

UNIVERSIDADE FEDERAL DA BAHIA
UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA
MESTRADO EM CIÊNCIA DA COMPUTAÇÃO

ITALO VALCY DA SILVA BRITO

Suporte a mobilidade em redes *mesh* sem fio
definidas por software

Salvador
2016

Italo Valcy da Silva Brito

Suporte a mobilidade em redes *mesh* sem fio definidas por software

Dissertação apresentada ao Mestrado em Ciência da Computação da Universidade Federal da Bahia e Universidade Estadual de Feira de Santana, como requisito parcial para obtenção do grau de Mestre.

Orientador: Prof. Dr. Gustavo Bittencourt Figueiredo

Salvador

2016

Ficha catalográfica.

Brito, Italo Valcy da Silva

Suporte a mobilidade em redes *mesh* sem fio definidas por software / Italo Valcy da Silva Brito – Salvador, 2016.

121p.: il.

Orientador: Prof. Dr. Gustavo Bittencourt Figueiredo.

Dissertação (mestrado) – Universidade Federal da Bahia, Instituto de Matemática, 2016.

1. Redes de Computadores. 2. Sistemas distribuídos.

I. Figueiredo, Gustavo Bittencourt. II. Universidade Federal da Bahia. Instituto de Matemática. III Título.

TERMO DE APROVAÇÃO

ITALO VALCY DA SILVA BRITO

SUORTE A MOBILIDADE EM REDES *MESH* SEM FIO DEFINIDAS POR SOFTWARE

Esta dissertação foi julgada adequada à obtenção do título de Mestre em Ciência da Computação e aprovada em sua forma final pelo Programa Multi-institucional de Pós-Graduação em Ciência da Computação da UFBA-UEFS.

Salvador, 20 de Julho de 2016

Prof. Dr. Gustavo Bittencourt Figueiredo
Universidade Federal da Bahia

Prof. Dr. Leobino Nascimento Sampaio
Universidade Federal da Bahia

Prof. Dr. Daniel Macedo Batista
Universidade Estadual de São Paulo

Prof. Dr. Miguel Elias Mitre Campista
Universidade Federal do Rio de Janeiro

*Dedico este trabalho à toda minha família e, em particular,
à minha esposa Illa.*

AGRADECIMENTOS

*Certas palavras não podem ser ditas em qualquer lugar e hora qualquer.
Estritamente reservadas para companheiros de confiança,
devem ser sacralmente pronunciadas em tom muito especial.*

– Carlos Drummond

Em primeiro lugar, agradeço a Deus pelo dom da vida, pela serenidade, pela sabedoria e pela saúde para concluir este trabalho.

Agradeço à minha família, aos meus pais Valdir e Ana, aos meus irmãos Anyele e Igor, à minha sogra Maria Pires e, em especial, à minha esposa e companheira Illa Brito, que me deu forças para concluir este trabalho, sempre com frases otimistas e carinhosas. Quaisquer palavras ditas aqui seriam insuficientes para expressar minha gratidão e alegria em tê-los ao meu lado.

Aos amigos(as) e companheiros(as) de trabalho do PoP-BA/RNP e da UFBA, novos e antigos, cujo relacionamento ultrapassa o ambiente profissional, formando laços de amizade que certamente serão mantidos por muito tempo.

Agradeço ao meu orientador Prof. Gustavo B. Figueiredo e ao grupo de pesquisa GRADE pelo apoio técnico e científico, fundamentais para a realização do mestrado, pelas experiências compartilhadas e pelas lições aprendidas.

Por fim, agradeço pelos esforços de todos, pois sozinho não seria capaz de chegar ao fim dessa jornada.

RESUMO

As Redes em Malha Sem Fio (WMN, do inglês *Wireless Mesh Networks*) têm sido apontadas como solução vantajosa para provimento de infraestrutura sem fio, principalmente, quando comparadas com outras arquiteturas baseadas no padrão IEEE 802.11. Adicionalmente, o uso de dispositivos móveis tem crescido de tal maneira que redes Wi-Fi ubíquas e suporte a mobilidade transparente são requisitos cruciais para os provedores de rede. De modo a proporcionar uma boa experiência de mobilidade aos usuários, a rede sem fio deve prover *handoff* transparente aos clientes móveis, preservando as métricas de QoS e QoE em níveis aceitáveis para acomodar aplicações de tempo real e de multimídia interativa, cuja utilização torna-se cada vez mais popular. Não obstante, as estratégias de gerenciamento de mobilidade tradicionais, baseadas no padrão 802.11, incorrem em latência e degradação da QoS na rede, impactando negativamente o desempenho das aplicações. Desse modo, este trabalho propõe uma nova estratégia de *handoff* definido por software e baseado em monitoramento passivo da qualidade dos enlaces. Nessa estratégia, o processo de *handoff* é inteiramente controlado pela rede, o que permite mobilidade transparente aos usuários e não requer qualquer mudança nos dispositivos do cliente. Assim, as principais contribuições deste trabalho são: i) propor uma métrica de qualidade de enlaces Wi-Fi, a fim de identificar oportunidades de *handoff*, baseado em monitoramento passivo; ii) propor um algoritmo de gerenciamento de *handoff* composto pelas estratégias de decisão e execução do *handoff*, além da seleção de AP para o cliente móvel, os quais, em conjunto, melhoram a experiência de mobilidade e otimizam o uso dos recursos de rede; e iii) desenvolver um protótipo como prova de conceito dessas estratégias, através do *framework* OpenWiMesh. Foi realizada uma análise detalhada sobre o desempenho da proposta, comparando-a com uma estratégia baseada no protocolo B.A.T.M.A.N. e *handoff* controlado pelo cliente. Os resultados mostram que a proposta de gerenciamento de *handoff* melhorou em até 70% as métricas de QoE durante a mobilidade de clientes, ao passo que manteve métricas chave de QoS abaixo de limites de restrição para aplicações de tempo real.

Palavras-chave: Redes Definidas por Software (SDN), Redes em Malha Sem Fio, Mobilidade, *handoff* transparente, Qualidade de Serviço (QoS), Qualidade de Experiência (QoE).

ABSTRACT

Wireless Mesh Networks (WMN) have been pointed out as one of the best cost-effective wireless solutions when compared to architectures based on the IEEE 802.11 standard. In addition, mobile data traffic has been growing to a point where ubiquitous Wi-Fi deployments and support for enhanced mobility experience have become requirements for network providers. To efficiently deliver an enhanced mobility experience, wireless networks must provide seamless handoff to mobile clients. This handoff process should preserve QoS and QoE at acceptable levels to accommodate real-time and interactive multimedia applications, whose usage is becoming increasingly popular. However, traditional mobility strategies based on the IEEE 802.11 standard incur latency and QoS degradation, deteriorating the application performance. This work proposes a new software-defined handoff strategy based on passive wireless link quality metrics. In our strategy, the handoff process is entirely controlled by network enabling transparent and seamless mobility, requiring no configuration changes in the mobile clients. The contributions of this work are three-fold. First, we propose a wireless link quality metric approach to identify handoff opportunities based on passive monitoring. Second, we propose a handoff management algorithm, composed of decision and execution strategies. The proposed handoff management algorithm improves the mobility experience and optimizes the network resources usage, providing a better selection of Wireless Access Points by the mobile client. Finally, we developed a system prototype as a proof-of-concept using the OpenWiMesh framework. We carefully analyze the performance of our proposal in comparison to a deployment based on B.A.T.M.A.N. protocol and a handoff controlled by client. Experimental results demonstrate the effectiveness of the proposed handoff strategy in providing seamless handoff, improving QoE up to 70% while keeping key QoS metrics below real-time applications constraints.

Keywords: Software-defined networks (SDN), wireless mesh networks, mobility, seamless handoff, Quality of Service (QoS), Quality of Experience (QoE).

LISTA DE FIGURAS

2.1	Rede Mesh Sem Fio (adaptado de Amir et al. (2010)).	23
2.2	Arquitetura SDN e suas abstrações fundamentais (adaptado de Kreutz et al. (2015)).	27
3.1	Potência de Sinal Relativa com Limiar e Histerese (PAUL, 2014)	40
3.2	Exemplo de funcionamento do protocolo <i>IP móvel</i>	44
3.3	Exemplo de funcionamento do protocolo PMIPv6	45
4.1	Arquitetura do OpenWiMesh	52
4.2	Exemplo de cenário de WMN com OpenWiMesh	53
4.3	Exemplo do estabelecimento da WMN no OpenWiMesh	59
4.4	Diagrama de Classes do OpenWiMesh	62
4.5	Dispersão dos fluxos conforme algoritmo de TE (BRITO et al., 2014)	68
5.1	Estratégia de mobilidade e <i>handoff</i> transparente no OpenWiMesh	70
5.2	Diagrama de blocos do gerenciamento de mobilidade do OpenWiMesh	71
5.3	Exemplo de coleta de estatísticas 802.11 via comando <i>iw</i>	76
5.4	Diagrama de Classes do OpenWiMesh com Mobilidade	81
6.1	Planta baixa do ambiente de teste	85
6.2	Mapa de intensidade de sinal (dBm) no ambiente de teste	86
6.3	Diagrama de mobilidade no teste de <i>handoff</i>	87
6.4	Taxa de Comutação de Pacotes com tráfego CBR entre MC e CN	88
6.5	Taxa de Comutação de Pacotes com I.C. 95% (Mac OS X)	89
6.6	Atraso bidirecional (RTT) com tráfego CBR entre MC e CN	91
6.7	Atraso bidirecional com I.C. 95% (Mac OS X)	91

6.8	Variação de atraso com tráfego CBR entre MC e CN	92
6.9	Variação de atraso com I.C. 95% (Mac OS X)	93
6.10	Potência de Sinal Recebido (RSSI) com tráfego CBR entre MC e CN	94
6.11	Potência de Sinal Recebido com I.C. 95% (Mac OS X)	94
6.12	Taxa de retransmissão de pacotes com tráfego CBR entre MC e CN	96
6.13	Avaliação PSNR de vídeo transmitido entre CN e MC	99
6.14	Avaliação PSNR com valores agregados	100
6.15	Mapeamento das medições PSNR para MOS	101
6.16	Avaliação SSIM de vídeo transmitido entre CN e MC	101
6.17	Avaliação SSIM com valores agregados	102
6.18	Mapeamento das medições SSIM para MOS	103
6.19	Captura do quadro 2220 durante o <i>handoff</i> em cada estratégia	104
6.20	Percentual de perda de dados na transmissão de vídeo	104
6.21	Taxa de comutação de pacotes com diferentes valores de T	105
6.22	Taxa de comutação de pacotes com diferentes valores de H	107
6.23	Quantidade de <i>handoffs</i> com diferentes valores de H	108
6.24	Taxa de comutação de pacotes com diferentes velocidades do MC	109
6.25	Utilização de CPU no controlador do OpenWiMesh-Mob	110
6.26	Utilização de memória no controlador do OpenWiMesh-Mob	110
6.27	Comparação da métrica de qualidade de enlace face às medições de RSSI e FRR	111
6.28	Análise dos efeitos da estratégia de duplicação de tráfego	112

LISTA DE TABELAS

6.1	Retransmissão de pacotes (pkts/sec) do AP para MC no MAC OS X	96
6.2	Mapeamento PSNR/MOS e SSIM/MOS	98
6.3	Qualidade de Vídeo PSNR em Transmissão do CN para MC	100
6.4	Qualidade de Vídeo SSIM em Transmissão do CN para MC	102
6.5	Percentual de perda de dados na transmissão de vídeo	104

LISTA DE ABREVIATURAS E SIGLAS

B.A.T.M.A.N.	Better Approach to Mobile Ad-Hoc Networking,	p. 25
AODV	Ad Hoc On-Demand Distance Vector,	p. 25
AP	Access Point,	p. 16
BSSID	Basic Service Set Identification,	p. 49
CBP	Constraint Based Path,	p. 65
CBR	Constant Bit Rate,	p. 87
CN	Correspondent Node,	p. 43
FRR	Frame Retransmission Rate,	p. 73
HC	Hop-Count,	p. 67
HLRB	Highest Link Residual Bandwidth,	p. 67
HLRB-SHC	Highest Link Residual Bandwidth in Same Hop Count,	p. 67
HWMP	Hybrid Wireless Mesh Protocol,	p. 25
IDT	Inter Departure Time,	p. 87
LMA	Local Mobility Anchor,	p. 45
LVAP	Light Virtual Access Points,	p. 49
MAG	Mobility Access Gateway,	p. 45
MANET	Mobile Ad-hoc Network,	p. 23
MC	Mobile Client,	p. 85
MIH	Media Independent Handoff,	p. 46
MIP	Mobile IP,	p. 43
MOS	Mean Opinion Score,	p. 97
MPR	Multipoint Relay,	p. 24
MPTCP	Multipath TCP,	p. 49
NOS	Network Operational System,	p. 27
OGM	Originators Messages,	p. 25
OLSR	Optimized Link State Routing,	p. 24
PS	Packet Size,	p. 87
PSNR	Peak Signal to Noise Ratio,	p. 98
QoE	Quality of Experience,	p. 97
RSSI	Received Signal Strength Indicator,	p. 38

RTT	Round-Trip Time,	p. 41
SDN	Software-Defined Networking,	p. 26
SNR	Signal to Noise Ratio,	p. 38
SSID	Service Set Identifier,	p. 70
SSIM	Structural Similarity,	p. 98
TE	Traffic Engineering,	p. 66
TU	Time Unit,	p. 75
WMN	Wireless Mesh Network,	p. 21

SUMÁRIO

1	Introdução	16
1.1	Estrutura da Dissertação	19
2	Redes Mesh Sem Fio Definidas por Software	21
2.1	Redes <i>mesh</i> sem fio	21
2.1.1	Arquitetura de Redes Mesh	22
2.1.2	Roteamento em redes <i>mesh</i>	24
2.2	Redes Definidas por Software	26
2.3	Integração entre SDN e WMN	29
2.4	Trabalhos Relacionados	32
2.5	Considerações Parciais	35
3	Gerenciamento de Mobilidade	36
3.1	Processo de Handoff	36
3.1.1	Tipos de Handoff	37
3.1.2	Detecção e Inicialização do Handoff	38
3.1.3	Seleção de AP	40
3.2	Mobilidade Intra-Domínio	41
3.3	Mobilidade Inter-Domínio	43
3.3.1	Protocolos MIB e PMIPv6	43
3.3.2	Padrão IEEE 802.21	46
3.3.3	Mobilidade em redes celulares de próxima geração	46

3.4	Trabalhos Relacionados	47
3.5	Considerações Parciais	50
4	Arquitetura do <i>framework</i> OpenWiMesh	51
4.1	Visão Geral do OpenWiMesh	51
4.1.1	Operação do Framework	53
4.2	Canal de Sinalização	57
4.3	Grafo da rede	58
4.4	Algoritmos de Roteamento	60
4.5	Desenvolvimento do OpenWiMesh	61
4.5.1	Diagrama de Classes	61
4.5.2	Pontos de extensão	63
4.5.3	Exemplo de código	65
4.6	Caso de uso: Engenharia de Tráfego em WMN	66
4.7	Considerações Parciais	68
5	Mobilidade Transparente Definida por Software e Controlada pela Rede	69
5.1	Visão geral da estratégia de <i>handoff</i>	70
5.2	Métrica de qualidade de enlaces Wi-Fi	72
5.3	Algoritmo de decisão de <i>handoff</i>	75
5.4	Execução do <i>handoff</i>	77
5.5	Desenvolvimento do Protótipo	80
5.6	Considerações Parciais	82
6	Avaliação	84
6.1	Ambiente de Testes	84
6.2	Metodologia	86
6.3	Avaliação da Qualidade de Serviço	87

6.3.1	Taxa de Comutação de Pacotes	88
6.3.2	Atraso Bidirecional	90
6.3.3	Variação de Atraso	92
6.3.4	Potência de Sinal Wi-Fi	93
6.3.5	Taxa de Retransmissão de Pacotes	95
6.4	Avaliação da Qualidade de Experiência	97
6.4.1	Medições de Qualidade de Vídeo	97
6.4.2	Avaliação do PSNR	99
6.4.3	Avaliação do SSIM	101
6.4.4	Avaliação Visual de Qualidade	103
6.5	Parametrizações e configurações da estratégia	104
6.5.1	Limiar de <i>handoff</i> T	105
6.5.2	Histerese de <i>handoff</i> H	106
6.5.3	Padrão de mobilidade	107
6.5.4	Imagem do controlador – escalabilidade	108
6.5.5	Isolamento de efeitos da estratégia de <i>handoff</i>	109
6.6	Considerações Parciais	112
7	Conclusões e Trabalhos Futuros	113
7.1	Trabalhos Futuros	114
	Referências	116

1 INTRODUÇÃO

As Redes em Malha Sem Fio (AKYILDIZ; WANG, 2009), neste trabalho referenciadas como simplesmente “redes *mesh*”, têm sido reiteradamente apontadas como solução vantajosa para provimento de infraestrutura sem fio, quando comparadas com arquiteturas de redes sem fio convencionais baseadas no padrão IEEE 802.11. Considerando seu baixo custo, simplicidade de implantação, tolerância à falhas e facilidade de expansão da área de convergência, as redes *mesh* sem fio são uma tecnologia ideal para prover uma série de serviços de rede emergentes, como acesso banda larga comunitário à Internet (ESLAMI; KARIMI; KHODADADI, 2014; BENYAMINA; HAFID; GENDREAU, 2012) e balanceamento de tráfego de dados em sistemas celulares 5G (APOSTOLARAS et al., 2014). Outros potenciais cenários de aplicabilidade das redes *mesh* sem fio são em projetos de cobertura de grandes áreas a baixos custos (e.g. zonas de cidades), construção de redes de recuperação de desastres, além de uma vasta gama de aplicações (ESLAMI; KARIMI; KHODADADI, 2014; AKYILDIZ; WANG, 2009).

Adicionalmente, o número de dispositivos portáteis (e.g. *notebooks*, *tablets*, *smartphones*) conectados às redes sem fio tem crescido sobremaneira nos últimos anos. Estudos apontam um aumento exponencial no volume de tráfego de dispositivos móveis nos últimos anos (KOVÁCS et al., 2011; LEE et al., 2016), com previsão de que a quantidade de dispositivos portáteis conectados excederá a quantidade de pessoas no planeta Terra, alcançando, em 2017, uma média de 1.4 dispositivos móveis *per capita* (CISCO, 2015). Esse crescimento, associado à necessidade dos usuários de estarem sempre conectados, torna evidente o requisito de mobilidade eficiente entre os pontos de acesso sem fio (AP, do inglês *Access Point*). O processo de migração de um cliente móvel entre APs é conhecido como *handoff* ou *handover*, ocorrendo quando um nó se distancia do AP atual, aproximando-se de um AP alvo. Assim, para prover mobilidade eficiente, a rede deve manter as conexões dos nós móveis ativas durante o processo de *handoff*.

O processo de *handoff*, idealmente, deve ser transparente para os clientes móveis, dispensando a necessidade de configurações ou protocolos adicionais no dispositivo do usuário. Além disso, o *handoff* deve ocorrer sem interrupção da conectividade de rede. A observância desses

requisitos leva ao que se chama de mobilidade transparente (do inglês, *seamless handoff*), cujo cumprimento é particularmente importante para suporte a aplicações de tempo real e de multimídia interativa. Estas aplicações tornam-se cada vez mais populares (CISCO, 2015), ao passo que impõem requisitos estritos de rede. Em particular, a rede sem fio deve preservar a Qualidade de Serviço (QoS) das conexões, evitando atraso excessivo, variação de atraso e perda de pacotes, além de manter a Qualidade de Experiência (QoE) dos usuários em níveis aceitáveis, sem deteriorar a qualidade de transmissão na percepção dos usuários.

As técnicas tradicionais de mobilidade em redes sem fio, incluindo aquelas disponíveis para dispositivos que adotam o padrão 802.11, baseiam-se no processo de *handoff* monitorado e executado pelo cliente. No entanto, trabalhos anteriores (AMIR et al., 2010; SCHULZ-ZANDER et al., 2014) mostram que essa estratégia implica em atrasos de grandes proporções na rede. Ademais, como o processo de *handoff* é controlado pelo cliente, não existem garantias de que ele seja iniciado no momento correto. Isto pode impactar negativamente no desempenho do cliente, bem como, na rede como um todo, levando à congestionamento do AP, retransmissões excessivas e até perda de pacotes. Outros mecanismos (CROITORU; NICULESCU; RAICIU, 2015; DELY; KASSLER; BAYER, 2011; LAMPROPOULOS; SALKINTZIS; PASSAS, 2008) requerem modificações ou protocolos adicionais no cliente, tornando-se um obstáculo para ampla adoção (SCHULZ-ZANDER et al., 2014).

O roteamento de pacotes em redes *mesh* sem fio baseia-se, comumente, em protocolos de redes *ad-hoc*, como o OLSR (CLAUSEN; JACQUET, 2003) e o B.A.T.M.A.N (JOHNSON; NTLATLAPA; AICHELE, 2008). Esses protocolos são responsáveis por determinar os caminhos que um pacote deve percorrer, através de múltiplos saltos, para prover a conectividade sem fio (BENYAMINA; HAFID; GENDREAU, 2012; AKYILDIZ; WANG, 2009). Além disso, esses protocolos são encarregados de detectar falhas nos enlaces e mobilidade dos nós para, então, reprocessar a tabela de rotas e definir novos caminhos na rede. Em particular, para permitir mobilidade otimizada dos nós, suportando aplicações de tempo real, três requisitos são de importância destacável: detecção da melhor oportunidade de *handoff*; mudança eficiente de ponto de acesso sem interrupção de conectividade; redescoberta e reconfiguração das rotas depois do *handoff*. Os protocolos ora mencionados, todavia, são limitados e de difícil personalização (DELY; KASSLER; BAYER, 2011), fazendo-se necessária uma abordagem que permita maior flexibilidade e programabilidade da rede para consubstanciar os requisitos previamente apresentados.

Tal flexibilidade e facilidade de inovação tem sido alcançada pela adoção do novo paradigma de Redes Definidas por Software (SDN, do inglês *Software-Defined Networking*), atra-

vés de sua implementação mais comum, o protocolo *OpenFlow* (KREUTZ et al., 2015; MCKEOWN et al., 2008). O uso desse paradigma permite tornar as redes mais flexíveis e programáveis, através de uma interface padronizada. Isso é possível devido à migração do plano de controle para um controlador centralizado, de maneira que é possível manter os roteadores e comutadores mais simples, apenas aplicando as regras definidas pelo nó controlador. Amplamente discutido e aceito para redes cabeadas, o paradigma de SDN também tem sido empregado em redes *mesh* sem fio (DELY; KASSLER; BAYER, 2011; COSTANZO et al., 2012; DETTI et al., 2013; SCHULZ-ZANDER et al., 2014), habilitando novas funcionalidades, maior flexibilidade e impulsionando a utilização das redes *mesh*. A integração entre SDN e redes *mesh* sem fio viabiliza, por exemplo, novas estratégias de gerenciamento de mobilidade, roteamento baseado em fluxos, engenharia de tráfego baseada em políticas administrativas etc.

Esta dissertação propõe uma abordagem para melhorar a experiência de mobilidade dos usuários no contexto de redes *mesh* sem fio definidas por software, por meio de uma estratégia de gerenciamento de *handoff* eficiente e monitoramento passivo com métricas de qualidade das associações Wi-Fi. Nessa estratégia, o processo de *handoff* é inteiramente controlado pela rede, o que viabiliza mobilidade otimizada, a partir de melhores decisões do ponto de vista global, e transparente aos dispositivos clientes, mantendo-os livres de qualquer configuração adicional. O monitoramento passivo e as métricas de qualidade de associações Wi-Fi ora propostos, contribuem significativamente para o bom funcionamento da estratégia de *handoff*, habilitando a rede a identificar melhores oportunidades de *handoff* dos clientes e a executá-lo com mínimo impacto à qualidade de transmissão dos enlaces.

A estratégia de *handoff* aqui apresentada foi implementada como prova de conceito através de uma extensão ao *framework* OpenWiMesh (BRITO et al., 2014), que é utilizado no gerenciamento e controle de uma rede *mesh* sem fio com canal de sinalização *in-band* e suporte à técnicas de engenharia de tráfego. Esse *framework* permite a experimentação de novas estratégias de roteamento em redes *mesh* sem fio, disponibilizando ao experimentador, além dos mecanismos de controle dos nós através de OpenFlow, um grafo dirigido da topologia da rede *mesh* sem fio com informações a nível da camada física da vizinhança de cada nó (nível de sinal nas associações, carga atual, relação sinal/ruído etc). Como não requer a existência de um outro canal para sinalização da rede de controle (e.g. múltiplos rádios, virtualização do canal sem fio com múltiplos SSIDs etc), o *framework* possui aplicabilidade prática em uma vasta gama de ambientes.

Foi realizada uma avaliação de desempenho detalhada da proposta, em comparação com uma implantação baseada no protocolo B.A.T.M.A.N. (JOHNSON; NTLATLAPA; AICHELE,

2008) e *handoff* controlado pelo cliente, utilizando o padrão 802.11. Levou-se em consideração métricas de Qualidade de Serviço (QoS) e Qualidade de Experiência (QoE) para avaliar o impacto do *handoff* na transmissão de dados, tanto do ponto de vista da rede quanto da percepção dos usuários, respectivamente. Os experimentos foram conduzidos em um *testbed* 802.11/OpenFlow implantado na Universidade Federal da Bahia (UFBA), a partir de equipamentos convencionais, e levou em consideração diferentes sistemas operacionais de clientes. Resultados mostram que a proposta de gerenciamento de *handoff* melhorou até 70% as métricas de QoE durante a mobilidade de clientes, ao passo que manteve métricas chave de QoS abaixo de limites de restrição para aplicações de tempo real (AMIR et al., 2010).

Dessa maneira, as principais contribuições desta dissertação são:

- Proposta de uma nova métrica de qualidade de enlaces Wi-Fi, obtidas de forma passiva, para identificar oportunidades de *handoff*;
- Proposta de um algoritmo de gerenciamento de *handoff* em redes *mesh* sem fio definidas por software, composto de estratégias de decisão e execução de *handoff*, que, juntas, melhoram a experiência de mobilidade e otimizam a utilização dos recursos da rede, provendo melhor seleção de AP para clientes móveis;
- Desenvolvimento de um protótipo de solução de gerenciamento de mobilidade como prova de conceito das estratégias ora propostas;
- Construção de um *testbed* 802.11/OpenFlow na Universidade Federal da Bahia para experimentação em redes *mesh* sem fio definidas por software;
- Apresentação detalhada de um *framework* para engenharia de tráfego em redes *mesh* sem fio baseado em SDN, chamado OpenWiMesh, desenvolvido pelo autor desta dissertação e outros membros do grupo de pesquisa GRADE¹ e publicamente disponível em (OpenWiMesh, 2013).

1.1 ESTRUTURA DA DISSERTAÇÃO

Esta dissertação está dividida da seguinte forma. O Capítulo 2 discute fundamentos relacionados à redes *mesh* sem fio e SDN, bem como os trabalhos relacionados. No Capítulo 3, são apresentados os conceitos, tipos e principais técnicas de gerenciamento de mobilidade no contexto de redes sem fio, incluindo uma revisão da literatura. Em seguida, a arquitetura e detalhes

¹<<http://grade.dcc.ufba.br>> Último acesso 20/Maio/2016

do *framework* OpenWiMesh são abordados no Capítulo 4. O Capítulo 5 apresenta a proposta de estratégia de mobilidade transparente, abordando os aspectos de monitoramento da qualidade das associações Wi-Fi, detecção de *handoff*, execução do *handoff* e detalhes de sua implementação no *framework* OpenWiMesh. A avaliação da proposta, por sua vez, será apresentada no Capítulo 6, utilizando as métricas de QoS e QoE para mostrar a eficácia da proposta. Por fim, no Capítulo 7, são apresentadas as considerações finais e direcionamentos para trabalhos futuros.

2 REDES MESH SEM FIO DEFINIDAS POR SOFTWARE

As *Redes em Malha Sem Fio* (WMN) têm sido apontadas como solução vantajosa para redes sem fio quando comparadas a outras arquiteturas baseadas no padrão IEEE 802.11. Em complemento, os recentes avanços nas *Redes Definidas por Software* têm viabilizado a programabilidade e a virtualização da rede, propiciando novas estratégias para roteamento e alterações na lógica fundamental das redes *mesh* sem fio, tais como roteamento baseado em fluxos e engenharia de tráfego com políticas administrativas. Dessa maneira, nota-se que a adoção do paradigma SDN no contexto de WMN pode alavancar e potencializar sua flexibilidade, capacidade e aplicabilidade em diversos contextos, como, por exemplo, o gerenciamento de mobilidade. A integração das arquiteturas de SDN e WMN, todavia, impõe alguns desafios relevantes a serem investigados e tratados. Neste capítulo, serão apresentados conceitos e trabalhos relacionados a esse contexto.

Na Seção 2.1, apresenta-se a arquitetura, conceitos e protocolos referentes às redes *mesh* sem fio. Em seguida, o paradigma SDN é discutido na Seção 2.2, levando em consideração seus componentes, papéis dos equipamentos de rede e do controlador, o protocolo OpenFlow e metodologias de implantação. Após apresentar os dois paradigmas individualmente, a Seção 2.3 aborda os desafios e estratégias para integração entre SDN e redes *mesh* sem fio. O capítulo encerra-se na Seção 2.4, com um levantamento de trabalhos relacionados.

2.1 REDES MESH SEM FIO

Uma rede *mesh* sem fio (WMN, do inglês *Wireless Mesh Network*) é uma rede sem fio de múltiplos saltos, na qual roteadores sem fio encaminham tráfego em benefício de outros roteadores ou de clientes (AKYILDIZ; WANG, 2009). Os roteadores *mesh* formam, destarte, uma malha de acesso sem fio para os clientes *mesh*. Uma das principais vantagens da WMN é sua arquitetura descentralizada, que incorpora maior resiliência e escalabilidade à rede. Outras

vantagens dessa tecnologia são o baixo custo, facilidade de instalação e manutenção, e robustez. WMNs são usadas em redes residenciais, cenários de emergência e recuperação de desastre, extensão para redes corporativas, em redes metropolitanas, dentre outras (ESLAMI; KARIMI; KHODADADI, 2014; AKYILDIZ; WANG, 2009). Mais recentemente, as WMNs são também utilizadas para prover serviços de rede emergentes, como acesso banda larga comunitário à Internet (BENYAMINA; HAFID; GENDREAU, 2012) e escoamento de tráfego de dados em sistemas celulares 5G (APOSTOLARAS et al., 2014).

2.1.1 ARQUITETURA DE REDES MESH

A implantação mais comum de redes sem fio envolve Pontos de Acesso (AP), comumente dispostos de forma a fornecer maior área de cobertura através da sobreposição de sinal, onde cada AP tem uma conexão cabeada até a rede de distribuição (também conhecida como *backhaul*). Em contrapartida, as WMNs somente requerem que alguns nós tenham conexões cabeadas ao *backhaul*, permitindo que todos os outros encaminhem pacotes por meio de múltiplos saltos.

Segundo Akyildiz e Wang (2009), uma WMN é composta por nós *gateway* (MGW), roteadores *mesh* (MR), pontos de acesso *mesh* (MAP) e clientes móveis (MC). Os nós *gateway* interconectam a WMN com outras redes locais e com a Internet, frequentemente através de interfaces cabeadas. Os APs *mesh* são responsáveis pela conectividade dos clientes, baseada majoritariamente no padrão IEEE 802.11 (CROW et al., 1997), no modo *infraestruturado*. Embora o padrão 802.11 seja a tecnologia mais utilizada para acesso dos clientes, outras tecnologias de redes de celulares, redes de sensores, entre outras, podem ser utilizadas, ampliando a aplicabilidade das redes *mesh* sem fio (ESLAMI; KARIMI; KHODADADI, 2014). Idealmente, a rede *mesh* sem fio deve suportar clientes não modificados (AMIR et al., 2010), provendo conectividade de forma simplificada e ágil. A rede *mesh* sem fio deve fornecer também suporte a clientes móveis, gerenciando suas conexões, sem interrupção de conectividade.

Os MRs são responsáveis pelo encaminhamento de pacotes entre os nós da rede, conforme a estratégia de roteamento definida. No nível físico, a comunicação pode ser constituída por variantes dos padrões IEEE 802.11 (WLAN/Wi-Fi), como o 802.11 em modo *ad-hoc*, ou ainda IEEE 802.15 (WPAN), um padrão para comunicação local em redes sem fio pessoais, e IEEE 802.16 (WiMAX), um padrão para comunicação sem fio em redes metropolitanas. Exceto os nós clientes, os outros tipos de nó podem acumular múltiplos papéis na rede *mesh* sem fio. Esse é o caso de roteadores *mesh*, que podem também operar como MAP ou MGW. É importante respeitar, todavia, a separação da rede de acesso e o núcleo da rede *mesh* sem fio, seja por redes

virtuais ou através de múltiplas interfaces físicas.

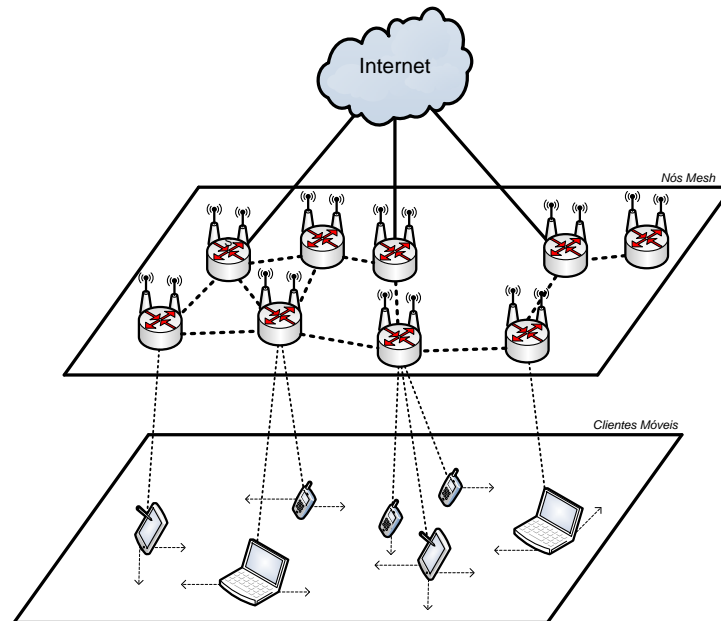


Figura 2.1: Rede Mesh Sem Fio (adaptado de Amir et al. (2010)).

A Figura 2.1 retrata uma visão geral da rede *mesh* sem fio. Enquanto os nós *mesh* (roteadores e APs) são habitualmente estacionários, clientes móveis conectados à WMN podem movimentar-se ao longo da área de cobertura sem perda de conectividade. Esta é uma das principais diferenças entre redes *mesh* sem fio e Redes Móveis Ad-hoc (MANET, do inglês *Mobile Ad-hoc Network*), onde todos os nós (clientes e nós *mesh*) podem movimentar-se e participar do protocolo de roteamento (AMIR et al., 2010).

Quando um cliente móvel desloca-se na rede *mesh* sem fio, afastando-se do ponto de acesso ao qual estava conectado e aproximando-se de outro, ele troca sua conectividade para esse ponto de acesso mais próximo. Essa mudança de conectividade envolve uma transição (*handoff*) seguida, em alguns casos, por uma atualização nas rotas relacionadas ao nó móvel nos múltiplos saltos da rede *mesh* sem fio. Todo esse processo é necessário para prover mobilidade. Idealmente, ele deve ser transparente aos clientes móveis, não implicando em interrupção das conexões previamente estabelecidas ou prejuízo à qualidade das transmissões.

A seguir, as estratégias tradicionais de roteamento em redes *mesh* sem fio serão apresentadas e discutidas, assim como algumas das suas limitações.

2.1.2 ROTEAMENTO EM REDES *MESH*

No que tange ao roteamento em redes *mesh* sem fio, diferentes estratégias têm sido investigadas. Algumas estratégias baseiam-se nos protocolos de redes *ad-hoc*, como o protocolo AODV (PERKINS; BELDING-ROYER; DAS, 2003) ou o protocolo B.A.T.M.A.N. (JOHNSON; NTLATLAPA; AICHELE, 2008), e outras são influenciadas por protocolos clássicos de LAN, como o OLSR (CLAUSEN; JACQUET, 2003). Em Setembro de 2011, o IEEE aprovou uma atualização ao padrão 802.11: o complemento 802.11s (IEEE, 2011b), que especifica melhorias e estratégias para suportar redes *mesh* sem fio. Nesse complemento, o roteamento é baseado na camada de enlace e usa, como mecanismo de seleção de caminhos, o protocolo híbrido HWMP (IEEE, 2011b).

As estratégias de roteamento em redes *mesh* sem fio são classificadas geralmente em três categorias (AKYILDIZ; WANG, 2009): protocolos proativos, reativos e híbridos. Nos protocolos proativos, computa-se as rotas para todos os nós durante a fase de inicialização do protocolo. Já nos protocolos reativos, as rotas são calculadas na medida em que são requisitadas pelas aplicações. Os protocolos reativos possuem a vantagem de não ocupar desnecessariamente a rede com tráfego de controle, ao passo que os protocolos pró-ativos geralmente apresentam menor atraso no estabelecimento de novos fluxos na rede, uma vez que os caminhos já estão definidos. Os protocolos híbridos tentam combinar as vantagens dos protocolos reativos e pró-ativos e adaptar seu funcionamento a diferentes cenários.

A seguir serão apresentados os dois principais protocolos referenciados na literatura para roteamento em redes *mesh* sem fio (REINERI; CASETTI; CHIASSERINI, 2009; AKYILDIZ; WANG, 2009; SANCHEZ-IBORRA; CANO, 2014), a saber o B.A.T.M.A.N. e OLSR, em conjunto com o HWMP, que é o protocolo base para o padrão 802.11s.

OLSR – O protocolo OLSR (do inglês *Optimized Link State Routing*) (JACQUET et al., 2001) é um protocolo de roteamento pró-ativo, baseado em algoritmo otimizado de estado do enlace e amplamente utilizado em redes *mesh* sem fio. A principal vantagem do OLSR em relação aos algoritmos de estado de enlace clássicos é a existência de “retransmissores multiponto” (MPR, do inglês *Multipoint Relay*). Os MPRs são nós especialmente selecionados para fazer a transmissão das informações de topologia, evitando que o processo de inundação seja realizado a partir de todos os nós. Esta técnica minimiza a contenção e o número de mensagens de controle necessárias para estabelecimento da tabela de roteamento. O protocolo faz uso de mensagens HELLO periódicas para descoberta de vizinhos e sinalização da seleção de MPRs. Além disso, utiliza também mensagens TC (controle de topologia) para troca de informações de topologia entre nós. Assim, todos os nós conhecem a topologia e calculam localmente as

melhores rotas para os demais nós.

B.A.T.M.A.N. – O protocolo B.A.T.M.A.N. (do inglês *Better Approach to Mobile Ad-Hoc Networking*) (JOHNSON; NTLATLAPA; AICHELE, 2008) também faz uso de uma estratégia pró-ativa, porém a técnica de distribuição da topologia e seleção de rotas é melhor alinhada com prováveis limitações de hardware dos nós *mesh*, como pouca memória RAM ou baixo poder de processamento. No B.A.T.M.A.N., os nós não conhecem toda a topologia da rede. Ao invés disso, eles conhecem apenas o vizinho que possui a melhor rota para determinado destino. O algoritmo de seleção de rotas baseia-se no fato de que os melhores enlaces entregarão as mensagens de forma mais rápida e confiável. Assim, a seleção de melhores rotas se dá de forma natural, sem necessidade de um cálculo baseado na topologia em cada nó. De forma geral, cada nó envia, por difusão, *mensagens de originadores* (OGM, do inglês *Originators Messages*), informando aos vizinhos sobre a sua existência. Ao receber uma mensagem OGM, o nó retransmite essa mensagem para seus vizinhos. Naturalmente, espera-se que as mensagens encaminhadas por enlaces de maior qualidade cheguem primeiro; por isso, a primeira mensagem recebida torna-se o melhor caminho para o originador daquela mensagem e as demais são descartadas. Apenas a primeira OGM recebida para determinado destino (melhor caminho) é retransmitida. Existe, portanto, uma inundação da rede com OGMs. Ademais, uma checagem bidirecional de cada enlace é feita a fim de verificar se o enlace detectado pode ser usado em ambas as direções. As mensagens OGM são usadas pelo B.A.T.M.A.N. para detectar mudanças na rede e na qualidade dos enlaces. Mais adiante, na Seção 3.2, será apresentado como o B.A.T.M.A.N. realiza o gerenciamento de clientes móveis na WMN.

HWMP – Usado como base no padrão IEEE 802.11s, o protocolo HWMP (do inglês *Hybrid Wireless Mesh Protocol*) foi baseado no protocolo AODV (do inglês *Ad Hoc On-Demand Distance Vector*) (PERKINS; BELDING-ROYER; DAS, 2003), todavia, adaptado para operar na camada de enlace. Esse protocolo fornece dois modos de operação: i) roteamento sob-demanda e ii) roteamento proativo (IEEE, 2011b). O roteamento proativo é utilizado para que todos os nós *mesh* tenham rotas para alcançar o nó *gateway* (ou múltiplos nós *gateway*). O nó *gateway* é configurado como “raiz” e, então, inicia a propagação de mensagens de roteamento para estabelecer e manter os caminhos entre ele e os nós *mesh* da rede, construindo uma árvore de caminhos. Por outro lado, a comunicação direta entre os nós *mesh* é realizada através da função de roteamento sob-demanda, utilizando o protocolo AODV. O protocolo HWMP utiliza uma métrica chamada *airtime* para verificar a qualidade das associações entre os nós. A métrica *airtime* indica o custo total de transmissão de um enlace, levando em consideração a taxa de transmissão do enlace, sobrecarga e taxa de erros, todos calculados de forma ativa a partir de um quadro de teste.

Todos os protocolos supracitados possuem limitações quanto à extensão e inovação na estratégia de roteamento (DELY; KASSLER; BAYER, 2011). Por exemplo, roteamento baseado em fluxos, onde pacotes com mesmo destino podem ser encaminhados por caminhos diferentes, ou utilização de métricas administrativas na política de roteamento são técnicas difíceis de serem implementadas, embora possam apresentar casos de uso interessantes. Outra deficiência dos protocolos tradicionais concerne na falta de interação entre as múltiplas camadas da pilha de rede. Nesse aspecto, o gerenciamento de mobilidade dos clientes *mesh* poderia valer-se dessa interação multi-camada para executar de forma mais eficiente (AKYILDIZ; WANG, 2009).

A inflexibilidade de alteração no processo de roteamento e encaminhamento de pacotes, acima apresentada, tem sido reconhecida pela comunidade científica e abordada no novo paradigma das Redes Definidas por Software, conforme será discutido adiante.

2.2 REDES DEFINIDAS POR SOFTWARE

A infraestrutura de rede atual, que, no geral, consiste de roteadores, comutadores etc., tem sua lógica de funcionamento definida pelos fabricantes dos equipamentos, com base em padrões e protocolos especificados por entidades como IETF e IEEE, ou ainda em padrões proprietários, especificados pelos próprios fabricantes. Nesse modelo, novos padrões de rede podem demorar muito tempo até serem amplamente suportados e adotados. A transição entre IPv4 e IPv6, por exemplo, já se prolonga por mais de dez anos e ainda com baixa adesão. Esse processo de padronização e a adoção pelos fabricantes, em conjunto com a dificuldade de inovação das redes IP atuais, podem acarretar um tempo de cinco anos ou mais para um novo protocolo de roteamento ser totalmente projetado, avaliado e adotado (KREUTZ et al., 2015). Esse modelo de plataforma dificulta o ajuste ou alteração do comportamento da rede, sendo obstáculo para inovação e para customizações avançadas na lógica de funcionamento dos equipamentos. Com objetivo de alcançar uma plataforma aberta e um padrão para desenvolvimento da rede, foi proposto o paradigma Redes Definidas por Software (SDN, do inglês *Software-Defined Networking*) (ONF, 2012b; MCKEOWN, 2009).

O paradigma SDN propõe a separação entre o plano de encaminhamento e o plano de controle da rede, permitindo que aplicações e serviços de rede possam ser diretamente programados em um controlador remoto (KREUTZ et al., 2015; MCKEOWN, 2009). Esse controlador remoto define o comportamento da rede através de aplicações programadas pelo administrador, que atuam no plano de controle da rede. O controlador, dessa forma, atua como um sistema operacional da rede (KREUTZ et al., 2015), provendo abstrações de suas funções e serviços.

O plano de encaminhamento permanece nos equipamentos de rede, chamados de comutadores SDN, que simplesmente aplicam as regras de encaminhamento, definidas pelo plano de controle via controlador remoto. Outro aspecto intrínseco ao paradigma SDN é a existência de uma interface de comunicação padronizada entre o controlador e o comutador SDN (ONF, 2012b). De maneira semelhante, propõe-se a adoção de uma abstração de programação bem definida para comunicação entre o controlador e as diversas aplicações de rede que podem ser desenvolvidas (KREUTZ et al., 2015).

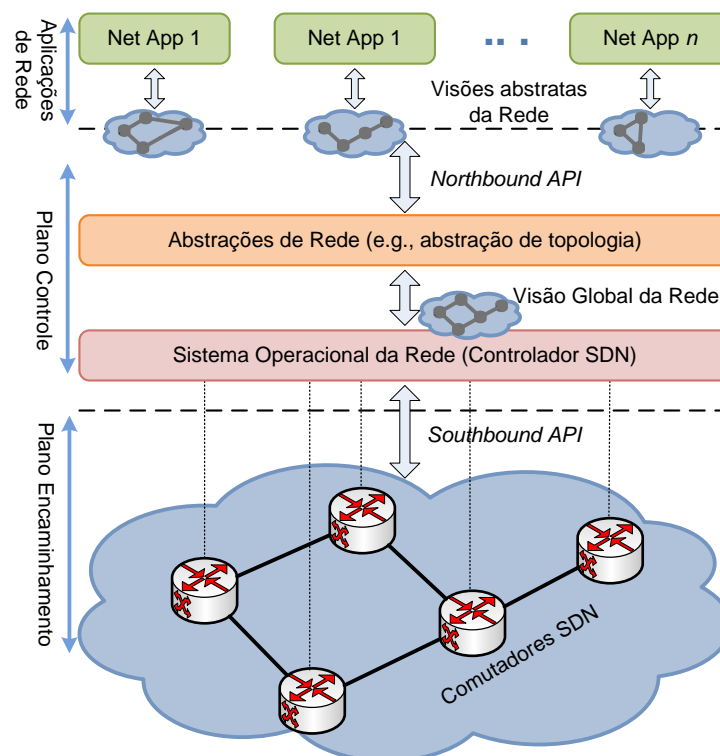


Figura 2.2: Arquitetura SDN e suas abstrações fundamentais (adaptado de Kreutz et al. (2015)).

A Figura 2.2 retrata a arquitetura previamente discutida do paradigma SDN, seus conceitos e componentes. Nessa figura é possível notar o conceito de *Southbound API*, que é a Interface de Programação de Aplicações utilizada para comunicação entre o controlador e os comutadores SDN, bem como o conceito de *Northbound API*, que fornece a interface de interação entre o controlador e as aplicações de rede. Na Figura 2.2, constata-se, ainda, o papel de Sistema Operacional da Rede (NOS, do inglês *Network Operational System*) desempenhado pelo componente Controlador SDN, que provê abstrações de rede (e.g., grafo da topologia), visão global, controle e monitoramento etc. As Aplicações de Rede, por sua vez, adicionam novas funções e funcionalidades à rede, como gerenciamento de mobilidade, engenharia de tráfego, controle de admissão, dentre outras (KREUTZ et al., 2015).

Uma das propostas mais adotadas para o paradigma SDN é o protocolo OpenFlow (MC-KEOWN et al., 2008; ONF, 2012a), que define uma API de comunicação entre o controlador e o comutador SDN (*Southbound API*). A arquitetura OpenFlow é composta por: i) comutadores OpenFlow, que são responsáveis pelo encaminhamento dos pacotes com base em regras pré-definidas; e ii) o controlador, que gerencia os comutadores OpenFlow, definindo suas regras de funcionamento. Cada comutador OpenFlow contém uma ou mais tabelas de fluxos (regras), que são usadas no casamento e encaminhamento de pacotes, e um canal seguro, que conecta o comutador ao controlador. Cada tabela de fluxos contém entradas que são usadas na comparação e encaminhamento de pacotes. Cada entrada da tabela de fluxos consiste de campos de casamento, contadores e um conjunto de instruções a serem aplicadas em pacotes correspondentes. Fica a cargo do controlador adicionar, remover ou modificar entradas nessa tabela de fluxos, bem como consultar estatísticas (ONF, 2012a).

O canal seguro é usado pelo controlador para gerenciar os comutadores, através do protocolo OpenFlow. O canal seguro pode valer-se de uma rede adicional, dedicada ao controle, caracterizando controle *out-of-band*, ou utilizar a mesma rede de transmissão dos dados, caracterizando controle *in-band*. Quando implementado no modo *out-of-band*, o comutador SDN precisa de uma conexão IP operacional independente do controle SDN, comumente implementada por protocolos legados. Já no modo *in-band*, o comutador SDN deve reconhecer as mensagens de gerenciamento e controle através do *datapath* SDN e o controlador SDN deve implementar uma estratégia de controle que seja apta a orquestrar o tráfego de dados e o próprio tráfego de controle. Se, por um lado, o controle *out-of-band* é mais simples de implantar e oferece isolamento para o encaminhamento SDN, por outro, o modo *in-band* oferece uso otimizado dos recursos, dispensando a necessidade de enlaces adicionais e protocolos legados para estabelecimento do canal seguro, além de possibilitar a aplicação da mesma política de roteamento do controlador no tráfego de controle (e.g. provendo engenharia de todo tráfego). A abordagem *out-of-band* tem sido adotada com maior frequência nas implementações SDN atuais, porém em alguns cenários não é possível, ou viável, criar uma rede *out-of-band* para comunicação com o plano de controle (KOBAYASHI et al., 2014), como é o caso de algumas redes Wi-Fi com OpenFlow ou equipamentos que apenas funcionam com controle SDN.

Ademais, deve-se configurar no comutador OpenFlow o conjunto de portas nas quais o controle SDN será aplicado (também conhecido como *datapath*). Um comutador OpenFlow pode operar com todas as portas pertencendo ao *datapath* OpenFlow, com algumas portas operando com protocolos legados e outras operando com OpenFlow, ou ainda com portas híbridas cujo funcionamento, ora é definido pelo controlador, ora funciona com protocolos legados¹. As por-

¹Poucos fabricantes suportam esse modo de operação, implementando-o, geralmente, através de separação

tas que fazem parte do *datapath* OpenFlow são chamadas de portas SDN, incluindo as portas híbridas. Dessa maneira, é possível implementar diferentes técnicas de encaminhamento, de forma flexível e eficiente. É possível também uma coexistência com protocolos de roteamento legados, oferecendo a capacidade de virtualização de redes (MCKEOWN et al., 2008).

Quando um comutador OpenFlow recebe um pacote em uma porta SDN, ele consulta a tabela de fluxos para decidir o que fazer com o pacote (e.g. encaminhá-lo para uma porta, descartá-lo, encaminhá-lo por inundação etc.). Caso não haja correspondência previamente estabelecida, o pacote (ou apenas seus cabeçalhos) é enviado para o controlador. O controlador, então, processa o pacote, por meio de aplicações e algoritmos que ele executa, e ensina o comutador OpenFlow a tratar os próximos pacotes daquele fluxo, inserindo uma nova entrada na tabela de fluxos do comutador. É possível definir regras iniciais no comutador OpenFlow, a fim de que ele possa encaminhar determinado tráfego sem depender do controlador.

O paradigma SDN e o protocolo OpenFlow têm sido amplamente aceitos e aplicados em redes cabeadas. Alguns trabalhos recentes procuram aplicá-los também a redes sem fio. Todavia, questões importantes ainda permanecem em aberto nesse contexto: como combinar a arquitetura distribuída das redes *mesh* sem fio com o modelo centralizado de SDN e OpenFlow? Como prover mobilidade eficiente na WMN? Como implementar a rede de sinalização OpenFlow *in-band* na rede de dados, eliminando a necessidade de duas interfaces de rede em cada nó sem fio? Na próxima seção, serão apresentados alguns trabalhos correlatos a esses desafios.

2.3 INTEGRAÇÃO ENTRE SDN E WMN

A arquitetura centralizada do paradigma SDN, com visão global da rede, permite o desenvolvimento de serviços de rede de alto nível no controlador, tais como: monitoramento e gerenciamento de mobilidade, controle de admissão, engenharia de tráfego etc. Estes serviços aprimoram os benefícios das redes *mesh* e podem impulsionar sua utilização. Não obstante, ao integrar SDN com redes *mesh* sem fio, desafios relacionados à escalabilidade e desempenho podem surgir (CHUNG et al., 2012), principalmente devido à arquitetura distribuída das redes *mesh* sem fio e ao aumento na quantidade de nós, face ao modelo centralizado desse novo paradigma. No paradigma SDN, as decisões de encaminhamento, que antes eram definidas por protocolos distribuídos de conhecimento da topologia e consenso, agora são firmadas pelo controlador central e repassadas aos comutadores SDN. Cria-se, portanto, uma forte dependência do controlador e sua escolha pode influenciar diretamente no funcionamento da rede. Em par-

de VLANs, onde algumas VLANs pré-definidas operam em modo SDN e todas as demais seguem o fluxo de encaminhamento padrão do equipamento.

ticular, a escolha do controlador deve levar em consideração sua localização e capacidade de processamento, fatores intimamente relacionados com a latência e escalabilidade da operação do plano de controle.

Chung et al. (2012) apresenta algumas considerações relativas à implantação de SDN sobre redes *mesh* sem fio. Uma questão importante a ser considerada é a confiabilidade da rede *mesh* sem fio, devido ao fato do controlador SDN se tornar um ponto único de falhas. Em particular, caso a ligação entre o comutador e o controlador seja interrompida, novos fluxos não serão mais instalados e, portanto, novas aplicações não serão aptas a trafegar na rede. Essa limitação ocorre devido a maneira como o OpenFlow funciona na sua configuração convencional, onde as regras de encaminhamento são configuradas sob-demanda de forma reativa. Outra consequência observada é o aumento na latência no primeiro pacote de um novo fluxo, que dispara a comunicação com o controlador para instalação das regras OpenFlow, chamado de *slow-path*, enquanto os demais pacotes daquele fluxo, cujas regras de encaminhamento já estão configuradas, conservam a latência particular do enlace, chamado de *fast-path*. A sobrecarga do *slow-path* pode ser especialmente prejudicial nas redes *mesh* sem fio devido à limitações de *buffer* dos MRs e MAPs, causando, por exemplo, propagação de erros em aplicações de transmissão de vídeo (MWELA, 2010).

Heller, Sherwood e McKeown (2012) investigam o impacto da quantidade e localização dos controladores em uma topologia qualquer, considerando redes cabeadas. Os autores avaliam o comportamento da rede ao aumentar a quantidade de controladores e ao acomodá-los aleatoriamente na rede, em comparação com a instalação em locais centrais. Os autores concluem que a quantidade de controladores, naquele contexto, tem pouca relação com o desempenho da rede e que uma escolha planejada do controlador (e.g. com centralidade na rede) oferece maior desempenho. Em uma rede *mesh* sem fio, devido às características de variabilidade dos enlaces sem fio, a quantidade de controladores tem impacto na latência do *slow-path* (ARAUJO, 2015). Além disso, a escolha do controlador, em termos de localização e capacidade de processamento e memória, pode afetar o desempenho geral das funções de monitoramento e controle. Em particular, o controlador executa os algoritmos de busca de caminhos, gerenciamento de mobilidade, controle de admissão, armazena informações de topologia de rede e controle de aplicações ativas, entre outras funções, por isso, requisitos mínimos de memória e processamento devem ser satisfeitos.

Uma abordagem para resolver os problemas de escalabilidade do controlador SDN em WMN é a utilização de múltiplos controladores, segmentando a rede em domínios e hierarquizando a tomada de decisão de encaminhamento intra e interdomínio, conforme proposto em

(PHAN; THOAI; KUONEN, 2013) e (ARAUJO, 2015). Neste trabalho, devido ao tamanho reduzido do *testbed*, foi adotado apenas um controlador SDN, sem prejuízo ao desempenho da rede, conforme será apresentado no Capítulo 6.

Outro aspecto que merece atenção é o mecanismo de implantação do canal seguro de comunicação entre o comutador e o controlador, que pode ser *out-of-band* ou *in-band*. Conforme previamente discutido na Seção 2.2, o modo *out-of-band* requer uma rede adicional, dedicada ao controle, ao passo que no modo *in-band* o tráfego de controle compartilha o mesmo *datapath* do tráfego de dados. No contexto de WMNs, o uso de controle *out-of-band* demanda roteadores sem fio capazes de configurar duas WMNs, seja por meio de interfaces físicas (e.g. múltiplos rádios) ou utilizando interfaces virtuais (e.g. múltiplos SSIDs²). O modo *in-band*, por consequência, aparenta ser mais apropriado para WMNs, pois ele não necessita de uma segunda rede *mesh* sem fio, diminuindo os custos de implantação da rede. Contudo, existem desafios técnicos que tornam sua adoção mais complexa. O principal desafio é o fato de que os MRs precisam de conectividade ao controlador para obter as regras de encaminhamento SDN, enquanto que essas regras são necessárias para o próprio encaminhamento *in-band*³. Nesta dissertação será apresentada uma proposta para viabilizar a adoção de canais de sinalização *in-band* em WMN com SDN (Seção 4).

No que diz respeito ao mecanismo de controle SDN para redes WMN, o protocolo Open-Flow, mais amplamente utilizado, possui limitações para orquestração do ambiente sem fio (SCHULZ-ZANDER et al., 2014; NASCIMENTO et al., 2014; YAP et al., 2010), não endereçando particularidades dos protocolos e redes Wi-Fi, por exemplo. Estas limitações podem ser observadas pela falta de funções para mitigação de interferência de sinal, gerenciamento de mobilidade, configuração de canais, medição de potência de sinal, contadores de estatísticas de retransmissão e tráfego por associação etc.

Por fim, considerando as limitações de hardware dos MRs, as funções de descoberta de topologia, descoberta de vizinhos, monitoramento da qualidade dos enlaces e comunicação dos comutadores SDN com o controlador devem ser adaptadas e otimizadas para diminuir a sobrecarga do tráfego de controle. Adicionalmente, a camada MAC do IEEE 802.11 utiliza mensagens de confirmação (ACK) para garantir entrega de quadros. Por isso, o endereço MAC do transmissor precisa ser modificado à medida que o pacote é transmitido pelos múltiplos saltos

²SSID, do inglês *Service Set Identifier*, é o nome dado a cada rede que um rádio Wi-Fi possui. Cada rádio Wi-Fi possui sua especificação de quantidade de SSIDs suportados, bem como as configurações de modos de operação. É comum, por exemplo, que os fabricantes suportem múltiplos SSIDs em modo AP simultaneamente; por outro lado, eles suportam apenas um SSID em modo *ad-hoc*. Pode-se associar contextos de operação diferentes para cada SSID (e.g. VLANs).

³Configurar remotamente uma rede, através da rede sendo configurada, é semelhante ao enigma *o ovo e a galinha* – um dilema de relação causa e consequência.

da rede. Do contrário, o emissor intermediário pode gerar excessivas retransmissões de quadros, uma vez que ele não recebeu corretamente o ACK do próximo salto (ACK enviado para o nó errado). Em complemento, a tabela de regras SDN (tabela de fluxos) deve manter-se simples e otimizada para evitar alto consumo de processamento e alta latência no encaminhamento de pacotes.

Com base nessas considerações, foi desenvolvido um *framework* SDN para WMN que supera as restrições apresentadas e viabiliza um controle eficiente e flexível na WMN (conforme Seção 4). Outros trabalhos relacionados são também apresentados na seção seguinte.

2.4 TRABALHOS RELACIONADOS

O trabalho de Dely, Kessler e Bayer (2011) foi o primeiro a integrar SDN/OpenFlow e WMNs. Sua principal contribuição é a proposta de uma arquitetura para controle e monitoramento de uma estratégia de roteamento baseado em OpenFlow/SDN em redes *mesh*, com estudo de caso em gerenciamento de mobilidade e balanceamento de carga. A arquitetura considera duas redes sem fio virtuais, implementadas através da funcionalidade de múltiplos SSIDs, sendo a primeira rede usada exclusivamente para a sinalização de controle do OpenFlow, e a segunda usada para os dados em si. A WMN usada para sinalização OpenFlow foi baseada no padrão 802.11 e no protocolo OSLR. Compreendem o núcleo da rede nessa arquitetura: um Servidor de Monitoramento e Controle e um controlador NOX (GUDE et al., 2008). O Servidor de Monitoramento e Controle é responsável por construir a topologia da rede, a partir de informações consultadas nos clientes e roteadores *mesh*, além de uma base de associações e oportunidades de associações. A topologia é mapeada em um grafo de rede, podendo, inclusive, conter métricas de qualidade dos enlaces. A base de associações e oportunidades de associações pode ser usada para prover mobilidade dos nós, sendo construída a partir de informações do sistema sem fio dos roteadores *mesh*. Já o NOX é usado para executar as tarefas de roteamento na rede, sendo alimentado e notificado pelo Servidor de Monitoramento e Controle. A arquitetura foi validada com o desenvolvimento e avaliação de uma solução para mobilidade em um *testbed* real chamado KAUMesh.

Esta dissertação diferencia-se da proposta citada (DELY; KASSLER; BAYER, 2011) quanto a abordagem de gerenciamento de *handoff*, que não depende de mudanças no dispositivo cliente, na combinação de métricas de qualidade do enlace Wi-Fi, que leva em consideração um monitoramento passivo, e no uso de sinalização *in-band*.

De maneira similar, Detti et al. (2013) apresenta uma Rede Mesh Definida por Software

(wmSDN) mostrando sua aplicabilidade para cenários de engenharia de tráfego, nos quais o controlador balanceia o tráfego de saída entre diversos roteadores de Internet. Assim como no trabalho anterior, a proposta wmSDN também faz uso de um enlace adicional e protocolos legados, como *OLSR*, para o tráfego de controle e gerenciamento da solução. Nesta dissertação, não foi necessário a utilização de enlaces adicionais para o tráfego de controle, cuja sinalização ocorre de maneira *in-band*.

Já em (CHUNG et al., 2012), além de apresentar desafios da integração de SDN e WMN, os autores realizam experimentações em cenários simples para avaliar o funcionamento da WMN com OpenFlow. No cenário *in-band*, os resultados dos testes reforçam a complexidade de funcionamento da rede nesse modo de operação, apresentando, no geral, um desempenho de baixa qualidade. Fez-se necessário, inclusive, a criação de entradas estáticas na tabela ARP dos equipamentos para que a rede tivesse estabilidade de funcionamento. Tal problema ocorre devido à possibilidade de criação de *loops* face ao roteamento multi-salto e ao fato dos comutadores OpenFlow operarem em modo promíscuo⁴. Já no cenário com controle *out-of-band* e uma rede 802.11s para o canal seguro com o controlador, o desempenho foi satisfatório atingindo índices de vazão UDP próximos ao cenário com protocolos legados. O uso de uma WMN secundária, todavia, resulta em uso ineficiente do espectro e dos recursos da rede Wi-Fi. Superando os desafios encontrados no uso de sinalização *in-band* discutidos em (CHUNG et al., 2012), esta dissertação apresenta uma estratégia eficiente para controle da rede *in-band*.

Yap et al. (2010) propuseram uma estratégia para orquestração de redes Wi-Fi através de uma plataforma chamada *OpenFlow Wireless*, que viabiliza pesquisas experimentais e ambientes reais empregando virtualização em redes móveis sem fio. Os autores integram diversas soluções nessa plataforma: *OpenFlow*, para separar o plano de controle do plano de encaminhamento através de sua API aberta; *FlowVisor*, para criação de fatias de rede com isolamento entre elas, permitindo, assim, múltiplos experimentos simultâneos, bem como a coexistência com rede de produção; *SNMP*, para intermediar a configuração dos elementos do *datapath* e obter estatísticas do meio sem fio nas diversas fatias da rede. A proposta foi implantada e avaliada no *campus* da Universidade de Stanford, em um projeto conhecido como OpenRoads (YAP et al., 2009). O projeto OpenRoads divide a rede em fatias, através do uso de múltiplos SSIDs.

Naquele trabalho, a rede OpenRoads foi composta por diferentes tecnologias sem fio, especialmente Wi-Fi e WiMAX. Os pesquisadores instalaram oitenta e cinco APs Wi-Fi e duas estações base WiMAX no *campus* de Stanford, interconectados ao controlador OpenFlow NOX através de comutadores Ethernet em um canal de sinalização *out-of-band*. Como estudo de caso

⁴No modo promíscuo, o Kernel remove os filtros de camada de enlace e a interface passa a receber todo tráfego que é transmitido na rede e está a seu alcance.

da proposta, foram implementadas duas estratégias de gerenciamento de mobilidade. Graças ao arcabouço fornecido pelo OpenRoads, as estratégias de gerenciamento de mobilidade foram implementadas de forma simples e rápida (YAP et al., 2010).

Em (NASCIMENTO et al., 2014), os autores propõem uma extensão do protocolo OpenFlow para adicionar funcionalidades específicas do ambiente de redes sem fio, bem como uma arquitetura dos nós para controle *out-of-band*. A principal mudança no protocolo OpenFlow foi a adição de novas mensagens para trazer informações do meio físico em interfaces sem fio, como o modo de operação da interface, frequência e canal configurados, lista de associações sem fio (vizinhos), níveis de sinal/ruído, potência de sinal recebido, entre outras. A respeito da arquitetura proposta, foram utilizadas duas interfaces físicas para estabelecer redes virtuais para controle e dados: duas interfaces virtuais operando em modo AP, para conexão dos clientes de produção e clientes de experimentação; uma interface virtual configurada em modo *mesh*, através do protocolo 802.11s; uma interface operando em modo *ad-hoc*, usada no encaminhamento do tráfego de dados da rede (*backhaul*). A proposta foi avaliada de forma simples em um *test-bed* com três nós, demonstrando a sua capacidade de suprir algumas demandas para integração de SDN em WMN.

Embora a proposta do OpenRoads (YAP et al., 2010) e a proposta (NASCIMENTO et al., 2014) não tenham como alvo principal prover suporte a mobilidade, algumas semelhanças podem ser observadas entre aqueles trabalhos e esta dissertação, principalmente pelo uso de metodologia complementar ao OpenFlow a fim de obter estatísticas do meio sem fio. Em particular, nesta dissertação fez-se uso de um componente no roteador *mesh* para obter as informações do meio sem fio e alimentar a topologia da rede, conforme será detalhado no Capítulo 4.

O trabalho (SCHULZ-ZANDER et al., 2014) propõe uma solução para orquestração de redes Wi-Fi com SDN, chamada Odin. Adota-se uma arquitetura de rede Wi-Fi convencional, em que os APs possuem conexão *out-of-band* cabeada até o *backhaul*. Para superar as limitações do protocolo OpenFlow em redes Wi-Fi, os autores utilizam um protocolo customizado adicional para obtenção de estatísticas do meio sem fio, configurar os APs, entre outras funções. Entre as contribuições do trabalho, destaca-se a proposta de APs virtuais, que provê isolamento dos clientes na camada MAC do 802.11 e divisão da rede Wi-Fi em fatias, favorecendo a virtualização da rede.

Abordagem semelhante a proposta por Schulz-Zander et al. (2014) foi utilizada nesta dissertação, onde os APs são configurados de forma a disponibilizar a mesma rede sem fio para o cliente móvel à medida que ele se distancia do AP atual e se aproxima do AP alvo. Com isso, o cliente tem a impressão de um AP virtual onipresente, o que viabiliza o controle do processo

de *handoff* pela rede, conforme detalhado no Capítulo 5.

Por fim, em (BRITO et al., 2014), foi proposto um *framework*, chamado OpenWiMesh, para orquestração de redes *mesh* sem fio que permite empregar técnicas de engenharia de tráfego, roteamento multi-caminhos, dentre outros, a partir da programação de novas funções no controlador e com base em um grafo da rede. Para superar a insuficiência do OpenFlow nas redes Wi-Fi, servimo-nos de um agente (*GraphClient*) que coleta as informações do meio sem fio e as envia ao controlador SDN, para atualização do grafo da rede. O OpenWiMesh servirá de base para o desenvolvimento da proposta de mobilidade transparente deste trabalho. Por conseguinte, retrata-se sua arquitetura e funcionamento geral em mais detalhes no Capítulo 4.

2.5 CONSIDERAÇÕES PARCIAIS

Este capítulo discutiu os conceitos, arquiteturas, protocolos e desafios correlatos à integração de WMNs e SDN, assim como os principais trabalhos relacionados. O embasamento teórico e a revisão da literatura ora apresentados são base para o desenvolvimento da solução de gerenciamento de mobilidade proposta neste trabalho. Foi possível compreender a arquitetura das WMNs e observar as deficiências das estratégias de roteamento comumente utilizadas. Além disso, discutiu-se o paradigma SDN, sua arquitetura e componentes, e como ele pode impulsionar as redes *mesh* sem fio para promover maior inovação e maior eficiência das funções de rede, como gerenciamento de mobilidade.

3 GERENCIAMENTO DE MOBILIDADE

O gerenciamento de mobilidade surgiu para resolver o problema de transferência do usuário entre as diversas redes celulares sem interrupção das chamadas em andamento, processo conhecido como *handover* ou *handoff*. Com o advento das redes sem fio domésticas, popularização do uso de dispositivos inteligentes portáteis e advento das redes de celulares de quinta geração (5G), o suporte à mobilidade transparente deixa de ser apenas necessidade das redes celulares e torna-se requisito fundamental nas redes móveis do futuro. Neste contexto, é necessário que o processo de *handoff* seja realizado de forma rápida e eficiente, pois a interrupção da transmissão significa perda de qualidade e credibilidade do serviço.

O gerenciamento de mobilidade em WMNs leva em consideração, além da transição de clientes móveis entre APs, a variabilidade dos enlaces sem fio que compõem a rede *mesh* sem fio, cujas rotas devem ser realocadas para novos caminhos à medida que os enlaces degradam a níveis que impactam a qualidade da transmissão.

Este capítulo apresenta uma revisão da literatura acerca das principais estratégias de gerenciamento de mobilidade em diferentes contextos. O capítulo está organizado da seguinte forma. Na Seção 3.1, serão discutidos os principais aspectos do gerenciamento de mobilidade e o processo de *handoff*. Na Seção 3.2, será abordada a mobilidade intra-domínio, ou seja, aquele em que o cliente móvel transita entre pontos de acesso que pertencem ao mesmo contexto de rede. Já na Seção 3.3, serão apresentadas as estratégias de mobilidade inter-domínio, em especial a mobilidade IP. Por fim, na Seção 3.4, são listados trabalhos relacionados ao gerenciamento de mobilidade em geral.

3.1 PROCESSO DE HANDOFF

O gerenciamento de mobilidade é composto pelo processo de *handoff* e pelo gerenciamento de localização do cliente móvel (AKYILDIZ; WANG, 2009). O primeiro consiste em manter as

conexões do usuário ativas, sem interrupções ou degradações, face à transição entre pontos de acesso da rede, ao passo que o gerenciamento de localização visa rotear corretamente o tráfego destinado ao cliente móvel à medida que ele se conecta em diferentes redes. O desafio do gerenciamento de mobilidade reside em monitorar o cliente móvel para garantir que a mobilidade ocorra de forma célere e apropriada, enquanto mantém a sobrecarga de sinalização em níveis aceitáveis para não sobrecarregar a rede.

O processo de *handoff* consiste basicamente em migrar um cliente móvel de ponto de acesso, alterando sua associação do *AP atual*, de quem o cliente móvel se distancia, para um *AP alvo*, ao qual o cliente móvel se aproxima. Esse processo é apenas um dos componentes de uma solução de gerenciamento de mobilidade, podendo ser complementado por outras ações que visam manter a conectividade IP do nó móvel. Outras ações que podem complementar o processo de *handoff* são a criação de túneis IP, tradução de endereços, uso de *proxies* intermediários etc. Essas ações dependem do tipo de *handoff* e do contexto em que ele está sendo executado. Para entender melhor esse processo, as próximas seções detalham os tipos de *handoff* (Seção 3.1.1), as etapas de detecção e inicialização do *handoff* (Seção 3.1.2) e o processo de escolha do AP alvo (Seção 3.1.3).

3.1.1 TIPOS DE HANDOFF

O processo de *handoff* pode ser analisado de duas perspectivas (PAUL, 2014): *intra-domínio*, quando o *handoff* ocorre entre APs coordenados que executam a mesma tecnologia de rede sem fio; ou *inter-domínio*, onde o *handoff* pode ocorrer entre APs não coordenados, provavelmente configurados em diferentes domínios de rede, ou mesmo entre diferentes tecnologias de rede sem fio (e.g. Wi-Fi, 3G/4G, WiMAX etc.). O *handoff* intra-domínio é por vezes referenciado na literatura como *micro-mobilidade* ou *handoff horizontal*, à medida que o *handoff* inter-domínio pode ser conhecido como *macro-mobilidade* ou *handoff vertical*. Esta dissertação concentra-se em mobilidade intra-domínio, embora possa ser estendido para dar suporte à mobilidade inter-domínio em trabalhos futuros.

Em geral, tanto na mobilidade intra-domínio quanto inter-domínio, existem duas estratégias genéricas para iniciar o *handoff* (PAUL, 2014): *hard handoff*, também conhecido como *break before make*, onde o cliente primeiro desconecta do AP atual e então conecta ao AP alvo; e *soft handoff*, ou *make before break*, que permite ao cliente móvel iniciar a comunicação com o AP alvo antes de desconectar do AP atual.

O processo de *handoff* pode ser classificado, ainda, quanto à entidade que o monitora, detecta e executa, dando origem às três categorias a seguir: i) *handoff controlado pelo cliente*

móvel, um processo descentralizado onde os clientes móveis continuamente monitoram a rede e executam o *handoff* ao atingir determinado critério; ii) *handoff controlado pela rede*, cujo processo de *handoff* é executado de forma centralizada e coordenada por uma entidade ou protocolo de controle da rede, sem interferência do cliente; iii) *handoff assistido pelo cliente móvel*, onde o *handoff* é controlado pela rede, porém o cliente participa de forma assistida do processo, trocando informações ou executando comandos.

Em redes 802.11 tradicionais, o *handoff* é um processo controlado pelo cliente, normalmente executado em três fases: i) *escaneamento*, onde o cliente móvel executa uma sondagem no meio sem fio para identificar APs candidatos para o *handoff*, decorrendo-se de forma ativa (cliente gera *probe requests*) ou passiva (cliente aprende sobre os APs pelos *beacons* gerados por eles); ii) *seleção do AP*, quando o cliente escolhe o AP mais apropriado para o *handoff*, utilizando métricas locais e informações difundidas pelos APs, usualmente métricas de qualidade do sinal baseadas em *Indicador de Potência do Sinal Recebido* (RSSI, do inglês *Received Signal Strength Indicator*); e iii) *associação*, onde mensagens de requisição de associação são trocadas, possivelmente ocorre autenticação do cliente e se dá a configuração de parâmetros de rede. Esse processo completo pode impactar significativamente as aplicações ativas, com latência que pode variar entre centenas de milésimos de segundos até alguns segundos (CROITORU; NICULESCU; RAICIU, 2015; AMIR et al., 2010; MISHRA; SHIN; ARBAUGH, 2003). Essa latência degrada a vazão da rede e inviabiliza o uso de aplicações de tempo real como voz sobre IP e transmissão de vídeo interativo.

3.1.2 DETECÇÃO E INICIALIZAÇÃO DO HANDOFF

Um ponto chave para obter melhor mobilidade dos clientes é determinar o momento mais oportuno para mudar de AP. A identificação do instante apropriado para inicializar o *handoff* está fortemente relacionada com o monitoramento da qualidade das associações. Isto implica que deve-se detectar o instante em que a qualidade de associação com o AP atual está abaixo de índices aceitáveis e, por outro lado, outro AP oferece melhores condições de associação. A abstração de uma métrica de qualidade do enlace Wi-Fi, em conjunto com um mecanismo de detecção eficiente, podem ser utilizados para encontrar a melhor oportunidade de *handoff*.

A qualidade de sinal, principalmente baseada em *Indicador de Potência do Sinal Recebido* (RSSI, do inglês *Received Signal Strength Indicator*) ou *Relação Sinal Ruído* (SNR, do inglês *Signal to Noise Ratio*), é a métrica mais utilizada para inicialização do *handoff* (SCHULZ-ZANDER et al., 2014; WU; DJUKIC; MOHAPATRA, 2008). Em estratégias onde o *handoff* é controlado pela rede, o RSSI ou SNR são medidos no AP, refletindo, dessa maneira, a qualidade

de sinal no *uplink* do cliente (sentido do cliente para o AP). Já em estratégias cujo *handoff* é controlado pelo cliente, o RSSI e SNR são medidos no cliente, retratando a qualidade do *downlink* (sentido do AP para o cliente). Considerando que ele é unidirecional e variável, o RSSI sozinho não apresenta precisão para ser usado como única métrica de qualidade do enlace.

Os contadores de pacotes retransmitidos podem ser utilizados como outro indicador para o *handoff* (OMHENI; ZARAI; KAMOUN, 2010). Esses contadores refletem a funcionalidade de confirmação de entrega da camada MAC do 802.11, de forma que o reenvio de um pacote ocorre sempre que sua confirmação não seja recebida em determinado intervalo de tempo, aplicando-se em tráfego *unicast*. A quantidade de retransmissões pode indicar um enlace com perdas ou com baixa capacidade de transmissão. Não obstante, observando o aspecto de oscilação da rede Wi-Fi, essa métrica individualmente pode levar à execução de *handoffs* desnecessários e redundantes, efeito conhecido como *ping-ponging* (PAUL, 2014). Por exemplo, considere um nó cliente transmitindo e recebendo dados de determinado fluxo. Em certo instante, o cliente executa a função de escaneamento ativo do 802.11 para descobrir os vizinhos na rede ou outras redes disponíveis. Nesse momento, devido à ocupação da interface sem fio, é possível que o nó cliente atrase o envio das confirmações do 802.11 e, caso o AP tome como base apenas o contador de retransmissão para medir a qualidade do enlace, uma ação de *handoff* poderia ser precipitadamente iniciada.

No complemento IEEE 802.11s (IEEE, 2011b), a métrica adotada chama-se *airtime*, que contabiliza o custo total de transmissão de pacotes, levando-se em consideração a taxa de transmissão, custos inerentes à pilha de protocolos e canal sem fio e taxa de erros para um pacote de teste de tamanho fixo (e.g. 1 Kbyte). Cada nó da rede *mesh* sem fio, portanto, monitora de forma ativa os seus vizinhos e calcula a métrica *airtime* de cada enlace unidirecional. Essa métrica é utilizada pelo protocolo HWMP na escolha de caminhos e também para identificação de oportunidades de *handoff*. Na rede de acesso de usuários, o *handoff* continua sendo controlado pelo cliente. Como alternativa, é possível configurar os clientes como parte da WMN, para dispositivos de usuário que suportem 802.11s, porém essa abordagem não se mostra factível em cenários convencionais (SCHULZ-ZANDER et al., 2014) uma vez que permite associação direta entre os clientes e pode incorrer em problemas de escalabilidade para pacotes encaminhados por inundação ou na negociação dos parâmetros Wi-Fi.

Em (PAUL, 2014), algumas estratégias de medição da qualidade do sinal e decisão de *handoff* são relatadas. Uma dessas estratégias é chamada *Potência de Sinal Relativo com Limiar e Histerese* (RSS-TH, do inglês *Relative Signal Strength with Threshold and Hysteresis*), cujo funcionamento geral pode ser visto na Figura 3.1. Nessa estratégia, o *handoff* é iniciado so-

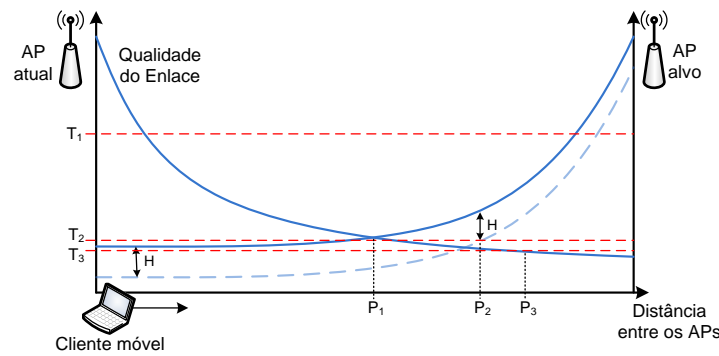


Figura 3.1: Potência de Sinal Relativa com Limiar e Histerese (PAUL, 2014)

mente se o RSSI do AP atual diminui abaixo de um limiar T (do inglês, *threshold*), ao passo que o AP alvo possui valor de RSSI maior que o atual por uma margem H (histerese). Conforme o exemplo da Figura 3.1, o *handoff* seria iniciado no ponto P_2 , caso o limiar fosse T_1 ou T_2 ; e no ponto P_3 , caso limiar fosse T_3 . No ponto P_1 , mesmo tendo atingido os limiares T_1 e T_2 , o *handoff* não seria iniciado devido ao critério da margem histerese; do contrário, a variação do sinal poderia levar a oscilações sobre qual AP oferece melhor RSSI. De modo complementar, mesmo configurando-se um limiar agressivo como T_1 , o *handoff* não seria desnecessariamente executado, sobretudo, constando-se que não há AP candidato com margem acima de H . Logo, essa estratégia evita o efeito *ping-ponging* e reduz a ocorrência de *handoffs* inoportunos. Uma estratégia similar ao RSS-TH foi utilizada nesta dissertação, exceto pelo uso de uma métrica de qualidade de enlace diferenciada, calculada a partir da combinação de duas outras medidas (Seção 5.2), em detrimento do RSSI apenas.

3.1.3 SELEÇÃO DE AP

Após identificar uma oportunidade de *handoff*, o próximo passo é escolher o AP alvo com o qual o cliente irá se associar. Mesmo em cenários onde o cliente possui apenas uma opção de candidato a AP alvo, o algoritmo de seleção de AP é importante para avaliar se o *handoff* é viável em termos de garantias de recursos (PAIVA et al., 2013) ou em termos de balanceamento de carga dos APs (HONG et al., 2015; LIU et al., 2011). No entanto, considerando a ampla implantação de redes IEEE 802.11 atual, é comum os dispositivos móveis se depararem com múltiplas opções de AP para escolher, tanto na conexão inicial quanto durante o *handoff*. A seleção de AP constitui-se, portanto, etapa primordial para o bom desempenho da rede e execução bem sucedida do *handoff*, dividindo de maneira equilibrada a carga entre APs e assim incrementando sua eficiência energética (XU et al., 2013).

Seleção de APs é um tema de pesquisa popular na academia e na indústria (HONG et al., 2015; LIU et al., 2011; HSIEH et al., 2007; IEEE, 2011a; IEEE, 2008b). O mecanismo padrão de seleção de AP do 802.11 é baseado unicamente em RSSI, geralmente medido no cliente móvel em relação aos APs na área de cobertura. Embora essa técnica seja simples e, em alguns casos, efetiva, ela pode causar a concentração de clientes em APs específicos. Os clientes móveis podem escolher se associar sempre com um subconjunto pequeno de APs, que possuem melhores recursos, enquanto que outros APs, com menos recursos, permanecem ociosos. Esse mecanismo resulta em uma carga de tráfego desbalanceada na rede sem fio, degradando, portanto, a qualidade do cliente móvel (LIU et al., 2011).

De modo suplementar aos algoritmos baseados em potência de sinal, utiliza-se métricas como: estimativa de largura de banda (VASUDEVAN et al., 2005); largura de banda combinada com Tempo de Ida e Volta (RTT, do inglês *Round-Trip Time*) (NICHOLSON et al., 2006); RSSI combinado com banda estimada e taxa de transmissão da camada MAC do 802.11 (LIU et al., 2011); *airtime*, ou sobrecarga de transmissão de pacotes (IEEE, 2011b); máxima menor largura de banda por cliente (XU et al., 2013); banda residual (HONG et al., 2015); carga no AP (HSIEH et al., 2007); dentre outros.

Novas estratégias e novas métricas para seleção de AP podem ser propostas a partir da “programabilidade” e visão centralizada do paradigma SDN. O SDN possibilita a seleção de APs controlada pela rede, a partir de função coordenada entre os APs vizinhos e sem alterar as configurações do cliente ou exigir protocolos adicionais. Nesta dissertação, propõe-se uma estratégia SDN para escolha do AP durante o *handoff* (Seção 5.3).

3.2 MOBILIDADE INTRA-DOMÍNIO

Soluções de gerenciamento de mobilidade intra-domínio são caracterizadas pela transição de conectividade de um nó móvel entre redes sobre o mesmo controle administrativo, cujas tecnologias envolvidas no acesso, geralmente, mas não obrigatoriamente, mantêm-se homogêneas. Esse é o caso, por exemplo, de redes sem fio locais ou WMN, baseadas no padrão 802.11, onde o gerenciamento de mobilidade consiste no *handoff* entre APs e na atualização da localização do nó móvel nas tabelas de encaminhamento dos equipamentos.

Uma das estratégias para viabilizar o gerenciamento de mobilidade intra-domínio em WMNs é o B.A.T.M.A.N. (SANCHEZ-IBORRA; CANO, 2014; BRITTON; COYLE, 2011; REINERI; CASETTI; CHIASSERINI, 2009). Em particular, o protocolo B.A.T.M.A.N. provê suporte para atualização da localização do nó móvel nas tabelas de encaminhamento dos equipamentos, ao

passo que a execução do *handoff* entre APs fica a cargo do dispositivo cliente. A estratégia empregada pelo B.A.T.M.A.N., portanto, consiste em resolver a questão de sincronização de rotas entre os roteadores *mesh*, de acordo com a estratégia apresentada a seguir. Ao deslocar-se na rede e após executar a transição entre APs, o cliente móvel passa a ser alcançável por outro roteador *mesh* (AP alvo). O roteador *mesh* checa em sua tabela de clientes para identificar quando trata-se de um novo cliente ou de um cliente móvel. Ao detectar que trata-se de um cliente móvel, o roteador *mesh* envia uma mensagem de sinalização para o roteador *mesh* antigo, orientando-o a redirecionar todas as mensagens destinadas àquele cliente para a sua nova localização. Em paralelo, o roteador *mesh* envia novas mensagens de OGM (conforme Seção 2.1.2) para atualizar a tabela de encaminhamento dos outros nós. Embora essa estratégia ofereça bons resultados em comparação com outras abordagens bem conhecidas como OLSR (SANCHEZ-IBORRA; CANO, 2014; BRITO; FIGUEIREDO, 2015), sua eficiência está limitada ao desempenho do processo de transição entre APs controlado pelo cliente.

Análises de desempenho do processo de *handoff* em equipamentos padrão 802.11 mostram que 90% do atraso no *handoff* é ocasionado pelo escaneamento do cliente em busca de APs candidatos (MISHRA; SHIN; ARBAUGH, 2003; SCHULZ-ZANDER et al., 2014; AMIR et al., 2010). Os experimentos mostram ainda que esse atraso pode variar em grande medida entre diferentes fabricantes de placas Wi-Fi, devido às particularidades nas estratégias de decisão de *handoff* de cada solução. São imprescindíveis, desse modo, mecanismos eficientes de execução de *handoff* independentes da plataforma cliente.

Ao longo dos anos, o IEEE propôs diferentes padrões para aperfeiçoar os componentes do *handoff*, como é o caso dos complementos 802.11k, 802.11v e 802.11r. O complemento 802.11r (IEEE, 2008a) oferece melhorias na última fase do *handoff*, através do armazenamento das chaves de autenticação e criptografia na memória *cache* dos APs, favorecendo rápida transição entre APs. A autenticação ocorre apenas uma vez: quando o cliente móvel entra no domínio de mobilidade. Movimentações subsequentes dentro daquele domínio fazem uso do material criptográfico derivado da primeira autenticação, reduzindo o tempo de reassociação. O complemento 802.11k (IEEE, 2008b) permite que o AP atual ofereça assistência ao cliente móvel enviando um relatório de APs candidatos na vizinhança do nó, em conjunto com seus respectivos canais. Essa lista de APs candidatos pode ser utilizada em conjunto com informações de topologia e carga enviadas ao cliente via 802.11v (IEEE, 2011a) como entrada para o processo de seleção de AP. Adicionalmente, mensagens do complemento 802.11v podem ser enviadas de forma não solicitada para redirecionar um cliente para um AP menos congestionado ou com melhor potência de sinal na vizinhança. As melhorias alcançadas com essas funcionalidades dependem do suporte a cada uma delas nos dispositivos clientes, não benefi-

ciando por vezes equipamentos legados, o que pode ser uma barreira para amplo benefício da rede (SCHULZ-ZANDER et al., 2014; CROITORU; NICULESCU; RAICIU, 2015).

A fim de contemplar um maior número de clientes, dispensando funcionalidades avançadas nos dispositivos, pode-se lançar mão de técnicas de gerenciamento de *handoff* controlado pela rede, por exemplo operando a rede de acesso a partir do padrão 802.11 em modo *ad-hoc* (AMIR et al., 2010). Nesse cenário, o cliente pode associar-se com múltiplos APs ao mesmo tempo, e a decisão de qual AP é utilizado no encaminhamento de pacotes cabe ao protocolo de roteamento da rede. Desse modo, a infraestrutura *mesh* pode monitorar a qualidade das associações de um cliente e executar o *handoff*, mudando o ponto de acesso do cliente pelo simples envio de mensagens de ARP gratuito (AMIR et al., 2010) ou de atualização de vizinho via ICMPv6. Adotando como premissa que os dispositivos Wi-Fi estão em conformidade com o padrão base 802.11, o modo de operação *ad-hoc* é amplamente suportado, o que potencializa sua aplicabilidade. Por outro lado, ao permitir associações diretas entre os dispositivos, o modo *ad-hoc* pode culminar em riscos a segurança e a privacidade dos clientes.

Soluções Wi-Fi corporativas reduzem o tempo de *handoff* a partir da visão global e capacidade de correlação de informações que possuem sobre os APs. Considera-se, ainda, a habilidade de execução coordenada de funções de gerenciamento como aumento ou redução de potência de sinal, reconfiguração de canais e identificadores da rede, levantamento de carga dos nós etc. Não obstante, essas técnicas ainda dependem da inicialização do *handoff* pelo cliente, não endereçando, então, o atraso do *handoff* na camada de enlace inerente das redes 802.11.

3.3 MOBILIDADE INTER-DOMÍNIO

3.3.1 PROTOCOLOS MIB E PMIPv6

A fim de permitir a mobilidade no âmbito da Internet, em 1996, a IETF lançou a padronização do protocolo *IP móvel* (MIP, do inglês *Mobile IP*), inicialmente especificado na RFC 2002 e atualizado na RFC 5944 (PERKINS, 2010). O protocolo MIP permite que dispositivos móveis permaneçam com o mesmo endereçamento IP ao realizar o processo de transição para outro ponto de acesso, que, frequentemente, se encontra configurado em uma rede diferente. Dessa maneira, um *Nó Correspondente* (CN, do inglês *Correspondent Node*), com o qual o cliente móvel está se comunicando, permanece inalterado. Para isso, o cliente móvel dependerá de dois roteadores para intermediar seu tráfego IP: o *agente doméstico* (HA, do inglês *Home Agent*) e o *agente estrangeiro* (FA, do inglês *Foreign Agent*). No primeiro passo, o cliente móvel registra seu endereço IP com o HA. Ao mover-se para outro ponto de acesso ou outro domínio

de rede, o cliente recebe um novo endereçamento IP (CoA, do inglês *Care-of-Address*) do FA. O cliente móvel então registra seu novo endereço IP com o HA e, a partir daí, o tráfego de dados é encaminhado através de um túnel com o HA.

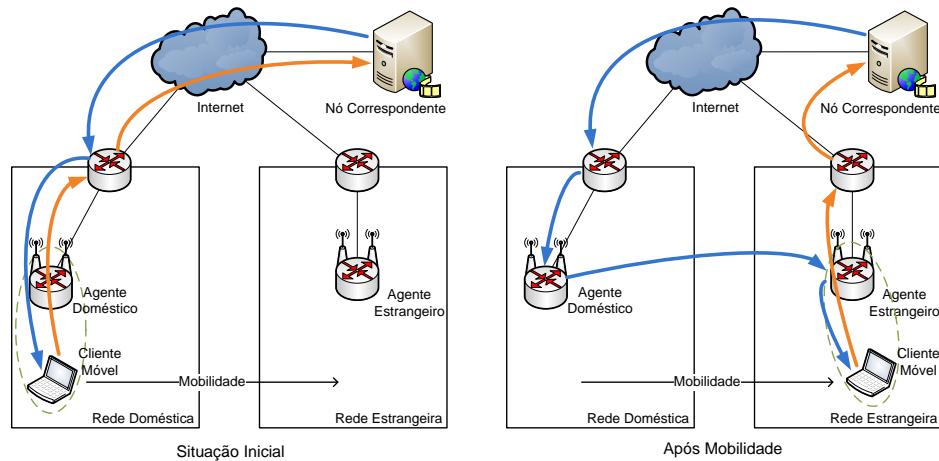


Figura 3.2: Exemplo de funcionamento do protocolo *IP móvel*

A Figura 3.2 ilustra um exemplo de funcionamento do protocolo MIP conforme apresentado anteriormente. É possível observar que existem alguns pontos negativos no funcionamento geral deste protocolo, a saber:

- O MIP exige que todo cliente móvel tenha um HA em sua rede de origem;
- Em redes que possuem filtro de pacote baseado no endereço IP de origem, o FA deve encaminhar todo o tráfego do cliente móvel para o HA via túnel, criando atraso na entrega dos pacotes;
- Em redes que não possuem filtro de pacotes baseado no endereço IP de origem existe assimetria de tráfego, ou roteamento triangular, pois todo tráfego destinado ao cliente móvel precisa primeiro ir para o HA, depois para o FA e posteriormente ser entregue ao cliente. Já no sentido oposto, o cliente pode enviar os pacotes diretamente para o CN, fazendo com que o caminho de ida e volta sejam diferentes, dificultando a aplicação de mecanismos de QoS e resolução de problemas;
- O HA pode ter problemas de escalabilidade caso existam muitos clientes móveis na rede de origem;
- Devido à sobrecarga gerada pela sinalização do protocolo MIP, o processo de *handoff* pode apresentar tempos elevados para convergência, inviabilizando aplicações multimídia sensíveis ao atraso e perda de pacotes (ITU-T, 2003);

- Para utilizar o MIP é necessário modificar o sistema operacional do cliente móvel com o objetivo de habilitar essas novas funções. Além disso, a execução do processo referente ao MIP no cliente móvel pode trazer sobrecarga e impacto na autonomia elétrica do dispositivo. Ambos os fatores dificultam a utilização efetiva dessa abordagem.

Os problemas decorrentes do MIP e a demanda por um protocolo que permitisse o gerenciamento de mobilidade baseada na rede motivou o desenvolvimento do protocolo PMIPv6 ou Proxy MIPv6 (GUNDAVELLI et al., 2008). O PMIPv6 herda algumas das características trazidas pelo precursor MIP, exceto pela retirada da função de gerenciamento de mobilidade do dispositivo do usuário. Ele é composto por dois tipos de dispositivos: o LMA (do inglês *Local Mobility Anchor*) e o MAG (do inglês *Mobility Access Gateway*). O MAG tem a função de detectar a chegada de um cliente móvel e registrar esse cliente junto ao LMA. O LMA, por sua vez, tem uma função similar a do HA no MIP, gerenciando o serviço de mobilidade dos clientes, prefixos alocados para cada cliente e criação de túneis com os respectivos MAGs. Pelo LMA passam todos os pacotes da comunicação entre os dispositivos externos ao domínio PMIP e o cliente móvel.

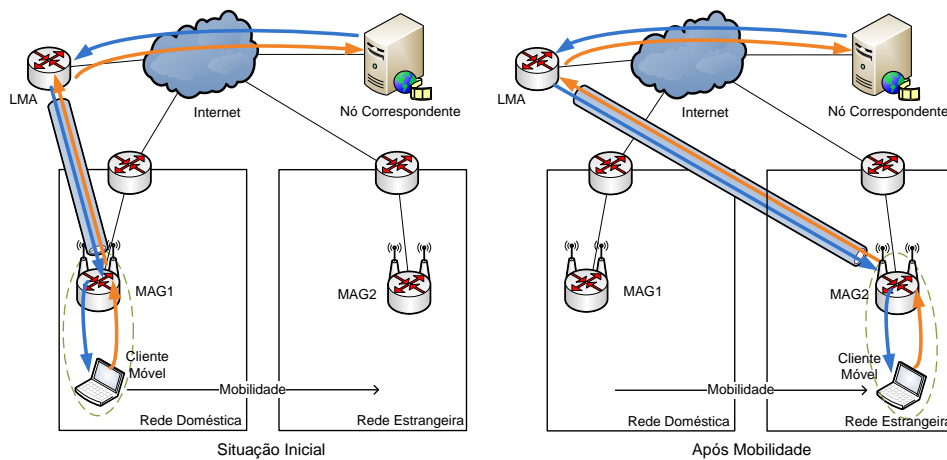


Figura 3.3: Exemplo de funcionamento do protocolo PMIPv6

Um resumo do funcionamento do PMIPv6 pode ser visto na Figura 3.3. Nessa figura, observa-se que toda comunicação do cliente móvel com o CN é intermediada pelo LMA. Além disso, o MAG é responsável por encapsular e desencapsular o tráfego na comunicação com o cliente móvel, mantendo-o conectado de forma transparente, com endereçamento IP independente do domínio de rede atual (prefixo alocado pelo LMA). Percebe-se, ainda, que nas trocas de mensagem do PMIPv6, o usuário é um agente passivo. Cabe ao dispositivo do usuário apenas a sinalização de mudança de rede, por meio de mensagens da camada de enlace (GUNDAVELLI

et al., 2008) ou de mensagens de solicitação e divulgação de rotas (e.g. RS/RA¹).

Para alcançar mobilidade eficiente no PMIPv6, mantendo a qualidade de serviço da rede para tráfego de tempo real, todo esse processo de *handoff* descrito acima precisa ser executado de forma otimizada, principalmente a detecção e sinalização de mobilidade. A antecipação de *handoff* (AVELAR; AVELAR; DIAS, 2013) e o monitoramento não intrusivo do cliente móvel são técnicas elegíveis para otimização desse processo.

3.3.2 PADRÃO IEEE 802.21

Em termos de normatizações, o IEEE propôs o padrão 802.21, que provê mecanismos para viabilizar *handoff* vertical. Esse padrão é também conhecido como *handoff independente de mídia* (MIH, do inglês *Media Independent Handoff*). O MIH fornece um arcabouço de funções para gerenciamento de mobilidade, além de uma interface comum para fornecer informações da camada física e de enlace às camadas acima. Não obstante, grande parte dos clientes atuais, bem como algumas plataformas de pontos de acesso, não oferecem suporte para essa funcionalidade, sendo necessária instalação de protocolos adicionais no dispositivo do cliente móvel.

3.3.3 MOBILIDADE EM REDES CELULARES DE PRÓXIMA GERAÇÃO

A comunicação em redes de celulares está movendo-se para uma nova geração de conectividade sem fio, onde a conexão dos usuários será provida através de redes heterogêneas (HetNet, do inglês *Heterogeneous Networks*) (NAEEM; NGAH; HASHIM, 2016). Nesse cenário, o conceito de *small cells* está sendo incorporado à infraestrutura de macro-células e micro-células a fim de prover maior capacidade e cobertura para a rede celular móvel. Consequentemente, espera-se que a quantidade de *handoffs* dos clientes móveis cresça significativamente, ao passo que se mantenha as métricas de qualidade de serviço e experiência na rede sem fio. Por outro lado, executar mobilidade transparente nas HetNets requer a troca rápida de informações e considerando parâmetros heterogêneos para decisão de *handoff*, o que pode se tornar um desafio para as redes de próxima geração.

Em Naeem, Ngah e Hashim (2016), os autores apresentam o estado da arte com relação a técnicas de *handoff* vertical para HetNets e apresentam alguns desafios que devem ser considerados nesse cenário. Entre os desafios apresentados destacam-se: a necessidade de garantia de qualidade de serviço e qualidade de experiência nas redes de acesso candidatas durante o *han-*

¹Mensagens referentes ao protocolo IPv6, *Router Solicitation* e *Router Advertisement*.

doff; necessidade de garantias de segurança na transferência de dados entre as diferentes redes; controle dos operadores de rede para validação dos planos aplicados a cada cliente; técnicas para minimização de efeitos *ping-ponging*; e balanceamento de carga entre as estações base e pontos de acesso.

As técnicas de *handoff* vertical para HetNets apresentadas em (NAEEM; NGAH; HASHIM, 2016) foram: i) métodos de estimativa de tempo para detecção de *handoff* e minimização de efeito *ping-ponging*, por exemplo através da análise do tempo de chegada de frames para identificação da movimentação dos usuários; ii) soluções baseadas em Lógica Fuzzy e Redes Neurais para processamento de parâmetros das HetNets; e iii) soluções baseadas em limiares e ponderações, onde pode-se definir limiares e diferentes pesos para cada métrica, combinando-as em uma matriz de entropia para tomada de decisão do *handoff*.

3.4 TRABALHOS RELACIONADOS

Variadas estratégias de mobilidade em redes Wi-Fi vem sendo apresentadas na literatura, combinando diferentes técnicas de detecção, execução, configuração da rede etc. O advento do paradigma SDN impulsiona, ainda, novas metodologias, algoritmos e métricas cada vez mais em conformidade com os requisitos das aplicações atuais. A seguir alguns desses trabalhos serão apresentados.

No trabalho de Amir et al. (2010), apresenta-se a arquitetura de uma rede *mesh* sem fio chamada SMesh, cujo foco principal é a mobilidade dos nós clientes, com *handoff* transparente e sem interrupção (*seamless handoff*), oferecendo suporte às aplicações de tempo real. Os clientes são dispositivos 802.11 não modificados, que enxergam a rede *mesh* sem fio com a abstração de um único ponto de acesso. O SMesh utiliza mensagens *multicast* e túneis para enviar dados por múltiplos caminhos até o cliente móvel, incrementando a probabilidade de recepção de pacotes durante o *handoff*. Outra característica desse trabalho é que tanto os pontos de acesso quanto os clientes são configurados com 802.11 em modo *ad-hoc*. Assim, o *handoff* é controlado pela infraestrutura *mesh* e não pelos clientes.

Para detectar uma oportunidade de *handoff*, o SMesh envia regularmente requisições DHCP ou ARP e quantifica a perda de pacotes e a qualidade do enlace sem fio, iniciando o processo de *handoff* ao atingir determinado limiar. A métrica RSSI pode também ser utilizada em nós com rádio adicional executando em modo *monitor*, porém não é empregada por padrão.

Um ponto negativo dessa proposta é o uso de túneis, o que pode causar sobrecarga na rede e limitações de tamanho de pacote. Outras questões concernem ao uso do modo *ad-hoc* no

acesso dos cliente. O uso do modo *ad-hoc* no acesso permite conexões diretas entre os clientes, trazendo implicações na segurança (e.g. interceptação de tráfego). Além disso, na prática, a maioria das implementações do modo *ad-hoc* suportam apenas esquemas de criptografia fracos (e.g. WEB), o que pode gerar problemas de confidencialidade. Um terceiro ponto de atenção é o monitoramento ativo da rede através de DHCP ou ARP, apresentando-se simples e efetivo, contudo podendo trazer impacto à rede devido ao tráfego adicional. Nesta dissertação é utilizado uma abordagem de monitoramento passivo da qualidade dos enlaces sem fio, conforme será apresentado no Capítulo 5.

Omheni, Zarai e Kamoun (2010), por sua vez, propõem mecanismos para reduzir o tempo de escaneamento do *handoff* e métricas para seleção do AP alvo. A partir de informações históricas de *handoff* e lista de vizinhos, o cliente móvel faz a busca de APs apenas em um subconjunto de canais disponíveis, diminuindo o atraso na descoberta de candidatos. Na seleção de AP, utiliza-se o custo total de transmissão de um quadro 802.11, chamada *airtime* mínimo. Para avaliar a proposta, os autores executaram testes simulados e mediram métricas de vazão e tempo de descoberta de APs no *handoff*. Em ambas as métricas, a abordagem de otimização do escaneamento possui bom desempenho, com redução de até 250 ms em relação ao *handoff* 802.11 tradicional. Em ambiente real essa estratégia dependeria de modificações no *driver* Wi-Fi do cliente, portanto, sua adoção estaria condicionada à customização no cliente. Nesta dissertação propõe-se uma abordagem que não depende de mudanças no dispositivo cliente.

Em Dely, Kessler e Bayer (2011) foi apresentada uma arquitetura para integração entre redes *mesh* sem fio e SDN, com implementação de uma função de gerenciamento de mobilidade onde o controlador monitora continuamente a qualidade dos enlaces e a capacidade dos APs *mesh*, disparando uma ação de *handoff* no cliente quando detecta a necessidade. Não foram apresentadas as métricas de qualidade e limiares de *handoff* utilizados. Na execução do *handoff*, o controlador configura as regras OpenFlow no novo caminho, mantendo temporariamente o antigo, e dispara no cliente um comando de IEEE 802.21 (protocolo de *handoff* independente de mídia) para iniciar a migração de conexão para o novo AP. Dessa forma, faz-se necessário que os clientes sejam modificados para suportar o protocolo 802.21 além da instalação de um agente Nagios para monitoramento remoto da rede sem fio. A proposta foi avaliada em um *testbed* real chamado KAUMesh.

A *Lógica Fuzzy* também tem sido utilizada na literatura como mecanismo de tomada de decisão do *handoff* (AVELAR; AVELAR; DIAS, 2013; WU, 2011), tomando como entrada parâmetros de rede como vazão, atraso e RSSI. No PMIPFlow-Fuzzy (AVELAR; AVELAR; DIAS, 2013), a técnica de *handoff* baseado em lógica Fuzzy foi implementada em uma infra-

estrutura SDN sem fio para prover gerenciamento de mobilidade IP (inter-domínio). Utilizando um componente no ambiente do cliente, o PMIPFlow-Fuzzy dispara o processo de *handoff* com base no sistema de decisão Fuzzy.

Para avaliar a proposta, os autores projetaram um *testbed* 802.11/OpenFlow e executaram testes de QoS e QoE. A partir de testes com a ferramenta *iperf*, foi possível notar significativa melhora nos parâmetros de QoS após a antecipação de *handoff* com lógica Fuzzy. As métricas de QoE foram aferidas a partir de testes de qualidade de vídeo em transmissões pela rede. Os resultados de QoE demonstraram ligeira vantagem, porém com valores muito próximos do cenário sem antecipação de *handoff*. A necessidade de instalação de componente PMIPFlow-Fuzzy no cliente não está em conformidade com o objetivo inicial de prover mobilidade transparente ao usuário, podendo ser um empecilho à sua ampla adoção.

Diferente da proposta PMIPFlow-Fuzzy, esta dissertação propõe o uso de monitoramento passivo e métricas de qualidade do enlace Wi-Fi para detecção da mobilidade, ao passo que independe de mudanças no dispositivo cliente para iniciar o processo de *handoff*.

Outro caminho em direção à mobilidade eficiente é eliminá-la na perspectiva do cliente, conforme proposto pelos autores Schulz-Zander et al. (2014). Utilizando um canal Wi-Fi único e arquitetura de APs virtuais, onde múltiplos APs coordenados anunciam o mesmo BSSID (do inglês *Basic Service Set Identification*), o cliente tem a visão de um único AP, onipresente na rede. O trabalho propõe uma solução para orquestração de redes Wi-Fi com SDN, chamada Odin. Entre as contribuições do trabalho, destaca-se a proposta do conceito de *pontos de acesso leve virtuais* (LVAP, do inglês *Light Virtual Access Points*), uma abstração da pilha de protocolos 802.11 para viabilizar a conectividade dos clientes. Com LVAPs, cada cliente recebe um BSSID único para conectar-se, tornando-o um AP específico do cliente. Para manter a associação, os LVAPs enviam periodicamente *beacons* específicos de cada cliente. Uma vez que o cliente se movimenta e um novo LVAP continue enviando *beacons* daquela associação na periodicidade aceitável, o cliente não faz o reescaneamento e reassociação, pois, do ponto de vista do cliente, o AP ainda está presente. Assim, migrando o LVAP de cada cliente entre os APs físicos, a infraestrutura pode então controlar o *handoff* do cliente, sem modificá-lo e sem técnicas de sinalização de canais adicionais. Uma abordagem semelhante ao LVAP foi utilizada nesta dissertação.

Já em (CROITORU; NICULESCU; RAICIU, 2015) os autores propõem uma estratégia para mobilidade eficiente dos clientes, porém independente de um processo de *handoff* otimizado. Em contrapartida, os autores preconizam a adoção do padrão emergente MPTCP (do inglês *Multipath TCP*) que possibilita o uso efetivo de múltiplos APs ao mesmo tempo e faz com

que o problema de *handoff* em camada de enlace passe a ser tratado em camada de transporte. Os autores avaliam a proposta através de um *testbed* 802.11, com diversificados cenários e métricas. A avaliação mostrou resultados satisfatórios na maioria das configurações, porém foram verificadas quedas de desempenho na combinação de tecnologias como 802.11n e 802.11g. Para contornar esse problema de desempenho, os autores propuseram e validaram modificações na função de estimativa de qualidade, usada pelo controle de congestionamento MPTCP para escolha dos APs mais eficientes. De maneira semelhante a outras propostas previamente discutidas, a abordagem de MPTCP requer modificações no dispositivo do cliente, que vai de encontro à premissa original de manter o cliente agnóstico à estratégia de mobilidade. A abordagem proposta nesta dissertação, por outro lado, não requer qualquer mudança no dispositivo cliente.

3.5 CONSIDERAÇÕES PARCIAIS

Este capítulo apresentou uma revisão da literatura com as principais estratégias de gerenciamento de mobilidade, tipos de *handoff* e limitações das técnicas comumente adotadas. Foi discutido o mecanismo de *handoff* presente no padrão 802.11 e as fases que o dispositivo cliente deve executar durante a migração entre pontos de acesso. Algumas propostas de otimização e complementos especificados pelo IEEE para melhorar o desempenho do processo de *handoff* foram apresentadas, porém todas elas requerem modificações no dispositivo do cliente, podendo ser um empecilho a ampla adoção. Discutiu-se ainda as fases de detecção e inicialização do *handoff* e seleção de AP, onde diversas métricas para monitoramento da qualidade das associações e escolha do AP alvo foram apresentadas. Por fim, foram relatados os principais protocolos e estratégias propostas na literatura para gerenciamento de mobilidade.

4 ARQUITETURA DO FRAMEWORK OPENWIMESH

A adoção do paradigma SDN no contexto de redes *mesh* sem fio abre espaço para novas funcionalidades como gerenciamento de mobilidade, engenharia de tráfego, encaminhamento de pacotes baseado em fluxos e, mais genericamente, “programabilidade” da rede. Se por um lado o novo paradigma agrega diversas perspectivas, por outro, alguns desafios precisam ser endereçados (conforme Seção 2.3), a exemplo do contraste entre o modelo centralizado de SDN e a arquitetura inerentemente distribuída da WMN; ou ainda, da necessidade de se fazer o controle de forma integrada à rede de dados (sinalização *in-band*), sem demandar recursos adicionais. O OpenWiMesh é um *framework* que integra SDN/OpenFlow em redes *mesh* sem fio, com sinalização *in-band* e visão global da infraestrutura através do grafo da rede. Nessa dissertação, o OpenWiMesh foi utilizado como base para desenvolvimento de uma estratégia de mobilidade transparente. Por conseguinte, é oportuno entender sua arquitetura e funcionamento geral, os quais serão objeto de estudo deste capítulo.

Em síntese, o capítulo está organizado da seguinte forma. A Seção 4.1 apresenta uma visão geral e os elementos que compõem a arquitetura do *framework* OpenWiMesh. O mecanismo de sinalização de controle da solução é detalhado na Seção 4.2. Em seguida, a Seção 4.3 aborda a modelagem da infraestrutura e topologia através do grafo da rede. Na Seção 4.4, discute-se estratégias de roteamento na rede *mesh* sem fio. Detalhes do desenvolvimento do *framework* são delineados na Seção 4.5. E, finalmente, a Seção 4.6 relata casos de uso para adoção do OpenWiMesh.

4.1 VISÃO GERAL DO OPENWIMESH

A Figura 4.1 sintetiza os principais componentes do OpenWiMesh e seus relacionamentos. O *framework* é composto por dois tipos de nós, a saber: *Roteadores Mesh* (MR) e *Controlador* (CTL). O primeiro desempenha os papéis de ponto de acesso, roteamento e *gateway* Internet, a

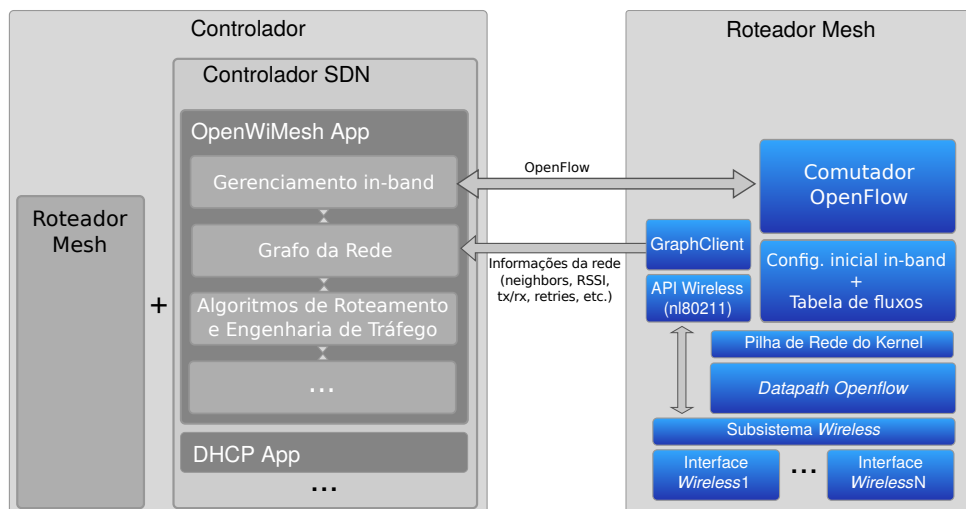


Figura 4.1: Arquitetura do OpenWiMesh

depende da infraestrutura de rede disponível e de sua conectividade. Ele executa um controlador OpenFlow, que é responsável por controlar as interfaces de rede pertencentes ao *datapath* SDN, baseado nas regras de encaminhamento enviadas pelo controlador. O controlador, por sua vez, além de também poder funcionar como MR, tem uma visão global da rede e implementa toda a lógica do plano de controle, abrangendo o controle de admissão, roteamento, monitoramento e outras funcionalidades adicionais programadas pelo administrador ou utilizador do *framework*. Os nós clientes não estão representados na Figura 4.1, uma vez que o OpenWiMesh não demanda qualquer configuração especial nestes tipos de nós.

Os principais componentes (estruturais e funcionais) do *framework* são:

- **Controlador SDN:** software de encaminhamento de pacotes no MR com suporte a OpenFlow. Consiste da tabela de fluxos, que executa o casamento e encaminhamento de pacotes e do canal de comunicação com o controlador;
- **Mecanismo de controle *in-band*:** conjunto de estratégias necessárias para viabilizar um canal de sinalização entre o nó controlador central e os roteadores *mesh* no mesmo *datapath* do tráfego de dados da rede;
- **Regras iniciais de encaminhamento:** definem como os MRs encaminham o tráfego inicialmente na WMN para que o canal de sinalização *in-band* possa ser estabelecido;
- **GraphClient:** componente instalado no MR que coleta informações de topologia da infraestrutura *mesh* e estatísticas de rede. O *GraphClient* é usado para estender as funcionalidades do OpenFlow, fornecendo estatísticas específicas da rede Wi-Fi, manuseando APIs padrões do kernel Linux (e.g. nl80211);

- **Grafo da rede:** as informações de topologia do *GraphClient* são enviadas ao controlador em conjunto com as mensagens OpenFlow. No controlador, essas informações e estatísticas são armazenadas em um grafo da rede, que é a estruturada de dados utilizada para modelar as informações de topologia, características dos nós e estatísticas do sistema sem fio;
- **Algoritmo de roteamento:** estratégia de escolha de caminhos entre uma origem e um destino na rede, podendo fazer uso das informações do grafo da rede e também de critérios não técnicos;
- **Controle geral da rede:** responsável pelo recebimento e tratamento das requisições de encaminhamento (*packet-in*), execução do algoritmo de roteamento e instalação da regra de encaminhamento em cada MR do caminho escolhido.

4.1.1 OPERAÇÃO DO FRAMEWORK

Um exemplo de cenário de WMN na arquitetura do OpenWiMesh pode ser visto na Figura 4.2. Nesse cenário, o MR1 exerce os papéis de roteador *mesh*, *gateway* Internet e provê conexão ao controlador; já MR2 exerce o papel de roteador *mesh*, além de também estar conectado ao controlador; MR3 atua apenas como roteador *mesh*; e MR4 atua como roteador *mesh* e ponto de acesso para clientes. No exemplo em questão (Figura 4.2), o cliente executa uma aplicação que se comunica com um servidor na Internet, por exemplo a aplicação de teste de conectividade *ping*, cuja rota envolve o MR4, MR3, MR2 e MR1.

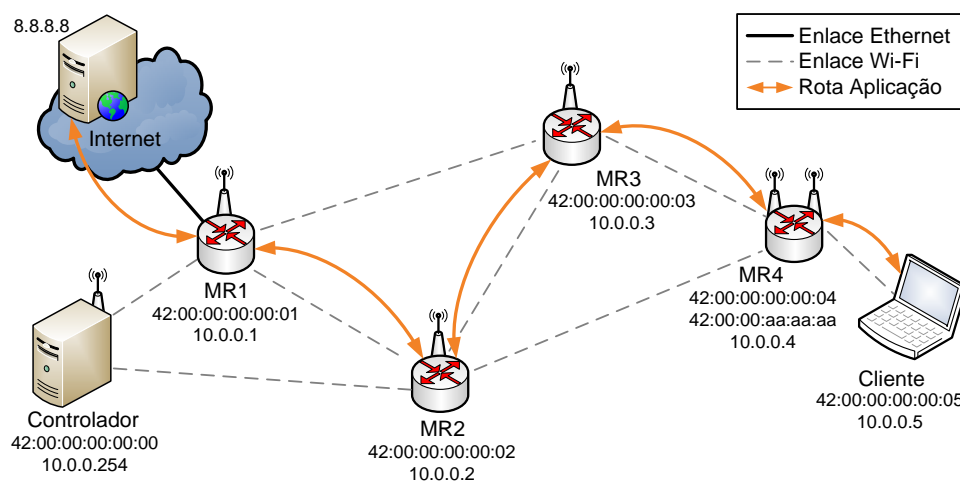


Figura 4.2: Exemplo de cenário de WMN com OpenWiMesh

A implantação de comutador SDN adotada nos MRs foi o Open vSwitch (Open vSwitch, 2013), que permite modificar regras iniciais de controle OpenFlow e oferece melhor desempenho

que outras implementações de comutadores SDN, como o comutador OpenFlow de referência¹ ou o *OpenFlow software switch*², por executar o encaminhamento de pacotes em espaço de kernel e não em espaço de usuário. O software de controle da rede é baseado no POX (POX, 2013), que suporta o desenvolvimento de aplicações SDN em Python e utiliza o protocolo OpenFlow para controlar remotamente os MRs através de um canal de sinalização.

No plano de controle da WMN, o nó controlador define, sob demanda, os caminhos que devem ser usados para cada fluxo. Assim, ao receber um pacote de um fluxo desconhecido, o MR encaminha-o para o nó controlador, realizando uma “consulta de encaminhamento”, através de mensagens OpenFlow do tipo *packet-in*. O controlador utiliza um algoritmo de roteamento sobre grafo da rede para decidir a rota que aquele tráfego deve seguir. Adicionalmente, o algoritmo de roteamento pode ser influenciado por critérios não técnicos, como o grau de credibilidade ou confiabilidade de um MR, ou ainda pode fazer uso de um subconjunto de nós com determinado grau de afinidade. Exemplos do potencial de uso do algoritmo de roteamento para influenciar na escolha de caminhos são apresentados na Seção 4.4. Em seguida, o nó controlador executa uma rotina de instalação do caminho na WMN, onde enviará uma mensagem de configuração para todos os nós pertencentes ao caminho. Essa mensagem consistirá de uma “regra de encaminhamento”, sendo representada por mensagens OpenFlow do tipo *flow-modify*. Pré-configurar todos os nós no caminho evita que consultas de encaminhamento adicionais sejam geradas para aquele fluxo.

Já no plano de encaminhamento, cada MR consulta sua tabela de fluxos em busca de uma regra de encaminhamento, que definirá o que deve ser feito com o pacote. Tal consulta pode levar em consideração valores de cabeçalhos de múltiplas camadas, além dos endereços da Camada de Enlace (endereços MAC). Após encontrar a regra, o comutador SDN executa as ações constantes na regra (e.g. encaminhar para o próximo nó, descartar, alterar o endereço MAC etc.). Caso nenhuma regra da tabela de fluxos corresponda ao tráfego em questão, o MR armazena aquele pacote no *buffer* do comutador SDN e o envia ao controlador, conforme previamente mencionado.

A cada salto o endereço MAC de destino é checado como campo de correspondência nas regras OpenFlow e os endereços MAC de origem e destino são reescritos para verificação no próximo salto, através das ações da regra. As ações da regra substituem o endereço MAC de origem do pacote pelo endereço MAC da interface Wi-Fi de saída e o endereço MAC de destino com o do próximo salto. Considere-se como exemplo o cenário da Figura 4.2 e a Listagem 4.1, que apresenta a tabela de fluxos de MR4 com as regras da aplicação *ping*. Na

¹Site <<http://archive.openflow.org/wp/downloads/>>, Último acesso 20/Maio/2016

²Site <<https://github.com/CPqD/ofsoftswitch13>>, Último acesso 20/Maio/2016

linha dois, que corresponde à requisição ICMP (*ping request*), é possível observar a checagem pelo MAC da interface de acesso de MR4 (`dl_dst=42:00:00:aa:aa:aa`) e a reescrita do MAC de origem pelo MAC da interface da WMN (`mod_dl_src:42:00:00:00:00:04`) e do MAC de destino pelo endereço MAC do próximo salto (`mod_dl_dst:42:00:00:00:00:03`). Procedimento equivalente ocorre na linha três, que corresponde à resposta ICMP (*ping reply*). Tudo isso é pré-computado pelo controlador central quando da ocorrência de um *packet-in* e configurado em cada MR do caminho. Esse processo permite que a interface sem fio do MR opere em modo não-promíscuo, no qual apenas os pacotes destinados àquele nó (*unicast*, *broadcast* ou *multicast*) são processados pela camada superior.

Listagem 4.1: Tabela de fluxos de MR4, conforme Figura 4.2

```

1 root@MR4 ~# ovs-ofctl dump-flows ofsw0 | grep 10.0.0.5
2  cookie=0x0 , duration=70.083s , table=0 , n_packets=64 , n_bytes=6174 ,
   idle_timeout=20 , idle_age=0 , icmp , dl_dst=42:00:00:aa:aa:aa ,
   nw_src=10.0.0.5 , nw_dst=8.8.8.8 , icmp_type=8 , icmp_code=0 actions=
   mod_dl_src:42:00:00:00:00:04 , mod_dl_dst:42:00:00:00:00:03 ,
   output:1
3  cookie=0x0 , duration=70.043s , table=0 , n_packets=63 , n_bytes=6272 ,
   idle_timeout=20 , idle_age=0 , icmp , dl_dst=42:00:00:00:00:04 ,
   nw_src=8.8.8.8 , nw_dst=10.0.0.5 , icmp_type=0 , icmp_code=0 actions=
   mod_dl_src:42:00:00:aa:aa:aa , mod_dl_dst:42:00:00:00:00:05 ,
   output:2

```

Essa estratégia preserva a infraestrutura *mesh* contra ocorrência de *loops*, pois, diferentemente de comutadores Ethernet convencionais³, nos roteadores *mesh*, em geral, a interface de entrada e saída do tráfego é a mesma em cada salto do caminho. Portanto, a checagem de endereços MAC na regra OpenFlow evita que um pacote seja processado em ciclo por dois MR vizinhos. Tomando como exemplo a Figura 4.2 novamente, pode-se analisar a tabela de fluxos dos roteadores *mesh* MR2 e MR3 acerca da aplicação *ping*, retratadas nas Listagens 4.2 e 4.3, respectivamente. É possível notar que, caso não fosse configurada a checagem e modificação do endereço MAC a cada salto, o pacote poderia causar um *loop* na rede, pois MR2 encaminharia o pacote pela interface de entrada, o pacote alcançaria MR3, que também enviaria pela interface de entrada, alcançando mais uma vez MR2 e assim sucessivamente.

³Nos comutadores Ethernet tradicionais, o encaminhamento de tráfego L2 ocorre para uma porta específica quando a localização do MAC de destino é conhecida; ou para todas as portas, exceto a porta de origem, quando um tráfego em *broadcast* ou quando não se conhece a localização do MAC destino.

Listagem 4.2: Tabela de fluxos de MR2, conforme Figura 4.2

```

root@MR2 ~# ovs-ofctl dump-flows ofsw0 | grep 10.0.0.5
cookie=0x0 , duration=70.240s , table=0 , n_packets=64 , n_bytes=6272 ,
idle_timeout=20 , idle_age=0 , icmp , dl_dst=42:00:00:00:00:02 ,
nw_src=10.0.0.5 , nw_dst=8.8.8.8 , icmp_type=8 , icmp_code=0 actions=
mod_dl_src:42:00:00:00:00:02 , mod_dl_dst:42:00:00:00:00:01 , IN_PORT
cookie=0x0 , duration=70.198s , table=0 , n_packets=63 , n_bytes=6174 ,
idle_timeout=20 , idle_age=0 , icmp , dl_dst=42:00:00:00:00:02 ,
nw_src=8.8.8.8 , nw_dst=10.0.0.5 , icmp_type=0 , icmp_code=0 actions=
mod_dl_src:42:00:00:00:00:02 , mod_dl_dst:42:00:00:00:00:03 , IN_PORT

```

Listagem 4.3: Tabela de fluxos de MR3, conforme Figura 4.2

```

root@MR3 ~# ovs-ofctl dump-flows ofsw0 | grep 10.0.0.5
cookie=0x0 , duration=70.240s , table=0 , n_packets=64 , n_bytes=6272 ,
idle_timeout=20 , idle_age=0 , icmp , dl_dst=42:00:00:00:00:03 ,
nw_src=10.0.0.5 , nw_dst=8.8.8.8 , icmp_type=8 , icmp_code=0 actions=
mod_dl_src:42:00:00:00:00:03 , mod_dl_dst:42:00:00:00:00:02 , IN_PORT
cookie=0x0 , duration=70.198s , table=0 , n_packets=63 , n_bytes=6174 ,
idle_timeout=20 , idle_age=0 , icmp , dl_dst=42:00:00:00:00:03 ,
nw_src=8.8.8.8 , nw_dst=10.0.0.5 , icmp_type=0 , icmp_code=0 actions=
mod_dl_src:42:00:00:00:00:03 , mod_dl_dst:42:00:00:00:00:04 , IN_PORT

```

Adicionalmente, o fato do MR não necessitar operar em modo promíscuo melhora sobremaneira a eficiência da rede. As implementações de comutadores SDN com suporte a OpenFlow comumente configuram as interfaces pertencentes a um *datapath* em modo promíscuo, para que possam receber pacotes destinados a qualquer interface (OPENVSWITCH, 2013). Não obstante, essa configuração gera alta sobrecarga de processamento na pilha OpenFlow de cada nó (busca na tabela de fluxos, *packet-in*, ocupação de buffer, instalação de fluxos etc.) e considerável aumento nos pacotes de controle na rede e no controlador. No OpenWiMesh, mesmo havendo MRs com múltiplas interfaces sem fio, não é necessário que essas interfaces sejam configuradas em modo promíscuo. Pelo contrário, recomenda-se manter o filtro de quadros na Camada Física.

Alguns outros mecanismos de gerenciamento da rede *mesh* sem fio, executados pelo controlador, incluem: manter controle sobre os caminhos instalados na rede, permitindo o gerenciamento pró-ativo e até mudança de rota em caminhos sobrecarregados; a partir das informações enviadas pelo *GraphClient*, verificar a consistência na lista de associações mantida no grafo,

removendo arcos cuja associação na WMN não mais exista, e, dessa forma, evitando que esses caminhos ruins sejam escolhidos no algoritmo de roteamento; apresentação visual da topologia da rede, entre outros. O *framework* suporta que outras aplicações ou extensões consumam os dados do grafo da rede, podendo, assim, programaticamente adicionar novas funções ao controlador ou modificar o comportamento da rede.

4.2 CANAL DE SINALIZAÇÃO

O OpenWiMesh utiliza um canal *in-band* para sinalização de controle. O controle *in-band*, ao contrário da estratégia *out-of-band*, proporciona um canal de comunicação eficiente e otimizado entre o roteador *mesh* e o controlador, uma vez que ele não depende de uma infraestrutura secundária e dedicada ao tráfego de controle, reduzindo os custos de implantação. De modo complementar, na estratégia *in-band*, o tráfego de controle pode tirar proveito da mesma lógica de roteamento empregada no tráfego de dados pelo controlador. Todavia, o principal desafio dessa abordagem é o fato de que os enlaces entre MRs precisam estar estabelecidos e os MRs precisam estar registrados no controlador para encaminhar o tráfego de controle ou de dados (Seção 2.3). Para endereçar os desafios dessa abordagem, foram necessárias uma adequação nas regras iniciais do comutador OpenFlow e a criação de uma estratégia de estabelecimento da rede *mesh* sem fio, que serão detalhados a seguir.

A modificação das regras OpenFlow iniciais é indispensável para o correto estabelecimento da rede em modo *in-band*. As implementações de comutadores OpenFlow atuais, comumente, pré-configuram um conjunto de regras para repassar mensagens do protocolo OpenFlow adiante, na expectativa de que elas, em determinado momento, alcancem o controlador OpenFlow. Esse repasse das mensagens deve ocorrer em modo inundação (*flooding*), característica que leva à criação de *loops* na rede e inviabiliza seu uso (Seção 4.1). Para contornar esse problema, o conjunto de regras iniciais do Open vSwitch foi alterado para que o comutador encaminhe apenas o tráfego gerado localmente. A Listagem 4.4 demonstra as regras OpenFlow iniciais em MR2 (Figura 4.2). Nessas regras é possível observar que apenas o tráfego ARP e TCP/6633 (protocolo OpenFlow) possuem correspondência; apenas o tráfego gerado localmente é encaminhado para a interface da rede *mesh* sem fio; e apenas o tráfego destinado àquele MR é aceito para processamento na pilha de rede do sistema.

Listagem 4.4: Regras OpenFlow iniciais em MR2 para estratégia *in-band* do OpenWiMesh

```

root@MR2 ~# ovs-ofctl dump-flows ofsw0
NXST_FLOW reply (xid=0x4):
  cookie=0x0 , duration=38.377s , table=0 , n_packets=1 , n_bytes=42 ,
    idle_age=26 , arp , arp_tpa=10.0.0.2 actions=LOCAL
  cookie=0x0 , duration=38.380s , table=0 , n_packets=9 , n_bytes=378 ,
    idle_age=26 , arp , in_port=LOCAL , arp_tpa=10.0.0.254
    actions=output:1
  cookie=0x0 , duration=38.365s , table=0 , n_packets=216 , n_bytes=38776 ,
    idle_age=0 , tcp , nw_src=10.0.0.254 , nw_dst=10.0.0.2 , tp_src=6633
    actions=LOCAL
  cookie=0x0 , duration=38.370s , table=0 , n_packets=559 , n_bytes=338963
    , idle_age=0 , tcp , in_port=LOCAL , nw_dst=10.0.0.254 , tp_dst=6633
    actions=output:1

```

Para estabelecer as rotas para a rede *mesh* sem fio, com sinalização *in-band*, foi utilizado uma estratégia evolutiva, conforme ilustrado na Figura 4.3. Inicialmente, apenas o comutador SDN do próprio controlador é estabelecido, via interface *loopback* (Figura 4.3a). Em seguida, os próximos nós a estabelecer conexão são os nós vizinhos ao controlador, aqueles que estão a um salto (Figura 4.3b). A vizinhança entre dois nós no meio sem fio se dá pelo alcance de sinal Wi-Fi, pela potência ou ganho de transmissão ou recepção das interfaces, pela interferência do meio e, de forma geral, pela propagação do sinal. Após estabelecer o canal de sinalização, o MR é capaz de encaminhar tráfego em favor de outros nós, através das “consultas de encaminhamento” (*packet-in*) ao controlador. Dessa forma, os nós que estão a dois saltos do controlador (Figura 4.3c) agora podem estabelecer o canal de sinalização. Por fim, os nós que estão a três saltos do controlador (Figura 4.3d) estabelecem o canal de sinalização.

Na Figura 4.3, é possível notar que, durante a inicialização, a topologia da rede assemelha-se a uma árvore de dispersão. À medida que os MRs são ativados, o componente *GraphClient* envia a lista de associações e as informações do meio sem fio de cada MR para o controlador, que atualiza a topologia da rede, transformando-a em um grafo conexo dirigido (Figura 4.3d), descrito na próxima seção.

4.3 GRAFO DA REDE

A rede *mesh* sem fio foi modelada como um grafo dirigido $D = (V, A)$, no qual o conjunto de vértices V é composto pelos MRs e o conjunto de arcos A pelos pares ordenados de associações

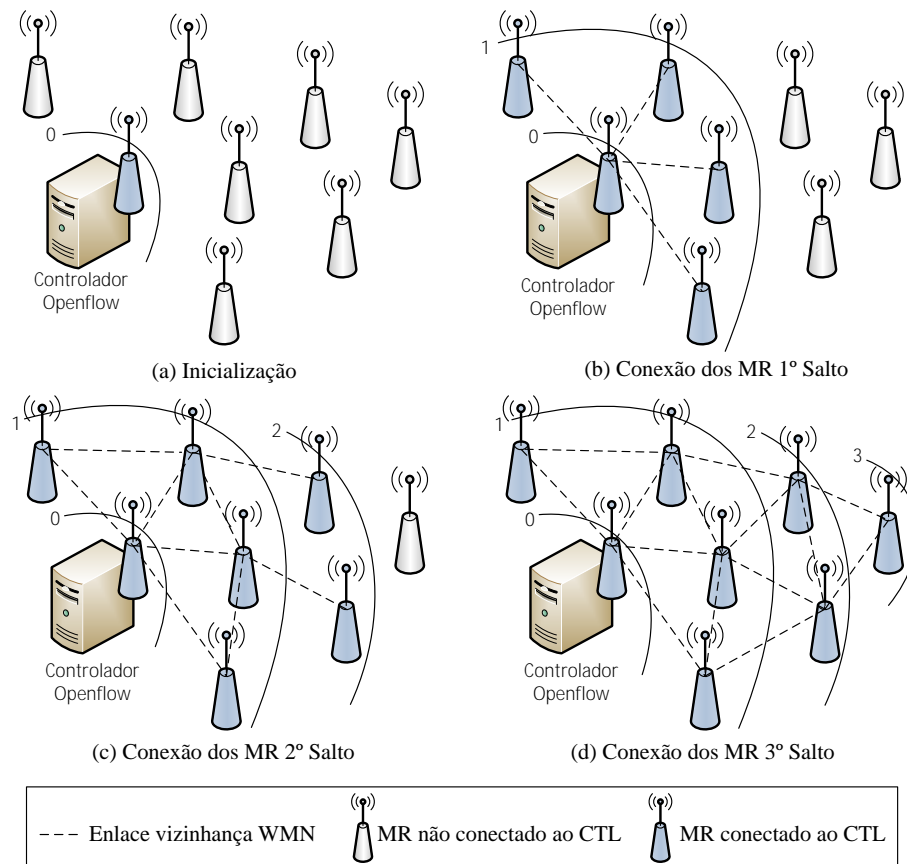


Figura 4.3: Exemplo do estabelecimento da WMN no OpenWiMesh

entre os MRs. Uma associação entre dois MRs (MR_1, MR_2) ocorre quando o sinal de transmissão da interface sem fio de MR_1 alcança MR_2 com nível de potência tal que o permita realizar a demodulação do sinal transmitido. Em uma WMN onde os MRs possuem interfaces sem fio homogêneas, a associação será, na prática, bidirecional. Ao pressupor associações unidirecionais, contempla-se também os cenários com roteadores *mesh* heterogêneos. Utilizou-se a biblioteca Python NetworkX (HAGBERG; SCHULT; SWART, 2008) para armazenar o grafo da rede. O grafo da rede é mantido no controlador central, em consonância com o modelo centralizado de SDN, embora pudesse estar disponível em outro servidor para aumentar o desempenho.

Cada aresta contém uma série de informações sobre a associação sem fio, tais como nível de sinal, nível de ruído, ocupação do enlace, taxa de retransmissão de quadros, dentre outras. O peso da aresta, usado pelos algoritmos de determinação de caminhos no grafo, é derivado dessas informações. O administrador ou usuário privilegiado do OpenWiMesh pode facilmente ajustar a função de cálculo do peso das arestas, para alcançar seus objetivos de engenharia de tráfego.

Os vértices do grafo mantêm informações dos MRs, como seus endereços IP e MAC, portas controladas pelo comutador SDN, lista de nós alcançáveis em cada porta, consumo de memória e CPU etc. O grafo é alimentado e atualizado a partir das informações enviadas pelo *GraphClient* que executa em cada MR. O tempo de atualização das informações do *GraphClient* foi arbitrado em um segundo.

O OpenWiMesh provê uma função de visualização do grafo da rede, inclusive contendo os pesos de cada arco. A visualização do grafo é importante para checar a efetividade do algoritmo de roteamento e também para resolução de problemas. Exemplos de visualização do grafo da rede podem ser encontrados no site do projeto (OpenWiMesh, 2013).

4.4 ALGORITMOS DE ROTEAMENTO

Um componente importante do *framework* é o algoritmo de roteamento, que viabiliza funcionalidades chave para orquestração da WMN, incluindo: engenharia de tráfego, balanceamento de carga dos enlaces, roteamento baseado em fluxos, roteamento baseado em IP, roteamento baseado em políticas administrativas etc. Através de extensões ou modificações no algoritmo de roteamento é possível modificar o comportamento da rede ou implantar uma lógica de encaminhamento completamente diferente do funcionamento tradicional.

O utilizador do *framework* pode valer-se das informações contidas no grafo da rede para definir seu algoritmo de roteamento. Lista de associações, largura de banda projetada, relação sinal/ruído e banda residual são algumas das informações disponíveis no grafo para apoiar o mecanismo de escolha do melhor caminho. Todas essas informações podem ser combinadas, por exemplo, em uma função de cálculo de peso dos arcos, deixando o algoritmo de Dijkstra convencional escolher as melhores rotas com base no caminho mais curto. Pode-se ainda definir outro algoritmo de roteamento, que faça, por exemplo, a busca de caminho apenas em um subconjunto de vértices (caminho baseado em restrições) ou que determine um caminho disjunto de outros caminhos congestionados.

Enfim, a flexibilidade na definição do algoritmo de roteamento em conjunto com a riqueza de informações disponíveis para o experimentador potencializam os casos de uso do *framework*. Na Seção 4.5.3, serão apresentados alguns exemplos de funções de determinação de caminhos e outros casos de uso do OpenWiMesh.

4.5 DESENVOLVIMENTO DO OPENWIMESH

O OpenWiMesh foi implementado com base no controlador SDN POX (POX, 2013), que permite o desenvolvimento de aplicações na linguagem Python e suporta OpenFlow 1.0 como API *southbound*. Além da aplicação no controlador, o *framework* é composto por um agente que executa nos roteadores *mesh*, o *GraphClient*, escrito em C/C++. Todo o código do OpenWiMesh está publicamente disponível na página do grupo de pesquisa GRADE (OpenWiMesh, 2013), podendo ser utilizado para extensões ou estudos comparativos futuros, de acordo com a licença AGPL⁴. Essa seção descreve aspectos importantes do desenvolvimento do *framework*, chamando a atenção para seus pontos de extensão e exemplos de uso com código-fonte.

4.5.1 DIAGRAMA DE CLASSES

A Figura 4.4 retrata de forma simplificada as classes utilizadas no *framework*. No pacote *openwimesh.controller* estão disponíveis as classes utilizadas pela aplicação de controle SDN da rede, composta por nove classes. A principal classe é a *Openwimesh*, ponto de partida para execução das funções de rede do controlador, assim como instanciação de outros objetos de monitoramento e serviços de rede. Já o pacote *openwimesh.agent* é responsável pela coleta de informações do meio sem fio no roteador *mesh*, estendendo as funcionalidades do protocolo OpenFlow. O *GraphClient* foi escrito, em sua maior parte, no paradigma funcional. Suas principais associações são com as bibliotecas *nl80211.h* e *net_snmp.h*, para coleta de informações do dispositivo Wi-Fi e outras informações do sistema, respectivamente.

Abaixo todas as classes são detalhadas para melhor entendimento do leitor:

- **OpenWiMesh** – Essa é a classe base de funcionamento de todo o *framework*, a partir da qual os outros componentes são instanciados. Seu funcionamento, no geral, é orientado a eventos, tanto do OpenFlow quando de atualização de informações do *GraphClient*, a exceção das rotinas de manutenção das informações internas e checagem de conexão dos MRs (*keepalive*). Nessa classe estão as funcionalidades de controle de admissão através de mensagens de ARP ou DHCP, recebimento e tratamento de requisições para novos fluxos (*packet-in*), recebimento das informações do *GraphClient*, encapsuladas em mensagens OpenFlow, controle de *keepalive* dos MRs, busca de caminhos no grafo da rede etc.;

⁴Licença AGPL (do inglês, *GNU Affero General Public License*) disponível em Site<<http://www.gnu.org/licenses/>>, Último acesso 20/Maio/2016

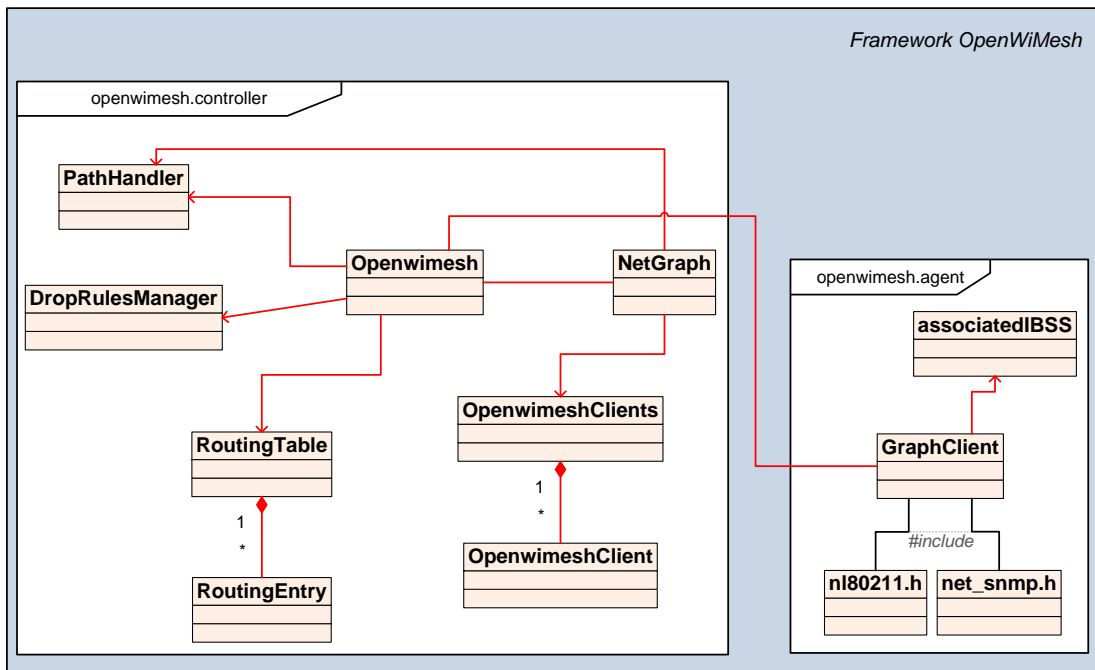


Figura 4.4: Diagrama de Classes do OpenWiMesh

- **NetGraph** – Essa classe possui herança da classe *NetworkX.DiGraph*, estendendo suas funções para a manutenção da topologia da rede *mesh* sem fio (Seção 4.3). A partir dessa classe são executadas tarefas como: inserção e remoção de nós *mesh* no grafo (roteadores e clientes); inserção e remoção de arestas, representando o enlace Wi-Fi entre os nós *mesh*; cálculo da função de peso de uma aresta; busca de caminhos no grafo; atualização de informações e estatísticas nas arestas e vértices, dentre outras;
- **OpenwimeshClient** – Embora os nós clientes façam parte do grafo da rede, seus atributos são armazenados em uma classe especializada, a fim de isolar e definir comportamentos particulares dos clientes, cujo nível de detalhamento é menor que nos MRs;
- **OpenwimeshClients** – Mantém controle sobre uma lista de clientes;
- **RoutingTable** – Mantém controle sobre as rotas IP intra e inter-domínio do *framework*, tanto através de autodescoberta via mensagens ARP, quanto de atualizações enviadas pelo *GraphClient*. Essa classe herda características da biblioteca Radix⁵ do Python, implementando a tabela de rotas através de árvores PATRICIA⁶;

⁵Site <<https://pypi.python.org/pypi/py-radix>>, Último acesso 20/Maio/2016

⁶As árvores PATRICIA, ou árvores *radix*, são estruturas de dados baseadas no tipo *Árvore*, porém com armazenamento otimizado através do agrupamento de nós com apenas um parentesco. Essa estrutura de dados é bastante utilizada para modelar tabelas de roteamento IP.

- **RoutingEntry** – Cada rota é composta pelo prefixo IP, *gateway* e peso. O peso é utilizado para escolha de rotas com o mesmo prefixo enviadas por diferentes MRs;
- **PathHandler** – Essa classe é responsável pelo armazenamento de todas as aplicações ativas na rede, sendo útil para futuras ações de roteamento ou auto recuperação de enlaces com falha. O *framework* pode funcionar com roteamento baseado em fluxos ou baseado em IP, aumentando ou diminuindo a granularidade de controle da rede. No roteamento por fluxos (modo padrão), a lista de aplicações mantidas no *PathHandler* cresce proporcionalmente à carga da rede *mesh* sem fio. Fica a cargo do administrador alterar essa configuração para otimizar o consumo de recursos no controlador;
- **DropRulesManager** – Em cenários onde o roteador *mesh* receba um tráfego inválido de outro nó, o *framework* automaticamente cria uma regra de descarte desse tráfego para evitar sobrecarga na rede. São considerados tráfegos inválidos situações de destino inalcançável, mensagens recebidas incorretamente por um nó vizinho (Seção 4.2) ou mensagens de protocolos desconhecidos;
- **GraphClient** – O *GraphClient* é responsável pela coleta e envio de informações do nó *mesh* para o controlador (Seção 4.3). Essa comunicação é feita de forma acoplada com o canal de sinalização, valendo-se das mensagens OpenFlow *packet-in* para encapsular os dados coletados via API nl80211 ou MIBs SNMP locais;
- **associatedIBSS** – Apesar de não ser exatamente uma classe, essa estrutura de dados desempenha papel importante no *framework*, pois nela estão definidos os parâmetros que o *GraphClient* irá coletar e enviar ao controlador.

4.5.2 PONTOS DE EXTENSÃO

Um *framework* possui pontos com funcionalidades fixa, chamados *frozen spots*, e pontos com funcionalidade variável, chamados *hotspots* (GUERRA, 2014). Os *hotspots* são os pontos de extensibilidade disponibilizados pelo *framework*, onde a aplicação que estende o *framework* pode inserir código novo ou modificar seu comportamento original.

Os pontos de extensão (*hotspots*) do OpenWiMesh são diversos, compreendendo desde as funções de controle e admissão até a coleta, processamento e armazenamento das informações no grafo da rede. Tais extensões são possíveis pela reescrita de funções existentes, pela adição de novas funções ou, ainda, pelo carregamento de outras aplicações SDN no POX e interação com o pacote *openwimesh.controller*. Em particular, os métodos abaixo são mais propícios

a extensão, uma vez que estão diretamente relacionados com funções de rede importantes no *framework* como controle de admissão, atualização do grafo, roteamento etc., a saber:

- *NetGraph* :: *update_edges_of_node()* – Esse método é chamado para cada atualização de informações do meio Wi-Fi enviada pelo *GraphClient*. Ele pode ser estendido para monitorar degradação de enlaces, detecção de *handoff* etc.;
- *NetGraph* :: *path()* – Esse é o método responsável pela descoberta de caminhos no grafo da rede, podendo ser customizado, por exemplo, para implementar mecanismos diferenciados de roteamento na rede;
- *NetGraph* :: *calc_weight()* – Esse método faz o cálculo do peso de uma aresta (enlace entre dois MRs) e pode ser utilizado para influenciar o comportamento padrão da busca de rotas Dijkstra (Seção 4.4);
- *Openwimesh* :: *_handle_PacketIn()* – Função responsável pelo tratamento de mensagens de *packet-in* do OpenFlow. Pode ser utilizado para customizar o controle de admissão dos nós ou para implementar estratégias de roteamento sensível a aplicação, dentre outras;
- *Openwimesh* :: *_handle_graphClient()* – Responsável por tratar as mensagens do *GraphClient*, podendo ser utilizada para adicionar novos itens de monitoramento do nó *mesh* ou disparar ações baseadas em características identificadas em um nó *mesh*;
- *Openwimesh* :: *_install_path()* – Essa função faz a instalação de caminhos na WMN. Sua customização pode agregar funcionalidades como roteamento multi-caminho, alternância entre encaminhamento baseado em fluxos ou baseado em IP, ou, ainda, outro critério a cargo do administrador;
- *Openwimesh* :: *check_assoc_list_changes()* – Nesse método estão disponíveis uma lista de vizinhos inseridos e removidos de um nó *mesh*, e pode ser útil para implementar balanceamento de carga ou realocação de aplicações entre novos enlaces ou enlaces que se tornaram indisponíveis;
- *GraphClient* :: *associatedIBSS* e *GraphClient* :: *handleInterfaceStationInfo()* – Essa estrutura de dados e a função são responsáveis pela coleta de estatísticas do meio sem fio no roteador *mesh*, estendendo a capacidade do OpenFlow. Elas podem ser utilizadas para agregar novos itens de monitoramento ou pré-tratamento das informações que serão enviadas ao controlador.

4.5.3 EXEMPLO DE CÓDIGO

Nesta seção, é apresentado um exemplo de extensão do *framework* OpenWiMesh para customizar a função de roteamento, a fim de implementar uma estratégia de *roteamento baseado em restrições*. O objetivo desse exemplo é ilustrar a viabilidade de customizar a lógica básica de funcionamento da rede *mesh* sem fio, a partir de poucas linhas de código.

A estratégia de *roteamento baseado em restrições* (CBP, do inglês *Constraint Based Path*), consiste do algoritmo de busca tradicional, porém observando-se um conjunto de restrições. Uma solução para esse problema é simplesmente remover do grafo os vértices ou arestas que violam a restrição, para, então, executar a busca convencional sobre o novo grafo. As restrições podem ser, por exemplo, requisitos de QoS, como largura de banda ou atraso, número máximo de saltos, nós incluídos ou excluídos do caminho etc. Para simplificar, a estratégia adotada toma como restrição uma lista de nós excluídos, os quais não devem pertencer a qualquer caminho requisitado.

O primeiro passo é editar a definição da classe *NetGraph*, no arquivo *net_graph.py*, e definir uma nova função conforme ilustrado na Listagem 4.5.

Listagem 4.5: Exemplo de código OpenWiMesh – criação de função CBP

```
def constraint_based_path(self , src , dst , unwanted_nodes
=[]) :
    g2 = self.copy()
    g2.remove_nodes_from(unwanted_nodes)
    path = shortest_path(g2, src , dst)
    return path
```

Em seguida, ainda no *NetGraph*, é preciso definir um parâmetro de controle para uso dessa estratégia, o que pode ser visto nas alterações da Listagem 4.6.

Listagem 4.6: Exemplo de código OpenWiMesh – como chamar função CBP

```

def __init__(self):
    ...
+     # choose which path search algorithm to use
+     self.path_algorithm = None
+     # list of unwanted nodes from path search (useful for
    CBP)
+     self.unwanted_nodes = []

    def path(self, src_ip, dst_ip):
        ...
        src_mac = self.get_by_attr('ip', src_ip)
        dst_mac = self.get_by_attr('ip', dst_ip)
+         if self.path_algorithm == 'CBP':
+             return self.constraint_based_path(src_mac, dst_mac
, self.unwanted_nodes)
            return self.l2_path(src_mac, dst_mac)

```

A partir do código da Listagem 4.6 o *framework* pode, ora executar o roteamento de forma tradicional, ora escolher rotas de acordo com restrições. Para tanto, basta modificar o parâmetro `NetGraph::path_algorithm` e preencher as restrições em `NetGraph::unwanted_nodes`.

4.6 CASO DE USO: ENGENHARIA DE TRÁFEGO EM WMN

O OpenWiMesh foi originalmente desenvolvido para viabilizar técnicas de Engenharia de Tráfego na rede *mesh* sem fio. Entretanto, com base na programabilidade que ele propicia, outros trabalhos estendem sua funcionalidade em variados contextos, por exemplo: roteamento multi-caminho (FERREIRA, 2013), distribuição e replicação de controladores SDN (ARAÚJO, 2015) e distribuição de conteúdo multimídia (SANTOS, 2016).

Engenharia de Tráfego (TE, do inglês *Traffic Engineering*) é um método de melhorar o desempenho da rede a partir de análises dinâmicas de suas características, predição de tendências e auto ajustes no encaminhamento do tráfego de dados na rede. Diversas aplicações de engenharia de tráfego têm sido propostas na literatura, conforme relatado em Kreutz et al. (2015). O principal objetivo da maioria dessas aplicações é influenciar o comportamento do tráfego para minimizar o consumo de energia, maximizar a utilização agregada da rede, aplicar balancea-

mento de carga e outras técnicas de otimização do tráfego em geral.

No OpenWiMesh, a Engenharia de Tráfego foi utilizada principalmente para balancear o tráfego na WMN, melhorando o consumo de energia agregado dos nós *mesh*. Em (BRITO et al., 2014) são propostos três diferentes algoritmos de determinação de caminhos, que se diferenciam quanto ao grau de dispersão do tráfego em diferentes fluxos de dados. Os algoritmos propostos foram:

- i) Contador de Saltos (HC, do inglês *Hop-Count*);
- ii) Maior Banda Residual (HLRB, do inglês *Highest Link Residual Bandwidth*);
- iii) Maior Banda Residual com Mesma Quantidade de Saltos (HLRB-SHC, do inglês *Highest Link Residual Bandwidth in Same Hop Count*).

Usando as informações disponíveis no grafo da rede, diferentes funções podem ser criadas para definir o peso dos arcos e, desse modo, influenciar diretamente na seleção de caminhos. Se um mesmo valor positivo constante for utilizado para o peso dos arcos, define-se uma estratégia que considera apenas a quantidade de saltos entre início e fim do caminho. Essa estratégia é conhecida como *algoritmo de roteamento por contador de saltos* (HC). Nota-se que o mesmo caminho é utilizado para todos os fluxos de um dado par de nós fonte/destino, mesmo que o caminho fique congestionado. Não há, destarte, engenharia de tráfego nessa estratégia, sendo utilizada apenas como comparação com as demais.

O *algoritmo de roteamento por maior banda residual* (HLRB) usa a informação de banda residual do enlace para computar o peso do seu arco. Este algoritmo escolhe, para cada aplicação, um caminho que ofereça a maior banda residual disponível entre nós fonte e destino. Agora, aplicações que trafegam entre o mesmo nó de origem e destino podem tomar diferentes caminhos na rede, otimizando o balanceamento de carga entre os nós *mesh*.

Já o *algoritmo de roteamento por maior banda residual com mesma quantidade de saltos* (HLRB-SHC) combina os elementos dos dois anteriores, buscando selecionar os caminhos de maior banda residual, dentre as alternativas que sejam de menor contador de saltos.

Através de sucessivos experimentos em ambiente emulado, os autores demonstram a viabilidade de influenciar o tráfego da WMN a partir de estratégias implantadas no controlador SDN, segundo pode ser visto na Figura 4.5. Naquela figura é possível observar uma topologia WMN com trinta nós, na qual fluxos de rede são enviados de um nó de origem *n19* (verde) para um nó de destino *n25* (vermelho) e encaminhados na rede de acordo com uma estratégia de

encaminhamento definida pelo controlador SDN, que executa no nó $n5$ (cinza), e cujo caminho é marcado pela linha cheia.

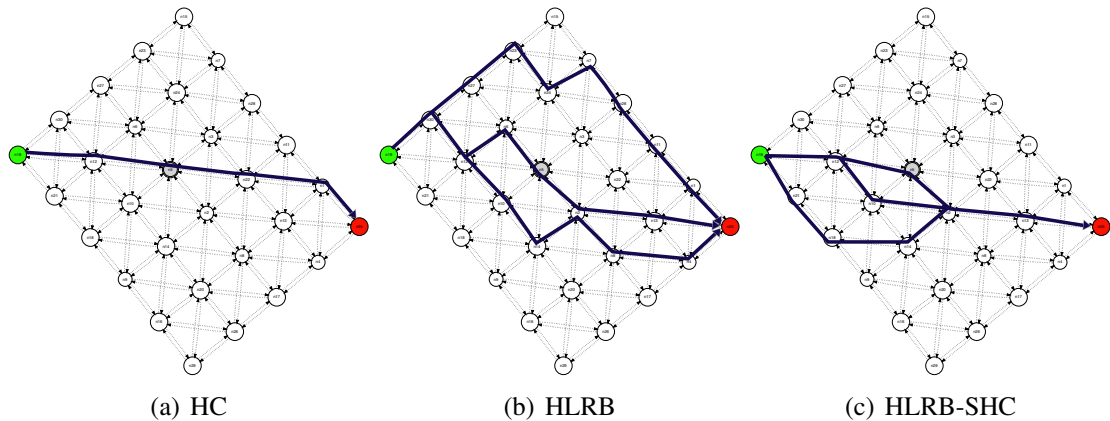


Figura 4.5: Dispersão dos fluxos conforme algoritmo de TE (BRITO et al., 2014)

Na Figura 4.5, é possível observar que a estratégia HC escolhe o menor caminho para diferentes aplicações, podendo repeti-lo e torna-lo congestionado à medida que novas aplicações são instaladas. Já a estratégia HLRB, pode escolher caminhos diferentes para novas aplicações, tornando a rede mais balanceada. Por fim, a estratégia HLRB-SHC busca caminhos alternativos aos previamente utilizados, porém sem aumentar a quantidade de saltos em comparação com o menor caminho.

4.7 CONSIDERAÇÕES PARCIAIS

Neste capítulo foi apresentado o *framework* OpenWiMesh, uma solução para orquestração de WMNs, com controle *in-band* e diversos pontos de extensão, como a customização do algoritmo de roteamento e das métricas de qualidade dos enlaces Wi-Fi, especialização do monitoramento da rede sem fio e até mesmo alterações na sua arquitetura base, com adoção de múltiplos controladores, por exemplo.

O OpenWiMesh será utilizado como base para o desenvolvimento da proposta de mobilidade transparente deste trabalho. Portanto, conhecer sua arquitetura e seus componentes foi oportuno para maior entendimento da estratégia de mobilidade apresentada no próximo capítulo.

5 MOBILIDADE TRANSPARENTE DEFINIDA POR SOFTWARE E CONTROLADA PELA REDE

O gerenciamento do *handoff* consiste em manter a conectividade dos clientes à medida que eles se movimentam na área de cobertura Wi-Fi, garantido a continuidade dos serviços e manutenção da qualidade da rede. O gatilho para início do *handoff* e a transição entre APs devem ser executados de forma rápida e otimizada, para que a mobilidade seja considerada transparente. Além disso, o cliente não deve ser afetado pela exigência de protocolos adicionais ou configurações específicas.

Nesse cenário, o *handoff* controlado pela rede apresenta-se como melhor alternativa ao mecanismo tradicional do 802.11 (SCHULZ-ZANDER et al., 2014), incorporando a visão ampla da infraestrutura e eximindo os dispositivos clientes de participação ativa. O *handoff* controlado pela rede pode ser dividido em três etapas gerais: i) monitoramento, onde realiza-se medições sobre a potência de sinal, distância, qualidade, taxa de erros etc; ii) decisão, onde compara-se os valores medidos com os limiares configurados, podendo levar em consideração margens de confiança para evitar efeitos de *ping-ponging* no *handoff* (PAUL, 2014); e iii) execução, consistindo da sinalização do *handoff*, alocação de recursos nos rádios e reestabelecimento de rotas na rede.

Este capítulo apresenta uma nova estratégia de gerenciamento de *handoff*, utilizando o paradigma SDN e controlado pela rede, que viabiliza mobilidade transparente para clientes não modificados. A proposta foi implementada no *framework* OpenWiMesh (conforme §4), sendo detalhada a seguir. A Seção 5.1 inicia o capítulo apresentando uma visão geral da estratégia de *handoff*. Na Seção 5.2, detalha-se a métrica de monitoramento da qualidade dos enlaces Wi-Fi. A Seção 5.3 aborda o algoritmo de detecção e inicialização do *handoff*. Logo após, na Seção 5.4, discute-se a execução do *handoff*. Por fim, as considerações finais são manifestadas na Seção 5.6.

5.1 VISÃO GERAL DA ESTRATÉGIA DE *HANDOFF*

A estratégia de mobilidade transparente proposta nesse trabalho é ilustrada na Figura 5.1. Nesta figura, é possível notar que o *cliente móvel*, representado por um *smartphone*, passa por 3 fases importantes no processo de *handoff*: na fase 1, o cliente móvel está associado com o MR1 que é o único AP alcançável naquela zona Wi-Fi; ao se mover em direção à direita (fase 2), o cliente móvel passa a ser alcançável pelo MR2 e tem sua qualidade de enlace também monitorada por aquele roteador *mesh*; o cliente móvel continua a mover-se na mesma direção, até que, na fase 3, o controlador SDN detecta que MR2 possui melhor qualidade para atender ao cliente móvel e executa o *handoff*. É também na fase 3 que as rotas do cliente são reconfiguradas. No exemplo ilustrado, a rota entre o cliente móvel e um nó correspondente (e.g. servidor de vídeo na Internet), que antes passava por MR1 e MR3, agora passa por MR2, CTL e MR3.

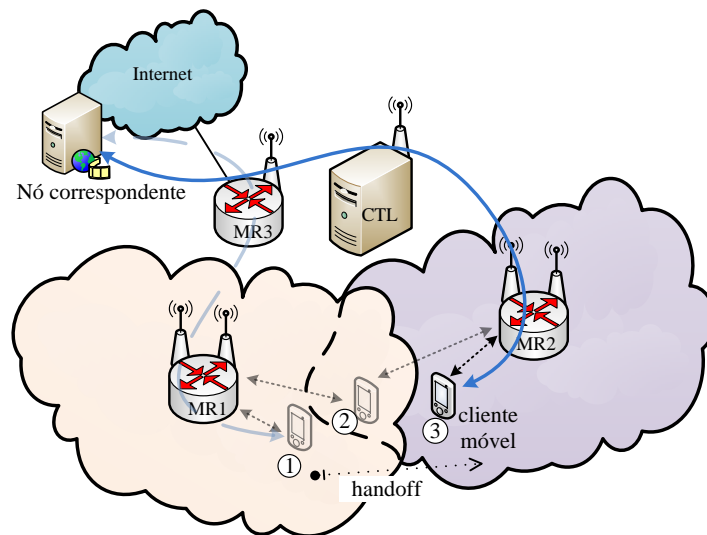


Figura 5.1: Estratégia de mobilidade e *handoff* transparente no OpenWiMesh

Do ponto de vista do cliente móvel, o *handoff* ocorre de forma transparente, não sendo necessária qualquer ação adicional. O cliente móvel continua recebendo os pacotes de controle da rede Wi-Fi (*beacons*), que antes eram enviados pelo MR1, agora originados em MR2. Então, para o dispositivo móvel, ilusoriamente, é como se não houvesse *handoff*, ele apenas percebe uma melhora do sinal e da qualidade da rede. Para que tal cenário seja possível, os pontos de acesso da WMN são especialmente configurados para anunciar a rede Wi-Fi com as mesmas características para cada cliente (SCHULZ-ZANDER et al., 2014), a saber: mesmo canal Wi-Fi; mesmos SSID e BSSID¹; e mesmas chaves de autenticação e criptografia, quando necessário.

¹O BSSID (*Basic Service Set Identification*) é o endereço MAC da interface sem fio do AP, diferente do SSID (do inglês *Service Set Identifier*), que é o nome da rede.

Do ponto de vista do controlador SDN, o gerenciamento da mobilidade ocorre conforme retratado no diagrama de blocos da Figura 5.2. O controlador periodicamente recebe atualizações com informações e estatísticas da rede, através do *GraphClient* para cada MR. A partir dessas informações, o controlador atualiza o grafo da rede, criando e removendo arestas ou atualizando seus atributos. Com base nos atributos da aresta (RSSI, taxa de envio/recebimento, taxa de retransmissão etc), o controlador calcula a qualidade do enlace. Caso a qualidade do enlace esteja dentro de um limiar aceitável, não há o que ser feito. Caso contrário, o *framework* verifica se aquele vizinho é um cliente ou MR. Sendo MR, o *framework* inicia o procedimento de realocação das aplicações do enlace. Em se tratando de cliente, inicia-se a execução do processo de *handoff*: criação da lista de APs candidatos elegíveis, seleção do candidato com melhor capacidade para atender ao cliente e atualização das rotas. Durante a atualização das rotas, adotou-se a estratégia de manter o caminho antigo ativo durante alguns segundos para aumentar a probabilidade de recepção de pacotes pelo cliente móvel (DELY; KASSLER; BAYER, 2011). Embora essa estratégia gere duplicação de pacotes, os benefícios justificam sua adoção, conforme será mostrado na análise de desempenho do Capítulo 6.

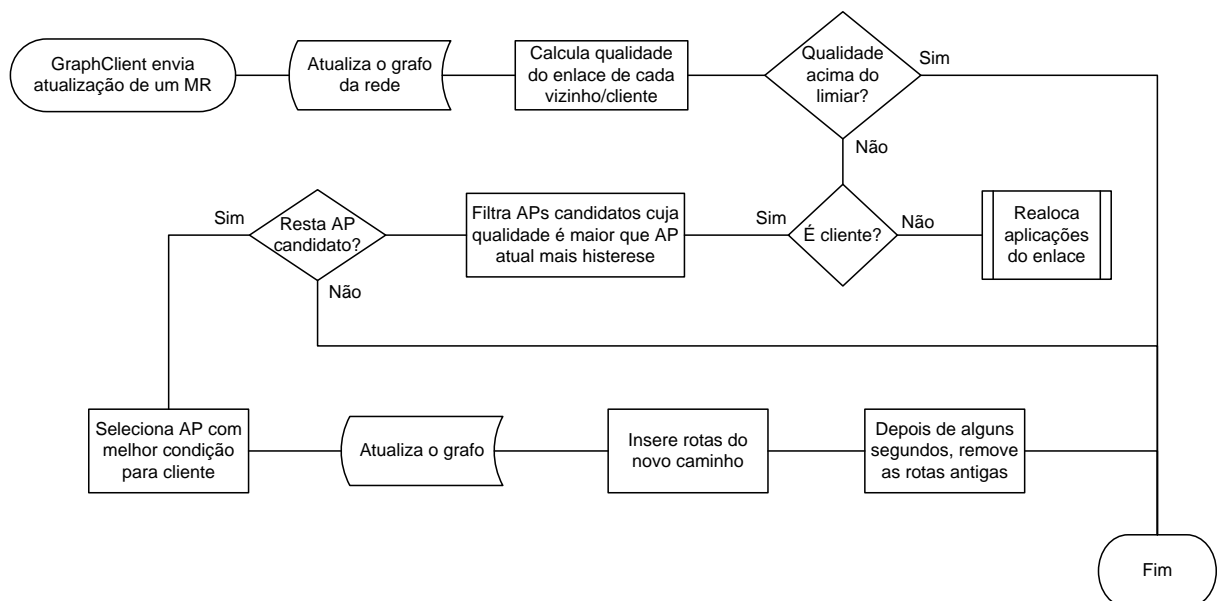


Figura 5.2: Diagrama de blocos do gerenciamento de mobilidade do OpenWiMesh

O papel do controlador SDN, portanto, consiste em monitorar a rede através do componente *GraphClient* e atualização o grafo, calcular a qualidade dos enlaces Wi-Fi, identificar oportunidades de *handoff* e executar o processo de *handoff*. O papel do MR, por outro lado, é executar continuamente o *GraphClient* e enviar as informações da rede para o controlador, executar o comutador SDN com as regras de encaminhamento definidas pelo controlador e executar, quando for o caso, o software AP com os parâmetros da rede sem fio configurados para

permitir a mobilidade transparente.

Considerando que a estratégia supracitada monitora os enlaces Wi-Fi e identifica uma oportunidade de *handoff* antes de sua deterioração, conclui-se que se trata de uma *abordagem proativa de gerenciamento de mobilidade*. Desse modo, torna-se factível alcançar a meta inicial de prover mobilidade transparente e eficiente para os clientes móveis, ao passo que os mantém sem modificações adicionais.

Nas próximas seções esse processo será detalhado para cada uma de suas etapas.

5.2 MÉTRICA DE QUALIDADE DE ENLACES WI-FI

Nesta seção, será detalhado como o OpenWiMesh afere passivamente a qualidade dos enlaces Wi-Fi e como essas métricas são combinadas para decisão do *handoff*. Antes de discutir as métricas e o processo de medição propriamente dito, é importante compreender a diferença entre o monitoramento passivo e ativo, bem como suas vantagens e limitações no contexto de WMN.

A técnica de monitoramento passivo utiliza mecanismos para monitorar o tráfego à medida que ele é encaminhado pela rede, sem interferir no item monitorado. Em outras palavras, no monitoramento passivo, o agente de monitoramento simplesmente coleta dados ou estatísticas já disponíveis no item monitorado e os envia para a estação de monitoramento. A técnica passiva, em essência, não produz tráfego na rede para medição. No entanto, o processo de envio das informações, do agente para a estação de monitoramento, gera tráfego na rede e seu volume é proporcional à quantidade de itens monitorados e ao formato dos dados. As técnicas de monitoramento passivo são eficientes e valiosas para resolução de problemas, embora sejam limitadas quanto à capacidade de reproduzir cenários de erro, ou validar a exata localização de uma falha.

O monitoramento ativo, por outro lado, baseia-se na injeção de pacotes de teste na rede, na produção de conteúdo artificial para medição e na interação com o item monitorado. Dessa forma, ele cria uma carga extra no item monitorado e mede seu comportamento. Em alguns cenários, como em redes sem fio, essa carga extra pode interferir no desempenho. Portanto, no contexto de redes sem fio, quanto mais informações puderem ser coletadas de forma passiva, melhor.

O OpenWiMesh leva em consideração duas métricas no monitoramento dos enlaces: o **RSSI (Received Signal Strength Indicator)** e a **Taxa de Retransmissão de Quadros (FRR)**,

do inglês *Frame Retransmission Rate*). O **RSSI** fornece a medida da potência de sinal Wi-Fi recebido, ao passo que o **FRR** é usado como um indicador indireto da taxa de transmissão alcançável, posto que ela decrementa quando excessivas retransmissões ocorrem. Ambas as informações são coletadas e armazenadas nos *drivers* de interfaces Wi-Fi durante sua operação normal, sendo contabilizadas por associação. Para obtenção dessas informações, alguns *drivers* de interfaces Wi-Fi suportam a API do Kernel Linux *nl80211*². Ela permite obter informações específicas do meio sem fio, a partir de quadros capturados pelo dispositivo Wi-Fi.

Em particular, a respeito das informações supracitadas, é importante destacar:

- **RSSI (Received Signal Strength Indicator)** – RSSI é uma medição da potência de sinal recebido pelo rádio. Os pacotes de controle da rede Wi-Fi (*beacons*) por padrão incluem essa informação e o *framework cfg80211* do Kernel Linux torna essa informação disponível através da API *nl80211*, disponibilizando o RSSI de cada associação (tanto no modo *ad-hoc* quanto *infra-estruturado*). Um ponto de atenção com relação à essa métrica é que sua faixa de valores depende de cada fabricante (AMIR et al., 2010);
- **802.11 Retransmission Flag** – Cada pacote 802.11 transmitido em *unicast* necessita de uma confirmação (ACK) do receptor. Se o pacote ou o ACK forem perdidos na transmissão, o emissor o retransmite e marca a *flag* de retransmissão no cabeçalho 802.11. O valor máximo de retransmissões depende da velocidade da associação e essa velocidade vai sendo reduzida caso não seja confirmado a entrega do pacote. Através da API *nl80211* é possível obter estatísticas sobre a quantidade total de pacotes transmitidos, retransmissões, quantidade de pacotes que falharam (esgotou-se a possibilidade de retransmissão) etc., para cada nó vizinho.

Diferente de trabalhos anteriores, que consideram estas métricas de forma isolada (AMIR et al., 2010; PAUL, 2014; TSUKAMOTO; YAMAGUCHI; KASHIHARA, 2007), neste trabalho é proposto seu uso combinado para calcular a qualidade do enlace Wi-Fi. Destarte, o RSSI contabiliza a potência de sinal do *uplink* do cliente e a taxa de retransmissão afere a qualidade do enlace de *downlink* do cliente. Ambas as métricas são monitoradas de forma passiva no AP, sem sobrecarga adicional. Adicionalmente, as taxas de transmissão e recepção de pacotes, TX_{pps} e RX_{pps} (expressas em *packets per second*), respectivamente, foram também aplicadas para balancear essas métricas, atribuindo pesos para o RSSI e para o FRR de acordo com o volume e direção do tráfego. Por conseguinte, o OpenWiMesh ajusta a importância de cada métrica de forma dinâmica e equilibrada. Os pesos de transmissão ($Weight_{tx}$) e recepção ($Weight_{rx}$) são

²Site <<https://wireless.wiki.kernel.org/en/developers/documentation/nl80211>>, Último acesso 20/Maio/2016

então definidos como:

$$Weight_{tx} = \frac{TX_{pps}}{TX_{pps} + RX_{pps}} \quad (5.1)$$

$$Weight_{rx} = \frac{RX_{pps}}{TX_{pps} + RX_{pps}} \quad (5.2)$$

Para que as métricas RSSI e FRR possam ser combinadas, elas são, antes de tudo, normalizadas. A Equação 5.3 apresenta a normalização do valor do RSSI (medido em dBm) para porcentagem da qualidade do sinal ($Signal_Q$). Esse cálculo é semelhante ao utilizado em alguns sistemas operacionais, como o Microsoft Windows³.

$$Signal_Q = 2 * (dBm + 100) / 100 \quad \text{onde } dBm \in [-100, -50] \quad (5.3)$$

Já a normalização da taxa de retransmissão (FRR) leva em consideração a taxa de pacotes retransmitidos ($Retries$, medido em pps) em comparação com o total de transmitidos, conforme Equação 5.4.

$$FRR = \frac{Retries_{pps}}{TX_{pps} + Retries_{pps}} \quad (5.4)$$

A qualidade do enlace Wi-Fi é expressa na Equação 5.5. Nessa equação, além de considerar o RSSI e taxa de retransmissão, balanceados pelos valores de $Weight_{rx}$ e $Weight_{tx}$, considera-se também o tempo de inatividade das estatísticas informadas pelo *nl80211*, pois quando um nó deixa de receber pacotes de controle (*beacons*) de determinado vizinho (por degradação do sinal ou movimentação do nó), o tempo de inatividade passa a incrementar diminuindo a importância das estatísticas.

$$Link_Q = \frac{Weight_{tx} * (1.0 - FRR) + Weight_{rx} * Signal_Q}{inactiveTime} \quad (5.5)$$

A métrica acima produz uma visão instantânea da qualidade do sinal Wi-Fi. Não obstante, devido às oscilações do meio sem fio (WU; DJUKIC; MOHAPATRA, 2008), esses valores podem variar e causar o efeito *ping-ponging* (PAUL, 2014). Com objetivo de evitar essas oscilações, a programação dessas funções pode fazer uso da média do RSSI no controlador, utilizar a média de RSSI disponibilizada pela API *cfg802.11* ou ainda aplicar uma média móvel ao valor de $Link_Q$. Nesta dissertação utilizou-se uma média móvel de tamanho σ , com valor arbitrado em 5.

³ Site<[https://msdn.microsoft.com/en-us/library/windows/desktop/ms706828\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms706828(v=vs.85).aspx)>, Último acesso 20/Maio/2016

Esse valor da qualidade do enlace é, então, utilizado como gatilho para o *handoff* dos clientes ou para um processo de realocação de aplicações em arestas que apresentem valor de $Link_Q$ abaixo de um limiar. O limiar é configurável pelo administrador, variando entre 0,0 e 1,0 (o padrão é 0,80).

5.3 ALGORITMO DE DECISÃO DE *HANDOFF*

A métrica de qualidade do enlace previamente apresentada é calculada a cada atualização do *GraphClient*, cujo intervalo padrão é um segundo. A partir do valor de qualidade do enlace, o *framework* pode tomar a decisão de executar o *handoff* dos clientes, desde que haja candidatos elegíveis para a transição de AP (Fig. 5.2). A rotina `pooler_check_assoc_list()` é responsável pela checagem da lista de vizinhos de um MR (incluindo os clientes), pelo cálculo da métrica de qualidade e pela verificação do limite de qualidade, culminando na decisão de *handoff*. Essa rotina verifica se o $Link_Q$ está abaixo de determinado limiar T (*threshold*) e se existe outro vizinho com melhores condições de atender ao cliente móvel.

A rotina `check_handoff()`, por sua vez, é responsável pela identificação de um AP alvo para a transição do cliente móvel. Primeiramente, é feita uma seleção de APs candidatos; notoriamente, seleciona-se aqueles com melhor qualidade de enlace que a associação atual e cuja margem de diferença seja maior que H (histerese). Essa abordagem é semelhante a apresentada em (PAUL, 2014), chamada de *Relative Signal Strength with Threshold and Hysteresis*; porém, aqui expandida levando-se em consideração não somente o RSSI, mas também a taxa de retransmissão de pacotes, ambos ponderados com a taxa de utilização do enlace (Tx/Rx), em razão do tempo de inatividade, como discutido previamente.

O tempo de inatividade contabiliza o atraso entre o envio de mensagens de controle do 802.11 (*beacons*), podendo indicar sobrecarga do nó, distância entre os nós (nó fora de alcance) ou até mesmo condições ruins do enlace. Os *beacons* são enviados em intervalos regulares para manter a coordenação com os pontos de acesso Wi-Fi, geralmente com periodicidade de 100 *unidades de tempo*⁴, embora ajustável. Quanto maior o tempo de inatividade, menor a qualidade do enlace e confiabilidade das informações de RSSI e FRR do vizinho.

A Figura 5.3 ilustra a coleta de estatísticas de um dispositivo 802.11 acerca de seus vizinhos, com destaque para as métricas supracitadas. Naquele exemplo foi utilizado o comando *iw*, que também é baseado na API `nl80211` do kernel Linux. Os valores de pacotes transmitidos (*tx*

⁴Uma *Unidade de Tempo* (TU, do inglês *Time Unit*) equivale a cerca de 1024 microssegundos, ou ainda, 100 milissegundos. Em alguns casos prefere-se utilizar a métrica TU invés de milissegundos pois a primeira é mais fácil implementar em equipamentos com *clock* de 1 MHz

packets), pacotes recebidos (*rx packets*) e retransmissões (*tx retries*) são cumulativos, sendo necessário calcular a diferença entre as amostras para obter as respectivas taxas.

```

fish /home/italo
Arquivo Editar Ver Pesquisar Terminal Ajuda
italo@pc -> sudo iw dev wlan4 station get c4:12:f5:48:33:4b
Station c4:12:f5:48:33:4b (on wlan4)
  inactive time: 28 ms
  rx bytes: 25841818
  rx packets: 61284
  tx bytes: 5750173
  tx packets: 36855
  tx retries: 19098
  tx failed: 2178
  signal: -55 dBm
  signal avg: -56 dBm
  tx bitrate: 57.8 MBit/s MCS 5 short GI
  rx bitrate: 26.0 MBit/s MCS 9
  expected throughput: 25.835Mbps
  authorized: yes
  authenticated: yes
  preamble: long
  WMM/WME: yes
  MFP: no
  TDLS peer: no
italo@pc ->

```

Figura 5.3: Exemplo de coleta de estatísticas 802.11 via comando *iw*

O pseudocódigo apresentado na Listagem 5.1 implementa a lógica de checagem da qualidade do enlace, através da função `pooler_check_assoc_list(node)`, e o algoritmo de decisão de *handoff*, por meio da função `check_handoff(node, neigh)`, ambas previamente mencionadas. A função `link_quality(node, neigh)` calcula a qualidade de enlace $Link_Q$ para o arco $(node, neigh)$, conforme Seção 5.2. Já a função `execute_handoff(neigh, target_ap)`, é responsável pela execução do *handoff* e será detalhada na próxima seção.

```

FUNCTION pooler_check_assoc_list(node):
  assoc_list = graphClient.get_assoc_list(node)
  FOR neigh IN assoc_list:
    IF link_quality(node, neigh) < T:
      target_ap = check_handoff(node, neigh)
      IF NOT target_ap.empty():
        execute_handoff(neigh, node, target_ap)

FUNCTION check_handoff(node, neigh):
  IF NOT is_client(neigh) OR neigh.get_ap() != node:
    RETURN
  eligible_cand = []
  cur_linkq = link_quality(node, neigh)
  FOR cand IN ap_candidates(neigh):
    IF link_quality(cand, neigh) > cur_linkq + H:
      capacity = calc_capacity(cand)

```

```

eligible_cand.append([cand, capacity])

sorted_eligible_cand = sort(eligible_cand)

RETURN sorted_eligible_cand.first()

```

Listagem 5.1: *Pseudocódigo para detecção e decisão de handoff*

Após a escolha dos APs candidatos elegíveis, deve-se selecionar um deles a ser utilizado como AP alvo no processo de *handoff*. A escolha de qual AP melhor atenderá ao cliente em *handoff* deve levar em consideração a capacidade dos APs candidatos em suportar o novo cliente. Para o cálculo da capacidade disponível em um AP, pode-se utilizar diferentes métricas como vazão (e.g. vazão máxima teórica, mínima, vazão derivada), CPU ou memória disponível no AP, quantidade de clientes associados etc. Nesta dissertação, optou-se por utilizar a média da porcentagem de banda residual entre todas as interfaces do AP *mesh*. A Equação 5.6 apresenta a abordagem de cálculo da capacidade de um nó.

$$calc_capacity = \frac{1}{n} \sum_{iface=1}^n 1 - \frac{cur_throughput_{iface}}{max_speed_{iface}} \quad (5.6)$$

Uma vez identificada a oportunidade de *handoff* e escolhido o AP alvo, a próxima etapa do processo de *handoff* é executar a transição propriamente dita, realizando as configurações necessárias na infraestrutura *mesh* para acomodar o cliente móvel em sua nova localização. A rotina de execução do *handoff* será explicada na próxima seção.

5.4 EXECUÇÃO DO *HANDOFF*

A migração de um cliente móvel, do seu AP atual para um AP alvo, demanda alguns ajustes na infraestrutura *mesh* a fim de manter a conectividade do cliente, a saber: provisionamento de recursos no AP alvo e reconfiguração de rotas para as aplicações do cliente. A estratégia utilizada na execução desses ajustes tem relação direta com a eficiência do *handoff*. Caso o AP alvo, por exemplo, não esteja previamente configurado para enviar os *beacons* com os parâmetros esperados pelo cliente, o cliente inicia o processo de *handoff* convencional do 802.11, com re-escaneamento e reassociação, degradando sobremaneira o desempenho da mobilidade (Seção 3.1.1). Outrossim, caso as rotas para a nova localização do nó não estejam corretamente estabelecidas quando da transição, a pilha de rede pode experimentar degradação devido ao controle de entrega e retransmissão da camada de transporte e até encerramento de conexões

TCP, por exemplo.

As estratégias propostas para endereçar tais desafios consistem em: realocar, de forma eficiente e proativa, as regras SDN para aplicações ativas ao detectar uma oportunidade de *handoff*; manter ativas as rotas antigas do nó, a fim de aumentar a probabilidade de entrega de pacotes; configurar, também de forma proativa, os parâmetros 802.11 no AP alvo, conservando a transição entre APs transparente ao cliente.

No que se refere ao roteamento, o OpenWiMesh mantém informações sobre todas as aplicações e rotas ativas na rede, indexando-as nos arcos do grafo. Em particular, cada arco entre um cliente e seu respectivo AP possui uma lista de índices para as *aplicações ativas*. Por *aplicações ativas* entende-se as regras da tabela de fluxos dos MRs. Caso o OpenWiMesh esteja configurado para fazer o roteamento baseado em fluxos, haverá tantas regras de encaminhamento quanto o número de conexões simultâneas do cliente. De outro modo, quando o OpenWiMesh está configurado para fazer o roteamento baseado em IP, haverá apenas uma regra de encaminhamento, cuja correspondência de cabeçalhos consiste nos endereços IP envolvidos na comunicação. Dessa maneira, na execução do *handoff*, o *framework* processa a lista de aplicações e, para cada aplicação, busca um novo caminho no grafo contemplando o AP alvo e o cliente. Em seguida, uma regra de encaminhamento para a aplicação em questão é configurada em cada MR do caminho. A reconfiguração das rotas ocorre através de mensagens OpenFlow do tipo *flow-mod*.

A reconfiguração de rotas ocorre, primeiramente, criando as novas rotas, para somente então remover as rotas antigas. Esse detalhe na ordem de execução do *handoff* é importante para diminuir o impacto no tráfego das aplicações, deixando-o assim mais transparente ao cliente. Na literatura, essa metodologia é conhecida como *make before break* (criar e depois desfazer) ou *soft-handoff* (PAUL, 2014), sendo utilizada nas redes de celulares CDMA e W-CDMA, mas não nas redes 802.11. Com a metodologia utilizada, é possível importar esse comportamento para as redes 802.11. Ademais, a fase de remoção das rotas é postergada por alguns segundos a fim de manter o caminho antigo ativo e assim aumentar a probabilidade de recepção de pacotes no cliente.

Em termos da associação Wi-Fi, como utiliza-se um único BSSID nos APs, não é necessária qualquer sinalização ou ação de reassociação no cliente. Na infraestrutura *mesh*, basta configurar o AP alvo com a rede Wi-Fi do cliente em transição. A configuração dos nós *mesh* com essa característica é bem simples, sendo suficiente provisionar ou conservar no AP alvo parâmetros como: SSID, BSSID, canal Wi-Fi e, caso utilize criptografia, chave da sessão. Nesta

dissertação foi utilizado o software AP *hostapd*⁵ para criação da rede Wi-Fi *infra-estruturada* dos clientes, sem criptografia.

A rotina de execução de *handoff* discutida acima é apresentada no pseudocódigo da Listagem 5.2. A principal função é `execute_handoff()` que recebe como argumentos o cliente móvel (*client*), o AP atual (*cur_ap*) e o AP alvo (*target_ap*). O procedimento consiste, basicamente, em: *linha 2*) configurar os parâmetros Wi-Fi no AP alvo e atualizar as estruturas de dados internas; *linha 3*) criar um arco no grafo entre o cliente e o AP alvo; *linhas 4 e 5*) reconfiguração de rotas ou realocação das aplicações; *linha 6*) armazenar o histórico do *handoff* a fim de evitar efeitos *ping-ponging* ou, futuramente, incorporar técnicas de previsibilidade de *handoff*.

```

FUNCTION execute_handoff(client , cur_ap , target_ap)
    net_graph.clients.update_ap(client , target_ap)
    net_graph.add_edge(client , target_ap)
    relocate_apps_using_edge(cur_ap , client)
    relocate_apps_using_edge(client , cur_ap)
    net_graph.clients.handoff_history(client , cur_ap)

FUNCTION relocate_apps_using_edge(u, v):
    apps = net_graph.edge[u][v]['apps']
    net_graph.remove_edge(u, v)
    FOR app IN apps:
        old_path = app['path']
        first_hop = app['ether_src']
        IF net_graph.is_client(first_hop):
            first_hop = net_graph.clients.get_ap(first_hop)
        _install_path(first_hop , app['ether_src'] , app['match'])
    FOR sw IN set(old_path) - set(app['path']):
        Timer(5.0 , _delete_flow_entry , (sw , app['match']))

RETURN

```

Listagem 5.2: Pseudocódigo para execução do *handoff*

A rotina de realocação das aplicações (`relocate_apps_using_edge()`), por sua vez, toma como entrada o arco no qual as aplicações que estão indexadas precisam ser realocadas e executa as tarefas necessárias para migrar essas aplicações para outro arco. É importante destacar que sempre existirá um caminho alternativo, uma vez que o AP atual não faz parte da eleição de APs candidatos e o grafo da rede é conexo. São executadas diversas sub-tarefas de migração

⁵O *hostapd* (do inglês, *host access point daemon*) é um software capaz de transformar uma interface de rede Wi-Fi convencional em um ponto de acesso e servidor de autenticação (via RADIUS, WPA, WPA2, etc). Mais informações: Site<<https://w1.fi/hostapd/>>, último acesso 20/Maio/2016

das aplicações, como: remoção do arco atual, para que não participe da busca no grafo; busca de novo caminho; instalação de fluxos do novo caminho. Chama-se a atenção para as linhas 17 e 18, que contemplam a estratégia de remoção das regras de encaminhamento do caminho antigo: i) é possível que um trecho do novo caminho seja idêntico ao caminho antigo, portanto, deve-se mantê-lo ativo e conforme reconfiguração recentemente executada; ii) a biblioteca de *threads* disponibiliza a função *Timer*, que foi utilizada a fim de postergar a remoção das regras antigas por alguns segundos (cinco segundos). Conforme previamente discutido, através dessa estratégia é possível aumentar a probabilidade de recepção de pacotes do nó móvel durante o *handoff*.

5.5 DESENVOLVIMENTO DO PROTÓTIPO

As estratégias de monitoramento, detecção e execução do *handoff* apresentadas anteriormente foram empregadas no desenvolvimento de um protótipo de gerenciamento de mobilidade, usando como base o OpenWiMesh (cf. §4). O protótipo foi chamado **OpenWiMesh-Mob** e seu desenvolvimento será detalhado a seguir. Uma vez que o gerenciamento de mobilidade é uma funcionalidade chave para o *framework*, o código-fonte do protótipo foi incorporado ao núcleo do OpenWiMesh, o qual está publicamente disponível em (OpenWiMesh, 2013).

A Figura 5.4 apresenta uma nova versão do diagrama de classes discutido anteriormente (cf. Fig. 4.4), incorporando as mudanças necessárias para o desenvolvimento do protótipo. Nesse novo diagrama, as classes adicionadas e modificadas estão em destaque. Apenas uma nova classe foi criada, a classe *OpenwimeshAPCands*, para modelar a lista de APs candidatos no processo de *handoff* de um cliente. Tal listagem é construída a partir das mensagens do *GraphClient*, em momento anterior ao *handoff*. As demais classes modificadas são detalhadas a seguir.

O registro das mudanças realizadas no OpenWiMesh para adicionar o suporte à mobilidade são listadas abaixo:

- Alteração na estrutura de dados *associatedIBSS* a fim de adicionar novas estatísticas a serem coletadas e utilizadas para cálculo da qualidade do enlace Wi-Fi, a saber: tempo de inatividade e taxa de retransmissão. Foi necessário modificar também algumas funções que fazem referência a essa estrutura de dados, para que os dados pudessem ser corretamente coletados, empacotados e enviados de “carona” com as mensagens OpenFlow;
- Criação da classe *OpenwimeshAPCands* para armazenar a lista de APs candidatos por

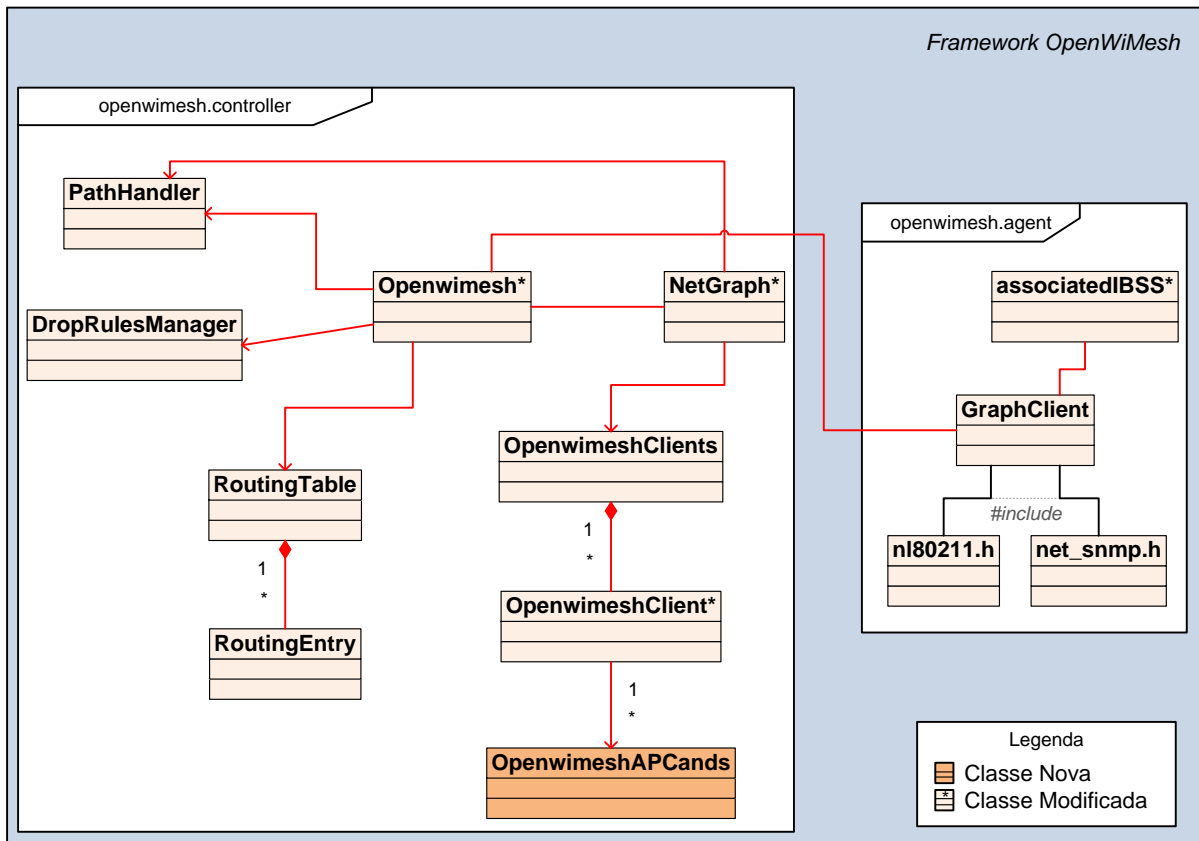


Figura 5.4: Diagrama de Classes do OpenWiMesh com Mobilidade

cliente;

- Inclusão de novas funções na classe *OpenwimeshClients*, principalmente, para: atualizar o AP de um cliente, armazenar o histórico de *handoff*, gerenciar os APs candidatos;
- Alteração da classe *NetGraph*, em particular, com a definição da função de cálculo da qualidade do enlace (cf. §5.2) e com alterações na função de processamento da lista de atualizações, visando i) executar o cálculo da qualidade de enlace em si, e ii) incorporar um novo valor de retorno – a lista de vizinhos cujo enlace está abaixo do limiar de qualidade configurado (*poorly_nodes*);
- Na classe *Openwimesh*, por sua vez, foi necessário alterar a função inter-relacionada com o *NetGraph*, a fim de receber o valor de retorno mencionado anteriormente (*poorly_nodes*) e executar a checagem de *handoff* para esses nós. Na classe *Openwimesh* foram incorporados, ainda, os algoritmos de decisão de *handoff* (cf. §5.3) e de execução do *handoff* (cf. §5.4).

A partir das mudanças acima foi possível prover o serviço de gerenciamento de mobilidade

aos clientes da rede *mesh*, de acordo com a estratégia ora proposta. Ao iniciar o *framework*, as funcionalidades de gerenciamento de *handoff* são carregadas com seus valores padrões. Caso o administrador deseje ajustar esses valores para as características da sua rede, os seguintes parâmetros de configuração estão disponíveis:

- **Intervalo de atualização do *GraphClient*** – Por padrão, o *GraphClient* envia as atualizações a cada um segundo. Esse é um tempo razoável ao considerar-se que a estratégia de *handoff* proposta atua de forma pró-ativa. No entanto, caso o administrador queira diminuir ou aumentar esse intervalo, será necessário alterar o script de inicialização do *GraphClient* e alterar um parâmetro de inicialização do controlador;
- **Limiar de *handoff*** – esse é o limiar para que um enlace seja considerado de baixa qualidade, de acordo com a métrica previamente definida (cf. §5.2). Esse limiar é um valor decimal entre 0.0 e 1.0 e representa a porcentagem de qualidade do enlace abaixo da qual ele deve ser realocado. O padrão é 0.80;
- **Histerese de *handoff*** – de forma complementar ao limiar de qualidade acima, o valor da histerese define uma margem de segurança que o AP candidato deve apresentar em relação ao AP atual para ser considerado elegível no processo de *handoff*. Esse valor possui o mesmo domínio de definição do limiar de qualidade. Por padrão, utiliza-se uma histerese de 0.05;
- **Roteamento baseado em IP ou fluxos** – essa configuração é indiretamente relacionada ao *handoff*, especificando uma lista de nós (clientes ou MR) cujo roteamento será baseado em IP. Os demais nós terão roteamento baseado em fluxo (padrão do *framework*). Por padrão, todos os nós *mesh* possuem roteamento baseado em fluxo.

5.6 CONSIDERAÇÕES PARCIAIS

Neste capítulo, foi apresentada uma nova estratégia de *handoff* controlado pela rede e definido por software. Tal estratégia endereça desafios importantes do processo de gerenciamento de mobilidade, como o monitoramento e detecção de oportunidade de *handoff*, seleção de APs para migração de conexão, migração de clientes entre APs e realocação de aplicações após o *handoff*. Dessa maneira, é possível contemplar clientes não modificados e prover qualidade nas associações Wi-Fi da infraestrutura *mesh* como um todo, inclusive face à mobilidade dos nós.

Um protótipo da estratégia proposta foi desenvolvido no *framework* OpenWiMesh, valendo-se de sua arquitetura e de seus componentes para incorporar o suporte à mobilidade transparente

dos clientes móveis. Nesse sentido, a riqueza de informações presentes no grafo da rede, a capacidade de monitoramento agregada pelo *GraphClient*, a flexibilidade e programabilidade do controle SDN e, principalmente, a possibilidade de uso de técnicas e algoritmos inovadores para o gerenciamento de mobilidade são os diferenciais para viabilizar os objetivos da proposta.

No próximo capítulo, a estratégia ora proposta será avaliada em comparação com uma abordagem tradicional, a fim de medir seu desempenho e impacto aos usuários móveis.

6 AVALIAÇÃO

A proposta de gerenciamento de mobilidade, apresentada no Capítulo 5, possui elementos importantes para melhorar o desempenho e a qualidade do processo de *handoff* em redes *mesh* sem fio. Não obstante, é importante avaliar os algoritmos e métricas propostas quanto à sua efetividade e impacto, tanto do ponto de vista da infraestrutura quanto do usuário propriamente dito. Outrossim, é oportuno que tal avaliação proporcione uma comparação com outra estratégia de semelhante objetivo, a fim de fornecer uma linha base para conclusões e inferências acerca dos resultados.

Este capítulo descreve o estudo avaliativo e comparativo do trabalho. Inicia-se pela caracterização do ambiente de testes utilizado (Seção 6.1) e, em seguida, apresenta-se a metodologia adotada (Seção 6.2). Nas Seções 6.3 e 6.4 apresenta-se a avaliação da proposta por meio de testes em ambiente real, considerando métricas de qualidade de serviço e qualidade de experiência, respectivamente. Na seção 6.5, são avaliadas diferentes configurações e parametrizações da proposta. Por fim, uma análise geral dos resultados é produzida na Seção 6.6.

6.1 AMBIENTE DE TESTES

Para avaliar a qualidade e efetividade da estratégia proposta, uma série de experimentos e medições foram conduzidos. Para tanto, a arquitetura proposta foi implementada em um *testbed* 802.11 com suporte a OpenFlow. A fim de prover uma linha base de comparação, foi utilizada uma segunda estratégia de gerenciamento de mobilidade, composta por tecnologias comumente abordadas na literatura, quais sejam: pontos de acesso 802.11, configurados de maneira convencional, funcionando no modo *infra-estruturado* e divulgando o mesmo SSID e diferentes BSSID's; protocolo de roteamento B.A.T.M.A.N.. Nesse cenário, nomeado BATMAN-MCHO (*B.A.T.M.A.N. with Mobile Controlled Handoff*), os clientes móveis controlam o processo de *handoff*. Por outro lado, no mecanismo de *handoff* proposto neste trabalho, chamado OpenWiMesh-Mob, a mobilidade é controlada pela rede e transparente para o cliente.

O *testbed* abrange um controlador (CTL) e dois roteadores *mesh* (MR). Os roteadores *mesh* foram implantados a partir de estações de trabalho convencionais, equipados com CPU Intel Core i5, 8GB de memória, S.O. Debian 8.3, interface sem fio nativa modelo Atheros AR9485 e uma interface sem fio adicional TP-Link WN721N. Uma dessas interfaces Wi-Fi foi utilizada para encaminhar pacotes da rede *mesh*, ao passo que a outra foi utilizada como ponto de acesso para os clientes, a partir do software *hostapd*¹. O controlador executa em um notebook Intel com hardware similar aos roteadores *mesh*, assumindo também o papel de Nó Correspondente (CN). O CN é o nó com quem o Cliente Móvel (MC, do inglês *Mobile Client*) irá se comunicar durante os testes. O CN hospeda o serviço de transferência de vídeo e servidor *iperf* (NLANR/DAST, 2013). O cliente móvel foi baseado em três das plataformas mais utilizadas atualmente como sistema operacional em computadores pessoais: Linux, Windows e MAC OS X. As três plataformas foram utilizadas para validar o comportamento do OpenWiMesh-Mob e do BATMAN-MCHO com relação ao gerenciamento de *handoff*. Em particular, é esperado que o OpenWiMesh-Mob seja agnóstico ao cliente, não demandando nenhum recurso ou configuração adicional.

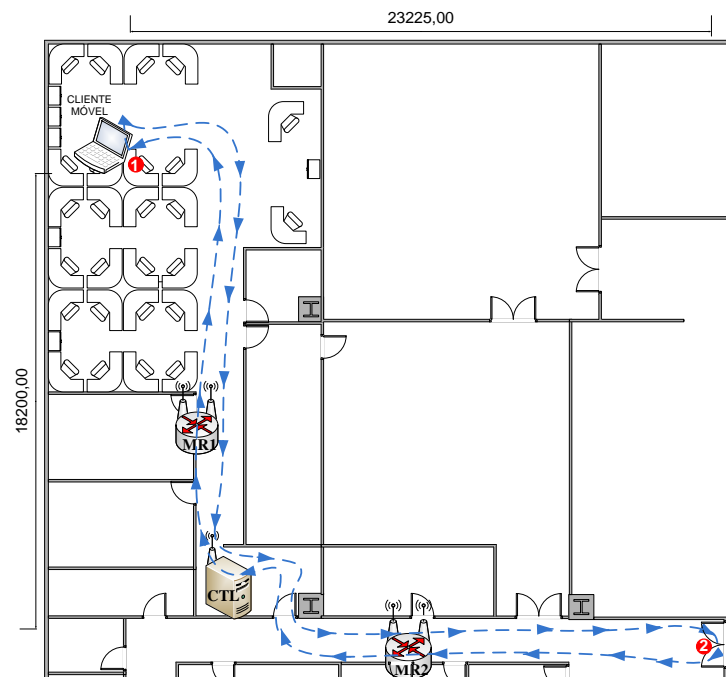


Figura 6.1: Planta baixa do ambiente de teste

A Figura 6.1 mostra a planta baixa da área na qual os testes foram executados. Nesse cenário, tem-se uma topologia de rede *mesh* sem fio com dois roteadores *mesh* e um controlador, atuando também como CN. O ambiente de teste foi implantado em um prédio administrativo

¹ Site <<https://w1.fi/hostapd/>>, Último acesso 20/Maio/2016

da Universidade Federal da Bahia, compreendendo uma área de mais de quatrocentos e vinte metros quadrados, com paredes, vidros, divisórias em madeira etc. Esse cenário é comum em outros prédios da universidade; portanto, sua escolha generaliza outros ambientes de possível uso da solução. Na figura acima, é possível visualizar o posicionamento dos roteadores *mesh* (MR1 e MR2) e do controlador e nó correspondente (CTL).

Acerca do ambiente de testes, outra informação relevante é a intensidade de sinal Wi-Fi na área do cenário, bem como os pontos de sobreposição de sinal. A Figura 6.2 retrata um mapa de estimativa de intensidade do sinal, medido em dBm, do ponto de vista de MR1 (Fig. 6.2a) e de MR2 (Fig. 6.2b). O mapa foi construído com base em sucessivas coletas ao longo da área em questão, totalizando mais de quarenta amostras, com metodologia similar ao aplicado em Avelar (2013).

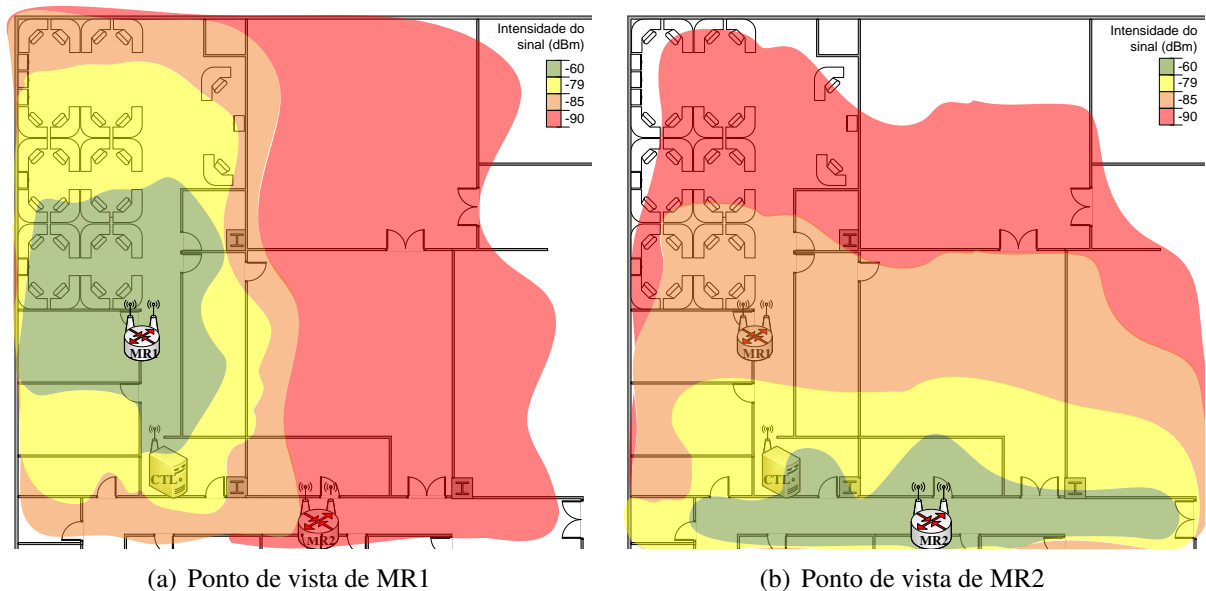


Figura 6.2: Mapa de intensidade de sinal (dBm) no ambiente de teste

6.2 METODOLOGIA

A metodologia de teste consistiu em mover-se ao redor da área de cobertura dos roteadores *mesh*, de forma a proporcionar o *handoff* entre os pontos de acesso. Para isso, o experimentador movimentou-se a pé no trajeto ilustrado na Figura 6.1, com velocidade aproximada de um metro por segundo (equivalente à uma pessoa em caminhada). Cada experimento teve duração de 200 segundos e foi repetido 10 vezes.

A Figura 6.3 delinea o diagrama de mobilidade do experimento, sendo composto por cinco

trechos: *trecho 1*) o MC inicia no ponto *P1* e permanece lá por 40 segundos; *trecho 2*) o MC move-se para o ponto *P2* com velocidade de 1 m/s e duração de 40 segundos; *trecho 3*) o MC permanece no ponto *P2* nos 40 segundos seguintes; *trecho 4*) o MC move-se de volta ao ponto *P1*, com velocidade de 1 m/s e duração de 40 segundos; *trecho 5*) o MC permanece no ponto *P1* durante os últimos 40 segundos.

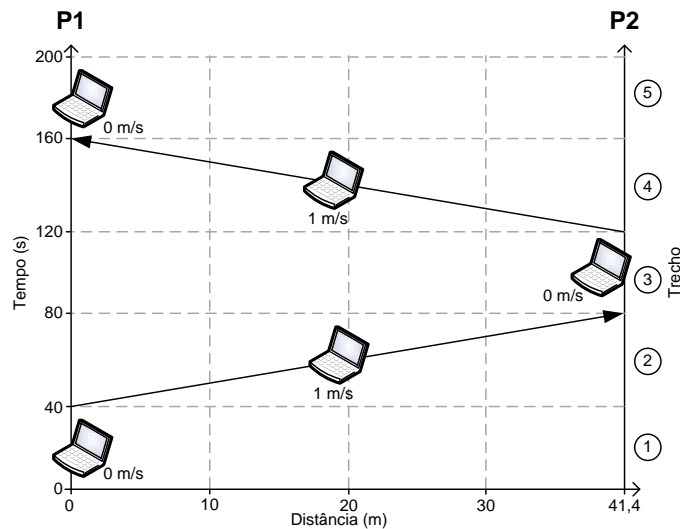


Figura 6.3: Diagrama de mobilidade no teste de *handoff*

O modelo de mobilidade acima se fundamenta em velocidade constante e períodos de pausa entre as mudanças de direção, padrão similar ao comportamento real do ser humano e frequentemente usado na literatura (MISHRA; DAS; JAISAWAL, 2013).

Dois diferentes modelos de tráfego foram utilizados para avaliar métricas de QoS e QoE durante a mobilidade do cliente. As próximas seções fornecem uma análise detalhada sobre essas métricas.

6.3 AVALIAÇÃO DA QUALIDADE DE SERVIÇO

Para medir a Qualidade de Serviço (QoS) da rede durante o *handoff*, fez-se uso de geração de tráfego baseado em taxa de bits constante (CBR, do inglês *Constant Bit Rate*), com intervalo de envio de pacotes (IDT, do inglês *Inter Departure Time*) e tamanho do pacote (PS, do inglês *Packet Size*) iguais a 8ms e 200bytes, respectivamente. O tráfego foi gerado através da ferramenta *iperf* (NLNR/DAST, 2013). A partir desses experimentos, foi possível aferir as seguintes métricas: taxa de comutação de pacotes, atraso, variação de atraso (*jitter*), potência de sinal Wi-Fi recebido e taxa de retransmissão de pacotes. Essas métricas foram escolhidas pois

elas têm relação direta com a QoS da rede e são comumente utilizadas na literatura (SCHULZ-ZANDER et al., 2014; AVELAR, 2013; DELY; KASSLER; BAYER, 2011; AMIR et al., 2010). Nas seções a seguir a importância de cada uma dessas métricas será apresentada, bem como seus respectivos resultados.

6.3.1 TAXA DE COMUTAÇÃO DE PACOTES

O gráfico da Figura 6.4 apresenta os resultados para taxa de comutação de pacotes entre o cliente móvel (MC) e nó correspondente (CN) em todas as plataformas de cliente e para ambas as estratégias OpenWiMesh-Mob e BATMAN-MCHO. Variações na linha do gráfico correspondem a perdas ou duplicação de pacotes e os valores próximos ao eixo x representam degradação de QoS.

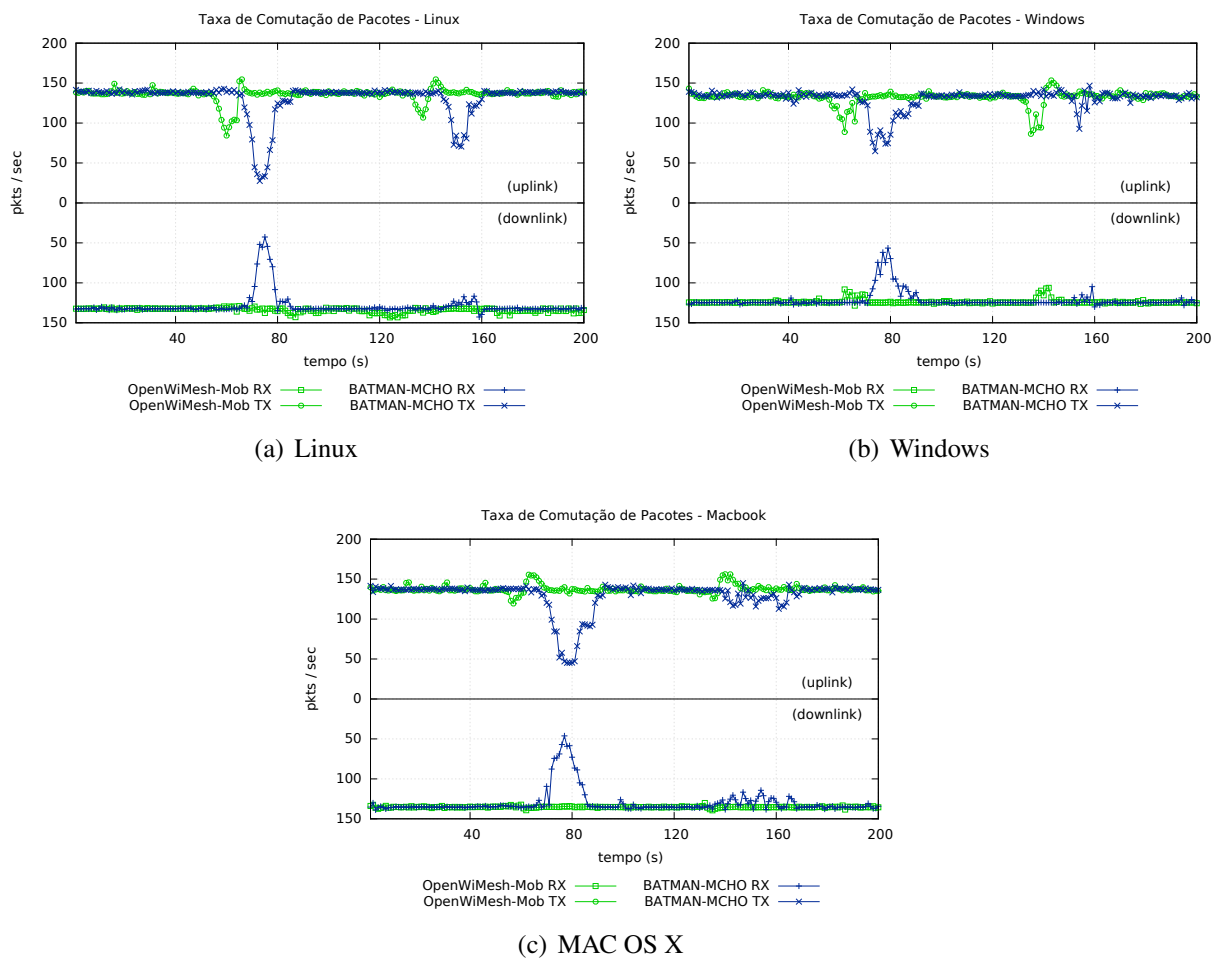


Figura 6.4: Taxa de Comutação de Pacotes com tráfego CBR entre MC e CN

Já o gráfico da Figura 6.5 retrata o nível de confiança de 95% para a mesma métrica. Ambas as plataformas apresentam intervalos de confiança semelhantes, por isso apresenta-se apenas os

dados do Mac OS X.

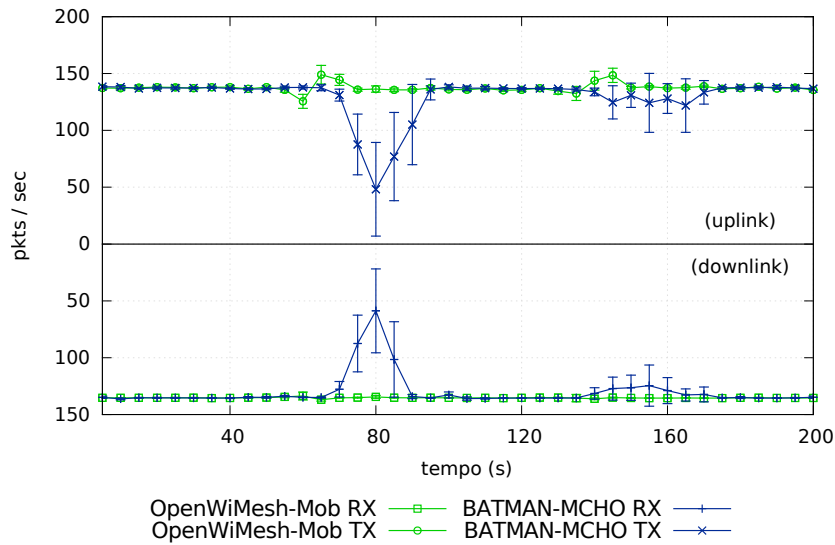


Figura 6.5: Taxa de Comutação de Pacotes com I.C. 95% (Mac OS X)

Pela análise dos gráficos, é possível verificar que a estratégia de *handoff* BATMAN-MCHO possui impacto significativo na qualidade de serviço da rede, tanto no tráfego de *uplink* quanto de *downlink*, levando a grande perda de pacotes nos momentos de *handoff*, independente da plataforma em questão. A taxa de entrega de pacotes chegou a ficar abaixo de 40% do esperado na plataforma MAC OS X. Essa perda se dá, principalmente, pelo fato de que a decisão de *handoff* é tomada pelo cliente, que não possui conhecimento das condições da rede e toma ações de maneira tardia, com pequena variação entre sistemas operacionais.

Por outro lado, ainda na Figura 6.4, o *handoff* controlado pelo OpenWiMesh-Mob fornece melhores condições de rede para o cliente, com pequenas oscilações no tráfego, porém sem causar interrupção de conexão. No MAC OS X, por exemplo, a taxa de comutação de pacotes no *downlink* do OpenWiMesh apresentou média de $135.30 pps$, com desvio padrão de 0.93, enquanto que o BATMAN-MCHO foi de $129.96 pps$, com desvio padrão de 15.54. Mais especificamente, no Trecho 2 (entre 40 e 80s), quando ocorre o primeiro *handoff*, o BATMAN-MCHO apresenta taxa de comutação de $118.43 pps$, com desvio de 28.82, uma degradação de 14.03% em relação ao OpenWiMesh-Mob, que teve média de $135.05 pps$ com desvio de 1.12. Essa degradação na taxa de computação de pacotes leva a uma experiência de uso da rede ruim, podendo causar inclusive a finalização de conexões TCP e grande perda de informações em tráfego UDP. Em se tratando de aplicações de tempo real, qualquer rajada de perda de pacotes resulta em degradação de qualidade da aplicação (FEAMSTER; BALAKRISHNAN, 2002).

Nota-se, entretanto, oscilação mais significativa do OpenWiMesh-Mob no tráfego de *uplink*

do cliente, com perda de pacotes e duplicação. A perda de pacotes apresentada foi devido a operação da versão original do *hostapd* usada na estratégia ora proposta. Trabalhos anteriores (SCHULZ-ZANDER et al., 2014) parecem superar esse problema através da modificação do código do software de AP, replicando previamente para os APs as informações de associação de cada cliente. Nesse trabalho, manteve-se o código original do *hostapd* e esse problema será melhor investigado em trabalhos futuros. Ademais, é possível notar que, logo após o *handoff*, o OpenWiMesh-Mob apresenta um tráfego de rede ligeiramente maior que a taxa de comutação constante esperada. O referido efeito é ocasionado devido a duplicação de pacotes na rede. Esse comportamento faz parte da estratégia de *handoff* do OpenWiMesh-Mob apresentada na Seção 5.4, que mantém o caminho antigo e atual ativos durante alguns instantes, a fim de aumentar a probabilidade de recepção de pacotes do nó. Apesar de causar duplicação de pacotes nesse período, o impacto para o cliente é mínimo, como será visto nas métricas de atraso e variação de atraso a seguir.

6.3.2 ATRASO BIDIRECIONAL

Algumas aplicações são bastante sensíveis ao atraso (*delay*) e variação do atraso (*jitter*), tolerando atraso máximo de 100ms e variação de 20ms a fim de garantir alta qualidade e interatividade (ITU-T, 2003; AMIR et al., 2010). Dessa maneira, essas métricas são sobremaneira importantes para qualidade de serviço da rede. Nesta seção, apresenta-se os resultados relacionados ao atraso bidirecional na rede, que foi medido através da métrica tempo de ida e volta (RTT). A próxima seção apresenta os resultados de variação de atraso. Na Figura 6.6, é possível verificar o atraso bidirecional, utilizando o mesmo modelo de tráfego anteriormente citado, porém, agora, com a ferramenta *ping*.

Já o gráfico da Figura 6.7 retrata o nível de confiança de 95% para a mesma métrica. Novamente apresenta-se apenas os dados de níveis de confiança apenas da plataforma Mac OS X sem prejuízo para as demais, que tiveram intervalos proporcionais.

Pela Figura 6.6 é possível observar que o OpenWiMesh-Mob esteve abaixo dos limites de atraso mencionados anteriormente nas plataformas Linux e MAC OS X, ao passo que na plataforma Windows estima-se que esses limites também foram respeitados, porém não é possível confirmar essa informação devido a medição bidirecional. Em particular, o OpenWiMesh não apresentou menor atraso bidirecional que o BATMAN-MCHO apenas nos testes realizados no Windows no Trecho 4 (segundo *handoff*, entre 120 e 160s). Isso ocorreu por dois motivos principais: a perda de pacotes no *uplink*, detalhado anteriormente (Seção 6.3.1), e pelo fato de que a ferramenta de teste de atraso utilizada no Windows apenas permitia o envio de um pacote

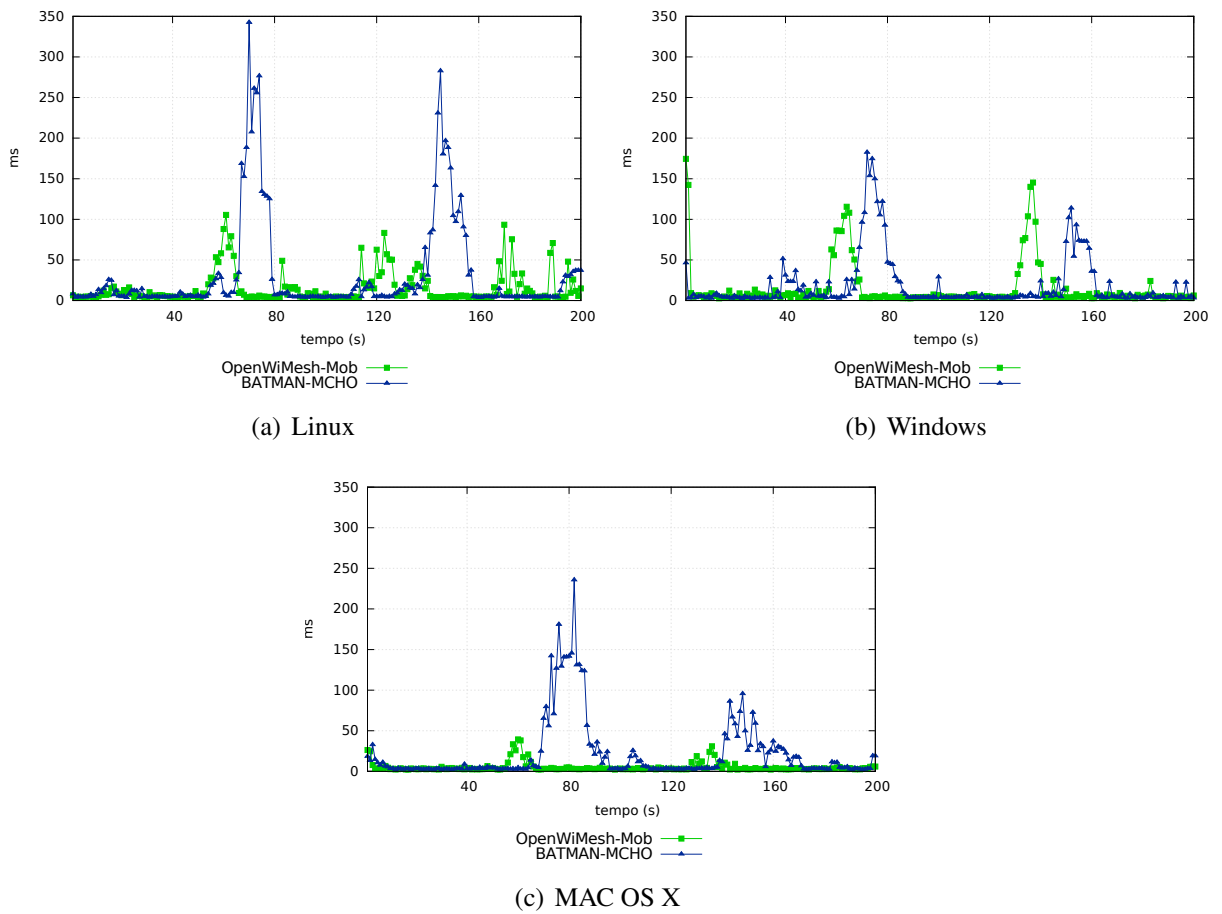


Figura 6.6: Atraso bidirecional (RTT) com tráfego CBR entre MC e CN

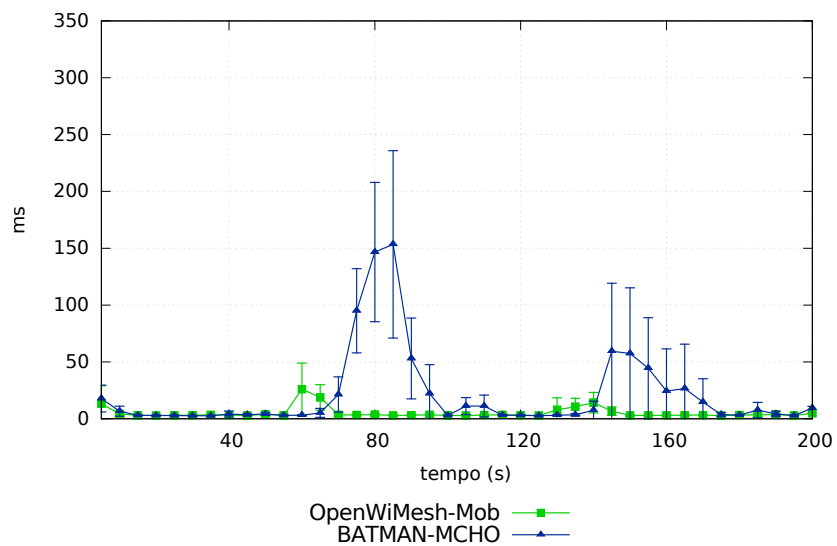


Figura 6.7: Atraso bidirecional com I.C. 95% (Mac OS X)

por segundo, maximizando o impacto de uma perda na análise dos dados. A principal causa do alto atraso observado nos testes com o BATMAN-MCHO foi, mais uma vez, o fato de o

handoff ser controlado pelo cliente. Por esse motivo, o *handoff* executava de forma tardia, em momento cuja baixa potência de sinal e elevada taxa de retransmissão já afetavam a conexão. O BATMAN-MCHO mantinha, portanto, uma associação Wi-Fi que já não oferecia condições de qualidade necessárias para as aplicações.

Os resultados mostram que o OpenWiMesh-Mob supera a estratégia BATMAN-MCHO em cerca de 300% durante as etapas de *handoff*, com medidas de atraso abaixo dos limites aceitáveis (ITU-T, 2003; AMIR et al., 2010). O resultado comprova que o OpenWiMesh-Mob é apropriado para aplicações interativas e aplicações de tempo real.

6.3.3 VARIAÇÃO DE ATRASO

Corroborando a análise anterior, o resultado da variação de atraso unidirecional, também conhecido como *jitter*, mostra que a estratégia ora proposta mantém a qualidade da rede em índices adequados para aplicações de requisito estrito, como as aplicações de tempo real. A avaliação do jitter pode ser vista na Figura 6.8.

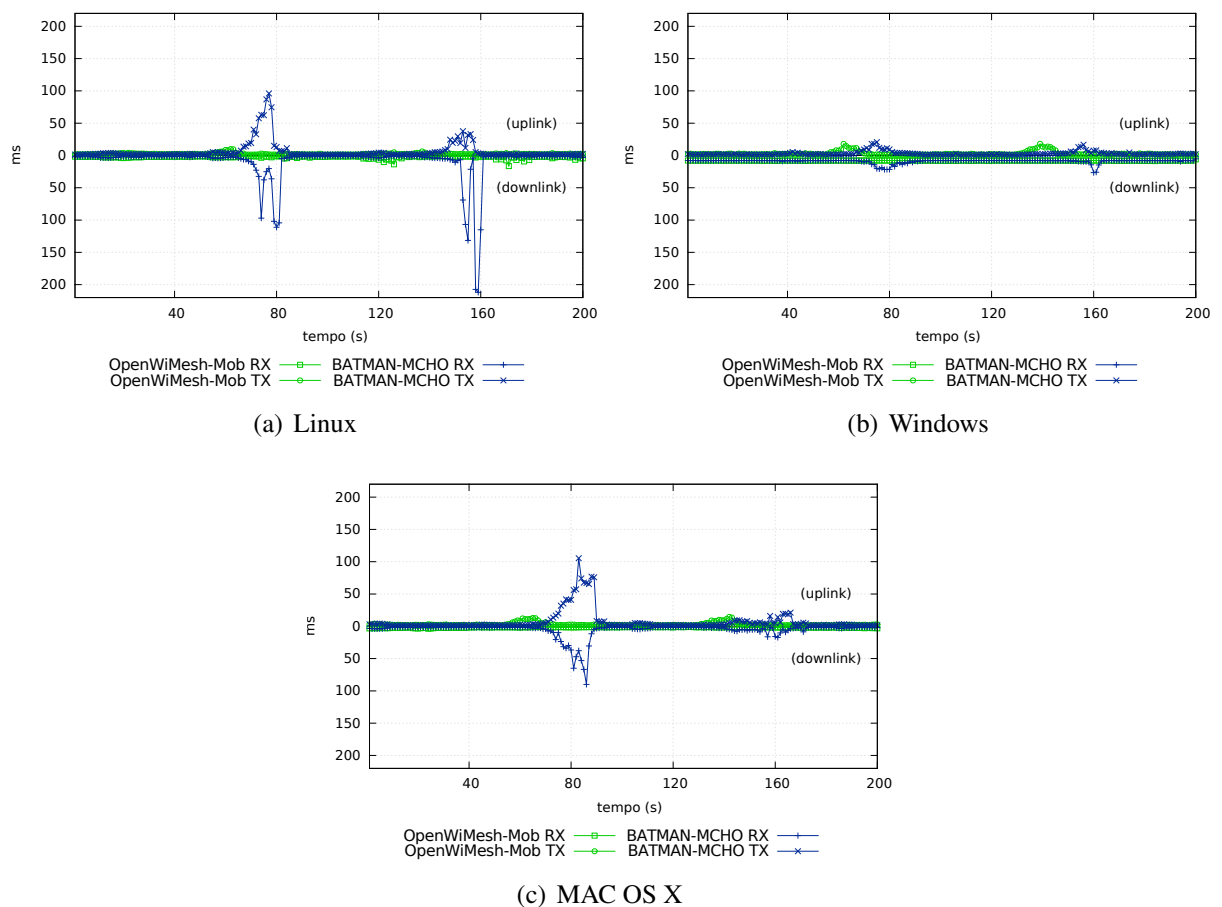


Figura 6.8: Variação de atraso com tráfego CBR entre MC e CN

No gráfico da Figura 6.9 é possível perceber o nível de confiança de 95% para a mesma métrica. Mais uma vez apresenta-se apenas os dados de I.C. apenas da plataforma Mac OS X sem prejuízo para as demais, que tiveram intervalos proporcionais.

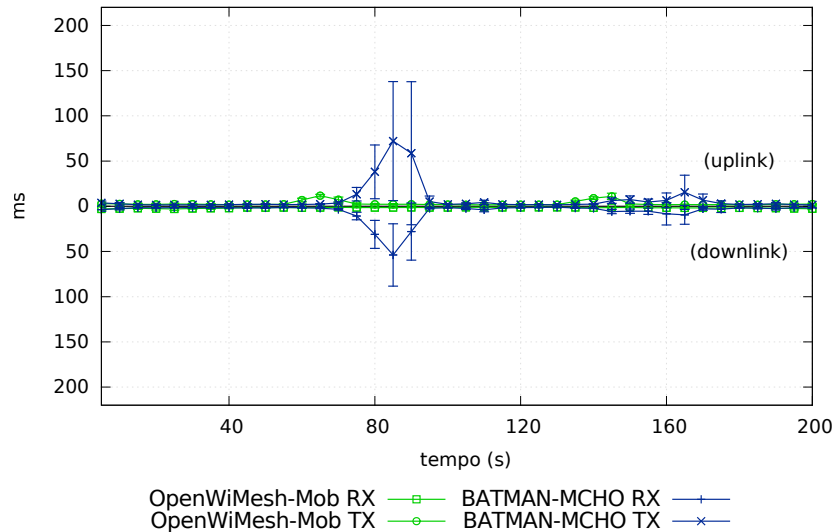


Figura 6.9: Variação de atraso com I.C. 95% (Mac OS X)

Analisando os resultados do pior caso, o OpenWiMesh-Mob, na plataforma Windows, apresentou uma variação de atraso de *uplink* e *downlink* na ordem de $3.02ms$ e $7.90ms$, com desvio padrão de 3.15 e 0.21 e valor máximo de $18.02ms$ e $8.71ms$, respectivamente. Por outro lado, o pior caso da abordagem usando BATMAN-MCHO acontece no Linux e apresenta valores ruins para aplicações de tempo real (ITU-T, 2003), com *jitter* de *uplink* e *downlink* na ordem de $6.48ms$ e $9.52ms$, com desvio de 13.75 e 28.90 e máximos de $96.07ms$ e $211.89ms$, respectivamente.

Nos resultados da plataforma MAC OS X, o OpenWiMesh-Mob teve um *jitter* de *downlink* e *uplink* na ordem de 1.76 ± 0.57 e $3.03 \pm 2.65ms$ (média e desvio padrão), com máximo de 3.77 e $14.57ms$, respectivamente. Já o BATMAN-MCHO apresentou *jitter* de *downlink* e *uplink* de 5.55 ± 11.44 e $7.47 \pm 15.65ms$, com máximo de 89.97 e $105.32ms$, respectivamente.

6.3.4 POTÊNCIA DE SINAL WI-FI

A métrica de potência de sinal Wi-Fi recebido (RSSI) possui relação indireta com a qualidade de serviço da rede, impactando na taxa de erros, atraso, jitter e probabilidade de recebimento de pacotes. Medido em decibéis (dBm ou decibel miliwatt), o RSSI pode variar entre $0e - 120dBm$ de acordo com o fabricante da interface Wi-Fi. Quanto mais próximo a zero for

o RSSI, mais forte é o sinal Wi-Fi e, teoricamente, melhor a qualidade do enlace, com exceção das interferências e ruídos.

Os resultados de medições de RSSI são ilustrados na Figura 6.10. Esse gráfico apresenta apenas o resultado obtido da plataforma MAC OS X, pois na plataforma Windows não foi possível coletar essa informação² e o resultado do Linux foi bem semelhante ao MAC OS X. As curvas do gráfico mostram que o OpenWiMesh-Mob fez boa alocação de cliente para AP, fornecendo RSSI no AP e no cliente sempre superiores a -72.1 e -67.8dBm , sobretudo, se comparado com a abordagem BATMAN-MCHO, cujos valores de RSSI no AP e no cliente chegaram a -84.4 e -78.3dBm , faixas de valores de menor qualidade do enlace.

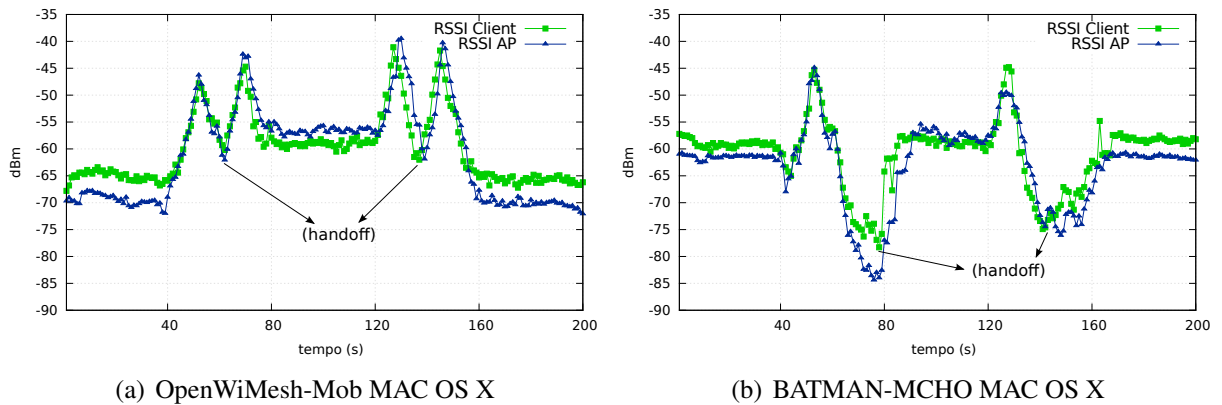


Figura 6.10: Potência de Sinal Recebido (RSSI) com tráfego CBR entre MC e CN

Na Figura 6.11 é possível perceber o nível de confiança de 95% para a mesma métrica.

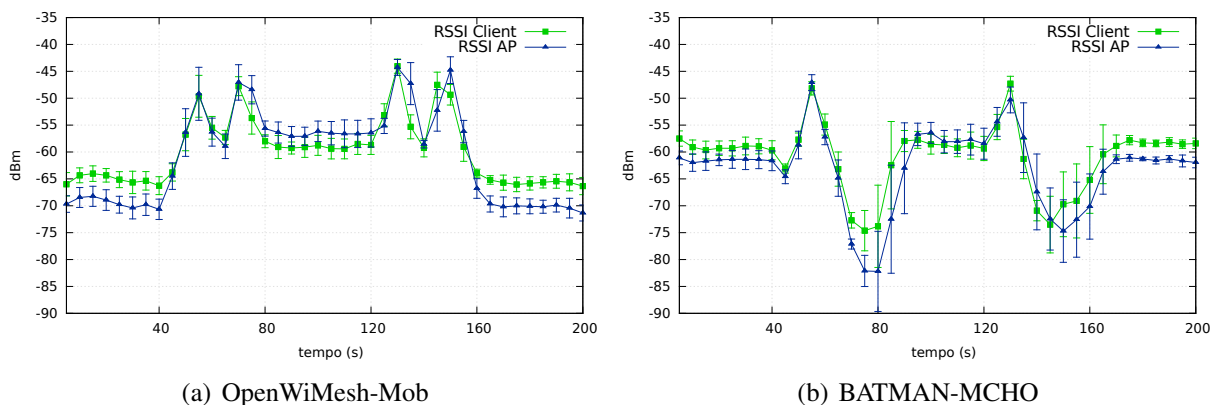


Figura 6.11: Potência de Sinal Recebido com I.C. 95% (Mac OS X)

Pela Figura 6.10a pode-se notar que no Trecho 2 (entre 40 e 80s) o RSSI melhora linearmente à medida que o cliente se aproxima do MR1, depois degrada notadamente quando se

²Foi utilizado o comando *netsh* para coleta de informações, porém a informação de RSSI está disponível apenas em porcentagem e apresenta variações de valores que não condizem com as características da rede.

distancia de MR1 e acontece o *handoff* para MR2, então, ele experimenta nova melhoria ao aproximar-se de MR2, e volta a cair até estabilizar no afastamento de MR2 e parada no ponto 2 (no instante 80s). Em contrapartida, a curva de RSSI do BATMAN-MCHO (Figura 6.10b) mostra que o sinal Wi-Fi se degrada a níveis muito baixos até que o cliente faça o *handoff*, que praticamente só ocorre quando ele já está no ponto 2. Comportamento similar é notado no Trecho 4 (entre 120 e 180s) em ambas as estratégias. Curiosamente é possível observar que o RSSI no OpenWiMesh-Mob foi ligeiramente pior no Trecho 1 e 5 (entre 0 e 40s e 160 e 200s, respectivamente), possivelmente fruto de interferências momentâneas do meio sem fio; porém, sem prejuízos destacáveis à qualidade da rede ou à análise dos resultados.

6.3.5 TAXA DE RETRANSMISSÃO DE PACOTES

Outra métrica que indiretamente impacta na qualidade da rede é a taxa de retransmissão de pacotes (FRR, *Frame Retransmission Rate*). A subcamada MAC, da camada de enlace do protocolo 802.11, requer que todos os pacotes transmitidos sejam confirmados através de quadros de controle ACK, como parte da função de acesso ao meio. Quando o ACK não é recebido em determinado período de tempo, fruto de um possível erro na rede ou no receptor, o emissor entra em estado de contenção e aguarda um tempo aleatório para reenviar o pacote não confirmado. Em caso de novos erros, a retransmissão é repetida, geralmente em taxa de transmissão menor, até um limite máximo ser alcançado. Em redes Wi-Fi com muitos usuários ou tráfego simultâneo, a retransmissão ocorre como resultado de colisões de transmissão. A retransmissão pode ocorrer também por interferências externas. Não obstante, quando em índices elevados, o processo de retransmissão impacta significativamente o desempenho da rede. Por esse motivo, como parte da avaliação da qualidade de serviço da rede, foram realizadas medições referentes à taxa de retransmissão de pacotes, avaliando o comportamento dos nós durante o *handoff*.

Na Figura 6.12 é possível verificar o FRR, na perspectiva do cliente e do AP. Apenas o gráfico da plataforma MAC OS X foi exibido novamente, pelos mesmos motivos supracitados. É possível notar que o OpenWiMesh-Mob mantém a taxa de retransmissões dentro de um intervalo bem definido, mesmo durante a execução dos *handoff*'s nos Trechos 2 e 4 (entre 40 e 80s e 120 e 160s, respectivamente). Enquanto isso, o mal gerenciamento do *handoff* é evidenciado na outra estratégia que apresenta taxa de retransmissão aproximadamente 50% maior nos trechos em que ocorre *handoff*.

A Tabela 6.1 mostra uma análise estatística mais detalhada dos resultados de retransmissão de pacotes.

O notório incremento na taxa de retransmissão durante as etapas de *handoff* do BATMAN-

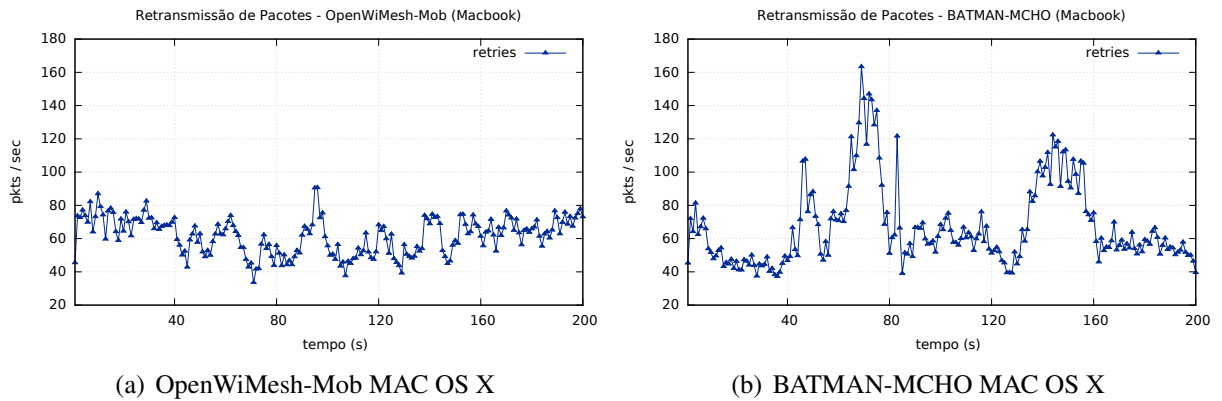


Figura 6.12: Taxa de retransmissão de pacotes com tráfego CBR entre MC e CN

Trecho	Proposta	Média	I.C. 95%	Mín.	Mediana	Máx.
Trecho 1	OWM	70.78	± 14.60	45.40	71.70	86.90
	BAT	49.49	± 20.76	37.20	46.35	81.30
Trecho 2	OWM	55.78	± 18.22	33.50	56.20	73.80
	BAT	88.56	± 63.53	47.00	76.15	163.30
Trecho 3	OWM	55.28	± 24.02	37.60	51.25	90.60
	BAT	62.04	± 24.34	39.00	60.45	121.60
Trecho 4	OWM	59.53	± 20.93	39.20	59.25	74.60
	BAT	80.76	± 51.78	39.20	86.40	122.30
Trecho 5	OWM	66.90	± 12.57	52.40	66.10	77.80
	BAT	55.23	± 11.30	39.50	54.70	69.90
Total	OWM	61.65	± 21.79	33.50	62.80	90.60
	BAT	67.21	± 48.68	37.20	58.80	163.30

Tabela 6.1: Retransmissão de pacotes (pkts/sec) do AP para MC no MAC OS X

MCHO é resultado da distância entre o AP e o cliente móvel, culminando, dessa forma, em uma associação Wi-Fi ruim. É fácil correlacionar esse resultado com os dados de potência de sinal, apresentado na Figura 6.10, e também com o mapa de intensidade de sinal, previamente discutido na Figura 6.2, sendo consequência da delongada tomada de decisão de *handoff* pelo cliente. Observa-se, no entanto, que os resultados do Trecho 1 (entre 0 e 40s) na Tabela 6.1 apontam para uma execução com maior taxa de retransmissão do OpenWiMesh-Mob face ao BATMAN-MCHO, coincidindo com os valores de potência de sinal apresentados na Figura 6.2, onde no primeiro trecho o sinal Wi-Fi observado para o OpenWiMesh-Mob foi ligeiramente pior. Tal fato foi ocasionado possivelmente por fonte de interferência externa especificamente próxima ao ponto inicial. Não houve, todavia, prejuízos à análise comparativa das soluções, tanto para essa métrica, quanto para todas as demais já apresentadas.

Evidencia-se, portanto, que a estratégia de *handoff* controlado pela rede do OpenWiMesh-

Mob executa de forma eficiente e no tempo correto, não sendo impactada por alto FRR.

6.4 AVALIAÇÃO DA QUALIDADE DE EXPERIÊNCIA

Em complemento às métricas de QoS, foi conduzida também uma avaliação de qualidade de experiência dos usuários para ambas as estratégias. O conceito de Qualidade de Experiência (QoE, do inglês *Quality of Experience*) está relacionado à análise das aplicações do ponto de vista da percepção humana (GOMES et al., 2010). O termo surgiu para suprir “lacunas” deixadas pela medição tradicional realizada pelas métricas de QoS, que examinam o impacto das aplicações do ponto de vista da rede, não refletindo a experiência do usuário com relação à aplicação. Especialmente em aplicações multimídia, as métricas de QoS não são suficientes para avaliar se a qualidade de um vídeo recebido pelo usuário, por exemplo, pode ser qualificado como bom, ou qual o seu grau de qualidade.

Por definição, as medições de QoE são baseadas em percepção subjetiva do usuário final, abrangendo qualquer interferência no sistema fim-a-fim (cliente, servidor, rede, infraestrutura de serviços etc.). No entanto, para medir adequadamente a QoE deve-se realizar testes com os usuários propriamente ditos, tornando-se inviável em termos de tempo e custo do processo. Assim, são utilizadas ferramentas e técnicas que refletem de forma objetiva, e com razoável grau de precisão, a opinião subjetiva dos usuários. Tal opinião subjetiva, mais conhecida como *Escore Médio de Opinião* (MOS, do inglês *Mean Opinion Score*), fornece uma escala de qualificação que varia de Excelente (5) à Ruim (1), cabendo à métrica objetiva fornecer uma correlação com o MOS para medição da QoE.

Uma das técnicas utilizadas para aferir de forma objetiva a QoE baseia-se em medições da qualidade de vídeo, comparando a versão original de um vídeo com uma amostra coletada após sua transmissão pela rede. Checa-se, então, o grau de semelhança entre elas. Este trabalho utiliza uma técnica semelhante de medição da QoE, tomando como base a qualidade de vídeo. Na próxima subseção, serão apresentadas duas métricas de comparação de vídeo, em conjunto com suas equivalências ao MOS.

6.4.1 MEDIÇÕES DE QUALIDADE DE VÍDEO

As medições de qualidade do vídeo recebido são realizadas através de modelos matemáticos que atribuem um escore de semelhança e qualidade para a amostra em avaliação. De acordo com Gomes et al. (2010), as principais métricas objetivas para medição de qualidade de vídeo são:

Tabela 6.2: Mapeamento PSNR/MOS e SSIM/MOS

MOS	PSNR (dB)	SSIM
(5) Excelente	> 37	> 0.90
(4) Bom	31 – 37	0.77 – 0.89
(3) Aceitável	25 – 31	0.61 – 0.76
(2) Pobre	20 – 25	0.38 – 0.60
(1) Ruim	< 20	< 0.38

PSNR (do inglês *Peak Signal to Noise Ratio*) e SSIM (do inglês *Structural Similarity*). Nesta dissertação, tais métricas são aplicadas na medição da qualidade dos vídeos (AVELAR, 2013; LAMBRECHT; VERSCHEURE, 1996; ENGELKE et al., 2009).

O PSNR é a métrica objetiva mais tradicional para avaliação de vídeos, comparando quadro a quadro a qualidade do vídeo recebido pelo usuário com a versão original. O SSIM, por sua vez, é uma medição da distorção estrutural do vídeo, tentando obter melhor correlação com a impressão subjetiva dos usuários. O SSIM compara a similaridade entre os vídeos nos seguintes aspectos: contraste, luminosidade e estrutura.

O PSNR é expresso em decibéis (dB). Para que um vídeo seja considerado de boa qualidade, ele deve ter um média de PSNR acima de 30 dB. Já o SSIM é expresso como um valor decimal entre 0 e 1. Quanto mais próximo de 1, melhor é a qualidade do vídeo. Os valores computados de PSNR e SSIM podem ser mapeados no MOS, como mostrado na Tabela 6.2 (LAMBRECHT; VERSCHEURE, 1996; ENGELKE et al., 2009), facilitando a avaliação da qualidade de experiência dos usuários.

O cenário utilizado para medição do QoE é semelhante ao utilizado nos testes anteriores, porém, com adição de uma aplicação de transmissão (*streaming*) de vídeo, onde o cliente móvel assiste a um vídeo enquanto se move pela rede. Para fazer o *streaming* de vídeo utilizou-se a ferramenta *live555*³, transmitindo a animação Big Buck Bunny na resolução de 240p (320x240), formato MPEG4-TS, com duração de 200 segundos e cadência de 15 fps. Esse experimento também foi repetido dez vezes. O vídeo transmitido pela rede foi comparado com o original através das métricas PSNR e SSIM, conforme supracitado.

Durante a transmissão do vídeo, assim como nos testes anteriores, o usuário permanecia em constante movimento de um ponto de acesso para outro na área do *testbed* (cf. §6.2). Após o recebimento do vídeo, foi utilizada a versão gratuita da ferramenta MSU VQMT (*Video Quality Measurement Tool*) (MSU Video Group, 2013), para computar as métricas PSNR e SSIM e extrair as informações necessárias para avaliação do vídeo recebido.

³Live555 é um conjunto de ferramentas e bibliotecas para *streaming* de vídeo via Internet, Wireless ou Multicast. Mais informações: Site<<http://www.live555.com/>>, Último acesso 20/Maio/2016

6.4.2 AVALIAÇÃO DO PSNR

Os resultados da avaliação de qualidade do vídeo através do PSNR podem ser vistos na Figura 6.13. Nesse gráfico, é possível visualizar claramente a queda de qualidade do vídeo após o primeiro *handoff*, porém o OpenWiMesh-Mob consegue manter uma média de qualidade de boa a excelente nos trechos seguintes. Já o BATMAN-MCHO apresenta queda de qualidade significativa apenas no final do Trecho 2 (por volta de 70 segundos), quando o nível de sinal Wi-Fi e qualidade da rede já estão bastante degradados, resultando, pois, em uma qualidade de experiência de pobre a ruim. Mesmo no trecho 3, onde o MC está parado, a qualidade do vídeo permanece em níveis menores, principalmente, devido aos fatores de propagação de erros em frames interdependentes de vídeos MPEG, problema já conhecido na literatura (FEAMSTER; BALAKRISHNAN, 2002).

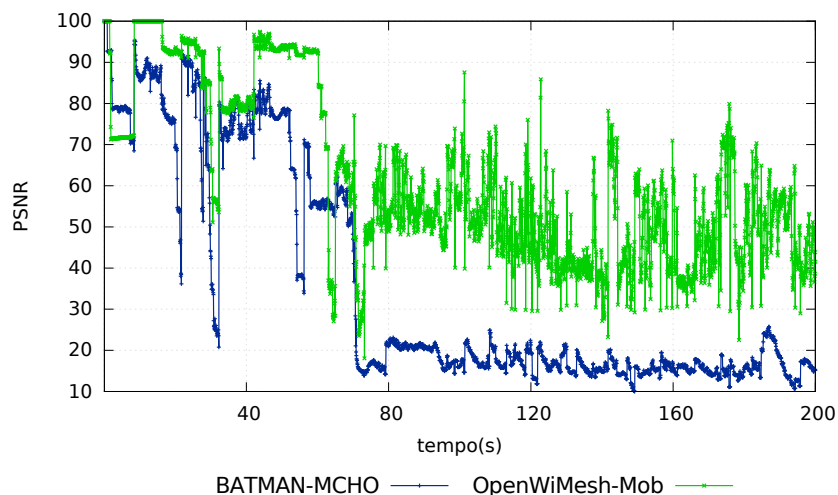


Figura 6.13: Avaliação PSNR de vídeo transmitido entre CN e MC

A Figura 6.14 apresenta a mesma medição anterior, porém com valores agregados em intervalos de quarenta segundos, a fim de facilitar a visualização das curvas. Foi acrescentado nessa figura, ainda, a faixa de nível de confiança 95%. A variação dos dados observada na Figura 6.14 é ocasionada pelas características da própria métrica, ao passo que o mapeamento de sua classe de valores para a métrica MOS permanece estável, conforme mostrado a seguir.

A Tabela 6.3 detalha os cálculos da métrica PSNR para a comparação entre as estratégias OpenWiMesh-Mob e BATMAN-MCHO. Ao considerar os valores mínimos de PSNR medidos para o OpenWiMesh-Mob, é possível notar que durante todos as etapas do teste, a QoE do usuário esteve sempre acima do índice MOS “Bom”.

O mapeamento entre o PSNR e o MOS foi retratado através da Figura 6.15, que considera

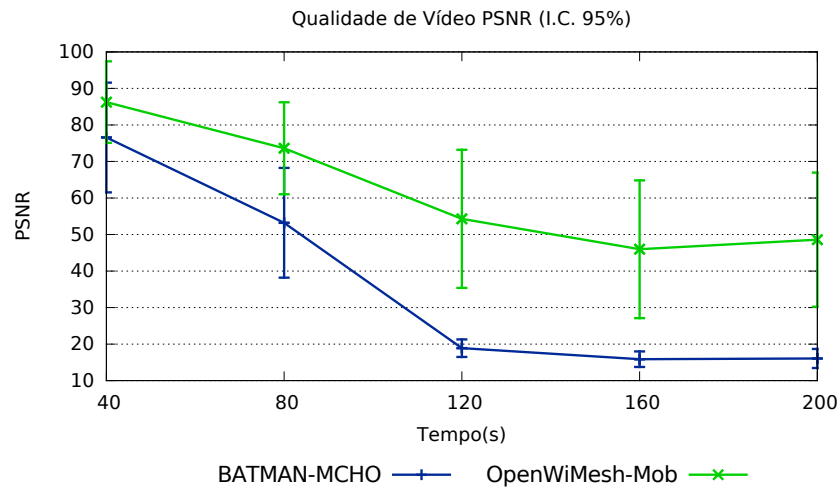


Figura 6.14: Avaliação PSNR com valores agregados

Trecho	Proposta	Média	I.C. 95%	Mín.	Mediana	Máx.
Trecho 1	OWM	86.25	± 24.21	51.08	92.09	100.00
	BAT	76.57	± 30.73	20.78	78.86	100.00
Trecho 2	OWM	73.61	± 42.95	18.05	80.11	97.45
	BAT	53.20	± 45.85	13.91	56.14	85.57
Trecho 3	OWM	54.29	± 15.42	29.82	53.23	87.55
	BAT	18.88	± 4.63	13.91	19.38	24.94
Trecho 4	OWM	45.96	± 20.03	23.22	42.27	85.89
	BAT	15.86	± 4.15	10.04	15.55	21.98
Trecho 5	OWM	48.59	± 21.09	22.50	47.09	79.90
	BAT	16.06	± 5.91	10.74	15.52	25.75
Total	OWM	61.74	± 40.47	18.05	56.32	100.00
	BAT	36.11	± 54.39	10.04	19.30	100.00

Tabela 6.3: Qualidade de Vídeo PSNR em Transmissão do CN para MC

o valor médio para cada período de quarenta segundos, facilitando a percepção do impacto da estratégia de *handoff* na qualidade de experiência do usuário ao longo dos trechos do teste.

A partir dos resultados acima, é possível notar que a estratégia de *handoff* baseada no OpenWiMesh-Mob apresenta desempenho médio 70% superior à estratégia BATMAN-MCHO na métrica PSNR. Esse resultado demonstra, independente da comparação com outra estratégia, que o OpenWiMesh-Mob consegue viabilizar um *handoff* transparente aos clientes móveis, sem degradar o QoE percebido pelos usuários.

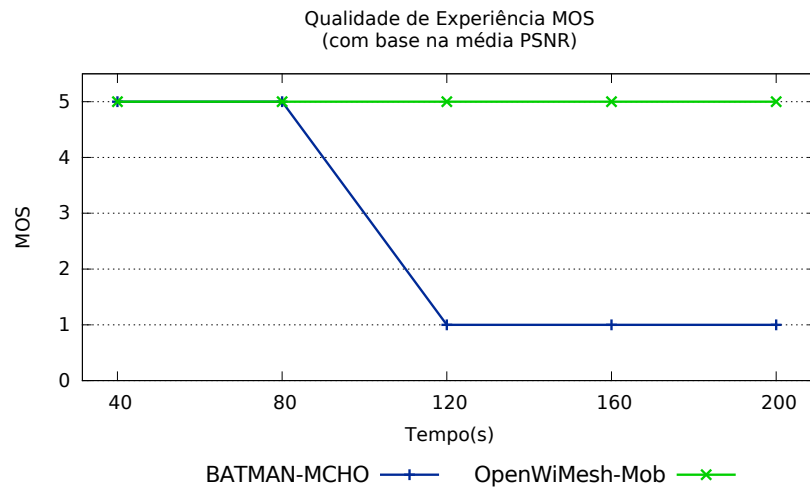


Figura 6.15: Mapeamento das medições PSNR para MOS

6.4.3 AVALIAÇÃO DO SSIM

Os resultados da avaliação de qualidade do vídeo através da métrica SSIM podem ser vistos nas Figuras 6.16 e 6.17. A Figura 6.16 contém as medições completas, quadro a quadro, enquanto que a Figura 6.17 apresenta as medições agregadas a cada quarenta segundos (de acordo com as etapas do teste). A métrica de SSIM revela que o OpenWiMesh-Mob consegue manter o vídeo em uma média de qualidade alta, com pequenas oscilações e que a estratégia do BATMAN-MCHO causa queda na qualidade do vídeo variando de aceitável a pobre a partir do final do trecho 2.

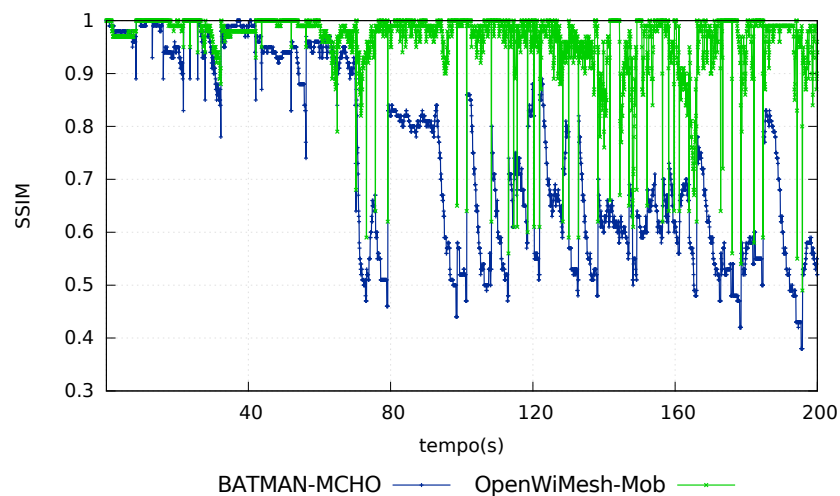


Figura 6.16: Avaliação SSIM de vídeo transmitido entre CN e MC

É possível visualizar as medições da métrica SSIM, com riqueza de detalhes, através da

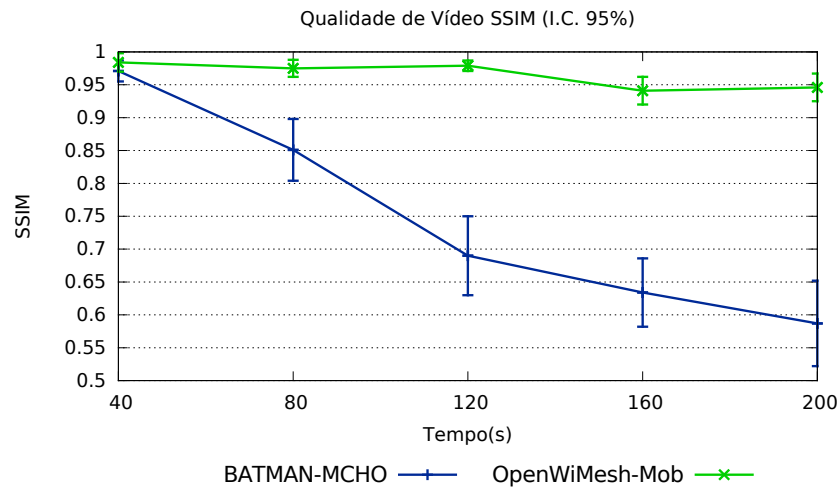


Figura 6.17: Avaliação SSIM com valores agregados

Tabela 6.4. Considerando os valores mínimos na medição do índice SSIM, é possível notar que o OpenWiMesh-Mob é qualificado como “Excelente” pelo mapeamento SSIM/MOS em todos os trechos. Os valores referentes ao BATMAN-MCHO, embora melhores que na métrica PSNR, ainda demonstram uma baixa qualidade dessa proposta: no trecho 3, a média e mediana apontam para uma qualidade “Aceitável”, embora o valor mínimo esteja na fronteira entre “Aceitável” e “Pobre”; nos trechos 4 e 5, se considerarmos o nível de confiança, tanto a média quanto a mediana e o mínimo apontam para uma qualidade “Pobre” do vídeo transmitido.

Trecho	Proposta	Média	I.C. 95%	Mín.	Mediana	Máx.
Trecho 1	OWM	0.98	± 0.04	0.88	0.99	1.00
	BAT	0.97	± 0.07	0.78	0.98	1.00
Trecho 2	OWM	0.98	± 0.08	0.59	0.99	1.00
	BAT	0.85	± 0.32	0.46	0.93	1.00
Trecho 3	OWM	0.98	± 0.09	0.56	0.99	1.00
	BAT	0.69	± 0.25	0.44	0.71	0.88
Trecho 4	OWM	0.94	± 0.15	0.59	0.96	1.00
	BAT	0.63	± 0.16	0.48	0.63	0.89
Trecho 5	OWM	0.95	± 0.15	0.49	0.98	1.00
	BAT	0.59	± 0.20	0.38	0.56	0.83
Total	OWM	0.97	± 0.11	0.49	0.98	1.00
	BAT	0.75	± 0.35	0.38	0.73	1.00

Tabela 6.4: Qualidade de Vídeo SSIM em Transmissão do CN para MC

A correlação entre a métrica de qualidade de vídeo SSIM e a qualidade de experiência MOS pode ser vista na Figura 6.18. Nessa figura, foi considerado apenas a média de cada trecho do teste. É possível notar que o OpenWiMesh-Mob mantém o índice MOS em “Excelente” durante

todos os trechos do teste, enquanto a estratégia BATMAN-MCHO apresenta degradação ao longo do teste, chegando a apresentar índice MOS “Pobre” no último trecho.

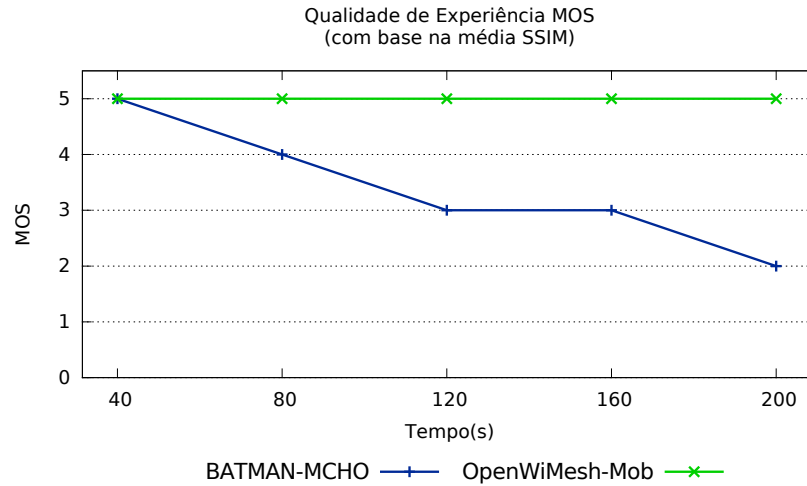


Figura 6.18: Mapeamento das medições SSIM para MOS

Em resumo, a métrica SSIM revela uma melhora na qualidade de experiência da ordem de 29% proporcionada pela estratégia OpenWiMesh-Mob. Adicionalmente, as medições de SSIM evidenciam que a qualidade das aplicações permanece em alto nível mesmo face à execução do *handoff* sucessivas vezes.

6.4.4 AVALIAÇÃO VISUAL DE QUALIDADE

Complementando as avaliações de qualidade do vídeo baseadas nas métricas PSNR e SSIM, na Figura 6.19 é apresentada uma captura de quadros (*frames*) do vídeo sendo transmitido exatamente durante o segundo *handoff* (Etapa 4). A Fig. 6.19a é o vídeo original, a Fig. 6.19b refere-se ao OpenWiMesh-Mob e a Fig. 6.19c ao BATMAN-MCHO. É possível perceber de forma clara o impacto na qualidade do vídeo ocasionado pela má gerência de mobilidade. Por outro lado, a diferença entre o quadro do vídeo original e do vídeo com transmissão via OpenWiMesh é praticamente imperceptível ao olhar humano.

Foi realizada também uma avaliação com relação ao percentual de perda de dados na transmissão do vídeo, comparando o tamanho em *bytes* do arquivo original com o arquivo obtido pela rede. A Figura 6.20 retrata o percentual de perda de ambas as propostas e a Tabela 6.5 traz mais detalhes sobre essa avaliação. É possível observar que, no final, a estratégia com BATMAN-MCHO apresenta uma perda média de 2.31%, com máxima de 3.17%, enquanto que a perda média apresentada pelo OpenWiMesh-Mob foi apenas 0.06%, com máximo de 0.08%.



Figura 6.19: Captura do quadro 2220 durante o *handoff* em cada estratégia

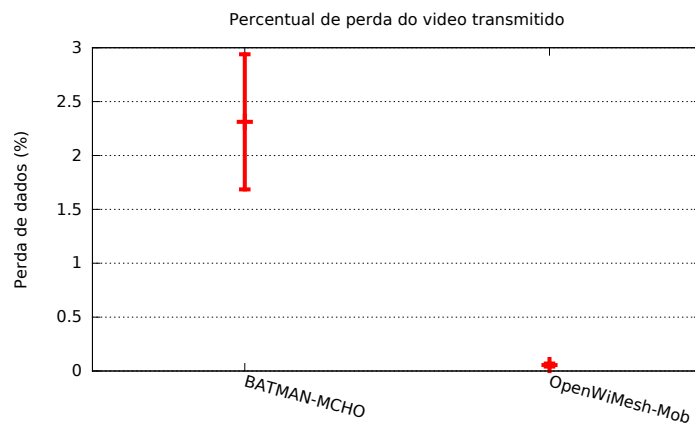


Figura 6.20: Percentual de perda de dados na transmissão de vídeo

6.5 PARAMETRIZAÇÕES E CONFIGURAÇÕES DA ESTRATÉGIA

Esta seção apresenta um estudo comparativo acerca de diferentes parametrizações e configurações da estratégia de *handoff* OpenWiMesh-Mob. O objetivo dessa comparação é fornecer detalhes e identificar tendências em relação a configuração do algoritmo de *handoff* e a metodologia de teste.

Proposta	Média	I.C. 95%	Mín.	Mediana	Máx.
OWM	0.055	± 0.014	0.024	0.051	0.089
BAT	2.312	± 0.627	0.191	2.678	3.173

Tabela 6.5: Percentual de perda de dados na transmissão de vídeo

6.5.1 LIMIAR DE *HANDOFF* T

O limiar de *handoff* T é usado para que um enlace sem fio seja considerado de baixa qualidade, de acordo com a métrica $Link_Q$ (Seção 5.2), e o outros APs sejam verificados para execução do *handoff*. Esse limiar é um valor decimal, entre 0.0 e 1.0, e representa a porcentagem de qualidade do enlace abaixo da qual ele deve ser realocado. A fim de avaliar o impacto da escolha do valor de T , foram executados três experimentos com T assumindo os valores 0.60, 0.80 e 1.0, ambos realizados no mesmo cenário e com a mesma metodologia apresentada nas Seções 6.1 e 6.2, porém com duração de cento e vinte segundos e com cinco repetições.

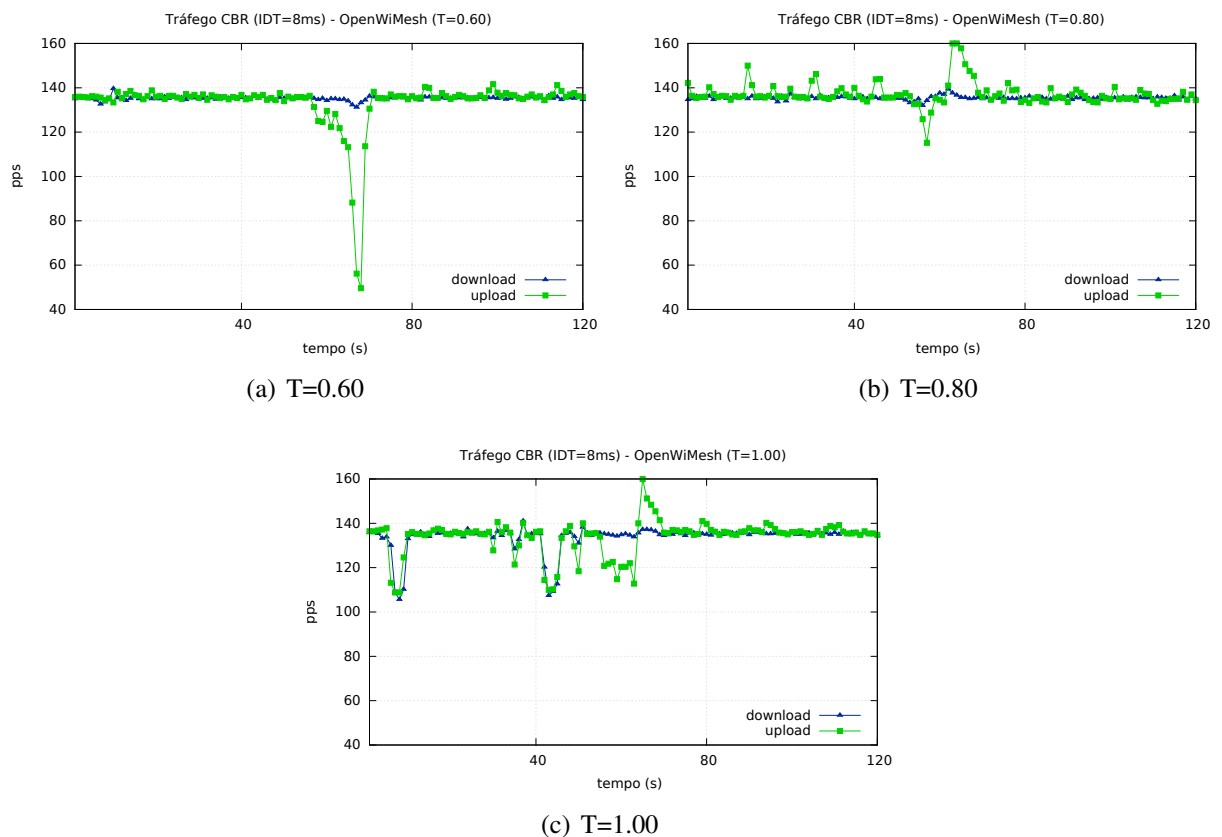


Figura 6.21: Taxa de comutação de pacotes com diferentes valores de T

O gráfico da Figura 6.21 apresenta os resultados para taxa de comutação de pacotes entre MC e CN com a estratégia OpenWiMesh-Mob e considerando diferentes valores de T . De acordo com a metodologia do experimento, o momento teoricamente ideal para o *handoff* é próximo aos 60 segundos, quando o MC está entre os dois APs e movendo em direção ao MR2. Na Figura 6.21a é possível observar a taxa de comutação com $T = 0.60$, onde o *handoff* ocorre em momento ligeiramente tardio, por volta de 65 a 70 segundos e, portanto, a taxa de comutação degrada acima de 50%. Já na Figura 6.21c, cujo valor de T foi configurado para 1.00, percebe-se diversas oscilações ao longo do teste, em geral ocasionadas por decisões de

handoff precipitadas. Nesse caso, o que ocorre é que, como o limiar assume valor máximo, qualquer medição de $Link_Q$ que apresente valor abaixo de 1.0 habilita o algoritmo a buscar APs candidatos que possuem melhores métricas de qualidade, levando a *handoffs* para APs alvos que não valem a pena naquele instante. Por fim, a Figura 6.21b apresenta uma boa relação de compromisso entre o limiar de *handoff* e a qualidade da rede, medida em função da taxa de comutação de pacotes. Nesta configuração, o algoritmo de *handoff* passa a procurar APs candidatos apenas quando o $Link_Q$ do AP atual fica abaixo de 0.8, quando as condições do enlace passam a impactar na qualidade da rede.

Valores de T intermediários aos apresentados foram considerados, porém os resultados não foram representativos no cenário utilizado. Dessa forma recomenda-se avaliá-los em cenários maiores e com qualidade maior de MCs e APs candidatos.

6.5.2 HISTERESE DE *HANDOFF* H

Conforme já foi discutido na Seção 5.3, o valor da histerese define uma margem de segurança que o AP candidato deve apresentar em relação ao AP atual para ser considerado elegível no processo de *handoff*. Para avaliar a relação do valor da histerese na estratégia de *handoff* proposta, foram considerados três valores de H: $H = 0.10$, $H = 0.05$ e $H = 0.00$. Os testes foram realizados no mesmo cenário e com a mesma metodologia apresentada nas Seções 6.1 e 6.2, porém com duração de cento e vinte segundos e com cinco repetições.

O gráfico da Figura 6.22 apresenta os resultados obtidos pela variação de H. Pela análise dos gráficos não é fácil notar distinção entre os resultados com cada valor de H. Essa característica é prevista pois como o MC está movendo em velocidade constante de MR1 para MR2, a degradação associada de sinal para o AP original com o deslocamento do nó contribui para inexistência do efeito *ping-ponging*. Foi necessário, dessa forma, avaliar esse parâmetro de outra maneira.

A maneira utilizada para avaliar o parâmetro H de forma mais efetiva foi manter o MC estacionário entre os dois APs e medir a quantidade de *handoffs* que a estratégia executa. Idealmente o cliente deve conectar a um AP e não executar nenhum ou o mínimo de *handoffs*, uma vez que ambos oferecem as mesmas condições, com variações decorrentes apenas do meio sem fio.

O gráfico da Figura 6.23 apresenta a quantidade de *handoffs* executados pelo OpenWiMesh-Mob para cada valor de H. É possível observar que para $H = 0.00$ ocorrem muitos *handoffs* na rede, em média 5 no experimento executado, caracterizando o efeito *ping-ponging*. Essa

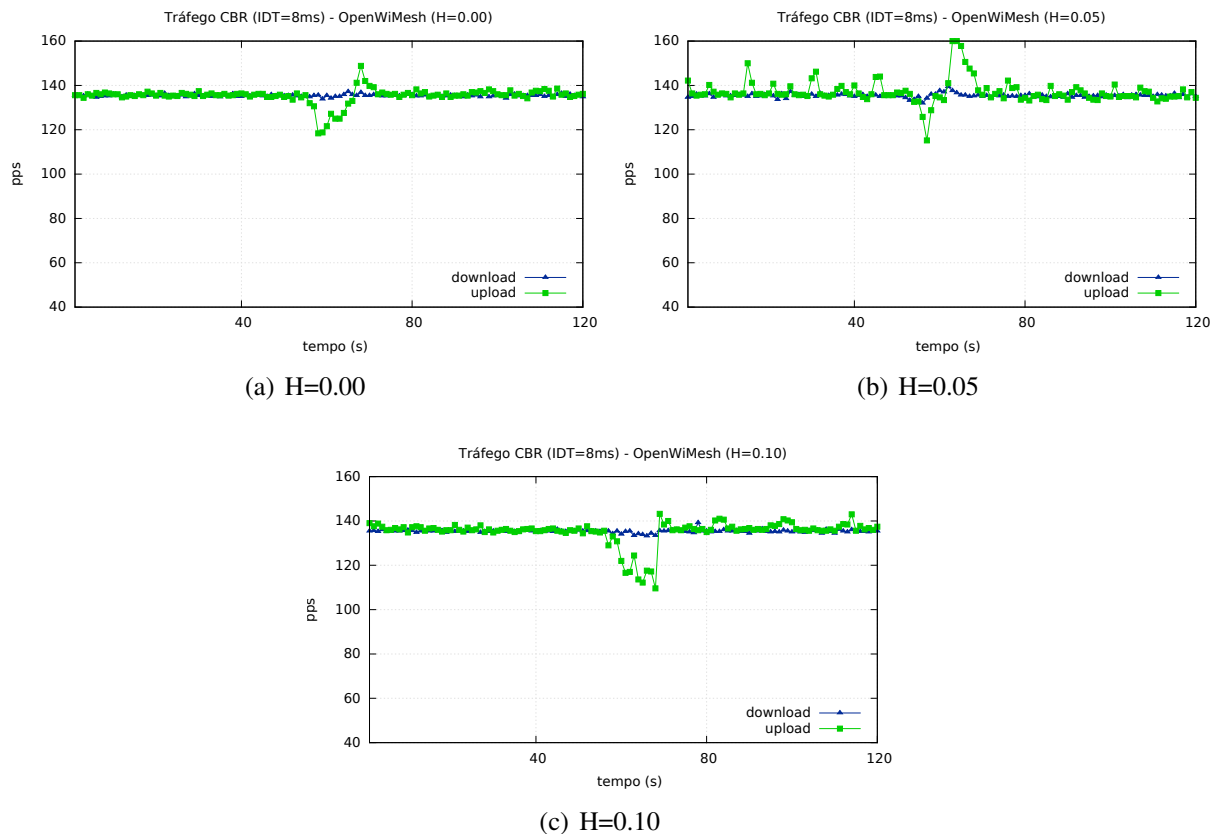


Figura 6.22: Taxa de comutação de pacotes com diferentes valores de H

incidência de *handoffs* é ocasionada pelo fato do cliente estar posicionada em uma área de cobertura Wi-Fi de ambos os APs (MR1 e MR2) e, portanto, obtendo valores oscilantes de $Link_Q$. Ao aplicar o valor $H = 0.05$ é possível notar significativa redução na quantidade de *handoffs*, dando maior estabilidade à rede. O mesmo ocorre com $H = 0.10$, porém com média de *handoffs* maior, devido a uma rodada de testes que apresentou dois *handoffs*. Nesse caso, é recomendável adotar o valor de $H = 0.05$ que mantém a estabilidade da rede ao passo que oportuniza a detecção de *handoffs* válidos.

6.5.3 PADRÃO DE MOBILIDADE

Outro aspecto que merece atenção é a relação entre o padrão de mobilidade e os resultados alcançados e apresentados nas seções anteriores. Em trabalho anterior (BRITO; FIGUEIREDO, 2015), foi avaliado a influência de diferentes padrões de mobilidade e padrão de tráfego no comparativo de estratégias de mobilidade em WMNs, concluindo-se que houve pouca diferença naquele cenário. Nesta dissertação, a relação do padrão mobilidade com a estratégia de *handoff* foi avaliada considerando diferentes velocidades de mobilidade do MC. Em especial, mensurou-se as métricas de QoS com nó móvel movendo-se a $0.5m/s$, $1.0m/s$ e $2.0m/s$. Os testes foram

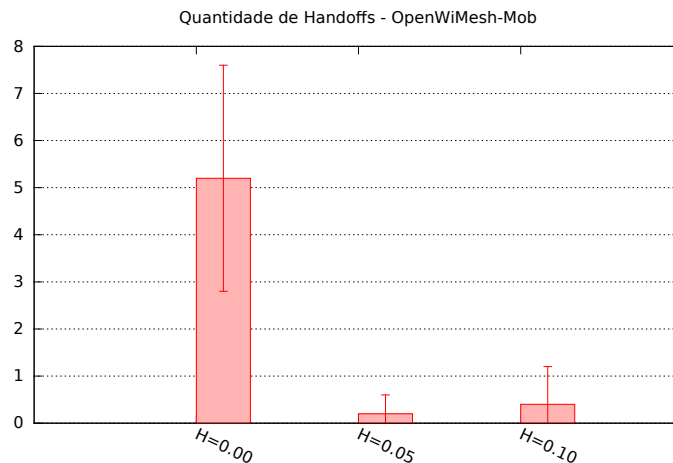


Figura 6.23: Quantidade de *handoffs* com diferentes valores de H

realizados no mesmo cenário anteriormente citado.

Salienta-se, entretanto, que a fim de manter a igualdade nas medições de ambas as estratégias, escolheu-se um instante de partida do MC diferente para cada velocidade, a saber: para velocidade $0.5m/s$, o MC inicia a movimentação aos 20 segundos; para a velocidade de $1.0m/s$, o MC inicia a movimentação aos 40 segundos; e para a velocidade de $2.0m/s$, o MC inicia a movimentação aos 50 segundos.

O gráfico da Figura 6.24 apresenta os resultados obtidos pela variação da velocidade. É possível notar ligeira diferença no momento em que o *handoff* ocorre porém sem impactos visivelmente notáveis na análise dos gráficos. Portanto, de forma similar a resultados anteriores (BRITO; FIGUEIREDO, 2015), identifica-se pouca relação entre o padrão de mobilidade e a estratégia de *handoff* nesse cenário.

6.5.4 IMAGEM DO CONTROLADOR – ESCALABILIDADE

Um dos principais componentes da solução de gerenciamento de *handoff* aqui proposta é o controlador SDN. Conforme discutido na Seção 2.3, é importante atentar-se ao dimensionamento e localização do controlador, a fim de evitar gargalos ou pontos de falha da WMN. No cenário utilizado para esta dissertação, foi adotado apenas um controlador e a seguir será apresentado uma análise acerca do seu desempenho a fim de verificar qual o impacto no funcionamento da solução proposta.

O primeiro aspecto que será analisado é o consumo de CPU. O gráfico da Figura 6.25 apresenta as estatísticas de utilização de CPU no controlador SDN do OpenWiMesh-Mob. Os

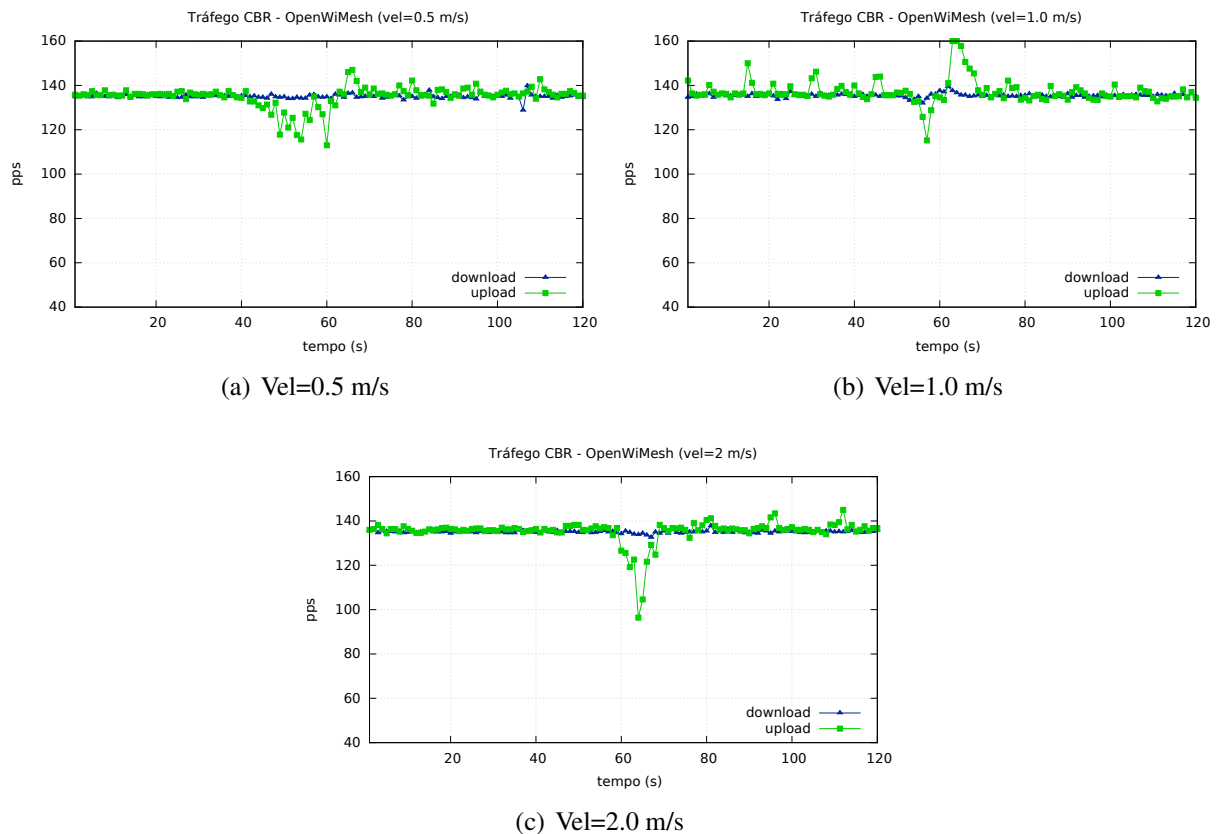


Figura 6.24: Taxa de comutação de pacotes com diferentes velocidades do MC

testes foram realizados no mesmo cenário anteriormente citado, porém capturando dados do controlador durante 40 segundos antes do teste e 40 segundos depois. É possível observar que o consumo de CPU permanece constante e abaixo de 10% ao longo de todo o experimento, mostrando que nesse cenário um controlador foi suficiente para acomodar confortavelmente os MRs e funcionamento da rede.

Já o gráfico da Figura 6.25 ilustra a utilização de memória RAM no controlador SDN do OpenWiMesh-Mob. É possível notar que o consumo de memória também mantém-se constante ao longo da execução do experimento, mesmo antes da sua inicialização e após sua conclusão. O consumo de memória, portanto, para a aplicação OpenWiMesh-Mob não acarreta impactos de ordem significativa ao controlador.

6.5.5 ISOLAMENTO DE EFEITOS DA ESTRATÉGIA DE *HANDOFF*

Nesta subseção serão analisados os efeitos de cada componente da estratégia de *handoff* apresentada, a saber: i) métrica de qualidade do enlace Wi-Fi; ii) algoritmo de decisão de *handoff* com limiar e histerese; e iii) algoritmo de execução do *handoff* com duplicação de

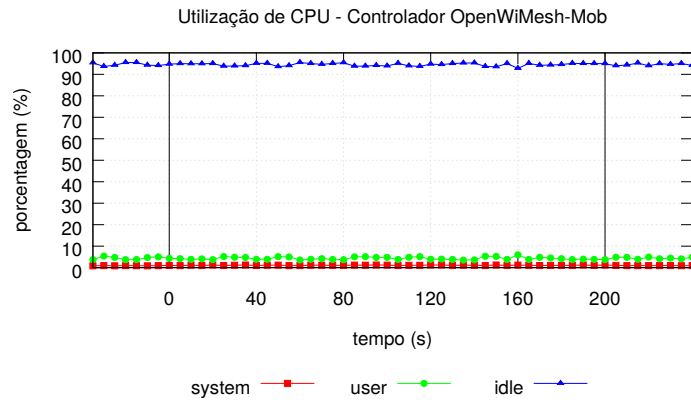


Figura 6.25: Utilização de CPU no controlador do OpenWiMesh-Mob

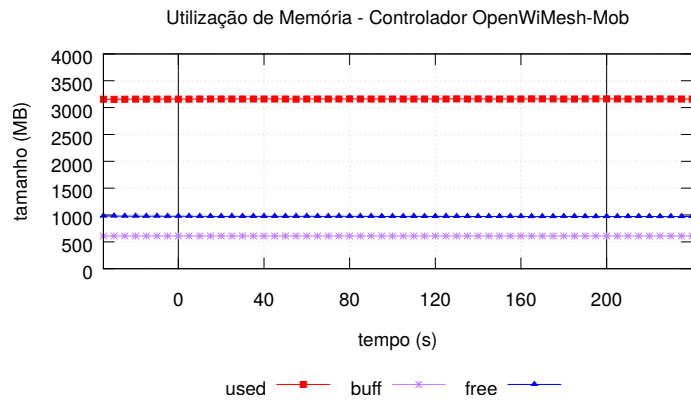


Figura 6.26: Utilização de memória no controlador do OpenWiMesh-Mob

tráfego.

Para analisar isoladamente os efeitos da métrica de qualidade $Link_Q$, foi utilizado um *trace* da execução do teste com MC parado entre os dois MRs (conforme Subseção 6.5.2) e com fonte de interferência externa, que gerou retransmissões de pacotes em determinado trecho do teste. O objetivo é mostrar como a métrica composta pelo RSSI e FRR (taxa de retransmissão) pode reagir melhor que uma métrica simples baseada somente em RSSI.

A Figura 6.28a mostra as medidas de RSSI e FRR durante o *trace* em questão. Já a Figura 6.28b apresenta os valores de qualidade do enlace baseada simplesmente no RSSI ($Simp - Link_Q$) e baseada na composição entre RSSI e FRR ($Comp - Link_Q$). É possível observar que a métrica simples baseada apenas em RSSI não sofre alterações significativas mesmo face ao aumento na taxa de retransmissões, ao passo que a métrica proposta nesta dissertação reflete o impacto do FRR e decrementa o valor de $Link_Q$. Tal cenário poderia levar o algoritmo de

handoff a escolher outro MR com melhores condições de atender ao MC, dependendo do valor de T e H , sendo, portanto, mais vantajosa a sua utilização.

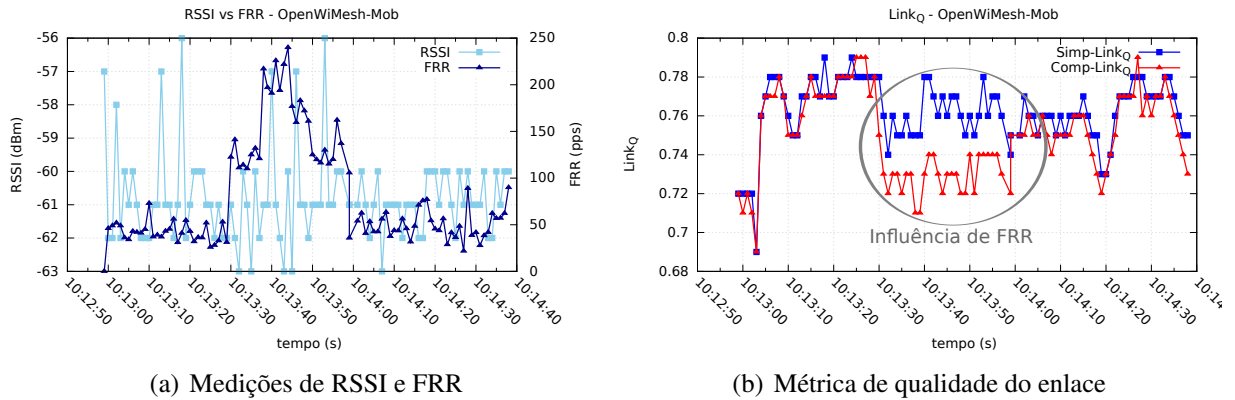


Figura 6.27: Comparação da métrica de qualidade de enlace face às medições de RSSI e FRR

A análise dos efeitos do algoritmo de decisão de *handoff* com limiar e histerese pode ser feita com base nos gráficos apresentados nas Subseções 6.5.1 e 6.5.2. Em particular, os gráficos da Figura 6.23 mostraram que se não for utilizado o parâmetro limiar T a rede estará sujeita a *handoffs* desnecessários ou executados em momento inoportunos. Por outro lado, a escolha do valor de histerese evita a ocorrência do efeito *ping-ponging*.

Por fim, outro aspecto importante a ser considerado na estratégia de *handoff* proposta é o efeito da abordagem de duplicação de tráfego. Conforme apresentado na Seção 5.4, a abordagem de manter o caminho anterior ativo, embora gere duplicação de pacotes, aumenta a probabilidade de recepção de pacotes do cliente, sendo importante na manutenção das métricas de QoS da rede. Para medir a contribuição específica dessa abordagem, foram realizados um conjunto de testes no mesmo ambiente apresentado anteriormente, porém alternando entre a ativação da abordagem de duplicação versus a sua desativação. A Figura ?? apresenta os resultados de cada cenário.

Na Figura ?? é possível observar uma sutil diferença na janela de degradação durante o *handoff*. A abordagem sem duplicação de tráfego possui uma janela de degradação de 7 segundos em média, ao passo que com duplicação de tráfego possui uma janela de degradação de apenas 3 segundos. Sem a estratégia de duplicação de tráfego, a perda de pacotes durante o trecho de mobilidade alcança uma média de 1.6% enquanto que com a estratégia a perda limita-se a uma média de 0.65%. Dessa maneira, é possível observar que o uso da abordagem de manutenção do caminho antigo traz benefícios importantes para o cliente.

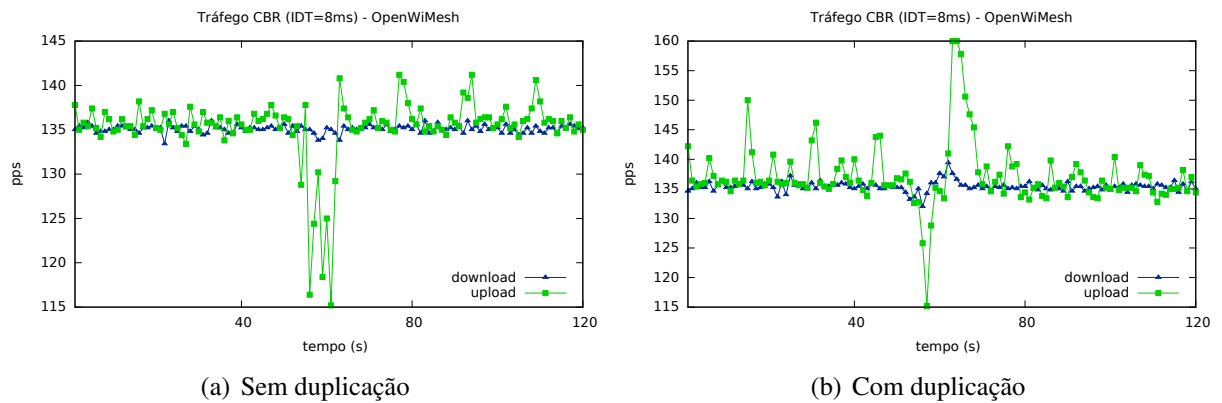


Figura 6.28: Análise dos efeitos da estratégia de duplicação de tráfego

6.6 CONSIDERAÇÕES PARCIAIS

Este capítulo mostrou um estudo avaliativo e comparativo da proposta de estratégia de *handoff* transparente (OpenWiMesh-Mob), em relação a uma estratégia comumente utilizada na literatura, composta pelo B.A.T.M.A.N. e *handoff* controlado pelo cliente (BATMAN-MCHO). A avaliação levou em consideração métricas de QoS e QoE, fornecendo uma análise aprofundada dos diversos aspectos que podem ser impactados pela mobilidade dos usuários.

No que tange às métricas de QoS, a taxa de comutação de pacotes manteve-se próxima ao valor esperado para o modelo de tráfego usado, com exceção do tráfego de *uplink*, que sofreu pequenas oscilações devido à implantação padrão do software AP (cf. §6.3.1). O atraso bidirecional esteve sempre abaixo de 150 ms, com diferença significativa em comparação ao BATMAN-MCHO. O *jitter* manteve-se abaixo de 20 ms, em conformidade com requisições de aplicações de tempo real. Ambas as métricas de potência de sinal e taxa de retransmissão, também corroboraram com a eficácia e eficiência do gerenciamento de mobilidade ora proposto.

Em se tratando do QoE, os resultados alcançados pela estratégia OpenWiMesh-Mob, notadamente superiores à estratégia comparada e dentro das expectativas de alta qualidade dos usuários, consubstanciam para a viabilidade técnica da proposta e para a relevância e efetividade das métricas e algoritmos apresentados.

Por fim, em todos os testes realizados, a estratégia OpenWiMesh-Mob mostrou-se vantajosa, atingindo níveis elevados de qualidade e superando a estratégia comparada. Destaca-se, ainda, que as melhorias incorporadas no suporte à mobilidade são independentes dos clientes móveis, aumentando a abrangência de dispositivos que podem se beneficiar delas.

7 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho apresentou uma proposta de métricas de monitoramento da qualidade de enlaces Wi-Fi e um algoritmo de decisão de *handoff* controlado pela rede, utilizando a abordagem de Redes Definidas por Software em uma rede *mesh* sem fio. Tais estratégias foram empregadas no desenvolvimento de um protótipo, no *framework* OpenWiMesh, tornando-a factível para utilização em ambientes reais e emulados. Através de minuciosas análises, tanto do ponto de vista da QoS da rede, quanto da QoE no uso de aplicações de tempo real, foi possível constatar que as abordagens ora propostas melhoram substancialmente a qualidade da rede. O OpenWiMesh-Mob é agnóstico à plataforma de S.O. do cliente e não requer qualquer configuração ou protocolo adicional, viabilizando o gerenciamento de mobilidade eficiente e transparente para dispositivos móveis, requisito sobremaneira importante nas redes sem fio modernas.

O uso da abordagem SDN favoreceu a adoção do controle centralizado e a correlação de informações multicamada no processo de *handoff*. Além disso, a execução coordenada entre os roteadores *mesh* culmina na realização do *handoff* de forma otimizada, tanto do ponto de vista do cliente quanto da própria infraestrutura *mesh*. Notadamente, o paradigma SDN permite, ainda, grande flexibilidade e extensibilidade da proposta, que pode ser facilmente adaptada para atender a requisitos específicos de determinada aplicação ou cenário.

Na avaliação da qualidade da proposta, as métricas diretamente relacionadas com a rede mostraram relevantes benefícios das estratégias aqui propostas: até 14.03% de melhoria em relação a média de taxa de comutação; redução de até 200 ms no atraso bidirecional durante o *handoff*; variação de atraso sempre inferior a 20 ms, em conformidade com requisitos de aplicações de VoIP e conferência de vídeo; melhor aproveitamento do sinal Wi-Fi; e menor sobrecarga na rede com redução na quantidade de retransmissões do 802.11.

Avaliou-se também o impacto da proposta na percepção dos usuários, através de métricas de QoE. A qualidade de experiência foi checada em termos da qualidade de vídeo transmitido pela rede durante o *handoff*. Nas duas métricas avaliadas, PSNR e SSIM, a proposta obteve

índices de qualidade média acima do aceitável em todos os momentos do teste, com avaliação global de MOS excelente, levando-se em consideração a média e desvio padrão em ambas as métricas.

O código-fonte, scripts de configuração e dados da avaliação estão todos disponíveis para análise e comparativos futuros, podendo ser usados ainda como base para desenvolvimento de novos algoritmos e estudos comparativos.

Por fim, considera-se que o trabalho ora desenvolvido instigou outras pesquisa no tema, resultando em trabalhos de conclusão de curso, apresentação de artigos, participação em bancas e outros trabalhos científicos em andamento.

7.1 TRABALHOS FUTUROS

Algumas possibilidades de extensão ou complementação dessa proposta, que podem ser realizados posteriormente, são listados a seguir. Um dos trabalhos futuros possíveis refere-se a experimentação com diferentes padrões de mobilidade. Neste trabalho, o OpenWiMesh-Mob foi avaliado em um ambiente real, a partir de diferentes critérios de QoS e QoE, porém é importante utilizar diferentes padrões de mobilidade nos experimentos, avaliando se o comportamento do *framework* permanece o mesmo. Outrossim, o OpenWiMesh-Mob foi avaliado com base em redes 802.11 (Wi-Fi), porém é importante validar sua integração com outras tecnologias como WiMAX e LTE. É importante projetar, ainda, estratégias de gerenciamento de *handoff* interdomínio, principalmente para dar suporte às redes de quinta geração (5G);

Outro trabalho futuro que pretende-se executar é a comparação da estratégia ora proposta com uma abordagem também controlada pela rede, pois o estudo comparativo aqui conduzido leva em consideração apenas uma abordagem controlada pelo cliente. Comparar o OpenWiMesh-Mob com outra abordagem baseada na rede poderá fornecer melhores recomendações aos usuários dessa estratégia sobre suas vantagens e oportunidades de melhoria.

A ampliação do *testbed* utilizado na experimentação é outra ação que pode ser executada posteriormente. Espera-se ampliar o *testbed* 802.11/OpenFlow utilizado nesse trabalho, construindo uma rede sem fio para experimentação no campos da Universidade Federal da Bahia, o que poderia viabilizar avaliações de grande escala para o *framework* e para outros trabalhos de pesquisa.

É importante investigar profundamente o funcionamento do software AP utilizado (*hostapd*), estendendo suas funcionalidades para permitir a pré-configuração dos parâmetros de es-

tado da associação Wi-Fi, evitando perdas de pacotes no *uplink* durante o *handoff* (cf. §6.3.1).

Por fim, deve-se trabalhar na melhoria da segurança da rede, a partir do uso de criptografia e autenticação do tráfego de dados e de controle.

REFERÊNCIAS

- AKYILDIZ, I.; WANG, X. *Wireless Mesh Networks*. [S.l.]: John Wiley & Sons, 2009.
- AMIR, Y. et al. The SMesh Wireless Mesh Network. *ACM Transactions on Computer Systems (TOCS)*, ACM, v. 28, n. 3, p. 6:1–6:49, 2010.
- APOSTOLARAS, A. et al. C2m: Mobile data offloading to mesh networks. In: IEEE. *Global Communications Conference (GLOBECOM), 2014 IEEE*. [S.l.], 2014. p. 4877–4883.
- ARAUJO, M. R. *Distributed-OpenWiMesh: Uma arquitetura de múltiplos controladores SDN para redes mesh sem fio*. Monografia (Graduação) — Departamento de Ciência da Computação, Instituto de Matemática, Universidade Federal da Bahia, Salvador, Bahia, 2015.
- AVELAR, E. A. M. *PMIPFlow: Uma proposta para gerenciamento de mobilidade em redes definidas por software*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2013.
- AVELAR, E. A. M.; AVELAR, L. M.; DIAS, K. L. Pmipflow: Uma proposta para gerenciamento de mobilidade em redes definidas por software. *Anais do XXXIII Congresso da SBC, XII Workshop em Desempenho de Sistemas Computacionais e de Comunicação (WPerformance)*, p. 820–833, 2013.
- BENYAMINA, D.; HAFID, A.; GENDREAU, M. Wireless mesh networks design - a survey. *Communications Surveys & Tutorials, IEEE*, IEEE, v. 14, n. 2, p. 299–310, 2012.
- BRITO, I.; FIGUEIREDO, G. B. Suporte à Mobilidade em Redes Mesh Sem Fio: estratégias comuns versus SDN. *arXiv preprint arXiv:1512.02594*, 2015.
- BRITO, I. et al. Openwimesh: um framework para redes mesh sem fio definidas por software. In: *32th Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*. [S.l.: s.n.], 2014. p. 413–426.
- BRITTON, M.; COYLE, A. Performance analysis of the B.A.T.M.A.N. wireless ad-hoc network routing protocol with mobility and directional antennas. In: *Military Communications and Information Systems Conference (MilCIS), 2011*. [S.l.: s.n.], 2011. p. 1–6.
- CHUNG, J. et al. Experiences and challenges in deploying openflow over a real wireless mesh network. In: IEEE. *Communications (LATINCOM), 2012 IEEE Latin-America Conference on*. [S.l.], 2012. p. 1–5.
- CISCO. *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015-2020*. 2015. Último acesso em 16 de Abril de 2016. Disponível em: <<http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html>>.

- CLAUSEN, T.; JACQUET, P. *Optimized Link State Routing Protocol (OLSR)*. IETF, out. 2003. 62 – 68 p. RFC 3626. (Request for Comments, 3626). Último acesso em 06 de Outubro de 2013. Disponível em: <<http://www.ietf.org/rfc/rfc3626.txt>>.
- COSTANZO, S. et al. Software defined wireless networks: Unbridling sdns. In: IEEE. *European Workshop on Software Defined Networking (EWSDN)*. [S.l.], 2012. p. 1–6.
- CROITORU, A.; NICULESCU, D.; RAICIU, C. Towards wifi mobility without fast handover. In: *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*. [S.l.: s.n.], 2015. p. 219–234.
- CROW, B. P. et al. Ieee 802.11 wireless local area networks. *Communications Magazine, IEEE*, IEEE, v. 35, n. 9, p. 116–126, 1997.
- DELY, P.; KASSLER, A.; BAYER, N. Openflow for wireless mesh networks. In: IEEE. *20th International Conference on Computer Communications and Networks (ICCCN)*. [S.l.], 2011. p. 1–6.
- DETTI, A. et al. Wireless mesh software defined networks (wmsdn). In: IEEE. *2013 IEEE 9th international conference on wireless and mobile computing, networking and communications (WiMob)*. [S.l.], 2013. p. 89–95.
- ENGELKE, U. et al. Reduced-reference metric design for objective perceptual quality assessment in wireless imaging. *Signal Processing: Image Communication*, Elsevier, v. 24, n. 7, p. 525–547, 2009.
- ESLAMI, M.; KARIMI, O.; KHODADADI, T. A survey on wireless mesh networks: Architecture, specifications and challenges. In: *Control and System Graduate Research Colloquium (ICSGRC), 2014 IEEE 5th*. [S.l.: s.n.], 2014. p. 219–222.
- FEAMSTER, N.; BALAKRISHNAN, H. Packet loss recovery for streaming video. In: PA: PITTSBURGH. *12th International Packet Video Workshop*. [S.l.], 2002. p. 9–16.
- FERREIRA, I. F. *Esquema de roteamento multicaminho em redes MESH sem fio com OpenFlow para suporte a aplicações VoIP*. Monografia (Graduação) — Departamento de Ciência da Computação, Instituto de Matemática, Universidade Federal da Bahia, Salvador, Bahia, 2013.
- GOMES, R. et al. A qoe fuzzy routing protocol for wireless mesh networks. In: *Future Multimedia Networking*. [S.l.]: Springer, 2010. p. 1–12.
- GUDE, N. et al. Nox: towards an operating system for networks. *ACM SIGCOMM Computer Communication Review*, ACM, v. 38, n. 3, p. 105–110, 2008.
- GUERRA, E. *Design Patterns com Java: Projeto orientado a objetos guiado por padrões*. [S.l.]: Editora Casa do Código, 2014.
- GUNDAVELLI, S. et al. *Proxy Mobile IPv6*. [S.l.]: IETF, ago. 2008. RFC 5213. (Request for Comments, 5213).
- HAGBERG, A. A.; SCHULT, D. A.; SWART, P. J. Exploring network structure, dynamics, and function using NetworkX. In: *Proceedings of the 7th Python in Science Conference (SciPy2008)*. [S.l.: s.n.], 2008. p. 11–16.

- HELLER, B.; SHERWOOD, R.; MCKEOWN, N. The controller placement problem. In: ACM. *Proceedings of the first workshop on Hot topics in software defined networks*. [S.l.], 2012. p. 7–12.
- HONG, K. et al. Channel measurement-based access point selection in IEEE 802.11 WLANs. *Pervasive and Mobile Computing*, Elsevier, v. 30, p. 58 – 70, 2015.
- HSIEH, I.-P. et al. Adaptive access points selection for 802.11 wireless networks. In: *Proceeding AsiaCSN '07 Proceedings of the Fourth IASTED Asian Conference on Communication Systems and Networks*. [S.l.: s.n.], 2007. p. 218–222.
- IEEE. IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2: Fast Basic Service Set (BSS) Transition. *IEEE Std 802.11r-2008 (Amendment to IEEE Std 802.11-2007)*, p. 1–126, July 2008.
- IEEE. IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 1: Radio Resource Measurement of Wireless LANs. *IEEE Std 802.11k-2008 (Amendment to IEEE Std 802.11-2007)*, p. 1–244, June 2008.
- IEEE. IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 8: IEEE 802.11 Wireless Network Management. *IEEE Std 802.11v-2011 (Amendment to IEEE Std 802.11-2007)*, p. 1–433, Feb 2011.
- IEEE. IEEE Standard for Information Technology–Telecommunications and information exchange between systems–Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 10: Mesh Networking. *IEEE Std 802.11s-2011 (Amendment to IEEE Std 802.11-2007)*, p. 1–372, Sept 2011.
- ITU-T. One-way transmission time. *Recommendation G*, v. 114, 2003. Último acesso em 20 de Abril de 2016.
- JACQUET, P. et al. Optimized link state routing protocol for ad hoc networks. In: *IEEE International Multi Topic Conference (IEEE INMIC). Technology for the 21st Century. Proceedings*. [S.l.: s.n.], 2001.
- JOHNSON, D.; NTLATLAPA, N.; AICHELE, C. Simple pragmatic approach to mesh routing using batman. *2nd IFIP International Symposium on Wireless Communications and Information Technology in Developing Countries, CSIR*, p. 62 – 68, 2008.
- KOBAYASHI, M. et al. Maturing of openflow and software-defined networking through deployments. *Computer Networks*, Elsevier, v. 61, p. 151–175, 2014.
- KOVÁCS, I. Z. et al. Mobile broadband traffic forecast modeling for network evolution studies. In: IEEE. *Vehicular Technology Conference (VTC Fall), 2011 IEEE*. [S.l.], 2011. p. 1–5.
- KREUTZ, D. et al. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, IEEE, v. 103, n. 1, p. 14–76, 2015.

- LAMBRECHT, C. J. v. d. B.; VERSCHURE, O. Perceptual quality measure using a spatiotemporal model of the human visual system. In: *Electronic Imaging: Science & Technology*. [S.l.: s.n.], 1996. p. 450–461.
- LAMPROPOULOS, G.; SALKINTZIS, A. K.; PASSAS, N. Media-independent handover for seamless service provision in heterogeneous networks. *Communications Magazine, IEEE*, IEEE, v. 46, n. 1, p. 64–71, 2008.
- LEE, S. et al. Forecasting mobile broadband traffic: Application of scenario analysis and delphi method. *Expert Systems with Applications*, Elsevier, v. 44, p. 126–137, 2016.
- LIU, Z. et al. A multi-rate access point selection policy in IEEE 802.11 WLANs. In: IEEE. *Multimedia Technology (ICMT), 2011 International Conference on*. [S.l.], 2011. p. 63–67.
- MCKEOWN, N. *Software-defined Networking*. abr. 2009. Infocom Keynote Talk.
- MCKEOWN, N. et al. Openflow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, ACM, v. 38, n. 2, p. 69–74, 2008.
- MISHRA, A.; SHIN, M.; ARBAUGH, W. An empirical analysis of the IEEE 802.11 MAC layer handoff process. *ACM SIGCOMM Computer Communication Review*, ACM, v. 33, n. 2, p. 93–102, 2003.
- MISHRA, S.; DAS, A. X.; JAISAWAL, A. Effect of mobility and different data traffic in wireless ad-hoc network through QualNet. *International Journal of Engineering and Advanced Technology (IJEAT)*, Citeseer, v. 2, n. 5, p. 364–368, 2013.
- MSU Video Group. *Video Quality Measurement Tool*. 2013. URL: http://www.compression.ru/video/quality_measure/video_measurement_tool_en.html. Último acesso em 20 de Abril de 2016.
- MWELA, J. S. *Impact of Packet Loss on the Quality of Video Stream Transmission*. Tese (Doutorado) — Blekinge Institute of Technology, 2010.
- NAEEM, B.; NGAH, R.; HASHIM, S. Z. M. Handovers in small cell based heterogeneous networks. In: *2016 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)*. [S.l.: s.n.], 2016. p. 268–271.
- NASCIMENTO, V. et al. Filling the gap between software defined networking and wireless mesh networks. In: IEEE. *Network and Service Management (CNSM), 2014 10th International Conference on*. [S.l.], 2014. p. 451–454.
- NICHOLSON, A. J. et al. Improved access point selection. In: *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services*. [S.l.]: ACM, 2006. (MobiSys '06), p. 233–245.
- NLANR/DAST. *A modern Alternative for Measuring Maximum TCP and UDP*. 2013.
- OMHANI, N.; ZARAI, F.; KAMOUN, L. Efficient MAC layer handoff management scheme for wireless mesh networks. In: IEEE. *Telecommunications (ICT), 2010 IEEE 17th International Conference on*. [S.l.], 2010. p. 387–392.
- ONF. *OpenFlow Switch Specification v1.3.0*. jun. 2012. ONF Specification.

- ONF. *Software-Defined Networking: The New Norm for Networks*. abr. 2012. White Paper.
- Open vSwitch. *Open Virtual Switch*. 2013. URL: <http://openvswitch.org/>. Último acesso em 20 de Abril de 2016.
- OPENVSWITCH. *Open vSwitch: an open virtual switch*. 2013. Último acesso em 10 de Outubro de 2013. Disponível em: [<http://openvswitch.org/>](http://openvswitch.org/).
- OpenWiMesh. *OpenWiMesh website and GIT repository*. 2013. URL: <http://grade.dcc.ufba.br/OpenWiMesh>. Último acesso em 20 de Abril de 2016.
- PAIVA, R. B. et al. Gerenciamento de recursos no processo de handoff em redes sem fio definidas por software. *III Workshop de Pesquisa Experimental da Internet do Futuro (WPEIF)*, p. 23–28, 2013.
- PAUL, L. C. Handoff/handover mechanism for mobility improvement in wireless communication. *Global Journal of Researches In Engineering*, v. 13, n. 16, 2014.
- PERKINS, C.; BELDING-ROYER, E.; DAS, S. *Ad hoc On-Demand Distance Vector (AODV) Routing*. IETF, jul. 2003. RFC 3561. (Request for Comments, 3561). Último acesso em 06 de Outubro de 2013. Disponível em: [<http://www.ietf.org/rfc/rfc3561.txt>](http://www.ietf.org/rfc/rfc3561.txt).
- PERKINS, C. E. *IP Mobility Support for IPv4, Revised*. IETF, nov. 2010. RFC 5944. (Request for Comments, 5944). Disponível em: [<https://rfc-editor.org/rfc/rfc5944.txt>](https://rfc-editor.org/rfc/rfc5944.txt).
- PHAN, X. T.; THOAI, N.; KUONEN, P. A collaborative model for routing in multi-domains openflow networks. In: IEEE. *Computing, Management and Telecommunications (ComManTel), 2013 International Conference on*. [S.l.], 2013. p. 278–283.
- POX. *Python-based OpenFlow Controller*. 2013. Último acesso em 10 de Outubro de 2013. Disponível em: [<http://www.noxrepo.org/pox/about-pox/>](http://www.noxrepo.org/pox/about-pox/).
- REINER, M.; CASETTI, C.; CHIASSERINI, C. F. Routing protocols for mesh networks with mobility support. In: *2009 6th International Symposium on Wireless Communication Systems*. [S.l.: s.n.], 2009. p. 71–75.
- SANCHEZ-IBORRA, R.; CANO, M.-D. Qoe-based performance evaluation of video transmission using the batman routing protocol. In: *Proceedings of the 10th ACM Symposium on QoS and Security for Wireless and Mobile Networks*. New York, NY, USA: ACM, 2014. (Q2SWinet '14), p. 9–16. Disponível em: [<http://doi.acm.org/10.1145/2642687.2642694>](http://doi.acm.org/10.1145/2642687.2642694).
- SANTOS, J. S. *Distribuição de conteúdo multimídia em redes em malha sem fio (provisório)*. Dissertação (Mestrado) — Departamento de Ciência da Computação, Instituto de Matemática, Universidade Federal da Bahia, Salvador, Bahia, 2016. Trabalho em Andamento.
- SCHULZ-ZANDER, J. et al. Programmatic orchestration of wifi networks. In: USENIX ASSOCIATION. *2014 USENIX Annual Technical Conference (USENIX ATC 14)*. [S.l.], 2014. p. 347–358.
- TSUKAMOTO, K.; YAMAGUCHI, T.; KASHIHARA, S. Experimental evaluation of decision criteria for wlan handover: Signal strength and frame retransmission. *IEICE transactions on communications*, The Institute of Electronics, Information and Communication Engineers, v. 90, n. 12, p. 3579–3590, 2007.

- VASUDEVAN, S. et al. Facilitating access point selection in ieee 802.11 wireless networks. In: USENIX ASSOCIATION. *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*. [S.l.], 2005. p. 26–26.
- WU, D.; DJUKIC, P.; MOHAPATRA, P. Determining 802.11 link quality with passive measurements. In: IEEE. *Wireless Communication Systems. 2008. ISWCS'08. IEEE International Symposium on*. [S.l.], 2008. p. 728–732.
- WU, S.-J. Fuzzy-based handover decision scheme for next-generation heterogeneous wireless networks. *Journal of Convergence Information Technology*, Advanced Institute of Convergence IT, v. 6, n. 4, p. 285 – 297, 2011.
- XU, F. et al. Smartassoc: Decentralized access point selection algorithm to improve throughput. *Parallel and Distributed Systems, IEEE Transactions on*, IEEE, v. 24, n. 12, p. 2482–2491, 2013.
- YAP, K.-K. et al. The stanford openroads deployment. In: ACM. *Proceedings of the 4th ACM international workshop on Experimental evaluation and characterization*. [S.l.], 2009. p. 59–66.
- YAP, K.-K. et al. Blueprint for introducing innovation into wireless mobile networks. In: *Proceedings of the Second ACM SIGCOMM Workshop on Virtualized Infrastructure Systems and Architectures*. [S.l.]: ACM, 2010. p. 25–32.