

# Mobilidade transparente em redes mesh sem fio definidas por software

Italo Brito<sup>1</sup>, Gustavo B. Figueiredo<sup>1</sup>

<sup>1</sup>Departamento de Ciência da Computação – Universidade Federal da Bahia (UFBA)  
Av. Adhemar de Barros, s/n – Campus Ondina – Salvador, BA – Brasil

{italo,gustavo}@dcc.ufba.br

**Abstract.** *The ever increasing mobile data traffic has made the support for seamless mobility an imperative requirement for network providers. To deliver enhanced mobility experience, the handoff process must avoid interruption of connections while preserving QoS and QoE at acceptable levels. However, traditional mobility strategies based on the 802.11 standard impose high latency and QoS degradation, deteriorating the application performance. In this paper we present a new strategy to provide seamless mobility to unmodified clients in wireless mesh networks that considers network controlled handoff and passive measurements. Experimental results show the effectiveness of the proposed strategy in providing seamless handoff, improving QoE up to 70% while keeping key QoS metrics below real-time applications constraints.*

**Resumo.** *O uso de dispositivos móveis tem crescido de tal maneira que o suporte a mobilidade transparente é requisito crucial para os provedores de rede. De modo a proporcionar uma boa experiência de mobilidade aos usuários, o processo de handoff deve preservar a conectividade e bons níveis de QoS e QoE. Por outro lado, as estratégias de gerenciamento de mobilidade baseadas no padrão 802.11 incorrem em latência e degradação da QoS na rede, impactando negativamente o desempenho das aplicações. Este trabalho apresenta uma estratégia de handoff definido por software e baseado em monitoramento passivo da qualidade dos enlaces, onde o processo de handoff é inteiramente controlado pela rede. Resultados experimentais em um testebed mostram uma melhora de até 70% nas métricas de QoE durante a mobilidade de clientes, ao passo que métricas chave de QoS mantiveram-se abaixo de limites de restrição para aplicações de tempo real.*

## 1. Introdução

O número de dispositivos portáteis (e.g. *notebooks, tablets, smartphones*) conectados às redes sem fio tem crescido sobremaneira nos últimos anos [Lee et al. 2016]. Esse crescimento, associado à necessidade dos usuários estarem sempre conectados, torna clara a demanda por mobilidade eficiente entre os clientes móveis (MC, do inglês *Mobile Client*) e os pontos de acesso (AP, do inglês *Access Point*) sem fio. O processo de migração da associação entre MCs e APs, conhecido como *handoff* ou *handover*, idealmente, deve ser transparente para os nós móveis, dispensando configurações ou protocolos adicionais e mantendo a conectividade sem interrupções. Deve-se, portanto, preservar a Qualidade de Serviço (QoS) das conexões durante o *handoff*, evitando atraso excessivo, variação

de atraso e perda de pacotes. Além disso, deve-se também manter a Qualidade de Experiência (QoE) dos usuários em níveis aceitáveis, sem deteriorar a qualidade de transmissão na percepção dos usuários. A observância desses requisitos é particularmente importante para suporte a aplicações de tempo real e de multimídia interativa.

As técnicas tradicionais de suporte à mobilidade em redes sem fio, incluindo aquelas disponíveis para dispositivos que adotam o padrão 802.11 (*Wi-Fi*), 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 a congestionamento no AP, retransmissões excessivas e perda de pacotes. Outros mecanismos [Croitoru et al. 2015, Dely et al. 2011] requerem modificações ou inserção de protocolos adicionais no cliente, tornando-se um obstáculo para sua ampla adoção [Schulz-Zander et al. 2014].

Este trabalho apresenta uma abordagem para mobilidade transparente em redes *mesh* sem fio, por meio do gerenciamento de *handoff* baseado no paradigma de Redes Definidas por Software (SDN, do inglês *Software-Defined Networking*) e monitoramento passivo da qualidade dos enlaces Wi-Fi. 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 mantém os dispositivos clientes livres de configurações adicionais. Foi realizada uma avaliação de desempenho detalhada da proposta, em comparação com uma implantação baseada no protocolo B.A.T.M.A.N. [Johnson et al. 2008] e *handoff* controlado pelo cliente, utilizando o padrão 802.11. Levou-se em consideração métricas de QoS e QoE para avaliar o impacto do *handoff* na transmissão de dados. 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.

O restante do trabalho está dividido da seguinte forma. A Seção 2 traz um resumo de trabalhos correlatos. A Seção 3 apresenta a estratégia de *handoff* proposta neste trabalho. Em seguida, na Seção 4, experimentos comparativos são apresentados e os resultados são discutidos. Por fim, na Seção 5 presente-se as conclusões e trabalhos futuros.

## 2. Trabalhos relacionados

Dely et al. [Dely et al. 2011] foi um dos primeiros a integrar SDN em redes *mesh*, propondo uma arquitetura para controle e monitoramento baseado em encaminhamento por fluxos. Como estudo de caso, foi implementada uma função de gerenciamento de mobilidade onde o controlador monitora continuamente a qualidade dos enlaces e a capacidade dos APs, disparando uma ação de *handoff* no cliente, através do protocolo IEEE 802.21, quando detecta a mobilidade. Além do suporte a 802.21, os clientes devem ser modificados para instalação do agente de monitoramento ativo da rede sem fio. Diferente do trabalho supracitado, a proposta aqui apresentada é independente de alterações no cliente e realiza o monitoramento passivo da rede.

Em termos de padronizações, o IEEE tem abordado o *handoff* de redes IP por meio de diferentes normas e complementos. O complemento 802.11k, por exemplo, re-

duz o tempo do *handoff* pois permite que o cliente determine mais rapidamente a qual AP ele deve se associar, através de informações sobre a vizinhança fornecida pelo AP atual. Já o 802.11r otimiza o tempo de autenticação pelo uso de *cache* das chaves de criptografia nos APs. O complemento 802.11v permite que os APs compartilhem com o cliente informações de sobrecarga da rede, a fim de melhorar a escolha do cliente pelo novo AP. Já o padrão 802.21 visa prover mobilidade entre diferentes tipos de rede, também conhecido como *handoff independente de mídia*. Todas essas abordagens requerem alterações no padrão 802.11 e, conseqüentemente, nos APs e nos clientes. Neste trabalho, nenhuma modificação no padrão 802.11 é necessária.

Avelar [Avelar 2013] propõe uma arquitetura baseada em SDN para gerenciamento de mobilidade em redes IP inspirada no protocolo PMIPv6 (do inglês *Proxy Mobile IPv6*), chamada PMIPFlow. Apresenta-se também um mecanismo de antecipação de *handoff* baseado na lógica *fuzzy* (PMIPFlow-Fuzzy) para proporcionar mobilidade transparente. A antecipação do *handoff* é possível para clientes que executam um agente do PMIPFlow-Fuzzy, que utiliza três variáveis (RTT, Vazão e RSSI) na criação das regras de inferência do sistema *fuzzy*. Demonstrou-se que a proposta de antecipação do *handoff* permite mobilidade transparente e com baixo impacto na qualidade de experiência de aplicações como vídeo *streaming*. Todavia, a necessidade de instalação de um agente PMIPFlow-Fuzzy pode se tornar uma barreira para amplo benefício dos nós clientes. Além disso, o monitoramento ativo pode gerar sobrecarga adicional na rede.

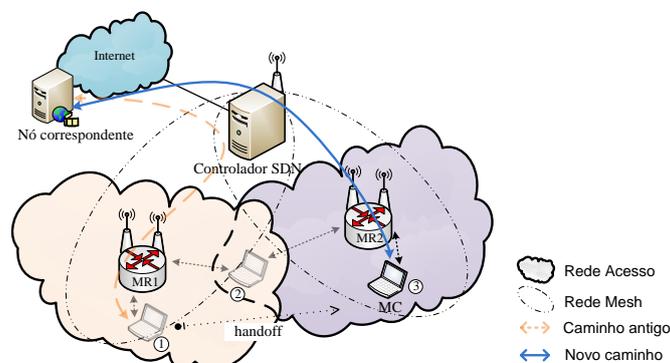
Schulz-Zander et al. [Schulz-Zander et al. 2014] propõem uma solução para orquestração de redes *Wi-Fi* com SDN, chamada Odin. Entre as contribuições do trabalho, destaca-se a proposta de pontos de acesso virtuais leves (LVAPs, do inglês *Light Virtual Access Points*), onde cada cliente associa-se com um AP virtual específico, através de um BSSID único. Para manter a associação, os LVAPs enviam periodicamente *beacons* específicos do cliente, inclusive ao longo de uma trajetória de movimentação. 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, porém simplificada, foi utilizada neste trabalho.

### 3. Estratégia de *handoff* transparente

Neste trabalho é proposta uma nova estratégia de gerenciamento de *handoff* transparente e controlado pela rede. O gerenciamento do *handoff* consiste em manter a conectividade de clientes móveis quando eles estão nos limites de cobertura de sinal de dois ou mais APs. O *handoff* ocorre quando a potência do sinal (RSSI), ou qualquer outra métrica de qualidade do enlace *Wi-Fi*, degrada até certos limiares mínimos. O processo do *handoff* é composto pelas etapas: 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* [Chandra Paul 2014]; e iii) execução, consistindo da sinalização do *handoff*, alocação de recursos nos rádios, re-estabelecimento de rotas na rede.

A estratégia ora proposta traz contribuições principalmente para o monitoramento e decisão de *handoff*, com uma combinação inédita de métricas obtidas de forma passiva na rede e um algoritmo de processamento das métricas para escolha do melhor AP

para o nó móvel. A proposta foi implementada no OpenWiMesh [Brito et al. 2014], um *framework* para orquestração de redes *mesh* sem fio com SDN, permitindo a programabilidade da rede *mesh* com base em um grafo que modela as características da rede.



**Figura 1. Estratégia de mobilidade e *handoff* transparente no OpenWiMesh**

A estratégia de mobilidade transparente aqui apresentada é ilustrada na Figura 1. Nesta figura, é possível notar a interação entre MC, roteadores *mesh* (MR) e Controlador SDN (CTL) nas diferentes fases do processo de *handoff*: na fase 1, o MC 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 MC passa a ser alcançável também pelo MR2 e tem sua qualidade de enlace monitorada por aquele roteador *mesh*; o MC continua a mover-se até que, na fase 3, a rede detecta que MR2 possui melhor qualidade para atendê-lo e executa o *handoff*. É também na fase 3 que as rotas do MC são reconfiguradas, como no exemplo ilustrado, onde a rota entre o MC e um nó correspondente (e.g. servidor de vídeo na Internet) que antes passava por MR1 e CTL agora passa por MR2 e CTL.

Todo o processo do *handoff*, mecanismos de monitoramento, detecção e execução serão detalhados nas subseções seguintes.

### 3.1. Métricas de qualidade de enlaces *Wi-Fi*

Alguns *drivers* de interfaces *Wi-Fi* do Kernel Linux suportam a API *nl80211*<sup>1</sup>, que permite obter informações específicas do meio sem fio, a partir de frames capturados pelo dispositivo *Wi-Fi*. Algumas dessas informações são:

- **Received Signal Strength Indicator (RSSI).** 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 infraestruturado).
- **802.11 Retry Counter (RTY).** 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 incrementa o contador de retransmissões. Através da API *nl80211* é possível obter estatísticas sobre a quantidade total de pacotes transmitidos, retransmissões, quantidade de pacotes que falharam (esgotou-se o limite de retransmissão), etc., para cada nó vizinho.

<sup>1</sup><https://wireless.wiki.kernel.org/en/developers/documentation/nl80211>

Diferente de trabalhos anteriores, que consideram estas métricas de forma isolada [Amir et al. 2010, Chandra Paul 2014, Tsukamoto et al. 2007], aqui propõe-se 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 permite a inferência da qualidade no enlace de *downlink*. Ambas as métricas são monitoradas de forma passiva no AP, sem sobrecarga adicional. Além disso, as taxas de transmissão e recepção de pacotes,  $TX_{pps}$  e  $RX_{pps}$  (expressas em *pacotes por segundo*), respectivamente, são aplicadas para balancear essas métricas, atribuindo pesos para o RSSI e RTY de acordo com o volume e direção do tráfego. Assim, ajusta-se a importância de cada métrica de forma dinâmica e equilibrada. Os pesos de transmissão ( $Weight_{tx}$ ) e recepção ( $Weight_{rx}$ ) são definidos como:

$$Weight_{tx} = \frac{TX_{pps}}{TX_{pps} + RX_{pps}} \quad (1) \quad Weight_{rx} = \frac{RX_{pps}}{TX_{pps} + RX_{pps}} \quad (2)$$

Para que as métricas RSSI e RTY possam ser combinadas, elas são, antes de tudo, normalizadas. A Equação 3 apresenta a normalização do valor do RSSI (medido em dBm) para porcentagem da qualidade do sinal ( $Signal_Q$ ). Já a normalização da taxa de retransmissão (FRR) leva em consideração a taxa de pacotes retransmitidos (RTY, medido em *pps*) em comparação com o total de transmitidos, conforme Equação 4.

$$Signal_Q = \frac{2 * (dBm + 100)}{100}, dBm \in [-100, -50] \quad FRR = \frac{RTY_{pps}}{TX_{pps} + RTY_{pps}} \quad (4)$$

(3)

A qualidade do enlace *Wi-Fi* é expressa na Equação 5. Nessa equação, além de considerar o RSSI e RTY, balanceados pelos valores de  $Weight_{rx}$  e  $Weight_{tx}$ , considera-se também o tempo de inatividade das estatísticas informadas pelo *nl80211*. 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.

$$Link_Q = \frac{Weight_{tx} * (1.0 - FRR) + Weight_{rx} * Signal_Q}{inactiveTime}, inactiveTime > 0 \quad (5)$$

A métrica acima produz uma visão instantânea da qualidade do sinal *Wi-Fi*. Não obstante, devido às oscilações do meio sem fio, esses valores podem variar e causar o efeito *ping-ponging* [Chandra Paul 2014]. Para evitar essas oscilações, a programação dessas funções pode fazer uso da média do RSSI no controlador, utilizar a média de RSSI disponibilizada pela API *nl80211* ou ainda aplicar uma média móvel ao valor de  $Link_Q$ . Neste trabalho utilizou-se uma média móvel de tamanho  $\sigma$ , arbitrado em 5.

### 3.2. Algoritmo de decisão de *handoff*

A métrica de qualidade do enlace previamente apresentada é calculada a cada atualização do grafo da rede, cujo intervalo padrão é um segundo. Após o cálculo da qualidade dos enlaces ( $Link_Q$ ) de todos os clientes associados a um nó, uma rotina de checagem da necessidade do *handoff* é chamada. Essa rotina verifica se o  $Link_Q$  está abaixo de determinado limiar  $T$  (*threshold*) e se existe outro vizinho com  $Link_Q$  melhor que a associação atual e com diferença  $H$  (histerese). Essa abordagem é semelhante à apresentada em [Chandra Paul 2014], chamada de *Relative Signal Strength with Threshold and Hysteresis*, porém aqui expandida considerando outras métricas (Equação 5).

O pseudo-código da Listagem 1 apresenta a lógica de verificação da qualidade do enlace e o algoritmo de decisão de *handoff* supracitado.

```

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, target_ap)

check_handoff(cur_ap, client):
    eligible_cand = []
    cur_linkq = link_quality(cur_ap, client)
    FOR cand IN client.neighbors():
        IF link_quality(cand, client) > cur_linkq + H:
            capacity = calc_capacity(cand)
            eligible_cand.append([cand, capacity])
    sorted_eligible_cand = sort(eligible_cand)
    RETURN sorted_eligible_cand.first()

```

**Listagem 1. Pseudo-código das rotinas de detecção e decisão de *handoff***

A escolha de qual AP melhor atenderá ao cliente em *handoff* deve levar em consideração a capacidade dos APs elegíveis para suportar o novo cliente. Para o cálculo da capacidade disponível em um AP, pode-se levar em consideração diferentes métricas como vazão (máxima teórica, mínima, vazão derivada), CPU ou memória disponível no AP, quantidade de clientes associados, etc. Neste trabalho optou-se por utilizar a média de banda residual entre todas as interfaces do AP *mesh*, conforme apresenta a Equação 6. Embora a utilização de outras métricas possa melhorar a estimativa de capacidade de um nó, em testes de laboratório realizados, foi possível constatar que o uso da banda residual possui fidelidade com a capacidade do nó, enquanto que seu cálculo é simplificado.

$$calc\_capacity = \frac{1}{n} \sum_{iface=1}^n \left(1 - \frac{cur\_throughput_{iface}}{max\_speed_{iface}}\right) \quad (6)$$

### 3.3. Execução do *handoff*

Ao detectar a necessidade de *handoff* e escolher o novo AP de um cliente móvel, a estratégia proposta inicia as rotinas de reconfiguração de rotas para as aplicações do cliente. Para isso, são mantidas informações sobre as aplicações e rotas ativas na rede, indexando-as nos arcos do grafo. Dessa maneira, a execução do *handoff* consiste em processar a lista de aplicações em cada arco entre o cliente e o AP atual, buscando novo caminho no grafo que contemple o novo AP e reconfigurá-las em cada MR do caminho. A reconfiguração das rotas ocorre através de mensagens Openflow do tipo *flow-mod*.

Em termos da associação *Wi-Fi*, como foi utilizado um único BSSID nos APs, não é necessária sinalização adicional ou re-associação no cliente. Para configurar os nós com essa característica basta manter-se o mesmo SSID, BSSID, canal e, caso utilize criptografia, chave da sessão. Neste trabalho foi utilizado o software AP *hostapd*<sup>2</sup> para criação da rede *Wi-Fi* infraestruturada dos clientes.

<sup>2</sup>O *hostapd* é um software que transforma uma interface de rede *Wi-Fi* convencional em um AP.

A reconfiguração de rotas ocorre de forma a primeiro criar as novas rotas, para somente então remover as rotas antigas. A 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* [Chandra 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. Essa abordagem pode gerar duplicação de pacotes na rede, porém seu custo benefício a torna interessante e viável, como será visto mais adiante.

#### 4. Avaliação da proposta

Para avaliar a estratégia proposta, uma série de experimentos e medições foram conduzidos, através de um *testbed* 802.11/OpenFlow. Neste *testbed*, um conjunto de MR se comunicam através de uma rede *mesh* orquestrada pelo OpenWiMesh e possuem uma interface *Wi-Fi* adicional para acesso dos clientes (baseada na metodologia LVAP). A fim de prover uma linha base de comparação, foi utilizada uma segunda estratégia de gerenciamento de mobilidade: APs divulgando o mesmo SSID com diferentes BSSID's; roteamento entre os MRs através do protocolo B.A.T.M.A.N. (rede *mesh*). Nesse cenário, nomeado BATMAN-MCHO (*B.A.T.M.A.N. with Mobile Controlled Handoff*), os clientes móveis controlam o *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* foi construído da seguinte maneira. Os MRs foram implantados a partir de máquinas com CPU Intel Core i5, 8GB de memória, S.O. Debian 8.3, interface *Wi-Fi* Atheros AR9485 (usada na rede *mesh*) e interface *Wi-Fi* TP-Link WN721N (usada na *hostapd*). O CTL executa em um notebook com hardware similar aos MRs, assumindo também o papel de Nó Correspondente (CN). O Cliente Móvel (MC) comunica-se com o CN durante os testes, utilizando o serviço de transferência de vídeo e serviço medições *iperf*. O MC foi baseado em três das plataformas mais utilizadas atualmente como sistema operacional em computadores pessoais: Linux, Windows e MAC OS X. Dessa forma, é possível validar o comportamento das estratégias com relação ao gerenciamento de *handoff* em diferentes plataformas.

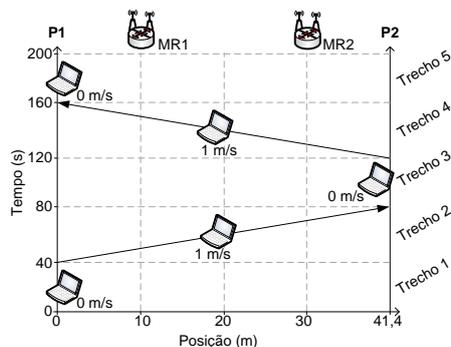
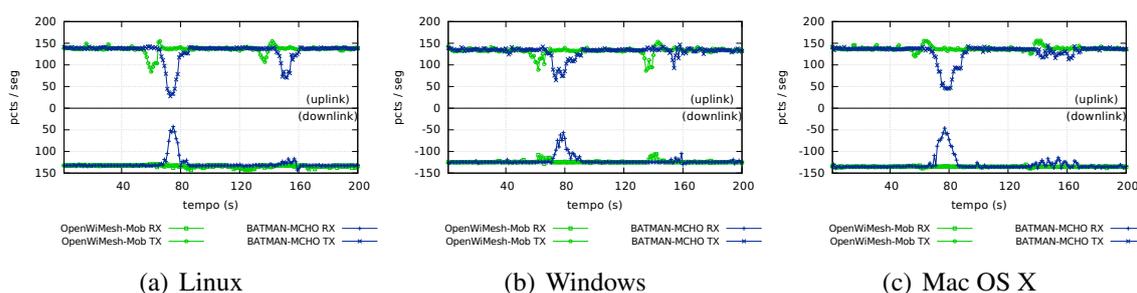


Figura 2. Diagrama de mobilidade no teste de *handoff*

O cenário utilizado é similar ao apresentado na Figura 1 e o diagrama de mobilidade é ilustrado na Figura 2. A metodologia de avaliação consistiu em testes com duração de 200 segundos, divididos da seguinte maneira: o MC começa no ponto P1 e permanece naquele ponto por 40 segundos; ele, então, move-se em direção ao ponto P2 com velocidade aproximada de 1 m/s e duração de 40 segundos; permanece no ponto P2 por mais 40 segundos; retorna a P1 com velocidade 1m/s e duração 40s; e, finalmente, permanece em P1 nos 40 segundos finais. Cada experimento foi repetido 10 vezes por plataforma.

Para medir a QoS fez-se uso de geração de tráfego baseado em taxa de bits constante, com intervalo de envio de pacotes de 8 ms e tamanho do pacote de 200 bytes, através da ferramenta *iperf*. A partir dos 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.

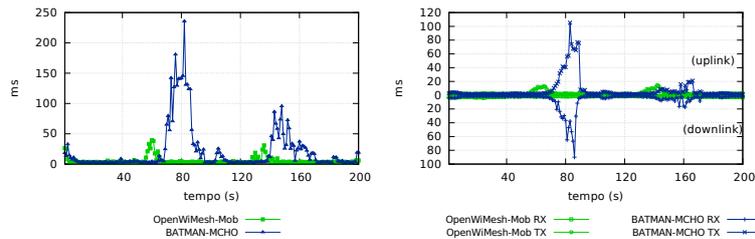
O gráfico da Figura 3 apresenta os resultados para taxa de comutação de pacotes. Variações na linha que representa a taxa de comutação podem ser entendidas como perda de pacotes e quanto mais próximas a zero pior. Pela análise dos gráficos é possível verificar que a abordagem de *handoff* BATMAN-MCHO possui impacto significativo na qualidade de serviço da rede, levando a grande perda de pacotes nos momentos de *handoff*, independente da plataforma em questão. 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, o *handoff* controlado pelo OpenWiMesh-Mob fornece melhores condições de rede para o cliente, com pequenas oscilações no tráfego, porém sem causar impactos perceptíveis ao usuário na conexão. No Mac OS X, por exemplo, a taxa de comutação de pacotes no *downlink* do OpenWiMesh-Mob apresentou média de 135.30 pps, com desvio padrão de 0.93, enquanto que o BATMAN-MCHO foi de  $129.96 \pm 15.54$  pps. Em particular, no Trecho 2, quando ocorre o primeiro *handoff*, o BATMAN-MCHO apresenta uma degradação de 14.03% em relação ao OpenWiMesh-Mob.



**Figura 3. Taxa de comutação de pacote entre MC e CN**

Nas Figuras 4a e 4b é possível verificar o atraso bidirecional e variação de atraso na rede, utilizando o mesmo modelo de tráfego anteriormente citado. Algumas aplicações são bastante sensíveis ao atraso e variação do atraso, tolerando atraso máximo de 100ms e variação de 20ms a fim de garantir alta qualidade e interatividade [ITU-T 2003, Amir et al. 2010]. Pela Figura 4a e 4b é possível observar que o OpenWiMesh-Mob esteve abaixo desses limites nas métricas de atraso e variação de atraso. Em relação ao atraso, o OpenWiMesh-Mob supera a estratégia BATMAN-MCHO

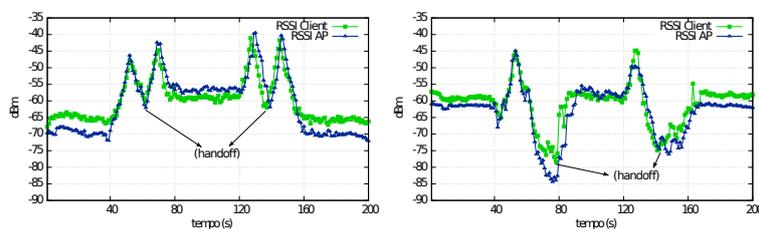
em cerca de 300% durante as etapas de *handoff*. Já com relação à variação de atraso, o OpenWiMesh-Mob teve um *jitter* de *downlink* e *uplink* na ordem de  $1.76 \pm 0.57$  e  $3.03 \pm 2.65ms$  (média e desvio padrão), com máximo de 3.77 e 14.57ms, respectivamente. Ao passo que o BATMAN-MCHO apresentou *jitter* de *downlink* e *uplink* de  $5.55 \pm 11.44$  e  $7.47 \pm 15.65ms$ , com máximo de 89.97 e 105.32ms, respectivamente. Devido a limitação de espaço e semelhança dos dados, apenas as métricas do Mac OS X foram apresentadas.



**Figura 4. Avaliação de QoS: (a) atraso bidirecional; (b) jitter**

A métrica de potência de sinal *Wi-Fi* recebido (RSSI), ilustrada através da Figura 5, possui relação indireta com a qualidade de serviço da rede, impactando na taxa de erros, atraso, jitter e probabilidade de recebimento de pacotes. 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 -67.8 dBm, sobretudo se comparado com a abordagem BATMAN-MCHO cujos valores de RSSI no AP e no cliente chegaram a -84.4 e -78.3 dBm, faixas de valores de menor qualidade do enlace.

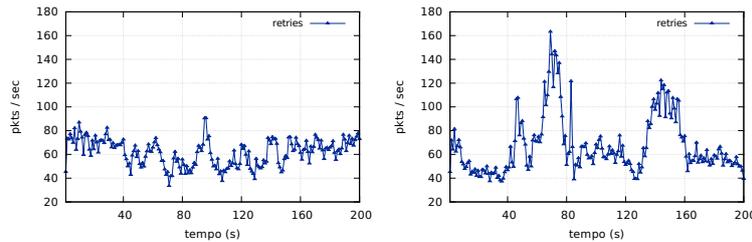
Pela Figura 5a pode-se notar que no Trecho 2 o RSSI melhora linearmente à medida que o cliente se aproxima do MR1, depois degrada notadamente quando se distancia de MR1 e acontece o *handoff* para MR2, então ele experimenta nova melhoria ao aproximar-se de MR2, e volta a cair até estabilizar no afastamento de MR2 e parada no ponto 2. Em contrapartida, a curva de RSSI do BATMAN-MCHO (Figura 5b) mostra que o sinal *Wi-Fi* se degrada à 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 em ambas as estratégias.



**Figura 5. Potência de Sinal Recebido (RSSI): esquerda, (a) OpenWiMesh-Mob; esquerda, (b) BATMAN-MCHO**

Outra métrica que indiretamente impacta na qualidade da rede é a taxa de retransmissão de pacotes da camada MAC do 802.11. Na Figura 6 é possível verificar as retransmissões de pacotes, na perspectiva do cliente e do AP, ambos com o mesmo modelo de tráfego usado até aqui. É possível notar que, independente do trecho e da existência de

*handoff*, há uma taxa considerável de retransmissão de pacotes, possivelmente ocasionada pelas interferências externas, uso intenso da rede no *uplink* e *downlink*, dentre outros fatores. No entanto, o OpenWiMesh-Mob consegue manter esse valor dentro de uma faixa aceitável mesmo durante a execução dos *handoff*'s nos Trechos 2 e 4. 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*, contabilizando média de 88.56 pps de retransmissões contra 59.53 pps do OpenWiMesh-Mob.



**Figura 6. Taxa de retransmissão de pacotes: esquerda, (a) OpenWiMesh-Mob; esquerda, (b) BATMAN-MCHO**

Para medir a QoE do usuário na utilização da rede, foi realizado um experimento com uma aplicação de *streaming* de vídeo, onde o cliente móvel assistia a um filme enquanto se movia pela rede. Para fazer o *streaming* de vídeo utilizou-se a ferramenta *live555*<sup>3</sup>, transmitindo a animação Big Buck Bunny na resolução de 240p (320x240), formato MPEG4-TS, com duração de 200s e cadência de 15fps. Esse experimento também foi repetido 10 vezes no mesmo cenário anterior. O vídeo transmitido pela rede foi comparado com o original através de medições com PSNR (*Peak Signal to Noise Ratio*) e SSIM (*Structural Similarity Index*), métricas comumente utilizadas nesse tipo de estudo [Lambrecht and Verscheure 1996, Engelke et al. 2009]. Para comparação, foi utilizado a ferramenta MSU VQMT<sup>4</sup>. A partir do resultado dessa comparação é possível mapeá-la em um valor de MOS (*Mean Opinion Score*) [Lambrecht and Verscheure 1996, Engelke et al. 2009], conforme Tabela 1, e assim avaliar a QoE dos usuários.

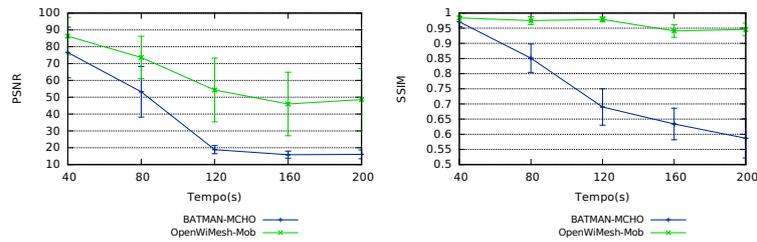
MOS	(5) Excelente	(4) Bom	(3) Aceitável	(2) Pobre	(1) Ruim
PSNR (dB)	> 37	31 – 37	25 – 31	20 – 25	< 20
SSIM	> 0.90	0.77 – 0.89	0.61 – 0.76	0.38 – 0.60	< 0.38

**Tabela 1. Mapeamento PSNR/MOS e SSIM/MOS**

Os resultados da avaliação de QoE podem ser vistos na Figura 7. No gráfico do PSNR é possível visualizar claramente a queda de qualidade do vídeo após o primeiro *handoff*, embora o OpenWiMesh-Mob consiga 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 70 segundos), quando o nível de sinal *Wi-Fi* e qualidade da rede já estão bastante degradados, resultando, portanto, em uma média de 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

<sup>3</sup>Website: <http://www.live555.com/>, Último acesso 20/12/2016

<sup>4</sup>Website: <http://goo.gl/Vio9qg>, Último acesso 20/12/2016



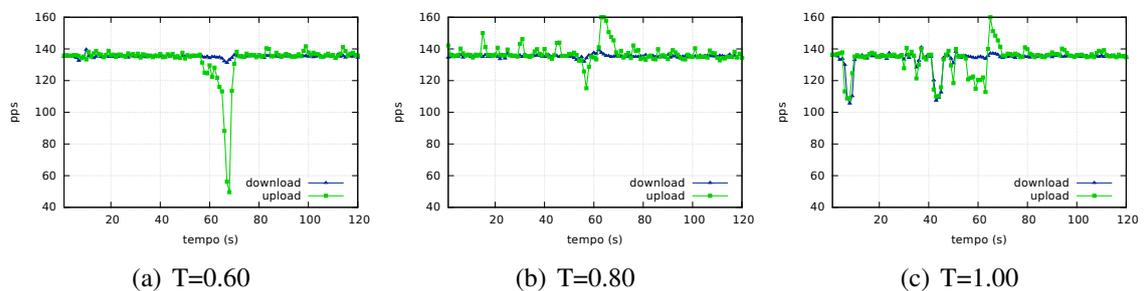
**Figura 7. Qualidade de Vídeo com valores agregados: esquerda, (a) PSNR; direita, (b) SSIM**

devido aos fatores de propagação de erros em frames interdependentes de vídeos MPEG [Feamster and Balakrishnan 2002]. Já a métrica SSIM revela que o OpenWiMesh-Mob consegue manter o vídeo em 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.

É possível notar que a estratégia de *handoff* proposta apresenta desempenho médio 70% superior na métrica PSNR e 29% superior na métrica SSIM se comparado com o BATMAN-MCHO.

Realizou-se ainda um estudo comparativo acerca de diferentes parametrizações e configurações do 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. Devido a limitações de espaço, serão apresentados apenas a avaliação do limiar de *handoff*  $T$ , histerese de *handoff*  $H$  e padrão de mobilidade.

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 anteriormente, porém com duração de 120 segundos e 5 repetições. 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.

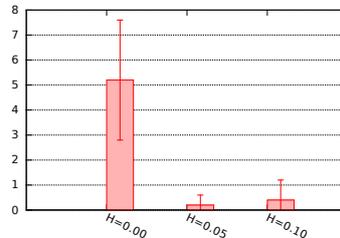


(a)  $T=0.60$  (b)  $T=0.80$  (c)  $T=1.00$

**Figura 8. Taxa de comutação de pacotes com diferentes valores de  $T$**

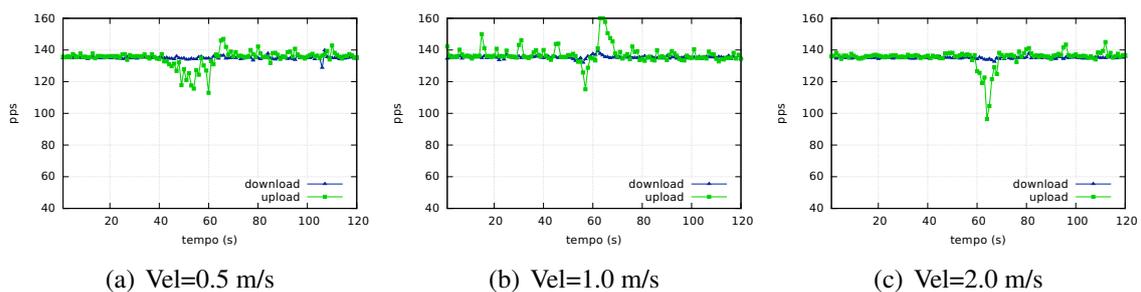
Na Figura 8a é possível observar a taxa de comutação com  $T = 0.60$ , onde o *handoff* ocorre em momento ligeiramente tardio, por volta de 65 a 70 segundos e, portanto, a taxa de comutação degrada acima de 50%. Já na Figura 8c, cujo valor de  $T$  foi configurado para 1.00, percebe-se diversas oscilações ao longo do teste, em geral ocasionadas por decisões de *handoff* precipitadas. Por fim, a Figura 8b apresenta uma boa relação

de compromisso entre o limiar de *handoff* e a qualidade da rede. 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.



**Figura 9. Quantidade de *handoffs* com diferentes valores de H**

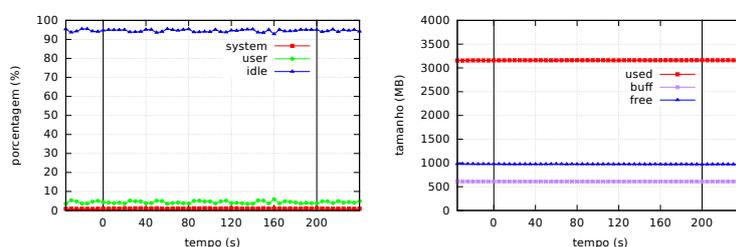
Para avaliar a relação do valor da histerese na estratégia de *handoff* proposta, foram considerados três valores de H:  $H = 0.10$ ,  $H = 0.05$  e  $H = 0.00$ . A metodologia de avaliação para o parâmetro H foi ligeiramente diferente das demais: o MC foi mantido estacionário entre os dois APs e mediu-se a quantidade de *handoffs* que a estratégia executou. 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 9 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*. 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  $H = 0.10$ . Nesse caso, o valor de  $H = 0.05$  mostra-se mais vantajoso pois mantém a estabilidade da rede ao passo que oportuniza a detecção de *handoffs* válidos.



**Figura 10. Taxa de comutação de pacotes com diferentes velocidades do MC**

Outro aspecto que merece atenção é a relação entre o padrão de mobilidade e os resultados ora apresentados. Para medir o impacto do padrão de mobilidade no processo de *handoff*, mensurou-se as métricas de QoS com MC movendo-se a  $0.5m/s$ ,  $1.0m/s$  e  $2.0m/s$ . Os testes foram realizados no mesmo cenário anteriormente citado. O gráfico da Figura 10 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, observa-se pouca relação entre o padrão de mobilidade e a estratégia de *handoff* nesse cenário.

Por fim, é crucial avaliar também o desempenho do controlador SDN, componente importante na estratégia de *handoff*. Neste trabalho foi utilizado apenas um controlador e, a fim de verificar o impacto do seu funcionamento na solução proposta, foram analisados o consumo de CPU e de memória RAM. Os testes foram realizados no mesmo cenário anteriormente citado, porém capturando dados do controlador durante 40 segundos antes do teste e 40 segundos depois. No gráfico da Figura 11a 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 11b ilustra a utilização de memória no controlador, cujo consumo também manteve-se constante ao longo da execução do experimento, mesmo antes da sua inicialização e após sua conclusão.



**Figura 11. Desempenho do controlador SDN no OpenWiMesh-Mob: esquerda, (a) Consumo de CPU; direita, (b) Consumo de memória**

## 5. Conclusões e Trabalhos Futuros

Esse 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 SDN em uma rede *mesh* sem fio<sup>5</sup>. A avaliação comparativa aponta melhoras substanciais na qualidade da rede quando do uso da estratégia proposta. O OpenWiMesh-Mob é agnóstico a plataforma de S.O. e não requer qualquer configuração especial no cliente, viabilizando e melhorando o gerenciamento de mobilidade de dispositivos sem fio, requisito sobremaneira importante nas redes modernas.

As métricas relacionadas com a QoS da rede apontam benefícios nas estratégias aqui propostas: até 14.03% de melhoria em relação à média de taxa de comutação; redução de até 200ms no atraso birecional durante o *handoff*; variação de atraso sempre inferior a 20ms; melhor aproveitamento do sinal *Wi-Fi*; e menor sobrecarga na rede com redução na quantidade de retransmissões do 802.11. A qualidade de experiência foi avaliada em termos da qualidade de vídeo. 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.

Em trabalhos futuros pretende-se i) ampliar o *testbed* 802.11/OpenFlow na universidade e ii) investigar o funcionamento do *hostapd* para permitir pré-configuração dos parâmetros de associação *Wi-Fi*, evitando perdas no *uplink* durante o *handoff*.

<sup>5</sup>O código-fonte, scripts de configuração e dados da avaliação estão disponíveis (<http://grade.dcc.ufba.br/OpenWiMesh>, último acesso 20/12/2016) para análise e comparativos futuros, podendo ser usados ainda como base para desenvolvimento de novos algoritmos e estudos comparativos.

## Referências

- Amir, Y., Danilov, C., Musuãloiu-Elefteri, R., and Rivera, N. (2010). The SMesh Wireless Mesh Network. *ACM Transactions on Computer Systems (TOCS)*, 28(3):6:1–6:49.
- Avelar, E. A. M. (2013). Pmipflow: Uma proposta para gerenciamento de mobilidade em redes definidas por software. Master's thesis, Universidade Federal de Pernambuco.
- Brito, I., Gramacho, S., Ferreira, I., Nazaré, M., Sampaio, L., and Figueiredo, G. (2014). Openwimesh: um framework para redes mesh sem fio definidas por software. In *32th Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, pages 413–426.
- Chandra Paul, L. (2014). Handoff/handover mechanism for mobility improvement in wireless communication. *Global Journal of Researches In Engineering*, 13(16).
- Croitoru, A., Niculescu, D., and Raiciu, C. (2015). Towards wifi mobility without fast handover. In *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*, pages 219–234.
- Dely, P., Kassler, A., and Bayer, N. (2011). Openflow for wireless mesh networks. In *20th International Conference on Computer Communications and Networks (ICCCN)*, pages 1–6. IEEE.
- Engelke, U., Kusuma, M., Zepernick, H.-J., and Caldera, M. (2009). Reduced-reference metric design for objective perceptual quality assessment in wireless imaging. *Signal Processing: Image Communication*, 24(7):525–547.
- Feamster, N. and Balakrishnan, H. (2002). Packet loss recovery for streaming video. In *12th International Packet Video Workshop*, pages 9–16. PA: Pittsburgh.
- ITU-T (2003). One-way transmission time. *Recommendation G*, 114. Último acesso em 20 de Abril de 2016.
- Johnson, D., Ntlatlapa, N., and Aichele, C. (2008). Simple pragmatic approach to mesh routing using batman. *2nd IFIP International Symposium on Wireless Communications and Information Technology in Developing Countries, CSIR*, pages 62 – 68.
- Lambrecht, C. J. v. d. B. and Verscheure, O. (1996). Perceptual quality measure using a spatiotemporal model of the human visual system. In *Electronic Imaging: Science & Technology*, pages 450–461.
- Lee, S., Cho, C., Hong, E.-k., and Yoon, B. (2016). Forecasting mobile broadband traffic: Application of scenario analysis and delphi method. *Expert Systems with Applications*, 44:126–137.
- Schulz-Zander, J., Suresh, L., Sarrar, N., Feldmann, A., Hühn, T., and Merz, R. (2014). Programmatic orchestration of wifi networks. In *2014 USENIX Annual Technical Conference (USENIX ATC 14)*, pages 347–358. USENIX Association.
- Tsukamoto, K., Yamaguchi, T., and Kashihara, S. (2007). Experimental evaluation of decision criteria for wlan handover: Signal strength and frame retransmission. *IEICE transactions on communications*, 90(12):3579–3590.