

PMIPFlow: Uma Proposta para Gerenciamento de Mobilidade em Redes Definidas por Software

Edson Adriano Maravalho Avelar, Lorena Marques Avelar, Kelvin Lopes Dias

Centro de Informática – Universidade Federal de Pernambuco (UFPE)
Caixa Postal 7851 – Cep 50740-540 – Recife – PE – Brasil

{eama, l1m, kld}@cin.ufpe.br

Abstract. *With the growth of open source initiatives and software-defined networks, the computer networks are leaving from a scenario where the services and infrastructure ecosystem was closed and proprietary and now is open and free. However, this advance seems not to be true for wireless networks, which remain closed and proprietary. In this context, this paper proposes a mobility management in software-defined network. The proposal is implemented on a testbed OpenFlow / IEEE 802.11 and the evaluation is done using metrics QoE (Quality of Experience).*

Resumo. *Com o crescimento de iniciativas “open source” e redes definidas por software, as redes de computadores estão saindo de um cenário onde o ecossistema de serviços e infraestrutura era fechado e proprietário e passa a ser aberto e livre. No entanto, esse avanço parece não ser realidade para redes sem fio, as quais ainda permanecem fechadas e proprietárias. Nesse contexto, este trabalho apresenta uma proposta de gerenciamento de mobilidade em redes definidas por software. A proposta é implementada em um testbed Openflow/IEEE 802.11(Wi-Fi) e a avaliação é conduzida utilizando tanto métricas de QoS(Qualidade de serviço) quanto de QoE(Qualidade de Experiência).*

1. Introdução

Com o crescimento de iniciativas *open source*, estamos saindo de um cenário onde o ecossistema de serviços e infraestrutura era fechado e proprietário e passa a ser aberto e livre [Yap et al. 2010b]. Entretanto, no contexto de infraestrutura e equipamentos de redes sem fio, as soluções, em geral, permanecem fechadas e proprietárias, tanto em função das estratégias de mercado dos principais fabricantes, quanto pela manutenção da operação ininterrupta das redes de produção e ISPs (*Internet Service Providers*) que são relutantes a modificações.

Dessa forma, novos requisitos são incorporados à pilha TCP/IP (*Transmission Control Protocol/Internet Protocol*) por meio de remendos. Isto cria um impasse, comumente chamado de *engessamento* da Internet e torna as redes complexas, de difícil gerenciamento, não extensíveis e fechada para inovação [Chowdhury and Boutaba 2009]. Neste contexto, as redes definidas por software, SDN (*Software-Defined Networks*) [McKeown et al. 2008], surgem como a tecnologia chave para promover a inovação nas redes tradicionais.

A ideia do conceito SDN é separar claramente o plano de dados do plano de controle. Desta forma, é possível usufruir de várias vantagens, como dispor de um controle maior da rede, de monitoramento eficiente e de um controlador com gerenciamento local ou remoto de todos os elementos da rede. Esse controle é realizado através de aplicações, da mesma forma como é feito com softwares para *desktop*, daí a denominação conferida a este novo conceito. Outra tecnologia importante neste processo é a virtualização.

A virtualização é uma estratégia para resolver muitos problemas e criar novos serviços. Em *hosts* individuais, a virtualização permite que um único computador realize o trabalho de muitos outros através do compartilhamento de seus recursos. No caso da virtualização de redes, recursos podem ser divididos em fatias, considerando cada roteador como um elemento que pode ser virtualizado. Um roteador pode executar múltiplas instâncias e, desse modo, o substrato físico pode executar múltiplas topologias virtuais [Sherwood et al. 2009].

A virtualização de redes pode ser alcançada através de várias formas, utilizando máquinas virtuais como roteadores ou por meio de redes programáveis, como é feito com o OpenFlow, que é um padrão aberto que permite a criação de redes virtuais utilizando somente recursos de camada 2 (Camada de Enlace). Neste caso, as redes virtuais são normalmente implementadas por comutadores (*switches*) *Ethernet*, com tabelas de encaminhamento internas e uma interface padrão para adicionar e remover entradas de fluxos. Assim, torna-se viável testes de novos protocolos em larga escala, usando os recursos das próprias redes de produção [McKeown et al. 2008].

Com a proliferação de *smartphones*, *tablets* e uma infinidade de dispositivos/sensores conectados à Internet, há necessidade de um gerenciamento de mobilidade eficiente e escalável, a fim de garantir tanto a obtenção das informações coletadas por objetos/sensores móveis conectados às redes sem fio, quanto para garantir a continuidade do serviço aos usuários móveis. Neste contexto, a filosofia SDN oferece uma solução ideal, no entanto, a maioria das proposta em inovação usando SDN foca em redes cabeadas.

Este trabalho propõe a arquitetura PMIPFlow, que oferece uma solução para o gerenciamento de mobilidade em redes definidas por software. A proposta foi inspirada nos protocolos NetLMM (*Network-Based Localized Mobility Management*) e PMIPv6 (*Proxy Mobile IPv6*) [Gundavelli et al. 2008a], que liberam o dispositivo móvel das funções ligadas à mobilidade e delegam à rede a tarefa de mudar o ponto de acesso (*handover*) do dispositivo decorrente de sua movimentação e/ou devido à busca por melhores canais.

A proposta é implementada em um *testbed* OpenFlow/802.11(Wi-Fi). Além disso, a fim de proporcionar mobilidade transparente (*seamless*), foi desenvolvido um mecanismo de decisão de *handover* baseado na Lógica Fuzzy [Klir and Yuan 1995]. Os resultados de desempenho demonstram a eficácia da proposta no fornecimento de mobilidade transparente/continuidade de serviço ao usuário, bem como o suporte adequado para os requisitos de tráfego de vídeo em termos de métricas de QoS (*Quality of Service*) [Firoiu et al. 2002] e QoE (*Quality of Experience*) [Winkler and Mohandas 2008].

Este artigo está organizado da seguinte forma. Na Seção 2, a arquitetura da proposta PMIPFlow será detalhada. Em seguida, na Seção 3, será apresentada a avaliação da proposta. Na Seção 4, os trabalhos relacionados serão discutidos e a Seção 5 mostra as considerações finais deste trabalho.

2. Arquitetura PMIPFlow

O PMIPFlow está dividido em duas partes principais: O OMAG (*OpenFlow Mobility Access Gateway*) e o componente PMIPFlow-LMA (*PMIPFlow - Localized Mobility Anchor*). Este último é uma aplicação executada no controlador OpenFlow (LMA-NOX). Nesta arquitetura, cada célula, servida por um OMAG, forma o chamado domínio PMIPFlow. A Figura 1 mostra o cenário, destacando as entidades da arquitetura PMIPFlow.

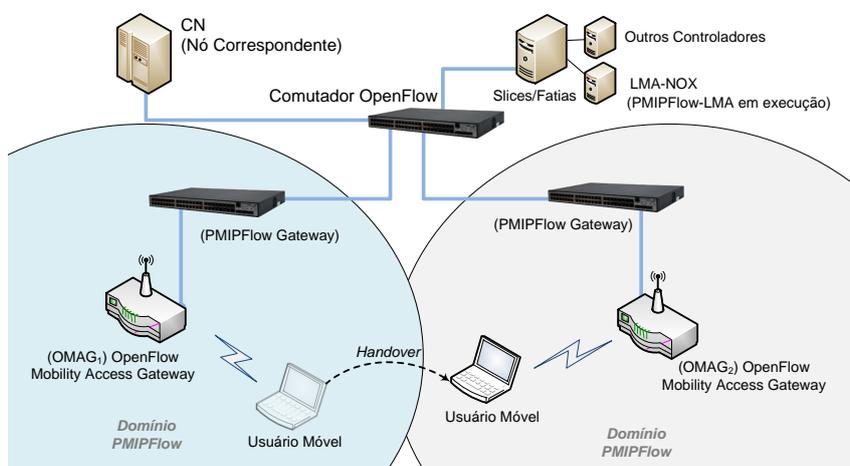


Figura 1. Arquitetura PMIPFlow

O OMAG é um agente de mobilidade, que reside na borda da rede OpenFlow monitorando os movimentos dos usuário sob o seu controle. Se algum usuário sair de sua cobertura, ele deve acionar o PMIPFlow-LMA informando o acontecimento deste evento. O OMAG também pode iniciar o procedimento de *handover* para um usuário que esteja na iminência de sair da área de cobertura da rede.

O LMA-NOX é um controlador especial no qual a aplicação PMIPFlow-LMA é executada. O PMIPFlow-LMA possui funções como: criar regras nos *switches* OpenFlow para atender às necessidades de mobilidade dos clientes do OMAG, receber e tratar mensagens PBU (*Proxy Binding Update*), enviar mensagens PBA (*Proxy Binding Ack*) ao nó móvel e servir como servidor de acesso barrando usuários não autorizados.

O PMIPFlow-LMA foi desenvolvido a partir do conceito de LMA no PMIPv6, porém com modificações para transformá-lo em um componente de controlador. Ao invés de criar túneis IP, como o LMA no PMIP, o PMIPFlow-LMA recebe o PBU, cria regras na tabela de encaminhamento do *switch* OpenFlow e envia o PBA de volta para os OMAGs.

Além do OMAG e do PMIPFlow-LMA, é possível verificar que existem *gateways* PMIPFlow na arquitetura representada na Figura 1. Estes *gateways* servem como ponto de

saída OpenFlow do domínio PMIPFlow. A proposta pode ser estendida tanto em termos de adição de novos usuário como em termos de expansão do núcleo da rede, pois se o núcleo da rede crescer, com a inclusão de elementos OpenFlow, a operação dos OMAGs e do PMIPFlow-LMA não serão afetadas por estarem na borda da rede.

O PMIPFlow pode ser executado concorrentemente com outras aplicações do controlador sem interferir na operação da rede. Pois, é possível criar fatias (*slices*) especializadas para clientes móveis e separar as fatias de clientes móveis das fatias de clientes fixos.

O comutador (*switch*) OpenFlow, mostrado na Figura 1, forma o núcleo da rede SDN do PMIPFlow. Nele são instaladas regras de encaminhamento, criadas fatias de redes, priorização de tráfego, enfim, todas as operações SDN. Ele é o elemento que o PMIPFlow-LMA irá controlar durante as operações do PMIPFlow.

2.1. Sinalização de *Handover*

O PMIPFlow possui algumas sinalizações necessárias para tratar o *handover* do usuário e manter ativa sua conexão com a rede. A Figura 2 detalha as trocas de mensagens durante esse processo.

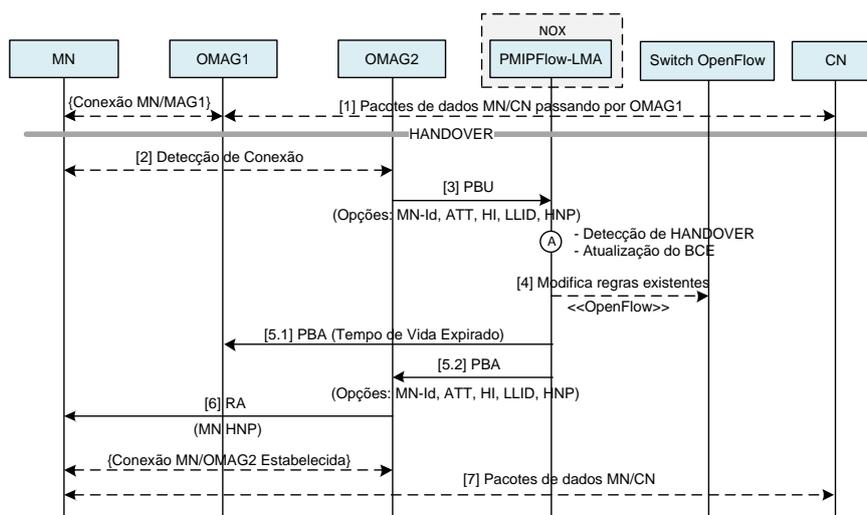


Figura 2. Sinalização de *Handover* dentro do Domínio PMIPFlow

A seguir a explicação de cada mensagem e evento mostrando na Figura 2.

- [1]: No cenário de *handover*, inicialmente o usuário está se comunicando com o CN conectado a um OMAG.
- [2]: Ao entrar em uma nova rede, o OMAG2 detectará a presença do MN através de mensagens da camada 2 (L2). Esta detecção é transparente ao usuário e não necessita de nenhuma intervenção do mesmo.
- [3]: Após detectar o novo usuário na rede, o OMAG2 envia um PBU para o PMIPFlow-LMA.
- [Evento A]: Ao receber um PBU, o PMIPFlow-LMA identificará que se trata de um *handover*, pois o endereço de origem deste PBU é diferente do endereço de origem do antigo, porém o MNid é o mesmo. O *handover* desencadeia duas

atualizações importantes: a primeira é a atualização do BC (*Binding Cache*) dentro do PMIPFlow-LMA e a segunda é na tabela de encaminhamento do *switch*.

- [4]: Após verificar a ocorrência do *handover*, além de atualizar o BC, o PMIPFlow-LMA também faz uma modificação nas regras existentes para aquele MNid dentro do *switch* OpenFlow.
- [5.1]: O PMIPFlow-LMA envia uma mensagem PBA com tempo de vida expirado para o OMAG1, este por sua vez, inicia o processo de remoção das informações do MN, para liberação de recursos.
- [5.2]: Após as atualizações necessárias, o PMIPFlow-LMA envia um PBA ao novo OMAG, com as informações necessárias para a configuração de túneis internos e endereços.
- [6]: Para completar a operação, o OMAG2 envia um RA (*Router Advertisement*) ao MN, para que este configure o endereço IPv6 utilizando o HNP (*Home Network Prefix*) contido no RA.
- [7]: Neste momento, tudo está configurado e o usuário poderá continuar sua comunicação com o CN. Vale lembrar que para maior satisfação do usuário, ou seja, para manter a qualidade de serviço em um nível alto, todo esse processo de *handover* descrito precisa ser executado o mais rápido possível, objetivando a mínima ou ausência de percepção do usuário sobre o processo.

2.2. Proposta de Antecipação de Handover

As RFCs de protocolos de gerenciamento de mobilidade não padronizam o gatilho de início de *handover*. A inteligência por trás do disparo desse gatilho fica por conta dos serviços de rede do sistema operacional ou de alguma proposta a ser implementada junto com a arquitetura. Reduzir o tempo do *handover* é crucial em serviços multimídia de tempo real, pois quebras de conexão e perdas de pacotes podem gerar degradações na qualidade do serviço.

Dessa forma, além da contribuição da arquitetura PMIPFlow, este trabalho propõe uma solução baseada em lógica *fuzzy* para antecipar o mecanismo de *handover* da arquitetura PMIPFlow. A proposta reduz de forma proativa o atraso desse processo, fazendo com que ele seja realizado de forma transparente ao usuário e ao serviço corrente, minimizando as perdas de pacotes devido possíveis quebras de conexão. Nessa e nas demais seções, “*PMIPFlow(Fuzzy)*” fará referência à arquitetura PMIPFlow com a proposta de antecipação Fuzzy.

Diferentemente da lógica tradicional, que utiliza valores exatos, a lógica *fuzzy* é um sistema capaz de trabalhar com informações imprecisas e transformá-las em uma linguagem matemática de fácil implementação computacional.

A Figura 3 apresenta o esquema da proposta PMIPFlow(Fuzzy). Nesta Figura, o Sistema de Monitoramento é o módulo de *software* responsável por executar o monitoramento das variáveis utilizadas como entrada para o sistema *fuzzy*. Essas variáveis fornecem dados numéricos que irão alimentar o sistema. Através desses valores, o sistema irá se basear para tomar as decisões. As entradas utilizadas para a proposta PMIPFlow(Fuzzy) são: Vazão, RTT (*Round Trip Time*) e RSS (*Received Signal Strength* ou Intensidade do Sinal Recebido).

O Fuzzificador é responsável por mapear os dados de entrada para o conjunto *fuzzy*

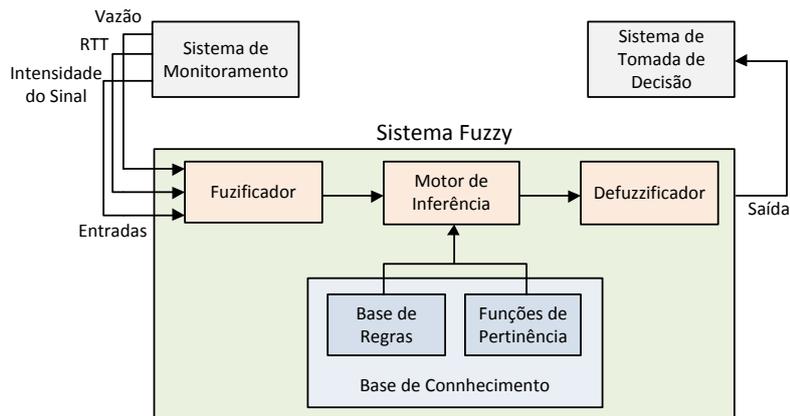
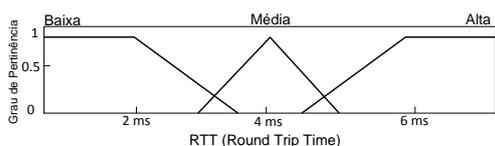


Figura 3. Representação do Sistema PMIPFlow(Fuzzy)

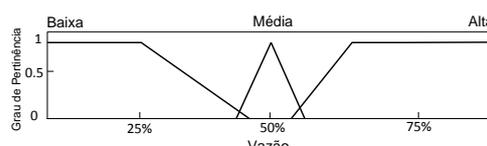
e Defuzzificador recebe as variáveis processadas pelo motor de inferência e as convertem em números para saída do sistema.

A Base de conhecimento é formada por uma Base de Regras e de Dados (Funções de Pertinência). Ambas são usadas pelo Motor de Inferência para computar uma *saída*. A saída do sistema *fuzzy* é usada como entrada para o Sistema de Decisão, esse sistema recebe os valores de saída para decidir se o usuário deve fazer *handover* ou não. Se o sistema decidir pelo *handover*, o processo de *handover* é imediatamente iniciado.

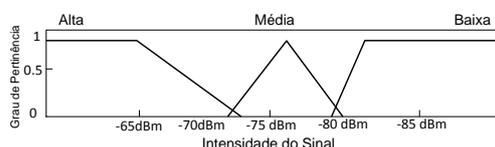
É empregado nesta proposta um sistema *fuzzy* do tipo Mamdani cuja intensidade de disparo de uma regra de inferência é uma média aritmética dos graus de pertinência obtidos. Cada variável de entrada possui três conjuntos *fuzzy* trapezoidal definidos sobre ela. A Figura 4(a) mostra o grau de pertinência para a variável RTT, todos os valores mínimos e máximos do conjunto foram definidos por meio de métodos empíricos.



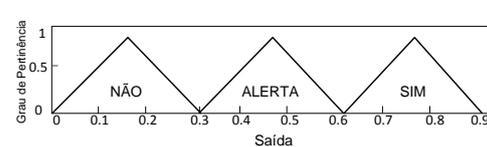
(a) Grau de pertinência para a entrada RTT



(b) Grau de pertinência para a entrada Vazão



(c) Grau de pertinência para a entrada RSS



(d) Saída do Sistema Fuzzy

Figura 4. Conjuntos e saída do sistema fuzzy

A Figura 4(b) mostra o grau de pertinência para a Vazão, os conjuntos estão definidos por meio de porcentagem em relação à largura de banda da interface de rede. Definiu-se que a partir de aproximadamente 60% da capacidade total da interface é considerada uma vazão alta, em 50% uma vazão média e abaixo de 40% uma vazão baixa.

O RSS indica que o terminal pode estar perto ou longe do ponto de acesso e,

quanto mais perto, teoricamente, melhor é a conexão. Porém, quando se trata de serviços multimídia, o RSS não é o suficiente para determinar a qualidade da aplicação. Por este motivo, foram realizados experimentos com o objetivo de determinar os valores correspondentes de PSNR, de acordo com o RSS. Assim, considerando um ambiente específico, construiu-se a tabela que avalia o PSNR em relação à distância, mapeando níveis de sinais em valores de PSNR para dar suporte à tomada de decisão de handover. Esse mapeamento é posteriormente utilizado para construção do grau de pertinência para o RSS.

Para criar o mapeamento “Nível de Sinal” versus “PSNR” realizou-se o seguinte experimento. Enviou-se o arquivo de vídeo, no sentido *downlink*, do PMIPFlow-LMA para o MN. Essa transmissão foi realizada variando-se a distância do MN em relação ao OMAG. Cada variação na transmissão foi executada 20 vezes. Ao final das transmissões, o arquivo de vídeo foi avaliado em termos de PSNR para a montagem da tabela PSNR/Distância. Em paralelo, o nível de sinal foi capturado durante toda a transmissão.

A Tabela 1 mostra a média dos valores coletados, considerando um intervalo de confiança de 95%. Vale ressaltar que os resultados destes experimentos refletem o ambiente e a configuração dos dispositivos utilizados, outros equipamentos ou outros ambientes podem fornecer resultados distintos dos apresentados na Tabela 1.

Tabela 1. Resultados do Experimento PSNR/Distância

Distância OMAG/MN (m)	< 1	5	10	15	20
Média do Nível do Sinal (dbm)	-66.76	-73.76	-78.61	-79.98	-88.20
PSNR(dB)	32.77	32.77	31.60	31.27	27.22
Perda (%)	0	0	0.05	2.45	8.35

A partir dos resultados obtidos e da análise dos vídeos no receptor, percebe-se, segundo as Tabelas 1 e 2, que, para um nível de sinal de -79,98 dbm (15 Metros), o vídeo é avaliado como BOM (31.27 dB), porém com um nível de sinal em -88,20 dbm (20 metros) o vídeo recebido torna-se ACEITÁVEL.

Tabela 2. Valores de Classificação do PSNR

PSNR (dB)	> 37	31 – 37	25 – 30	20 – 25	< 20
Qualidade	Excelente	Bom	Aceitável	Pobre	Péssimo

A partir das informações obtidas no mapeamento PSNR/Distância construiu-se o conjunto *fuzzy* para a variável de entrada RSS, a Figura 4(c) mostra o grau de pertinência para este conjunto. Para manter a qualidade do vídeo em um nível "BOM", segundo a Tabela 2, é necessário que o PSNR esteja em torno de 31dB e portanto, segundo o mapeamento, o RSS do dispositivo móvel precisa estar abaixo de "-80dBm".

A Figura 4(d) mostra o grau de pertinência para a saída, o qual varia entre 0 e 1. A partir desse resultado, o sistema de decisão da arquitetura PMIPFlow (*Fuzzy*) não fará nada, se a saída for menor que 0.3. Entrará em estado de alerta se a saída for entre 0.3 e 0.6. Ou realizará o *handover* de forma antecipado se a saída for acima de 0.6.

O tempo de monitoramento foi definido para cada 10 segundos de execução e no modo ALERTA esse tempo cai para 1 segundo. O usuário pode modificar alguns

parâmetros. Entretanto, os valores escolhidos como padrão (10s e 1s, monitoramento e alerta, respectivamente) se mostraram bastante adequados nos testes empíricos. Na Seção 3, a proposta PMIPFlow(*Fuzzy*) foi avaliada e os resultados mostraram sua eficácia.

3. Avaliação

Não é do conhecimento dos autores deste trabalho a existência, na literatura, de arquitetura similar à proposta e que possua código fonte aberto. Com isso, o PMIPFlow (*Fuzzy*) foi comparado com o PMIPFlow sem a proposta de antecipação.

A proposta foi avaliada em um *testbed* OpenFlow/IEEE 802.11(*Wi-Fi*), utilizando uma infraestrutura com equipamentos físicos e sistemas virtualizados. O OMAGs foram instalados em PCs com linux, portando placa de rede sem fio com chipset Atheros, que permite sua modificação para o modo *Master*. Isto possibilita que elas operem como um ponto de acesso. Os *gateways* PMIPFlow, o servidor de vídeo, o FlowVisor e o LMA-NOX foram implementados através de máquinas virtuais.

Os testes foram conduzidos em um laboratório no Centro de Informática da Universidade Federal de Pernambuco. Para avaliarmos desconexões de rede dentro do cenário sem fio, foi necessário mover-se pelo ambiente com velocidade de aproximadamente um metro por segundo (equivalente à uma pessoa caminhando) para forçar a desconexão de uma rede e reconexão em outra.

A avaliação de desempenho de um sistema de rede é realizada de duas formas: através de análise de métricas de QoS retiradas da rede (perdas de pacotes, vazão e *jitter*) e através de métricas de QoE (PSNR (*Peak Signal to Noise Ratio*) e SSIM (*Structural Similarity Index*)), que refletem a percepção do usuário sobre o serviço oferecido.

3.1. Avaliação de QoS

A métrica *jitter* é bastante sensível às perdas de pacotes e, portanto, bastante utilizada para avaliação de *handovers*. Para computar o *jitter*, foi enviado um fluxo de 20 Mbps pela rede, no sentido *downlink* (Servidor para Cliente), durante 60 segundos. Este teste foi repetido 10 vezes para dar mais credibilidade aos resultados.

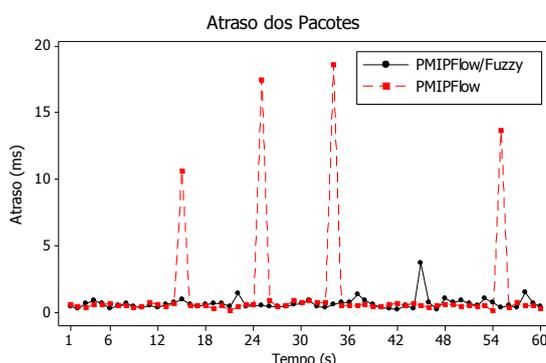


Figura 5. Gráfico de atraso dos pacotes

A Figura 5 mostra o gráfico de uma das coletas comparando-se o PMIPFlow com a proposta de antecipação usando *Fuzzy*. Ela mostra alguns picos de variação de atraso de pacotes quando não se usa a proposta *fuzzy*. Cada pico acontece no momento do *handover*.

Como alguns pacotes são perdidos, os mesmos precisam ser reenviados de modo que o atraso para entrega desse pacote aumenta de forma acentuada.

A Tabela 3 mostra as estatísticas descritivas para os dados de atraso de pacotes durante todos os testes. A média com a proposta *fuzzy* possui um atraso médio de 43.59% menor que os testes sem a proposta. Em relação ao desvio padrão, o PMIPFlow (*Fuzzy*) é mais vantajoso, em média 0.48 ms contra 3.76 ms, com e sem a proposta, respectivamente.

Tabela 3. Estatística descritiva para o atraso dos pacotes

Proposta	Média	Desvio	Mínimo	Q1	Mediana	Q3	Máximo
PMIPFlow	1,489	3,764	0,092	0,444	0,523	0,634	18,639
PMIPFlow (Fuzzy)	0,6506	0,4869	0,1900	0,4183	0,5510	0,7062	3,7100

A Figura 6 mostra a vazão obtida em um dos testes. Nessa Figura, percebe-se que as perdas de *handover* que ocorrem influenciam diretamente na vazão.

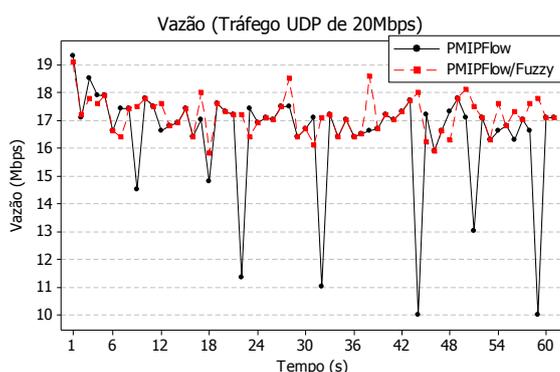


Figura 6. Vazão do tráfego UDP limitado à 20Mbps

A Tabela 4 mostra o resumo estatístico dos testes. Observando-se a Tabela, na média da vazão, a proposta é 3,93% maior que sem a proposta, sendo que os valores podem ser maiores se o tráfego perdurar por mais tempo. Uma métrica importante nesse caso é o valor mínimo atingido pela vazão, de 10 Mbps com PMIPFlow e de 15,8 Mbps com a proposta PMIPFlow (*Fuzzy*), mostrando que sem a proposta de antecipação a vazão degrada bastante devido às perdas.

Tabela 4. Estatística descritiva para vazão de 20Mbps

Proposta	Média	Desvio	Mínimo	Q1	Mediana	Q3	Máximo
PMIPFlow	16,51	1,81	10,00	16,55	17,00	17,35	19,30
PMIPFlow (Fuzzy)	17,16	0,66	15,80	16,65	17,20	17,60	19,10

Na avaliação de *handovers* uma métrica importante é a perda de pacotes. A Figura 7 mostra a porcentagem de perda de pacotes durante dez testes. A Figura mostra que sem a proposta o número de perdas de pacotes é consideravelmente maior, chegando até 6,2 % de perda.

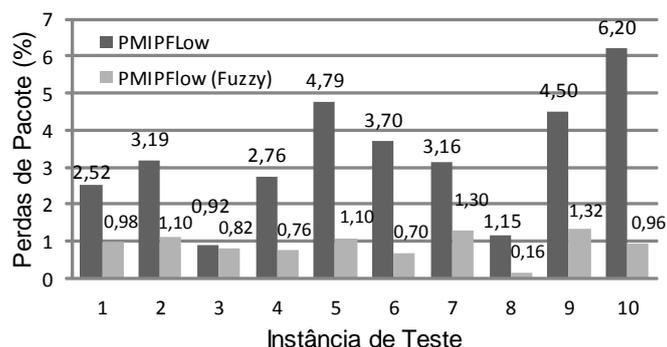


Figura 7. Gráfico em barra da porcentagem de Perdas de Pacotes

3.2. Avaliação de QoE

Os novos estudos de desempenho em sistemas multimídia têm como base as métricas de QoE. Dentre estas, o PSNR é uma métrica tradicional de QoE que estima a qualidade do vídeo em decibéis, comparando o vídeo original com o vídeo recebido pelo usuário. Para cada faixa de valores de PSNR, há uma qualificação para o vídeo que foi recebido pelo usuário.

A métrica SSIM também é bastante utilizada para avaliação de QoE. Ela é baseada na medição quadro a quadro do vídeo original com o vídeo recebido pelo usuário. O SSIM compara a similaridade entre os vídeos nos seguintes aspectos: contraste, luminosidade e estrutura. O SSIM é expresso como um valor decimal entre 0 e 1. Quanto mais próximo de 1, melhor é a qualidade do vídeo.

Para avaliação do vídeo no cenário foram utilizados *trailers* do filme "Os Vingadores" com três resoluções diferentes: 160x90, 320x180 e 640x360.

Os vídeos foram enviados um de cada vez. Durante a transmissão, o usuário permanecia em constante movimento de um ponto de acesso para outro. Após o recebimento do vídeo foi utilizado a versão gratuita da ferramenta MSU [MSU Video Group 2013] para analisar e extrair as informações necessárias para avaliação do vídeo recebido.

A Figura 8 resume os resultados obtidos. A média do PSNR é exibida na Figura 8(a). Nela é possível observar que a diferença entre as duas abordagens é maior com resoluções mais altas. Isso ocorre porque quanto maior a resolução, maior será o número de pacotes necessários para formar um quadro. Sendo assim, quanto maior o número de pacotes, maior serão as perdas no caso de desconexão não transparente. A Figura 8(b) mostra o desvio padrão, quanto menor o espalhamento dos dados mais estável é o sistema.

A Figura 8(c) mostra a média para a métrica SSIM. Neste caso, o PMIPFlow com a proposta de antecipação se mostrou mais eficaz. O desvio padrão, mostrado na Figura 8(d), é menor com a proposta.

Em termos quantitativos, no teste de atraso de pacotes a proposta *fuzzy* foi 128,88% melhor que sem a proposta. Em relação à vazão, para o teste de 20Mbps, a proposta foi cerca de 4% melhor. Em relação à porcentagem de perda de pacotes, o PMIPFlow (*Fuzzy*) é superior em 260%.

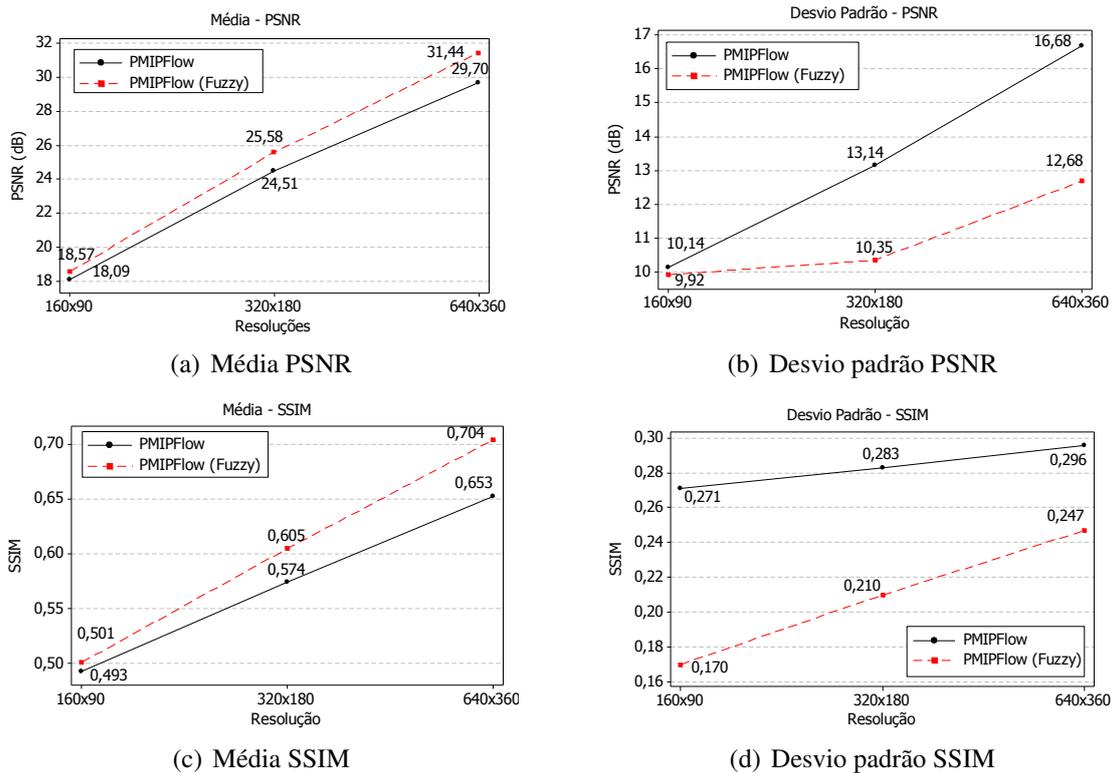


Figura 8. Média e desvio padrão das métricas PSNR e SSIM para cada resolução

Levando-se em conta a avaliação das métricas de QoE, na análise de vídeo, foram utilizados três resoluções diferentes: 160x90, 320x180 e 640x360. No teste da métrica PSNR, a proposta foi superior em 2,65%, 4,36% e 5,85%, respectivamente. No caso da métrica SSIM, a superioridade foi de 1,6%, 5,40% e 7,8%, respectivamente.

Para visualizar de forma prática os benefícios da proposta, a Figura 9 mostra o *frame* 598 retirado do vídeo recebido durante o processo de *handover*. É possível perceber que as perdas de *frames*, durante o processo de *handover*, prejudicam a reconstrução e degradam a qualidade do vídeo. Quando o PMIPFlow (*Fuzzy*) é utilizado as perdas são minimizadas e a reconstrução do vídeo é realizada de forma completa, com isso, o usuário recebe um vídeo com qualidade e não toma conhecimento da ocorrência do *handover*.

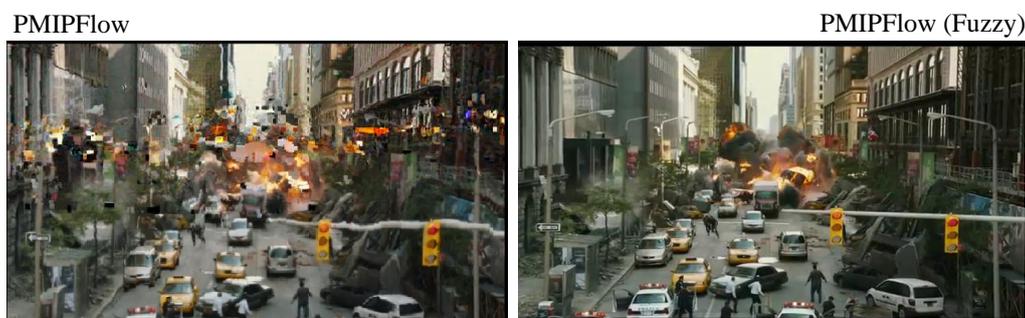


Figura 9. Frame 598 no instante do *handover* com o PMIPFlow e com o PMIPFlow (*Fuzzy*)

4. Trabalhos Relacionados

Alguns trabalhos abordam mobilidade em redes definidas por *software*. O trabalho [Yap et al. 2010a] discute como as redes sem fio ainda estão paradas no tempo, fechadas e sem inovação. No entanto, os autores não fornecem uma proposta para melhorar o processo de *handover* do usuário e admitem a existência de desconexão no *handover*. Sendo assim, fica a critério do projetista de redes implementar aplicações no controlador para o gerenciamento de mobilidade.

Os autores em [Nakauchi et al. 2011] propõem uma plataforma de virtualização chamada AMPHIBIA. Ela possibilita a criação de fatias fim-a-fim sobre redes virtuais cabeadas e sem fio. Contudo, os autores afirmam que o foco da virtualização está no ambiente cabeado. O trabalho cita projetos como GENI [Turner 2006], ORBIT [Orbit-Lab 2005] e 4WARD [Niebert et al. 2008] que tratam de redes virtuais sem fio, mas com foco apenas no isolamento dos recursos das redes sem fio.

Os autores em [Li et al. 2012] abordam SDN em redes LTE (*Long Term Evolution*). O trabalho inicialmente discute que as redes celulares sofrem com a inflexibilidade, equipamentos caros e protocolos complexos no plano de controle. Nos entanto, para que esta nova arquitetura SDN seja utilizada alguns problemas precisam ser resolvidos como: suporte a vários assinantes, mobilidades frequentes, controle e medições mais apuradas da rede e adaptações em tempo real. O sistema LTE é bastante semelhante à arquitetura PMIP, de modo que, o PMIPFlow poderia ser implementado também nesta arquitetura de rede celular.

A aplicação de lógica *fuzzy* para decisão de *handover* em redes heterogêneas foi apresentada em [Wu 2011]. Foram utilizados o RSS, velocidade e carga do sistema como entradas do sistema *fuzzy*. Na avaliação, o sistema, nomeado de FUN-HODs (*Fuzzy Normalization for Handover Decision Strategy*), se mostrou vantajoso, no entanto, o cálculo de velocidade se torna inviável em dispositivos com restrições de energia.

O trabalho em [Mun-Yee Lim and Chow 2012] também implementa um sistema de decisão de *handover* baseado em lógica *fuzzy*. São utilizadas como entrada as métricas: RSS, velocidade e distância. A avaliação da proposta foi realizada usando a tecnologia 802.11 com o protocolo de mobilidade MIPv6. As simulações foram conduzidas no *framework* OMNET++/xMIPv6 [Yousaf et al. 2008]. A proposta foi comparada com outras três. Nas simulações, a solução obteve um bom desempenho, porém, toda a proposta baseada em velocidade precisa de coletas com alto grau de consumo de energia, como por exemplo usando GPS, o que dificulta a utilização em dispositivos com restrição de energia.

Outro trabalho que se baseia em lógica *fuzzy* para o mecanismo de *handover* é apresentado em [Sharma and Kholá 2012]. Este trabalho usa como entrada as métricas: largura de banda, predição do RSS e preferências do usuário. Ao contrário dos outros, esse artigo usa uma predição de RSS e não o próprio valor. Nessa proposta, uma das entradas do sistema *fuzzy* é a preferência do usuário. Nela o usuário deixa as tomadas de decisão do *handover* predeterminadas.

Diferente dos artigos citados, este trabalho propõe uma nova arquitetura para gerenciamento de mobilidade em redes definidas por *software*. Foi proposto um esquema baseado em lógica *fuzzy* para reduzir a duração do processo de *handover*. Além disso, a

arquitetura foi implementada em um *testbed* sem fio OpenFlow.

5. Conclusão

Este trabalho apresentou a avaliação da proposta de antecipação de *handover* para o PMIPFlow. Os resultados dos testes mostraram que a proposta é eficiente e evita a perda excessiva de pacotes que normalmente ocorre durante o processo de *handover*.

Em todos os testes, a utilização da proposta se mostrou vantajosa. Quanto maior a resolução dos vídeos, maior sua vantagem. Um vídeo com uma resolução maior requer mais pacotes para enviar um *frame*, o que significa um maior tráfego pela rede. Se mais pacotes trafegam, maiores serão as perdas devido às possíveis desconexões.

No futuro, os dispositivos serão mais sofisticados, com mais memória, maior poder de processamento, de modo, que a resolução das aplicações de vídeo tendem a aumentar. O que antes eram apenas voz e texto, no futuro, serão vídeos de altas resoluções. Por isso, garantir a continuidade de serviço é algo de extrema importância para as redes móveis do futuro.

Referências

- Chowdhury, N. M. K. and Boutaba, R. (2009). Network virtualization: state of the art and research challenges. *Communications Magazine, IEEE*, 47(7):20–26.
- Firoiu, V., Le Boudec, J.-Y., Towsley, D., and Zhang, Z.-L. (2002). Theories and models for internet quality of service. *Proceedings of the IEEE*, 90(9):1565–1591.
- Gundavelli, S., Chowdhury, K., Devarapalli, V., Patil, B., Leung, K., et al. (2008a). Proxy mobile ipv6.
- Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., and Patil, B. (2008b). Rfc 5213, proxy mobile ipv6.
- Internet Engineering Task Force (2010). Network-based Localized Mobility Management. <http://datatracker.ietf.org/doc/charter-ietf-netlmm/>.
- Kempf, J. (2007). Problem statement for network-based localized mobility management (netlmm).
- Klir, G. J. and Yuan, B. (1995). *Fuzzy sets and fuzzy logic*. Prentice Hall New Jersey.
- Li, L., Mao, Z., and Rexford, J. (2012). Toward software-defined cellular networks. In *Software Defined Networking (EWSN), 2012 European Workshop on*, pages 7–12.
- McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and Turner, J. (2008). Openflow: enabling innovation in campus networks. *SIGCOMM Comput. Commun. Rev.*, 38(2):69–74.
- MSU Video Group (2013). Video Quality Measurement Tool. http://compression.ru/index_en.htm.
- Mun-Yee Lim, J. and Chow, C.-O. (2012). Smart handover based on fuzzy logic trend in ieee802. 11 mobile ipv6 networks.

- Nakauchi, K., Ishizu, K., Murakami, H., Nakao, A., and Harada, H. (2011). Amphibia: A cognitive virtualization platform for end-to-end slicing. In *Communications (ICC), 2011 IEEE International Conference on*, pages 1–5. IEEE.
- Niebert, N., Baucke, S., El-Khayat, I., Johnsson, M., Ohlman, B., Abramowicz, H., Wuenstel, K., Woesner, H., Quittek, J., and Correia, L. M. (2008). The way 4ward to the creation of a future internet. In *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pages 1–5. IEEE.
- Orbit-Lab (2005). Orbit: Open-Access Research Testbed for Next-Generation Wireless Networks. <http://www.orbit-lab.org/>.
- Sharma, M. and Khola, R. (2012). Fuzzy logic based handover decision system. *International Journal*, 3.
- Sherwood, R., Gibb, G., Yap, K., Appenzeller, G., Casado, M., McKeown, N., and Parulkar, G. (2009). Flowvisor: A network virtualization layer. *OpenFlow Switch Consortium, Tech. Rep.*
- Turner, J. S. (2006). A proposed architecture for the geni backbone platform. In *Architecture for Networking and Communications systems, 2006. ANCS 2006. ACM/IEEE Symposium on*, pages 1–10. IEEE.
- VMware (2012). VMware vSphere Hypervisor (ESXi) 5.0. <http://www.vmware.com/products/vsphere-hypervisor/overview.html>.
- Winkler, S. and Mohandas, P. (2008). The evolution of video quality measurement: from psnr to hybrid metrics. *Broadcasting, IEEE Transactions on*, 54(3):660–668.
- Wu, S.-J. (2011). Fuzzy-based handover decision scheme for next-generation heterogeneous wireless networks. *Journal of Convergence Information Technology*, 6(4).
- Yap, K., Sherwood, R., Kobayashi, M., Huang, T., Chan, M., Handigol, N., McKeown, N., and Parulkar, G. (2010a). Blueprint for introducing innovation into wireless mobile networks. In *Proceedings of the second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures*, pages 25–32. ACM.
- Yap, K.-K., Kobayashi, M., Sherwood, R., Huang, T.-Y., Chan, M., Handigol, N., and McKeown, N. (2010b). Openroads: Empowering research in mobile networks. *ACM SIGCOMM Computer Communication Review*, 40(1):125–126.
- Yousaf, F. Z., Bauer, C., and Wietfeld, C. (2008). An accurate and extensible mobile ipv6 (xmip6) simulation model for omnet++. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, page 88. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).