



VIRTUALIZAÇÃO DE REDES IEEE 802.11 PARA BALANCEAMENTO DE CARGA

Tibério Menezes de Oliveira

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: José Ferreira de Rezende

Rio de Janeiro
Setembro de 2011

VIRTUALIZAÇÃO DE REDES IEEE 802.11 PARA BALANCEAMENTO DE
CARGA

Tibério Menezes de Oliveira

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO
ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE
ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE
JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A
OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA
ELÉTRICA.

Examinada por:

Prof. José Ferreira de Rezende, Dr.

Prof. Aloysio de Castro Pinto Pedroza, Dr.

Prof. Artur Ziviani, Dr.

RIO DE JANEIRO, RJ – BRASIL
SETEMBRO DE 2011

Oliveira, Tibério Menezes de

Virtualização de redes IEEE 802.11 para balanceamento de carga/Tibério Menezes de Oliveira. – Rio de Janeiro: UFRJ/COPPE, 2011.

XVII, 67 p.: il.; 29, 7cm.

Orientador: José Ferreira de Rezende

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2011.

Referências Bibliográficas: p. 62 – 67.

1. Virtualização de redes sem fio. 2. Balanceamento de carga. 3. Redes infraestruturadas. I. Rezende, José Ferreira de. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

*Dedico este trabalho a todos os
meus familiares e amigos.*

Agradecimentos

Gostaria de agradecer à minha família pelo apoio e incentivo que me foi dado.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

VIRTUALIZAÇÃO DE REDES IEEE 802.11 PARA BALANCEAMENTO DE CARGA

Tibério Menezes de Oliveira

Setembro/2011

Orientador: José Ferreira de Rezende

Programa: Engenharia Elétrica

Em uma rede IEEE 802.11 infraestruturada composta por múltiplos pontos de acesso, cada estação precisa decidir em qual deles deve se associar. Na abordagem tradicional, a estação se baseia apenas na relação sinal-ruído para a escolha do ponto de acesso. Este tipo de abordagem pode levar a um desbalanceamento no número de estações associadas a cada ponto de acesso, trazendo problemas de injustiça e de desempenho. Nesta dissertação, é apresentada uma nova proposta que combina a relação sinal-ruído com a carga em cada ponto de acesso e a capacidade de virtualização das redes sem fio. Desta forma, as estações estabelecem e alternam associações em diferentes pontos de acesso, permitindo o balanceamento de carga entre os pontos de acesso de forma a distribuir a capacidade disponível na rede de forma justa para os clientes, mantendo a ocupação máxima da rede. Resultados de simulação demonstram os ganhos obtidos com a proposta comparada a outras da literatura.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

IEEE 802.11 NETWORKS VIRTUALIZATION FOR LOAD BALANCE

Tibério Menezes de Oliveira

September/2011

Advisor: José Ferreira de Rezende

Department: Electrical Engineering

In an infrastructured IEEE 802.11 network composed by multiple access points, each station needs to make a decision on which access point to associate with. In the traditional approach, the station uses only the signal strength to support its decision. This type of approach can lead to an unbalanced number of stations in the access points, consequently causing performance and unfairness problems on the distribution. The present article proposes a new approach that combines the signal strength with the load in each access point plus the virtualization of the wireless network. Therefore, the stations establish and switch associations on different access points allowing the load balancing between the access points in order to distribute the available network capacity in a fair way, while keeping the maximum network utilization. The simulation results confirm the improvement obtained with the new approach in comparison to others in the literature.

Sumário

Lista de Figuras	x
Lista de Tabelas	xiii
Lista de Símbolos	xiv
Lista de Abreviaturas	xvi
1 Introdução	1
1.1 Organização da Dissertação	3
2 Virtualização de Redes IEEE 802.11	4
2.1 Técnicas de Virtualização de Redes sem fio	4
2.1.1 Divisão por Frequência	5
2.1.2 Divisão por Tempo	5
2.1.3 Combinação da Divisão por Tempo e Frequência	6
2.1.4 Salto de Frequência	7
2.1.5 Divisão por Código	8
2.2 Utilizando virtualização em redes IEEE 802.11	9
2.2.1 Algoritmo de virtualização FH em redes IEEE 802.11	9
2.2.2 Algoritmo de virtualização TDMA em redes IEEE 802.11	10
2.3 Conclusão do Capítulo	11
3 Transição Rápida entre Pontos de Acesso	12
3.1 Funcionamento da Transição entre Pontos de Acesso	12
3.2 Protocolos de Transição Rápida entre Pontos de Acesso	13
3.2.1 CAPWAP	14
3.2.2 HOKEY	15
3.2.3 IEEE 802.11r	16
3.3 Cenário de Utilização dos Protocolos de Transição Rápida	18
3.4 Conclusão do Capítulo	19

4	Algoritmos para Balanceamento de Carga	20
4.1	Tipos de Algoritmos para Balanceamento de Carga	20
4.1.1	Balanceamento Baseado nas Estações	20
4.1.2	Balanceamento Baseado na Rede	21
4.2	Algoritmo de Virtualização para Balanceamento de Carga	22
4.2.1	Algoritmo Estático - Modo 0	26
4.2.2	Algoritmo com Ocupação do Meio - Modo 1	26
4.2.3	Algoritmo com Ocupação do Meio e Limiar de Aceitação para RSSI - Modo 2	27
4.2.4	Algoritmo com Taxa de Erro de Pacotes - Modo 3	28
4.2.5	Algoritmo com Vazão Média - Modo 4	28
4.3	Algoritmos da Literatura Implementados	29
4.3.1	Algoritmo de Seleção Padrão	29
4.3.2	Algoritmo DLBA	29
4.3.3	Algoritmo MLT	30
4.4	Conclusão do Capítulo	31
5	Resultados e Discussões	33
5.1	Simulador e Cenários de Simulação	33
5.2	Algoritmos de Balanceamento de Carga com Virtualização	34
5.2.1	Conclusão Parcial	39
5.3	Algoritmos da Literatura	39
5.3.1	Resultados do Algoritmo DLBA	39
5.3.2	Resultados do Algoritmo MLT	44
5.4	Comparação entre os Algoritmos de Balanceamento	49
5.5	Conclusão do Capítulo	52
6	Conclusões e Trabalhos Futuros	60
	Referências Bibliográficas	62

Lista de Figuras

2.1	Representação do funcionamento da técnica de virtualização FDMA. . .	6
2.2	Representação do funcionamento da técnica de virtualização TDMA. . .	6
2.3	Representação do funcionamento da técnica de virtualização TDMA combinada com a FDMA.	7
2.4	Representação do funcionamento da técnica de virtualização FH. . . .	8
2.5	Representação do funcionamento da técnica de virtualização CDMA. . .	9
2.6	Técnica de virtualização FH em redes IEEE 802.11 utilizando o protocolo PSM.	10
2.7	Técnica de virtualização TDMA com múltiplos rádios em redes IEEE 802.11.	11
3.1	Arquitetura do protocolo CAPWAP mostra a autenticação inicial com o sistema AAA seguido de uma transição entre WTPs.	15
3.2	Transição entre APs de AAAs diferentes utilizando o protocolo HOKEY. . .	17
3.3	Protocolo IEEE 802.11r executando o processo de transição entre APs. . .	18
4.1	Representação do esquema de virtualização FH utilizando fatias de tempo ponderadas para cada AP.	24
4.2	Escalonamento entre APs utilizando o protocolo IEEE 802.11r.	25
5.1	Gráficos da vazão do algoritmo com virtualização utilizando o protocolo IEEE 802.11b.	35
5.2	Gráficos da vazão do algoritmo com virtualização utilizando o protocolo IEEE 802.11g.	36
5.3	Gráficos do índice de justiça do algoritmo com virtualização utilizando o protocolo IEEE 802.11b.	37
5.4	Gráficos do índice de justiça do algoritmo com virtualização utilizando o protocolo IEEE 802.11g.	38
5.5	Gráficos da latência das páginas <i>web</i> das estações comuns do algoritmo com virtualização utilizando o protocolo IEEE 802.11b.	38
5.6	Gráficos da latência das páginas <i>web</i> das estações comuns do algoritmo com virtualização utilizando o protocolo IEEE 802.11g.	39

5.7	Gráficos da vazão do algoritmo DLBA utilizando o protocolo IEEE 802.11b.	40
5.8	Gráficos da vazão do algoritmo DLBA utilizando o protocolo IEEE 802.11g.	41
5.9	Gráficos do índice de justiça do algoritmo DLBA utilizando o protocolo IEEE 802.11b.	42
5.10	Gráficos do índice de justiça do algoritmo DLBA utilizando o protocolo IEEE 802.11g.	43
5.11	Gráficos da latência das páginas <i>web</i> das estações comuns do algoritmo DLBA utilizando o protocolo IEEE 802.11b.	43
5.12	Gráficos da latência das páginas <i>web</i> das estações comuns do algoritmo DLBA utilizando o protocolo IEEE 802.11g.	44
5.13	Gráficos da vazão do algoritmo MLT utilizando o protocolo IEEE 802.11b.	45
5.14	Gráficos da vazão do algoritmo MLT utilizando o protocolo IEEE 802.11g.	46
5.15	Gráficos do índice de justiça do algoritmo MLT utilizando o protocolo IEEE 802.11b.	47
5.16	Gráficos do índice de justiça do algoritmo MLT utilizando o protocolo IEEE 802.11g.	48
5.17	Gráficos da latência das páginas <i>web</i> das estações comuns do algoritmo MLT utilizando o protocolo IEEE 802.11b.	48
5.18	Gráficos da latência das páginas <i>web</i> das estações comuns do algoritmo MLT utilizando o protocolo IEEE 802.11g.	49
5.19	Gráficos da vazão dos algoritmos desenvolvidos sem estações comuns e utilizando o protocolo IEEE 802.11b.	50
5.20	Gráficos da vazão dos algoritmos desenvolvidos com 30 estações comuns e utilizando o protocolo IEEE 802.11b.	51
5.21	Gráficos da vazão dos algoritmos desenvolvidos sem estações comuns e utilizando o protocolo IEEE 802.11g.	52
5.22	Gráficos da vazão dos algoritmos desenvolvidos com 30 estações comuns e utilizando o protocolo IEEE 802.11g.	53
5.23	Gráficos do índice de justiça dos algoritmos desenvolvidos sem estações comuns e utilizando o protocolo IEEE 802.11b.	54
5.24	Gráficos do índice de justiça dos algoritmos desenvolvidos com 30 estações comuns e utilizando o protocolo IEEE 802.11b.	55
5.25	Gráficos do índice de justiça dos algoritmos desenvolvidos sem estações comuns utilizando o protocolo IEEE 802.11g.	56

5.26	Gráficos do índice de justiça dos algoritmos desenvolvidos com 30 estações comuns e utilizando o protocolo IEEE 802.11g.	57
5.27	Gráficos da latência das páginas <i>web</i> das estações comuns dos algoritmos desenvolvidos utilizando o protocolo IEEE 802.11b.	58
5.28	Gráficos da latência das páginas <i>web</i> das estações comuns dos algoritmos desenvolvidos utilizando o protocolo IEEE 802.11g.	59

Lista de Tabelas

4.1	Resumo dos algoritmos de balanceamento de carga utilizados e suas respectivas métricas.	32
5.1	Amostra de resultados numéricos atingidos pelos algoritmos de virtualização.	40

Lista de Símbolos

$ARSSI_y$	Média do RSSI das estações associadas ao AP y , p. 30
$DM_x(AP_y)$	Desocupação que a estação x calcula para o AP_y , p. 27
$DR_x(y)$	Diferença entre o RSSI da estação x e a média do RSSI que o AP y tem com as estações associadas, p. 30
$N_{APs}(x)$	Quantidade de APs alcançáveis pela estação x , p. 26
OC_y	Ocupação do meio média medida no AP_y através protocolo IEEE 802.11k, p. 27
$P_x(AP_y)$	Peso associado à interface virtual da estação x , p. 23
T_{CDMA}	Tempo de ciclo de virtualização gasto pela técnica CDMA, p. 9
T_{FDMA}	Tempo de ciclo de virtualização gasto pela técnica FDMA, p. 5
T_{FH}	Tempo de ciclo de virtualização gasto pela técnica FH, p. 8
T_{TDMA}	Tempo de ciclo de virtualização gasto pela técnica TDMA, p. 5
T_{TFDMA}	Tempo de ciclo de virtualização gasto pela técnica que combina TDMA com FDMA, p. 7
ms	Milissegundos, p. 5
n_{freq}	Quantidade total de frequências particionadas, p. 7, 8
n_{virt}	Quantidade total de interfaces virtuais, p. 5
t_{ch}	Tempo gasto pela troca de canais em uma interface IEEE 802.11, p. 5

t_{ct}	Tempo gasto na troca de contexto pelo sistema operacional, p. 5
t_{decode}	Tempo gasto pela técnica CDMA para decodificar um sinal, p. 9
t_{virt}	Tempo que uma interface virtual fica ativa, p. 5
$Gbps$	<i>Gigabits</i> por segundo, p. 34

Lista de Abreviaturas

AAA	<i>Authentication, Authorization and Accounting</i> , p. 14
AALP	<i>Avoiding APs with Larger PER</i> , p. 31
AC	<i>Access Controller</i> , p. 14
AP	<i>Access Point</i> , p. 1
ARF	<i>Auto Rate Fallback</i> , p. 33
CAPWAP	<i>Control and Provisioning of Wireless Access Point</i> , p. 14
CDMA	<i>Code Division Multiple Access</i> , p. 8
DLBA	<i>Dynamic Load Balance Algorithm</i> , p. 29
EAP	<i>Extensible Authentication Protocol</i> , p. 13
ERP	<i>EAP Re-authentication Protocol</i> , p. 15
ESS	<i>Extended Service Set</i> , p. 33
FDMA	<i>Frequency Division Multiple Access</i> , p. 5
FH	<i>Frequency Hopping</i> , p. 7
FTP	<i>File Transfer Protocol</i> , p. 33
HOKEY	<i>Handover Keying</i> , p. 14
HTTP	<i>HyperText Transfer Protocol</i> , p. 33
IAPP	<i>Inter Access Point Protocol</i> , p. 19
IEEE	<i>Institute of Electrical and Electronic Engineers</i> , p. 1
IETF	<i>Internet Engineering Task Force</i> , p. 13
MAC	<i>Controle de Acesso ao Meio - Medium Access Control</i> , p. 13

MLT	<i>Maximizing Load Throughput</i> <i>Maximizing Load Throughput</i> , p. 30
NS-2	<i>Network Simulator</i> versão 2.34, p. 33
PER	Taxa de Erro de Pacote – <i>Package Error Ratio</i> , p. 25
PHY	Camada Física - <i>Physical Layer</i> , p. 14
PMK	<i>Pairwise Master Key</i> , p. 16
PSM	<i>Power Saving Mode</i> , p. 9
R0KH	<i>R0 Key Holder</i> , p. 16
SINR	<i>Signal to Interference-plus-Noise Ratio</i> , p. 1
TDMA	<i>Time Division Multiple Access</i> , p. 5
UML	<i>User Mode Linux</i> , p. 10
WLANs	<i>Wireless Local Area Networks</i> , p. 1
WTP	<i>Wireless Termination Point</i> , p. 14
dei80211mr	Biblioteca de Adaptação de Taxa do protocolo IEEE 802.11 da Universidade de Padova, p. 33

Capítulo 1

Introdução

Atualmente, há uma grande disponibilidade de pontos de acesso (APs – *Access Points*) sem fio IEEE 802.11 [1], tanto em redes privadas como em serviços de acesso público, os quais são encontrados em cafeterias, hotéis, restaurantes, livrarias, aeroportos, dentre outros locais. Tradicionalmente nas redes IEEE 802.11 infraestruturadas o AP com maior relação sinal-ruído é o escolhido pela estação cliente, o qual permanece conectada ao mesmo até que a conectividade seja perdida ou o usuário cliente decida encerrar a conexão. Essa abordagem tradicional pode oferecer ao usuário cliente um serviço ruim em vários aspectos, como, por exemplo, congestionamento e vazão reduzida.

Na especificação tradicional do padrão IEEE [1], as estações clientes 802.11 utilizam a relação sinal-ruído (SINR – *Signal to Interference-plus-Noise Ratio*) de uma resposta de sondagem para escolher em qual AP devem se associar. A sondagem é feita em todos os canais disponíveis, e o AP que tiver enviado a resposta com maior relação sinal-ruído será eleito para que a associação seja realizada. Conforme mostrado em BEJERANO *et al.* [2], este tipo de associação não garante o uso eficiente dos recursos da rede, e pode levar estações a obter baixo desempenho em termos de vazão.

Alguns estudos em redes locais sem fio (WLANs – *Wireless Local Area Networks*) como BALACHANDRAN *et al.* [3], KOTZ e ESSIEN [4], BALAZINSKA e CASTRO [5] têm mostrado que a carga do tráfego gerado é geralmente mal distribuída entre os APs. Como por padrão as estações clientes 802.11 fazem a seleção do AP pela maior relação sinal-ruído, essa abordagem negligencia a condição de carga do AP. Como as estações clientes, tipicamente, são distribuídas no espaço de forma não uniforme, alguns APs tendem a sofrer com uma carga alta, enquanto seus APs vizinhos têm uma carga baixa ou estão desocupados. Nesse contexto, o desbalanceamento de carga entre os APs ocorre, evitando que a rede possa oferecer serviços justos às estações clientes, além de causar eventuais congestionamentos na rede [6].

Para resolver esses problemas de desbalanceamento, várias pesquisas já foram

realizadas. A abordagem mais utilizada para solucionar o problema, inicialmente abordada em BALACHANDRAN *et al.* [3], BALAZINSKA e CASTRO [5], SHEU e WU [7], PAPANIKOS e LOGOTHETIS [8], é balancear a carga dos APs utilizando uma seleção controlada da associação entre estação cliente e AP, este processo é conhecido como controle de associação. O controle de associação pode ser utilizado para alcançar diferentes objetivos, como por exemplo, maximizar a vazão do sistema como um todo, mudar associações de estações clientes para APs com menos carga ou desocupados, e além disso, permitir que cada AP sirva as estações clientes com a taxa de transmissão máxima. Entretanto, maximizar a vazão não é o comportamento desejado para o sistema que pretende balancear a carga, do ponto de vista de justiça da rede. Portanto, para que o sistema seja balanceado e justo, o maior objetivo é dividir a banda da rede de forma justa entre as estações clientes, essa política de compartilhamento é conhecida como *max-min fairness*.

Afim de solucionar o problema de balanceamento de carga em redes infraestruturadas, esta dissertação propõe uma solução utilizando virtualização de rede sem fio. Esta proposta permite que cada estação cliente usufrua dos recursos da rede de forma justa utilizando virtualização de interfaces físicas de rede. Para que se possa virtualizar o meio sem fio existem algumas técnicas descritas em PAUL e SESHAN [9], as quais permitem compartilhar os recursos da rede entre diferentes experimentos virtuais, ou seja, entre interfaces físicas diferentes ou até mesmo entre interfaces virtuais em uma única interface física.

Fundamentalmente, a virtualização é um conceito de abstração que esconde detalhes, lidando com a complexidade e a heterogeneidade, e por isso, oferece um certo nível de indireção (*indirection*) e compartilhamento de recursos. Dessa forma, permitindo flexibilidade e reuso dos recursos físicos, resultando no aumento da eficiência. Além de características como programabilidade, permitir topologia mais complexas, configuração rápida, isolamento dos recursos e abstração da rede. Há vários estudos que utilizaram a virtualização para solucionar problemas de recursos de redes sem fio [10–14].

A solução de virtualização de redes sem fio utilizada nesta dissertação sugere o compartilhamento de recurso da rede entre as interfaces virtuais. Nessa abordagem há uma interface virtual conectada com cada um dos APs da rede que esteja ao alcance da estação cliente. As interfaces virtuais são escalonadas de forma que cada uma permaneça um certo tempo conectada ao AP correspondente. O balanceamento da carga da rede é feito atribuindo um peso a cada uma dessas de acordo com uma das soluções propostas. Algumas soluções propostas nessa dissertação foram avaliadas de forma a saber quão justo o balanceamento de carga pode ser.

Para realizar o balanceamento de carga entre APs de uma rede foi utilizada a virtualização de interfaces física de rede. A virtualização de redes permite acesso

a diferentes redes através das interfaces virtuais, possibilita múltiplas associações, independência da tecnologia 802.11 e facilidade na gestão das camadas superiores. Para que a virtualização seja realizada, é preciso comutar entre as interfaces virtuais, mantendo-as ativas uma de cada vez durante um tempo específico. A proposta desta dissertação procura determinar o tempo que cada interface virtual permanece ativa para que a carga seja balanceada de forma justa.

Como a virtualização facilita a gestão das camadas superiores de rede através de interfaces virtuais, é possível realizar o balanceamento de carga de forma justa. A distribuição dos recursos da rede é feita de forma justa através da virtualização. As estações virtualizam suas interfaces físicas de rede sem fio para manter associações com todos os APs alcançáveis. Esta dissertação analisa soluções que procuram a melhor maneira de usar a virtualização para balancear carga em redes sem fio infraestruturadas. Os resultados das soluções desenvolvidas apresentam valores melhores que a implementação tradicional. Além disso, essas soluções devem manter compatibilidade com a implementação tradicional. Outra característica das soluções propostas, é o fato de não precisar modificar o protocolo 802.11, e nenhuma outra entidade de rede, a não ser a estação cliente para que se possa utilizar as soluções propostas. Simulações apresentam o quanto cada métrica atrapalha as estações tradicionais.

1.1 Organização da Dissertação

A seguir é apresentada uma breve descrição do conteúdo dos capítulos desta dissertação:

- Capítulo 2: Breve explanação do que são e como funcionam as técnicas de virtualização em redes sem fio.
- Capítulo 3: Explanação sobre alguns dos tipos de protocolos de *handoff* rápido.
- Capítulo 4: Apresentação do algoritmo de balanceamento de carga proposto nesta dissertação.
- Capítulo 5: Exibe os resultados obtidos comentando cada um deles.
- Capítulo 6: Breve conclusão do trabalho realizado nesta dissertação, seguido de comentários dos possíveis trabalhos futuros.

Capítulo 2

Virtualização de Redes IEEE 802.11

Neste capítulo serão apresentadas as técnicas de virtualização de redes sem fio definidas por PAUL e SESHAN [9], seguida de uma breve explicação da utilização das técnicas em redes IEEE 802.11.

2.1 Técnicas de Virtualização de Redes sem fio

Há um crescente reconhecimento por parte dos usuários de redes sem fio em relação aos benefícios da adição de interfaces físicas de rede. A utilização de múltiplas interfaces físicas de rede pode aumentar a banda efetiva através de diversos provedores [15], aliviando perdas através da diversidade do espectro [16], e melhora do gerenciamento da mobilidade através das rápidas transições entre APs [17]. Apesar dessas vantagens convincentes, contudo, estações com múltiplas interfaces físicas de rede continuam sendo uma exceção e não uma regra, segundo NICHOLSON *et al.* [14].

Virtualização está sendo amplamente utilizada em redes cabeadas [10, 18, 19], e nos últimos anos a virtualização em redes sem fio também vem sendo bastante abordada. As dificuldades encontradas em virtualização de redes sem fio são diferentes das existentes em redes cabeadas. O maior desafio desta abordagem é virtualizar o enlace sem fio [20]. Para que seja estabelecido um enlace sem fio é preciso que o par transmissor-receptor esteja configurado com os mesmos parâmetros de canal. Apesar dessas dificuldades, a virtualização de redes permite que uma única interface física seja utilizada para oferecer às camadas superiores conectividade em mais de uma rede.

Existem diferentes técnicas para se virtualizar uma rede sem fio. Dentre as definidas em PAUL e SESHAN [9], há as técnicas de divisão por tempo, frequência e código, além de outras que mesclam estas. A seguir serão apresentadas as técnicas de virtualização de redes sem fio.

2.1.1 Divisão por Frequência

A técnica de virtualização FDMA (*Frequency Division Multiple Access*) define que a interface física de rede é virtualizado particionando as frequências, ou seja, utilizando-se diferentes canais. Esta técnica pode ser aplicada em uma única interface física ou em múltiplas interfaces, cada uma em canais diferentes. Quando se utiliza uma única interface física, esta emula múltiplas interfaces virtuais. A Figura 2.1 ilustra esta técnica. A troca entre canais em uma interface física IEEE 802.11 tem um custo de tempo que depende do fabricante da placa, sendo de $5ms$ para a placa Atheros e $20ms$ para a placa da Intel [9]. Em RAMANI e SAVAGE [21] esse tempo de atraso na troca de canal é de $10ms$ em média. Existe também um custo temporal, relacionado à troca de contexto do sistema operacional, para que uma interface virtualizada fique ativa. Este tempo no linux encontra-se no intervalo entre 1 e $10ms$ [9]. Contudo, o tempo da troca de contexto e troca de canal, só são contabilizados caso se utilize interfaces virtuais em uma única interface física, caso contrário, não são contabilizados. A Equação 2.1 apresenta o custo total de tempo que a técnica FDMA gasta, este custo é o ciclo de virtualização, que representa a soma de todos os tempos ativos de todas as interfaces virtuais em um ciclo.

$$T_{FDMA} = n_{virt} \cdot (t_{ch} + t_{ct} + t_{virt}) \quad . \quad (2.1)$$

Onde T_{FDMA} é o tempo de ciclo gasto pela técnica FDMA, t_{ch_i} é o gasto pelo na troca de canais, t_{ct_i} é o tempo gasto na troca de contexto pelo sistema operacional, t_{virt_i} é o tempo que cada interface virtual fica ativa, ou seja, fica utilizando o recurso compartilhado, e n_{virt} é o total de interfaces virtuais.

2.1.2 Divisão por Tempo

TDMA (*Time Division Multiple Access*) é a técnica que virtualiza uma interface física de rede particionando-a ao longo do tempo, ou seja, diferentes interfaces virtuais utilizam uma dada frequência da interface física particionada em diferentes “fatias” (*slots*) de tempo. A Figura 2.2 representa a técnica de virtualização TDMA.

O tempo da troca de contexto do sistema operacional não é negligenciado, como já mencionado na Seção 2.1.1. A Equação 2.2 mostra o custo total de tempo que a técnica TDMA gasta no ciclo de virtualização.

$$T_{TDMA} = n_{virt} \cdot (t_{ct} + t_{virt}) \quad . \quad (2.2)$$

Onde T_{TDMA} é o tempo de ciclo gasto pela técnica TDMA, t_{ct} é o tempo gasto na troca de contexto pelo sistema operacional, t_{virt} é o tempo que cada interface virtual fica ativa e n_{virt} é o total de interfaces virtuais.

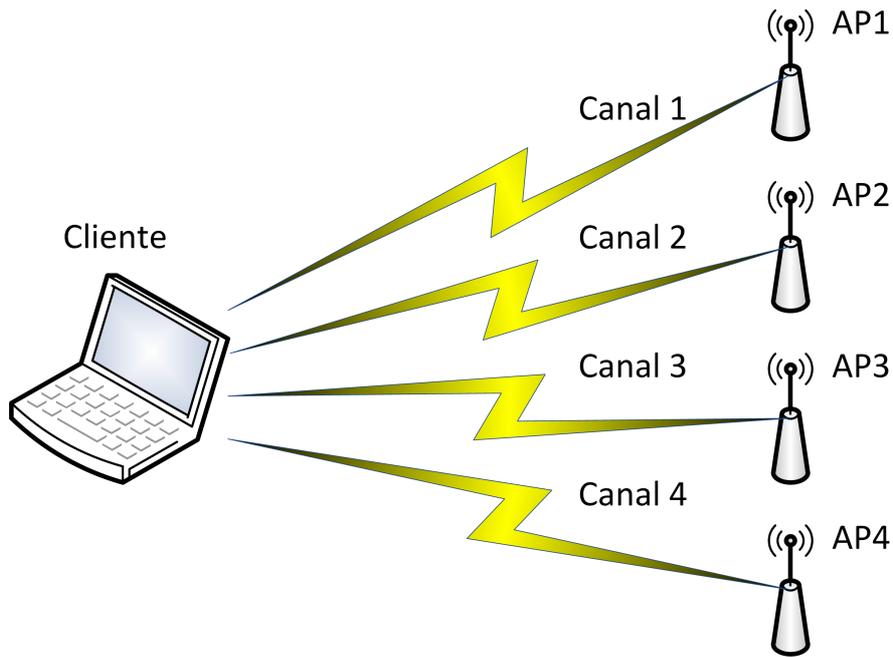


Figura 2.1: Representação do funcionamento da técnica de virtualização FDMA.

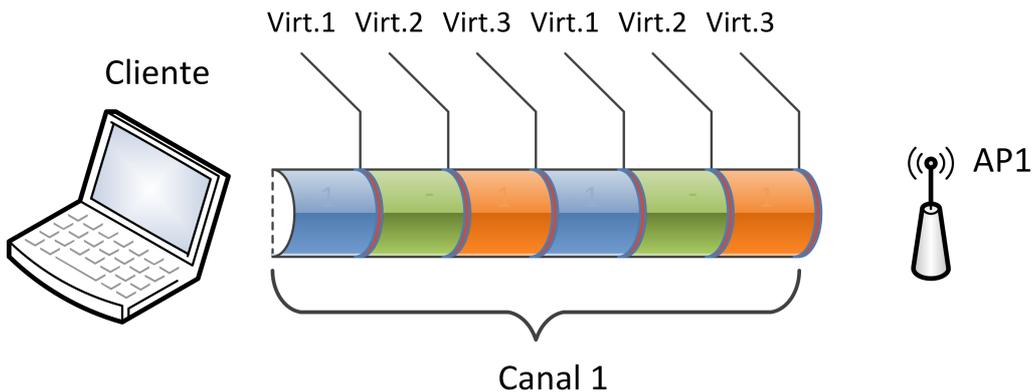


Figura 2.2: Representação do funcionamento da técnica de virtualização TDMA.

2.1.3 Combinação da Divisão por Tempo e Frequência

A combinação da técnica TDMA com a FDMA admite que uma interface física de rede sem fio possa ser virtualizada permitindo que diferentes interfaces virtuais utilizem frequências particionadas em diferentes fatias de tempo. Assim, uma interface virtual é identificada por uma única combinação de frequência particionada e fatia de tempo. A Figura 2.3 demonstra a técnica de virtualização TDMA combinada com a FDMA.

O tempo do ciclo de virtualização dessa técnica é definido na Equação 2.3. Na Seção 2.1.1 explica que não se negligencia a troca de contexto do sistema operacional

e nem de canais.

$$T_{TFDMA} = n_{freq} \cdot n_{virt} \cdot (t_{ch} + t_{ct} + t_{virt}) \quad . \quad (2.3)$$

Onde T_{TFDMA} é o tempo de ciclo gasto pela técnica TDMA e FDMA combinadas, t_{ct} é o tempo gasto pela troca de contexto, t_{ch} é o tempo gasto na troca de canal, t_{virt} é o tempo que cada interface virtual fica ativa (este valor é uma constante, todas as interfaces têm o mesmo tempo ativo), n_{virt} é o total de interfaces virtuais em uma frequência particionada e n_{freq} é o total de frequências particionadas.

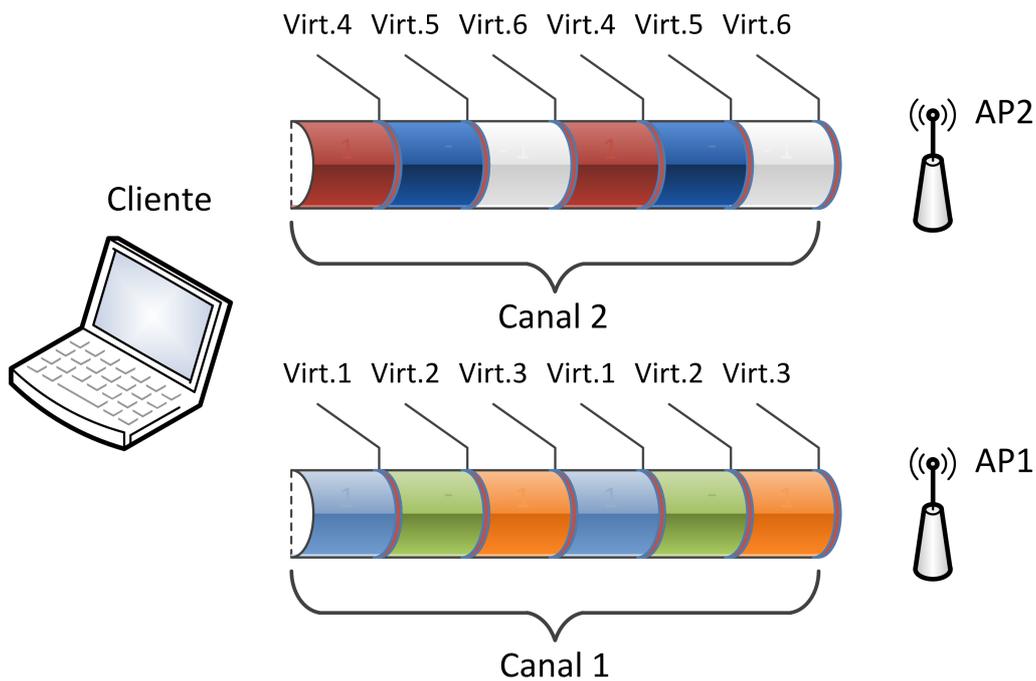


Figura 2.3: Representação do funcionamento da técnica de virtualização TDMA combinada com a FDMA.

2.1.4 Salto de Frequência

A técnica de virtualização FH (*Frequency Hopping*) pode fazer uma interface física de rede sem fio ser virtualizada de forma a permitir que diferentes interfaces virtuais usem diferentes frequências particionadas em diferentes fatias de tempo. A Figura 2.4 representa a técnica FH. Então, a virtualização da rede utilizando essa técnica consiste em um conjunto de nós cujas interfaces virtuais podem usar uma sequência pré-definida de frequência e fatia de tempo. Essa sequência se repete ciclicamente durante o ciclo de virtualização, onde o tempo do ciclo de virtualização

da técnica FH é definido pela Equação 2.4.

$$T_{FH} = \sum_{i=1}^{n_{freq}} \left[n_{virt_i} \cdot (t_{ch} + t_{ct}) + \sum_{j=1}^{n_{virt_i}} (t_{virt_{ij}}) \right] . \quad (2.4)$$

Onde T_{FH} é o tempo de ciclo gasto pela técnica FH, t_{ct} é o tempo gasto pela troca de contexto, t_{ch} é o tempo gasto na troca de canal, $t_{virt_{ij}}$ é o tempo que cada interface virtual fica ativa (este valor é variável, todas as interfaces têm valores diferentes), n_{virt_i} é o total de interfaces virtuais em uma frequência particionada e n_{freq} é o total de frequências particionadas.

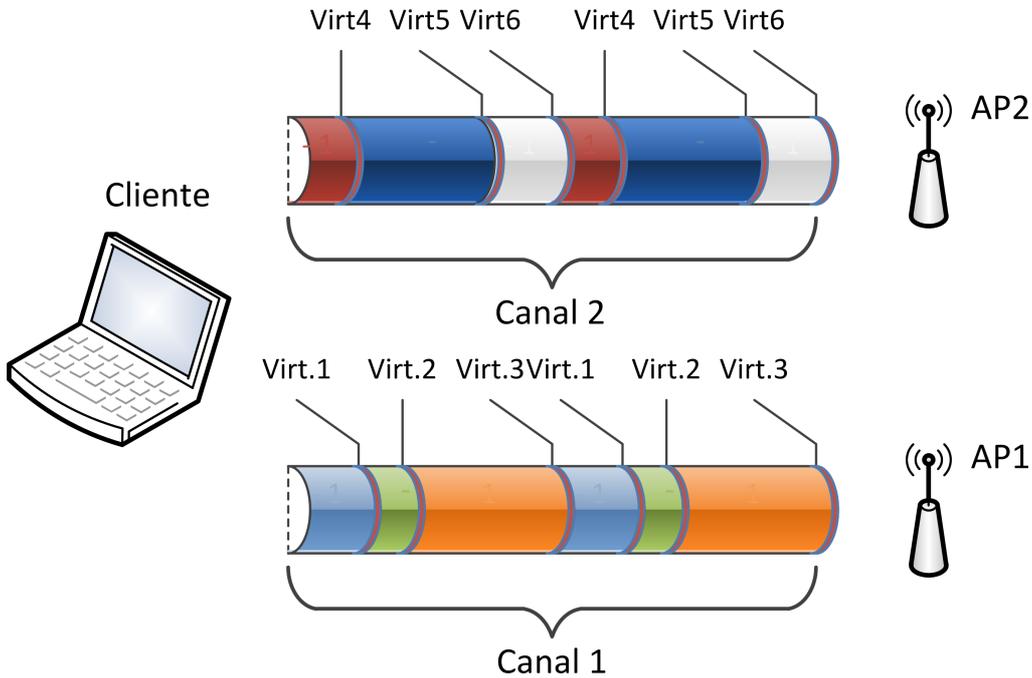


Figura 2.4: Representação do funcionamento da técnica de virtualização FH.

Salto de Frequência vs. Combinação da Divisão de Tempo e Frequência

Diferentemente do esquema que combina TDMA com FDMA, o FH permite que a mesma interface virtual possa utilizar diferentes sequências de frequência e fatia de tempo. Essa diferença faz a técnica FH ser mais escalável que a técnica que combina TDMA com FDMA [9].

2.1.5 Divisão por Código

A técnica CDMA (*Code Division Multiple Access*) virtualiza a interface física de rede sem fio, particionando-a em codificações ortogonais. Este esquema faz a troca de códigos dentro de uma interface física para emular múltiplas interfaces virtuais. A

Figura 2.5 apresenta o funcionamento desta técnica. O tempo do ciclo desta técnica é apenas a soma dos tempos ativos das interfaces virtuais, como apresentado pela Equação 2.5.

$$T_{CDMA} = n_{virt} \cdot (t_{decode} + t_{virt}) \quad . \quad (2.5)$$

Onde T_{CDMA} é o tempo de ciclo gasto pela técnica CDMA, t_{decode} é o tempo gasto para decodificar o sinal, t_{virt} é o tempo que cada interface virtual fica ativa (valor é constante) e n_{virt} é o total de interfaces virtuais.



Figura 2.5: Representação do funcionamento da técnica de virtualização CDMA.

2.2 Utilizando virtualização em redes IEEE 802.11

A virtualização de uma interface IEEE 802.11 pode ser feita através das diferentes técnicas apresentadas na Seção 2.1. Essa tecnologia é utilizada para otimizar o uso dos recursos da rede sem fio, portanto, a melhor técnica a se utilizar depende do problema a ser abordado. Em CHANDRA e BAHL [12] e NICHOLSON *et al.* [14] é utilizada uma solução de virtualização para redes infraestruturadas e *ad hoc* que utiliza a técnica FH. Também em RAMANI e SAVAGE [21] é utilizada a técnica FH, mas aborda apenas redes infraestruturadas. Já em SMITH *et al.* [20] a técnica de virtualização aplicada foi a TDMA, que utiliza múltiplos rádios em um escalonamento *Round-Robin*.

2.2.1 Algoritmo de virtualização FH em redes IEEE 802.11

A virtualização de redes IEEE 802.11 sugerida por CHANDRA e BAHL [12], NICHOLSON *et al.* [14] e RAMANI e SAVAGE [21] faz uso do protocolo PSM (*Power Saving Mode*). Para que o cliente possa virtualizar sua interface de rede, o anúncio do estado de economia de energia é ativado pelo protocolo PSM. A partir desse

momento, o AP passa a armazenar os pacotes de dados desse cliente, que ativou uma nova interface virtual, ou seja, está com uma outra rede ativa. Mantendo sempre uma rede ativa e as outras em estado de economia de energia, esse algoritmo consegue virtualizar redes IEEE 802.11. A Figura 2.6 representa o funcionamento da técnica de virtualização FH utilizando o estado de economia de energia. O passo 1 da Figura 2.6 representa o aviso da estação ao AP1 de que ela saiu do modo de economia de energia e voltou a ficar ativa. No passo 2, a estação passa a trocar dados. Já no passo 3, a estação envia uma mensagem de que irá entrar em modo de economia de energia, e nesse momento realiza uma transição para o AP2. Os passos 4, 5 e 6 são iguais aos passos 1, 2 e 3, respectivamente.

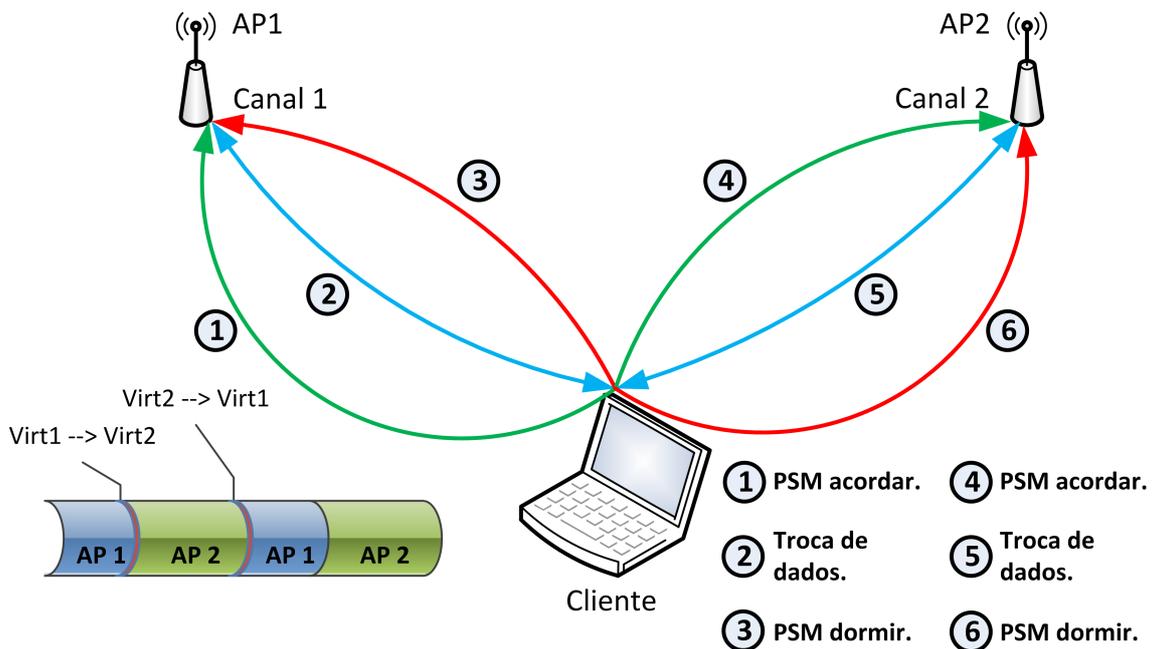


Figura 2.6: Técnica de virtualização FH em redes IEEE 802.11 utilizando o protocolo PSM.

2.2.2 Algoritmo de virtualização TDMA em redes IEEE 802.11

A técnica TDMA em redes IEEE 802.11 aplicada por SMITH *et al.* [20] utiliza múltiplas interfaces físicas. Cada interface física de uma estação cliente virtualiza uma interface virtual, e cada interface virtual pertence a uma rede. As interfaces virtuais de uma estação cliente são virtualizadas pela plataforma UML (*User Mode Linux*) [19], e um escalonamento *Round-Robin* é realizado para comutar o acesso entre as interfaces. A Figura 2.7 representa o funcionamento da virtualização utilizando a técnica TDMA para virtualizar uma rede IEEE 802.11. Nessa técnica TDMA tem que haver sincronismo entre as estações da rede [20]. Caso as estações

não estejam sincronizadas, as redes virtualizadas vão perder pacotes. A sincronização pode ser realizado via NTP, por exemplo.

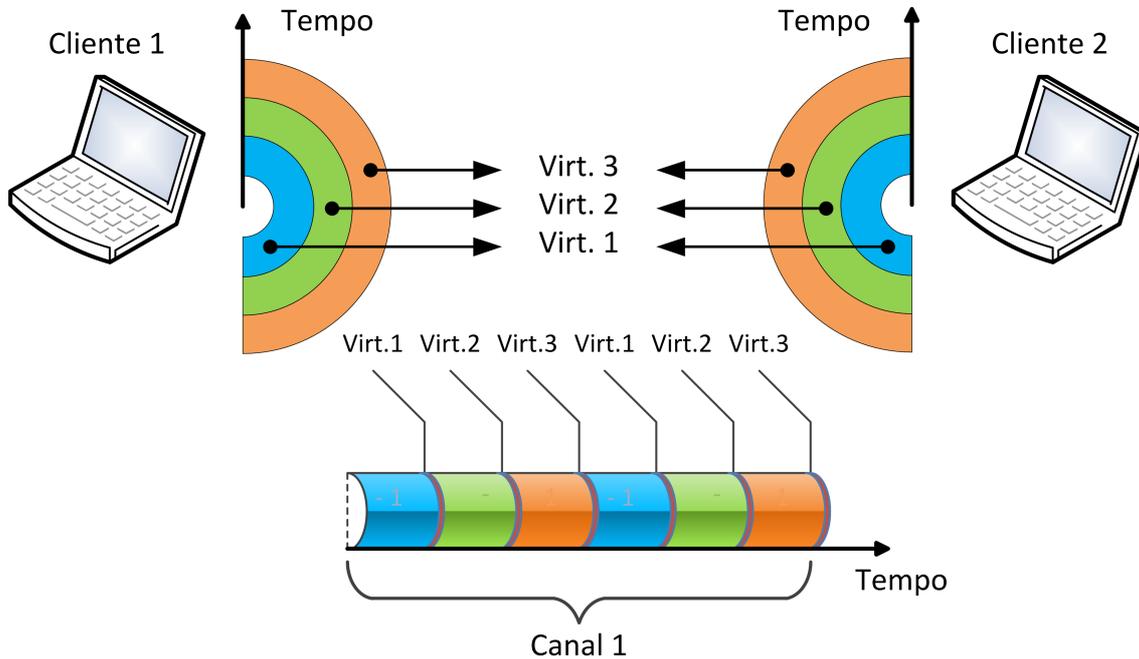


Figura 2.7: Técnica de virtualização TDMA com múltiplos rádios em redes IEEE 802.11.

2.3 Conclusão do Capítulo

Nesse capítulo foram apresentadas as técnicas de virtualização de redes sem fio. Essas foram definidas pelo GENI em PAUL e SESHAN [9] e alguns trabalhos já haviam utilizado algumas dessas técnicas. Em redes IEEE 802.11 os principais trabalhos relacionados com virtualização são PAUL e SESHAN [9], CHANDRA e BAHL [12], NICHOLSON *et al.* [14], SMITH *et al.* [20], RAMANI e SAVAGE [21]. Dentre as técnicas apresentadas, a mais flexível é a técnica FH, por isso, esta foi utilizada no desenvolvimento desta dissertação. No próximo capítulo será apresentado o funcionamento da transição de pontos de acesso em redes infraestruturadas IEEE 802.11.

Capítulo 3

Transição Rápida entre Pontos de Acesso

Neste capítulo, serão apresentados alguns protocolos de transição rápida entre APs, seguido de uma explicação diferenciando as abordagens descritas.

3.1 Funcionamento da Transição entre Pontos de Acesso

A transição entre APs em redes IEEE 802.11 infraestruturadas é o processo de mudança de um AP para um outro, conhecido como *roaming*. Esse processo, tradicionalmente, é ativado quando a estação começa a perceber perdas de dados com o AP a qual está associado. O processo de transição é iniciado no momento que a estação se desconecta do AP a qual mantinha conexão, a partir daí, a procura por um novo AP para se conectar é realizada, e quando um novo AP é encontrado estabelece-se uma conexão, e o processo de transição é concluído.

Há alguns estudos propondo formas diferentes para saber de forma mais eficaz o momento mais oportuno de ativar a transição entre APs, como por exemplo RAMANI e SAVAGE [21], VELAYOS e KARLSSON [22]. Em VELAYOS e KARLSSON [22] a transição é iniciada após a estação cliente perceber três perdas consecutivas de pacotes de dados do AP a qual se mantém associado. Já em RAMANI e SAVAGE [21], a melhor oportunidade acontece quando a diferença entre a SINR do AP a qual está associado e os APs alcançáveis chega a um certo limiar, nesse momento é iniciada a transição entre APs.

A busca por um novo AP no processo transição entre APs, é realizada pela escolha da maior potência de sinal recebida, processo especificado na Seção 11.8 do padrão [1]. O processo de busca é realizado através de amostras coletadas dos *Beacons* dos APs. As redes IEEE 802.11 infraestruturadas possuem dois tipos de

busca, Ativa e Passiva. A busca Ativa é realizada através do envio de um pacote de *Probe Request* em cada canal disponível na rede. A resposta dos APs a esses pacotes são coletadas pela estação que iniciou a busca Ativa. Já a busca Passiva, ao contrário da busca Ativa, apenas espera o recebimento dos pacotes de *Beacon* enviados pelos APs, esse processo de espera também é feito em cada canal disponível na rede. É fácil perceber que o tempo gasto na busca Ativa é inferior ao da busca Passiva. Um estudo realizado por RAMANI e SAVAGE [21] realiza a busca Ativa ou Passiva apenas nos canais ortogonais, ou seja, se aproveita da superposição de canais adjacentes para receber também *Beacons* de APs que não pertencem aos canais ortogonais, dessa forma o tempo das buscas é reduzido. Em RIZVI *et al.* [23], é proposto um modelo matemático para detecção e prevenção de colisão, assim como a fase de procura. Com esse modelo matemático utilizando uma busca Ativa por APs, RIZVI *et al.* [23] conseguiu diminuir o tempo de transição entre APs.

O resultado final do processo de busca por um AP é utilizado no processo de autenticação da estação com o AP escolhido. O processo de autenticação é iniciado então, e o protocolo mais utilizado para isso é o EAP (*Extensible Authentication Protocol*) [24], que por sua vez faz as trocas de mensagens via *four-Way handshake*. Quando o processo de autenticação chega ao fim, o de associação inicia, e as trocas de mensagens para que a estação cliente possa se associar ao novo AP é iniciada. Logo que a estação cliente termina a associação, é realizada uma atualização da rede por parte do novo AP, que envia uma mensagem de *gratuitous ARP* para a rede. Essa mensagem atualiza a tabela do Roteador/*Switch* com o MAC (Controle de Acesso ao Meio - *Medium Access Control*) da estação cliente e a nova porta associada a este. Nesse ponto, o fim do processo de transição entre APs é finalizado.

Algumas desvantagens no processo de transição entre APs são o tempo gasto para realizar a autenticação com o novo AP e o tempo que se leva para se fazer a varredura dos canais à procura de APs [25]. Estudos como RAMANI e SAVAGE [21], MISHRA *et al.* [26], WANG *et al.* [27], ZHANG e PIERRE [28] propõem soluções para diminuir o tempo gasto durante a varredura dos canais. Para tornar mais eficaz a fase de autenticação, alguns protocolos foram desenvolvidos, a Seção 3.2 apresenta os mais importantes.

3.2 Protocolos de Transição Rápida entre Pontos de Acesso

De forma a tornar mais eficiente a transição entre APs, mais especificamente o processo de autenticação, alguns protocolos foram criados. Dentro do IETF (*Internet Engineering Task Force*), o grupo de trabalho CAPWAP (*Control and Provisioning*

of *Wireless Access Point*) desenvolvem um protocolo que suporta gerenciamento distribuído em redes WLAN empresarial, e um outro que realiza uma transição rápida entre APs [29, 30]. O grupo HOKEY (*Handover Keying*) do IETF está desenvolvendo uma extensão da arquitetura AAA (*Authentication, Authorization and Accounting*) para suportar credenciais de segurança distribuídas e derivadas sem precisar de uma autenticação completa EAP [31]. Por último, o grupo 802.11r do IEEE desenvolveu um protocolo que suporta trocar credenciais diretamente entre APs quando uma estação cliente faz uma transição entre APs [32]. Os protocolos EAP e IEEE 802.11r foram apresentados por ZHENG e SARIKAYA [33], já os protocolos HOKEY, CAPWAP e IEEE 802.11r foram abordados por CLANCY [25], nestes estudos foram realizadas comparações e apresentados cenários em que cada protocolo se encaixa. A seguir, o funcionamento desses protocolos é melhor apresentado.

3.2.1 CAPWAP

O objetivo do CAPWAP é facilitar o gerenciamento de redes WLAN, de forma que todos os APs sejam gerenciáveis por um sistema de autorização central. Na arquitetura de rede do CAPWAP, o AP é dividido entre dois componentes lógicos. O primeiro é a implementação tradicional das camadas Físicas (PHY - *Physical layer*) e de Controle de Acesso ao Meio (MAC) do protocolo IEEE 802.11. O segundo componente que possui o CAPWAP propriamente dito, tem a camada PHY e a camada MAC de baixo nível implementadas no chamado WTPs (*Wireless Termination Points*), e a parte de alto nível da MAC é implementada por um controlador centralizado chamado de AC (*Access Controller*).

A autenticação e o controle de acesso do protocolo IEEE 802.11 ficam centralizada no AC, permitindo conectar clientes em qualquer WTP sem realizar uma re-autenticação EAP. Quando a estação cliente quer fazer uma transição para um novo WTP, antes de realizar a fase de busca, o AC executa um novo *four-way handshake* com o cliente e simplesmente entrega uma nova chave de tráfego sem derivação de uma nova chave de sessão. A Figura 3.1 apresenta a autenticação e o processo de transição de WTPs. O passo 1 da Figura 3.1 representa a autenticação EAP entre a estação cliente e o servidor AAA. O passo 2 representa a troca de chaves entre os servidores AC e AAA. O passo 3 apresenta o processo de *four-way handshake* entre a estação cliente e o servidor AC. O passo 4 apresenta a mensagem de adição da estação cliente, enviada pelo servidor AC para o WTP1, juntamente com a chave de tráfego. Quando a estação cliente decide realizar uma transição para outro WTP, representado pelo passo 5, como o servidor AC já possui a chave de tráfego para essa estação, é realizado apenas um processo de *four-way handshake* entre o WTP2

e o servidor AC. No passo 6 é feita a entrega da chave de tráfego para o *WTP2*. Os passos 7 e 8 são iguais aos passos 5 e 6, respectivamente, quando a estação cliente realiza uma transição do *WTP2* para o *WTP3*.

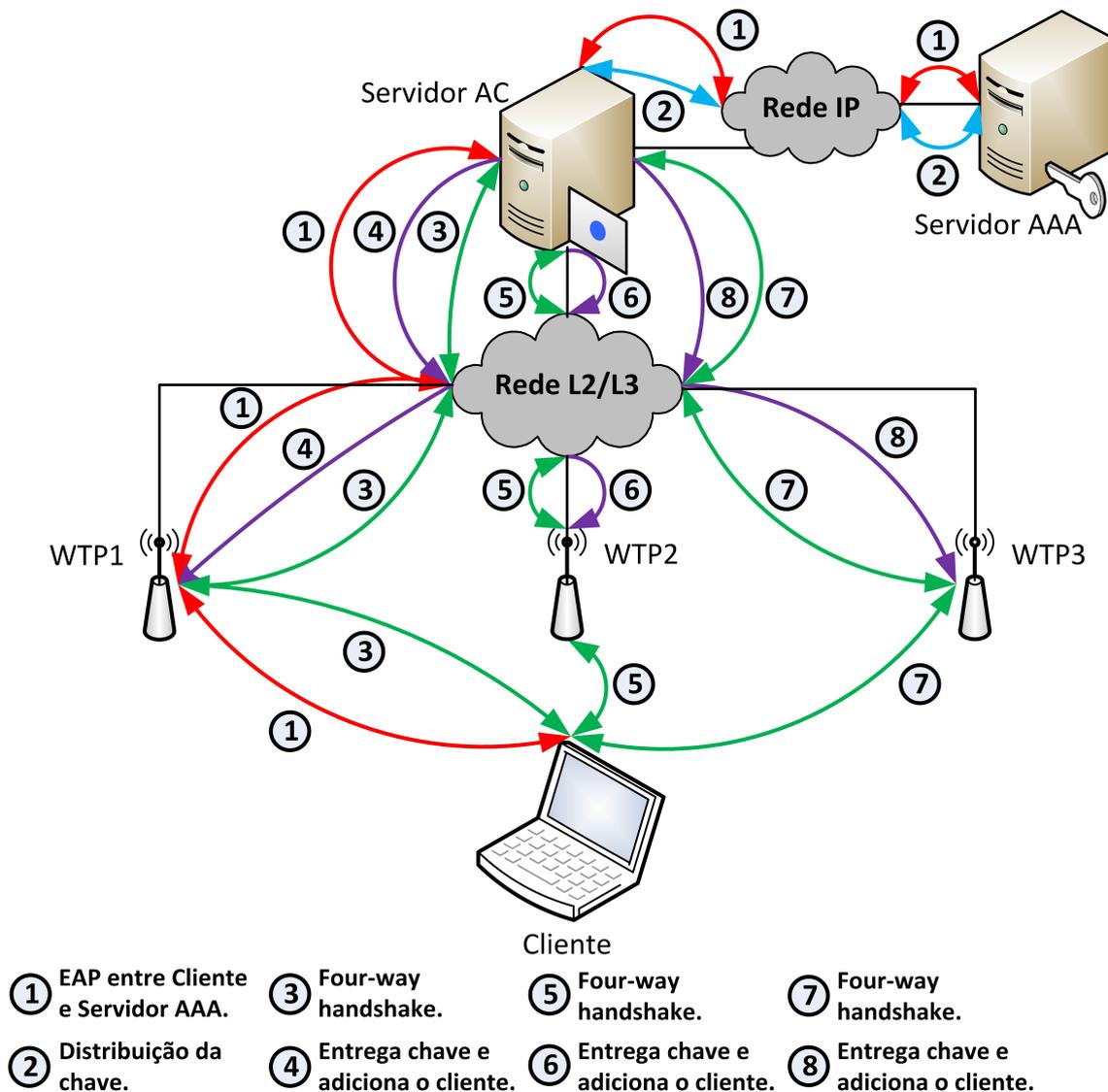


Figura 3.1: Arquitetura do protocolo CAPWAP mostra a autenticação inicial com o sistema AAA seguido de uma transição entre WTPs.

3.2.2 HOKEY

O protocolo HOKEY suporta além da transição rápida entre APs, a transição entre AAAs. Para isso, o HOKEY estendeu para o EAP o suporte nativo à associação rápida, conhecido por ERP (*EAP Re-authentication Protocol*) [34]. O ERP permite uma estação cliente utilizar chaves derivadas durante a autenticação inicial EAP, isso possibilita que a reassociação com um novo AP seja feita a partir de uma chave derivada e que aconteça com apenas uma troca de mensagens. No HOKEY, cada

AP da WLAN é um autenticador independente, e o ERP permite que os protocolos AAA sejam utilizados para gerar múltiplas chaves de sessões novas, sem executar novamente o método EAP original.

Outra característica do HOKEY é permitir que uma estação se associe a um AP de outra rede, caso essa rede tenha alguma relação com a rede original da estação, a autorização pode ser concedida para a realização da transição entre redes. Esse processo suporta a reautenticação rápida diminuindo drasticamente o tempo gasto durante a transição entre os APs. Durante a reautenticação, os protocolos AAAs são utilizados para passar as chaves do servidor AAA antigo para o servidor AAA da nova rede. A Figura 3.2 mostra o funcionamento do processo de transição entre APs de redes diferentes. O passo 1 da Figura 3.2 representa a autenticação EAP entre o servidor AAA nativo e a estação cliente. O passo 2 representa o envio da chave de tráfego do servidor AAA nativo para o AP1. O passo 3 apresenta o processo de *four-way handshake*. Quando a estação cliente vai realizar a transição do AP1 para o AP2 (que pertence a outro domínio), representado pelo passo 4, a estação inicia o processo de autenticação ERP com o servidor AAA local do novo domínio. O passo 5 representa o processo de pedido e recebimento da chave de tráfego de reautenticação entre os servidores AAA nativo e local. O passo 6 representa a geração da nova chave de tráfego para o novo domínio, e a envia para o AP2. O passo 7 representa o processo de *four-way handshake* entre o AP2 e a estação cliente.

3.2.3 IEEE 802.11r

No protocolo IEEE 802.11r, o primeiro AP que a estação cliente se autentica irá armazenar a chave PMK (*Pairwise Master Key*), esta é utilizada para derivar chaves de sessão para outros APs do mesmo domínio. Esse AP é chamado de R0KH (*R0 Key Holder*) que possui a chave PMK-R0, que simboliza o nível 0 da hierarquia PMK. Quando a estação faz a transição entre APs, o R0KH irá gerar uma nova chave PMK-R1, o qual deriva da chave de sessão PMK-R0 que é encaminhada para o novo AP, e este passa a ser nomeado de R1KH (*R1 Key Holder*). Para assegurar a ortogonalidade, o R0KH usa a segunda metade do PMK, não utilizada pelo padrão *four-way handshake*.

O processo de transição no protocolo IEEE 802.11r consiste, inicialmente, em uma troca de mensagens entre a estação cliente e o AP o qual está associada, que sinaliza a transição entre APs. Em seguida, há uma troca de mensagens entre a estação e o novo AP, que confirma se a chave foi entregue e derivada. Nesse momento também é feito a troca de chaves de tráfego. A Figura 3.3 mostra como são realizadas as trocas de mensagens no protocolo IEEE 802.11r. O passo 1 da Figura 3.3 representa a autenticação EAP entre a estação cliente e o servidor AAA.

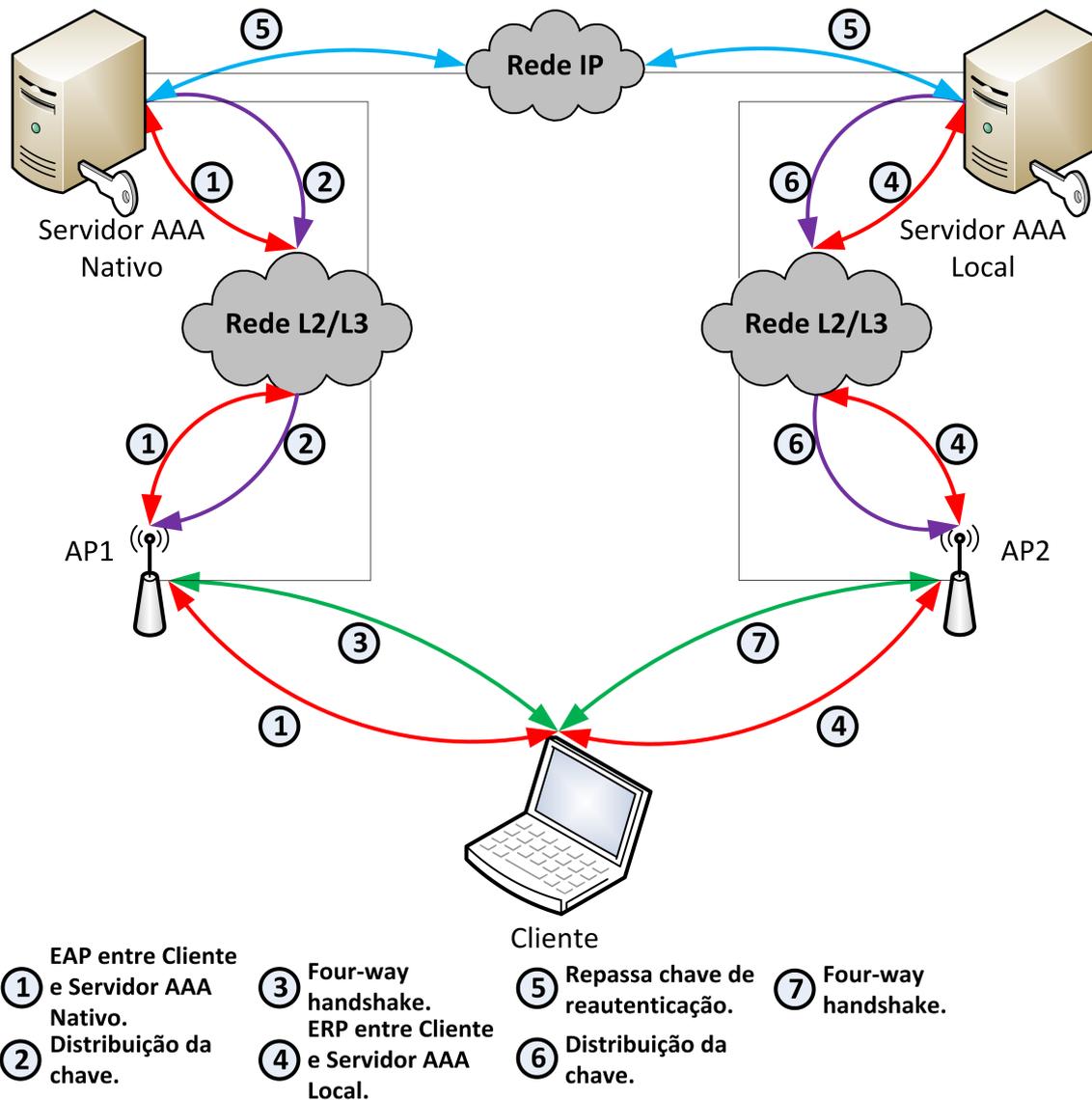


Figura 3.2: Transição entre APs de AAAs diferentes utilizando o protocolo HOKEY.

O passo 2 representa a geração da chave $R0KH$ para o $AP0$, e o envio desta para o $AP0$. O passo 3 representa o processo de *four-way handshake* entre o $AP0$ e a estação. Quando a estação cliente vai realizar a transição do $AP0$ para o $AP1$, representada passo 4, a estação faz uma requisição para pode realizar a transição. O passo 5 é a geração da chave $R1KH$ para a autenticação da estação cliente no $AP2$, e a envia para o $AP2$. O passo 6 é a finalização do processo de transição entre o $AP0$ e $AP1$, derivando a nova chave de tráfego para a estação. Quando a estação quer realizar a transição do $AP1$ para o $AP2$, ela faz uma requisição ao $AP1$, representado pelo passo 7. O passo 8 representa a requisição do $AP1$ para o $AP0$ gerar uma chave de tráfego para o $AP2$. O passo 9 é o envio da chave $R1KH$ gerada pelo $AP0$ para o $AP2$. O passo X representa a finalização do processo de transição do $AP1$ para o $AP2$, e a derivação da chave do $AP2$ para a estação cliente.

ampliam a rede da empresa para um outro local geográfico por meio de tunelamento do CAPWAP sob a *Internet*.

Já o protocolo HOKEY é mais voltado para redes de empresas prestadoras de serviço. Por permitir uma rápida transição entre servidores AAAs necessitando de confiança entre esses servidores, o HOKEY é menos apropriado para redes de empresas. Em redes de empresas prestadoras de serviço esses servidores são confiáveis, o que diminui a possibilidade de ataque.

Por último, o protocolo IEEE 802.11r é melhor aplicado em redes WLAN que não precisam ter gerenciamento centralizado, ou que necessite de uma baixa latência de tempo na execução do processo de transição entre APs, em TABASSAM *et al.* [35] foram realizadas medições que demonstram as vantagens desse protocolo em relação ao protocolo padrão. Contudo, como o protocolo IEEE 802.11r requer pares de chaves semelhantes entre todos os APs da WLAN, isso dificulta a aplicação deste em larga escala.

É importante salientar que o protocolo IAPP (*Inter Access Point Protocolo*) [36] foi descontinuado em Março de 2006, por isso, não é descrito nessa dissertação. Entretanto, o protocolo IEEE 802.11r foi criado para substituir o protocolo IAPP.

3.4 Conclusão do Capítulo

Nesse capítulo foi comentado como que é o processo de transição entre APs e discutido alguns dos novos protocolos que aceleram esse processo. Cada protocolo tem sua característica, e pode-se usá-los em conjunto para melhorar o desempenho da rede. O protocolo IEEE 802.11r foi escolhido para realizar o processo de transição acelerado desta dissertação, por apresentar uma baixa latência, além do cenário de rede desta dissertação não precisar ter gerenciamento centralizado. No próximo capítulo será apresentada a proposta de algoritmo desenvolvida nessa dissertação.

Capítulo 4

Algoritmos para Balanceamento de Carga

Este capítulo apresenta alguns algoritmos de balanceamento de carga e os algoritmos desenvolvidos nesta dissertação. Os algoritmos desenvolvidos são descritos e avaliados separadamente, ao fim do capítulo há uma comparação geral entre estes.

4.1 Tipos de Algoritmos para Balanceamento de Carga

Qualquer abordagem de balanceamento de carga deve se preocupar com duas questões primárias. A primeira consiste em como definir uma solução para o balanceamento, e também, em como obter as medições relacionadas à esta solução. A segunda questão consiste em como balancear ou, pelo menos, distribuir toda a carga de tráfego entre todos os APs disponíveis. Todos os protocolos de balanceamento de carga distribuem a carga do tráfego gerenciando as associações entre APs e estações. Dependendo em qual parte da rede se executa o processo de distribuição de carga, o esquema de distribuição de carga pode ser caracterizado como *Baseado nas Estações* ou *Baseado na Rede* [6].

4.1.1 Balanceamento Baseado nas Estações

A distribuição de carga baseada nas estações, segundo YEN *et al.* [6], conhece o estado de carga dos APs de alguma forma, e portanto seleciona um AP que otimiza seus recursos (como por exemplo, sua banda [37]). APs atuam passivamente em todo o processo de seleção de um novo AP para a estação. Muitas abordagens baseadas nas estações não são desenvolvidas para alcançar o balanceamento de carga – estações selecionam simplesmente APs pelo próprio interesse. Entretanto, como o

algoritmo busca por um AP que ofereça a maior vazão, indiretamente implementa a seleção do AP com menor carga, que é a solução mais utilizada.

O gerenciamento de associações entre APs e estações pode ser realizado de uma maneira estática ou dinâmica. No caso estático, uma estação realiza a seleção de AP antes de se associar com o AP alvo, e não se reassocia com outro AP enquanto durar a associação com o AP atual. A desvantagem da seleção de AP estática é a falta de flexibilidade de se adaptar a redes dinâmicas. Para o caso da seleção dinâmica de AP, uma estação pode realizar uma reassociação com outro AP mesmo se a associação atual ainda estiver ativa. A seleção dinâmica de AP é a abordagem mais aconselhável para um ambiente de redes dinâmicas. Contudo, isso pode também acarretar associações instáveis entre APs e estações, conhecido como efeito ping-pong, que é o fenômeno de um ciclo de associações curtas que uma estação tem de um AP para outro.

Uma das causas do efeito ping-pong é a troca descoordenada de APs, porém sincronizada entre as estações. Por exemplo, se um AP aparecesse subitamente perto de um outro que estivesse congestionado, todas as estações iriam detectar o novo AP e decidiriam se associar a este praticamente ao mesmo tempo. Por conta disso, o novo AP se tornaria congestionado rapidamente devido a essa migração repentina, e em seguida todas as estações voltariam novamente para o AP antigo, pois o atual ficou congestionado, e então o cenário se repetiria. Para se evitar o efeito ping-pong, pode ser utilizada a seleção de AP estática ou criar um mecanismo para distribuir as reassociações no tempo. Por exemplo, em YEN e YEH [38] a estação fica procurando periodicamente um AP menos ocupado, ao achar este AP, a estação sorteia um valor n , em seguida a estação repete a pesquisa n -vezes. Caso o AP tenha sido o melhor resultado dentro dessas n pesquisas, a estação realiza a transição para o AP que foi previamente escolhido como o melhor.

As abordagens que levam em consideração a distribuição de carga baseada nas estações têm como vantagem utilizar APs com poucas ou nenhuma modificações [7, 21, 37, 39–41]. Entretanto, como cada estação escolhe o AP do seu próprio interesse, esse tipo de abordagem pode não balancear a carga da rede como um todo. Uma possível alternativa é distribuir a carga da rede se baseando na rede.

4.1.2 Balanceamento Baseado na Rede

Nas abordagens baseadas na rede, segundo YEN *et al.* [6], o comportamento das estações é passivo em relação as modificações da associação entre AP e estação. Essas modificações são feitas nas entidades da rede (pode ser um AP, um *switch*, ou um servidor dedicado) que controlam a distribuição da carga dos APs. A seguir, três técnicas para APs controlarem a própria carga.

- **Ajuste do alcance:** APs sobrecarregados podem diminuir sua potência de transmissão do *Beacon* para que novas estações não os descubram [42]. APs podem ajustar a área de cobertura de seus rádios afim de APs menos carregados cubram uma área maior que APs carregados, além de não haver buracos de cobertura para assegurar uma cobertura contínua [43, 44].
- **Controle de admissão:** Um AP sobrecarregado pode rejeitar um novo pedido de associação. Já APs não sobrecarregados decidem se uma estação é aceita baseando-se no nível de carga. Uma requisição somente pode ser aceita se a medição preliminar do nível de carga do AP não ultrapassar um certo limiar [2].
- **Gerenciamento de associação:** Um AP sobrecarregado pode enviar uma mensagem não solicitada de desassociação¹ para uma certa estação já associada a este, esperando que esta possa se associar a um AP que não esteja sobrecarregado. A estação escolhida para se desassociar, teoricamente é a que pode balancear a carga entre os APs. Contudo, para saber qual estação escolher, o AP precisa conhecer a carga dos APs vizinhos e os APs que a estação é capaz de se associar. O efeito ping-pong ainda pode ser um problema, caso precise desassociar a mesma estação mais de uma vez [45].

Para uma visão global da distribuição de carga, os APs precisam trocar informações de estado de carga através da rede cabeada, pois nem todos os APs estão no alcance um dos outros. Os APs se identificam sobrecarregados utilizando algumas das técnicas mencionadas nesta seção para obter facilmente as cargas de cada um [7, 44]. Uma outra possibilidade é ter um servidor dedicado na rede cabeada que possa coletar essas informações de carga dos APs [2].

4.2 Algoritmo de Virtualização para Balanceamento de Carga

Nesta dissertação, os algoritmos de balanceamento de carga propostos são baseados nas estações. Os APs não sofrem qualquer tipo de modificação, e é utilizada a técnica de virtualização FH nas estações para realizar o balanceamento de carga. O balanceamento da carga da rede é realizado atribuindo um peso ao tempo ativo de cada interface virtual, de acordo com as soluções propostas nessa dissertação, onde os pesos são realimentados seguindo estas soluções.

¹Possivelmente com código de estado dizendo: "Desassociação porque o AP está incapaz de manter conexão com todos as estações associadas" [6].

A virtualização é realizada através do processo de transição entre APs e da comutação de interfaces virtuais. As estações possuem uma interface virtual para cada AP o qual podem se associar, e esta cria uma abstração da interface física de rede sem fio, então é possível gerenciar as associações e conexões da estação. Nesta dissertação a técnica de virtualização utilizada foi do tipo FH, esta permite que as interfaces virtuais sejam escalonadas uma de cada vez durante um ciclo, e a cada ciclo os pesos são recalculados. Portanto, a soma dos pesos do tempo que cada interface virtual permanece ativa é igual a 1, conforme apresentado pela Equação 4.1.

$$\sum_{y=0}^{n_{virt}} P_x(AP_y) = 1 \quad , \quad (4.1)$$

onde n_{virt} é a quantidade de interfaces virtuais e $P_x(AP_y)$ são os pesos associados a estas interfaces da estação x . Cada interface virtual está relacionada com um AP, logo o número de interfaces virtuais de cada estação é referente à quantidade de APs os quais uma estação pode se associar.

As medidas necessárias para se efetuar o cálculo dos pesos dependem de cada algoritmo desenvolvido. Na Figura 4.1, é apresentado um diagrama genérico de como os ciclos da técnica de virtualização FH funcionam no sistema proposto. A cada ciclo são coletadas medições, e ao fim deste, o cálculo dos pesos dos APs é refeito para o ciclo seguinte. As medições são coletadas no final do tempo ativo de cada interface virtualizada, onde o tempo ativo representa o tempo que cada interface virtual tem para trocar dados com o AP. A solicitação de inicialização das medições do AP é feita no início de cada tempo ativo e recebida ao final deste, quando o ciclo termina essas medidas coletadas são utilizadas para calcular os pesos do ciclo seguinte. As medições coletadas estão sendo representadas na Figura 4.1 pelos rótulos $coletas[j](AP_i)$. O rótulo $fatiassetempo$ da Figura 4.1 representa as T unidades de tempo que um ciclo possui, onde uma unidade, por exemplo, pode ser o intervalo de envio dos *Beacons* de um AP. Para evitar eventuais problemas de sincronização entre os tempos ativos nos ciclos de diferentes estações, cada uma delas adiciona uma variação aleatória de 10% do tamanho do próprio ciclo.

O escalonamento entre as interfaces de rede é feito através do processo de transição entre APs, para isto é utilizado o protocolo IEEE 802.11r [32]. A Figura 4.2 representa como é realizado o escalonamento entre APs utilizando a técnica de virtualização FH. Uma estação cliente, ao comutar para um AP, deve se reassociar a este, enviando um quadro *Reassociate Request*, representado pelo número 2 na Figura 4.2. O AP, ao receber esse quadro, envia um quadro de *Reassociate Notify* para o AP antigo da estação e termina o processo de reassociação, representado pelo número 3 na Figura 4.2. Após a reassociação da estação, o AP atual atualiza as

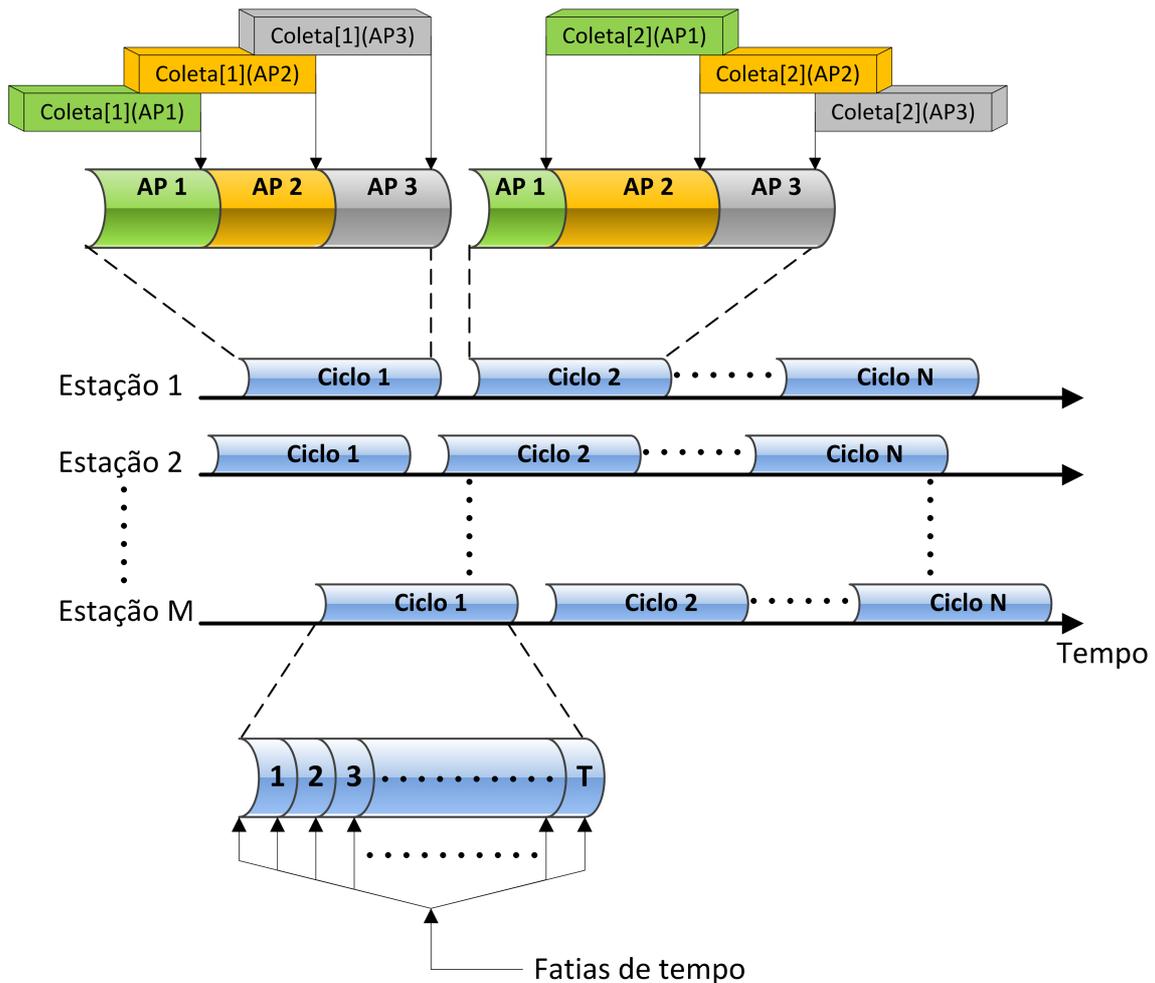
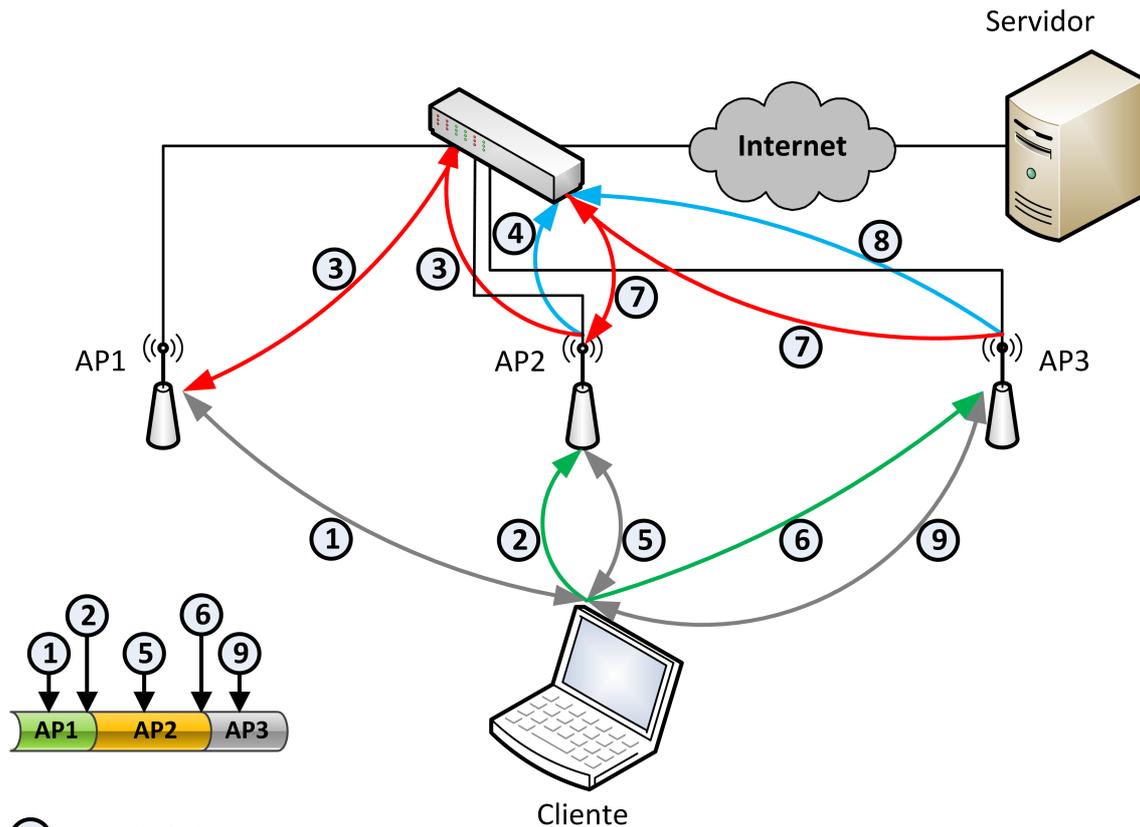


Figura 4.1: Representação do esquema de virtualização FH utilizando fatias de tempo ponderadas para cada AP.

tabelas de encaminhamento de nível 2 dos equipamentos do sistema de distribuição pelo envio de um quadro em difusão utilizando o endereço MAC da estação cliente, representado pelo número 4 na Figura 4.2.

Para se aproveitar o esquema de virtualização, esta dissertação baseia-se na ponderação de fatias de tempos que são disponibilizadas para cada dispositivo virtual. A estação tem um ciclo de escalonamento entre os dispositivos, que é dividido em pequenas fatias de tempo, cada fatia representa uma unidade de tempo para o escalonador. O tempo em que um dispositivo virtual fica ativo é da ordem desta unidade de tempo.

O esquema de virtualização FH é realizado através da transição rápida entre os APs. O protocolo IEEE 802.11r permite que a transição entre os APs seja acelerada o suficiente para que a virtualização seja realizada. Além de permitir a utilização de um protocolo de criptografia durante o processo de virtualização. O processo de transição entre APs introduz um determinado atraso para que os pacotes passem a ser encaminhados ao AP ao qual o cliente acabou de se associar. Além disso,



① Troca de dados.

② (re)Associação e início da transição.

④ Atualização das tabelas de nível 2.

⑥ (re)Associação e início da transição.

⑧ Atualização das tabelas de nível 2.

③ Desassociação do AP antigo.

⑤ Troca de dados.

⑦ Desassociação do AP antigo.

⑨ Troca de dados.

Figura 4.2: Escalonamento entre APs utilizando o protocolo IEEE 802.11r.

existe um atraso associado à comutação de canal entre os APs, que consiste no tempo necessário para a mudança de frequência e na resincronização do novo canal. Caso um protocolo de criptografia seja utilizado, também é possível haver um atraso durante a reassociação. Esses atrasos foram incluídos no modelo de simulação desenvolvido, e o impacto destes no desempenho da rede foi também avaliado em CLANCY [25], MISHRA *et al.* [26], ZHENG e SARIKAYA [33].

Para que os algoritmos de virtualização propostos possam ter suas medições, foi utilizado o protocolo IEEE 802.11k [46]. Dessa forma não é necessário modificar o padrão IEEE 802.11 para obter essas métricas. Algumas das métricas utilizadas são ocupação do meio, quantidade de associados no AP, PER (Taxa de Erro de Pacote – *Package Error Ratio*) e vazão da estação. Para obter as medidas de ocupação do meio e quantidade de associados de um AP, foi feito uso dos recursos do protocolo IEEE 802.11k. A partir da mensagem *STA statistic* é possível solicitar as medidas de quantidade de associados e de ocupação do meio. Para obter esses valores, é usado um quadro para solicitação e outro para resposta. No quadro de solicitação é

possível determinar por quanto tempo o AP coletará as medições. Os algoritmos de virtualização propostos que precisam dessas medidas, fazem desse tempo de coleta o tempo ativo que a estação tem com o AP. Ao final da medição, o AP envia o quadro de resposta com a média dos valores medidos durante o tempo solicitado. Já a obtenção dos valores de PER é realizada a partir de medições pacote a pacote durante as transmissões que as estações têm com os APs, assim como a vazão da estação.

Os algoritmos de balanceamento de carga desenvolvidos nesta dissertação estão separados em duas classes. A primeira considera a potência do sinal de recepção para realizar o balanceamento de carga [47, 48], já a segunda considera a taxa de transferência. Dentro da primeira classe há algoritmos que utilizam as medidas de ocupação do meio. Da mesma forma, dentro da segunda classe há algoritmos que utilizam as medidas de taxa de erro de pacotes. Os algoritmos serão apresentados a seguir juntamente com uma explicação do funcionamento de cada um. Foram gerados alguns modos de virtualização para cada classe de algoritmo.

4.2.1 Algoritmo Estático - Modo 0

Para efeito de comparação, foi desenvolvido o algoritmo mais simples de escalonamento, onde os tempos ativos têm o mesmo valor, ou seja, os pesos possuem os mesmos valores. A Equação 4.2 apresenta como é feito o cálculo dos pesos dos tempos ativos que a estação tem em cada AP.

$$P_x(AP_y) = \frac{1}{N_{APs}(x)} \quad , \quad (4.2)$$

onde $P_x(AP_y)$ é o peso estático que a estação x calcula para o AP_y , e $N_{APs}(x)$ representa a quantidade de APs alcançáveis pela estação x .

4.2.2 Algoritmo com Ocupação do Meio - Modo 1

Este algoritmo de balanceamento de carga baseado em virtualização utiliza as medições da ocupação do meio e da quantidade média de estações associadas ao AP para realizar o balanceamento de carga. A Equação 4.3 apresenta o cálculo realizado para obter o valor da desocupação média que um AP pode ter.

$$\begin{cases} x \in AP_y, & DM_x(AP_y) = \frac{1 - OC_y}{NA_y} \quad , \\ x \notin AP_y, & DM_x(AP_y) = \frac{1 - OC_y}{NA_y + 1} \quad , \end{cases} \quad (4.3)$$

onde x é a estação em questão, $DM_x(AP_y)$ é o valor da desocupação média do AP_y calculado pela estação x , OC_y é a ocupação do meio média medida no AP_y através do protocolo IEEE 802.11k e NA_y é o número médio de estações associadas que o AP_y possui durante o tempo ativo da estação. Como nos algoritmos de virtualização a estação fica escalonando de AP a todo momento, para obter o valor de número de associados médio em um tempo ativo, foi feita uma média do número de associados que o AP tem durante o período ativo que a estação x tem com o AP_y .

Os pesos para os algoritmos de virtualização são normalizados, e a Equação 4.4 apresenta como o algoritmo de balanceamento baseado na ocupação do meio normaliza os pesos que a estação deve ter para definir os tempos ativos.

$$P_x(AP_y) = \frac{DM_x(AP_y)}{\sum_{i=1}^{N_{APs}(x)} DM_x(i)} \quad , \quad (4.4)$$

onde $P_x(AP_y)$ é o peso normalizado que a estação x tem com o AP_y , $DM_x(AP_y)$ é o valor da desocupação média do AP_y calculado pela estação x e $N_{APs}(x)$ é a quantidade de APs alcançáveis pela estação x .

4.2.3 Algoritmo com Ocupação do Meio e Limiar de Aceitação para RSSI - Modo 2

Este algoritmo funciona da mesma maneira que o anterior, utilizando a Equação 4.3 para determinar os pesos que a estação calcula para cada AP. Entretanto, é utilizado também a medida do RSSI para limitar os pesos que cada AP terá, realizando uma classificação de APs entre bons e ruins. Caso haja APs bons e ruins, o algoritmo utilizará um classificador baseado na diferença do RSSI. Então, se um AP apresentar um peso de valor alto mas possui uma relação sinal-ruído muito baixa com a estação, o seu peso é substituído pelo valor mínimo e classificado como ruim. O restante dos APs que apresentarem bons resultados e uma boa relação sinal-ruído, são classificados como bons, e seus pesos são aumentados. Logo, os pesos originais são normalizados mais uma vez, de forma a deixar os APs classificados como ruins com peso igual a 5%, valor sugerido para que esses APs apenas tenham a chance de realizar medições para calcular seu peso no próximo ciclo. Além do mais, as trocas de dados da estação com esses APs não são tão significativas, pois o tempo que ela se mantém associada é muito curto, dando apenas para realizar algumas medições a partir dos dados trocados. A Equação 4.5 mostra o peso residual dos APs que foram classificados como ruins, esse valor residual é usado na normalização da Equação 4.6, que calcula a normalização propriamente dita. Contudo, caso todos os APs sejam classificados como bons ou ruins, este algoritmo funciona da mesma forma que o

algoritmo da Subseção 4.2.2.

$$P_{residual} = \sum_{i=1}^{N_{APs-RUINS}} (P_x(AP_i) - 0.05) \quad , \quad (4.5)$$

onde $P_{residual}$ é o peso que sobra dos APs que são classificados como ruins, $P_x(AP_i)$ é o peso normalizado que a estação x tem com o AP_i , e $N_{APs-RUINS}$ é a quantidade de APs classificados como ruins pela estação x .

$$\left\{ \begin{array}{l} AP_y \in C_{RUINS}, \quad P_x(AP_y) = 0.05 \quad , \\ AP_y \in C_{BONS}, \quad P_x(AP_y) = P_x(AP_y) + P_{residual} \cdot \left(\frac{P_x(AP_y)}{\sum_{i=1}^{N_{APs-BONS}} P_x(AP_i)} \right) \quad , \end{array} \right. \quad (4.6)$$

onde C_{RUINS} e C_{BONS} são os conjuntos em que cada AP é classificado, e $N_{APs-BONS}$ representa a quantidade de APs classificados como bons.

4.2.4 Algoritmo com Taxa de Erro de Pacotes - Modo 3

As medidas de PER coletadas de cada AP e a quantidade de estações associadas a este, são utilizadas para calcular os pesos de cada tempo ativo de uma estação. A Equação 4.7 apresenta como o peso foi calculado utilizando as medições de PER médio e a quantidade de estações associadas que o AP em questão tem.

$$P_x(AP_y) = \frac{1 - PER_x(AP_y)}{NA_y} \quad , \quad (4.7)$$

onde $P_x(AP_y)$ é o peso que o AP_y tem com a estação x , $PER_x(AP_y)$ é o valor médio das medições feitas pela estação x , e NA_y é a média do número de associados que o AP_y tem em um tempo ativo.

4.2.5 Algoritmo com Vazão Média - Modo 4

Este algoritmo, além de utilizar as medidas de PER como o algoritmo da Subseção 4.2.4, faz uso das medidas das taxas de transferências. Com essas duas medidas, é possível estimar a vazão que a estação possui com o AP. O cálculo dos pesos proposto para este algoritmo é apresentado na Equação 4.8.

$$P_x(AP_y) = TX_x(AP_y) \cdot \left(\frac{1 - PER_x(AP_y)}{NA_y} \right) \quad , \quad (4.8)$$

onde $P_x(AP_y)$ é o peso que a estação x calcula para o AP y , $PER_x(AP_y)$ é a média dos valores medidos pela estação x para o AP_y , $TX_x(AP_y)$ é a medição média dos valores coletados das taxa de transferências que a estação x teve com o AP_y , e NA_y é a quantidade de estações associadas ao AP_y .

4.3 Algoritmos da Literatura Implementados

Além dos algoritmos propostos, foram implementados outros três algoritmos. Dois desses são baseados nas medições de RSSI, sendo que o primeiro algoritmo implementado é a abordagem padrão para realizar as associações com os APs. Sendo que essas associações realizadas levando apenas em consideração o nível do sinal recebido, essa abordagem é melhor detalhada na Subseção 4.3.1. A segunda abordagem desenvolveu uma fórmula que relaciona o nível de sinal recebido e a quantidade de estações associadas ao AP. Essa segunda abordagem é baseada no algoritmo de SHEU e WU [7], conhecido como DLBA (*Dynamic Load Balance Algorithm*), e apresentada na Subseção 4.3.2. O terceiro algoritmo implementado utiliza a taxa de erro de pacotes transmitidos para definir a associação de cada estação. Esse algoritmo foi desenvolvido por FUKUDA *et al.* [49] e apresentado na Subseção 4.3.3.

4.3.1 Algoritmo de Seleção Padrão

O algoritmo padrão de associação é descrito em IEEE802.11 [1]. A seleção de APs é feita pela medida apenas da potência do sinal recebido, o que de certa maneira não é a melhor forma de compartilhar os recursos da rede. Para uma estação receber as medidas de potência de sinal de cada AP que está ao seu alcance, é necessário fazer uma sondagem em todos os canais. Para isso a estação envia um pacote chamado de *probe request* e aguarda as respostas dos APs que receberam este pacote. O pacote de resposta dos APs apresenta o valor da relação sinal ruído, e o maior valor é o escolhido, independe se o AP está congestionado ou não. Essa operação é bastante demorada, o que também atrapalha o processo de transição entre APs. Em MISHRA *et al.* [26] foram realizadas medições do tempo que levam os processos de escaneamento de canais, autenticação, associação e atualização das tabelas de roteamento.

4.3.2 Algoritmo DLBA

O algoritmo DLBA, desenvolvido por SHEU e WU [7], tenta melhorar a distribuição de carga dos APs entre as estações que estão no alcance destes. Para isto, é necessário adicionar duas informações extras quando um AP responde à um *probe request* de uma estação. A primeira delas é a potência do sinal recebido pelo *probe request*

da estação. A segunda é a média da potência do sinal das estações associadas ao AP incluindo a estação que enviou o pedido de sondagem. A Equação 4.9 mostra como é o cálculo da média da potência do sinal de todas as estações que o AP tem associadas.

$$\begin{cases} x \notin S_y, & ARSSI_y = \frac{\sum_{z \in S_y} RSSI_y(z) + RSSI_y(x)}{NA_y + 1} \\ x \in S_y, & ARSSI_y = \frac{\sum_{z \in S_y} RSSI_y(z)}{NA_y} \end{cases}, \quad (4.9)$$

onde $ARSSI_y$ é a média do RSSI das estações associadas ao AP_y , S_y é o conjunto de estações associadas ao AP_y , $RSSI_y(x)$ é a medição do RSSI entre a estação x , $RSSI_y(z)$ é a medição do RSSI entre a estação z (que pertence ao conjunto S_y) e o AP_y , e NA_y é a quantidade de estações associadas ao AP_y . A diferença entre o RSSI da estação x e o valor calculado na Equação 4.9, é dada pela Equação 4.10.

$$DR_x(y) = RSSI_y - ARSSI_x, \quad (4.10)$$

onde $DR_x(y)$ é a diferença entre o RSSI da estação x e a média do RSSI que o AP y tem com as estações associadas. Para fazer a escolha do melhor AP, o algoritmo DLBA propõe um peso proporcional, como mostra a Equação 4.11.

$$\begin{cases} DR_x(y) > 0, & P_x(AP_y) = DR_x(y) \times \left(1 + \frac{ARSSI_y}{RSSI_x}\right) \\ DR_x(y) \leq 0, & P_x(AP_y) = DR_x(y) \times \left(1 - \frac{ARSSI_y}{RSSI_x}\right) \end{cases}, \quad (4.11)$$

onde $P_x(AP_y)$ é o peso do algoritmo DLBA calculado pela estação x para o AP_y . Portanto, a estação escolhe o AP que apresentar o maior peso, frisando que este peso não tem relação com os pesos dos algoritmos propostos nessa dissertação. A cada vez que a $DR_x(y)$ apresenta um valor negativo, é preciso fazer um recálculo dos pesos de todos os APs. Para evitar o efeito ping-pong, a cada vez que o recálculo é feito um contador é incrementado, quando este chega a um limite o recálculo dos pesos é realizado.

4.3.3 Algoritmo MLT

O algoritmo de seleção de AP chamado MLT (*Maximizing Local Throughput*), descrito em FUKUDA *et al.* [49], primeiro estima a vazão alcançável que cada AP pode oferecer para a estação, em seguida, a estação escolhe qual o AP que maximiza sua

vazão para se associar. A Equação 4.12 mostra como é calculado o peso que cada AP tem com a estação.

$$\begin{cases} x \in y, & P_x(AP_y) = \frac{1 - PER_x(AP_y)}{NA_y} \\ x \notin y, & P_x(AP_y) = \frac{1 - PER_x(AP_y)}{NA_y + 1} \end{cases}, \quad (4.12)$$

onde $P_x(AP_y)$ é o peso determinado pela estação x para o AP_y , $PER_x(AP_y)$ é média dos valores de PER obtido pelas transmissões do AP_y para a estação x , e NA_y é a quantidade de estações associadas ao AP_y . O PER é calculado a partir das medições de erros de pacotes. Já para obter o valor da quantidade de estações associadas ao AP, este algoritmo modifica o padrão para que a mensagem de resposta ao *probe request* e ao *beacon*, possa ser enviada com este valor. Finalmente, a estação escolhe o AP que apresenta o maior peso.

Para complementar o algoritmo MLT, foi adicionado uma condição chamada de AALP (*Avoiding APs with Larger PER*), também descrita em FUKUDA *et al.* [49]. A Equação 4.13 apresenta a condição que serve para evitar APs que possuem um PER muito ruim. Portanto, se um AP tiver alguma estação com um valor muito alto de PER, seu peso é diminuído para ser evitado pela estação.

$$\begin{cases} PER_{max}(y) > 0.5, & P_x^{AALP}(AP_y) = \left(\frac{1}{2} \cdot \sqrt{2 \cdot (1 - PER_{max}(y))} \right) \cdot P_x(AP_y) \\ PER_{max}(y) < 0.5, & P_x^{AALP}(AP_y) = P_x(AP_y) \end{cases}, \quad (4.13)$$

onde $P_x^{AALP}(AP_y)$ é o peso final definido pela condição AALP para o AP y . Então, a estação escolhe o AP que apresenta o maior valor de P_x^{AALP} para se associar

4.4 Conclusão do Capítulo

Nesse capítulo são apresentados os algoritmos propostos por esse trabalho, além daqueles com os quais eles estão sendo comparados. Cada algoritmo apresentado tem uma solução distinta e formas diferentes de adquirir as medições necessárias para sua solução. Os algoritmos propostos não necessitam de modificações no padrão IEEE 802.11, e em nenhuma entidade de rede. Entretanto, os outros algoritmos apresentam modificações no padrão e/ou em alguma entidade de rede. O caso do algoritmo MLT é possível utilizar o protocolo IEEE 802.11k para adquirir informações do número de associados de um AP, assim como o algoritmo DLBA. Entretanto, no algoritmo DLBA, o fato de precisar das medidas de RSSI da estação em relação ao AP, isto o deixa como o único algoritmo, apresentado nesse trabalho, que realmente precisa fugir do padrão IEEE 802.11. A Tabela 4.1 apresenta todos os algoritmos

desenvolvidos e suas respectivas métricas para balancear a carga da rede.

Tabela 4.1: Resumo dos algoritmos de balanceamento de carga utilizados e suas respectivas métricas.

Algoritmos	Medição utilizada
Virtualização Modo 0	Nenhuma medição (pesos iguais)
Virtualização Modo 1	Ocupação do meio
Virtualização Modo 2	Ocupação do meio e limiar da potência de sinal
Virtualização Modo 3	Taxa de erro de pacotes
Virtualização Modo 4	Vazão média
Padrão	Potência de sinal
DLBA	Potência de sinal
MLT	Taxa de erro de pacotes

O capítulo seguinte apresenta os resultados obtidos pelas simulações realizadas dos algoritmos apresentados nesse capítulo. As configurações das simulações realizadas também são descritas no próximo capítulo.

Capítulo 5

Resultados e Discussões

Neste capítulo são apresentados os resultados obtidos pelos algoritmos de balanceamento de carga desenvolvidos nesta dissertação e alguns algoritmos da literatura. Foi feita uma comparação entre as métricas dos algoritmos desenvolvidos, e o que apresentou melhores resultados foi comparado com os da literatura.

5.1 Simulador e Cenários de Simulação

As simulações dessa dissertação foram realizadas no NS-2.34 (*Network Simulator* versão 2). Esta versão do simulador vem com o biblioteca `dei80211mr` de adaptação de taxa, que também foi utilizado nas simulações. Para realizar as simulações foi utilizado o código de propagação Ricean e o modelo de adaptação de taxa ARF (*Auto Rate Fallback*).

Os cenários de simulação foram gerados com posições randômicas em cada rodada. O cenário é um quadrado cujos lados representam 60% ou 80% do alcance da uma estação. Para tirar a uniformidade das posições das estações, o cenário foi dividido em 4 quadrantes. Para cada quadrante foi sorteado uma quantidade de estações e suas posições, no quadrante seguinte é sorteado o que restou do total de estações e assim por diante. O cenário foi montado de forma que todas as estações fiquem ao alcance umas das outras. Os APs são alcançáveis por todas as estações também, e trabalham em canais ortogonais. Além disso, os APs pertencem ao mesmo ESS (*Extended Service Set*). O tráfego utilizado pelas estações especiais é o FTP (*File Transfer Protocol*) e o HTTP (*HyperText Transfer Protocol*) para as comuns. Estações especiais são as que executam os algoritmos de balanceamento de carga, enquanto as comuns são estações que não usam esse tipo de algoritmo. Foi considerado que ambos os tipos de estação têm a direção do tráfego de *download*.

As simulações têm duração de 200 segundos e foram realizadas 30 rodadas para cada ponto das curvas dos resultados. O intervalo de confiança dos resultados é de 95%. Há 3 pontos de acesso nos cenários, uniformemente distribuídos no cenário.

A vazão apresentada nos resultados é a vazão agregada dos 3 APs. Foi variada a quantidade de estações especiais no cenário e a presença ou não de estações comuns. O número máximo de cada tipo de estação é 30, e a cada cenário há um incremento de 5 novas estações especiais. As simulações possuem pacotes de tamanho fixo em 1500, e o tráfego FTP só finaliza com o término da simulação. As estações comuns tem um tráfego HTTP baseado em LEE e GUPTA [50]. Ambos os tipos de estações iniciam seus tráfegos em tempos aleatórios no período de *warmup*, cuja duração é de 50 segundos.

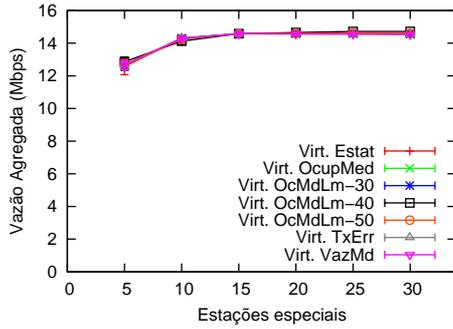
O tráfego das estações possui a direção de *download*, ou seja, os dados trafegam do servidor para as estações. A estação servidor é cabeada e fica ligada aos APs por um *link* de 1Gbps. Portanto, as estações que têm que realizar alguma medição fazem seguindo o sentido do tráfego.

A avaliação de desempenho entre os algoritmos foi feita de três maneiras. A primeira avaliação é feita entre as vazões médias obtidas de cada algoritmo durante o período de simulação. A segunda expressa o quanto os algoritmos são justos ao dividir a carga da rede entre as estações, nesta avaliação foi utilizado o método de JAIN *et al.* [51] para medir a justiça. Por último, a terceira avaliação apresenta o quanto as estações que utilizam os algoritmos de balanceamento de carga influenciam no tráfego das estações que não os possuem, essa avaliação é dada pelo cálculo do atraso médio que as páginas *web* do tráfego HTTP possuem por conta do tráfego FTP da estações especiais.

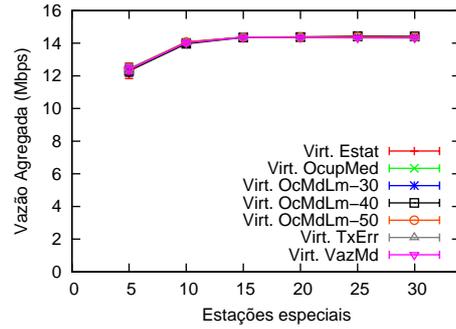
5.2 Algoritmos de Balanceamento de Carga com Virtualização

O desempenho dos algoritmos de virtualização, nesse primeiro momento, estão sendo comparados apenas entre si. Um dos algoritmos de virtualização desenvolvidos, o que se baseia na classificação dos APs, segundo a Seção 4.2.3, foi avaliado com três valores diferentes para o limiar de classificação. O restante dos algoritmos não têm nenhum tipo de variação a ser analisada. Nas Figuras 5.1, 5.2, 5.3, 5.4, 5.5 e 5.5, o algoritmo de virtualização estático é representado pela curva *Virt. Estat*(Modo 0), o que utiliza a ocupação média do meio é representado pela curva *Virt. OcupMed* (Modo 1), já o que utiliza a ocupação e o limiar do RSSI é representado por *Virt. OcMdLm* (Modo 2) mais o valor do limiar usado, o algoritmo de virtualização que utiliza a taxa de erro de pacotes é representado pela curva *Virt. TxErr* (Modo 3), e o que considera a vazão média é representado pela curva *Virt. VazMd* (Modo 4).

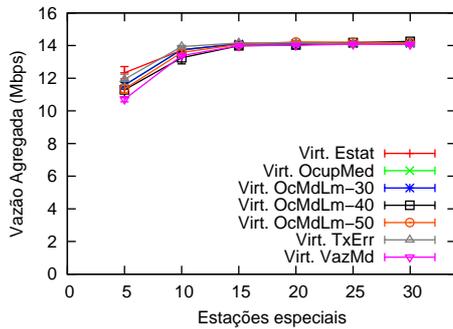
Primeiramente, foi analisado a vazão agregada gerada pelas estações especiais durante a simulação. As Figuras 5.1 e 5.2 demonstram que os resultados dos al-



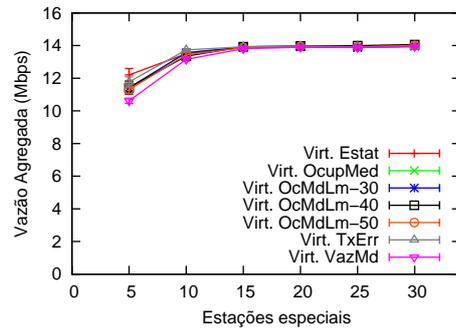
(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.



(c) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.

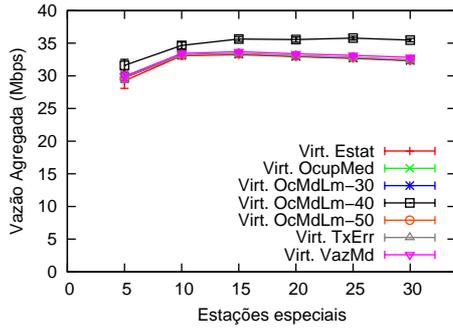


(d) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

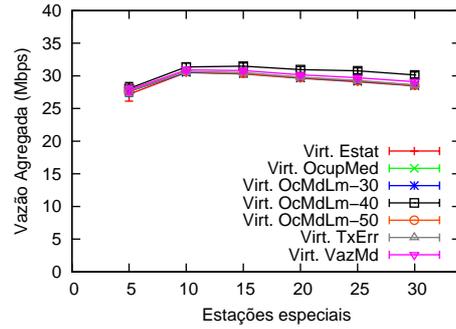
Figura 5.1: Gráficos da vazão do algoritmo com virtualização utilizando o protocolo IEEE 802.11b.

algoritmos de virtualização alcançam a mesma vazão agregada. Independente do cenário avaliado, seja com o cenário quadrado com lado representando 60% ou 80% do valor do alcance de uma estação, as estações especiais mantiveram valores muito próximos. O mesmo se percebe com a mudança do protocolo, de IEEE 802.11b (Figura 5.1) para IEEE 802.11g (Figura 5.2). Portanto, pode-se dizer que a vazão agregada de todos os algoritmos de virtualização é a mesma. O algoritmo com virtualização baseado na vazão média apresentou um resultado abaixo da média dos outros algoritmos quando havia cinco estações especiais e trinta estações comuns. Isso demonstrou fragilidade no algoritmo quando há um valor muito maior de estações comuns que estações especiais. Segundo as Figuras 5.1c, 5.1d, 5.2c e 5.2d, os valores do algoritmo com virtualização baseado na vazão média foi o que apresentou a menor vazão agregada quando há cinco estações especiais. Entretanto, isto não ocorre nas Figuras 5.1a, 5.1b, 5.2a e 5.2b, quando não há estações comuns, o que reforça a explicação de que o esse algoritmo é atrapalhando quando há um grande número de estações comuns em relação a estações especiais.

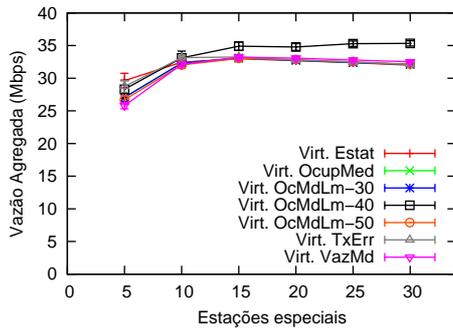
Apesar de todos os algoritmos de balanceamento de carga com virtualização



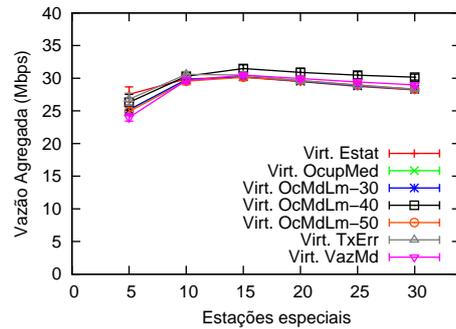
(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.



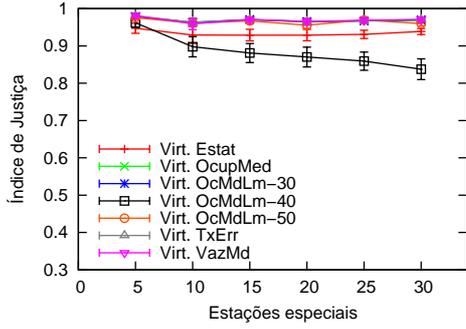
(c) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



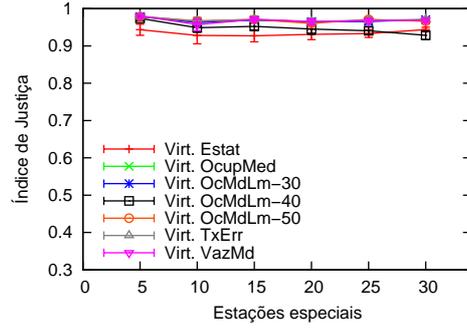
(d) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.2: Gráficos da vazão do algoritmo com virtualização utilizando o protocolo IEEE 802.11g.

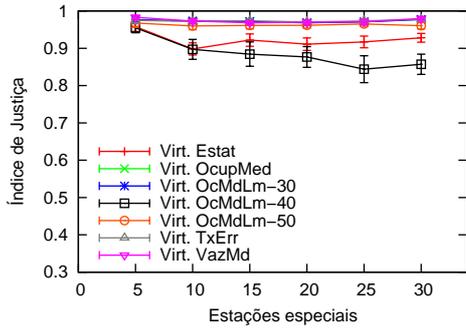
apresentarem valores próximos com relação a vazão, o índice de justiça, baseado no em JAIN *et al.* [51], apresentou diferença apenas para o algoritmo que mantém o tempo estático entre os APs. O índice de justiça demonstra o quanto um recurso foi bem distribuído, segundo a Equação 5.1. Então, se há uma pequena variação desse valor, isso implica que há uma injustiça considerável. Analisando os gráficos das Figuras 5.3 e 5.4, o algoritmo com virtualização estático apresentou um valor de índice de justiça abaixo do restante nos cenários de simulação. Essa distância ocorre por conta da permanência em um ou mais APs que não disponibiliza uma boa conexão para a estação especial, dessa forma a estação perde um bom tempo em APs que a fazem diminuir sua vazão média. O resultado é que algumas estações apresentam vazão média maior que os outros, o que caracteriza uma injustiça na



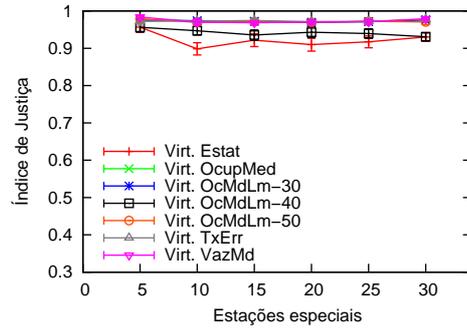
(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.



(c) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



(d) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

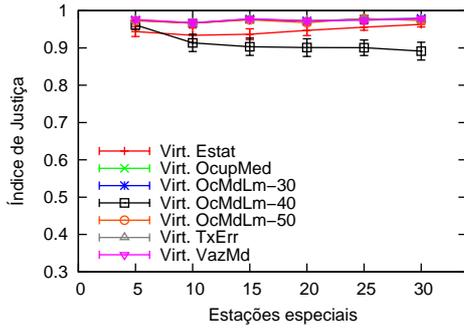
Figura 5.3: Gráficos do índice de justiça do algoritmo com virtualização utilizando o protocolo IEEE 802.11b.

rede, ou seja, uma má distribuição dos recursos.

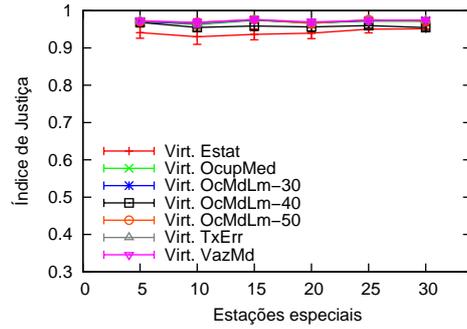
$$fi = \frac{\left(\sum_{i=0}^N v_i\right)^2}{N \times \left(\sum_{i=0}^N (v_i)^2\right)} \quad (5.1)$$

Onde fi é o índice de justiça, v_i é a vazão da estação i , e N é o total de estações.

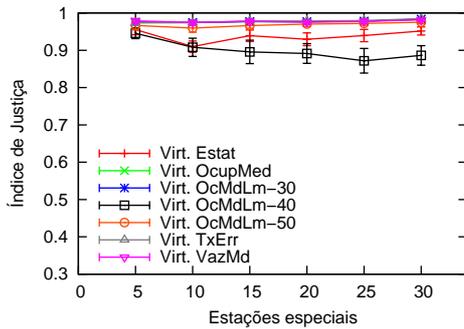
A última avaliação realizada foi em relação à latência das páginas *web* requisitadas causada pelas estações que possuem algoritmos de balanceamento de carga com virtualização. As Figuras 5.5 e 5.6 apresentam os resultados obtidos por esta avaliação. As estações especiais que utilizam o algoritmo com virtualização baseado na vazão média apresentou o melhor resultado dentre todos os algoritmos desenvolvidos. Isso significa que as estações especiais que utilizam esse tipo de algoritmo não atrapalha tanto o tráfego das estações comuns, ou seja, as estações especiais com algoritmo de balanceamento de carga com virtualização baseado na vazão média causam um menor impacto no atraso das páginas *web* requisitadas pelas estações



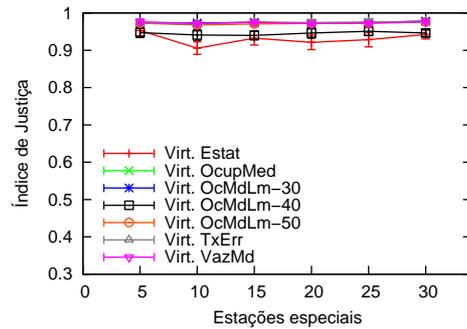
(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.

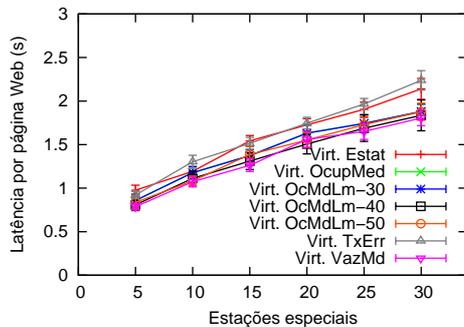


(c) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.

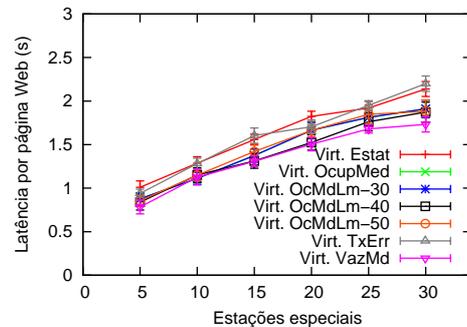


(d) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.4: Gráficos do índice de justiça do algoritmo com virtualização utilizando o protocolo IEEE 802.11g.



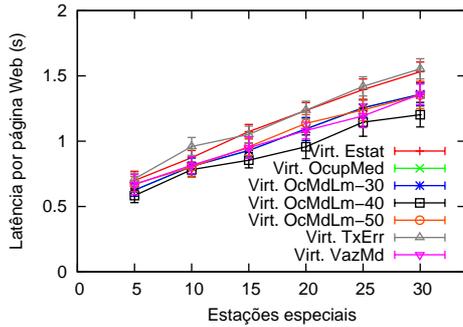
(a) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



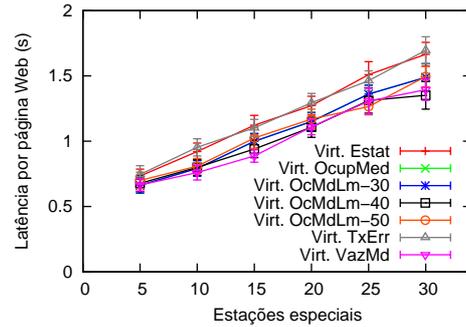
(b) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.5: Gráficos da latência das páginas *web* das estações comuns do algoritmo com virtualização utilizando o protocolo IEEE 802.11b.

comuns.



(a) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



(b) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.6: Gráficos da latência das páginas *web* das estações comuns do algoritmo com virtualização utilizando o protocolo IEEE 802.11g.

5.2.1 Conclusão Parcial

De modo geral, os resultados da vazão agregada são praticamente os mesmos para todos os algoritmos com virtualização. Já em relação ao resultado do índice de justiça, apenas o algoritmo que possui tempos ativos iguais e estáticos apresentou um valor inferior aos demais, contudo, todos os outros mostraram resultados muito próximos. O último resultado avaliado, latência das páginas *web* requisitadas, mostrou que o algoritmo com virtualização baseado na vazão média se demonstrou mais amigável com a convivência em comum com outras estações comuns. Portanto, para comparar melhor os resultados, a Tabela 5.1 apresenta os resultados dos algoritmos de virtualização para 30 estações especiais utilizando o protocolo IEEE 802.11b em um cenário de 60% do raio de alcance das estações juntamente com 30 estações comuns. Segundo essa tabela, os resultados do algoritmo baseado na vazão média (Modo 4) se demonstraram um pouco melhores que os demais, apesar de apresentar um valor de menor em relação à vazão, este algoritmo causa uma latência menor nas vazões das estações normais e o índice de justiça um pouco melhor que os demais.

5.3 Algoritmos da Literatura

Os algoritmos de balanceamento de carga que foram escolhidos para comparar com o algoritmo proposto nesse trabalho, possuem configurações que podem ser variadas. A seguir são apresentados os resultados desses algoritmos e suas variações.

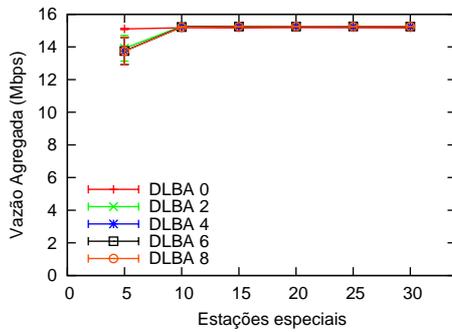
5.3.1 Resultados do Algoritmo DLBA

Os resultados obtidos pelo algoritmo DLBA variam de acordo com a quantidade de tentativas necessárias para realizar uma transição. Esse limiar de tentativas é

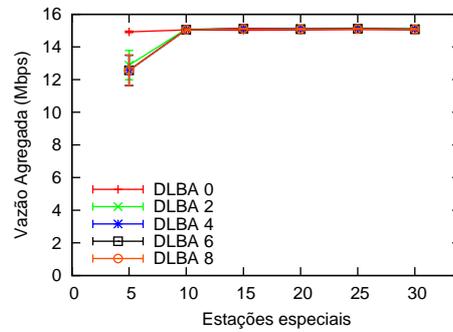
Tabela 5.1: Amostra de resultados numéricos atingidos pelos algoritmos de virtualização.

Modos de virtualização	Vazão Agregada	Índice de Justiça	Latência
Virtualização Modo 0	14,1672	0,9279	2,1393
Virtualização Modo 1	14,1404	0,9770	1,8804
Virtualização Modo 2 - lm. 30	14,1404	0,9770	1,8804
Virtualização Modo 2 - lm. 40	14,2592	0,8572	1,8359
Virtualização Modo 2 - lm. 50	14,1751	0,9604	1,8772
Virtualização Modo 3	14,1019	0,9777	2,2382
Virtualização Modo 4	14,0720	0,9797	1,8049

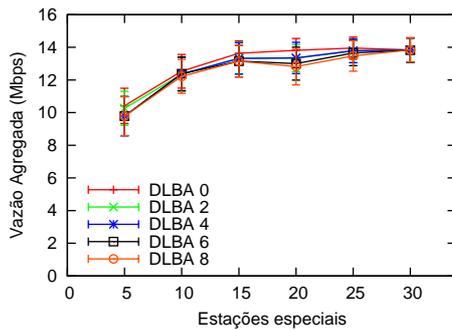
variado e os gráficos foram gerados com o intuito de determinar qual o limiar que apresenta o melhor desempenho.



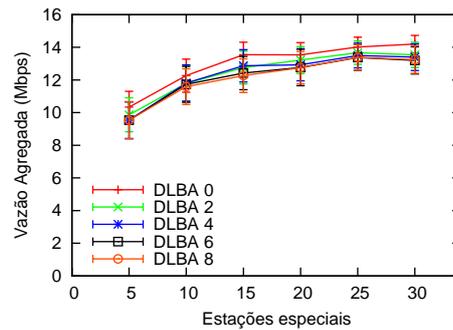
(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.



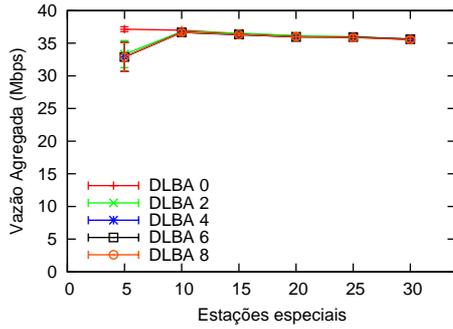
(c) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



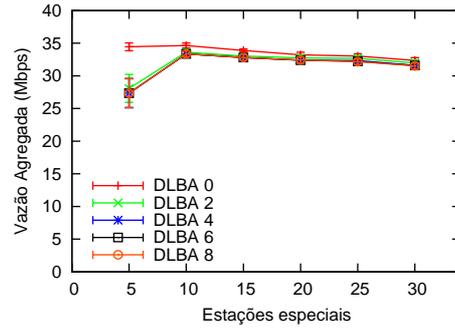
(d) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.7: Gráficos da vazão do algoritmo DLBA utilizando o protocolo IEEE 802.11b.

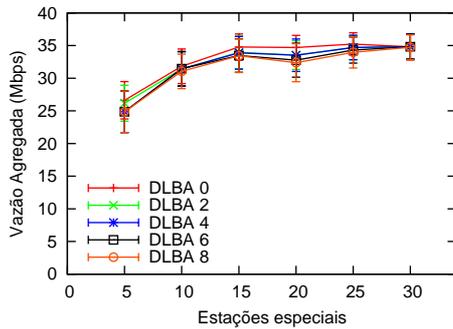
O limiar de tentativas para realizar uma transição no algoritmo DLBA foi variado com os seguintes valores: 0, 2, 4, 6, 8. Os gráficos apresentados nas Figuras 5.7 e 5.8 representam a vazão agregada do algoritmo DLBA. A partir dos gráficos que não



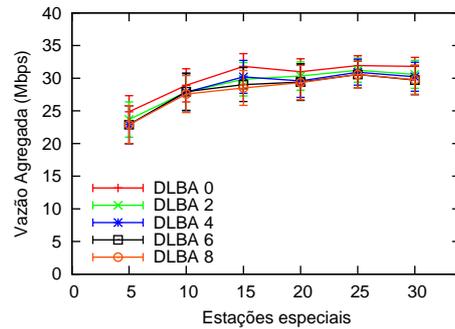
(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.



(c) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.

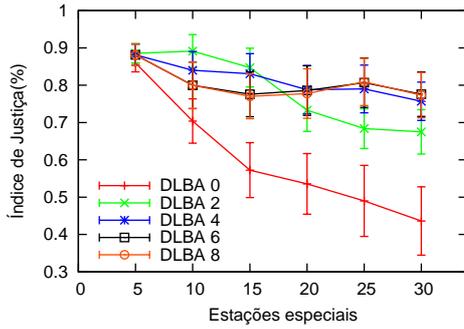


(d) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

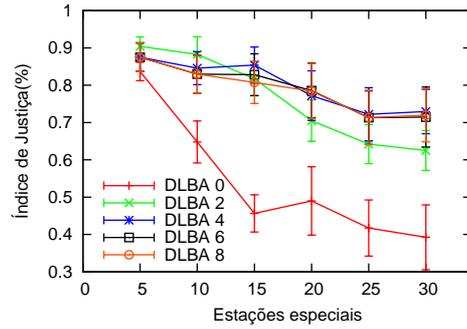
Figura 5.8: Gráficos da vazão do algoritmo DLBA utilizando o protocolo IEEE 802.11g.

possui estações comuns como tráfego de fundo, Figuras 5.7a, 5.7b, 5.8a e 5.8b, percebe-se que não há diferença significativa entre as vazões agregadas, além de não apresentar grandes variações com o aumento das estações. Contudo, nas Figuras 5.7c, 5.7d, 5.8c e 5.8d, os resultados demonstram que o algoritmo sofreu com o acréscimo das estações comuns, apresentando uma média abaixo dos gráficos que não têm estações comuns. É possível notar também que a variação do intervalo de confiança dos resultados cresceu com a adição das estações comuns, mas mesmo assim não apresenta diferenças significativas entre as curvas. A variação do tamanho dos lados do cenário de 60% para 80% do alcance de uma estação, também não afetou o algoritmo DLBA significativamente.

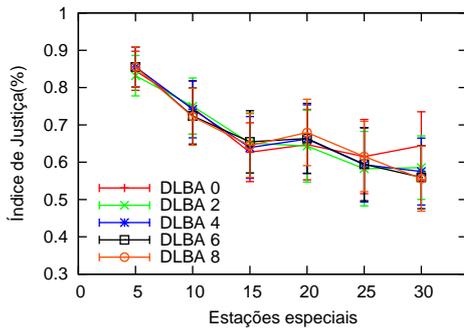
O algoritmo DLBA não distribui muito bem os recursos da rede entre as estações, mesmo que a rede tenha apenas estações especiais. Os gráficos das Figuras 5.9 e 5.10 demonstram a má distribuição dos recursos da rede por parte das variações do algoritmo DLBA. Nos gráficos das Figuras 5.9a, 5.9b, 5.10a e 5.10b, os índices de justiça apresentam curvas bem distintas, além de uma grande variação do intervalo de confiança dos resultados. Essa grande variação do intervalo de confiança dos



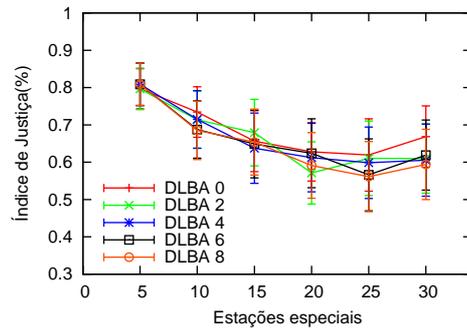
(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.



(c) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.

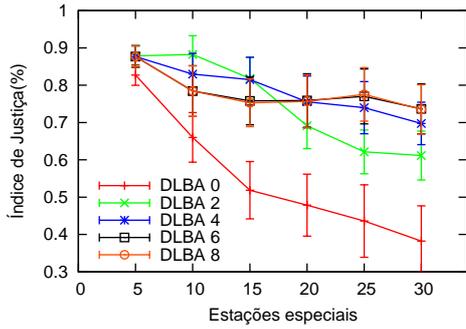


(d) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

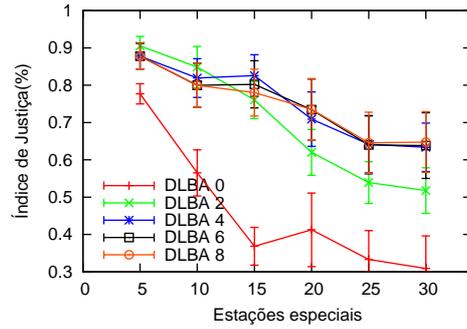
Figura 5.9: Gráficos do índice de justiça do algoritmo DLBA utilizando o protocolo IEEE 802.11b.

resultados também é apresentado nos gráficos das Figuras 5.9c, 5.9d, 5.10c e 5.10d, mas as curvas se apresentam bem mais próximas com a variação da quantidade de estações especiais, justamente pelo fato de ter muitas estações comuns na rede. O excesso de estações comuns na rede desequilibra o algoritmo DLBA, e deixa as curvas bem próximas com o aumento de estações especiais, isto é apresentado pelas curvas da Figuras 5.9c, 5.9d, 5.10c e 5.10d. Quando se varia os lados do cenário de 60% para 80%, os resultados do índice de justiça do algoritmo DLBA sofre uma leve redução.

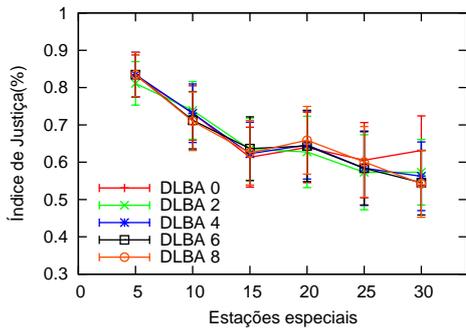
As estações comuns sofrem bastante quando compartilham o meio com as estações especiais que executam o algoritmo DLBA. Segundo os gráficos das Figuras 5.11 e 5.12, o atraso causado pelas estações especiais, que utilizam esse algoritmo, é alto e aumenta ainda mais com o acréscimo de estações especiais, apesar que essa característica se apresentam em todos os outros algoritmos. O que ficou perceptível pelos resultados, é o fato de os resultados não variarem substancialmente com o aumento dos lados do cenário de 60% para 80%. Já a mudança de protocolo de IEEE 802.11b para IEEE 802.11g, apresenta um resultado esperado, pois como



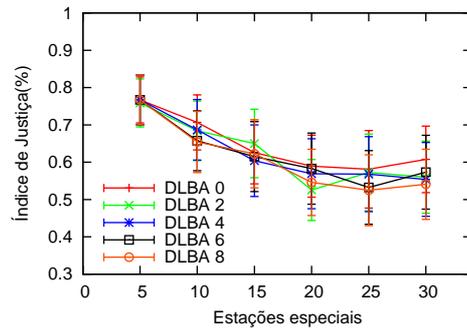
(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.

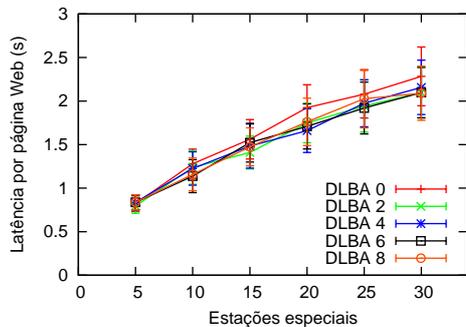


(c) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.

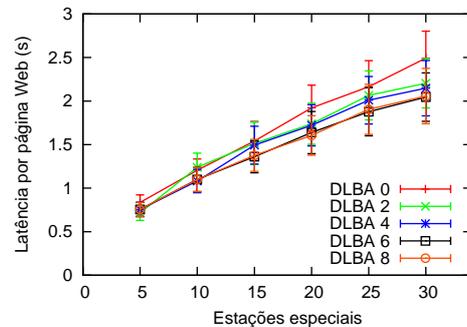


(d) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.10: Gráficos do índice de justiça do algoritmo DLBA utilizando o protocolo IEEE 802.11g.



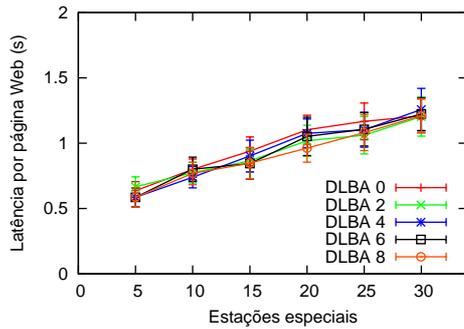
(a) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



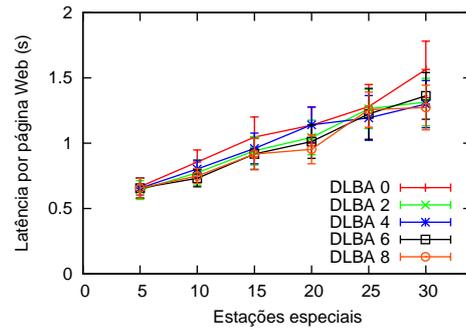
(b) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.11: Gráficos da latência das páginas *web* das estações comuns do algoritmo DLBA utilizando o protocolo IEEE 802.11b.

a taxa de transferência do protocolo IEEE 802.11g é maior, o atraso nas estações comuns tende a diminuir por passarem menos tempo trocando dados.



(a) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



(b) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

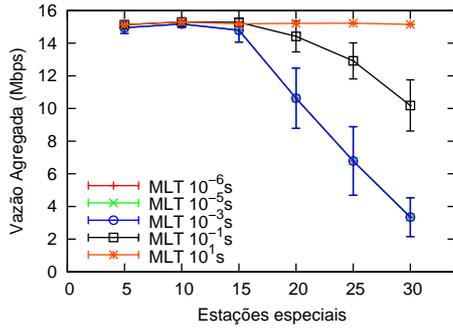
Figura 5.12: Gráficos da latência das páginas *web* das estações comuns do algoritmo DLBA utilizando o protocolo IEEE 802.11g.

Conclusão Parcial

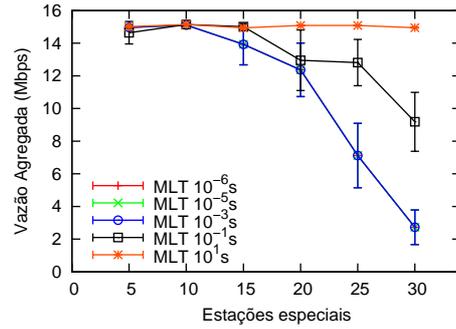
O algoritmo DLBA apresenta resultados parecidos quando há a presença das estações comuns, nem os índices de justiça, nem os resultados da vazão e nem a latência das páginas *web* distinguem as curvas. Entretanto, o que diferenciam as curvas quando não há as estações comuns, são os resultados dos índices de justiça. As curvas que têm os valores de tentativas igual a 6 e 8 apresentam os melhores resultados, pois possuem um índice de justiça melhor e mais estável que as demais curvas. Agora comparando entre essas duas curvas, a que possui o índice de justiça um pouco maior é a curva que representa 6 tentativas para realizar a transição.

5.3.2 Resultados do Algoritmo MLT

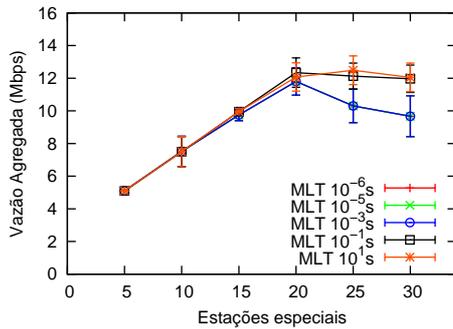
A avaliação de desempenho do algoritmo MLT foi realizada variando o tempo de espera que o algoritmo tem durante o estado de espera. Os resultados apresentados nas curvas do algoritmo MLT demonstram como é o seu comportamento com a variação da quantidade de estações comuns (as que possuem tráfego HTTP). Nesse algoritmo, o tempo de espera padrão descrito em [49] é o tempo de um *backoff*. Portanto, como o tempo de espera é muito pequeno, há muitas reassociações, no próprio artigo que descrevesse algoritmo, o autor mediu a quantidade de vezes que a estação realiza uma mudança de AP. A variação desse tempo de espera é definido pelo valor padrão do tempo de espera multiplicado por um valor de base 10 e um expoente variável entre 0 e 7. Como as simulações possuem 200 segundos, a variação do expoente não é maior que 7. Então, os valores utilizados foram 0, 1, 3, 5 e 7, onde 0 representa o valor definido no trabalho original, e 7 a variação máxima. Os resultados gerados para os valores entre 0 e 3 são iguais, não há nenhuma variação, mas os valores seguintes apresentam variação e resultados bem melhores.



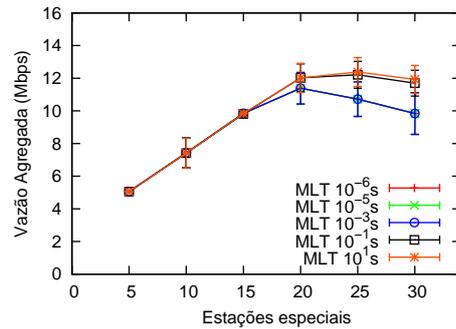
(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.



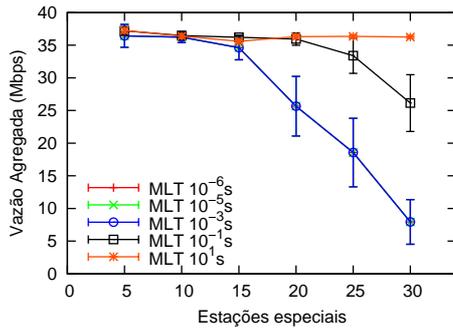
(c) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



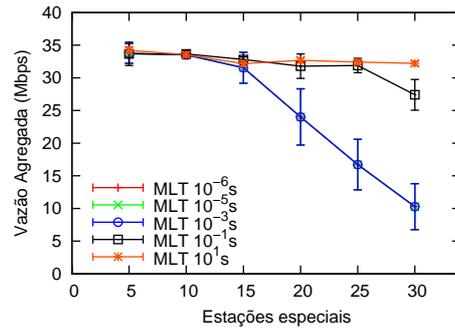
(d) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.13: Gráficos da vazão do algoritmo MLT utilizando o protocolo IEEE 802.11b.

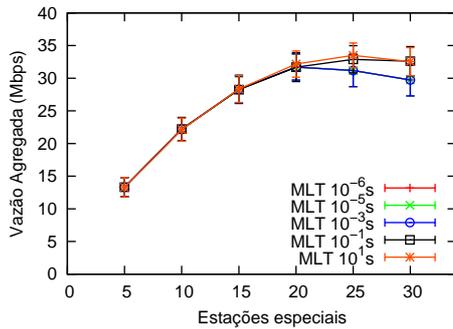
Os resultados apresentados nas Figuras 5.13 e 5.14 demonstra o comportamento da vazão do algoritmo MLT. Primeiramente, nos resultados das curvas dos cenários que não possuem estações comuns, as curvas apresentaram uma queda da vazão quando há um aumento na quantidade de estações especiais, e isto ocorre a partir de 15 estações. O aumento da quantidade de estações especiais, nesse tipo de cenário, faz com que esse algoritmo perca muito tempo em transições entre APs. Quando o tempo de espera é menor que o tempo de transição entre APs, a vazão cai consideravelmente. O fato das estações ficarem refazendo o cálculo constantemente não é um problema, o que se torna um problema é ter mais tempo realizando transições entre APs que trocando dados. O algoritmo MLT apresenta o efeito *ping-pong* por sempre ficar variando os pesos dos APs, esse efeito é mais visível principalmente quando o tempo de espera desse algoritmo é muito pequeno. Como apresentado pelos resultados, para evitar essa contante troca, o melhor resultado foi o que possui o maior tempo de espera. Já nos resultados das curvas dos cenários que possui tráfego de fundo, que são as estações comuns, o algoritmo apresenta um comportamento parecido entre si independente do tempo de espera. Isso ocorre por já haver



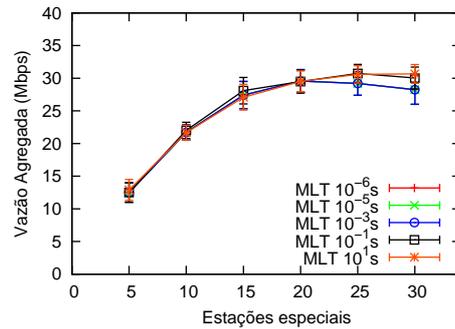
(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.



(c) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.

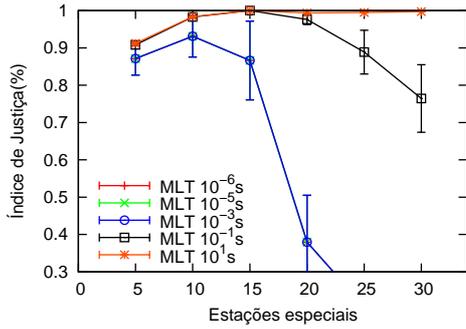


(d) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

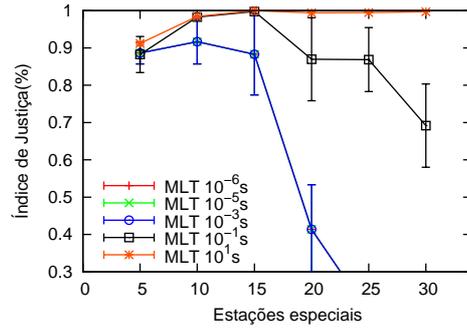
Figura 5.14: Gráficos da vazão do algoritmo MLT utilizando o protocolo IEEE 802.11g.

uma quantidade fixa de estações associadas aos APs, pois as estações comuns não trocam de AP. Como as estações especiais procuram escolher APs que possuem a menor quantidade de estações associadas, o problema é que as estações especiais vão se concentrar nesses APs, e portanto vão concorrer entre si para trocar dados, diminuindo a vazão média das estações especiais.

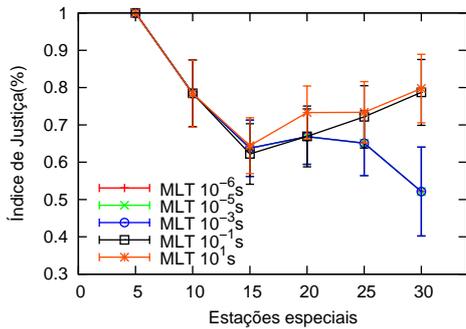
O algoritmo MLT também apresentou nos resultados do índice de justiça, uma grande diferença dos resultados entre os cenários propostos. Nas Figuras 5.15 e 5.16 são apresentados os resultados do índice de justiça do algoritmo MLT. As Figuras 5.15a, 5.15b, 5.16a e 5.16b apresentam resultados similares, e assim como os resultados da vazão agregada desse algoritmo, há um problema relacionado com o tempo de espera. Nos resultados apresentados pelas curvas desse cenário sem estações comuns, quando o tempo de espera é muito pequeno e a quantidade de estações é grande, o índice de justiça é muito baixo, ficando a baixo dos 30%. O melhor resultado de índice de justiça nesse tipo de cenário foi o do algoritmo MLT com tempo de espera de 10 segundos. Entretanto, os resultados para os cenários que possuem um tráfego de fundo gerado por estações comuns, apresentaram um desempenho ruim e



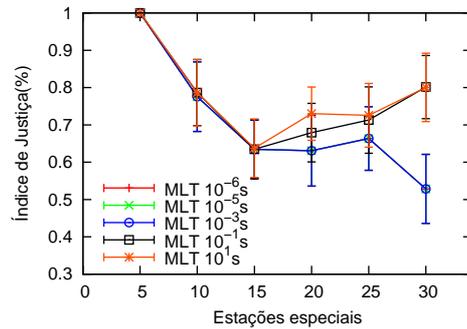
(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.



(c) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.

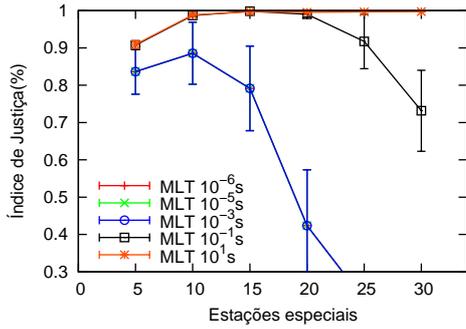


(d) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

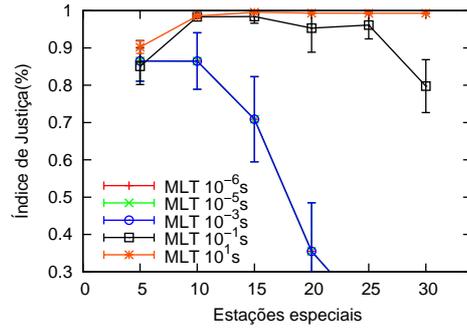
Figura 5.15: Gráficos do índice de justiça do algoritmo MLT utilizando o protocolo IEEE 802.11b.

parecido entre as curvas, e o motivo é o mesmo da vazão agregada. Como uma boa parte das estações especiais se agregam em poucos APs, quando a quantidade destas aumenta, elas começam a disputar o meio entre si, mantendo uma conexão com poucas trocas de dados, ou seja, diminuindo a vazão. Contudo, as estações especiais que ficam em APs mais populadas por estações comuns, possuem uma conexão boa e assim uma vazão bem melhor. Por conta dessa diferença de qualidade de conexões, o índice de justiça varia drasticamente, assim como o intervalo de confiança dos resultados obtidos. Ressaltando que os resultados para os tempos de espera de 10^{-3} segundos ou inferior são os mesmos, e por isso se distinguem apenas 3 curvas nessas figuras.

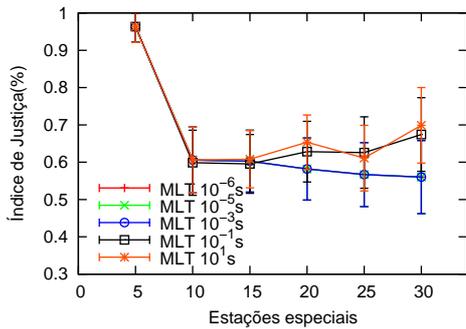
Os resultados obtidos após as simulações do algoritmo MLT, em relação ao tempo de latência das páginas *web*, são apresentados nas Figuras 5.17 e 5.18. Quando há poucas estações especiais no cenário, a latência é baixa e com valores bem próximos entre as curvas. A partir de 15 estações especiais as curvas se diferenciam, as curvas dos resultados de tempo de espera acima de 10^{-3} segundos se distancia um pouco das demais. Isso ocorre porque as estações com tempo de espera maior trocam mais



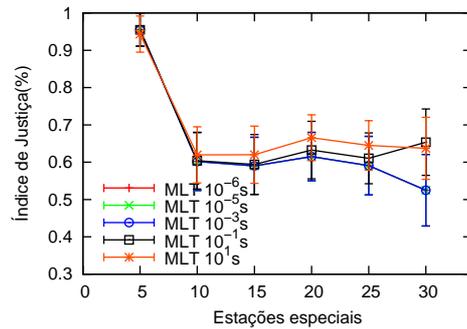
(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e sem estações comuns.



(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e sem estações comuns.

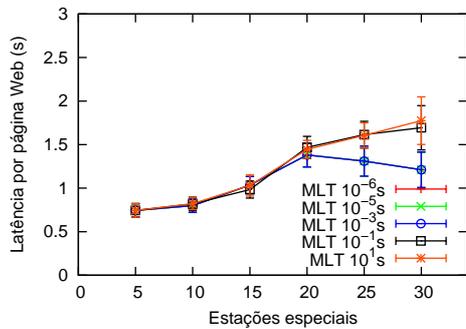


(c) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.

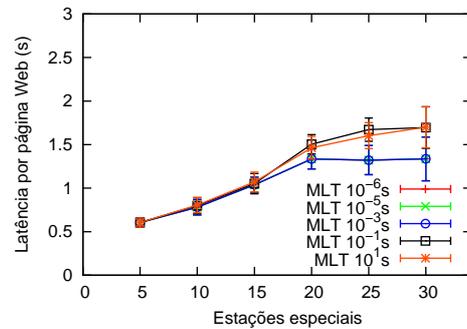


(d) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.16: Gráficos do índice de justiça do algoritmo MLT utilizando o protocolo IEEE 802.11g.



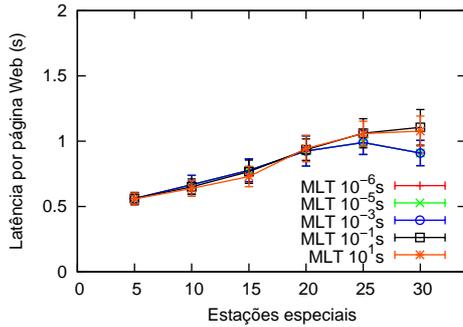
(a) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



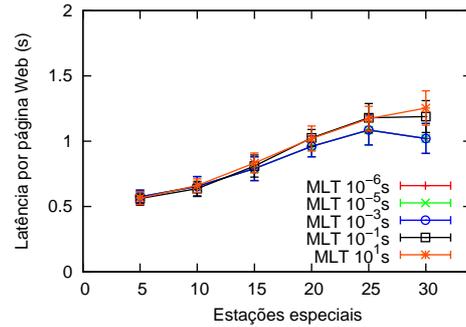
(b) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.17: Gráficos da latência das páginas *web* das estações comuns do algoritmo MLT utilizando o protocolo IEEE 802.11b.

dados e disputam o meio mais vezes com as estações comuns, causando uma latência maior.



(a) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 60% do raio de alcance das estações e com 30 estações comuns.



(b) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 80% do raio de alcance das estações e com 30 estações comuns.

Figura 5.18: Gráficos da latência das páginas *web* das estações comuns do algoritmo MLT utilizando o protocolo IEEE 802.11g.

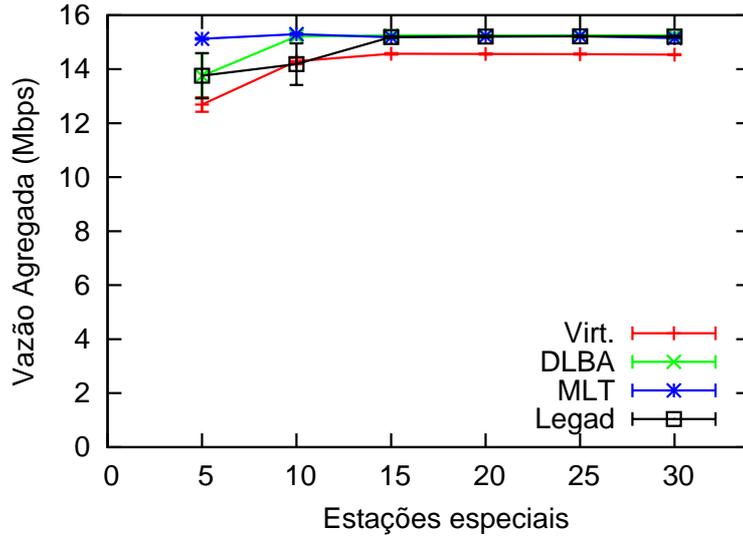
Conclusão Parcial

O algoritmo MLT é muito bom quando há apenas estações que usam esse algoritmo. A configuração padrão desse algoritmo não apresentou bons resultados, por isso foram realizados testes com valores diferentes de tempo de espera. Os resultados obtidos com um tempo de espera maior foi bem melhor que o apresentado como padrão. Entretanto, nos cenários que têm estações comuns, esse algoritmo apresentou um péssimo desempenho, tanto em relação a vazão quanto em índice de justiça. Quanto a latência das páginas *web*, a consequência de não ter tido um vazão alta já resulta em uma diminuição da latência causada por esse algoritmo.

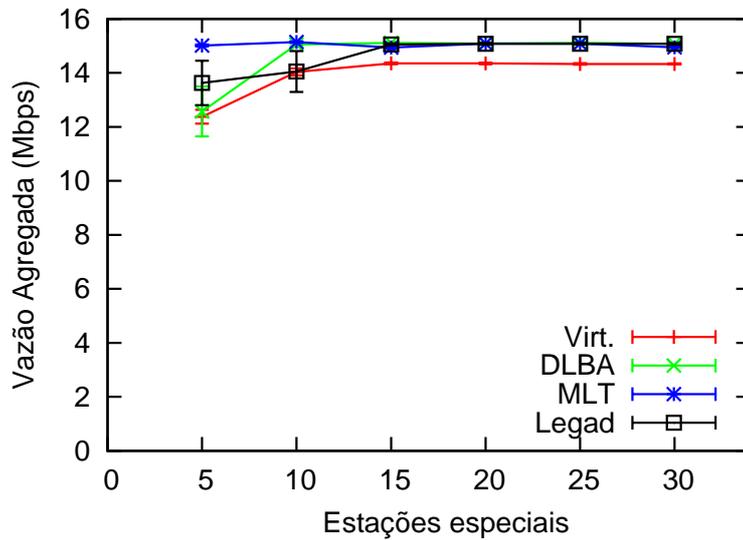
5.4 Comparação entre os Algoritmos de Balanceamento

Para finalizar a apresentação de resultados, foi posto o algoritmo com virtualização que obteve os melhores resultados em comparação como os algoritmos da literatura, além dos que apresentaram os melhores resultados dos algoritmos da literatura, incluindo o algoritmo tradicional de associação. Então, foram escolhidos o algoritmo de virtualização que utiliza a taxa de erro de pacotes e a taxa de transmissão para calcular a vazão média, o algoritmo DLBA que espera 6 tentativas transição, e o algoritmo MLT de tempo de espera de 10 segundos. As Figuras 5.19, 5.20, 5.21, 5.22, 5.23, 5.24, 5.25, 5.26, 5.27 e 5.27, apresentam os resultados de todos esses algoritmos.

De acordo com o que as Figuras 5.19, 5.20, 5.21 e 5.22 o algoritmo que apresentou uma vazão melhor foi o MLT em cenários que não há estações normais, já nos cenários que tem estações comuns para gerar tráfego de fundo, o algoritmo que



(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance de uma estação.

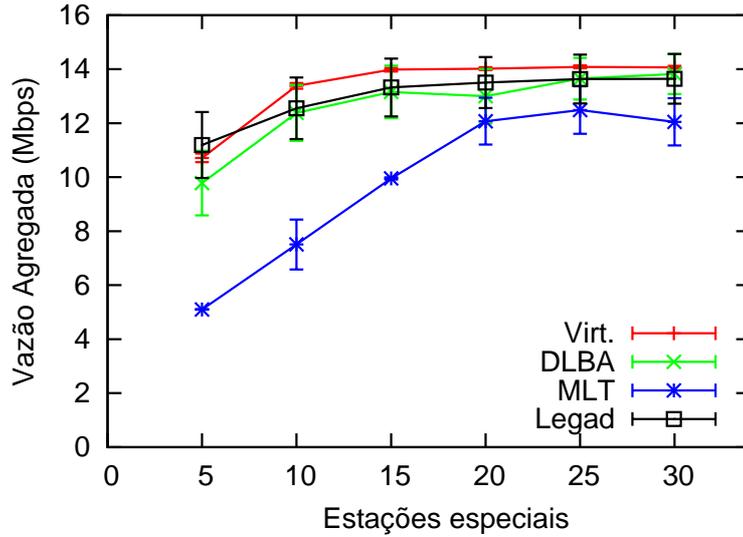


(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance de uma estação.

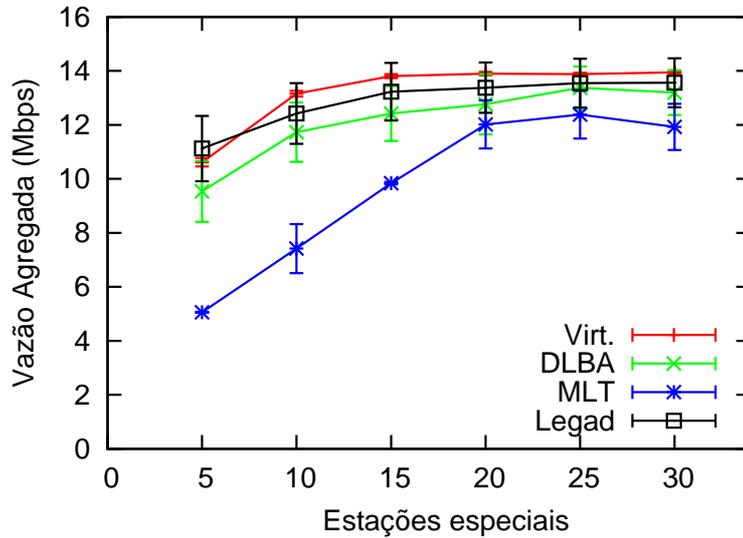
Figura 5.19: Gráficos da vazão dos algoritmos desenvolvidos sem estações comuns e utilizando o protocolo IEEE 802.11b.

apresentou os melhores resultados foi o que utiliza a virtualização. Os resultados obtidos pelo algoritmo de associação tradicional foram muito bons, e próximos ao do algoritmo DLBA.

Os resultados dos índices de justiça apresentados nas Figuras 5.23, 5.24, 5.25 e 5.26 seguem os resultados da vazão agregada. Portanto, nos cenários que não há estações comuns o algoritmo MLT apresentou os melhores índices de justiça, e nos cenários que têm estações comuns o algoritmo que utiliza virtualização. O algoritmo tradicional de associação apresenta o pior resultado dentro os desenvolvidos, apesar disso, nos cenários que têm estações comuns os



(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance de uma estação.

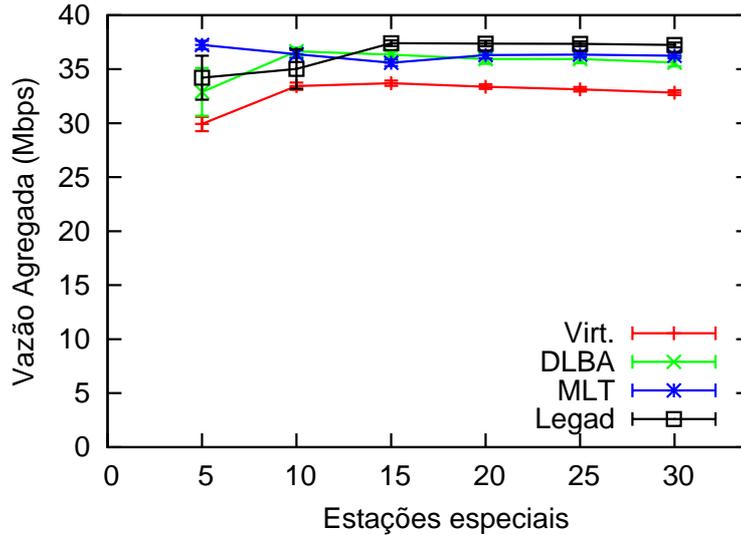


(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance de uma estação.

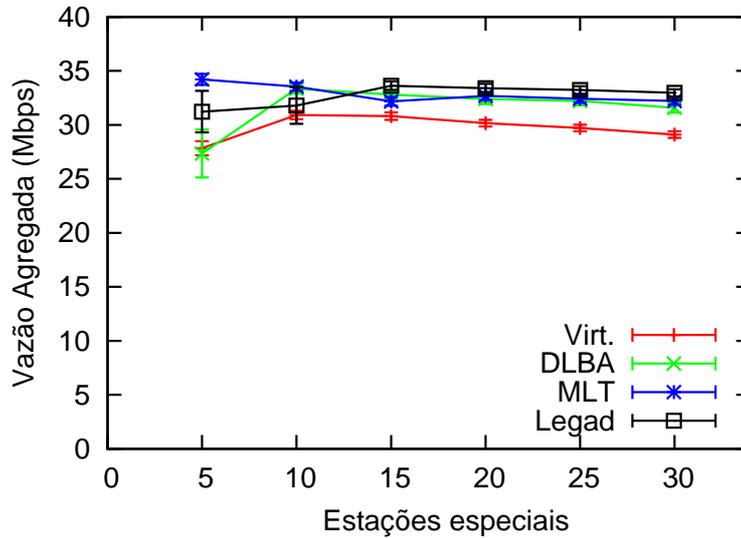
Figura 5.20: Gráficos da vazão dos algoritmos desenvolvidos com 30 estações comuns e utilizando o protocolo IEEE 802.11b.

índices de justiça dos algoritmos MLT, DLBA e o tradicional são bem próximos.

Os resultados da latência das páginas *web* são apresentados nas Figuras 5.27a, 5.27b, 5.28a e 5.28b. O algoritmo MLT foi o que causou uma menor latência nas páginas *web* das estações comuns, em seguida o algoritmo que utiliza virtualização foi o que causou a menor latência.



(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance de uma estação.

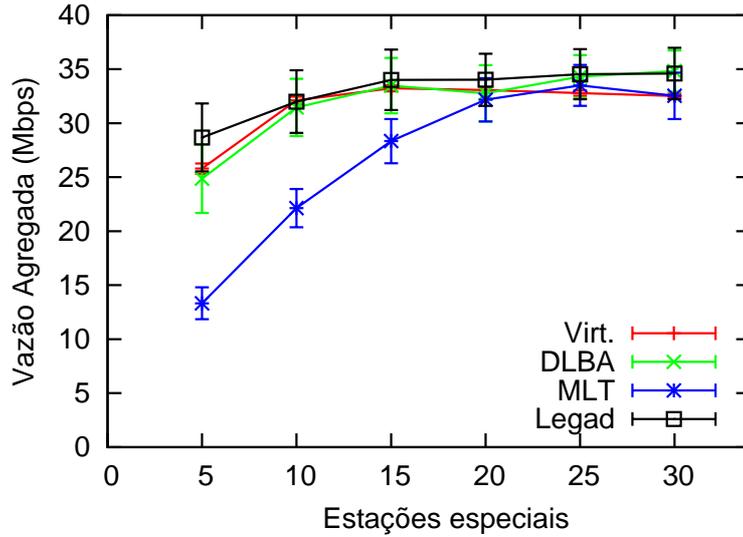


(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance de uma estação.

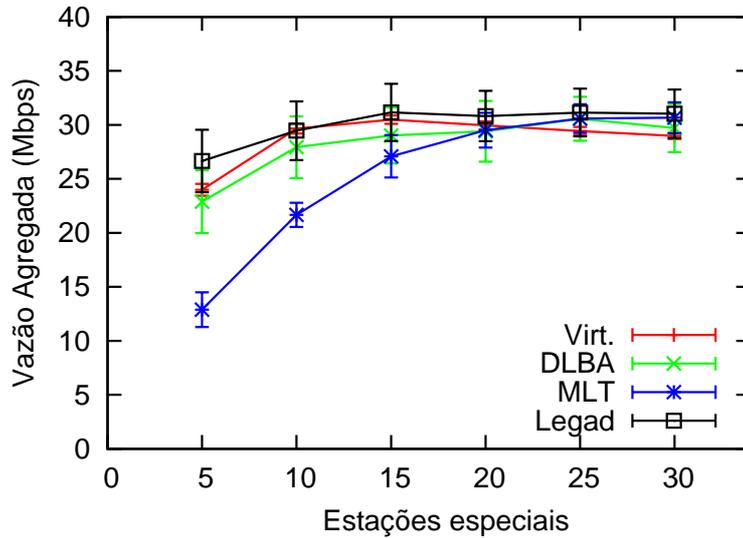
Figura 5.21: Gráficos da vazão dos algoritmos desenvolvidos sem estações comuns e utilizando o protocolo IEEE 802.11g.

5.5 Conclusão do Capítulo

Neste capítulo foram apresentados os resultados de todos os algoritmos e todas as variações destes avaliadas. Os resultados dos algoritmos que utilizam a virtualização apresentam o desempenho das diferentes formas de calcular os pesos deste, apresentados na Seção 5.2. Também foram apresentadas algumas avaliações dos algoritmos da literatura que foram utilizados para comparação, na Seção 5.3 foram apresentados os resultados destes. Na Seção 5.4 foi realizada uma comparação entre todos os algoritmos apresentados nesse trabalho, cada algoritmo configurado com sua melhor



(a) Gráfico da vazão agregada no cenário com lado de 60% do raio de alcance de uma estação.

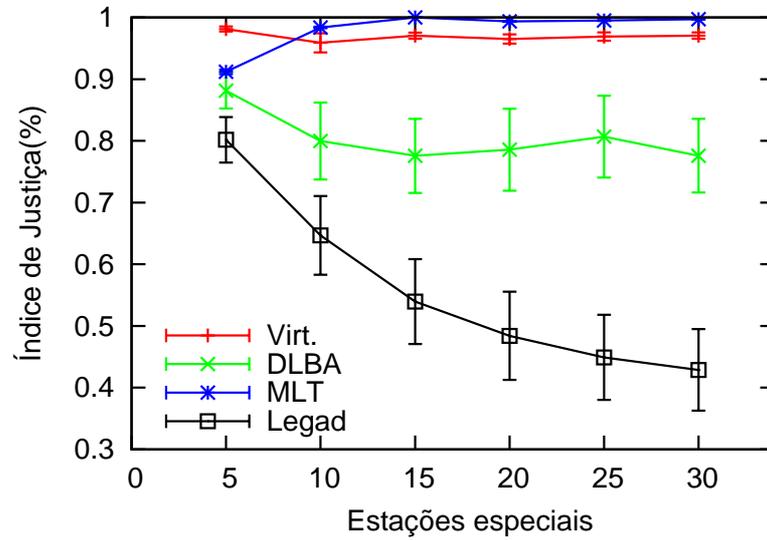


(b) Gráfico da vazão agregada no cenário com lado de 80% do raio de alcance de uma estação.

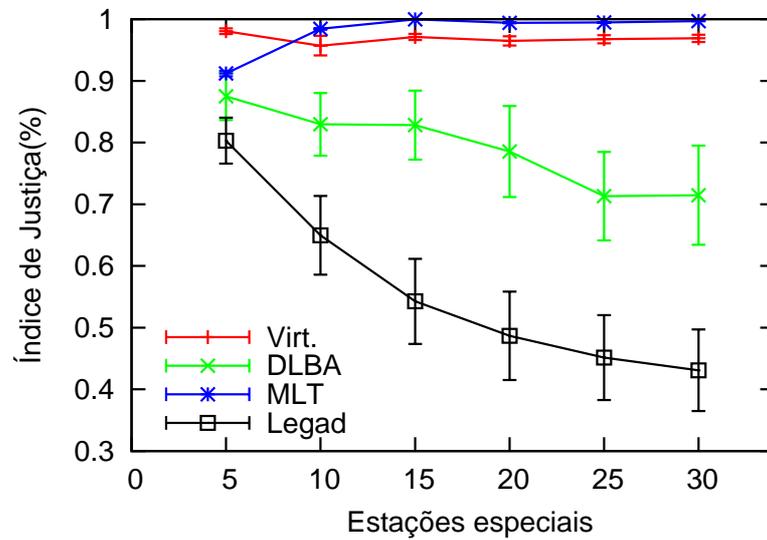
Figura 5.22: Gráficos da vazão dos algoritmos desenvolvidos com 30 estações comuns e utilizando o protocolo IEEE 802.11g.

avaliação.

Os resultados apresentados nesse capítulo demonstram que dependendo do cenário o algoritmo de balanceamento de carga que utiliza virtualização possui os melhores resultados. Nos cenários que não têm estações que geram tráfego de fundo, o algoritmo com virtualização apresenta bons resultados, mas o algoritmo MLT apresenta os melhores resultados neste cenário. Contudo, nos cenários com estações comuns, o algoritmo com virtualização demonstra resultados bem estáveis, com intervalo de confiança muito pequeno, além disso, a os resultados do índice de justiça se apresenta bem melhor que os dos outros algoritmos.

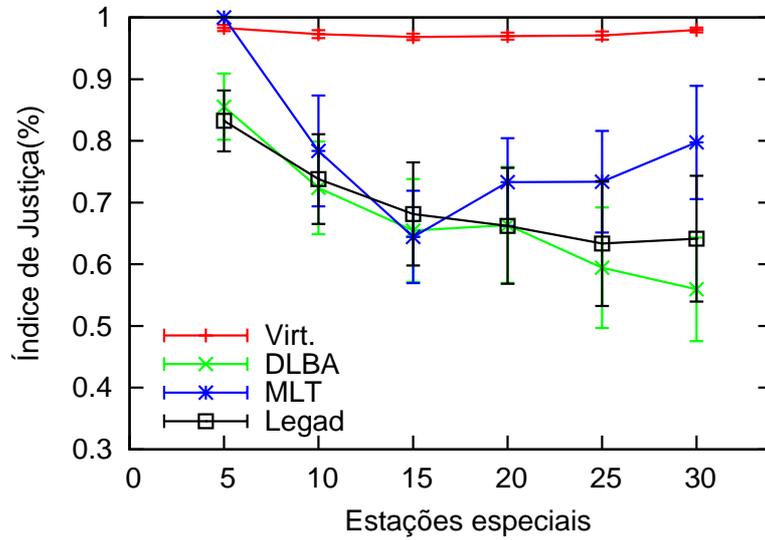


(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance de uma estação.

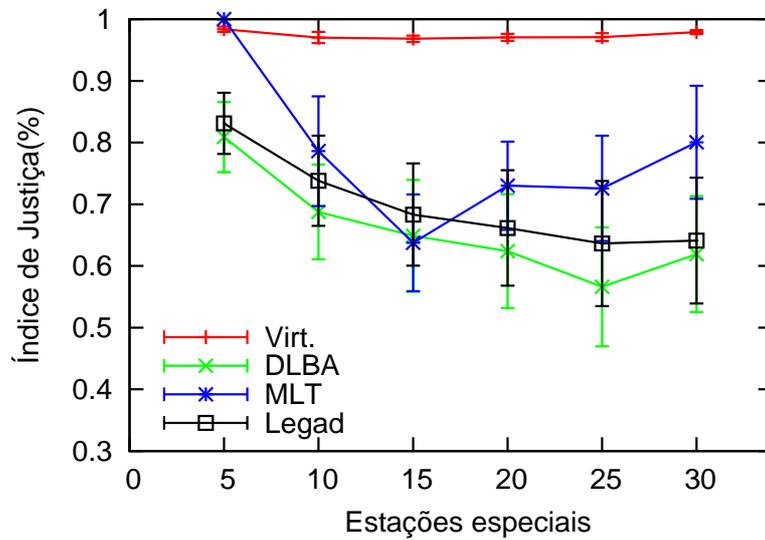


(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance de uma estação.

Figura 5.23: Gráficos do índice de justiça dos algoritmos desenvolvidos sem estações comuns e utilizando o protocolo IEEE 802.11b.

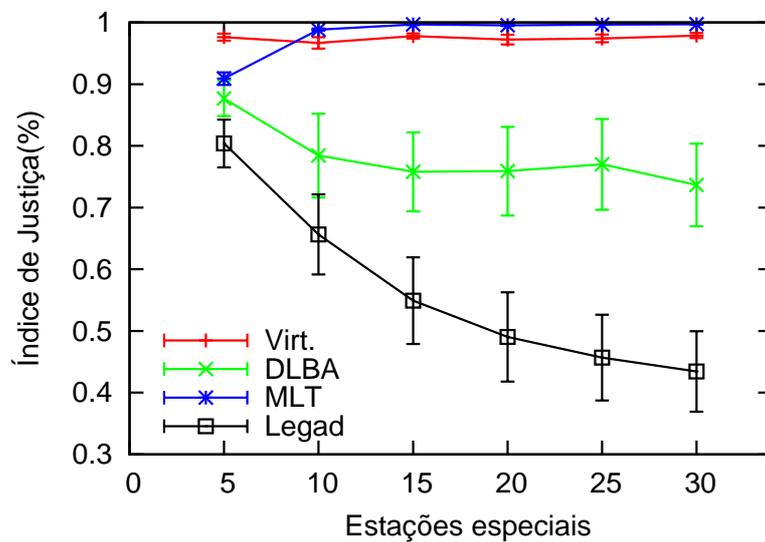


(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance de uma estação.

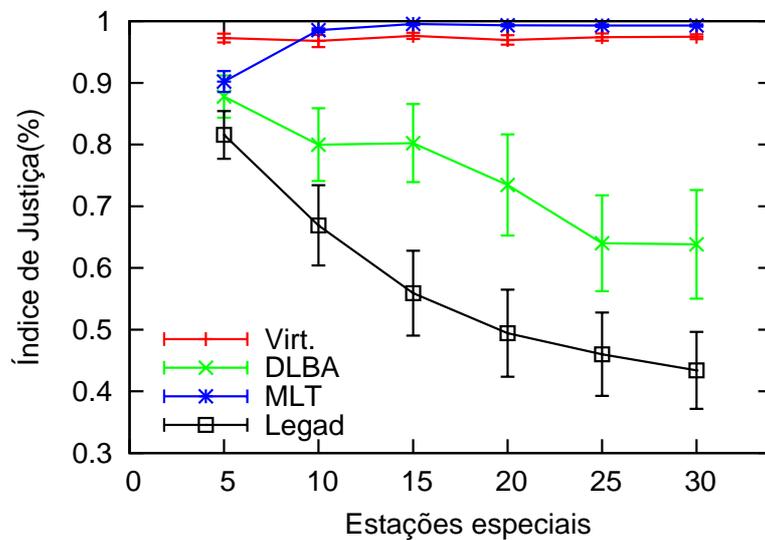


(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance de uma estação.

Figura 5.24: Gráficos do índice de justiça dos algoritmos desenvolvidos com 30 estações comuns e utilizando o protocolo IEEE 802.11b.

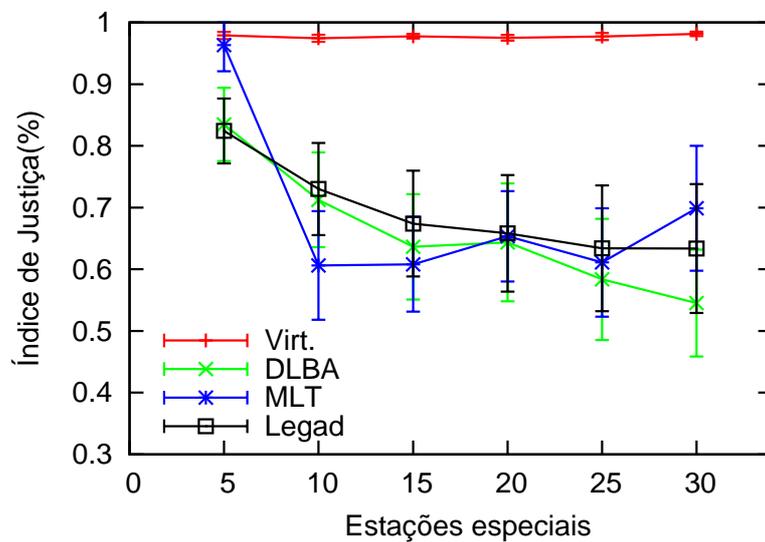


(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance de uma estação.

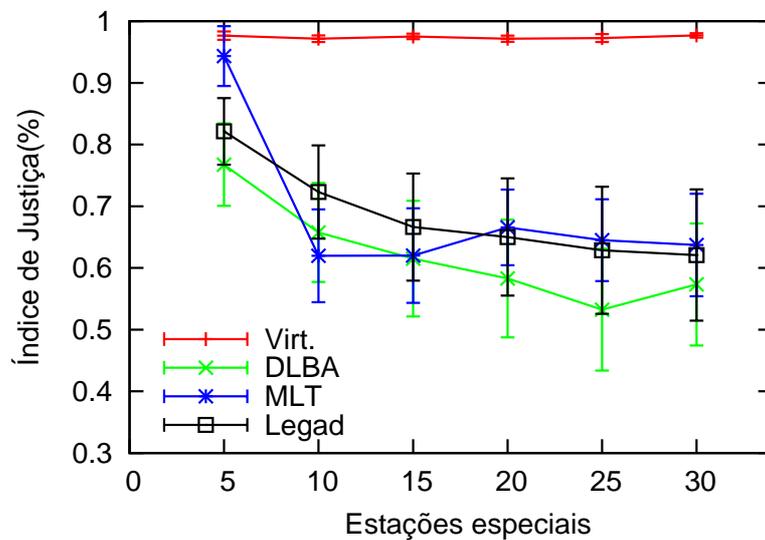


(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance de uma estação.

Figura 5.25: Gráficos do índice de justiça dos algoritmos desenvolvidos sem estações comuns utilizando o protocolo IEEE 802.11g.

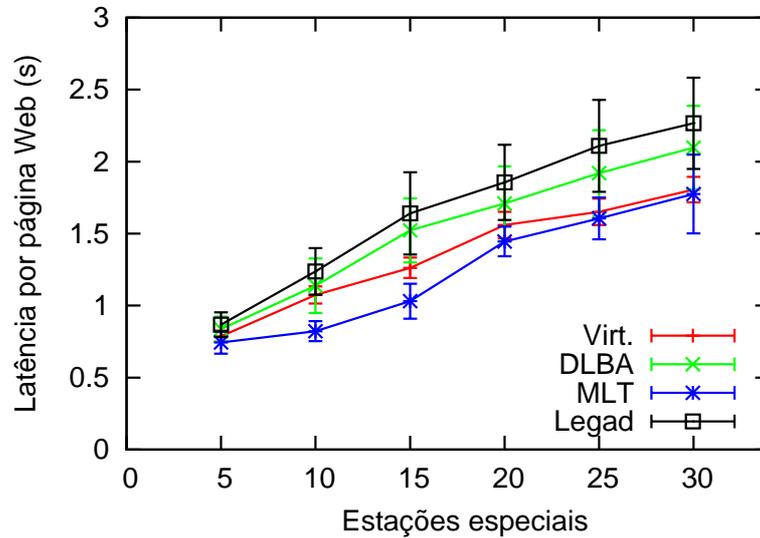


(a) Gráfico do índice de justiça no cenário com lado de 60% do raio de alcance de uma estação.

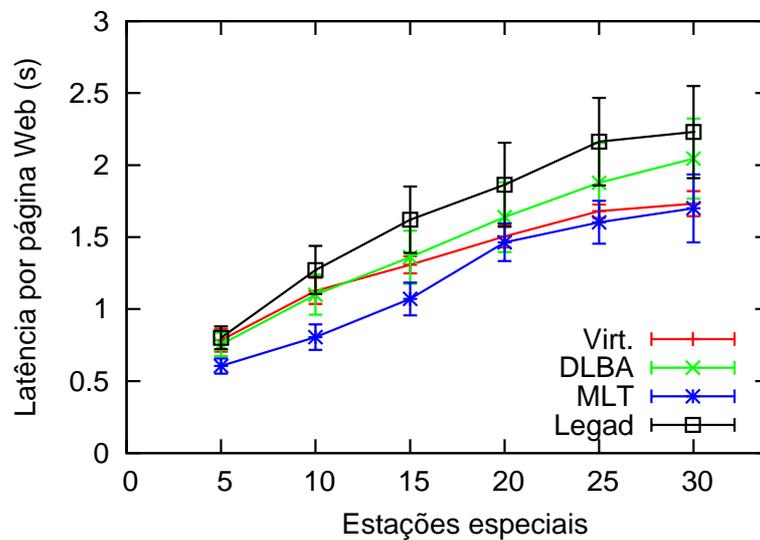


(b) Gráfico do índice de justiça no cenário com lado de 80% do raio de alcance de uma estação.

Figura 5.26: Gráficos do índice de justiça dos algoritmos desenvolvidos com 30 estações comuns e utilizando o protocolo IEEE 802.11g.

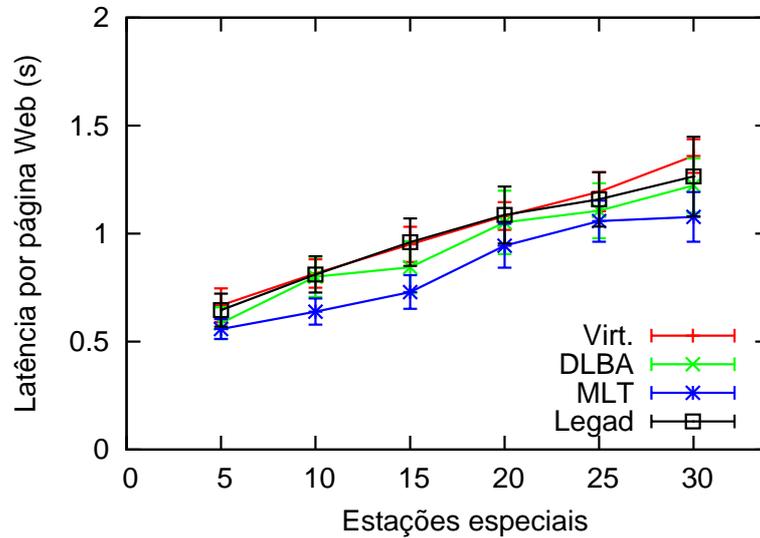


(a) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 60% do raio de alcance de uma estação e com 30 estações comuns.

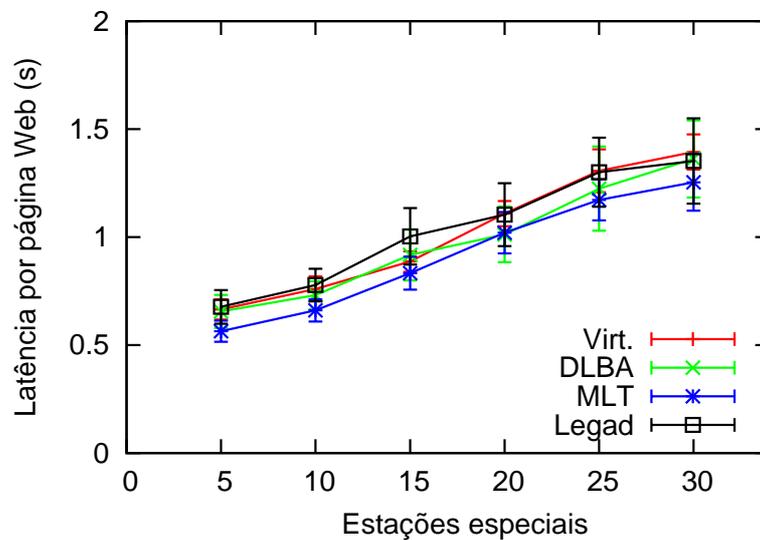


(b) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 80% do raio de alcance de uma estação e com 30 estações comuns.

Figura 5.27: Gráficos da latência das páginas *web* das estações comuns dos algoritmos desenvolvidos utilizando o protocolo IEEE 802.11b.



(a) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 60% do raio de alcance de uma estação e com 30 estações comuns.



(b) Gráfico da latência das páginas *web* das estações comuns no cenário com lado de 80% do raio de alcance de uma estação e com 30 estações comuns.

Figura 5.28: Gráficos da latência das páginas *web* das estações comuns dos algoritmos desenvolvidos utilizando o protocolo IEEE 802.11g.

Capítulo 6

Conclusões e Trabalhos Futuros

O balanceamento de carga em redes IEEE 802.11 infraestruturadas é um problema para o qual há inúmeras soluções. Uma grande parte dessas soluções modificam o padrão ou utilizam recursos centralizados para conseguir suas métricas e assim poder resolver o problema. A fim de resolver o problema de uma maneira que não infrinja o padrão e que seja distribuída, a virtualização parece ser uma boa solução. Além da utilização dos protocolos IEEE 802.11r e IEEE 802.11k para acelerar a transição entre APs e adquirir medições do AP, respectivamente.

O trabalho realizado durante o mestrado foi executado em etapas.

- Foram realizados estudos em cima de trabalhos que utilizam a virtualização de redes IEEE 802.11, e foi proposto uma técnica de balanceamento de carga utilizando virtualização.
- Iniciou estudos para avaliar que métricas eram utilizadas nos algoritmos de balanceamento de carga. Então, foi decidido que a ocupação do meio e a taxa de erro de pacotes, ambas aliadas à quantidade de estações associadas a um AP, seriam boas métricas para avaliar e comparar como ficaria a distribuição dos recursos da rede.
- Em seguida, iniciou a etapa de modificação necessárias do simulador NS-2 para oferecer as funcionalidades dos protocolos IEEE 802.11r e IEEE 802.11k. A partir dessas funcionalidades seria possível utilizar a virtualização e as métricas para desenvolver o algoritmo de balanceamento de carga proposto.
- Desenvolvimento dos algoritmos propostos e dos algoritmos da literatura.
- Simulações em cenários aleatórios a fim de avaliar distribuição dos recursos da rede, com e sem estações que não utilizam algoritmo de balanceamento de carga. Os resultados apresentados foram relacionados a vazão, índice de justiça e atrasos causados nas estações que não usam algoritmo de balanceamento de carga.

Nesta dissertação, foi apresentada uma nova técnica para balanceamento de carga em redes IEEE 802.11 através do uso de virtualização, onde esta foi avaliada e comparada com alguns algoritmos da literatura e também a abordagem padrão. Foram desenvolvidos alguns algoritmos combinando a técnica de virtualização FH proposta e as métricas escolhidas. Dentre os algoritmos propostos foi escolhido o que mais se destacou, e este foi comparado aos algoritmos da literatura. Os algoritmos da literatura foram desenvolvidos e avaliados, assim foi possível escolher o melhor ajuste dos seus parâmetros.

Os resultados apresentados, em relação à abordagem proposta, demonstram que há uma sobrecarga por conta da utilização da virtualização. Apesar disso, essa sobrecarga é negligenciável pois a proposta conseguiu obter ocupação máxima da capacidade disponível na rede. Apesar do algoritmo MLT apresentar os melhores resultados nos cenários que não possuem estações comuns, este não apresenta um bom desempenho em cenários com estações comuns. O algoritmo DLBA apresenta resultados medianos em todos os cenários, apesar do grande intervalo de confiança. Já o algoritmo padrão apresenta o pior resultado e um intervalo de confiança tão grande quanto o algoritmo DLBA em todos os cenários.

A solução proposta por essa dissertação apresentou ganhos significativos nos cenários propostos em que há estações normais. Contudo, os resultados para cenários que não há estações normais são bem próximos dos do algoritmo MLT, além de serem bem mais estáveis. O intervalo de confiança é bem pequeno e os resultados não apresentam tanta grande variação entre os cenários com e sem estações normais. Portanto, a solução proposta oferece a todos as estações especiais uma vazão média aproximadamente igual em todos os cenários, e também minimiza o impacto em estações comuns, evitando a degradação de desempenho destas. Então, é possível concluir que a solução utilizando virtualização para balanceamento de carga em redes IEEE 802.11 se apresenta como uma solução vantajosa e resistente às variações do meio e dos cenários.

Como trabalhos futuros, pretende-se criar cenários mais densos com mobilidade para avaliar impacto da técnica proposta, e compará-la a propostas da literatura que atuam nesses tipos de cenários. Avaliar outros possíveis mecanismo para agregar a técnica, como por exemplo a utilização de um protocolo de qualidade de serviço. Também pode ser avaliada a mesclagem de redes *ad-hoc* e infraestruturada realizando virtualização entre estas e balanceando a carga. Além disso, pretende-se implementar o mecanismo de virtualização proposto nesta dissertação em interfaces IEEE 802.11 disponíveis comercialmente que utilizem o *driver* MadWifi de código aberto.

Referências Bibliográficas

- [1] IEEE802.11. *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. Relatório técnico, IEEE Std. 802.11, June 2007.
- [2] BEJERANO, Y., HAN, S.-J., LI, L. E. “Fairness and load balancing in wireless LANs using association control”. In: *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pp. 315–329, New York, NY, USA, September 2004. ACM.
- [3] BALACHANDRAN, A., VOELKER, G. M., BAHL, P., et al. “Characterizing user behavior and network performance in a public wireless LAN”, *SIGMETRICS Perform. Eval. Rev.*, v. 30, n. 1, pp. 195–205, 2002. ISSN: 0163-5999. doi: <http://doi.acm.org/10.1145/511399.511359>.
- [4] KOTZ, D., ESSIEN, K. “Analysis of a campus-wide wireless network”, *Wireless Network*, v. 11, n. 1-2, pp. 115–133, 2005. ISSN: 1022-0038. doi: <http://dx.doi.org/10.1007/s11276-004-4750-0>.
- [5] BALAZINSKA, M., CASTRO, P. “Characterizing mobility and network usage in a corporate wireless local-area network”. In: *MobiSys '03: Proceedings of the 1st international conference on Mobile systems, applications and services*, pp. 303–316, New York, NY, USA, 2003. ACM. doi: <http://doi.acm.org/10.1145/1066116.1066127>.
- [6] YEN, L.-H., YEH, T.-T., CHI, K.-H. “Load Balancing in IEEE 802.11 networks”, *IEEE Internet Computing*, v. 13, pp. 56–64, January 2009. ISSN: 1089-7801. doi: <http://doi.ieeecomputersociety.org/10.1109/MIC.2009.11>.
- [7] SHEU, S.-T., WU, C.-C. “Dynamic Load Balance Algorithm (DLBA) for IEEE 802.11 Wireless LAN”, *Tamkang Journal of Science and Engineering*, v. 2, pp. 45–52, 1999.
- [8] PAPANIKOS, I., LOGOTHETIS, M. “A Study on Dynamic Load Balance for IEEE 802.11b Wireless LAN”. In: *8th International Conference on Ad-*

vances in Communications and Control, COMCON 8, Rethymna, Crete, Greece, June 2001.

- [9] PAUL, S., SESHAN, S. *Technical Document on Wireless Virtualization*. Relatório Técnico GENI Technical Report GDD-06-17, GENI Wireless Working Group, September 2006.
- [10] NAKAO, A. “Network Virtualization as Foundation for Enabling New Network Architectures and Applications”, *IEICE Transactions*, v. 93-B, n. 3, pp. 454–457, 2010.
- [11] BAHL, P., CHANDRA, R., DUNAGAN, J. “SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks”. In: *Mobile Computing and Networking (MobiCom)*, pp. 216–230, Philadelphia, Pennsylvania, USA, September 2004. doi: 10.1145/1023720.1023742.
- [12] CHANDRA, R., BAHL, P. “MultiNet: Connecting to multiple IEEE 802.11 networks using a single wireless card”. In: *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, v. 2, pp. 882–893 vol.2, March 2004.
- [13] CHANDRA, R., PADMANABHAN, V. N., ZHANG, M. “WiFiProfiler: cooperative diagnosis in wireless LANs”. In: *International Conference on Mobile Systems, Applications, and Services*, pp. 205–219, Uppsala, Sweden, June 2006. doi: 10.1145/1134680.1134702.
- [14] NICHOLSON, A. J., WOLCHOK, S., NOBLE, B. D. “Juggler: Virtual Networks for Fun and Profit”, *IEEE Transactions on Mobile Computing*, v. 9, pp. 31–43, 2010. ISSN: 1536–1233. doi: <http://doi.ieeecomputersociety.org/10.1109/TMC.2009.97>.
- [15] BAHL, P., ADYA, A., PADHYE, J., et al. “Reconsidering wireless systems with multiple radios”, *Computer Communication Review*, v. 34, pp. 39–46, 2004. doi: 10.1145/1039111.1039122.
- [16] MIU, A. K., BALAKRISHNAN, H., KOKSAL, C. E. “Improving Loss Resilience with Multi-Radio Diversity in Wireless Networks”. In: *11th ACM MobiCom Conference*, pp. 16–30, Cologne, Germany, September 2005.
- [17] QURESHI, A., CARLISLE, J. N., GUTTAG, J. V. “Tavarua: video streaming with WWAN striping”. In: *ACM Multimedia Conference*, pp. 327–336, Santa Clara, CA, USA, October 2006. doi: 10.1145/1180639.1180714.

- [18] BARHAM, P., DRAGOVIC, B., FRASER, K., et al. “Xen and the art of virtualization”. In: *SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles*, pp. 164–177, New York, NY, USA, 2003. ACM.
- [19] DIKE, J. “The User-mode Linux kernel home page”. 2002. Disponível em: <http://user-mode-linux.sourceforge.net>.
- [20] SMITH, G., CHATURVEDI, A., MISHRA, A., et al. “Wireless virtualization on commodity 802.11 hardware”. In: *WinTECH '07: Proceedings of the the second ACM international workshop on Wireless network testbeds, experimental evaluation and characterization*, pp. 75–82, New York, NY, USA, September 2007. ACM. ISBN: 978-1-59593-738-4. doi: <http://doi.acm.org/10.1145/1287767.1287782>.
- [21] RAMANI, I., SAVAGE, S. “SyncScan: Practical fast handoff for 802.11 infrastructure networks”. In: *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, v. 1, pp. 675–684 vol. 1, March 2005. doi: 10.1109/INFCOM.2005.1497933.
- [22] VELAYOS, H., KARLSSON, G. “Techniques to reduce IEEE 802.11b MAC layer handover time”. In: *in Proceedings International Conference on Computer Communication (ICC)*, v. 7, pp. 3844–3848, Washington, DC, USA, June 2004. IEEE Computer Society.
- [23] RIZVI, S., KHAN, M., RIASAT, A. “Active Scanning: A better approach to reduce handover time at MAC layer for wireless networks”. In: *Computer, Control and Communication, 2009. IC4 2009. 2nd International Conference on*, pp. 1–4, Feb. 2009. doi: 10.1109/IC4.2009.4909172.
- [24] ABODA, B., BLUNK, L. J., VOLLBRECHT, J. R., et al. *Extensible Authentication Protocol (EAP)*. RFC, IETF, June 2004. Disponível em: <http://www.ietf.org/rfc/rfc3748.txt>.
- [25] CLANCY, T. C. “Secure handover in enterprise WLANs: CAPWAP, HOKEY and IEEE 802.11r”, *IEEE Wireless Communication*, v. 15, n. 5, pp. 80–85, October 2008. ISSN: 1536-1284. doi: <http://dx.doi.org/10.1007/s11276-004-4750-0>.
- [26] MISHRA, A., SHIN, M., ARBAUGH, W. “An empirical analysis of the IEEE 802.11 MAC layer handoff process”, *SIGCOMM Comput. Commun. Rev.*,

v. 33, n. 2, pp. 93–102, 2003. ISSN: 0146-4833. doi: <http://doi.acm.org/10.1145/956981.956990>.

- [27] WANG, W.-C., HSU, C.-H., CHEN, Y.-M., et al. “SCTP-based Handover for VoIP over IEEE 802.11 WLAN Using Device Virtualization”, *Advanced Communication Technology, The 9th International Conference on*, v. 2, pp. 1073–1076, February 2007.
- [28] ZHANG, L. J., PIERRE, S. “Optimizing the performance of handoff management in wireless LANs”, *International Journal of computer Science and Network Security*, v. 8, n. 7, pp. 87–94, July 2008.
- [29] CALHOUN, P., MONTEMURRO, M., STANLEY, D. *Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification*. RFC, IETF, March 2009. Disponível em: <http://tools.ietf.org/html/rfc5415.html>.
- [30] CALHOUN, P., MONTEMURRO, M., STANLEY, D. *Control And Provisioning of Wireless Access Points (CAPWAP) Binding for IEEE 802.11*. RFC, IETF, March 2009. Disponível em: <http://tools.ietf.org/html/rfc5416.html>.
- [31] CLANCY, T., NAKHJIRI, M., NARAYANAN, V., et al. *Handover Key Management and Re-Authentication Problem Statement*. RFC, IETF, March 2008. Disponível em: <http://www.ietf.org/rfc/rfc5169.html>.
- [32] IEEE802.11R. *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2: Fast Basic Service Set (BSS)*. Relatório técnico, IEEE Std. 802.11r, September 2008.
- [33] ZHENG, X., SARIKAYA, B. “Handover keying and its uses”, *IEEE Wireless Communication*, v. 23, n. 2, pp. 27–34, March 2009. ISSN: 0890-8044. doi: <http://dx.doi.org/10.1007/s11276-004-4750-0>.
- [34] NARAYANAN, V., DONDETI, L. *EAP Extension for EAP Re-authentication Protocol (EAP)*. RFC, IETF, August 2008. Disponível em: <http://www.ietf.org/rfc/rfc5296.txt>.
- [35] TABASSAM, A. A., TRSEK, H., HEISS, S., et al. “Fast and Seamless Handover for Secure Mobile Industrial Applications with 802.11r”. In: *Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on*, pp. 750–757, Washington, DC, USA, Oct 2009. IEEE Computer Society. doi: <http://doi.ieeecomputersociety.org/10.1109/LCN.2009.5355069>.

- [36] IEEE802.11F. *Recommended Practice for Multi-Vender Access Point Interoperability via an Inter-Access Point Protocol Across Distribution Systems Supporting IEEE 802.11 Operation*. Relatório técnico, IEEE Std. 802.11f, June 2003.
- [37] VASUDEVAN, S., PAPAGIANNAKI, K., DIOT, C., et al. “Facilitating access point selection in IEEE 802.11 wireless networks”. In: *IMC '05: Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*, pp. 26–26, Berkeley, CA, USA, October 2005. USENIX Association.
- [38] YEN, L.-H., YEH, T.-T. “SNMP-Based Approach to Load Distribution in IEEE 802.11 Networks”. In: *VTC Spring*, pp. 1196–1200, Melbourne, Australia, 2006. doi: 10.1109/VETECS.2006.1683024.
- [39] VILLEGAS, E. G., FERRE, R. V., ASPAS, J. P. “Load Balancing in WLANs through IEEE 802.11k Mechanisms”, *Computers and Communications, 2006. ISCC '06. Proceedings. 11th IEEE Symposium on*, pp. 844–850, June 2006.
- [40] FUJIWARA, A., SAGARA, Y., NAKAMURA, M. “Access point selection algorithms for maximizing throughput in wireless LAN environment”. In: *ICPADS '07: Proceedings of the 13th International Conference on Parallel and Distribution Systems*, v. 1, pp. 1–8, Washington, DC, USA, December 2007. IEEE Computer Society. ISBN: 978-1-4244-1889-3. doi: <http://dx.doi.org/10.1109/ICPADS.2007.4447746>.
- [41] VILLEGAS, E. G., FERRE, R. V., ASPAS, J. P. “Cooperative load balancing in IEEE 802.11 networks with cell breathing”, *Computers and Communications, 2008. ISCC 2008. IEEE Symposium on*, pp. 1133–1140, July 2008.
- [42] ALEO, V. *Load distribution in IEEE 802.11 cells*. Tese de Mestrado, KHL, Royal Institute of Technology, Stockholm, Sweden, March 2003. Adviser-Héctor Velayos.
- [43] BRICKLEY, O., REA, S., PESCH, D. “Load Balancing for QoS enhancement in IEEE 802.11e using cell breathing techniques”. In: *WMASH '06: Proceedings of the 4th International Workshop on Wireless Mobile Application and Service on WLAN Hotspots*, pp. 11–19, New York, NY, USA, September 2006. Association for Computing Machinery (ACM). ISBN: 1-59593-470-7. doi: <http://dx.doi.org/10.1145/1161023.1161026>.

- [44] WANG, Y., CUTHBERT, L. G., BIGHAM, J. “Intelligent radio resource management for IEEE 802.11 WLAN”. In: *Wireless Communication Networking Conference*, v. 3, pp. 1365–1370, Washington, DC, USA, March 2004. IEEE Computer Society. ISBN: 0-7803-8344-3. doi: <http://dx.doi.org/10.1109/WCNC.2004.1311641>.
- [45] ATHANASIOU, G., KORAKIS, T., ERCETIN, O., et al. “Dynamic Cross-Layer Association in 802.11-Based Mesh Networks”, *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pp. 2090–2098, May 2007.
- [46] IEEE802.11K. *Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification Amendment 1: Radio Resource Measurement of Wireless LANs*. Relatório técnico, IEEE Std. 802.11k, September 2008.
- [47] DE OLIVEIRA, T. M., CARDOSO, K. V., DA SILVA, M. W. R., et al. “Balanceamento de Carga Utilizando Virtualização em Redes IEEE 802.11”. In: *XXVII Simpósio Brasileiro de Redes de Computadores - SBRC'09*, pp. 511–524, Recife, PE, Brasil, Maio 2009.
- [48] DE OLIVEIRA, T. M., CARDOSO, K. V., DA SILVA, M. W. R., et al. “Virtualization for Load Balance on IEEE 802.11 Networks”. In: *7th International ICST Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services - MobiQuitous'2010*, Sidney, Australia, December 2010.
- [49] FUKUDA, Y., ABE, T., OIE, Y. “Decentralized access point selection architecture for wireless LANs”. In: *WTS '04: Proceedings of Wireless Telecommunications Symposium*, pp. 137–145, Washington, DC, USA, May 2004. IEEE Computer Society. ISBN: 0-7803-8246-3. doi: <http://dx.doi.org/10.1109/WTS.2004.1319576>.
- [50] LEE, J. J., GUPTA, M. *A new traffic model for current user web browsing behavior*. Relatório técnico, Intel Corporation, Santa Clara, CA, USA, September 2007.
- [51] JAIN, R. K., CHIU, D.-M. W., HAWKES, W. R. “A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System”, *submitted for publication to ACM Transaction on Computer Systems*, Sep 1984.