel6 id]; addr=[$ifacel6 node-
addr]"

##set-PeerNode-SNR PeerNode SNR
$ns at 1.3 "$bs_sched set-PeerNode-SNR [expr 3 + 3 + 2] 12.32"

##set-BwRequestSendingPeriod BwRequestSendingPeriod_
$ss_sched set-BwRequestSendingPeriod 10

```

```

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0 => SERVICE_UGS, 1 => SERVICE_rtPS, 2 =>
SERVICE_nrtPS, 3 => SERVICE_BE

$ss_sched add-flow 1 0 0 3

    set udp_bel [new Agent/UDP]
    $udp_bel set packetSize_ 1000
    $udp_bel set fid_ 302
    $udp_bel set prio_ 3

    #create video traffic
    set cbr_bel [new Application/Traffic/CBR]
    $cbr_bel set packetSize_ 1000 ;#600x8/0.005=960k
    #$cbr_bel set interval_ 0.40
    $cbr_bel set rate_ 1.5Mb
    $cbr_bel attach-agent $udp_bel

    set sink_bel [new Agent/LossMonitor]

    #sinkNode is transmitter
    $ns attach-agent $router0 $udp_bel
    $ns attach-agent $multiFaceNode7 $sink_bel

### BE 3 ###

$ns node-config -wiredRouting OFF \
                -macTrace ON                                ;# Mobile nodes
cannot do routing.

set iface18 [$ns node 5.0.9]                                ;#
node id is 8.
$iface18 random-motion 0                                    ;#
disable random motion
$iface18 base-station [AddrParams addr2id [$bstation802_16 node-addr]]
;#attach mn to basestation
$iface18 set X_ 440.0
$iface18 set Y_ 310.0
$iface18 set Z_ 0.0
set clas [new SDUClassifier/Dst]
[$iface18 set mac_(0)] add-classifier $clas
#set the scheduler for the node. Must be changed to -shed [new
$opt(sched)]
set ss_sched [new WimaxScheduler/SS]
[$iface18 set mac_(0)] set-scheduler $ss_sched
[$iface18 set mac_(0)] set-channel 0
# define node movement. We start from outside the coverage, cross it
and leave.
$ns at 5.0 "$iface18 setdest 540.0 310.0 [lindex $argv 1]"
#$ns at 5.0 "$iface18 setdest 440.1 310.0 5.0"
puts "iface18: tcl=$iface18; id=[$iface18 id]; addr=[$iface18 node-
addr]"

##set-PeerNode-SNR PeerNode SNR
$ns at 1.3 "$bs_sched set-PeerNode-SNR [expr 3 + 3 + 3] 12.32"

##set-BwRequestSendingPeriod BwRequestSendingPeriod_
$ss_sched set-BwRequestSendingPeriod 10

```



```

## add-flow TrafficPriority MaximumSustainedTrafficRate
MinimumReservedTrafficRate ServiceFlowSchedulingType
##ServiceFlowSchedulingType: 0 => SERVICE_UGS, 1 => SERVICE_rtPS, 2 =>
SERVICE_nrtPS, 3 => SERVICE_BE

$ss_sched add-flow 1 0 0 3

        set udp_be2 [new Agent/UDP]
        $udp_be2 set packetSize_ 1000
        $udp_be2 set fid_ 303
        $udp_be2 set prio_ 3

        #create video traffic
        set cbr_be2 [new Application/Traffic/CBR]
        $cbr_be2 set packetSize_ 1000 ;#600x8/0.005=960k
        #$cbr_be2 set interval_ 0.45
        $cbr_be2 set rate_ 1.5Mb
        $cbr_be2 attach-agent $udp_be2

        set sink_be2 [new Agent/LossMonitor]

        #sinkNode is transmitter
        $ns attach-agent $router0 $udp_be2
        $ns attach-agent $multiFaceNode8 $sink_be2

#####
#####

$ns duplex-link $bstation802_16 $router2 100Mb 40ms CBQ

$ns duplex-link-op $router0 $router1 orient right
$ns duplex-link-op $router1 $router2 orient down
$ns duplex-link-op $router2 $bstation802_16 orient left-down
$ns duplex-link-op $router2 $bstation802 orient right-down

# add interfaces to MultiFaceNode
$multiFaceNode add-interface-node $iface1 ;# UGS
$multiFaceNode add-interface-node $iface2 ;# UGS

$multiFaceNode1 add-interface-node $iface3 ;# UGS1
$multiFaceNode1 add-interface-node $iface4 ;# UGS1

$multiFaceNode2 add-interface-node $iface5 ;# UGS2
$multiFaceNode2 add-interface-node $iface6 ;# UGS2

$multiFaceNode3 add-interface-node $iface7 ;# rtPS
$multiFaceNode3 add-interface-node $iface8 ;# rtPS

$multiFaceNode4 add-interface-node $iface9 ;# rtPS1
$multiFaceNode4 add-interface-node $iface10 ;# rtPS1

$multiFaceNode5 add-interface-node $iface11 ;# rtPS2
$multiFaceNode5 add-interface-node $iface12 ;# rtPS2

$multiFaceNode6 add-interface-node $iface13 ;# BE
$multiFaceNode6 add-interface-node $iface14 ;# BE

```

```

$multiFaceNode7 add-interface-node $iface15 ;# BE1
$multiFaceNode7 add-interface-node $iface16 ;# BE1

$multiFaceNode8 add-interface-node $iface17 ;# BE2
$multiFaceNode8 add-interface-node $iface18 ;# BE2

# install ND modules

# now WLAN
set nd_bs [$bstation802 install-nd]
$nd_bs set-router TRUE
$nd_bs router-lifetime 1800

set nd_mn [$iface1 install-nd]
set nd_mn3 [$iface3 install-nd]
set nd_mn5 [$iface5 install-nd]
set nd_mn7 [$iface7 install-nd]
set nd_mn9 [$iface9 install-nd]
set nd_mn11 [$iface11 install-nd]
set nd_mn13 [$iface13 install-nd]
set nd_mn15 [$iface15 install-nd]
set nd_mn17 [$iface17 install-nd]

# now WIMAX
set nd_bs2 [$bstation802_16 install-nd]
$nd_bs2 set-router TRUE
$nd_bs2 router-lifetime 20 ;#just enough to expire while we are
connected to wlan.

set nd_mn2 [$iface2 install-nd]
set nd_mn4 [$iface4 install-nd]
set nd_mn6 [$iface6 install-nd]
set nd_mn8 [$iface8 install-nd]
set nd_mn10 [$iface10 install-nd]
set nd_mn12 [$iface12 install-nd]
set nd_mn14 [$iface14 install-nd]
set nd_mn16 [$iface16 install-nd]
set nd_mn18 [$iface18 install-nd]

# install interface manager into multi-interface node and CN
Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2 set debug_ 1

set handover [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode install-ifmanager $handover
$nd_mn set-ifmanager $handover
$handover nd_mac $nd_mn [$iface1 set mac_(0)] ;#to know how to send RS
$nd_mn2 set-ifmanager $handover
$handover nd_mac $nd_mn2 [$iface2 set mac_(0)] ;#to know how to send
RS

set handover1 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode1 install-ifmanager $handover1
$nd_mn3 set-ifmanager $handover1
$handover1 nd_mac $nd_mn3 [$iface3 set mac_(0)] ;#to know how to send
RS
$nd_mn4 set-ifmanager $handover1
$handover1 nd_mac $nd_mn4 [$iface4 set mac_(0)] ;#to know how to send
RS

```

```

set handover2 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode2 install-ifmanager $handover2
$nd_mn5 set-ifmanager $handover2
$handover2 nd_mac $nd_mn5 [$iface5 set mac_(0)] ;#to know how to send
RS
$nd_mn6 set-ifmanager $handover2
$handover2 nd_mac $nd_mn6 [$iface6 set mac_(0)] ;#to know how to send
RS

```

```

set handover3 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode3 install-ifmanager $handover3
$nd_mn7 set-ifmanager $handover3
$handover3 nd_mac $nd_mn7 [$iface7 set mac_(0)] ;#to know how to send
RS
$nd_mn8 set-ifmanager $handover3
$handover3 nd_mac $nd_mn8 [$iface8 set mac_(0)] ;#to know how to send
RS

```

```

set handover4 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode4 install-ifmanager $handover4
$nd_mn9 set-ifmanager $handover4
$handover4 nd_mac $nd_mn9 [$iface9 set mac_(0)] ;#to know how to send
RS
$nd_mn10 set-ifmanager $handover4
$handover4 nd_mac $nd_mn10 [$iface10 set mac_(0)] ;#to know how to
send RS

```

```

set handover5 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode5 install-ifmanager $handover5
$nd_mn11 set-ifmanager $handover5
$handover5 nd_mac $nd_mn11 [$iface11 set mac_(0)] ;#to know how to
send RS
$nd_mn12 set-ifmanager $handover5
$handover5 nd_mac $nd_mn12 [$iface12 set mac_(0)] ;#to know how to
send RS

```

```

set handover6 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode6 install-ifmanager $handover6
$nd_mn13 set-ifmanager $handover6
$handover6 nd_mac $nd_mn13 [$iface13 set mac_(0)] ;#to know how to
send RS
$nd_mn14 set-ifmanager $handover6
$handover6 nd_mac $nd_mn14 [$iface14 set mac_(0)] ;#to know how to
send RS

```

```

set handover7 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode7 install-ifmanager $handover7
$nd_mn15 set-ifmanager $handover7
$handover7 nd_mac $nd_mn15 [$iface15 set mac_(0)] ;#to know how to
send RS
$nd_mn16 set-ifmanager $handover7
$handover7 nd_mac $nd_mn16 [$iface16 set mac_(0)] ;#to know how to
send RS

```

```

set handover8 [new Agent/MIHUser/IFMNGMT/MIPV6/Handover/Handover2]
$multiFaceNode8 install-ifmanager $handover8
$nd_mn17 set-ifmanager $handover8
$handover8 nd_mac $nd_mn17 [$iface17 set mac_(0)] ;#to know how to
send RS
$nd_mn18 set-ifmanager $handover8

```

```

$handover8 nd_mac $nd_mn18 [$iface18 set mac_(0)] ;#to know how to
send RS

set ifmgmt_cn [$router0 install-default-ifmanager]

# install MIH in multi-interface node
set mih [$multiFaceNode install-mih]
set mih1 [$multiFaceNode1 install-mih]
set mih2 [$multiFaceNode2 install-mih]
set mih3 [$multiFaceNode3 install-mih]
set mih4 [$multiFaceNode4 install-mih]
set mih5 [$multiFaceNode5 install-mih]
set mih6 [$multiFaceNode6 install-mih]
set mih7 [$multiFaceNode7 install-mih]
set mih8 [$multiFaceNode8 install-mih]

$handover connect-mih $mih ;#create connection between MIH and iface
management
$handover1 connect-mih $mih1 ;#create connection between MIH and iface
management
$handover2 connect-mih $mih2 ;#create connection between MIH and iface
management
$handover3 connect-mih $mih3 ;#create connection between MIH and iface
management
$handover4 connect-mih $mih4 ;#create connection between MIH and iface
management
$handover5 connect-mih $mih5 ;#create connection between MIH and iface
management
$handover6 connect-mih $mih6 ;#create connection between MIH and iface
management
$handover7 connect-mih $mih7 ;#create connection between MIH and iface
management
$handover8 connect-mih $mih8 ;#create connection between MIH and iface
management

# install MIH on AP/BS
set mih_bs [$bstation802 install-mih]
set tmp_bs [$bstation802 set mac_(0)]
$tmp_bs mih $mih_bs
$mih_bs add-mac $tmp_bs

set mih_bs [$bstation802_16 install-mih]
set tmp_bs [$bstation802_16 set mac_(0)]
$tmp_bs mih $mih_bs
$mih_bs add-mac $tmp_bs

# Now we can register the MIH module with all the MACs
set tmp20 [$iface1 set mac_(0)] ;#in 802.11 one interface is created
$tmp20 mih $mih
$mih add-mac $tmp20 ;#inform the MIH about the local MAC
set tmp20 [$iface2 set mac_(0)] ;#in 802.16 one interface is created
$tmp20 mih $mih
$mih add-mac $tmp20 ;#inform the MIH about the local MAC

set tmp21 [$iface3 set mac_(0)] ;#in 802.11 one interface is created
$tmp21 mih $mih1

```

```

$mih1 add-mac $tmp21                ;#inform the MIH about the local MAC
set tmp21 [$iface4 set mac_(0)] ;#in 802.16 one interface is created
$tmp21 mih $mih1
$mih1 add-mac $tmp21                ;#inform the MIH about the local MAC

set tmp22 [$iface5 set mac_(0)] ;#in 802.11 one interface is created
$tmp22 mih $mih2
$mih2 add-mac $tmp22                ;#inform the MIH about the local MAC
set tmp22 [$iface6 set mac_(0)] ;#in 802.16 one interface is created
$tmp22 mih $mih2
$mih2 add-mac $tmp22                ;#inform the MIH about the local MAC

set tmp23 [$iface7 set mac_(0)] ;#in 802.11 one interface is created
$tmp23 mih $mih3
$mih3 add-mac $tmp23                ;#inform the MIH about the local MAC
set tmp23 [$iface8 set mac_(0)] ;#in 802.16 one interface is created
$tmp23 mih $mih3
$mih3 add-mac $tmp23                ;#inform the MIH about the local MAC

set tmp24 [$iface9 set mac_(0)] ;#in 802.11 one interface is created
$tmp24 mih $mih4
$mih4 add-mac $tmp24                ;#inform the MIH about the local MAC
set tmp24 [$iface10 set mac_(0)] ;#in 802.16 one interface is created
$tmp24 mih $mih4
$mih4 add-mac $tmp24                ;#inform the MIH about the local MAC

set tmp25 [$iface11 set mac_(0)] ;#in 802.11 one interface is created
$tmp25 mih $mih5
$mih5 add-mac $tmp25                ;#inform the MIH about the local MAC
set tmp25 [$iface12 set mac_(0)] ;#in 802.16 one interface is created
$tmp25 mih $mih5
$mih5 add-mac $tmp25                ;#inform the MIH about the local MAC

set tmp26 [$iface13 set mac_(0)] ;#in 802.11 one interface is created
$tmp26 mih $mih6
$mih6 add-mac $tmp26                ;#inform the MIH about the local MAC
set tmp26 [$iface14 set mac_(0)] ;#in 802.16 one interface is created
$tmp26 mih $mih6
$mih6 add-mac $tmp26                ;#inform the MIH about the local MAC

set tmp27 [$iface15 set mac_(0)] ;#in 802.11 one interface is created
$tmp27 mih $mih7
$mih7 add-mac $tmp27                ;#inform the MIH about the local MAC
set tmp27 [$iface16 set mac_(0)] ;#in 802.16 one interface is created
$tmp27 mih $mih7
$mih7 add-mac $tmp27                ;#inform the MIH about the local MAC

set tmp28 [$iface17 set mac_(0)] ;#in 802.11 one interface is created
$tmp28 mih $mih8
$mih8 add-mac $tmp28                ;#inform the MIH about the local MAC
set tmp28 [$iface18 set mac_(0)] ;#in 802.16 one interface is created
$tmp28 mih $mih8
$mih8 add-mac $tmp28                ;#inform the MIH about the local MAC

# set the starting time for Router Advertisements
$ns at 2 "$nd_bs start-ra"
$ns at 2 "$nd_bs2 start-ra"

#####MPLS
#MPLS Settings
# RSVPTTE Settings

```

```

$ns PATH-color "purple"
$ns PATHERR-color "red"
$ns PATHTEAR-color "red"
$ns RESV-color "grey"
$ns RESVERR-color "red"
$ns RESVCONF-color "red"
$ns color 1002 green
Agent/RSVP set noisy_ 255
Agent/RSVP set refresh_ 30
Agent/RSVP set lifetime_factor_ 3
Agent/RSVP set ip6_ 0
Agent/RSVP set nam_ 1

# One RSVP-TE Agent on each MPLS Node
$ns configure-rsvpte-on-all-mpls-nodes

$ns cfg-cbq-for-SBTS 2040 DropTail 990000.000000 0.010000 auto
0.000000
$ns cfg-cbq-for-HBTS 10 DropTail 0.000000 0.000000 auto 0.000000
$ns cfg-cbq-for-STS 10 DropTail 0.000000 0.000000 auto 0.000000
$ns cfg-cbq-for-RTS 2040 DropTail 0.000000 0.990000 auto 0.000000
$ns bind-rsvpte-to-SBTS

$ns enable-control-driven

# Mobile IP reg req message and other things
$ns bind-flowid-to-SBTS 0
$ns bind-flowid-to-SBTS 1
# MPLS test connection

# MPLS test tunnel
$ns compute-routes
#####MPLS

##### UGS 1 #####
$ns bind-flowid-to-SBTS 1

$handover add-flow $sink_ugs $udp_ugs1 $iface2 1

$ns at 2.0 "$cbr_ugs1 start"
#$ns at 15.0 "$cbr_ugs1 stop"

##### UGS 2 #####
$ns bind-flowid-to-SBTS 2

$handover1 add-flow $sink_ugs1 $udp_ugs2 $iface4 1

$ns at 2.0 "$cbr_ugs2 start"
#$ns at 20.0 "$cbr_ugs2 stop"

##### UGS 3 #####
$ns bind-flowid-to-SBTS 3

$handover2 add-flow $sink_ugs2 $udp_ugs3 $iface6 1

$ns at 2.0 "$cbr_ugs3 start"
#$ns at 20.0 "$cbr_ugs3 stop"

##### rtPS 1 #####
$ns bind-flowid-to-SBTS 101

```

```

$handover3 add-flow $sink_rtps $udp_rtPS1 $iface8 1

$ns at 2.0 "$cbr_rtps start"
#$ns at 20.0 "$cbr_rtps stop"

##### rtPS 2 #####
$ns bind-flowid-to-SBTS 102

$handover4 add-flow $sink_rtps1 $udp_rtPS2 $iface10 1

$ns at 2.0 "$cbr_rtps1 start"
#$ns at 20.0 "$cbr_rtps1 stop"

##### rtPS 3 #####
$ns bind-flowid-to-SBTS 103

$handover5 add-flow $sink_rtps2 $udp_rtPS3 $iface12 1

$ns at 2.0 "$cbr_rtps2 start"
#$ns at 20.0 "$cbr_rtps2 stop"

##### BE 1 #####
$ns bind-flowid-to-SBTS 301

$handover6 add-flow $sink_be $udp_be $iface14 1

$ns at 2.0 "$cbr_be start"
#$ns at 15.0 "$cbr_be stop"

##### BE 2 #####
$ns bind-flowid-to-SBTS 302

$handover7 add-flow $sink_bel $udp_bel $iface16 1

$ns at 2.0 "$cbr_bel start"
#$ns at 15.0 "$cbr_bel stop"

##### BE 3 #####
$ns bind-flowid-to-SBTS 303

$handover8 add-flow $sink_be2 $udp_be2 $iface18 1

$ns at 2.0 "$cbr_be2 start"
#$ns at 15.0 "$cbr_be2 stop"

#defines function for flushing and closing files
proc finish {} {
    global ns f f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 f10 f11 f12 f13 f14 f15
    f16 f17 f18 f19 f20 f21 f22 f23 f24 f25 f26 namtrace
    $ns flush-trace
    close $f
    close $f0
    close $f1
    close $f2
    close $f3
    close $f4
    close $f5
    close $f6
    close $f7

```

```
close $f8

close $f9
close $f10
close $f11
close $f12
close $f13
close $f14
close $f15
close $f16
close $f17

close $f18
close $f19
close $f20
close $f21
close $f22
close $f23
close $f24
close $f25
close $f26

close $namtrace
puts " Simulation ended."
# Plot Recorded Statistics
exec xgraph rtps.tr rtps1.tr rtps2.tr be.tr bel.tr be2.tr -
geometry 800x400 &
exec xgraph lostrtps.tr lostrtps1.tr lostrtps2.tr lostbe.tr
lostbel.tr lostbe2.tr -geometry 800x400 &
exec xgraph delayrtps.tr delayrtps1.tr delayrtps2.tr delaybe.tr
delaybel.tr delaybe2.tr -geometry 800x400 &
exit 0
}

# Initialize Flags
set holdtime 0
set holdseq 0

set holdtime1 0
set holdseq1 0

set holdtime2 0
set holdseq2 0

set holdtime3 0
set holdseq3 0

set holdtime4 0
set holdseq4 0

set holdtime5 0
set holdseq5 0

set holdtime6 0
set holdseq6 0

set holdtime7 0
set holdseq7 0

set holdtime8 0
```



```

set holdseq8 0

# Function To record Statistcis (Bit Rate, Delay, Drop)
proc record {} {
    global sink_ugs sink_ugs1 sink_ugs2 sink_rtps sink_rtps1
    sink_rtps2 sink_be sink_be1 sink_be2 f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 f10
    f11 f12 f13 f14 f15 f16 f17 f18 f19 f20 f21 f22 f23 f24 f25 f26
    holdtime holdseq holdtime1 holdseq1 holdtime2 holdseq2 holdtime3
    holdseq3 holdtime4 holdseq4 holdtime5 holdseq5 holdtime6 holdseq6
    holdtime7 holdseq7 holdtime8 holdseq8 handover handover1 handover2
    handover3 handover4 handover5 handover6 handover7 handover8
    #Get an instance of the simulator
    set ns [Simulator instance]
    #Set the time after which the procedure should be called again
    set time 0.5
    #How many bytes have been received by the traffic sinks?
    set bw0 [$sink_ugs set bytes_]
        set bw1 [$sink_rtps set bytes_]
        set bw2 [$sink_be set bytes_]
        set bw3 [$sink_ugs1 set bytes_]
        set bw4 [$sink_rtps1 set bytes_]
        set bw5 [$sink_be1 set bytes_]
        set bw6 [$sink_ugs2 set bytes_]
        set bw7 [$sink_rtps2 set bytes_]
        set bw8 [$sink_be2 set bytes_]

    set bw9 [$sink_ugs set nlost_]
        set bw10 [$sink_rtps set nlost_]
        set bw11 [$sink_be set nlost_]
        set bw12 [$sink_ugs1 set nlost_]
        set bw13 [$sink_rtps1 set nlost_]
        set bw14 [$sink_be1 set nlost_]
        set bw15 [$sink_ugs2 set nlost_]
        set bw16 [$sink_rtps2 set nlost_]
        set bw17 [$sink_be2 set nlost_]

        set bw18 [$sink_ugs set lastPktTime_]
        set bw19 [$sink_ugs set npkts_]
        set bw20 [$sink_rtps set lastPktTime_]
        set bw21 [$sink_rtps set npkts_]
        set bw22 [$sink_be set lastPktTime_]
        set bw23 [$sink_be set npkts_]
        set bw24 [$sink_ugs1 set lastPktTime_]
        set bw25 [$sink_ugs1 set npkts_]
        set bw26 [$sink_rtps1 set lastPktTime_]
        set bw27 [$sink_rtps1 set npkts_]
        set bw28 [$sink_be1 set lastPktTime_]
        set bw29 [$sink_be1 set npkts_]
        set bw30 [$sink_ugs2 set lastPktTime_]
        set bw31 [$sink_ugs2 set npkts_]
        set bw32 [$sink_rtps2 set lastPktTime_]
        set bw33 [$sink_rtps2 set npkts_]
        set bw34 [$sink_be2 set lastPktTime_]
        set bw35 [$sink_be2 set npkts_]

    #Get the current time
    set now [$ns now]

    #Calculate the bandwidth (in MBit/s) and record it to the
files

```

```

puts $f0 "$now [expr $bw0/$time*8/1000000]"
puts $f1 "$now [expr $bw1/$time*8/1000000]"
puts $f2 "$now [expr $bw2/$time*8/1000000]"
puts $f3 "$now [expr $bw3/$time*8/1000000]"
puts $f4 "$now [expr $bw4/$time*8/1000000]"
puts $f5 "$now [expr $bw5/$time*8/1000000]"
puts $f6 "$now [expr $bw6/$time*8/1000000]"
puts $f7 "$now [expr $bw7/$time*8/1000000]"
puts $f8 "$now [expr $bw8/$time*8/1000000]"

# Record Packet Loss Rate in File
puts $f9 "$now [expr $bw9/$time]"
puts $f10 "$now [expr $bw10/$time]"
puts $f11 "$now [expr $bw11/$time]"
puts $f12 "$now [expr $bw12/$time]"
puts $f13 "$now [expr $bw13/$time]"
puts $f14 "$now [expr $bw14/$time]"
puts $f15 "$now [expr $bw15/$time]"
puts $f16 "$now [expr $bw16/$time]"
puts $f17 "$now [expr $bw17/$time]"

# Record Packet Delay in File
if { $bw19 > $holdseq } {
    puts $f18 "$now [expr ($bw18 - $holdtime)/($bw19 -
$holdseq)]"
} else {
    puts $f18 "$now [expr ($bw19 - $holdseq)]"
}

if { $bw21 > $holdseq1 } {
    puts $f19 "$now [expr ($bw20 - $holdtime1)/($bw21 -
$holdseq1)]"
} else {
    puts $f19 "$now [expr ($bw21 - $holdseq1)]"
}

if { $bw23 > $holdseq2 } {
    puts $f20 "$now [expr ($bw22 - $holdtime2)/($bw23 -
$holdseq2)]"
} else {
    puts $f20 "$now [expr ($bw23 - $holdseq2)]"
}

if { $bw25 > $holdseq3 } {
    puts $f21 "$now [expr ($bw24 - $holdtime3)/($bw25 -
$holdseq3)]"
} else {
    puts $f21 "$now [expr ($bw25 - $holdseq3)]"
}

if { $bw27 > $holdseq4 } {
    puts $f22 "$now [expr ($bw26 - $holdtime4)/($bw27 -
$holdseq4)]"
} else {
    puts $f22 "$now [expr ($bw27 - $holdseq4)]"
}

if { $bw29 > $holdseq5 } {
    puts $f23 "$now [expr ($bw28 - $holdtime5)/($bw29 -
$holdseq5)]"
} else {

```

```

        puts $f23 "$now [expr ($bw29 - $holdseq5)]"
    }

    if { $bw31 > $holdseq6 } {
        puts $f24 "$now [expr ($bw30 - $holdtime6)/($bw31 -
$holdseq6)]"
    } else {
        puts $f24 "$now [expr ($bw31 - $holdseq6)]"
    }

    if { $bw33 > $holdseq7 } {
        puts $f25 "$now [expr ($bw32 - $holdtime7)/($bw33 -
$holdseq7)]"
    } else {
        puts $f25 "$now [expr ($bw33 - $holdseq7)]"
    }

    if { $bw35 > $holdseq8 } {
        puts $f26 "$now [expr ($bw34 - $holdtime8)/($bw35 -
$holdseq8)]"
    } else {
        puts $f26 "$now [expr ($bw35 - $holdseq8)]"
    }

    $handover UGS [expr $bw0/$time*8/1000000]
    $handover1 UGS [expr $bw3/$time*8/1000000]
    $handover2 UGS [expr $bw6/$time*8/1000000]
    $handover3 RTPS [expr $bw1/$time*8/1000000]
    $handover4 RTPS [expr $bw4/$time*8/1000000]
    $handover5 RTPS [expr $bw7/$time*8/1000000]
    $handover6 BE [expr $bw2/$time*8/1000000]
    $handover7 BE [expr $bw5/$time*8/1000000]
    $handover8 BE [expr $bw8/$time*8/1000000]

    #Reset the bytes_ values on the traffic sinks
    $sink_ugs set bytes_ 0
    $sink_ugs1 set bytes_ 0
    $sink_ugs2 set bytes_ 0
    $sink_rtps set bytes_ 0
    $sink_rtps1 set bytes_ 0
    $sink_rtps2 set bytes_ 0
    $sink_be set bytes_ 0
    $sink_be1 set bytes_ 0
    $sink_be2 set bytes_ 0

    $sink_ugs set nlost_ 0
    $sink_ugs1 set nlost_ 0
    $sink_ugs2 set nlost_ 0
    $sink_rtps set nlost_ 0
    $sink_rtps1 set nlost_ 0
    $sink_rtps2 set nlost_ 0
    $sink_be set nlost_ 0
    $sink_be1 set nlost_ 0
    $sink_be2 set nlost_ 0

    set holdtime $bw18
    set holdseq $bw19
    set holdtime1 $bw20
    set holdseq1 $bw21
    set holdtime2 $bw22

```

```
set holdseq2 $bw23
set holdtime3 $bw24
set holdseq3 $bw25
set holdtime4 $bw26
set holdseq4 $bw27
set holdtime5 $bw28
set holdseq5 $bw29
set holdtime6 $bw30
set holdseq6 $bw31
set holdtime7 $bw32
set holdseq7 $bw33
set holdtime8 $bw34
set holdseq8 $bw35

#Re-schedule the procedure
$ns at [expr $now+$time] "record"
}
#####

$ns at 2.0 "record"

$ns at 50.0 "finish"

$ns run
#####
```