Universidade Federal da Paraíba

Centro de Informática

Programa de Pós-Graduação em Informática

# DISpatCH: Uma abordagem SDWN para o gerenciamento do processo de *handoff* nas redes *Wi-Fi*

JOSÉ GOMES QUARESMA FILHO

**João Pessoa, Paraíba, Brasil**

31 de Agosto de 2016

JOSÉ GOMES QUARESMA FILHO

# DISpatCH: Uma abordagem SDWN para o gerenciamento do processo de *handoff* nas redes *Wi-Fi*

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Informática da Universidade Federal da Paraíba como parte dos requisitos necessários para obtenção do grau de Mestre em Informática.

Área de Concentração: Ciência da Computação

Linha de Pesquisa: Computação Distribuída

Prof. Dr. Fernando Menezes Matos

(Orientador)

**João Pessoa, Paraíba, Brasil**

Q1d Quaresma Filho, José Gomes.
DISpatCH: uma abordagem SDWN para o gerenciamento do processo de handoff nas redes Wi-Fi / José Gomes Quaresma Filho.- João Pessoa, 2016.
71f.
Orientador: Fernando Menezes Matos
Dissertação (Mestrado) - UFPB/CI
1. Informática. 2. SDWN. 3. WI-Fi. 4. Handoff. 5. QoS. 6. IEEE 802.11.

UFPB/BC CDU: 004(043)

**UNIVERSIDADE FEDERAL DA PARAÍBA**
**CENTRO DE INFORMÁTICA**
**PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

Ata da Sessão Pública de Defesa de Dissertação de Mestrado de **JOSÉ GOMES QUARESMA FILHO**, candidato ao título de Mestre em Informática na Área de Sistemas de Computação, realizada em 31 de agosto de 2016.

Aos trinta e um dias do mês de agosto, do ano de dois mil e dezesseis, às sete horas, no Centro de Informática da Universidade Federal da Paraíba, em Mangabeira, reuniram-se os membros da Banca Examinadora constituída para julgar o Trabalho Final do Sr. JOSÉ GOMES QUARESMA FILHO, vinculado a esta Universidade sob a matrícula nº 2014107989, candidato ao grau de Mestre em Informática, na área de "Sistemas de Computação", na linha de pesquisa *"Computação Distribuída"*, do Programa de Pós-Graduação em Informática, da Universidade Federal da Paraíba. A comissão examinadora foi composta pelos professores: Fernando Menezes Matos, Orientador e Presidente da Banca, Iguatemi Eduardo da Fonseca, Examinador Interno ao Programa, e Reinaldo Cezar de Morais Gomes, Examinador Externo à Instituição. Dando início aos trabalhos, o Presidente da Banca, cumprimentou os presentes, comunicou aos mesmos a finalidade da reunião e passou a palavra ao candidato para que o mesmo fizesse a exposição oral do trabalho de dissertação intitulado "Uma Abordagem SDWN para o Gerenciamento do Processo de Handoff nas Redes WiFi ". Concluída a exposição, o candidato foi arguido pela Banca Examinadora que emitiu o seguinte parecer: "*aprovado*". Do ocorrido, eu, Clauirton de Albuquerque Siebra, Coordenador do Programa de Pós-Graduação em Informática, lavrei a presente ata que vai assinada por mim e pelos membros da Banca Examinadora.

João Pessoa, 31 de agosto de 2016.

Prof. Dr. Clauirton de Albuquerque Siebra

Prof. Dr. Fernando Menezes Matos
Orientador (PPGI-UFPB)

Prof. Dr. Iguatemi Eduardo da Fonseca
Examinador interno ao Programa (PPGI-UFPB)

Prof. Dr . Reinaldo Cezar de Morais Gomes
Examinador externo à Instituição (UFCG)

## DEDICATÓRIA

Dedico este trabalho a minha esposa Anike e a meus três lindos filhos: Igor, Pedrinho e João. Vocês são os meus tesouros!

Zé Filho

# Agradecimentos

Agradeço primeiramente a DEUS por ter me dado saúde e força para enfrentar os desafios, que não foram poucos, e enfim conquistar mais este grande objetivo.

Aos meus pais que nunca mediram esforços para proporcionar, a mim e a meus irmãos, a melhor educação possível, sempre indicando os caminhos que deveriam ser seguidos para tornar-nos seres humanos dignos e honrados.

Aos meus irmãos e irmã, e especialmente a minha esposa e meus filhos pelo amor, paciência, cumplicidade, carinho, compreensão e apoio fundamentais para concluir mais essa etapa da minha vida profissional.

A todos os meus colegas de curso, pela parceria e ajuda mútua.

A todos os meus alunos pela oportunidade de dividir conhecimentos e assim aumentar o aprendizado e o crescimento como seres humanos.

A todos os professores que tive ao longo da vida pela generosidade e pelo grande dom de ensinar e compartilhar seus conhecimentos contribuindo para o desenvolvimento da sociedade como um todo em especial a todos os professores do PPGI, meu orientador Professor Dr. Fernando Matos, a banca formada pelo Professor Dr. Iguatemi Fonseca e Professor Doutor Reinaldo Gomes que deram contribuições importantíssimas e aos Professores Doutores Clauirton Siebra e Alisson Brito, muito obrigado !

Agradecimento especial a dois grandes amigos que sem suas ajudas esse trabalho não teria sido concluído: Professor Doutor Anand Subramanian e ao Mestre Nailson Cunha, muito obrigado!

DEUS ABENÇOE A TODOS.

# Resumo

Um dos aspectos mais importantes na implantação de uma rede local sem fio *Wireless Local Area Network* (WLAN) é que em uma área coberta por vários *Access Points* (APs), devem existir mecanismos que sejam suficientes para que o usuário possa se manter conectado a qualquer hora independentemente de sua localização física dentro da rede e, principalmente, garantir que as aplicações não sofram com falhas ou perdas de conexão. Neste contexto, o desempenho do *handoff*, que consiste na troca de AP à medida que uma estação se move dentro de uma mesma área, é fator determinante para o uso de aplicações sensíveis ao atraso. Normalmente o processo é iniciado pelas estações, que ao se desconectarem de um AP iniciam uma varredura nos canais existentes buscando outros APs disponíveis de forma suave, porém na ordem de segundos, o que dificulta o uso de aplicações em tempo real. O desafio é criar um mecanismo mais eficiente para gerenciar este processo provendo a Qualidade de Serviço (QoS) necessária para as aplicações. Este trabalho apresenta uma solução denominada *Detection and dIScovery Control in Handoff* (*DISpatCH*) que utiliza mecanismos para melhorar o desempenho do processo de *handoff* utilizando uma abordagem baseada na arquitetura *Software Defined Wireless Network* (SDWN) os quais foram implementados para validar a proposta. Testes realizados demonstraram uma diminuição no tempo gasto no *handoff* contribuindo para garantir o QoS das aplicações.

**Palavras-chave: DISpatCH, SDWN, *Wi-Fi*, *Handoff*, QoS, IEEE 802.11**

# Abstract

One of the most important aspects concerning the implementation of a Wireless Local Area Network (WLAN) is that in an area covered by several Access Points (APs), there must exist mechanisms to keep the user connected at any time, regardless of his/her physical location in the network and, primarily, ensure that the applications do not suffer from loss of connectivity. In this context, handoff performance, which consists of changing the AP as the station moves within the same area, is a preponderant factor when it comes to applications that are delay-sensitive. The process is usually started by the stations that once disconnected from an AP, start scanning the existing channels searching for other available APs in a smooth fashion, but in the order of seconds, which makes it difficult to use real-time applications. The challenge is to create a more efficient mechanism to manage this process, providing the required Quality of Service (QoS) for the applications. This work presents the use of mechanisms to improve the performance of the handoff process by employing an approach based on the a Software Defined Wireless Network (SDWN), which was implemented to validate the proposal. The tests performed showed a decrease in the time spent in the handoff, contributing to guarantee the QoS of the applications.

**Keywords: DISpatCH, SDWN, Wi-Fi, Handoff, QoS, IEEE 802.11**.

# Sumário

# Lista de acrônimos e siglas

**AP** : *Access Point*

**ESS** : *Extended Service Set*

**BSS** : *Basic Service Set*

**DoS** : *Denial of Service*

**FT** : *Fast Basic Service Set Transition*

**IAPP** : *Inter Access Point Protocol*

**IEEE** : *Institute of Electrical and Electronics Engineers*

**IFPB** : Instituto Federal de Educação Ciência e Tecnologia da Paraíba

**IETF** : *Internet Engineering Task Force*

**IoT** : *Internet of Things*

**MIB** : *Management Information Base*

**OID** : *Object Identifier*

**ONF** : *Open Networking Foundation*

**QoS** : *Quality of Service*

**RF** : *Radio Frequency*

**RSSI** : *Received Signal Strength Indication*

**SDN** : *Software Defined Network*

**SDWN** : *Software Defined Wireless Network*

**SNMP** : *Simple Network Management Protocol*

**STA** : *Station*

**UAI** : Unidade Acadêmica de Informática

**VoIP** : *Voice over IP*

**WLAN** : *Wireless Local Area Network*

# Lista de Figuras

# Lista de Tabelas

# Capítulo 1

# Introdução

## 1.1 Contextualização

Uma rede sem fio (*wireless network*) é uma infraestrutura que permite a transmissão de dados sem a necessidade do uso de cabos sejam eles de cobre ou ótico. Nesse caso, os dispositivos finais (estações) se comunicam por meio de ondas de radiofrequência. Para as redes locais sem fio (WLAN, do inglês *Wireless Local Area Networks*), isto é, aquelas utilizadas para conectar tais dispositivos utilizando o ar como meio de propagação, o padrão adotado é o definido no protocolo 802.11. Desenvolvido pelo *Institute of Electrical and Electronics Engineers* (IEEE), esse protocolo é também conhecido como *Wi-Fi*, termo definido pela *Wi-Fi Alliance*[1], uma rede de empresas que certifica os produtos e serviços desenvolvidos com base nesse protocolo. Dessa forma, entende-se por rede *Wi-Fi* aquela que utiliza os conceitos determinados pelo protocolo IEEE 802.11.

Uma das desvantagens do IEEE 802.11 é que esse protocolo foi projetado inicialmente para prover mobilidade em redes locais de pequeno porte, como por exemplo, escritórios e residências onde normalmente apenas um *Access Point* (AP) consegue fornecer a cobertura de sinal necessária para todo o ambiente. Em geral, para que a área de cobertura seja ampliada, são necessários dois ou mais APs para que, à medida que o usuário se afaste de um, seu dispositivo se conecte a outro automaticamente e de forma transparente para o mesmo. Esse processo de troca de APs é chamado de *handoff* (HU et al., 2016).

[1] http://www.wi-fi.org

De acordo com o IEEE 802.11 o processo de *handoff* é gerenciado pelo próprio dispositivo móvel e pelo AP ao qual está associado. Por outro lado, a arquitetura baseada nas Redes Definidas por Software (SDN, do inglês *Software Defined Networks*) separa o plano de controle (*software*) do plano de dados (*hardware*) possibilitando que algumas fases do processo sejam gerenciadas por um *software* instalado em um servidor localizado fora do núcleo da rede. Assim sendo, fases importantes como o momento de iniciar o *handoff* e para qual AP a estação deve migrar são implementadas em um servidor externo que toma decisões baseadas em *softwares* desenvolvidos em qualquer linguagem introduzindo uma maior flexibilidade, retirando assim as decisões do núcleo da rede (COSTANZO et al., 2012). A arquitetura SDN no âmbito das redes sem fio é denominada de Redes Sem Fio Definidas por Software (SDWN, do inglês *Software Defined Wireless Networks*).

Esse trabalho apresenta uma solução denominada *Detection and dIScovery Control in Handoff (DISpatCH)* desenvolvida tendo como base a arquitetura SDWN com o objetivo de melhorar o desempenho do processo de *handoff* no âmbito das redes *Wi-Fi* através da programação de um módulo de *software* executado em um servidor externo que, baseado em algumas premissas pré-determinadas, passa a tomar decisões do momento em que o processo de *handoff* deve ser iniciado e para qual AP a estação (STA) deve migrar. Um protótipo foi implementado e testes experimentais foram realizados para verificar a eficiência do DISpatCH. Os resultados dos testes mostraram que, utilizando DISpatCH, foi possível diminuir o tempo do processo de *handoff* melhorando assim a qualidade das comunicações principalmente para aplicações em tempo real e sensíveis ao atraso.

## 1.2 Motivação

Com a padronização das redes locais sem fio pelo IEEE que lançou em 1997 a primeira versão do protocolo 802.11, as redes *Wi-Fi* logo passaram a dominar os ambientes corporativos, domésticos e comerciais. Alguns fatores contribuem para impulsionar ainda mais o desenvolvimento desse protocolo, a saber: o número crescente de dispositivos móveis tais como *smartphones*, *tablets* e *notebooks*; a difusão do novo paradigma das cidades inteligentes e Internet das Coisas (IoT, do inglês *Internet of Things*); a grande utilização de *softwares* de comunicação instantânea a exemplo do *WhatsApp* e *Telegram*; o frequente uso de apli-

cativos de redes sociais como o *Facebook*, *Instagram* e *Snapchat*; a tendência de uso do protocolo por parte das operadoras de telefonia celular para complementar seus serviços e sua cobertura; entre outros. Além disso, as redes *Wi-Fi* possuem uma relação custo-benefício favorável e tem na simplicidade da instalação e configuração grandes aliados. A maioria dos dispositivos chega ao mercado pré-configurado, habilitando usuários com pouco ou quase nenhum conhecimento de redes a instalá-los e colocá-los em funcionamento. No Brasil não é raro encontrar equipamentos baratos necessários para se montar uma rede *Wi-Fi* como, por exemplo, APs, roteadores, repetidores, antenas e adaptadores *usb*.

Ao mesmo tempo em que se tornam presentes em quase todos os locais, as redes *Wi-Fi* são exigidas cada vez mais pelos usuários que precisam estar conectados em qualquer lugar e a qualquer instante. Além disso, aplicações multimídia em tempo real como *Voice over IP* (VoIP), *streaming* de vídeo (*youtube*, *netflix*) e jogos passaram a ser cada vez mais frequentes e a exigir novos requisitos de desempenho da rede. Nesse contexto, faz-se necessário a utilização de vários APs, tanto para intensificar a força do sinal recebido pelas estações, como para aumentar a área de cobertura. Dessa forma, é importante que o *handoff* aconteça de forma ágil, suave e sem prejuízo para as aplicações.

Apesar de funcional e já bem estabelecido, o processo de *handoff* definido pelo IEEE 802.11 deixa a desejar em cenários onde são executadas aplicações multimídia, sobretudo em relação ao tempo, em virtude de suas sensibilidades ao atraso. Com o intuito de aprimorar esse processo, o IEEE tem feito um esforço publicando emendas sobre o assunto. No entanto, a fase de detecção, que determina o momento em que o processo de *handoff* é iniciado, não é contemplada pelo protocolo, ficando sua implementação sob responsabilidade de cada fabricante. Para minimizar este problema, os administradores de redes tem a opção de utilizar equipamentos e *softwares* de gerenciamento proprietários que tem em seu custo um grande empecilho para sua implementação. Os algoritmos utilizados normalmente não são flexíveis, pois dependem dos fabricantes para atualizações ou programações mais específicas.

Uma potencial solução para o problema supracitado é a utilização de uma abordagem SDWN. É possível encontrar na literatura alguns trabalhos que tal abordagem foi utilizada com sucesso, porém ainda sem levar em consideração a fase de detecção, como por exemplo, Sun e Qian (2016a), Luengo (2016), Sanghavi e Bansode (2015), Moura et al. (2015), Yan et al. (2015), Tetarwal, Kuntal e Karmakar (2014) e Al-Shaikhli (2014). Assim sendo, o pre-

sente trabalho apresenta o DISpatCH, uma solução em que um controlador externo faz uso de mecanismos programáveis determinando não só o momento apropriado para o início do *handoff* (detecção) como também a escolha do AP de destino (descoberta). Os resultados obtidos através dos experimentos realizados em um ambiente real sugerem que tal mecanismo é mais eficiente que a abordagem tradicional, reduzindo em média, o tempo de *handoff* em 1 s.

## 1.3 Objetivos

### 1.3.1 Objetivo Geral

O objetivo deste trabalho é propor uma abordagem utilizando os conceitos de SDWN para gerenciar o processo de *handoff* em uma rede *Wi-Fi* com a finalidade de diminuir o tempo consumido nas fases de detecção e descoberta, utilizando *software* livre e dispositivos comercias de baixo custo.

### 1.3.2 Objetivos Específicos

Os objetivos específicos são os seguintes:

- Compreender de forma detalhada o funcionamento da abordagem tradicional do processo de *handoff* definidos pelo protocolo IEEE 802.11.
- Identificar os principais fatores que exercem influência no desempenho durante as fases de detecção e descoberta.
- Desenvolver um *software* gerente, utilizando a abordagem SDWN para monitorar os indicadores da rede que exercerão influência de tais fatores.
- Desenvolver um módulo para instalar as tabelas de fluxo nos APs usando um controlador externo.
- Realizar experimentos com as abordagens tradicional e SDWN utilizando dispositivos comerciais e de baixo custo.
- Comparar os resultados obtidos a partir da implementação das duas abordagens.

## 1.4 Estrutura da Dissertação

O restante da dissertação está organizado como segue. O Capítulo 2 discorre sobre os fundamentos teóricos necessários ao entendimento dos mecanismos propostos bem como são abordados alguns trabalhos relacionados. No Capítulo 3 é apresentado o DISpatCH e a abordagem proposta para o *handoff* utilizando o paradigma das SDWN e no Capítulo 4 são apresentados detalhes dos experimentos realizados bem como os resultados alcançados comparando a abordagem tradicional com a abordagem proposta. Por fim, o Capítulo 5 apresenta as conclusões e aponta propostas para trabalhos futuros.

# Capítulo 2

# Fundamentação Teórica

Neste capítulo serão abordados os principais assuntos que serviram como base para o desenvolvimento da proposta apresentada.

## 2.1 O protocolo IEEE 802.11

O protocolo IEEE 802.11, também conhecido como *Wi-Fi*, é um conjunto de especificações nas camadas *Media Access Control (MAC)* e física (PHY) desenvolvidas com o objetivo de implementar comunicação sem fio em redes locais nas faixas de freqüência de 900 MHz e 2,4, 3,6, 5 e 60 GHz. O protocolo foi criado e é mantido no comitê IEEE 802 do Instituto de Engenharia Elétrica e Eletrônica (IEEE) e desde que teve sua primeira versão aprovada em 1997 vem sofrendo várias transformações. As modificações são apresentadas inicialmente como emendas e recebem uma letra do alfabeto complementando o nome do protocolo. A primeira emenda foi lançada no ano de 1999 e se chamava 802.11a. Alcançava 54 Mbps de taxa de transmissão e operava na faixa de frequência de 5 GHz. A última alteração ao padrão foi aprovada no comitê em setembro de 2015 e tem seu lançamento previsto para setembro de 2017. Esta é a emenda 802.11az e especifica um protocolo de posicionamento nas redes *Wi-Fi* (KREUTZ et al., 2015). O protocolo e suas alterações fornecem a base para produtos de rede sem fio que usam a marca *Wi-Fi*. Embora cada alteração seja oficialmente revogada quando incorporada na versão mais recente da norma, o mundo corporativo tende a comercializar produtos utilizando os nomes das emendas pois desta forma fica mais fácil para os usuários identificar de maneira m ais precisa a capacidade dos produtos.

As principais alterações desde que foi aprovado em 1997 e que tenham relação com o processo de *handoff* estão descritas na Tabela 2.1.

**Tabela 2.1:** IEEE 802.11 e suas principais alterações

| Padrão | Descrição |
|---|---|
| Std P802.11-1997 | *Std for Wireless Lan MAC and PHY Specifications* |
| Std P802.11-1999 | *Part II Wireless Lan MAC and PHY Specifications* |
| Std P802.11f-2003 | *Inter-Access Point Protocol Across Distribution Systems Supporting* |
| Std P802.11i-2004 | *MAC Security Enhancements* |
| Std P802.11e-2005 | *MAC Enhancements QoS* |
| Std P802.11-2007 | *802.11 Standard Maintenance Revision* |
| Std P802.11k-2008 | *Radio Resource Measurement* |
| Std P802.11r-2008 | *Fast Roaming* |
| Std P802.11u-2011 | *InterWorking with External Networks* |
| Std P802.11ai 2010 | *Fast Initial Link Setup (previsto pra dez/2016)* |
| Std P802.11aq 2012 | *Pre-association Discovery (previsto pra dez/2016)* |
| Std P802.11-2012 | *802.11 Accumulated Maintenance Changes* |
| Std P802.11ae-2012 | *Prioritization of Management Frames* |
| Std P802.11ac-2013 | *Very High Throughput 6GHz* |

Algumas dessas emendas foram desenvolvidas visando melhorar a mobilidade nas redes *Wi-Fi* apresentando novos mecanismos com o objetivo de diminuir o tempo gasto no processo de *handoff*. Entre elas podemos citar a 802.11f, 802.11k, 802.11r e 802.11u. Todas essas emendas foram incorporadas ao padrão na revisão que ocorreu no ano de 2012. Importante ressaltar que a fase de detecção, objeto da proposta aqui apresentada, não é comtemplada por nenuma delas.

### 2.1.1 Mobilidade

Impulsionada pelo novo paradigma das cidades inteligentes e Internet das Coisas (IoT, do inglês *Internet of Things*), a mobilidade é, nos dias atuais, uma das propriedades mais importantes das redes de comunicação, onde os dispositivos precisam estar conectados a qualquer hora e em qualquer lugar (ATZORI; IERA; MORABITO, 2010). Neste cenário, problemas como computação ubíqua, controle, vigilância e monitoramento de objetos e pessoas emergem e devem ser encarados com seriedade. Como exemplo da utilização desses conceitos pode-se citar um sistema de monitoramento móvel em um campus universitário onde os vigilantes utilizariam capacetes dotados de câmeras conectadas à rede *wireless* para, ao se

deslocarem, transmitirem imagens em tempo real para uma central de monitoramento. Neste caso o *Wi-Fi* é a tecnologia de comunicação mais utilizada e este cenário está exemplificado na Figura 2.1.

**Figura 2.1:** Cenário Exemplo

Apesar de outros padrões terem sido desenvolvidos para as Redes Locais Sem Fio (WLAN, do inglês *Wireless Local Area Network*), o 802.11 é o protocolo dominante para este tipo de rede. O problema é que este protocolo foi projetado inicialmente para prover mobilidade em redes locais de pequeno porte, como por exemplo, pequenos escritórios e residências onde normalmente apenas um *Access Point* (AP) consegue fornecer a cobertura de sinal necessária para todo o ambiente. Em geral, para que a área de cobertura seja ampliada, são necessários dois ou mais APs para que, à medida que o usuário se afasta de um, seu dispositivo se conecte a outro automaticamente e de forma transparente para o mesmo. Esse processo de troca de APs é chamado de *seamless handoff* (COSTANZO et al., 2012). Apesar de funcional e já bem estabelecido, o processo de *handoff* deixa a desejar em cenários onde são executadas aplicações multimídia como, por exemplo, *Voice over IP* (VoIP) e/ou transmissão de vídeo em tempo real.

### 2.1.2 *Handoff*

De acordo com o padrão IEEE 802.11 as redes *Wi-Fi* são formadas por 3 elementos básicos: *Access Point* (AP), Estação (STA) e Sistema de Distribuição (DS) que normalmente é a rede cabeada de uma organização. Este conjunto forma um *Basic Service Set* (BSS). Cada BSS

possui 2 parâmetros de identificação: o *Basic Service Set Identification* (BSSID) formado pelo endereço MAC da interface de rede do AP identificado por um conjunto de 48 bits denotados em hexadecmimal e o *Service Set Identification* SSID que é um conjunto de até 32 bytes (32 caracteres ASCII) utilizado pelos usuários para identificar a rede que deseja se conectar. A mobilidade é a característica mais importante de uma rede sem fio porém nas redes *Wi-Fi*, para prover mobilidade em uma área de cobertura maior, são necessários dois ou mais APs criando assim um serviço extendido de células - *Extended Service Set* (ESS), que se caracteriza por um conjunto de dois ou mais BSSs interconectados, que aparentam ser para o usuário um único BSS. O movimento de uma STA dentro de um mesmo ESS pode ocasionar a associação deste dispositivo em outro BSS. Este processo é chamado de *handoff* e está ilustrado na Figura 2.2.

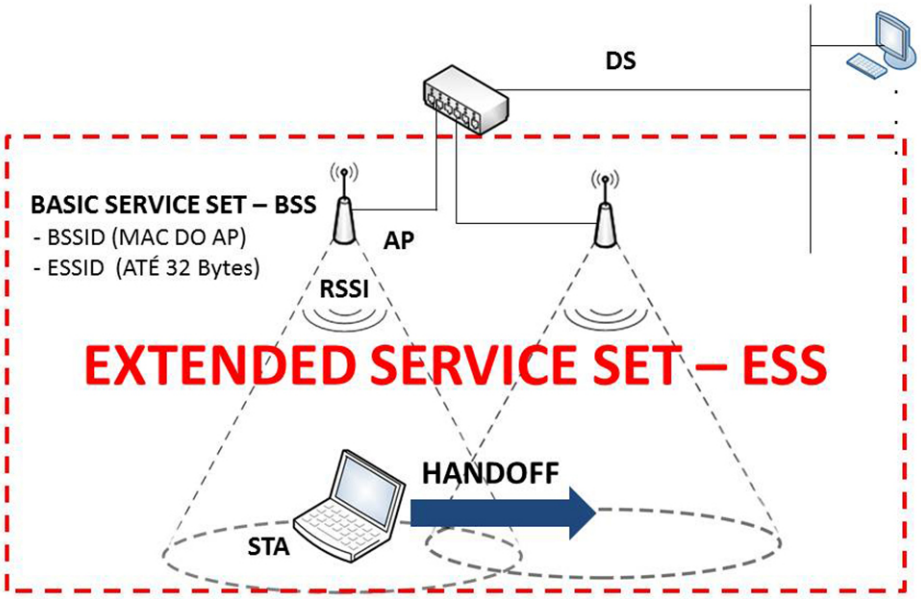



**Figura 2.2:** Processo de *handoff*

No padrão IEEE 802.11 (GROUP et al., 2012) o processo de *handoff* é definido em três etapas: descoberta, autenticação e reassociação, porém para dar início ao processo, a STA precisa detectar a necessidade da troca de AP. Esta fase de detecção não é padronizada, ficando sua implementação sob a responsabilidade de cada fabricante. O tempo dispendido em cada etapa acrescenta um atraso ao processo inteiro e esses valores são muito importantes para aplicações em tempo real que normalmente são sensíveis ao atraso e ao *jitter* (variação do atraso). A Figura 2.3 apresenta o processo de *handoff* e suas fases bem como

as mensagens trocadas em cada etapa.

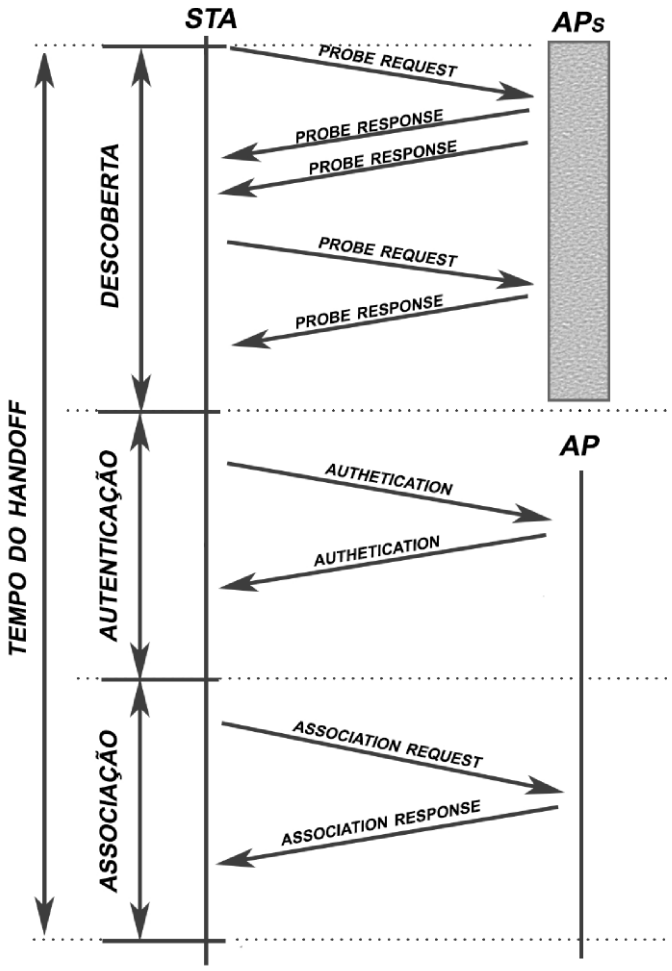



**Figura 2.3:** Fases do processo de *handoff*

#### 2.1.2.1 Estrutura dos *Frames*

O padrão 802.11 define três tipos de *frames*:

- *Frames* de gerenciamento
- *Frames* de controle
- *Frames* de dados

Um *frame* ou quadro *Wi-Fi* é formado por um conjunto de campos variáveis onde apenas alguns deles aparecem em todos os quadros. A Figura 2.4 representa o formato geral do quadro.

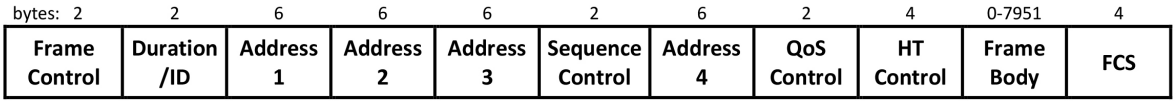



**Figura 2.4:** Formato geral do *frame Wi-Fi*. Adaptado de (GROUP et al., 2012).

Os campos *Frame Control*, *Duration/ID* e *Address 1* e o campo (FCS) constituem o formato de quadro mínimo e estão presentes em todos os *frames*, incluindo tipos e subtipos reservados. Os primeiros 16 bits formam o *Frame Control*. Esses dois octetos possuem campos que definem as funções do quadro. Dentre estes destacamos os campos *Type*, *Subtype* e *Power Management*. Os campos que compõem o *Frame Control* estão demonstrados na Figura 2.5.

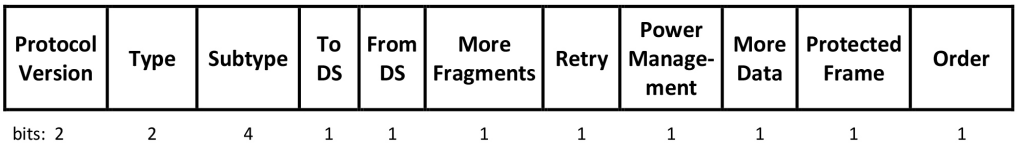



**Figura 2.5:** Campos do *Frame Control*.Adaptado de (GROUP et al., 2012).

Os campos *type s subtype* identificam se o quadro é de gerenciamento, controle ou dados. Os *frames* de gerenciamento tem o campo *type* com valor 0. Já o *frames* de controle possuem o campo *type* definido em 1 e os *frames* de dados correspondem ao campo *type* igual a 2.

Para cada tipo de *frame* o IEEE padronizou até 16 subtipos identificados por 4 *bits* do campo *subtype*. O campo *subtype* varia de acordo com a função que o quadro pode assumir.

Durante o processo de *handoff* a maioria das mensagens trocadas fazem parte dos *frames* de gerenciamento. O *frame* de dados com subtipo *null function* é utilizado para indicar perda de contato entre os dispositivos.

Neste trabalho o *software Wireshark* foi usado para analisar o tráfego. Os *frames* mais relevantes para este estudo, bem como a sintaxe dos filtros utilizados no *Wireshark* estão apresentados na Tabela 2.2.

A seguir são descritas de maneira geral as funções de cada *frame*.

- *Data frame*: basicamente um *frame* contendo dados. Significa que no seu *payload* está um datagrama IP ou uma mensagem ARP. O campo *protected frame* do *Frame Control* definido com o valor 1 indica que o *payload* está criptografado.

**Tabela 2.2:** *Frames* usados no *handoff*

| Tipo | Subtipo | Descrição | Filtro *Wireshark* |
|---|---|---|---|
| 2 | 0 | *Data Frame* | wlan.fc.type_subtype===0x20 |
| 2 | 8 | *QoS Data* | wlan.fc.type_subtype===0x28 |
| 2 | 4 | *Null Function* | wlan.fc.type_subtype===0x24 |
| 0 | 0 | *Association request* | wlan.fc.type_subtype===0x00 |
| 0 | 1 | *Association response* | wlan.fc.type_subtype===0x01 |
| 0 | 2 | *Reassociation request* | wlan.fc.type_subtype===0x02 |
| 0 | 3 | *Reassociation response* | wlan.fc.type_subtype===0x03 |
| 0 | 4 | *Probe request* | wlan.fc.type_subtype===0x04 |
| 0 | 5 | *Probe response* | wlan.fc.type_subtype===0x05 |
| 0 | 8 | *Beacon* | wlan.fc.type_subtype===0x08 |
| 0 | 10 | *Disassociation* | wlan.fc.type_subtype===0x0a |
| 0 | 11 | *Authentication* | wlan.fc.type_subtype===0x0b |
| 0 | 12 | *Deauthentication* | wlan.fc.type_subtype===0x0c |

- *QoS frame*: uma versão do data *frame* habilitado com QoS
- *Null Function*: Não contem dados, porém é um indicador que está havendo falhas na transmissão e ao ser usado com o campo *power management* do *frame control* definido com o valor 1 indica que a STA ou o AP irá se desconectar/desligar.
- *Association Request*: enviado por uma STA para se associar a um BSS
- *Association Response*: enviado pelo AP- em resposta a um *Association Request*
- *Reassociation Request*: enviado pelo AP requisitando uma associação a outro AP no mesmo ESS indicando um *romaing* entre APs ou uma reassociação no mesmo AP.
- *Reassociation Response*: enviado pelo AP em resposta a um *Reassociation Request*
- *Probe Request*: sondagem realizada pela STA em todos os canais disponíveis em busca de um BSS compatível. Essa mensagem é enviada em *broadcast*.
- *Probe Response*: resposta do AP a um *Probe Request*
- *Beacon*: é um *frame* enviado periodicamente pelos APs, normalmente a cada 100ms, divulgando informações sobre o BSS, como SSID, canal, etc Esse *frame* também é enviado em *broadcast*.
- *Desassociation*: enviado pelo AP para encerrar uma associação de uma estação

- *Authentication*: é um *frame* usado para uma autenticação baseada no 802.11. O padrão define 2 tipos de autenticação: *Open System* ou *Pre-Shared Key* (PSK)
- *Deauthentication*: é um *frame* que encerra a autenticação de uma estação.

#### 2.1.2.2 Fases do *handoff*

O processo de *handoff* é dividido em 4 fases, conforme ilustrado na Figura 2.6, porém a fase de detecção não é compreendida pelo protocolo IEEE 802.11. Cada uma das fases representa um processo separado que introduz um determinado atraso.

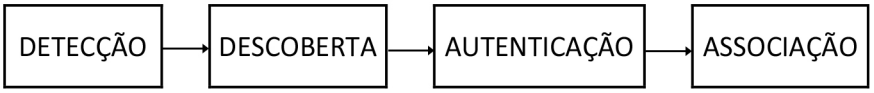



**Figura 2.6:** Fases do processo de *handoff* nas redes *Wi-Fi*

A fase de detecção é responsável por iniciar o *handoff*. As ações durante esta etapa variam dependendo de qual entidade iniciou o processo. Quando iniciada pela rede, a detecção começa pelo envio de um *disassociation frame* enviado pelo AP para a STA alertando que irá encerrar a associação. No entanto, esta fase é normalmente originada na STA ao detectar falhas na transmissão de quadros ou um baixo valor do *Received Signal Strength Indication* (RSSI), indicando um nível de energia insuficiente no sinal de rádio recebido (VELAYOS; KARLSSON, 2004). Por não ser padronizada, a fase de detecção depende da implementação de cada fabricante, gerando assim uma variação muito grande no *delay* adicionado pela mesma em todo o processo.

A fase de descoberta tem sido intensivamente estudada, uma vez que é responsável, em conjunto com a fase de detecção, por parte considerável no tempo gasto no *handoff*. A descoberta pode ser ativa ou passiva. Na descoberta passiva, a STA espera por *Beacon Frames* enviados pelos APs. Esses *frames* são enviados a cada 100 ms em cada canal. Como existem 14 canais, o tempo máximo desse tipo de varredura seria de 1400 ms (CHAN; LIN, 2014). Já na descoberta ativa, a STA envia frames do tipo *Probe Request* em broadcast varrendo todos os canais disponíveis. O padrão 802.11 especifica dois parâmetros que determinam quanto tempo uma STA deve esperar por um *Probe Response* após enviar um *Probe Request*, os quais são chamados *MinChannelTime* e *MaxChannelTime*.

*MinChannelTime* é o tempo mínimo que a STA espera em um canal enquanto que *MaxChannelTime* é o máximo. Esses valores são configuráveis, porém a maioria dos APs utiliza 20 e 40 ms (MISHRA; SHIN; ARBAUSH, 2004). Com essa estratégia, o tempo gasto para varrer os 14 canais nessa fase seria entre 280 ms a 560 ms, ainda assim tempos muito altos para aplicações multimídia.

Finalmente, o tempo gasto nas fases de Autenticação e Reassociação foi otimizado pelo IEEE com as emendas 802.11f, 802.11r, 802.11k e 802.11v e normalmente adicionam valores abaixo de 50 ms (SALIH; UDDIN; MASTORAKIS, 2015). Desta forma, a proposta desse trabalho foca ações para diminuir o tempo compreendido entre as fases de detecção e descoberta.

### 2.1.3 Emendas IEEE 802.11

Melhorar o desempenho do processo de *handoff* e adequá-lo aos requisitos de QoS das aplicações multimídia, tem sido um grande desafio para a indústria e academia ao redor do mundo. Várias soluções foram propostas ao longo dos anos, sendo algumas delas descritas a seguir.

O IEEE responsável pela padronização das redes *Wi-Fi* tem feito um esforço publicando emendas sobre o assunto. Em 2003 foi publicada a emenda 802.11f que define o protocolo de interoperabilidade entre APs – *Inter Access Point Protocol* (IAPP). Este padrão define um protocolo que implementa a troca segura de informações de contexto das estações entre o ponto de acesso atual e o ponto de acesso que a estação irá migrar durante o processo de *handoff*. Este protocolo abrange duas fases do processo: associação e reassociação (HUANG; TSENG; TSAI, 2006).

Em 2008 foi lançada a emenda 802.11r que define que ao iniciar o *handoff* o dispositivo usará um recurso chamado *Fast Basic Service Set Transition* (FT) para fazer a autenticação de forma mais rápida. Ainda em 2008, o IEEE lançou a emenda 802.11k que tem por objetivo acelerar a busca por APs próximos que estejam disponíveis criando uma lista otimizada de destinos. Quando a força do sinal do AP atual diminui, o dispositivo busca os APs na lista melhorando a eficiência da fase de descoberta.

Em 2011 o IEEE lançou a emenda 802.11v permitindo que dispositivos possam trocar informações sobre os APs próximos, incluindo dados sobre o ambiente de *Radio Frequency*

(RF) facilitando a decisão na hora de escolher a qual AP se conectar ao executar o *handoff*. Todas as emendas acima mencionadas foram incorporadas na revisão do padrão feita em 2012 (GROUP et al., 2012).

## 2.2 *Software Defined Wireless Network* (SDWN)

A Internet tem crescido de maneira muito mais rápida que o entendimento de como administrar e gerenciar redes tão grandes e com tantas aplicações diferentes. Problemas de segurança como *spam*, *phishing*, ataques de negação de serviço (DoS, do inglês *Denial of Service*), crescimento exponencial do número de dispositivos móveis, aplicações em tempo real (IPTV e vídeo conferência), IoT são grandes desafios para a arquitetura atual da Internet.

As Redes Definidas por Software estabelecem um novo paradigma que engloba vários tipos de tecnologias utilizadas na Internet visando tornar a administração da rede mais ágil e flexível. O principio que norteia este novo paradigma é a separação do plano de dados do plano de controle. Em redes tradicionais os planos de controle e de dados estão combinados em um dispositivo de rede como demonstrado na Figura 2.7.

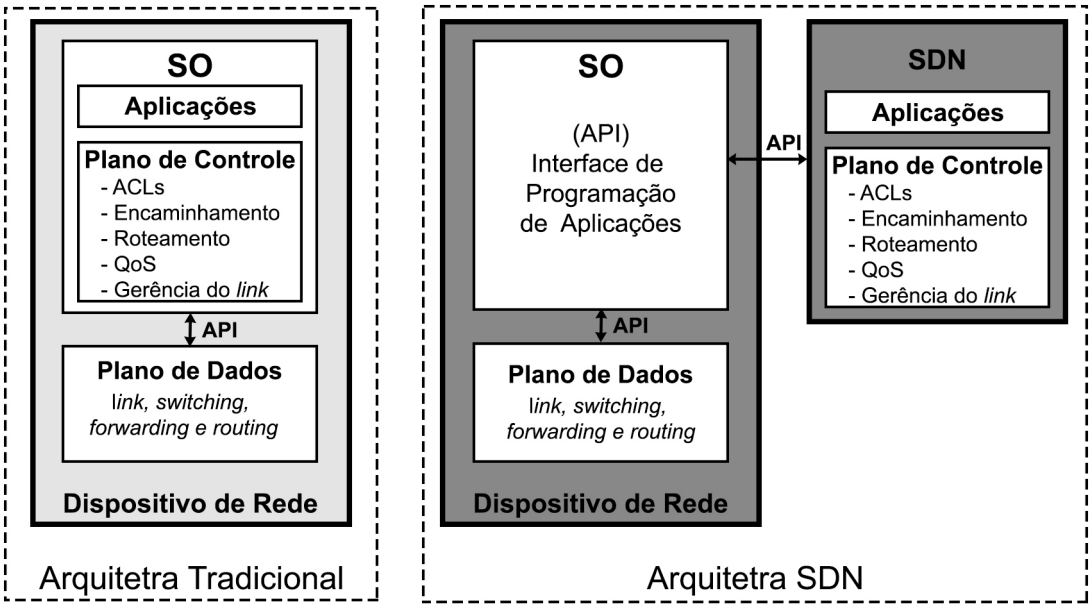



**Figura 2.7:** Arquitetura SDN x Arquitetura Tradicional. Adaptado de Sezer et al. (2013)

O plano de controle é responsável pela decisão de quais caminhos vão ser utilizados pelos fluxos de dados. Uma vez determinados, estes caminhos são passados para o plano de

dados para a transmissão em nível de *hardware*. Essas regras são chamadas de política de encaminhamento e são definidas de forma estática ou por algum protocolo de roteamento ou encaminhamento. A única maneira de fazer um ajuste nessa política é através de alterações na configuração do dispositivo ou realizando modificações nos protocolos existentes (KREUTZ et al., 2015). Estes fatos são obstáculos para os administradores de redes que necessitam de flexibilidade e rapidez em resposta às mudanças nas demandas do tráfego, bem como no aumento do uso de dispositivos móveis.

O conceito de Redes Sem Fio Definidas por Software (SDWNs) vem evoluindo rapidamente como a solução para atender à demanda por serviços dinâmicos e a computação ubíqua (HU et al., 2016). A SDWN tem sido proposta como uma solução de baixo custo e alta eficiência para o gerenciamento das redes sem fio desacoplando o plano de dados do plano de controle, permitindo programação direta em vários serviços da rede. Em uma rede com os conceitos de SDWN, é possível criar novos mecanismos adaptáveis a diferentes demandas dos usuários como mobilidade (*handoff*), segurança e QoS (HU et al., 2015). No entanto o desenvolvimento de soluções SDN visando gerenciar uma rede *wireless* é fundamentalmente mais difícil pela complexidade dos mecanismos necessários para efetuar o controle de acesso, manter o sigilo e a integridade dos dados e garantir a mobilidade de forma suave e transparente enquanto que em uma rede cabeada as soluções SDN visam predominantemente ações de *switching*.

### 2.2.1 *OpenFlow*

O protocolo *OpenFlow* é a base de uma SDN. Foi o primeiro padrão a definir uma interface entre os planos de dados e controle para essa arquitetura. O desenvolvimento do *OpenFlow* começou em 2007, e hoje tem uma colaboração entre o mundo acadêmico e o mundo comercial. Conduzido originalmente pela Universidade de *Stanford* e pela Universidade da Califórnia em *Berkeley*, o padrão agora está sendo definido pela *Open Networking Foundation* (ONF)[1].

Um *switch OpenFlow* é um *switch* com suporte ao protocolo *OpenFlow*. Sua principal

[1]https://www.opennetworking.org/

característica é uma tabela com fluxos especificados através de campos de alguns protocolos da pilha TCP/IP como arp, ip, tcp e udp para a realização de correspondências com os pacotes recebidos a fim de executar o encaminhamento adequado dos mesmos. Um canal seguro é estabelecido com um controlador externo como demonstrado na Figura 2.8. O controlador gerencia o *switch* através do canal seguro usando o protocolo *OpenFlow*.

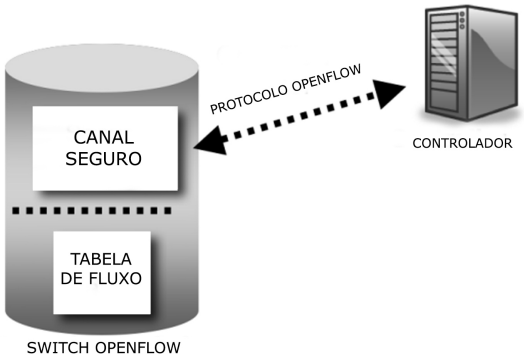



**Figura 2.8:** *Switch* e Controlador *OpenFlow*

*OpenFlow* e SDN são termos muitas vezes utilizados em conjunto. Como demonstrado na Figura 2.9, SDN é a nova arquitetura proposta para redes onde a inteligência, localizada no plano de controle, fica fora do núcleo das mesmas. Já o *OpenFlow* é o protocolo usado entre o controlador externo e os dispositivos da rede: *switches*, roteadores e *access-points* (TIWARI, 2013). Segundo McKeown et al. (2008), o objetivo inicial do *OpenFlow* é prover uma maneira para que os pesquisadores ao redor do mundo possam executar protocolos experimentais nas redes de uso diário, sem prejuízos para o ambiente de produção. O *OpenFlow* foi desenvolvido tendo como base um *switch ethernet* com uma tabela de fluxo e uma *interface* padronizada para gerenciar as entradas bem como as ações relacionadas a cada fluxo.

Quando uma rede SDN começa a operar, todos os seus dispositivos estão com as suas tabelas de fluxo vazias. Quando o equipamento que serve de porta de entrada na rede (ex: *switch*) recebe um pacote, a princípio ele não sabe que decisão tomar, uma vez que sua tabela de fluxo está vazia. O *switch* então, repassa o pacote para o controlador utilizando o protocolo *OpenFlow*. O controlador por sua vez, baseado nos campos do cabeçalho do pacote e de regras já definidas, decide o que fazer com aquele pacote. Alguns exemplos de decisões são: descartar o pacote; encaminhar o pacote para uma interface de saída específica; realizar

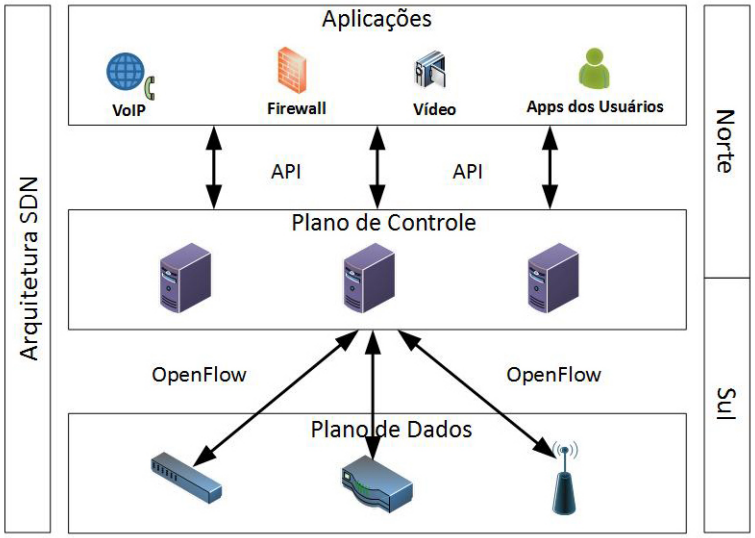



**Figura 2.9:** SDN e *OpenFlow*. Adaptado de Tiwari (2013)

*broadcast* do pacote; dentre outras. Uma vez tomada a decisão, o controlador, também através do *OpenFlow*, cria uma entrada na tabela de fluxo do *switch* e devolve o pacote. O *switch* então realiza a operação que está designada na tabela de fluxo. A partir daí, sempre que chegar um pacote com as mesmas características, o *switch* não precisa mais consultar o controlador. Ele apenas realiza uma busca em sua tabela de fluxo para encontrar a entrada apropriada e consequentemente a ação a ser realizada.

Neste trabalho, o *OpenFlow* v1.0.0 (OPEN NETWORK FOUNDATION, 2009) foi utilizado para gerenciar remotamente as tabelas de fluxo dos *Access Points*. Nesta versão é utilizado um conjunto de 12 campos dos protocolos Ethernet, IP, TCP e UDP, como mostrado na Figura 2.10.

| Ingress Port | Ether Src | Ether Dst | Ether Type | VLAN ID | VLAN Priority | IP src | IP dst | IP proto | IP ToS | TCP/UDP src port | TCP/UDP dst port |
|---|---|---|---|---|---|---|---|---|---|---|---|

**Figura 2.10:** Campos dos pacotes utilizados

A seguir são descritos cada campo do cabeçalho.

- *Ingress Port*: porta de entrada do *frame* no *switch*.
- *Ether src*: endereço MAC de origem do cabeçalho *ethernet*.
- *Ether dst*: endereço MAC de destino do cabeçalho *ethernet*.

- *Ether type*: campo *type* do cabeçalho *ethernet* que define o tipo do *payload*.
- *VLAN id*: identificador de VLAN definido pelo 802.11q.
- *VLAN prority*: define a prioridade do *frame* de acordo com o 802.11q.
- *IP src*: campo IP de origem do cabeçalho IP.
- *IP dst*: campo IP de destino do cabeçalho IP.
- *IP proto*: campo *protocol* do cabeçalho IP que define o *payload* do pacote IP.
- *IP ToS*: campo *ToS* do cabeçalho IP que define a prioridade no encaminhamento do pacote.
- *TCP/UDP src port*: corresponde a porta de origem do cabeçalho TCP ou UDP quando o campo IP proto assumir os valores 6 e 17 respectivamente. Caso o campo IP proto for igual a 1, ou seja, o protocolo ICMP, este campo representará o ICMP *type*.
- *TCP/UDP dst port*: corresponde a porta de destino do cabeçalho TCP ou UDP quando o campo IP proto assumir os valores 6 e 17 respectivamente. Caso o campo IP proto for igual a 1, ou seja, o protocolo ICMP, este campo representará o ICMP *code*.

Após a recepção de um pacote, um *switch OpenFlow* executa as ações mostradas na Figura 2.11. Ao receber um *frame* em uma porta específica, o *switch* faz uma consulta às suas tabelas tentando uma correspondência (*match*) com os valores dos campos dos cabeçalhos dos fluxos recebidos.

Caso um fluxo corresponda a um fluxo já instalado o *switch* executará uma ação correspondente. Para todos os pacotes que não tenham um fluxo de entrada correspondente, uma mensagem *packet-in* é enviada para o controlador. Se o *switch* tem memória suficiente para armazenar os pacotes que são enviados para o controlador, as mensagens *packet-in* podem conter uma parte do cabeçalho (por padrão 128 *bytes*) e um *buffer ID* que será utilizado pelo controlador quando estiver pronto para encaminhar o pacote de volta ao *switch*. *Switches* que não possuem memória suficiente, ou ficam momentaneamente sem espaço para o *buffer* interno enviam o pacote completo para o controlador como parte do evento (OPEN NETWORK FOUNDATION, 2009). Ao descrever uma entrada de fluxo a estrutura descrita na Tabela 2.3 deverá ser usada. Quando um dos campos não for declarado com um valor

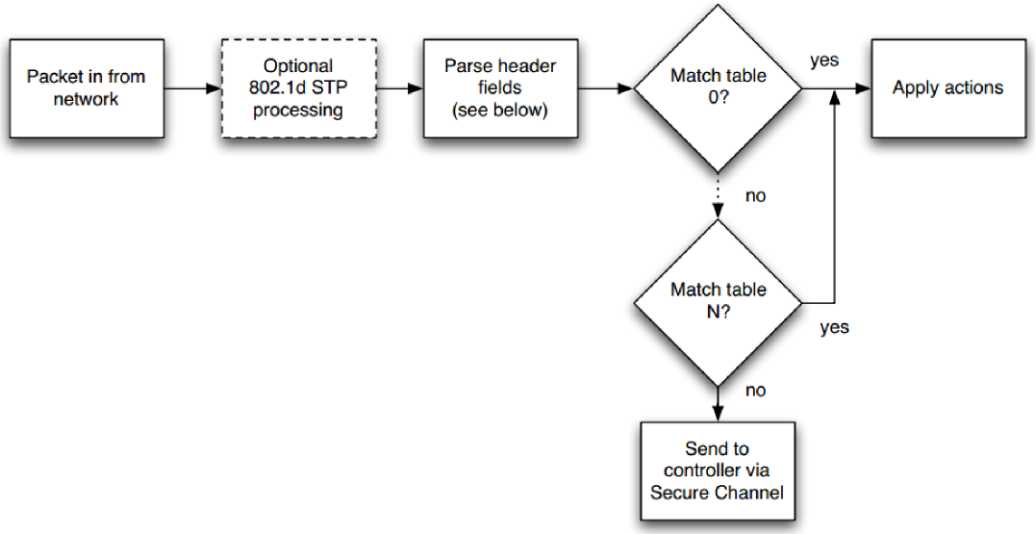



**Figura 2.11:** Fluxo do pacote em um *switch OpenFlow*

específico, este assume o valor * que funciona como um coringa representando todos os valores possíveis.

**Tabela 2.3:** Descrição dos campos utilizados em uma entrada de fluxo

| **Campo** | **Protocolo** | **Descrição** |
|---|---|---|
| *in_port* | - | Porta física de entrada do *switch* |
| *dl_src* | *Ethernet* | MAC de origem |
| *dl_dst* | *Ethernet* | MAC de destino |
| *dl_vlan* | *Ethernet* 802.1q | *Vlan ID* |
| *dl_vlan_pcp* | *Ethernet* 802.1q | *Vlan priority* |
| *dl_type* | *Ethernet* | Tipo do *payload ethernet* |
| *nw_tos* | IP | Campo TOS do cabeçalho |
| *nw_proto* | IP | Campo *protocol* do cabeçalho IP |
| *nw_src* | IP | IP de origem |
| *nw_dst* | IP | IP de destino |
| *tp_scr* | TCP ou UDP | porta de origem |
| *tp_dst* | TCP ou UDP | porta de destino |

Ao ser enviado de volta para ser instalado no *switch*, um fluxo deve conter certa ação a ser executada pelo mesmo com todos os fluxos correspondentes. O protocolo *OpenFlow v 1.0* define as seguintes ações:

- output: encaminha o pacote para uma porta de saída do *switch*;
- *set_vlan_vid*: define o campo *vlan* id do 802.1q;

- *set_vlan_pcp*: define o campo *priority code point* do 802.1q;
- *set_dl_src*: define o endereço mac de origem;
- *set_dl_dst*: defne o endereço mac de destino;
- *set_nw_src*: define o endereço IP de origem;
- *set_nw_dst*: define o endereço IP de origem;
- *set_nw_tos*: define o campo tos do cabeçalho IP;
- *set_tp_src*: define a porta de origem no cabeçalho TCP ou UDP;
- *set_tp_dst*: define a porta de destino no cabeçalho TCP ou UDP;
- *enqueue*: encaminha para uma fila.

## 2.3 SNMP

O Simple Network Management Protocol (SNMP) é um protocolo desenvolvido pela *Internet Engineering Task Force* (IETF) com objetivo de gerenciar dispositivos em redes baseadas na pilha de protocolos TCP/IP. Implementado na camada de aplicação o uso do protocolo define uma arquitetura que leva o seu nome e é formada pelos seguintes componentes: Agente, Gerente, *Management Information Base* (MIB) e o protocolo SNMP. Esta arquitetura está demostrado na Figura 2.12.

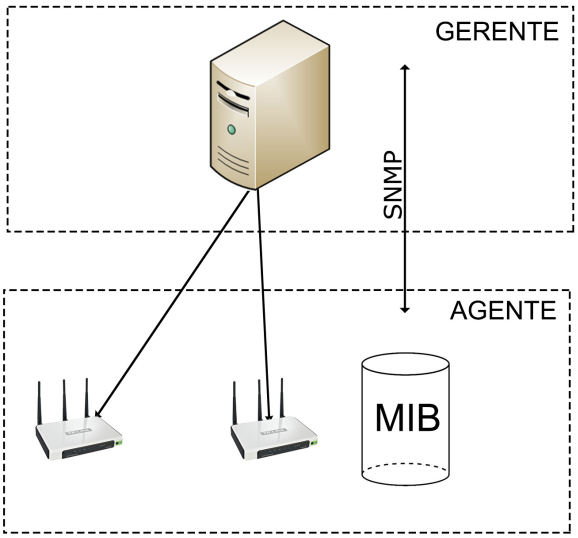



**Figura 2.12:** Arquitetura SNMP

Como todos os serviços definidos na pilha TCP/IP, o SNMP usa a arquitetura cliente/-servidor porém o *software* cliente está implementado no gerente enquanto que o *software* servidor está sendo executado no módulo agente. Desta forma ao instalar um agente SNMP em um dispositivo, por padrão a porta udp 161 fica no modo *listening* aguardando as solicitações do gerente. Apesar desse procedimento, existem situações em que o agente envia mensagens sem que o gerente tenha solicitado. Essas mensagens são denominadas de *trap* e o gerente as recebe na porta udp 162 assumindo assim o papel de servidor na comunicação.

A MIB é uma base de dados quem contem um conjunto de objetos que poderão ser gerenciados. Essa coleção de objetos está organizada em um banco de dados hierárquico estruturado em árvore onde cada entrada é endereçada através de um identificador de objeto (OID, do inglês *Object Identifier*). A árvore MIB está demonstrada na Figura 2.13.

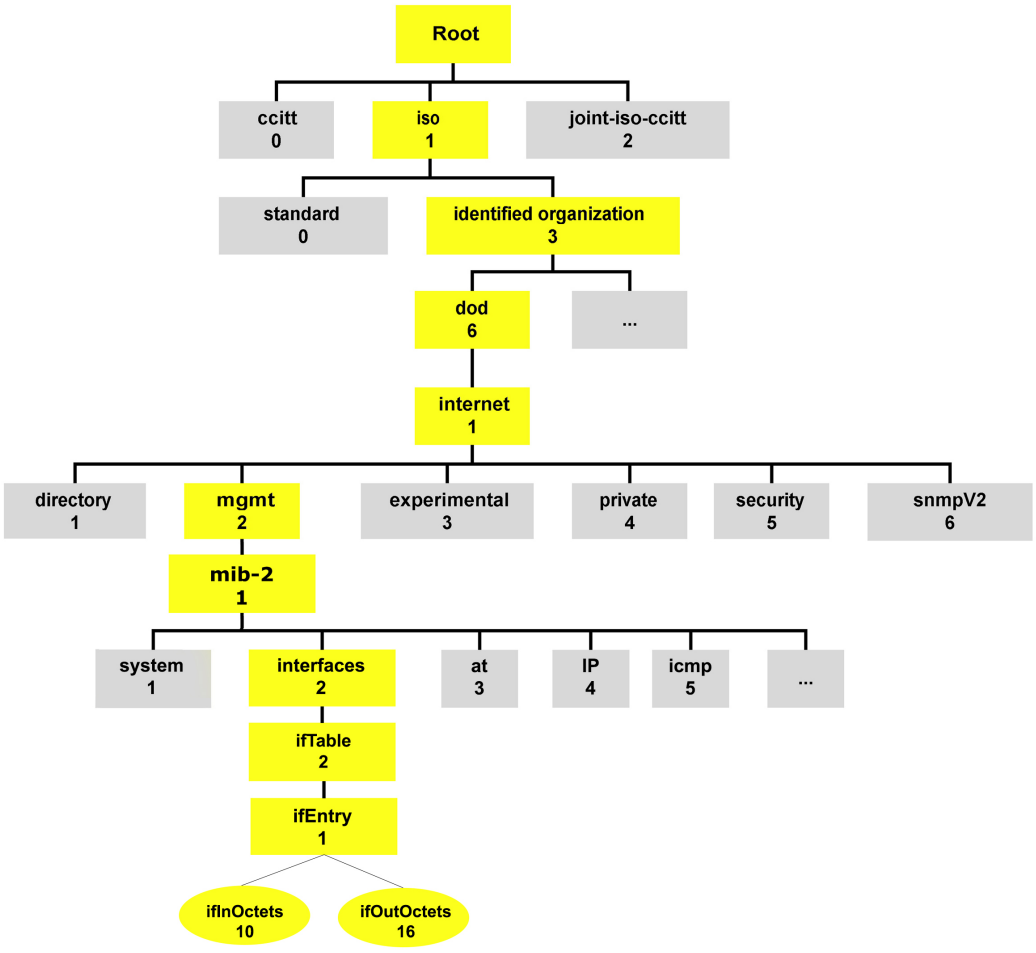



**Figura 2.13:** Management Information Base

Dois objetos foram utilizados neste trabalho para medir a quantidade de tráfego na interface dos APs: *ifInOctets* e *ifOutOctets*. Estes objetos significam os *bytes* que chegam e saem da interface respectivamente. Estão localizados na subárvore interfaces e estão representados pelos seguintes OIDs:

- *ifInOctets*: .1.3.6.1.2.1.2.2.1.10
- *iOutOctets*: .1.3.6.1.2.1.2.2.1.16

## 2.4 Trabalhos Relacionados

No âmbito acadêmico, muitos trabalhos também exploram este tema, buscando alternativas para reduzir o tempo de *handoff*. Em Misal e Sambare (2014) e Mishra, Shin e Arbaush (2004) os autores propõem reduzir o tempo do *handoff* utilizando esquemas que identifiquem o posicionamento da STA com relação aos APs para que, dessa maneira, se possa fazer um escaneamento seletivo baseado na sua localização. Em redes *Wi-Fi*, quando o processo de *handoff* se inicia, há tradicionalmente uma interrupção na troca de dados. Pensando nisto, os autores Sanghavi e Bansode (2015), Chan e Lin (2014), Jin e Choi (2014) e Brik, Mishra e Banerjee (2005) equiparam os APs com duas *interfaces Wi-Fi* com o objetivo de manter a troca de dados enquanto a outra *interface* realiza a fase de descoberta. No entanto, normalmente os APs existentes no mercado são equipados com apenas uma *interface*, fazendo com que tais abordagens tenham seu uso muito limitado.

Conforme já mencionado, o tempo gasto na fase de descoberta ao se utilizar um escaneamento ativo depende dos valores *MinChannelTime* e *MaxChanelTime*. Apesar disso o IEEE não padronizou esses valores.Velayos e Karlsson (2004) propõem um cálculo matemático para otimizar estes valores a fim de reduzir o tempo gasto nessa fase.

Todos os trabalhos supracitados são baseados em soluções implantadas no *core* da rede, seja nos APs ou nas STAs. Em contrapartida, a arquitetura baseada em SDN separa o plano de controle (*software*) do plano de dados (*hardware*) retirando as decisões do núcleo da rede. Dessa forma, a inteligência da rede é implementada em um controlador externo que toma decisões baseadas em regras especificadas pelo administrador da rede, permitindo assim uma grande flexibilidade no gerenciamento da rede. A comunicação com os dispositivos é

feita através do protocolo *OpenFlow* (MCKEOWN et al., 2008).

Dely et al. (2013) apresentam uma arquitetura baseada em SDN que permite que as estações se associem a vários APs simultaneamente permitindo assim uma troca rápida entre eles. Para que isso seja possível foram utilizadas estações com duas interfaces WLAN uma dedicada apenas para o escaneamento de novos APs. Na outra interface foram criadas várias interfaces virtuais que podem se conectar a vários APs simultaneamente fazendo com que o tráfego de dados sofra o mínimo de interrupção no momento do *handoff*. O problema dessa proposta é sua aplicação prática, pois nem todos os drivers das placas de rede permitem a criação de interfaces virtuais.

O trabalho de (PAIVA et al., 2014) propõe uma aplicação, chamada de *HandoffSDN*, que não apenas identifica o processo de *handoff* em si, mas também aloca, realoca e desaloca recursos, de forma transparente ao usuário a fim de prover o QoS necessário às aplicações. Essa proposta utiliza as mensagens *DHCPREQUEST* do protocolo DHCP enviadas pela STA em movimento para detectar que houve uma associação a um novo AP se limitando apenas a identificar que houve o processo de *handoff*, contundo sem interferir no mesmo.

Moura et al. (2015) apresentam Ethanol, uma arquitetura aberta que permite a criação de algoritmos de controle sob medida para as necessidades dos usuários de redes locais sem fio onde o controlador pode administrar a mobilidade das estações, a criação de uma rede virtual, o QoS e, até mesmo, a autenticação e localização do usuário. Apesar de propor mecanismos de controle de novas associações, essa arquitetura não atua nas fases de detecção e descoberta que são os grandes gargalos do processo de *handoff*.

Uma abordagem em que os autores não utilizaram a arquitetura SDN foi apresentada por Sanghavi e Bansode (2015). Nesta solução a decisão de qual será o AP de destino no momento do *handoff* é executada pelo AP atual através de mensagens trocadas com os APs ao redor que estão equipados com múltiplas interfaces de rede. Os testes para validação da proposta foram realizados em ambientes não reais com o simulador NS2.

Yan et al. (2015), Utiliza os níveis de RSSI percebidos pela STA a fim de prever para qual AP deverá ser realizado o *handoff*. Dessa forma o controlador modifica as tabelas de fluxo dos APs passando a enviar em *multicast* todos os pacotes com destino a estação móvel tanto ao AP atual como para o AP previsto garantindo que o cliente possa receber pacotes imediatamente após ocorrer a reassociação com o novo AP minimizando assim a perda de

pacotes durante o *handoff*. A fase de detecção está fora do escopo deste trabalho.

Sun e Qian (2016b) propõem um algoritmo que utiliza a potência do sinal recebido (RSSI), previsão do RSSI e a largura de banda disponível como parâmetros para, baseado em uma lógica *fuzzy*, melhorar o processo de *handoff*. A decisão de iniciar o processo ainda fica sob responsabilidade da estação ao passo que nesta proposta as decisões de detecção e descoberta estão concentradas no controlador externo.

Luengo (2016) utiliza a arquitetura SDN para obter dados estatísticos de utilização da rede, bem como, informações do número de estações conectadas nos AP com a finalidade de detectar a carga de tráfego nos APs e utiliza essas informações para a escolha do AP de destino no momento da *handoff* a fim de realizar um melhor balanceamento de carga entre os APs próximos. Essencialmente, ele monitora a rede periodicamente em busca de assimetria rede. Uma vez que uma condição desequilibrada surge, o WUMS determina a quantidade de carga que deve ser redistribuído, a fim de neutralizar o efeito. No entanto, movendo-se a carga excessiva em um meio WLAN migrar os usuários sem fio que estão causando o problema. Assim, os WUMS seleciona usuários sem fio no AP-over carregado e, consequentemente, migra para outra sub-carregado AP.

# Capítulo 3

# DISpatCH

Neste capítulo será apresentado o *Detection and dIScovery Control in Handoff* (DISpatCH) uma solução para o processo de *handoff* nas redes *Wi-Fi* baseada na arquitetura SDWN.

## 3.1 Arquitetura

Como já mencionado, as fases de detecção e descoberta são as principais responsáveis pelo atraso do processo de *handoff*. Portanto, este trabalho se concentrou nessas duas fases. Em DISpatCH, um controlador gerencia as tabelas de fluxo nos APs e um gerente SNMP/RSSI monitora constantemente a rede para tomar duas decisões importantes: (i) quando iniciar o processo de transferência (fase de detecção) e; (ii) a que AP o STA deve ligar (fase de descoberta).

A Figura 3.1 apresenta a arquitetura DISpatCH. O DISpatCH é composto por quatro módulos: o Agente SNMP, o Agente RSSI, o Gerente SNMP/RSSI e o Controlador. O Agente RSSI reside na STA e tem duas funções. Em primeiro lugar, envia periodicamente ao Gerente RSSI os RSSIs percebidos pela STA de todos os APs dentro de sua área de cobertura. Em segundo lugar, força a STA a se conectar ao AP escolhido pelo Gerente. O Gerente SNMP por sua vez solicita periodicamente a quantidade de tráfego nas interfaces *Wi-Fi* dos APs ao alcance.

No servidor reside o módulo Gerente que se comunica tanto com o agente RSSI instalado na STA como com o agente SNMP instalado nos APs requisitando informações sobre o RSSI perecebido pela STA e a quantidade de tráfego de todos os APs ao alcane. Esses valores

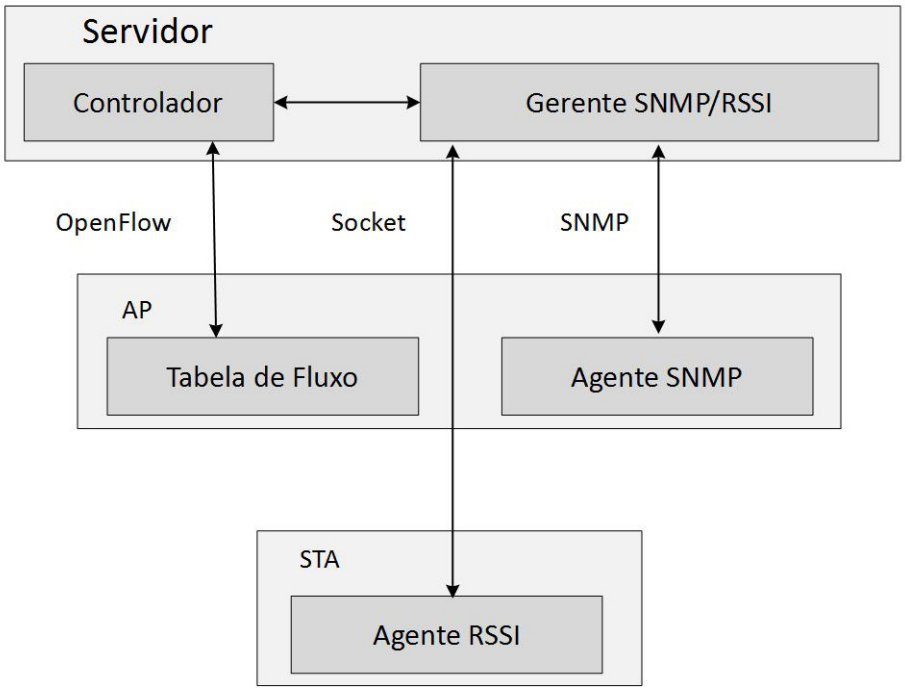



**Figura 3.1:** Arquitetura DISpatCH

são utilizados para definir o momento de iniciar o *handoff* bem como escolher para qual AP a STA deve migrar. Um módulo Controlador também é executado no servidor e tem a finalidade de gerenciar as tabelas de fluxos nos APs através do protocolo *OpenFlow*. O controlador utilizado foi o POX e o *OpenFlow* na versão 1.0. Ao tomar a decisão de iniciar o processo, o Gerente comunica ao Controlador que instala os fluxos correspondentes no AP de destino indicado e então o processo de *handoff* é iniciado.

## 3.2 Funcionamento

As informações dos RSSIs recebidos e o tráfego nas interfaces dos APs são trocadas até o instante em que o RSSI do AP de origem atinge o limiar de -70 dBm. Ao atingir esse valor o Gerente escolhe o AP de destino e informa ao Controlador que o processo de *handoff* deve ser iniciado para este AP. Em seguida o Controlador instala os fluxos referentes a STA no AP de destino indicado pelo Gerente e informa ao mesmo que os fluxos foram instalados. Nesse momento o Gerente envia para a estação um comando para que a mesma se desconecte do AP atual e conecte-se no novo AP. A partir desse momento o processo de *handoff* é executado de acordo com o padrão IEEE 802.11.

Para esclarecer como funciona o DISpatCH, a Figura 3.2 apresenta um diagrama de seqüência que ilustra todas as principais atividades envolvidas no processo de tomada de decisão realizado pelo Gerente e pelo Controlador. Neste trabalho, assume-se como AP de origem e AP de destino o AP em que o STA está atualmente conectado e o AP ao qual a STA se conectará, respectivamente.

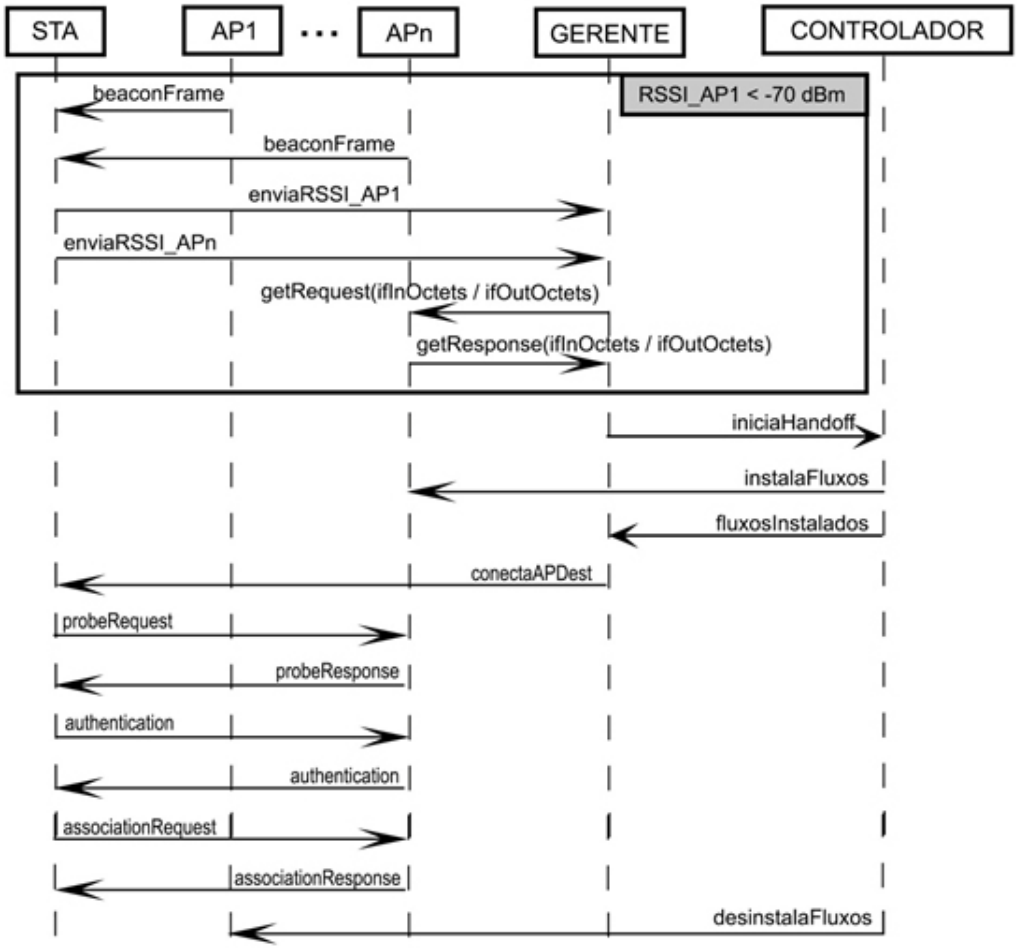



**Figura 3.2:** DISpatCH - Diagrama de Sequência e principais operações

Como definido pelo IEEE 802.11, por padrão os APs transmitem continuamente quadros *beaconFrame* que são capturados pela STA. Na STA, o Agente RSSI extrai os valores RSSI de todos os APs ao alcance e os envia para o Gerente (enviaRSSI). O Gerente compara o RSSI do AP de origem com um limiar definido (checarRSSI). Ao atingir esse limiar a troca de AP deverá ser iniciada. Esta operação caracteriza a fase de detecção do processo de *handoff*. O Gerente também solicita constantemente informações sobre o trafego nas interfaces *Wi-Fi* dos APs (getRequest). Essas informações de tráfego são necessárias para a

seleção do AP de destino definido assim a fase de descoberta do processo de *handoff*.

Todas essas atividades são continuamente executadas até que o limite de RSSI seja alcançado. Neste ponto, considerando que APo é o AP de origem, APd é o AP de destino e L é o limiar definido, o Gerente então verificará as seguintes condições:

- RSSI(APo) > L: Nesse caso, o Controlador não toma nenhuma ação, uma vez que o RSSI do AP de origem é forte o suficiente para manter a qualidade da conexão;
- RSSI (APo) <= L e RSSI(APo) > = RSSI (APd): Neste caso, o RSSI do AP de origem atingiu o limiar, no entanto, não há outro AP ao alcance com um RSSI mais forte. Assim, o controlador não toma nenhuma ação também;
- RSSI (APo) <= L e RSSI(APo) < RSSI (APd): Neste caso, o RSSI do AP de origem atingiu o limiar definido e há um AP na vizinhança com um RSSI mais forte do que o RSSI do AP de origem. Portanto, o Gerente decide que o processo de *handoff* deve ser iniciado. Esse mecanismo corresponde à fase de detecção.

Após o Gerente decidir que o processo de *handoff* deve ser iniciado (detecção), ele deve agora escolher o AP de destino, ou seja, a qual AP a STA deverá conectar (descoberta). Se houver apenas um AP de destino possível, então a escolha é óbvia, caso contrário, o Gerente usa alguns parâmetros para decidir. A abordagem mais simples é comparar os valores RSSI próprios e selecionar o AP com o valor RSSI mais alto. No entanto, neste trabalho, também é utilizada a quantidade de tráfego como um parâmetro de decisão para o processo de seleção. Uma vez em posse de tais informações, o Gerente seleciona o AP de destino com base na quantidade de tráfego (checarTráfegoAP). Considerando TR como o tráfego medido em um intervalo de tempo, MAX TR como o tráfego máximo permitido, AP1 e AP2 como dois APs de destino possíveis e APx como qualquer outro destino AP possível, o Gerente pode encontrar duas situações:

- RSSI(AP1) >= RSSI(AP2) e RSSI(AP1) >= RSSI(APx) e TR(AP1) <= MAX TR: Neste caso, o RSSI do AP1 é melhor do que o RSSI de qualquer outro AP ao alcance incluindo AP2 e a quantidade de tráfego em AP1 é menor ou igual ao tráfego máximo permitido. Assim, o AP1 é selecionado como o AP de destino.

- RSSI(AP1) >= RSSI(AP2) e RSSI(AP1) >= RSSI(APx) e TR(AP1) > MAX TR >= TR(AP2) e TR(AP2) < TR(APx): Neste caso , O RSSI do AP1 é também superior ao RSSI de qualquer outro AP, no entanto, a sua quantidade de tráfego é maior do que o tráfego máximo permitido. Devido a isso, o Gerente seleciona como o AP de destino aquele com a menor quantidade de tráfego.

Com o AP de destino selecionado pelo Gerente, o mesmo informa ao Controlador para iniciar o processo de *handoff* (iniciaHandoff) para o AP escolhido. O Controlador por sua vez instala os fluxos referentes a STA no AP de destino indicado pelo Gerente e em seguida informa ao Gerente que os fluxos foram instalados (fluxosInstalados). A decisão de instalar os fluxos antes do processo de *handoff* se iniciar tem como objetivo não causar interrupção na transmissão de dados entre a STA e o AP de origem. Após os fluxos estarem instalados no AP de destino o Gerente envia para a STA um comando para a mesma se conectar no novo AP (conectaApDestino). Para migrar para o AP de destino, o Agente RSSI executa o comando "iwconfig wlan0 essid SSIDname", onde SSIDName é o SSID do AP de destino. Basicamente, este comando resulta em duas ações. Primeiro, ele força o STA a enviar um quadro de Deauthentication indicando que ele está se desconectando do AP de origem. Em segundo lugar, o comando obriga o STA a fornecer um quadro de *probe request* especificando o SSID do AP de destino. Portanto, somente este AP responde com o quadro *Probe Response*. Em seguida, os quadros de *handoff* restantes são trocados, completando assim a associação da STA no AP de destino. As tabelas de fluxo são instalados com um parâmetro de *timeout* de forma que o Controlador ao perceber que já não há mais tráfego referente a STA no AP de origem o mesmo desinstala as tabelas de fluxo (desinstalaFluxos).

Usando DISpatCH, a responsabilidade em decidir quando iniciar o processo de textithandoff (fase de detecção) e a qual AP a STA deve se conectar a (fase de descoberta) é transferida para o Gerente. Isto resulta em duas vantagens principais. Primeiro, aumenta a flexibilidade na gestão da rede. Em segundo lugar, torna a rede menos dependente do fornecedor.

## 3.3 Tráfego Estimado nos APs

Um dos parâmetros que o DISpatCH leva em consideração para selecionar o AP de destino é a quantidade de tráfego em cada AP. Para calcular esses valores, são usadas consultas via

protocolo SNMP a dois objetos da *Management Information Base* (MIB) localizada em cada AP, que são: (i) ifInOctets, que especifica o número total de *bytes* que chega à *interface* e; (ii) ifOutOctets, que especifica o número total de *bytes* que são transmitidos a partir da *interface*. Os valores desses dois objetos são cumulativos a partir do momento em que o agente SNMP é iniciado e nos APs utilizados nos experimentos esses valores são atualizados a cada 15 segundos. Devido a isso, a cada 15 s as informações de tráfego são solicitadas e a quantidade de tráfego é calculada. Assim, assume-se que o tráfego corrente no AP são os valores de *bytes* atuais informados menos os últimos valores informados, de acordo com a Equação (3.1), em que:

$$TU = \frac{TRin(Ct) + TRout(Ct) - TRin(Lt) - TRout(Lt)}{T(Ct) - T(Lt)} \tag{3.1}$$

- TR: Quantidade total de tráfego na *interface* (B/s);
- Ct: requisição de octetos atuais;
- Lt: última requisição de octetos;
- TRin: Valor do objeto ifInOctets;
- TRout: Valor do objeto ifOutOctets
- T: Hora da requisição

# Capítulo 4

# Experimentos e Resultados

Para a realização dos experimentos foi montado um *testbed* no 1º andar do prédio da Unidade Acadêmica de Informática (UAI) no Instituto Federal de Educação Ciência e Tecnologia da Paraíba (IFPB) conforme ilustrado na Figura 4.1.

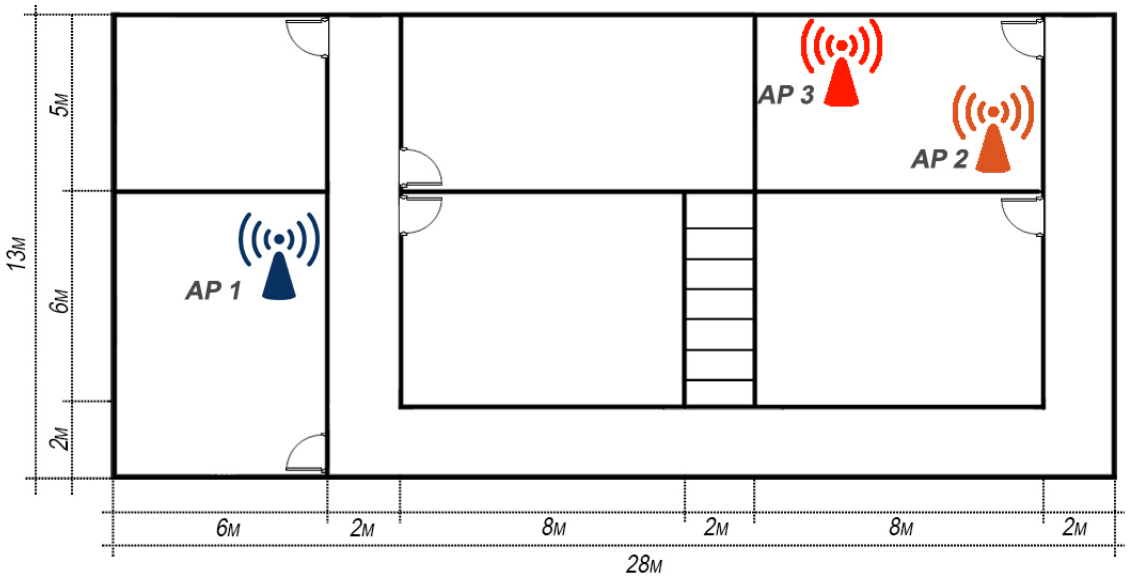



**Figura 4.1:** *Testbed*

Para validar a presente proposta os experimentos foram divididos em 2 fases. Em primeiro lugar a abordagem proposta foi testada utilizando apenas o valor do RSSI percebido pela estação e usado pelo controlador como parâmetro para a fase de detecção bem como para indicar qual o AP a STA deve se conectar. Na segunda fase, além do RSSI, foi adicionado o tráfego nos APs como mais um parâmetro melhorando assim a capacidade de decisão do controlador no momento da descoberta. O objetivo desse parâmetro é manter o nível

de QoS da aplicação. As métricas utilizadas foram o tempo gasto no *handoff*, o *jitter* e o percentual de pacotes perdidos.

Ao decidir qual o AP de destino o controlador instala via *OpenFlow* os fluxos correspondentes antes de indicar para a STA que o *handoff* deva ser iniciado. Como consequência o tempo consumido por este processo não influencia no tempo do *handoff*.

### 4.0.1 Dispositivos utilizados: *Openwrt+OpenFlow*

Neste trabalho foi decidido utilizar dispositivos em cenários reais para testar e comprovar as soluções propostas. Essa decisão teve como consequência imediata a necessidade da utilização de *switches* e *access points* habilitados com o protocolo *OpenFlow*. Ocorre que esses equipamentos não são de fácil acesso, principalmente para as redes *Wi-Fi*. A solução então foi utilizar APs comerciais facilmente encontrados no mercado e trocar o seu *firmware* para o *Openwrt*[1], um *firmware* baseado em Linux desenvolvido para vários sistemas embarcados.

O projeto *Pantou*[2] elaborado por pesquisadores da universidade de Stanford recompilou esse *firmware* adicionando o protocolo *OpenFlow* na sua versão 1.0 para 2 dispositivos específicos: TP-LINK 1043ND e Linksys WRT54GL, transformando esses roteadores sem fio em *switches OpenFlow enable*.

Este trabalho utilizou 3 dispositivos TP-LINK WR1043ND Versão 1.8 e o *firmware Openwrt* na versão *Attittude Adjustment 12.09* pré-compilada adicionando a versão 1.0 do *OpenFlow*.

## 4.1 Métricas

Para a medição do tempo do *handoff* foi utilizado o software *iperf3*, que simulou o envio de tráfego UDP a uma taxa de 1 Mbps entre a STA e um servidor localizado na rede cabeada. Para medir o tempo gasto em todo o processo foi considerado o seguinte: TEMPO DO HANDOFF = TFinal – TInicial, onde TInicial é o tempo do último pacote UDP transmitido pela STA ao servidor estando a STA ainda associada ao AP origem e TFinal é o tempo

[1] www.openwrt.org

[2] http://archive.openflow.org/wk/index.php/Pantou_:_OpenFlow_1.0_for_OpenWRT

do primeiro pacote UDP transmitido pela STA ao servidor estando a STA já associada ao AP de destino. Enquanto eram realizados os deslocamentos pelo *testbed* o tráfego foi capturado para posterior avaliação utilizando o software *Wireshark* conforme demonstrado na Figura 4.2.

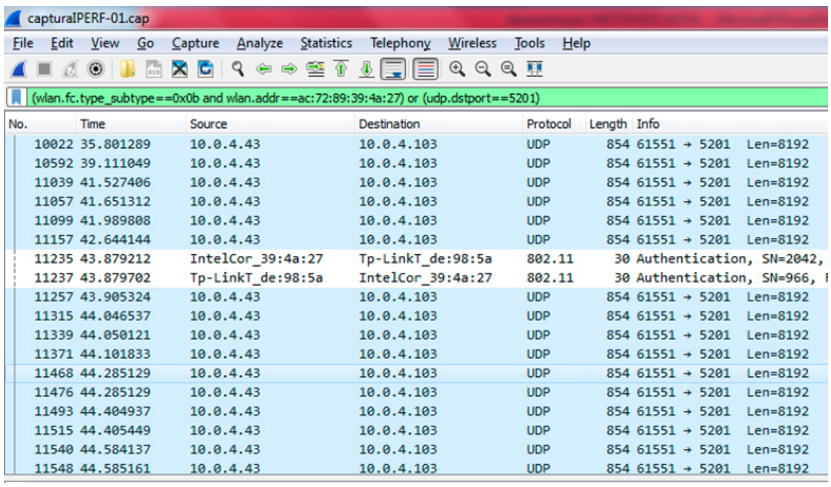



**Figura 4.2:** Análise do tráfego com Wireshark

Nesse exemplo o TInicial corresponde a 42,64 s e o TFinal a 43,90 s o que resulta no tempo de *handoff* em 43,90 - 42,64 = 1,26 s.

## 4.2 Primeira fase de testes: RSSI como parâmetro

O objetivo desta fase foi comprovar a eficiência da abordagem proposta utilizando apenas o RSSI para definir o momento da troca e a escolha do AP de destino utilizando para isso apenas 2 APs. A métrica utilizada neste caso foi o tempo gasto no processo de *handoff*.

Para poder comparar os resultados foram realizados dois testes da seguinte forma. O primeiro com o *handoff* ocorrendo de forma tradicional sendo iniciado pela STA. Já o segundo, foi usada a abordagem SDWN proposta neste trabalho.

Os experimentos se iniciaram com a STA conectada ao AP origem. Um deslocamento, então, foi realizado no sentido do AP destino. Ao se aproximar deste último o *handoff* foi executado e o deslocamento foi realizado no sentido contrário para que a STA se reconectasse ao AP origem conforme demonstrado na Figura 4.3.

Dez deslocamentos foram realizados no *testbed* sendo 5 no sentido AP origem $\rightarrow$ AP destino e os outros 5 no sentido AP destino $\rightarrow$ AP origem.

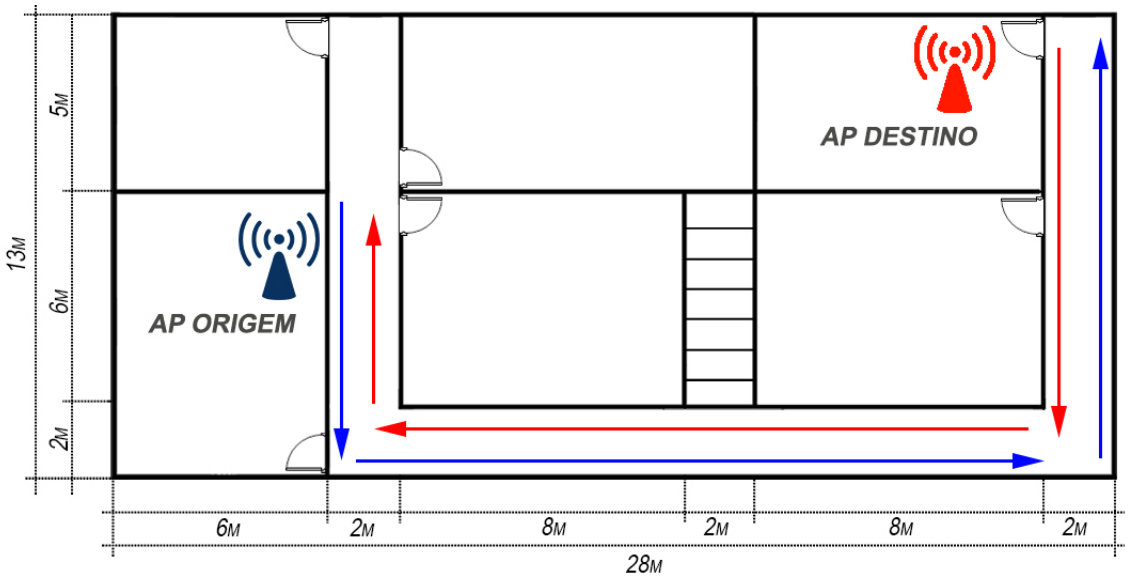



**Figura 4.3:** Deslocamentos na primeira fase dos testes

Os APs tiveram a sua potência de transmissão reduzida para 10 dBm (10 mW) com o objetivo de adequar a força do sinal às distâncias do *testbed*. Esta calibragem foi necessária para executar a primeira parte do experimento, onde o processo de *handoff* foi iniciado pela STA que, ao se afastar de um determinado AP, o valor do RSSI recebido ficaria abaixo de um limiar suficiente para dar início ao processo. O sinal então foi medido utilizando o *software Inssider v3* por toda extensão do ambiente de testes conforme pode ser visto na Figura 4.4.

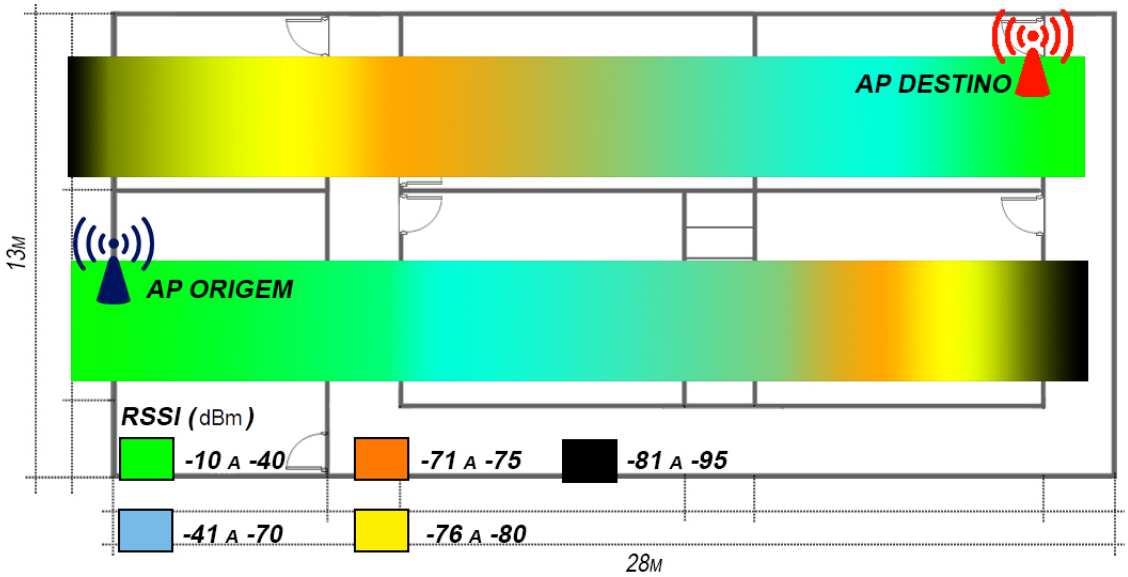



**Figura 4.4:** RSSI percebido pela STA

Para a realização deste experimento, foram utilizados os seguintes equipamentos/progra-

mas:

- Dois *switches Wi-Fi* da marca TPLINK modelo WR1043ND com *firmware* modificado para *Openwrt* com *OpenFlow v1.0* Os dois *switches* funcionam como APs origem e destino;
- Um controlador POX instalado em um computador com processador Core i5 e 4 GB de memória RAM e com sistema operacional Linux/Mint 64 bits. Nesse mesmo computador está instalado o *software* gerente que coleta informações da rede e repassa para o controlador tomar as decisões de detecção e descoberta;
- Um computador com processador *quadcore* com 8 GB de memória RAM com o *software iperf3* sendo executado no modo servidor responsável por receber o tráfego da STA;
- Um *notebook* com processador core i5 e 6 GB de memória RAM, com placa de rede *Intel Centrino Wireless* N1030 com o *software iperf3* executando no modo cliente e gerando tráfego udp à taxa de 1 Mbps. Este computador é utilizado como STA.
- Uma máquina virtual rodando o sistema operacional Linux/Backtrack R3 com adapatador usb *wireless* dlink dwa-125. O tráfego foi capturado com o *software airodump-ng* e analisado posteriormente com o *software Wireshark*. Esta máquina virtual foi instalada no *notebook* usado como dispositivo móvel (STA) no experimento.

## 4.3 Abordagem tradicional

O primeiro teste foi realizado com a abordagem tradicional, onde a STA inicia o processo de detecção e em seguida o processo de descoberta do AP no qual deve se conectar. As tabelas de fluxo dos *switches* foram implantadas previamente redirecionando o tráfego da porta *Wi-Fi* para a porta onde estava conectado o servidor *iperf3* para que desta forma o tempo gasto com troca de mensagens *OpenFlow* não interferisse no processo.

Com o servidor *iperf3* iniciado, executou-se o cliente *iperf3* no dispositivo móvel transferindo dados a uma taxa de 1 Mbps através de uma conexão udp, repetindo assim as características de uma transmissão multimídia e em tempo real. Foram feitos 10 (dez) deslocamentos no cenário com os dados sendo sempre capturados pelo *airodump-ng* e em seguida

analisados com o auxílio do *software Wireshark*.

Ao realizar os testes nestas condições, foi percebido, inicialmente de forma empírica e depois confirmado com a análise dos dados, que a STA inicia o processo de descoberta com um nível considerável de degradação do sinal chegando a valores próximos a -90 dBm. Foi considerado para esta medição o tempo inicial como sendo o tempo do último pacote udp transmitido pela STA ao servidor estando a STA ainda associada ao AP origem e o tempo final como o primeiro pacote udp transmitido pela STA ao servidor estando a STA já associada ao AP destino. Os dados coletados e analisados estão demonstrados na Tabela 4.1.

**Tabela 4.1:** Tempo de *handoff* tradicional

| **Deslocamento** | ***handoff* (s)** | **RSSI (dBm)** |
|---|---|---|
| 1 | 1,26 | -85,00 |
| 2 | 3,19 | -91,00 |
| 3 | 1,01 | -87,00 |
| 4 | 1,80 | -86,00 |
| 5 | 0,90 | -80,00 |
| 6 | 1,20 | -70,00 |
| 7 | 4,07 | -90,00 |
| 8 | 3,05 | -87,00 |
| 9 | 3,01 | -85,00 |
| 10 | 1,08 | -78,00 |
| **Tempo Médio** | **2,06** | **-83,90** |
| **Desvio Padrão** | **1,10** | **5,97** |

## 4.4 Abordagem SDWN

A segunda parte dos testes foi realizada com a presença de um controlador externo. Foi desenvolvido um módulo para executar a função de gerente recebendo dados da STA em movimento bem como dos APs de forma a decidir o melhor momento de iniciar o processo de *handoff*. O módulo gerente se comunica com o controlador POX que por sua vez instala os fluxos nos APs de acordo com a movimentação da STA.

Como no primeiro teste foram realizados 10 deslocamentos no total, iniciando com a STA conectada ao AP origem e se deslocando em direção ao AP destino. Os dados coletados pela STA são extraídos dos *beacon frames* recebidos dos APs ao seu alcance e enviados

ao controlador, que ao comparar os valores dos RSSI recebidos desinstala e instala fluxos nos APs referentes a STA em movimento, e envia para a mesma para qual AP ela deve se conectar.

Desta forma os processos de detecção e descoberta são melhorados em uma média de 0,9 s em comparação à abordagem tradicional. Os tempos obtidos, bem como a média e o desvio padrão estão descritos na Tabela 4.2.

**Tabela 4.2:** Tempo *handoff* SWDN

| **Deslocamento** | ***handoff* (s)** | **RSSI (dBm)** |
|---|---|---|
| 1 | 0,90 | -76,00 |
| 2 | 0,93 | -78,00 |
| 3 | 0,61 | -70,00 |
| 4 | 0,75 | -72,00 |
| 5 | 0,80 | -73,00 |
| 6 | 0,78 | -73,00 |
| 7 | 0,96 | -78,00 |
| 8 | 1,29 | -79,00 |
| 9 | 0,96 | -78,00 |
| 10 | 1,24 | -80,00 |
| **Valor Médio** | **0,92** | **-75,70** |
| **Desvio Padrão** | **0,20** | **3,26** |

## 4.5 Comparação dos resultados

Ao comparar os resultados, fica evidente o ganho obtido pela abordagem proposta, conforme pode ser visto na Figura 4.5.

O tempo gasto no processo de *handoff* utilizando a abordagem SDWN foi aproximadamente 50% menor que no modelo tradicional onde a STA é responsável pelas fases de detecção e descoberta. O ganho de quase 1 s é muito significativo para aplicações sensíveis ao atraso como as aplicações de VoIP e *streaming* de vídeo. Além disso, o baixo valor do desvio padrão indica um melhor equilíbrio no processo como um todo causando um ganho na estabilidade da comunicação e no funcionamento das aplicações.

A detecção é o momento em que o *handoff* se inicia e normalmente a STA toma essa decisão baseada na força do sinal recebido do AP. Por não ser padronizada, cada fabricante

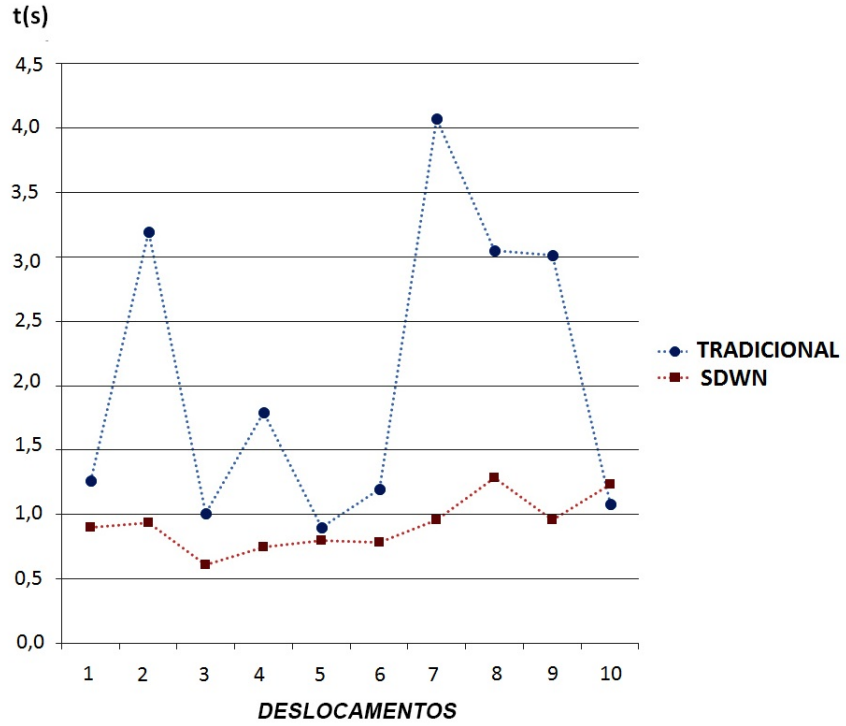



**Figura 4.5:** Tempo SDWN X Tradicional

pode determinar um valor diferente para utilizar como *threshold*.

A Figura 4.6 compara os valores do RSSI no momento inicial do *handoff*. Percebe-se que o sinal degrada muito quando a STA fica responsável por começar o processo, ao passo que na abordagem aqui proposta a troca é controlada, o que a torna mais eficiente.

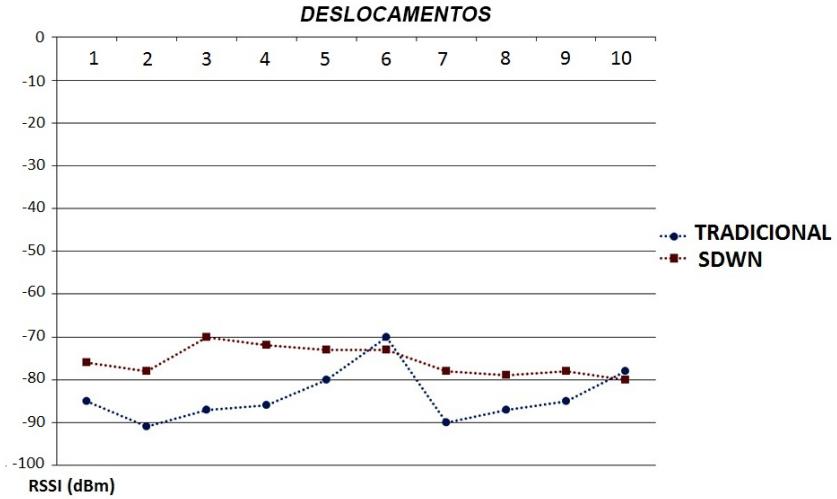



**Figura 4.6:** RSSI coletado no momento inicial do *handoff*

No experimento realizado com a abordagem tradicional verificou-se que ao se deslocar

em direção a outro AP, mesmo com o sinal bem mais forte recebido, a STA permanecia conectada ao AP original apesar de valores muito baixos de RSSI. Como consequência da demora em iniciar o processo de *handoff* ocorre uma degradação da comunicação causando muitas tentativas de retransmissão e um consequente aumento no tempo dispendido.

A média e o desvio padrão também são importantes para demonstrar o quanto os tempos aferidos com a abordagem SDWN se mantiveram mais constantes ao longo dos testes conforme a Figura 4.7.

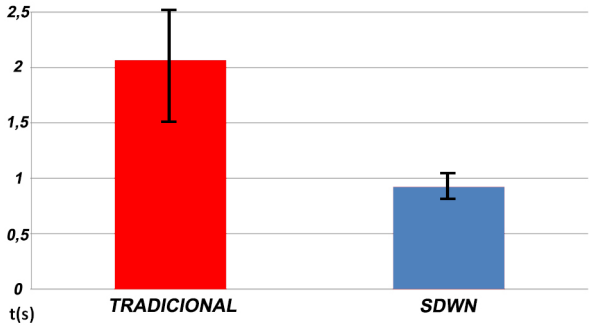



**Figura 4.7:** Média e Desvio Padrão

## 4.6 Segunda fase dos testes: RSSI e Tráfego nos APs como parâmetros

Após a primeira fase comprovar a eficiência da proposta aqui apresentada para a fase de detecção do processo de *handoff*, realizou-se a segunda parte dos experimentos adicionando mais um parâmetro para melhorar a decisão do controlador no momento da descoberta, ou seja, a decisão de qual AP a STA deve se conectar. O parâmetro utilizado para este fim foi a quantidade de tráfego na interface *Wi-Fi* dos APs de destino e as métricas empregadas foram o *jitter* e a perda de pacotes no momento da troca.

O valor para o limiar da quantidade de tráfego utilizado foi de 40 Mbps.

Para capturar os valores do tráfego nos APs foi utilizado o protocolo SNMP que consultou os valores dos objetos *ifInOctets* e *ifOutOctets* referentes a interface *Wi-Fi* dos APs.

Para realização dessa parte dos testes a rede foi configurada conforme demonstrado na Figura 4.8.

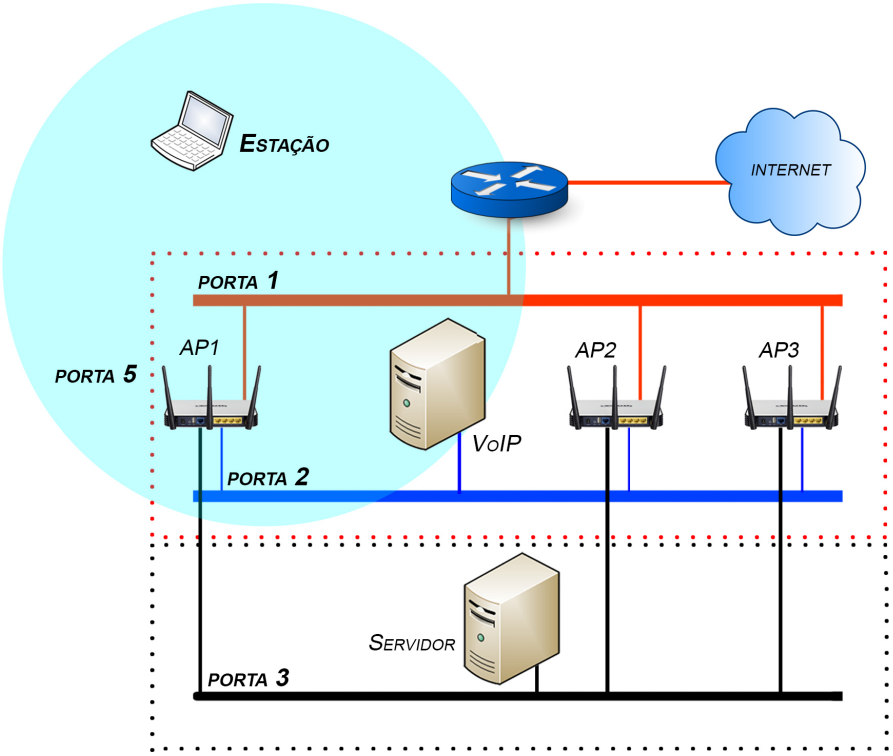



**Figura 4.8:** Cenário lógico da rede

O controlador utilizado para gerenciar as tabelas de fluxo de forma dinâmica nos *switches* foi o controlador POX. Os fluxos são instalados imediatamente após o controlador detectar a necessidade do *handoff* conforme demonstrado na Tabela 4.3. Os campos não citados recebem o valor * indicando qualquer valor ou recebem o valor *default*.

**Tabela 4.3:** Tabela de fluxo nos APs

| **Fluxos** | **Campos** | |
|---|---|---|
| | ***dl_dst*** | ***action*** |
| 1 | MAC_DO_ROTEADOR | port 1 |
| 2 | FF:FF:FF:FF:FF:FF | FLOOD |
| 3 | MAC_DO_VoIP | port 2 |
| 4 | MAC_DO_SERVIDOR | port 3 |
| 5 | MAC_DA_ESTAÇÃO | Port 5 (*Wi-Fi*) |

Nos experimentos dessa fase também foi utilizado o *software iperf3* para simular o tráfego de VoIP gerando tráfego udp à taxa de 1 Mbps.

Para poder comparar os resultados foram realizados 5 testes da seguinte forma. O primeiro com o *handoff* ocorrendo de forma tradicional sendo iniciado pela STA e trocando

para o AP que estivesse com o melhor valor de RSSI sendo que este AP estaria descongestionado, ou seja, com o nível de tráfego abaixo dos 40 Mbps definidos como limiar. O segundo teste ocorreu da mesma maneira só que o AP no momento da troca estaria com tráfego acima de 40 Mbps.

Já no terceiro e quarto testes foi usada a abordagem SDWN proposta neste trabalho utilizando como parâmetro apenas o RSSI, ou seja, a decisão da troca independe da quantidade de tráfego nos APs de destino.

O quinto experimento foi realizado levando em consideração a quantidade de tráfego nos APs de destino. Apesar do AP estar com RSSI melhor, estaria congestionado então a melhor decisão seria conectar no outro AP.

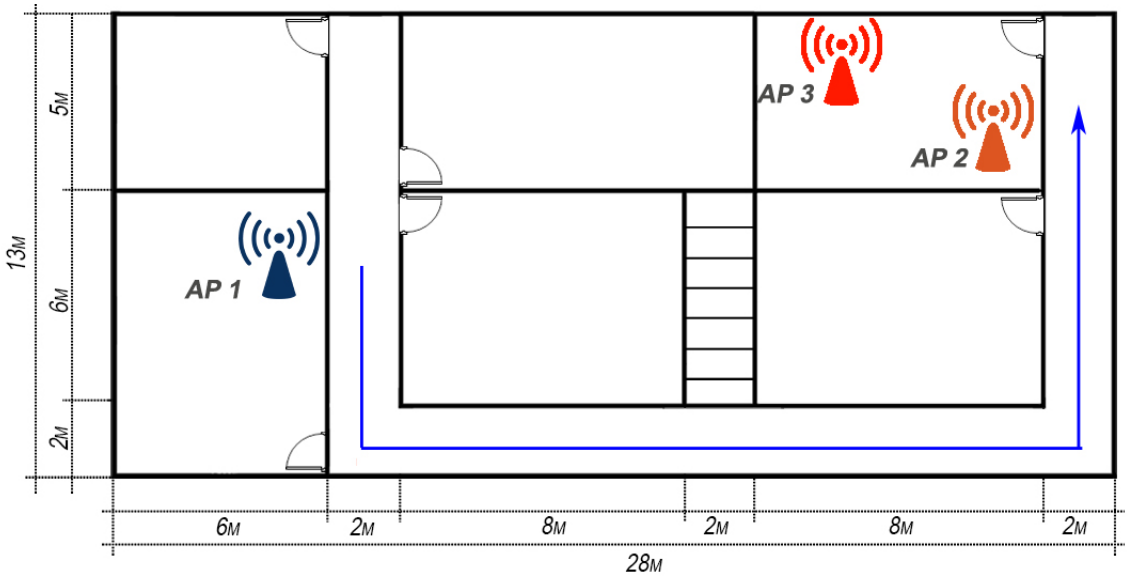



**Figura 4.9:** Deslocamentos na segunda fase do experimento

Foram realizados 10 deslocamentos pelo *testbed* conforme indicado na Figura 4.9 sendo que a STA começava conectada no AP1 e iniciava o fluxo *iperf3* com o Servidor VoIP. Em seguida iniciava-se o deslocamento em direção aos APs 2 e 3.

Da mesma maneira que na primeira fase, os APs 1 e 2 tiveram sua potência de transmissão reduzida para 10 dBm (10 mW). O AP3 foi configurado com sua potência máxima: 27 dBm (501 mW). Essa calibragem se fez necessária, pois os APs 2 e 3, apesar de estarem fisicamente no mesmo ambiente, precisavam estar com RSSI diferentes para a realização dos testes e validação dos resultados.

Após essa configuração foi feito uma medição utilizando o *software Inssider v3* por toda

extensão do *testbed* conforme pode ser visto na Figura 4.10. Os valores estão em dBm.

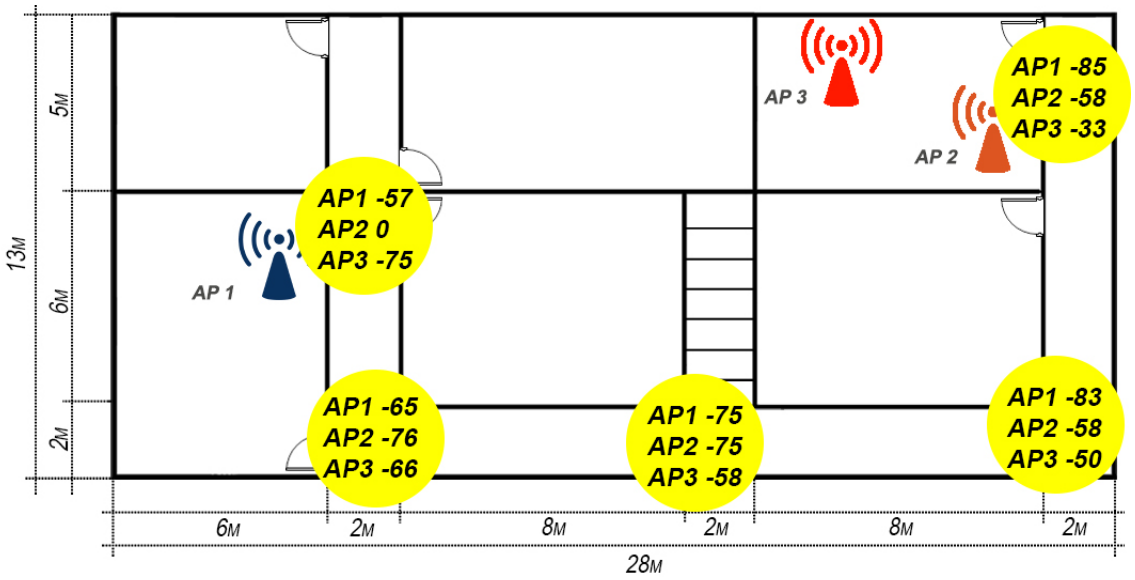



**Figura 4.10:** Força do sinal ao longo do *testbed*

Os dois primeiros testes foram realizados de acordo com a descrição a seguir.

- Foram feitos 10 deslocamentos pelos *testbed* com a estação conectada inicialmente no AP1 e se movendo em direção aos APs 2 e 3.
- No momento da escolha do AP destino, o RSSI tem prioridade sobre a quantidade tráfego.
- O AP3 está pré-configurado para ter um valor de RSSI melhor do que o AP2, portanto será escolhido como AP destino.
- Foi utilizado o *software iperf3* transferindo um total de 5 MB de tráfego UDP à uma taxa de 1 Mbps simulando uma aplicação em tempo real.

Esses valores foram utilizados nos testes demostrados nos itens 4.6.1 e 4.6.2 e os resultados estão descritos a seguir.

### 4.6.1 Primeiro teste: Abordagem Tradicional com RSSI como parâmetro prioritário com tráfego MENOR que 40 Mbps no AP destino

Os resultados obtidos no primeiro teste estão demonstrados na Tabela 4.4.

**Tabela 4.4:** Tradicional - AP destino com tráfego < 40 Mbps

| Teste | *handoff* (s) | RSSI AP1 (dBm) | *Jitter* (ms) | Perdidos (und) | Total (und) | Perdidos (%) |
|---|---|---|---|---|---|---|
| 1 | 1,69 | **-78,00** | 30,82 | 166,00 | 639,00 | 25,98% |
| 2 | 4,46 | **-76,00** | 47,72 | 245,00 | 639,00 | 38,34% |
| 3 | 1,70 | **-71,00** | 35,30 | 108,00 | 639,00 | 16,90% |
| 4 | 3,13 | **-78,00** | 33,10 | 211,00 | 639,00 | 33,02% |
| 5 | 4,11 | **-89,00** | caiu a conexão | 298,00 | 639,00 | 46,64% |
| 6 | 4,08 | **-89,00** | caiu a conexão | 345,00 | 639,00 | 53,99% |
| 7 | 0,95 | **-70,00** | 21,64 | 51,00 | 639,00 | 7,98% |
| 8 | 2,10 | **-90,00** | 21,94 | 306,00 | 639,00 | 47,89% |
| 9 | 1,17 | **-88,00** | 29,70 | 111,00 | 639,00 | 17,37% |
| 10 | 2,59 | **-88,00** | 24,19 | 150,00 | 639,00 | 23,47% |
| **Média** | **2,60** | **-81,70** | **24,44** | **199,10** | **639,00** | **31,16%** |
| **Desvio padrão** | **1,22** | **7,52** | **14,18** | **92,87** | **-** | **14,53%** |

Na abordagem tradicional a STA é responsável pelas fases de detecção e descoberta Ao analisar os dados nota-se que o sinal do RSSI passa de -80 dBm em média para o processo ser iniciado o que causa de quase 32% pacotes perdidos em média. Em alguns casos o sinal degradou ao ponto de causar a perda da conexão como servidor como demonstrado nos testes 5 e 6.

### 4.6.2 Segundo teste: Abordagem Tradicional com RSSI como parâmetro prioritário com tráfego MAIOR que 40 Mbps no AP destino

Nesse teste foi gerado um tráfego de 40 Mbps no AP3 que está pré-programado para ter um RSSI maior que o AP2, portanto será escolhido como AP destino, mesmo estando congestionado. Os resultados obtidos estão demonstrados na Tabela 4.5.

O RSSI é utilizado como único parâmetro tanto para a fase de detecção como a fase de descoberta na abordagem tradicional. Como consequência a mudança é sempre feita para o AP com melhor RSSI mesmo que o mesmo esteja congestionado. Esse fato acarreta em um aumento no número médio de pacotes perdidos que passou de 31,16% no primeiro teste para aproximadamente 50% nesse teste. Os valores muito baixos do RSSI no momento da troca degradam tanto a conexão que a mesma caiu em 40% dos casos como mostram os

**Tabela 4.5:** Tradicional - AP destino com tráfego > 40 Mbps

| Teste | *handoff* (s) | RSSI AP1 (dBm) | *Jitter* (ms) | Perdidos (und) | Total (und) | Perdidos (%) |
|---|---|---|---|---|---|---|
| 1 | 1,21 | -78,00 | 19,01 | 150,00 | 639,00 | 23,47% |
| 2 | 1,89 | -85,00 | caiu a conexão | 408,00 | 639,00 | 63,85% |
| 3 | 1,90 | -80,00 | 24,007 | 113,00 | 639,00 | 17,68% |
| 4 | 6,62 | -89,00 | caiu a conexão | 387,00 | 639,00 | 60,56% |
| 5 | 1,37 | -87,00 | caiu a conexão | 563,00 | 639,00 | 88,11% |
| 6 | 2,71 | -88,00 | caiu a conexão | 420,00 | 639,00 | 65,73% |
| 7 | 1,79 | -78,00 | 16,568 | 144,00 | 639,00 | 22,54% |
| 8 | 2,34 | -79,00 | 32,117 | 138,00 | 639,00 | 21,60% |
| 9 | 1,34 | -78,00 | 15,002 | 380,00 | 639,00 | 59,47% |
| 10 | 5,95 | -88,00 | caiu a conexão | 390,00 | 639,00 | 61,03% |
| **Média** | **2,71** | **-83,00** | **10,67** | **309,30** | **639,00** | **48,40%** |
| **Desvio padrão** | **1,84** | **4,54** | **11,53** | **149,83** | **-** | **23,45%** |

deslocamentos 2, 4, 5, 6 e 10. Um dado interessante nesses dois testes é que o tempo gasto no processo de *handoff* se manteve estável com valor médio de 2,6 s.

Os próximos 2 testes utilizaram a abordagem SDWN proposta neste trabalho. Foram executados deslocamentos pelos *testbed* utilizando as seguintes configurações:

- 10 deslocamentos pelos com a estação conectada inicialmente no AP1 e se movendo em direção aos APs 2 e 3.
- No momento da escolha do AP destino, o RSSI tem prioridade sobre a quantidade tráfego.
- O AP3 está pré-configurado para ter um valor de RSSI melhor do que o AP2, portanto será escolhido como AP de destino.
- Foi utilizado o *software iperf3* transferindo um total de 5MB de tráfego udp à uma taxa de 1 Mbps simulando uma aplicação em tempo real.

Esses valores foram utilizados nos testes demostrados nos itens 4.6.3 e 4.6.4 e os resultados estão descritos a seguir.

### 4.6.3 Terceiro teste: Abordagem SDWN com RSSI como parâmetro prioritário e com tráfego MENOR que 40 Mbps no AP destino

Como nos testes anteriores, o AP3 continua com RSSI melhor que o AP2, portanto será escolhido como AP destino e está descongestionado. A abordagem proposta foi utilizada nas fazes de detecção e descoberta. Os resultados obtidos estão demonstrados na Tabela 4.6.

**Tabela 4.6:** SDWN - AP destino com tráfego < 40 Mbps

| **Teste N** | ***handoff* (s)** | **RSSI AP1 (dBm)** | ***Jitter* (ms)** | **Perdidos (und)** | **Pacotes (und)** | **Perdidos (%)** |
|---|---|---|---|---|---|---|
| 1 | 1,28 | **-71,00** | 0,12 | 50,00 | 640,00 | 7,81% |
| 2 | 1,07 | **-77,00** | 0,37 | 48,00 | 640,00 | 7,50% |
| 3 | 1,29 | **-70,00** | 1,16 | 50,00 | 640,00 | 7,81% |
| 4 | 1,17 | **-74,00** | 4,50 | 48,00 | 640,00 | 7,50% |
| 5 | 1,57 | **-70,00** | 0,17 | 46,00 | 640,00 | 7,19% |
| 6 | 1,17 | **-73,00** | 6,37 | 48,00 | 640,00 | 7,50% |
| 7 | 1,18 | **-70,00** | 3,45 | 48,00 | 640,00 | 7,50% |
| 8 | 1,17 | **-71,00** | 7,19 | 48,00 | 640,00 | 7,50% |
| 9 | 1,20 | **-74,00** | 6,40 | 48,00 | 640,00 | 7,50% |
| 10 | 1,18 | **-71,00** | 0,06 | 50,00 | 640,00 | 7,81% |
| **Média** | **1,23** | **-72,10** | 2,98 | **48,40** | **640,00** | **7,56%** |
| **Desvio padrão** | **0,13** | **2,21** | **2,79** | **1,20** | **-** | **0,19%** |

Analisando os dados obtidos nesse experimento e descritos na Tabela 4.6, percebe-se que uma grande redução no percentual de pacotes perdidos com valores abaixo de 8% contra os 31% da abordagem tradicional. O valor de 2,79 no desvio padrão do *jitter* indica que os valores medidos tiveram uma baixa variação com 7 valores abaixo de 6 ms 1 valor com 7 ms e apenas 2 valores se afastando da média obtida de 2,98 ms. Os valores do RSSI no momento da troca também variaram muito pouco com um desvio padrão de 2,21 e uma média de -72,1 dBm indicando que o *handoff* é iniciado com um valor muito bom de RSSI, ao contrário da abordagem tradicional que permite o sinal degradar bastante chegando a quase -90 dBm causando muitas retransmissões e consequentes perdas de pacotes. Outro ponto muito importante foi o tempo gasto no processo de *handoff* que baixou em mais de 1 s o tempo médio obtido com a abordagem tradicional com uma média de 1,2 s.

### 4.6.4 Quarto teste: Abordagem SDWN com RSSI como parâmetro prioritário e com tráfego MAIOR que 40 Mbps no AP destino

Para medir o desempenho da abordagem proposta com o AP destino congestionado, foi gerado tráfego de 40 Mbps no AP3 que estava com o melhor RSSI e por isso foi escolhido como AP destino.

Os resultados obtidos estão demonstrados na Tabela 4.7.

**Tabela 4.7:** SDWN - AP destino com tráfego > 40 Mbps

| **Teste N** | ***handoff* (s)** | **RSSI AP1 (dBm)** | ***Jitter* (ms)** | **Perdidos (und)** | **Total (und)** | **Perdidos (%)** |
|---|---|---|---|---|---|---|
| 1 | 1,22 | **-75,00** | 8,77 | 56,00 | 640,00 | 8,75% |
| 2 | 1,48 | **-76,00** | 16,74 | 59,00 | 640,00 | 9,22% |
| 3 | 1,17 | **-73,00** | 12,20 | 75,00 | 640,00 | 11,72% |
| 4 | 1,18 | **-71,00** | 8,66 | 48,00 | 640,00 | 7,50% |
| 5 | 1,24 | **-71,00** | 5,74 | 58,00 | 640,00 | 9,06% |
| 6 | 1,19 | **-71,00** | 4,85 | 59,00 | 640,00 | 9,22% |
| 7 | 1,08 | **-73,00** | 30,07 | 60,00 | 640,00 | 9,38% |
| 8 | 1,43 | **-74,00** | 6,66 | 60,00 | 640,00 | 9,38% |
| 9 | 1,24 | **-74,00** | 29,05 | 91,00 | 640,00 | 14,22% |
| 10 | 1,12 | **-71,00** | 10,73 | 53,00 | 640,00 | 8,28% |
| **Média** | **1,24** | **-72,90** | 13,35 | **61,90** | **640,00** | **9,67%** |
| **Desvio padrão** | **0,12** | **1,76** | **8,74** | **11,68** | **-** | **1,83%** |

A quantidade de tráfego no AP destino não afetou o tempo do *handoff* que continuou com valor médio de 1,2 s. O RSSI no momento da troca se manteve muito próximos do valor médio de -72,9 dBm o que causou uma perda de pacotes com valor médio de 9,67%. Devido ao congestionamento no AP destino as perdas foram um pouco maiores, porém ainda em um patamar muito melhor que os quase 50% de perdas da abordagem tradicional.

### 4.6.5 Quinto teste: Abordagem SDWN com QUANTIDADE DE TRÁFEGO no AP Destino como parâmetro prioritário

Para finalizar os experimentos realizou-se um teste com as seguintes configurações.

- 10 deslocamentos pelos com a estação conectada inicialmente no AP1 e se movendo em direção aos APs 2 e 3.
- No momento da escolha do AP destino, a QUANTIDADE DE TRÁFEGO tem prioridade sobre o RSSI.
- O AP3 está pré-configurado para ter um valor de RSSI melhor do que o AP2, e congestionado com um tráfego maior do que 40 Mbps. O AP2 está com um RSSI inferior porém descongestionado. O AP2 será escolhido como AP destino.
- Foi utilizado o software iperf3 transferindo um total de 5MB de tráfego udp à uma taxa de 1 Mbps simulando uma aplicação em tempo real.

Esses valores foram utilizados no próximo teste e os resultados estão demonstrados na Tabela 4.8.

**Tabela 4.8:** SDWN - AP destino com tráfego < 40 Mbps. Troca para AP com menor RSSI

| **Teste N** | ***handoff* (s)** | **RSSI AP1 (dBm)** | ***Jitter* (ms)** | **Perdidos (und)** | **Pacotes (und)** | **Perdidos (%)** |
|---|---|---|---|---|---|---|
| 1 | 1,21 | **-73,00** | 5,62 | 48,00 | 640,00 | 7,50% |
| 2 | 1,21 | **-77,00** | 7,35 | 48,00 | 640,00 | 7,50% |
| 3 | 1,22 | **-78,00** | 6,87 | 48,00 | 640,00 | 7,50% |
| 4 | 1,13 | **-70,00** | 7,96 | 48,00 | 640,00 | 7,50% |
| 5 | 1,29 | **-77,00** | 8,62 | 57,00 | 640,00 | 8,91% |
| 6 | 1,23 | **-73,00** | 7,88 | 49,00 | 640,00 | 7,66% |
| 7 | 1,23 | **-70,00** | 12,45 | 48,00 | 640,00 | 7,50% |
| 8 | 1,24 | **-71,00** | 4,57 | 48,00 | 640,00 | 7,50% |
| 9 | 1,12 | **-75,00** | 9,02 | 48,00 | 640,00 | 7,50% |
| 10 | 1,20 | **-76,00** | 14,89 | 55,00 | 640,00 | 8,59% |
| **Média** | **1,21** | **-74** | 8,52 | **49,70** | **640,00** | **7,77%** |
| **Desvio padrão** | **0,05** | **2,86** | **2,92** | **3,20** | **-** | **0,50%** |

Este teste evidencia a importância de utilizar outro parâmetro, além do RSSI, para a decisão de qual AP deverá ser o AP destino. Houve uma melhora tanto no *jitter* como no percentual médio de pacotes perdidos que voltou a ficar abaixo de 8% como no teste explicado na subseção 4.6.4.

### 4.6.6 Comparação dos resultados

Os testes foram realizados com o objetivo de comparar o desempenho do *handoff* usando a abordagem tradicional e a abordagem SDWN proposta neste trabalho utilizando as métricas de tempo do *handoff*, *jitter* e perda de pacotes. As Figuras 4.11, 4.12 e 4.13 apresentam uma comparação entre os resultados obtidos utilizando o RSSI como parâmetro de troca e com o AP de destino livre (itens a) e congestionado (itens b), respectivamente.

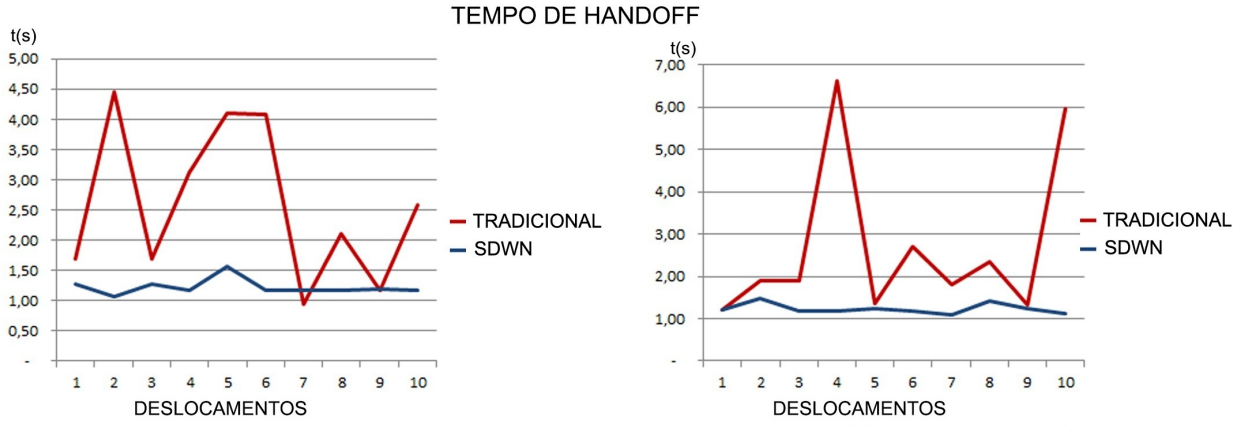



**Figura 4.11:** Tempo de *handoff* trocando para o AP com melhor RSSI

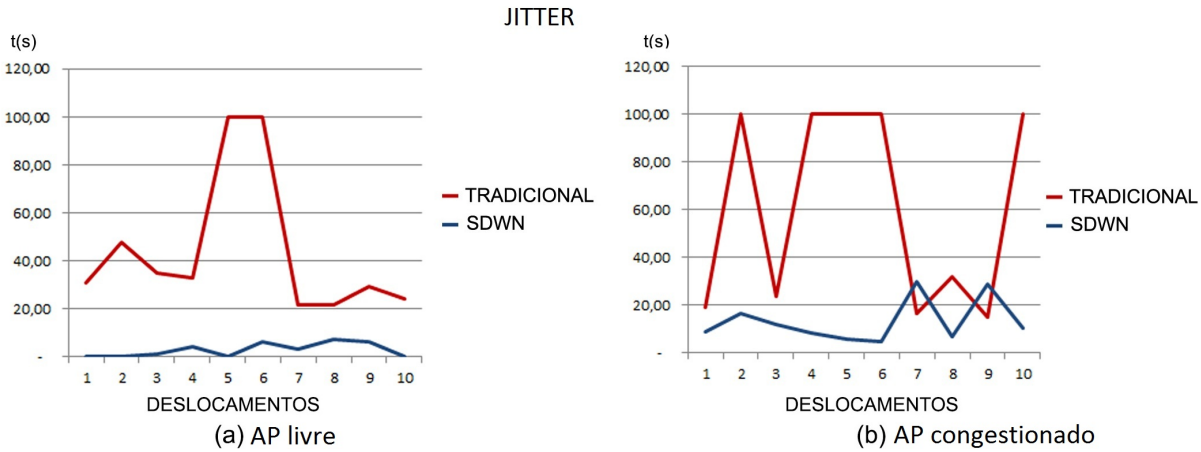



**Figura 4.12:** *Jitter* trocando para o AP com melhor RSSI

Ao analisar os gráficos percebe-se que a abordagem aqui proposta é mais eficiente em todas as três métricas analisadas. O tempo do *handoff* se manteve estável e muito próximo de 1 s, ao contrário da abordagem tradicional aonde o tempo chegou a passar de 6 s quando o AP destino estava congestionado. Analisando a métrica de perda de pacotes observa-se claramente que os valores obtidos com abordagem SDWN ficaram abaixo de 10%. Em

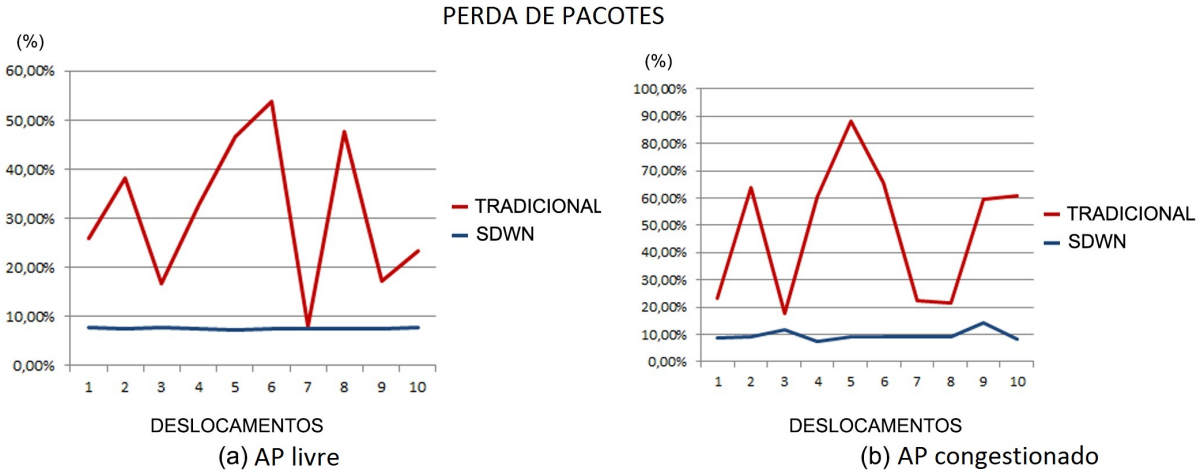



**Figura 4.13:** Perda de Pacotes trocando para o AP com melhor RSSI

contrapartida a abordagem tradicional chegou a gerar perdas superiores a 50% no AP destino descongestionado e próximo de 90% com o AP destino congestionado.

Na abordagem SDWN proposta nesse trabalho os valores do tempo gasto no *handoff* se mantiveram constantes com o AP destino livre ou congestionado com valores próximos a 1 s. No entanto ao observar os valores do *jitter* e da perda de pacotes nota-se que houve um aumento nesses indicadores, sugerindo que quando o AP destino está congestionado, caso haja um outro AP que esteja com o nível de RSSI inferior porém sem tráfego, é melhor trocar para o mesmo.

Para comprovar essa hipótese, realizou-se outro experimento programando o controlador para que a escolha do destino seja efetuada levando em consideração a quantidade de tráfego na interface *Wi-Fi* dos APs, ou seja, se um determinado AP estiver com melhor RSSI, porém congestionado, o controlador decicirá trocar para um AP que esteja com um RSSI inferior, no entanto descongestionado. Os resultados obtidos estão demonstrados nas Figuras 4.14, 4.15 e 4.16.

Ao analisar as figuras acima nota-se que quando o tráfego é utilizado para decidir qual AP será escolhido como destino, os indicadores de tempo de *handoff*, *jitter* e perda de pacotes são melhores. Pode-se concluir que o RSSI não é o melhor parâmetro para decidir o AP destino.

Com os experimentos realizados comprovou-se que ao utilizar a quantidade de tráfego como parâmetro os resultados obtidos foram melhores.

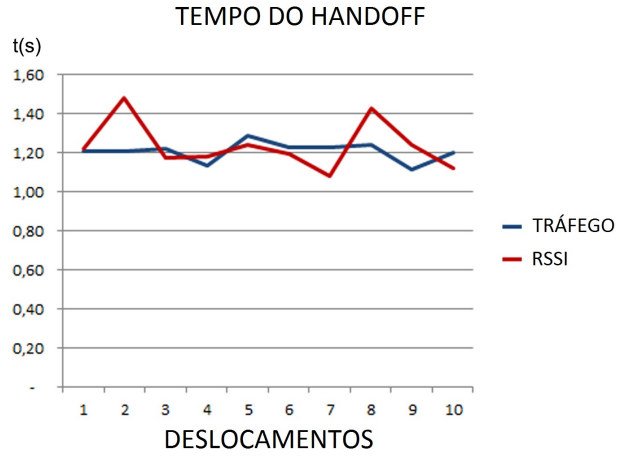



**Figura 4.14:** Tempo do *handoff*. Escolha efetuada pela quantidade de tráfego na interface dos APs

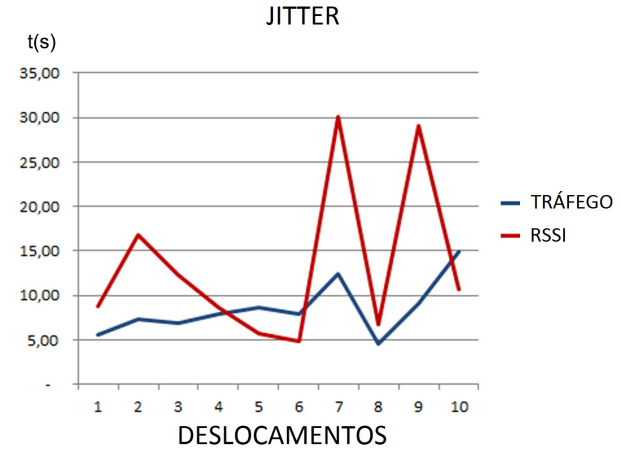



**Figura 4.15:** *Jitter*. Escolha efetuada pela quantidade de tráfego na *interface* dos APs

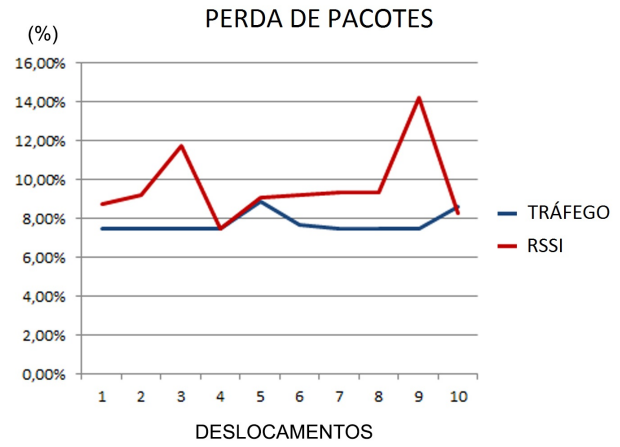


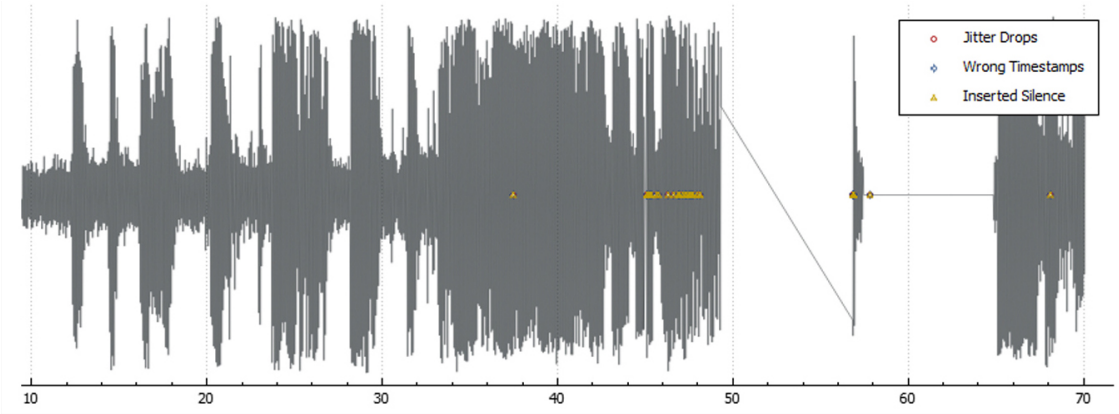

**Figura 4.16:** Perda de Pacotes. Escolha efetuada pela quantidade de tráfego na *interface* dos APs

### 4.6.7 Teste com uma aplicação real de VoIP

Como já mencionado, todos os testes foram realizados com um *software* gerador de tráfego simulando uma aplicação em tempo real. Todos os resultados obtidos comprovam as vantagens da abordagem aqui apresentada sobre a abordagem tradicional. Nesta seção apresenta-se uma comparação entre os resultados obtidos utilizando um aplicação real de VoIP.

Para realização desse experimento utilizou-se o *softphone Ekiga* para realizar uma chamada de voz e o *software Wireshark* para captura do tráfego e posterior análise e decodificação do mesmo. O *codec* utilizado foi o G711U com um *sample rate* de 8000 Hz. No primeiro experimento foi utilizada a abordagem tradicional. Foi realizado um deslocamento pelo *testbed* iniciando com a estação conectada no AP1. Em seguida foi realizada a chamada de voz e no momento que a conexão foi estabelecida iniciou-se o deslocamento em direção aos APs de destino.

O tráfego capturado e decodificado pelo *Wireshark* está demonstrado na Figura 4.17.




| Source Address | Source Port | Destination Address | Destination Port | SSRC | Setup Frame | Packets | Time Span (s) | Sample Rate (Hz) | Payloads |
|---|---|---|---|---|---|---|---|---|---|
| 192.168.0.100 | 5062 | 192.168.0.200 | 5086 | 0xe36a4d74 | 77 | 2669 | 9.28 - 70 (60.8) | 8000 | g711U |

**Figura 4.17:** Chamada de VoIP com *handoff* realizado com a abordagem tradicional

Em uma chamada de VoIP os pacotes com o áudio são transportados pelo *Real-Time Transport Protocol* (RTP) que é um protocolo de redes utilizado para aplicações em tempo real. Ele funciona como uma sub-camada na camada de transporte normalmente utilizando o protocolo *User Datagram Protocol* (UDP) e define como deve ser feita a fragmentação do fluxo de dados de áudio, adicionando a cada fragmento informação de sequência e de tempo de entrega. Utilizando o Wireshark é possível capturar os pacotes e separar os pacotes RTP que contêm os dados da conversa real em uma chamada VoIP. O *software* contém um

utilitário embutido que permite decodificar os dados capturados em um formato de áudio reproduzível sendo possível não apenas ouvir os diálogos efetuados duarante a chamada como visualizar a onda gerada. Percebe-se que houve uma interrupção muito grande no tráfego de voz e que, apesar da conexão continuar estabelecida, a comunicação foi prejudicada.

O mesmo experimento foi realizado utilizando os mecanismos SDWN propostos por este trabalho e o resultado está demonstrado na Figura 4.18.

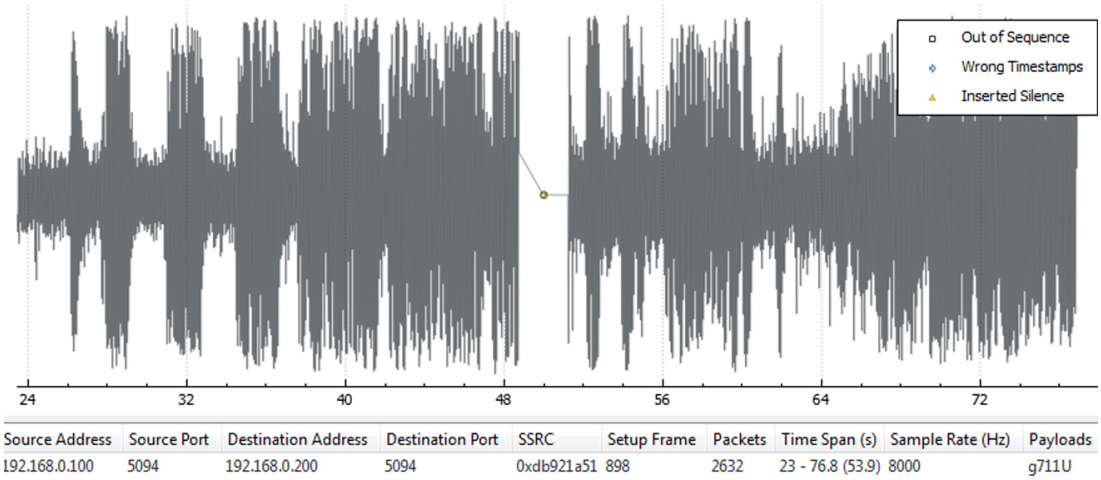



**Figura 4.18:** Chamada de VoIP com *handoff* realizado com a abordagem SDWN

Apesar de ter sido realizado apenas um experimento nota-se claramente, ao comparar as duas figuras appresentadas, que ao tutilizar os mecanismos da abordagem aqui proposta o tempo de interrupção é muito menor causando uma redução significativa na quebra da comunicação trazendo grandes benfícios para as aplicações em tempo real que, para obterem o QoS necessário, necessitam de pequenos atrasos.

# Capítulo 5

# Considerações Finais

As fases de detecção e descoberta são as maiores responsáveis pelo atraso adicionado durante o processo de *handoff*. Este trabalho descreveu uma abordagem baseada no paradigma SDWN para gerenciar o processo de *handoff* em uma rede *Wi-Fi* atuando nestas duas fases. Para realização dos testes apresentados, foram utilizados APs comercias e de baixo-custo com o *firmware* modificado para o *Openwrt* com o *OpenFlow* habilitado na versão 1.0.0. O protocolo *OpenFlow* foi utilizado para instalar as tabelas de fluxo nos APs levando em consideração o tipo de tráfego sendo o tráfego VoIP redirecionado para uma porta específica do AP a fim de garantir um canal exclusivo e com possibilidade de implementar mecanismos de provisão de QoS.

Foi proposto também um mecanismo de controle para que, baseado em decisões tomadas por um *software* executado em um controlador externo, o tempo do processo de *handoff* fosse reduzido. Isto foi conseguido através da verificação dos RSSIs percebidos pela STA e também pela decisão unilateral por parte do controlador de qual AP a STA deve se conectar. A abordagem proposta incialmente foi testada usando 1 AP como opção de destino e apenas o RSSI como parâmetro. Os testes realizados demonstraram um ganho aproximado de 1s no tempo total como também valores mais constantes, resultando em uma estabilidade maior em todo o processo. Em seguida acrescentou-se outro AP como opção de troca e passou-se a levar em consideração a quantidade de tráfego nas *interfaces Wi-Fi* dos APs resultando em dois parâmetros para o controlador decidir para qual AP a estação deveria efetuar uma nova associação.

Esta proposta comprovou que ao utilizar um controlador externo para tomar as decisões

de detecção, ou seja, o momento de iniciar o *handoff*, e a descoberta decidindo em qual AP a estação em movimento deve se associar, o processo de *handoff* foi mais eficiente. As métricas utilizadas para medir e comparar a eficiência do processo apresentaram sempre melhores resultados.

Uma proposta de trabalho de futuro seria a inclusão de outros parâmetros além do RSSI e da quantidade de tráfego nos APs medidos no momento da troca, aumentando assim o grau de inteligência da solução e tornando o mecanismo mais sofisticado para garantir a mobilidade das estações, prover o QoS adequado às aplicações como também oferecer mecanismos de segurança.

# Bibliografia


AL-SHAIKHLI, R. *Mobileflow: Applying SDN to Mobility in Wireless Networks*. Tese (Doutorado) — Texas A & M University, 2014.

ATZORI, L.; IERA, A.; MORABITO, G. The Internet of Things: A survey. *Computer Networks*, v. 54, n. 15, p. 2787–2805, out. 2010. ISSN 13891286.

BRIK, V.; MISHRA, A.; BANERJEE, S. Eliminating handoff latencies in 802.11 wlans using multiple radios: Applications, experience, and evaluation. In: USENIX ASSOCIATION. *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*. [S.l.], 2005. p. 27–27.

CHAN, Y.-C.; LIN, D.-J. The design of an ap-based handoff scheme for ieee 802.11 wlans. *International Journal of e-Education, e-Business, e-Management and e-Learning*, IACSIT Press, v. 4, n. 1, p. 72, 2014.

COSTANZO, S. et al. Software defined wireless networks: Unbridling sdns. In: IEEE. *2012 European Workshop on Software Defined Networking*. [S.l.], 2012. p. 1–6.

DELY, P. et al. A software-defined networking approach for handover management with real-time video in wlans. *Journal of Modern Transportation*, Springer, v. 21, n. 1, p. 58–65, 2013.

GROUP, I. . W. et al. 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 6: Wireless access in vehicular environments. *IEEE Std*, v. 802, p. 11p, 2012.

HU, H. et al. Software defined wireless networks (sdwn): Part 1 [guest editorial]. *IEEE Communications Magazine*, IEEE, v. 53, n. 11, p. 108–109, 2015.

HU, H. et al. Software defined wireless networks: Part 2 [guest editorial]. *IEEE Communications Magazine*, v. 54, n. 1, p. 10–11, January 2016. ISSN 0163-6804.

HUANG, P.-J.; TSENG, Y.-C.; TSAI, K.-C. A fast handoff mechanism for ieee 802.11 and iapp networks. In: IEEE. *2006 IEEE 63rd Vehicular Technology Conference*. [S.l.], 2006. v. 2, p. 966–970.

IEEE. *802.11ac-2013 - 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 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz*. [S.l.], 2013.

JIN, S.; CHOI, S. A seamless handoff with multiple radios in ieee 802.11 wlans. *IEEE Transactions on Vehicular Technology*, IEEE, v. 63, n. 3, p. 1408–1418, 2014.

KREUTZ, D. et al. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, IEEE, v. 103, n. 1, p. 14–76, 2015.

LUENGO, E. *An openflow-based wireless user management system*. Dissertação (Mestrado) — University of Ontario Institute of Technology, Ontario, 1 2016.

MCKEOWN, N. et al. Openflow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, ACM, v. 38, n. 2, p. 69–74, 2008.

MISAL, A.; SAMBARE, S. Reduction of handover latency by horizontal distance measurement using gps. *International Journal of Computer Applications*, Foundation of Computer Science, v. 105, n. 11, 2014.

MISHRA, A.; SHIN, M.; ARBAUSH, W. Context caching using neighbor graphs for fast handoffs in a wireless network. In: IEEE. *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*. [S.l.], 2004. v. 1.

MOURA, H. et al. Ethanol: Software defined networking for 802.11 wireless networks. In: IEEE. *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. [S.l.], 2015. p. 388–396.

OPEN NETWORK FOUNDATION. *OpenFlow Switch Specification v1.0.0*. [S.l.], 2009.

PAIVA, R. B. et al. Gerenciamento de recursos no processo de handoff em redes sem fio definidas por software. *IV Workshop de Pesquisa Experimental da Internet do Futuro (WPEIF)*, SBRC, p. 23–28, 2014.

SALIH, Q. M.; UDDIN, M.; MASTORAKIS, N. A review of handoff latency reducing techniques in ieee 802.11 wlan networks. *Scanning*, v. 24, p. 25, 2015.

SANGHAVI, N.; BANSODE, R. S. Improving ieee 802.11 wlan handoff latency by access point-based modification. *Global Journal of Computer Science and Technology*, v. 15, n. 5, 2015.

SEZER, S. et al. Are we ready for sdn? implementation challenges for software-defined networks. *IEEE Communications Magazine*, IEEE, v. 51, n. 7, p. 36–43, 2013.

SUN, M.; QIAN, H. Handover management scheme in sdn-based wireless lan. *Journal of Communications*, IACSIT Press, v. 11, n. 3, p. 282–289, 2016.

SUN, M.; QIAN, H. Handover management scheme in sdn-based wireless lan. *Journal of Communications*, v. 11, n. 3, p. 282–289, 2016.

TETARWAL, M. L.; KUNTAL, A.; KARMAKAR, P. Article: A review on handoff latency reducing techniques in ieee 802.11 wlan. *IJCA Proceedings on National Seminar on Recent Advances in Wireless Networks and Communications*, NWNC, n. 2, p. 22–28, April 2014. Full text available.

TIWARI, V. Sdn and openflow for beginners with hands on labs. *MMDD Multimedia LLC., Kindle Edition, Northville*, 2013.

VELAYOS, H.; KARLSSON, G. Techniques to reduce the ieee 802.11 b handoff time. In: IEEE. *Communications, 2004 IEEE International Conference on*. [S.l.], 2004. v. 7, p. 3844–3848.

YAN, M. et al. {\ AE} therflow: Principled wireless support in sdn. *arXiv preprint arXiv:1509.04745*, 2015.