



M 2015

Plataforma de simulação ns-3 para avaliação de um protocolo de ligação de dados fiável sobre canais IEEE 802.11

PEDRO JOSÉ VINHAS DE BRITO
DISSERTAÇÃO DE MESTRADO APRESENTADA
À FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO EM
ÁREA CIENTÍFICA

A Dissertação intitulada

“Plataforma de simulação ns-3 para avaliação de um protocolo de ligação de dados fiável sobre canais IEEE 802.11”

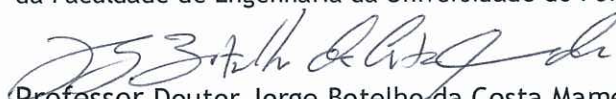
foi aprovada em provas realizadas em 16-10-2015

o júri



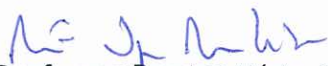
Presidente Professor Doutor Ricardo Santos Morla

Professor Auxiliar do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto



Professor Doutor Jorge Botelho da Costa Mamede

Professor Adjunto do Departamento de Engenharia Eletrotécnica da Instituto Superior de Engenharia do Porto



Professor Doutor Mário Jorge Moreira Leitão

Professor Associado do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

O autor declara que a presente dissertação (ou relatório de projeto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extratos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são corretamente citados.



Autor - Pedro José Vinhas de Brito

Faculdade de Engenharia da Universidade do Porto

Faculdade de Engenharia da Universidade do Porto



**Plataforma de simulação ns-3 para avaliação de
um protocolo de ligação de dados fiável sobre
canais IEEE 802.11**

Pedro José Vinhas de Brito

VERSÃO FINAL

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Eletrotécnica e de Computadores
Major Telecomunicações

Orientador: Mário Jorge Moreira Leitão (PhD)
Coorientador: José António Ruela Simões Fernandes (PhD)
Coorientador: Rui Lopes Campos (PhD)

Outubro 16, 2015

Resumo

A presente dissertação tem como principal objetivo a caracterização de um novo protocolo de comunicação designado TPAP (*TDMA Polling with Ack Piggyback*) utilizando a ferramenta de simulação ns-3.

O protocolo TPAP foi desenvolvido para oferecer uma solução fiável e com elevado desempenho em redes baseadas em protocolos de acesso do tipo CSMA/CA (adotados nos *standards* da família IEEE 802.11). Suporta comunicação bidirecional e tem como alvo uma topologia ponto-multiponto, constituindo uma camada protocolar normalmente designada 2.5, dado que se situa entre a camada MAC e a camada IP.

O protocolo TPAP é baseado na retransmissão seletiva de tramas e combina controlo centralizado de acesso ao meio, baseado em *polling*, com um mecanismo de recuperação de erros do tipo janela deslizante. A janela constitui igualmente uma forma de controlo de fluxo.

Foram desenvolvidas duas versões tendo em atenção as funcionalidades das diferentes variantes da família IEEE 802.11. A versão completa engloba todos os mecanismos supracitados, e tem como alvo principal os *standards* que não suportam agregação de tramas. Uma versão mais simples destina-se aos *standards* que suportam no nível MAC a agregação de tramas e a respetiva confirmação em blocos, o que justifica a exclusão do mecanismo de retransmissão seletiva de tramas do TPAP.

Para além do enquadramento do trabalho e seus objetivos, e da revisão da literatura e do estado da arte, é descrita a metodologia adotada para a implementação das duas versões do protocolo em ns-3, sendo no final apresentados e analisados os resultados das simulações.

Foram considerados diferentes cenários de teste, com o objetivo de comparar as duas versões do protocolo. Para evidenciar as suas vantagens, foram também realizadas, em idênticas condições de tráfego, simulações sem o protocolo TPAP, o que permitiu emular a execução de IP diretamente sobre os mecanismos nativos do *standard* IEEE 802.11, considerando como alternativas a ativação e a inativação do mecanismo opcional de RTS/CTS.

Palavras-Chave: Confirmação de tramas em bloco, IEEE 802.11, ns-3, *Polling*, Retransmissão seletiva de tramas.

Abstract

The present thesis has as main goal the characterization of a new communications protocol called TPAP (TDMA Polling with Ack Piggyback) using ns-3 as the simulation tool.

The TPAP protocol was developed to offer a reliable solution with high performance in networks based on CSMA/CA access protocols (adopted in the IEEE 802.11 family of standards). It supports bidirectional communication and targets a point-to-multipoint topology, constituting a protocol layer usually called 2.5, since it is located between the MAC layer and the IP layer.

The TPAP protocol is based on selective retransmission of frames and combines centralized access control to the medium, based on polling, together with an error recovery mechanism of the sliding window type. The window is also a form of flow control.

Two versions were developed taking into account the features of the different variants of the IEEE 802.11 family. The complete version covers all aforementioned mechanisms, and has as main target the standards that do not support the aggregation of frames. A simpler version is intended for standards that support MAC level aggregation frames and their confirmation in blocks, which justifies the exclusion of the selective retransmission mechanism of frames of TPAP.

In addition to the framework and goals, and literature and state of the art review, the methodology adopted for the implementation of the two versions of the protocol in ns-3 is described, while at the end the simulation results are presented and analyzed.

Different test scenarios were considered, in order to compare the two versions of the protocol. To highlight its benefits, simulations without the TPAP protocol were also carried out in similar traffic conditions, which allowed emulating the execution of IP directly over the native mechanisms of the IEEE 802.11 standard, considering as alternatives the activation and inactivation of the optional RTS/CTS mechanism.

Key-Words: Block Acknowledgment, IEEE 802.11, ns-3, Polling, Selective Repeat.

Agradecimentos

Em primeiro lugar gostaria de deixar o meu mais sincero agradecimento ao Professor Doutor José Ruela por todo o apoio, disponibilidade e orientação que me proporcionou. Ao longo da dissertação todos os esclarecimentos e toda a partilha de conhecimento que me facultou foram essenciais para o sucesso desta dissertação, e posso afirmar que não poderia ter escolhido um melhor e mais dedicado orientador.

Em segundo lugar gostaria de deixar uma palavra de apreço ao Professor Doutor Mário Leitão, ao Doutor Rui Campos, ao Professor Doutor Manuel Ricardo e ao Mestre Filipe Ribeiro por todo o apoio e disponibilidade demonstrados durante a realização da dissertação.

Gostaria também de agradecer aos meus pais, António Brito e Isolina Brito, por todo o apoio que me deram até hoje; sempre estiveram ao meu lado nas horas menos boas e sempre tiveram uma palavra de amizade e um conselho para me dar, e tudo o que hoje sou devo-o a eles.

Por fim, mas não menos importante, gostaria de agradecer à minha namorada, Ana Moreira, por todo o apoio, conselhos e incentivo dados ao longo desta jornada de cinco anos que agora termina.

Obrigado a todos!

Pedro Brito

*“Some men see things as they are and say,
‘Why?’ I dream of things that never
were and say, ‘Why not?’”*

- George Bernard Shaw

Índice

RESUMO	III
ABSTRACT	V
AGRADECIMENTOS.....	VII
LISTA DE FIGURAS.....	XIV
LISTA DE TABELAS.....	XVII
ABREVIATURAS E SÍMBOLOS.....	XVIII
CAPÍTULO 1	1
INTRODUÇÃO	1
1.1 Contexto e motivação	1
1.2 Objetivos.....	2
1.3 Organização do documento	3
CAPÍTULO 2	4
REVISÃO DA LITERATURA E ESTADO DA ARTE	4
2.1 Introdução ao standard IEEE 802.11	5
2.2 Modos de operação e tipos de redes IEEE 802.11	6
2.3 Camada MAC.....	7
2.3.1 CSMA/CA	8
2.3.2 RTS/CTS	8
2.3.3 Point Coordination Function (PCF)	9
2.4 Camada PHY.....	10
2.5 Evolução do standard IEEE 802.11	10
2.5.1 Standards de Nova Geração	11
2.5.1.1 Agregação de tramas.....	11
2.5.1.2 Confirmação de tramas em bloco	13
2.6 Protocolos do Tipo Polling	15

2.6.1	<i>Interleaved Polling with Adaptive Cycle Time (IPACT)</i>	15
2.6.2	<i>Dynamic credit distribution (D-CRED)</i>	17
2.7	<i>Protocolos do Tipo Selective Repeat</i>	18
2.7.1	<i>“Highly Efficient Selective Acknowledgement (SACK) for TCP”</i>	18
2.7.2	<i>Compressed Block Ack em redes IEEE 802.11</i>	18
2.8	<i>Ferramentas de Simulação</i>	20
2.8.1	<i>OPNET (Optimized Network Engineering Tool)</i>	20
2.8.2	<i>QualNet (QualNet Developer)</i>	21
2.8.3	<i>GloMoSiM (Global Mobile Information System Simulator)</i>	21
2.8.4	<i>OMNeT++ (Objective Modular Network Testbed in C++)</i>	21
2.8.5	<i>ns-2 (network simulator-2)</i>	22
2.8.6	<i>ns-3 (network simulator-3)</i>	22
2.8.6.1	<i>Módulo Wi-Fi existente no ns-3</i>	23
		24
		24
CAPÍTULO 3		26
PROTOCOLO TPAP		26
3.1	<i>Considerações Preliminares</i>	26
3.2	<i>Funcionamento do protocolo TPAP</i>	28
3.3	<i>Modelo Teórico</i>	31
3.3.1	<i>Modelo básico para TPAP completo, sem agregação e sem erros</i>	31
3.3.2	<i>Extensão do modelo básico para TPAP completo, sem agregação, mas considerando erros</i>	34
3.3.3	<i>Modelo para TPAP simples, considerando agregação na camada MAC</i>	35
CAPÍTULO 4		37
METODOLOGIA		37
4.1	<i>Implementação do protocolo TPAP no ns-3</i>	37
4.1.1	<i>Alterações efetuadas ao módulo WifiNetDevice</i>	38
4.1.1.1	<i>Métodos para controlo do acesso ao meio (implementação do mecanismo de polling)</i>	40
4.1.1.2	<i>Métodos para controlo de erros (implementação do Selective Repeat)</i>	40
4.1.1.3	<i>Métodos para envio e receção de tramas de dados</i>	42
4.1.1.4	<i>Métodos auxiliares</i>	43
4.1.1.5	<i>Extração de dados do WifiTPAP</i>	44
4.1.1.6	<i>Criação de Erros</i>	44
4.1.1.7	<i>Protocolo TPAP com mecanismo de agregação e BACK de nível MAC</i>	45
4.1.1.8	<i>Ficheiro de Testes</i>	45
CAPÍTULO 5		47

SIMULAÇÃO DE CENÁRIOS DE TESTE E ANÁLISE DE RESULTADOS	47
5.1 <i>Resultados obtidos com o standard IEEE 802.11a</i>	49
5.1.1 TPAP completo com parâmetros MAC <i>default</i>	49
5.1.1.1 Validação do modelo teórico	51
5.1.2 TPAP completo com eliminação de <i>gaps</i> no nível MAC	53
5.1.2.1 Validação do modelo teórico	54
5.1.3 TPAP completo com variação do tamanho dos pacotes (<i>P</i>) e do tamanho da janela (<i>W</i>)	56
5.1.4 Tráfego Assimétrico	58
5.1.4.1 Validação do modelo teórico	59
5.1.5 Introdução de atrasos de processamento	60
5.1.5.1 Validação do modelo teórico	62
5.1.6 Introdução de erros	63
5.1.6.1 Validação do modelo teórico	66
5.1.7 Outros modelos de tráfego (débito não constante).....	67
5.1.8 Resultados obtidos sem o protocolo TPAP	68
5.1.9 Conclusões	70
5.2 <i>Resultados obtidos com o standard IEEE 802.11n, com agregação de nível MAC inibida</i>	71
5.2.1 TPAP completo com eliminação de <i>gaps</i> no nível MAC	71
5.2.1.1 Validação do modelo teórico	72
5.2.2 Resultados obtidos sem o protocolo TPAP	74
5.2.3 Conclusões	75
5.3 <i>Resultados obtidos com o standard IEEE 802.11n, com agregação e BACK de nível MAC</i>	77
5.3.1 TPAP simples	77
5.3.1.1 Validação do modelo teórico	79
5.3.2 Resultados obtidos sem o protocolo TPAP	82
5.3.3 Conclusões	83
CAPÍTULO 6	84
CONCLUSÕES E TRABALHO FUTURO.....	84
ANEXO A	87
REFERÊNCIAS	118

Lista de figuras

Figura 2.1 - Modelo TCP/IP evidenciando as camadas definidas pelo <i>standard</i> IEEE 802.11	5
Figura 2.2 - Interoperabilidade entre equipamentos usando <i>standards</i> diferentes [2]	6
Figura 2.3 - Modos de operação e respectivos tipos de redes suportadas no <i>standard</i> IEEE 802.11 [2].....	7
Figura 2.4 - Arquitetura MAC [3]	7
Figura 2.5 - Ilustração do problema dos “nós escondidos”	9
Figura 2.6 - Evolução do <i>standard</i> IEEE 802.11 [5]	10
Figura 2.7 - Mecanismo de agregação de tramas em dois níveis [8]	13
Figura 2.8 - Etapas do Algoritmo IPACT [10]	16
Figura 2.9 - Etapas do algoritmo D-DRED [11]	17
Figura 2.10 - Ilustração da organização em módulos do simulador ns-3 [20]	23
Figura 2.11 - Arquitetura do módulo <i>Wi-Fi</i> existente no ns-3 [1]	24
Figura 3.1 - Modelo TCP/IP com protocolo TPAP.....	26
Figura 3.2 - Configuração ponto-multiponto.....	27
Figura 3.3 - Exemplo ilustrativo do <i>bitmap</i>	29
Figura 3.4 - Sequência de mensagens trocadas entre a EB e cada uma das ECs	30
Figura 4.1 - Módulo <i>Wi-Fi</i> do ns-3 antes e depois da implementação do protocolo TPAP	38
Figura 4.2 - Interligação dos métodos criados para a implementação do protocolo TPAP	39
Figura 4.3 - Atualização do <i>bitmap</i> recebido	41
Figura 4.4 - Dados extraídos do módulo <i>WifiTPAP</i>	44
Figura 5.1 - Variação da ocupação da janela com o aumento do débito por fluxo	50
Figura 5.2 - Variação do tempo de ciclo com o aumento do débito por fluxo	50

Figura 5.3 - Variação do <i>goodput</i> total com o aumento do débito por fluxo.....	50
Figura 5.4 - Figura obtida através do método de regressão linear, considerando o cenário com o <i>standard</i> IEEE 802.11a, com <i>gaps</i>	52
Figura 5.5 - Comparação dos tempos de ciclo para os cenários com e sem <i>gaps</i>	53
Figura 5.6 - Comparação da ocupação da janela para os cenários com e sem <i>gaps</i>	54
Figura 5.7 - Comparação do <i>goodput</i> total máximo obtido para os cenários com e sem <i>gaps</i>	54
Figura 5.8 - Figura obtida através do método de regressão linear, considerando o cenário com o <i>standard</i> IEEE 802.11a, sem <i>gaps</i>	55
Figura 5.9 - Variação do <i>goodput</i> máximo com a variação do tamanho dos pacotes e da janela.....	56
Figura 5.10 - Variação do tempo de ciclo máximo com a variação do tamanho dos pacotes e da janela	57
Figura 5.11 - Variação do tempo de ciclo considerando tamanho de pacotes diferentes, com $W = 16$	57
Figura 5.12 - Variação do <i>goodput</i> total máximo com o aumento do atraso de processamento introduzido.....	61
Figura 5.13 - Variação do tempo de ciclo com o aumento do atraso de processamento introduzido	61
Figura 5.14 - Variação da ocupação da janela com o aumento do atraso de processamento introduzido	61
Figura 5.15 - Variação da ocupação da janela com a variação na taxa de erros	64
Figura 5.16 - Variação do tempo de ciclo com a variação na taxa de erros.....	64
Figura 5.17 - Variação do <i>goodput</i> total máximo com a variação na taxa de erros.....	64
Figura 5.18 - Variação do <i>goodput</i> total com o mecanismo nativo do <i>standard</i> IEEE 802.11, .	69
Figura 5.19 - Variação do <i>goodput</i> total com o mecanismo nativo do <i>standard</i> IEEE 802.11, com RTS/CTS ativo.....	69
Figura 5.20 - Variação da ocupação da janela considerando o TPAP completo e os <i>standards</i> IEEE 802.11a e IEEE 802.11n.....	71
Figura 5.21 - Variação do tempo de ciclo considerando o TPAP completo e os <i>standards</i> IEEE 802.11a e IEEE 802.11n	72
Figura 5.22 - <i>Goodput</i> total máximo obtido considerando o protocolo TPAP completo e os <i>standards</i> IEEE 802.11a e IEEE 802.11n.....	72
Figura 5.23 - Figura obtida através do método de regressão linear, considerando o cenário com TPAP completo e <i>standard</i> IEEE 802.11n, com eliminação dos <i>gaps</i> no nível MAC ..	73
Figura 5.24 - Variação do <i>goodput</i> total com o mecanismo nativo do <i>standard</i> IEEE 802.11, com RTS/CTS inativo	74

Figura 5.25 - Variação do <i>goodput</i> total com o mecanismo nativo do <i>standard</i> IEEE 802.11, com RTS/CTS ativo.....	75
Figura 5.26 - Variação do tempo de ciclo com o aumento do débito por fluxo, considerando as diferentes versões do protocolo TPAP com o <i>standard</i> IEEE 802.11n ...	77
Figura 5.27 - Variação da ocupação da janela com o aumento do débito por fluxo, considerando as diferentes versões do protocolo TPAP com o <i>standard</i> IEEE 802.11n ...	78
Figura 5.28 - Variação do <i>goodput</i> total máximo, considerando as diferentes versões do protocolo TPAP com o <i>standard</i> IEEE 802.11n	78
Figura 5.29 - Relação do tempo de ciclo com o aumento de n , considerando o TPAP simples com <i>standard</i> IEEE 802.11n	80
Figura 5.30 - Figura obtida através do método de regressão linear, considerando o cenário com <i>standard</i> IEEE 802.11n, com protocolo TPAP simples (primeira zona de $T(n)$).....	80
Figura 5.31 - Figura obtida através do método de regressão linear, considerando o cenário com <i>standard</i> IEEE 802.11n, com protocolo TPAP simples (terceira zona de $T(n)$)	81
Figura 5.32 - Variação do <i>goodput</i> total com o mecanismo nativo do <i>standard</i> IEEE 802.11, com RTS/CTS inativo	82

Lista de tabelas

Tabela 5.1 - Valor dos parâmetros obtidos para o modelo teórico considerando o TPAP completo e <i>standard</i> IEEE 802.11a, com parâmetros MAC <i>default</i>	53
Tabela 5.2 - Valor dos parâmetros obtidos para o modelo teórico considerando o TPAP completo e <i>standard</i> IEEE 802.11a, com eliminação de <i>gaps</i> no nível MAC	55
Tabela 5.3 - Valores dos parâmetros do modelo teórico considerando tráfego assimétrico ...	60
Tabela 5.4 - Valores dos parâmetros do modelo teórico considerando atrasos de processamento.....	62
Tabela 5.5 - Valores dos parâmetros do modelo teórico considerando taxa de erros para tramas de dados de 1%	66
Tabela 5.6 - Valores dos parâmetros do modelo teórico considerando taxa de erros para tramas de dados de 10%.....	67
Tabela 5.7 - Valores dos parâmetros do modelo teórico considerando o protocolo TPAP completo e os <i>standards</i> IEEE 802.11a e IEEE 802.11n	73

Abreviaturas e Símbolos

Lista de abreviaturas

A-MPDU	<i>Aggregated-MAC Protocol Data Unit</i>
A-MSDU	<i>Aggregated-MAC Service Data Unit</i>
AP	<i>Access Point</i>
ARP	<i>Address Resolution Protocol</i>
BER	<i>Bit Error Ratio</i>
BSS	<i>Basic Service Set</i>
CFP	<i>Contention Free Period</i>
CP	<i>Contention Period</i>
CSMA/CA	<i>Carrier Sense Multiple Access with Collision Avoidance</i>
DBA	<i>Dynamic Bandwidth Allocation</i>
DCF	<i>Distributed Coordination Function</i>
D-CRED	<i>Dynamic Credit Distribution</i>
DIFS	<i>DCF Inter-Frame Spacing</i>
DSSS	<i>Direct Sequence Spread Spectrum</i>
EPON	<i>Ethernet Passive Optical Network</i>
ESS	<i>Extended Service Set</i>
FHSS	<i>Frequency Hopping Spread Spectrum</i>
GloMoSiM	<i>Global Mobile Information System Simulator</i>
GUI	<i>Graphical User Interface</i>
IBSS	<i>Independent Basic Service Set</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IP	<i>Internet Protocol</i>
IPACT	<i>Interleaved Polling with Adaptive Cycle Time</i>
ISM	<i>Industrial, Scientific and Medical</i>
LAN	<i>Local Area Network</i>
LLC	<i>Logical Link Control</i>

MAC	<i>Medium Access Control</i>
NAM	<i>Network Animator</i>
NAV	<i>Network Allocation Vector</i>
ns-2	<i>network simulator-2</i>
ns-3	<i>network simulator-3</i>
OLT	<i>Optical Line Termination</i>
OMNeT++	<i>Objective Modular Network Testbed in C++</i>
OPNET	<i>Optimized Network Engineering Tool</i>
ONU	<i>Optical Network Unit</i>
PCF	<i>Point Coordination Function</i>
PIFS	<i>PCF Inter-Frame Spacing</i>
PON	<i>Passive Optical Network</i>
QoS	<i>Quality of Service</i>
RIFS	<i>Reduced Inter-Frame Spacing</i>
RTS/CTS	<i>Request To Send/Clear To Send</i>
RTT	<i>Round-Trip Time</i>
SIFS	<i>Short Inter-Frame Spacing</i>
SINR	<i>Signal to Interference plus Noise Ratio</i>
TCP	<i>Transmission Control Protocol</i>
TDD	<i>Time Division Duplex</i>
TDMA	<i>Time Division Multiple Access</i>
TID	<i>Traffic Identifier</i>
TPAP	<i>TDMA Polling with Ack Piggyback</i>
VHT	<i>Very High Throughput</i>
VoIP	<i>Voice over Internet Protocol</i>
Wi-Fi	<i>Wireless Fidelity</i>
WLAN	<i>Wireless Local Area Network</i>

Capítulo 1

Introdução

Neste capítulo será apresentada a contextualização e motivação do problema a tratar na presente dissertação, juntamente com a enumeração do conjunto de objetivos do trabalho proposto e desenvolvido.

Por fim será apresentada a estrutura deste documento, referindo de forma sucinta os assuntos tratados em cada capítulo.

1.1 Contexto e motivação

A presente dissertação surge enquadrada num projeto desenvolvido pelo INESC TEC denominado “*HiperWireless* - Comunicações micro ondas ponto-multiponto na banda *hiperlan* livre (17 GHz)”. Do ponto de vista arquitetónico a solução alvo deveria suportar comunicação bidirecional entre uma Estação Base (EB) e múltiplas Estações Clientes (ECs), numa configuração ponto-multiponto, usando o protocolo IP (*Internet Protocol*) na camada de rede e um protocolo de acesso ao meio baseado no *standard* IEEE 802.11. Como principais objetivos pretendia-se garantir comunicação fiável, robusta e com elevado desempenho.

No âmbito do projeto foi proposto e desenvolvido um novo protocolo designado TPAP (*TDMA Polling with Ack Piggyback*), que foi igualmente caracterizado por meio de um modelo teórico ideal, com o objetivo de, por um lado, derivar resultados aproximados e, por outro, identificar fatores críticos de desempenho com vista à sua otimização.

O protocolo TPAP constitui uma camada protocolar normalmente designada 2.5, uma vez que se situa entre a camada MAC (nível dois), e a camada IP (nível três). É baseado na retransmissão seletiva de tramas e combina controlo centralizado de acesso ao meio, baseado em *polling*, com um mecanismo de recuperação de erros do tipo janela deslizante. A janela constitui igualmente uma forma de controlo de fluxo.

Foram desenvolvidas duas versões tendo em atenção as funcionalidades das diferentes variantes da família IEEE 802.11. A versão completa engloba todos os mecanismos supracitados, e tem como alvo principal os *standards* que não suportam agregação de tramas. Uma versão mais simples destina-se aos *standards* que suportam no nível MAC a agregação de tramas e a respetiva confirmação em blocos, o que justifica a exclusão do mecanismo de retransmissão seletiva de tramas do TPAP.

É neste contexto que se integra esta dissertação, que tem como objetivos genéricos validar o protocolo e o modelo teórico e avaliar o seu desempenho, usando uma ferramenta de simulação de redes, designada ns-3 (*network simulator-3*), tendo como base diferentes cenários de simulação.

Como principais pontos de motivação para este projeto, é de referir a integração numa equipa com uma vasta experiência na área de redes de comunicação, a participação num projeto em desenvolvimento e para o qual a dissertação poderia dar contribuições valiosas e por fim o desenvolvimento de competências numa ferramenta de simulação globalmente utilizada e que é hoje em dia uma referência relevante no desenvolvimento de novos protocolos, no aperfeiçoamento de protocolos existentes ou no planeamento de estruturas de comunicação com e sem fios.

1.2 Objetivos

Os objetivos desta dissertação, em termos gerais, passam pela implementação do protocolo TPAP no ns-3, pela sua caracterização considerando diferentes cenários de estudo, e pela validação do modelo teórico. Para tal, foram identificados objetivos específicos que correspondem a diferentes fases de desenvolvimento:

- Conceção de uma solução que permita a integração do protocolo TPAP no módulo *Wi-Fi* existente no ns-3;
- Implementação da camada correspondente ao protocolo TPAP no ns-3, nas suas duas variantes;
- Identificação e criação de cenários de simulação que permitam testar o protocolo, com a opção de variar parâmetros de simulação, quer os relativos às camadas físicas e MAC, quer os relativos ao protocolo TPAP;
- Extração de dados da simulação que permitam analisar e avaliar o desempenho do protocolo em condições reais e em condições ideais, fornecendo estas uma referência para uma possível otimização do protocolo, tendo em atenção parâmetros críticos;
- Extração de dados de simulação que permitam realizar uma comparação crítica entre as duas versões do protocolo TPAP e uma solução em que o protocolo TPAP é removido, correspondendo à execução de IP sobre os mecanismos nativos do *standard* IEEE 802.11.

1.3 Organização do documento

Para além do presente capítulo de Introdução, este documento é composto por cinco capítulos adicionais.

No segundo capítulo será realizada a revisão da literatura, com a apresentação do estado da arte, sendo revistos alguns conceitos teóricos essenciais para a compreensão do restante documento.

No terceiro capítulo será apresentado o protocolo TPAP, sendo explicadas e justificadas as diferenças entre as duas versões desenvolvidas, e de seguida evidenciadas as suas principais características e modo de funcionamento. No final do terceiro capítulo será apresentado o modelo teórico, a validar com base nos resultados de simulação.

No quarto capítulo será revista a metodologia adotada para a implementação do protocolo TPAP, em particular a forma como a camada correspondente foi integrada no ns-3.

No quinto capítulo serão apresentados os resultados obtidos após a integração do protocolo TPAP no ns-3. Será realizada uma análise dos mesmos, com vista à validação do modelo teórico, e serão comparadas as duas versões propostas do protocolo TPAP juntamente com a alternativa sem o protocolo TPAP.

Por fim, no sexto capítulo, serão apresentadas as conclusões a retirar do trabalho desenvolvido, bem como sugestões de trabalho futuro.

Capítulo 2

Revisão da Literatura e Estado da Arte

Pretende-se neste segundo capítulo fazer o enquadramento do trabalho desenvolvido no contexto de soluções existentes (arquiteturas e tecnologias de comunicação).

Para tal, será realizada inicialmente uma introdução ao *standard* IEEE 802.11, que tem como âmbito de aplicação redes locais sem fios - WLAN (*Wireless Local Area Network*). Serão referidos em primeiro lugar os modos de funcionamento (*ad-hoc* e infraestrutura) definidos no *standard* e a sua relação com a arquitetura de referência.

De seguida serão abordadas algumas características da camada MAC normalizadas no *standard* IEEE 802.11, nomeadamente os mecanismos propostos para controlo de acesso ao meio, baseados em CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*), e após a análise da camada MAC será também realizada uma breve apresentação das características da camada PHY.

Uma vez que a família de *standards* IEEE 802.11 integra hoje múltiplas variantes, com diferentes âmbitos de aplicação, será apresentada a sua evolução, com destaque para as soluções concebidas para as redes de nova geração, referindo e justificando os principais mecanismos propostos, como a agregação de tramas e a confirmação em bloco das mesmas.

Após a introdução ao *standard* IEEE 802.11 e às suas características mais relevantes, serão apresentados exemplos de alguns protocolos do tipo *Selective Repeat* e do tipo *Polling*, dois elementos que constituem a base do protocolo TPAP, objeto de análise e avaliação no âmbito deste trabalho.

Por fim será realizada uma breve apresentação de algumas ferramentas de simulação disponíveis no mercado, evidenciando algumas das suas principais características e justificando a escolha do ns-3 como ferramenta de simulação para a presente dissertação.

2.1 Introdução ao *standard* IEEE 802.11

A existência de *standards* tem como principal objetivo contribuir para a interoperabilidade entre soluções desenvolvidas por diferentes empresas, uma vez que estas respeitarão os mesmos princípios definidos pelo *standard*.

O *standard* IEEE 802.11 foi o primeiro a ser desenvolvido para WLANs, sendo globalmente aceite, e constituindo a base das redes que correntemente designamos por redes *Wi-Fi* (*Wireless Fidelity*).

O *standard* IEEE 802.11 adota o mesmo modelo arquitetónico das LANs (*Local Area Networks*) IEEE 802.x, ou seja, especifica uma camada física (PHY) e uma camada de controlo de acesso ao meio (MAC) adequada às necessidades das comunicações sem fios e que oferece a mesma interface às camadas superiores, de forma a manter a interoperabilidade [2]. Estas camadas são ilustradas na Figura 2.1 no contexto do modelo TCP/IP.

Em redes locais sem fios são necessários mecanismos que permitam a partilha de múltiplas redes na mesma área, mecanismos para gerir interferências, quer entre estações que transmitem simultaneamente no mesmo canal, originando colisões, quer entre canais de comunicação adjacentes, que se manifestam como ruído, mecanismos para gerir o problema dos “nós escondidos”, bem como mecanismos para salvaguardar a privacidade dos dados e o controlo no acesso ao meio.

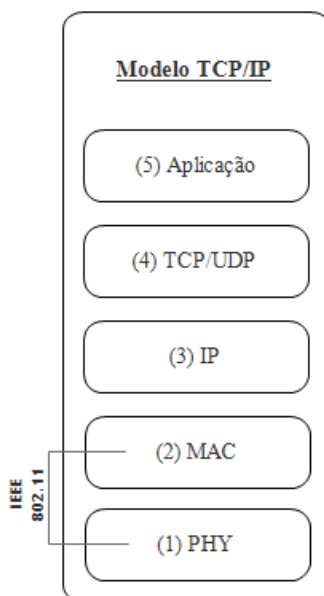


Figura 2.1 - Modelo TCP/IP evidenciando as camadas definidas pelo *standard* IEEE 802.11

A Figura 2.2 ilustra a possibilidade de comunicação entre estações numa WLAN IEEE 802.11 e numa LAN IEEE 802.3/*Ethernet*, ou seja, a interoperabilidade entre equipamentos usando *standards* diferentes, em que as camadas MAC e PHY são específicas e a camada LLC (*Logical Link Control*) oferece uma interface comum para as camadas superiores.

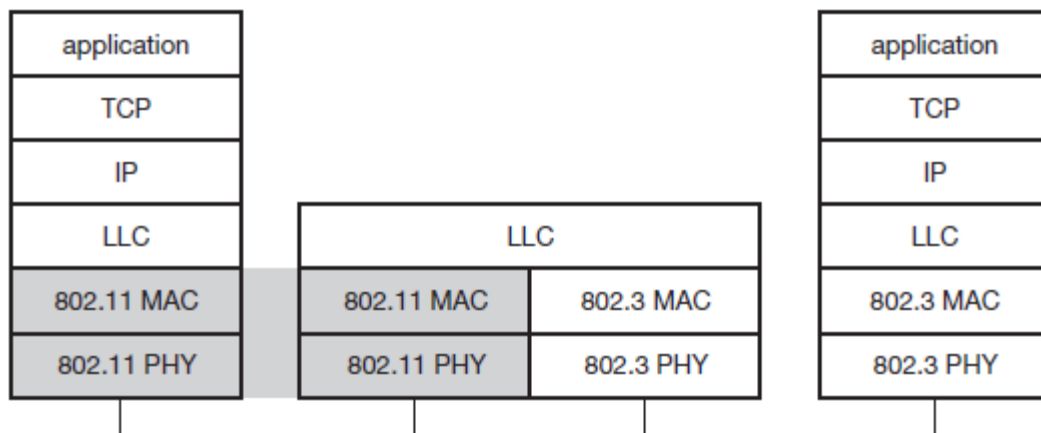


Figura 2.2 - Interoperabilidade entre equipamentos usando *standards* diferentes [2]

2.2 Modos de operação e tipos de redes IEEE 802.11

Existem essencialmente dois modos de operação numa WLAN IEEE 802.11, a que correspondem dois tipos de redes - infraestruturada e *ad-hoc*.

Numa rede infraestruturada existem dois tipos de componentes, o *Access Point* (AP) e os nós sem fios. O AP desempenha funções de reenvio de tramas (de um nó de origem para um ou mais nós de destino), controlo de acesso ao meio e acesso a outras redes. Toda a comunicação ao nível físico é realizada tipicamente entre os nós sem fios e o AP, mas não diretamente entre os nós [2].

Numa rede *ad-hoc* todos os nós podem comunicar diretamente entre si, desde que estejam ao alcance rádio, ou desde que outro nó possa reenviar as tramas recebidas. Este modo de operação requer maior complexidade em cada nó, uma vez que cada nó necessita de implementar mecanismos de controlo de acesso ao meio e mecanismos para gerir o problema dos “nós escondidos” [2].

No *standard* IEEE 802.11, os nós sem fios referidos anteriormente são designados por estações (STA), e comunicam diretamente com o AP no modo infraestruturado. As estações e o AP dentro do mesmo raio de cobertura rádio formam um BSS (*Basic Service Set*).

Diferentes BSS podem ser ligados através do *distribution system*, que liga vários APs, formando o ESS (*Extended Service Set*). Outro elemento apresentado na Figura 2.3 é o portal, que tem como objetivo a integração das redes IEEE 802.11 com as redes cabçadas tradicionais (*wired networks*) [2, 3].

As redes *ad-hoc*, de acordo com o *standard* IEEE 802.11, constituem-se pela ligação entre estações que usam a mesma frequência rádio, formando uma *independent BSS* (IBSS) [2].

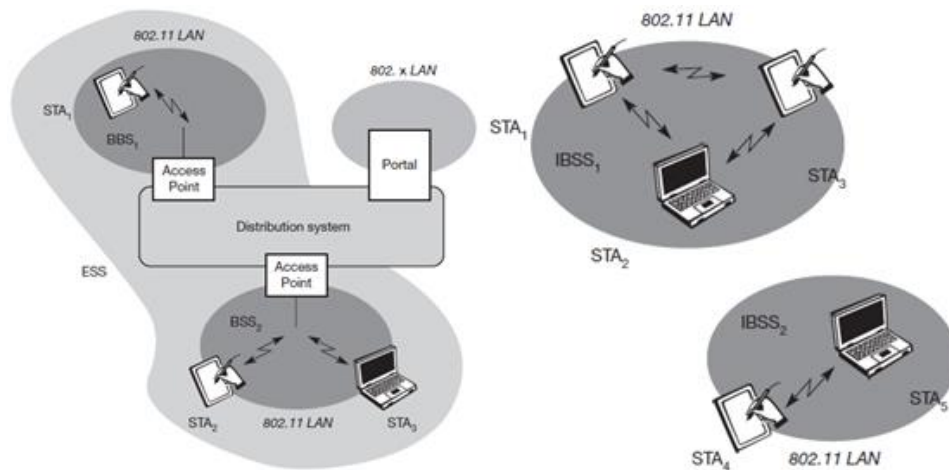


Figura 2.3 - Modos de operação e respectivos tipos de redes suportadas no *standard* IEEE 802.11 [2]

A Figura 2.3 ilustra os dois modos de operação e respectivos tipos de redes, à esquerda o modo infraestruturado e à direita o modo *ad-hoc*.

2.3 Camada MAC

A camada MAC (*Medium Access Control*) é responsável por diversas tarefas; no entanto, e como o próprio nome indica, a principal é o controlo do acesso ao meio.

Três mecanismos foram desenvolvidos com este propósito: o CSMA/CA, mecanismo básico e obrigatório, o RTS/CTS (*Request To Send / Clear To Send*), um método para evitar o problema dos “nós escondidos” e que constitui um mecanismo opcional, e por fim um método de contenção baseado em *Polling*, que não é suportado na maioria das cartas de rede. Os dois primeiros fazem parte da *distributed coordination function* (DCF) e o terceiro da *point coordination function* (PCF) [2].

O DCF constitui a plataforma básica sobre a qual se constroem funções adicionais, entre as quais o PCF, como representado na Figura 2.4.

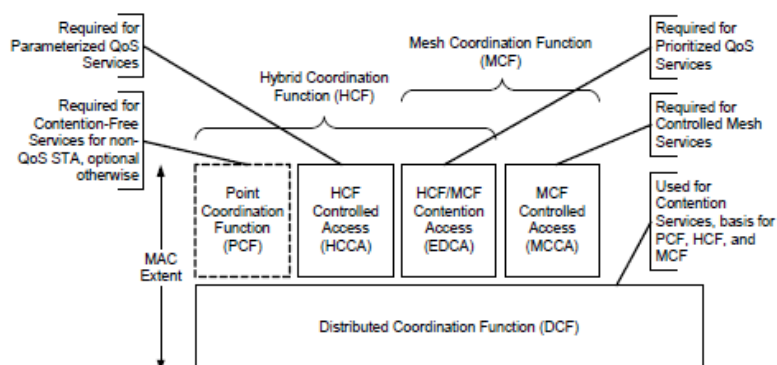


Figura 2.4 - Arquitetura MAC [3]

Todos os métodos de controlo de acesso usam parâmetros para controlar o tempo de espera, designados intervalos de guarda, que podem ser usados para definir prioridades no acesso ao meio. Têm valores variáveis conforme a camada PHY usada e são definidos em relação ao *slot time*, que é derivado tendo em conta parâmetros como o atraso de propagação médio [2].

Estes tempos são designados por SIFS (*Short inter-frame spacing*), PIFS (*PCF inter-frame spacing*) e DIFS (*DCF inter-frame spacing*), sendo $DIFS > PIFS > SIFS$, o que quer dizer que SIFS, por ter o menor valor, representa o intervalo de guarda que antecede as mensagens com maior prioridade, normalmente curtas, como por exemplo as confirmações (*acknowledgments*) [2].

2.3.1 CSMA/CA

O mecanismo CSMA-CA tem como objetivo permitir que múltiplos dispositivos acedam ao mesmo canal de frequência, evitando colisões que ocorrem num nó quando dois ou mais nós transmitem ao mesmo tempo e estão no raio de alcance do recetor. Como o meio de comunicação é partilhado, o envio simultâneo de pacotes irá provocar colisões e consequentemente erros nos pacotes no recetor alvo.

O mecanismo CSMA-CA funciona da seguinte forma [2, 3]:

- I. Uma estação que tenha um pacote para enviar monitoriza o canal de transmissão:
Se o meio estiver livre e permanecer livre durante um período igual a DIFS, acede ao meio; se entretanto o meio ficar ocupado, procede como se estivesse ocupado; Se estiver ocupado, é selecionado um intervalo aleatório de *backoff*, é iniciado um contador com esse valor e a estação espera que o meio fique livre durante um período igual a DIFS. O contador é então decrementado, sendo parado se uma transmissão for detetada antes de atingir o valor zero e reativado quando o canal ficar livre outra vez durante um intervalo de tempo igual a DIFS. A estação transmite quando o contador atingir o valor zero;
- II. Para evitar captura de canal, uma estação espera um tempo aleatório de *backoff* entre duas transmissões consecutivas, mesmo que o meio esteja vazio.

2.3.2 RTS/CTS

Este mecanismo tem como objetivo eliminar o problema dos “nós escondidos”, que surge em situações em que os nós estão distribuídos como ilustrado na Figura 2.5.

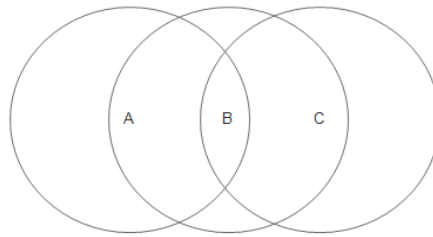


Figura 2.5 - Ilustração do problema dos “nós escondidos”

Nesta figura observamos que o raio de transmissão de A alcança B mas não C, que o raio de alcance de C inclui B mas não A e por fim que o raio de alcance de B inclui A e C, donde se conclui que nem A deteta C nem C deteta A.

O problema surge quando A inicia uma transmissão para B, C não deteta esta transmissão e ao querer transmitir para B monitoriza o meio e este aparece como livre, falhando o mecanismo de CSMA. C inicia a sua transmissão o que leva a uma colisão em B, que A não deteta, continuando com a sua transmissão, ou seja, A é um nó escondido para C, e C é um nó escondido para A.

Para contornar este problema foi proposto um mecanismo adicional que utiliza duas tramas de controlo, RTS e CTS.

O mecanismo funciona da seguinte forma [2, 3]:

- I. A estação que pretende transmitir um pacote, espera DIFS e envia uma trama RTS com um parâmetro de reserva de tempo de ligação;
- II. O recetor envia a trama CTS após esperar SIFS, contendo também o tempo de reserva da transmissão;
- III. A estação emissora de RTS após receber CTS espera SIFS e envia a trama de dados. Após o recetor receber a trama de dados espera SIFS e envia a trama ACK;
- IV. As outras estações ao receberem as tramas RTS e/ou CTS marcam o chamado *net allocation vector* (NAV) com o tempo reservado para esta ligação, especificando o momento em que a estação pode tentar aceder ao meio novamente, ou seja, este mecanismo leva a uma reserva do meio de transmissão para um único emissor.

2.3.3 Point Coordination Function (PCF)

Este mecanismo é implementado no AP, o que exclui o uso deste mecanismo em redes *ad-hoc*. O PCF permite a entrega de dados com tempo limitado através de comunicações síncronas e usando o mecanismo de *Polling* [3, 4].

Para o uso deste mecanismo, o AP divide o tempo de acesso em períodos designados *super frame periods*, que contêm o *contention free period* (CFP) e o *contention period* (CP). O CP pode ser usado tanto pelo PCF como pelo DCF.

Para iniciar a transmissão, assim que o meio de transmissão estiver livre, o AP tem de esperar PIFS antes de aceder ao meio, o que, visto PIFS ser inferior a DIFS, garante que nenhuma outra estação inicia a transmissão antes [2, 3].

O AP efetua a *polling* das outras estações, “perguntando” se estas têm alguma trama para enviar, ou seja, a ordem de transmissão é controlada pelo AP, não ocorrendo colisões.

Durante o uso do PCF as outras estações marcam o NAV, evitando que estas tentem aceder o meio [2, 3].

Se o uso do mecanismo de *polling* for distribuído uniformemente pelas estações, também a largura de banda é uniformemente distribuída, o que é semelhante a um sistema TDMA (*time division multiple access*) com controlo centralizado e estático, com uma transmissão *time division duplex* (TDD) [2, 3].

2.4 Camada PHY

O *standard* IEEE 802.11 suporta três camadas físicas diferentes, uma baseada em FHSS (*Frequency Hopping Spread Spectrum*), outra em DSSS (*Direct Sequence Spread Spectrum*), na banda ISM (*Industrial, Scientific and Medical radio band*) dos 2.4 GHz e uma camada física *infra red* com comprimento de onda entre 850 e 950 nm e que permite um alcance máximo de 10 m [2, 3].

2.5 Evolução do *standard* IEEE 802.11

A rápida evolução na área das comunicações sem fios e a constante necessidade de melhorar e adaptar o *standard* IEEE 802.11, de forma a cobrir maiores distâncias, a oferecer débitos mais elevados, bem como a suportar *QoS* (*Quality of Service*) e mobilidade dos terminais, levaram a que ao longo dos anos fossem criadas extensões e melhorias do mesmo. A evolução da família IEEE 802.11 está ilustrada na Figura 2.6.

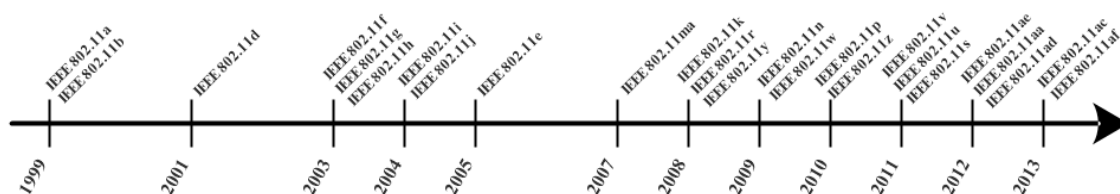


Figura 2.6 - Evolução do *standard* IEEE 802.11 [5]

2.5.1 *Standards* de Nova Geração

Com a introdução de novos serviços e aplicações, como televisão em alta definição, videoconferências, *streaming* multimédia, voz sobre IP (VoIP), transferência de ficheiros e jogos *online*, surgiu a necessidade por parte do IEEE de corresponder às exigências dos utilizadores, avançando para os *standards* de nova geração [6].

O problema das anteriores variantes do *standard* está relacionado com a baixa eficiência dos protocolos da camada MAC e da camada PHY, que se agrava com o aumento do débito binário no canal físico. Esta baixa eficiência deve-se ao elevado *overhead* introduzido, visto que a transmissão de uma única trama engloba, para além dos cabeçalhos MAC e PHY (este enviado com um débito binário mais baixo), um ciclo que começa com DIFS e o tempo de *backoff* que precedem a transmissão da trama, a espera de SIFS por parte do recetor antes do envio do ACK, juntamente com o tempo de propagação das tramas [6].

Com o objetivo de alcançar elevado *throughput*, requerido pelos serviços referidos foi desenvolvido em primeiro lugar o *standard* IEEE 802.11n, que pretendia alcançar o mínimo de 100Mbit/s de *throughput*, e mais tarde o *standard* IEEE 802.11ac, que constitui uma evolução do IEEE 802.11n e que foi projetado para o designado VHT (*very high throughput*), com o objetivo de alcançar *throughput* máximo próximo de 7 Gbit/s [6, 7].

No IEEE 802.11n foram propostas novas soluções tanto para o nível MAC, como para o nível PHY. No nível MAC as principais soluções passam pela introdução da agregação de tramas, bem como pela melhoria do mecanismo de *block acknowledgment*, introduzido no *standard* IEEE 802.11e, orientado para o suporte de Qualidade de Serviço.

2.5.1.1 Agregação de tramas

A agregação de tramas tem como principal objetivo reduzir o *overhead* introduzido no nível MAC e PHY, uma vez que o mecanismo de agregação combina múltiplos pacotes de dados do nível superior numa única trama (enquanto unidade de transmissão) de muito maior dimensão [8]. Esta trama maior pode ser vista como agregando sub-tramas MAC, existindo dois modos básicos de agregação a seguir referidos e um terceiro modo que os combina.

A redução do *overhead* deve-se ao facto de, em vez do uso de um cabeçalho PHY por cada trama, ser usado apenas um para a trama agregada, ao mesmo tempo que os intervalos de guarda entre tramas são eliminados.

Para a agregação de tramas foram propostos dois métodos, o A-MSDU (*Aggregated-MAC Service Data Unit*) e o A-MPDU (*Aggregated-MAC Protocol Data Unit*).

A diferença entre uma MSDU e uma MPDU é que uma MSDU corresponde a informação que é trocada a partir da subcamada superior MAC com os níveis superiores, enquanto uma MPDU corresponde a informação que é trocada pela subcamada inferior MAC com o nível PHY [6].

Numa A-MSDU, múltiplas MSDUs (sub-tramas) para o mesmo destino são agregadas numa única MPDU (neste caso existe um único cabeçalho MAC comum a todas as sub-tramas), o que aumenta a eficiência, principalmente para tramas pequenas, como por exemplo as tramas ACK do protocolo TCP (*Transmission Control Protocol*) [6].

Para que uma A-MSDU seja formada, a subcamada superior do nível MAC tem de colocar as MSDUs recebidas num *buffer* até que, ou o tamanho máximo da A-MSDU seja atingido, ou a MSDU mais antiga tenha ultrapassado um determinado tempo de espera [6].

Há algumas restrições para a agregação das MSDUs, nomeadamente todas as MSDUs têm de ter o mesmo valor de TID (*Traffic Identifier*), o tempo de vida da A-MSDU deve corresponder ao tempo máximo de vida dos elementos constituintes e, por fim, o endereço de destino e o endereço do remetente de cada sub-trama devem corresponder, respetivamente, ao mesmo endereço do emissor e do recetor no cabeçalho MAC.

A maior desvantagem deste mecanismo de agregação está relacionada com os canais de transmissão sujeitos a erros; uma vez que não existe proteção individual para cada sub-trama, mas apenas para a A-MSDU, a ocorrência de erros obriga a retransmitir toda a A-MSDU mesmo que apenas uma sub-trama tenha sido corrompida [6].

Uma A-MPDU consiste na agregação de múltiplas MPDUs, todas com o mesmo endereço de destino, à qual é atribuída um único cabeçalho PHY [6].

Ao contrário da A-MSDU, para se formar uma A-MPDU, não há tempo de espera, ou seja, o número de MPDUs agregadas depende do número de MPDUs que se encontrarem na fila de transmissão [6].

As duas técnicas anteriormente referidas podem ser combinadas, levando a uma agregação em dois níveis. No primeiro nível desde que as MSDUs respeitem as condições referidas, estas podem ser agregadas numa A-MSDU, passando então ao segundo nível. Uma MSDU que possua um TID diferente das restantes MSDUs não é agregada numa A-MSDU e é passada diretamente ao segundo nível, onde é agregada conjuntamente com A-MSDUs, levando à criação de uma A-MPDU. De referir que o tamanho máximo de cada MPDU agregada tem limite máximo de 4095 bytes, pelo que quer A-MSDUs quer MSDUs que ultrapassem esse limite não serão agregadas, bem como fragmentos, quer de uma quer de outra [6].

O mecanismo de agregação em dois níveis está ilustrado na Figura 2.7.

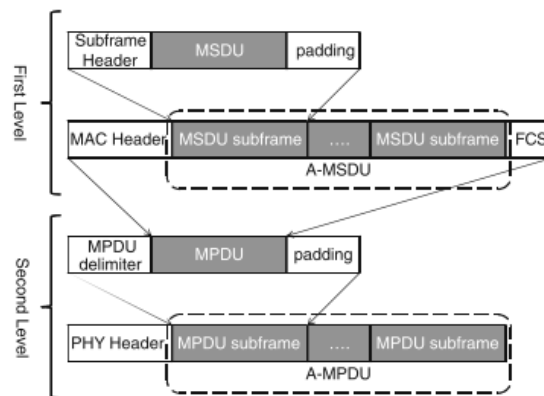


Figura 2.7 - Mecanismo de agregação de tramas em dois níveis [8]

O campo de *padding* ilustrado na Figura 2.7 é usado para garantir que quer o tamanho das MSDUs quer o tamanho das MPDUs é múltiplo de quatro bytes. O MPDU *delimiter* é usado para separar MPDUs dentro de uma A-MPDU [8].

Com a introdução do *standard* IEEE 802.11ac foram realizadas melhorias no nível MAC com o aumento do tamanho das A-MSDUs e das A-MPDUs, respetivamente, de 3839 ou 7935 bytes para 11454 bytes e de 65535 bytes para 1 Mbyte. Este aumento deve-se à necessidade de usar os dois tipos de agregação para obter melhor eficiência e conseguir alcançar um maior *throughput* [6].

Segundo [9] o uso combinado dos dois tipos de agregação permite um aumento de eficiência de 75% quando comparado com os 40% obtidos usando apenas A-MPDU, resultados obtidos considerando o aumento referido no tamanho das A-MSDUs e das A-MPDUs [9].

2.5.1.2 Confirmação de tramas em bloco

O mecanismo de confirmação de tramas em bloco sofreu alterações no *standard* IEEE 802.11n relativamente ao mecanismo proposto no *standard* IEEE 802.11e, de forma a ser utilizado em conjunto com o mecanismo de agregação.

Embora o mecanismo de agregação permita a redução do *overhead*, leva, por outro lado, a um aumento da probabilidade de erros, no caso das A-MSDUs, devido à inexistência de proteção individual das sub-tramas. Quanto maior forem as sub-tramas, maior será a A-MSDU, e dado que tramas grandes num ambiente com elevado BER (*Bit Error Ratio*) têm maior probabilidade de ocorrência de erros, haverá a necessidade de retransmissões [8].

Para atenuar este problema, o *standard* IEEE 802.11n propôs o uso do mecanismo *block acknowledgment* modificado para suportar múltiplos MPDUs numa A-MPDU, e cuja base de funcionamento é a seguinte:

- I. A estação emissora envia a A-MPDU e a estação recetora analisa as diferentes MPDUs contidas na A-MPDU, enviando de seguida uma trama *Block ACK* que assinala apenas as MPDUs corretamente recebidas;
- II. A estação emissora ao receber a trama *Block ACK*, apenas retransmitirá as MPDUs que não foram confirmadas positivamente [8].

Este mecanismo apenas é aplicado às A-MPDUs, não sendo aplicado às A-MSDUs, pelo que, e como já referido, sempre que uma MSDU está incorreta, toda a A-MSDU tem de ser retransmitida, pois é impossível identificar a(s) sub-trama(s) errada(s) [8].

O número máximo de MPDUs numa A-MPDU está limitado a 64, dado que o tamanho do *bitmap* do *Block ACK* apenas pode confirmar 64 MPDUs [8].

2.6 Protocolos do Tipo *Polling*

Serão apresentados nesta secção alguns exemplos de protocolos do tipo *Polling*, referindo os objetivos, as suas principais características e por fim as conclusões retiradas de simulações desses protocolos disponíveis na literatura.

2.6.1 *Interleaved Polling with Adaptive Cycle Time (IPACT)*

Uma PON (*Passive Optical Network*) é uma rede de acesso (*last mile*) que utiliza fibra ótica e componentes óticos e que é partilhada por múltiplos utilizadores, numa configuração ponto-multiponto. As transmissões numa PON são realizadas entre ONUs (*Optical Network Unit*) e uma OLT (*Optical Line Termination*) [10].

Dada a topologia ponto-multiponto, na direção *upstream*, ou seja, das ONUs para a OLT, as ONUs têm de partilhar os recursos do canal usando, por exemplo, o mecanismo de TDMA, que segundo [10] é o método preferido para essa partilha, dado que permite usar um único comprimento de onda, levando a uma solução rentável.

Contudo, e dado que o TDMA pode desperdiçar largura de banda se *slots* temporais atribuídos de forma estática a uma ONU não forem utilizados, um mecanismo de atribuição dinâmica de *slots* temporais permite que a largura de banda não usada por uma ONU possa ser aproveitada por outras ONUs [10].

De forma a possibilitar partilha eficiente de recursos (não possível com atribuição estática), e dado que em PONs é possível implementar no nó central (OLT) um mecanismo de atribuição dinâmica de períodos (*slots*) variáveis de transmissão às ONUs em função das suas necessidades em cada momento, foi proposto o algoritmo de *polling* intercalado designado IPACT (*Interleaved Polling with Adaptive Cycle Time*). Para evitar sobreposição de transmissões e desperdício de tempo entre transmissões, o sinal de *polling* é enviado para a próxima ONU no instante apropriado, antes da comunicação com a ONU anterior estar terminada [10]; tal é possível pois a transmissão nos dois sentidos é suportada em fibras diferentes ou em comprimentos de onda diferentes na mesma fibra.

O algoritmo está ilustrado na Figura 2.8, e é descrito da seguinte forma:

- I. Na Figura 2.8 a), assume-se que a OLT conhece o número total de bytes na fila de cada ONU bem como o RTT (*Round-Trip Time*) para cada ONU, sendo estes dados guardados na *polling table*. A OLT envia uma trama de controlo para a ONU1 autorizando-a a transmitir 6000 bytes;
- II. Após receber a trama de controlo a ONU1 inicia a sua transmissão até ao limite permitido pela OLT, neste caso 6000 bytes. Enquanto envia os seus dados, a ONU1 continua a receber novos pacotes dos utilizadores e, no final da transmissão dos 6000 bytes, envia para a OLT uma trama de controlo informando quantos bytes

possui nesse momento no *buffer*, 550 bytes no caso da Figura 2.8 b). Esta trama permite que a OLT atualize a *polling table*;

- III. A OLT sabe quando irá receber o último bit da ONU1, dado que sabe o RTT e sabe o número de bytes que a ONU está autorizada a transmitir. Por isso, a OLT envia a mensagem de controlo para a ONU2 antes do término da ligação com a ONU1, uma vez que, conhecendo o RTT da ONU2, consegue organizar a transmissão para que o primeiro bit da ONU2 chegue após o último bit da ONU1, com apenas um pequeno intervalo de guarda entre eles, de forma a salvaguardar variações no RTT e no tempo de processamento das tramas de controlo [10].

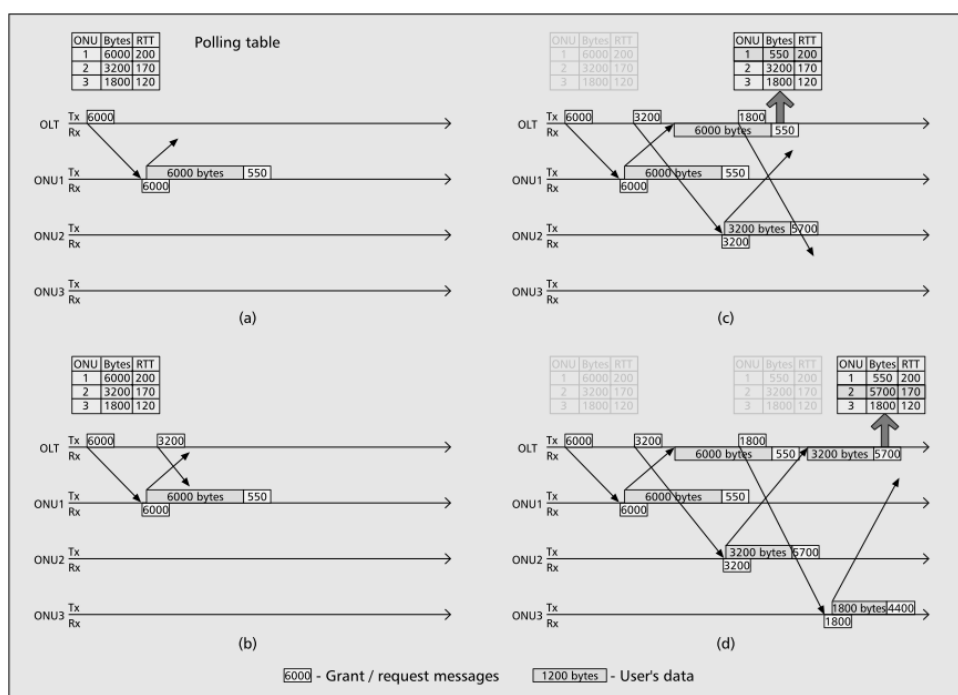


Figura 2.8 - Etapas do Algoritmo IPACT [10]

Este mecanismo permite que quando a OLT possui a informação de que uma ONU contém o seu *buffer* vazio, lhe conceda zero bytes para transmissão de dados. A ONU está, no entanto, autorizada a enviar a sua trama de controlo para informar se entretanto recebeu novos pacotes [10].

Simulações realizadas permitiram concluir que este algoritmo possui várias qualidades desejadas, tais como o facto de ser escalável com o número de ONUs no sistema e o facto de o tempo de ciclo *polling* se adaptar à carga nas filas das ONUs, permitindo uma distribuição da largura de banda pelas mesmas de acordo com as suas necessidades [10].

2.6.2 *Dynamic credit distribution (D-CRED)*

O D-CRED (*Dynamic credit distribution*) é um algoritmo do tipo DBA (*dynamic bandwidth allocation*); a atribuição dinâmica de largura de banda permite alcançar uma melhor utilização e um menor atraso das filas, em EPONs (*Ethernet Passive Optical Networks*) [11].

Apenas uma ONU pode transmitir durante o seu *slot time*, sendo o tempo de início da transmissão e o tamanho de cada *slot* atribuídos pelo algoritmo D-CRED [11].

A ideia base do algoritmo passa pela definição de um *threshold*, que os autores designaram como crédito, e que representa o máximo que a ONU pode transmitir. Esse crédito é definido pela OLT e é alterado dinamicamente, permitindo elevada precisão na atribuição de largura de banda [11].

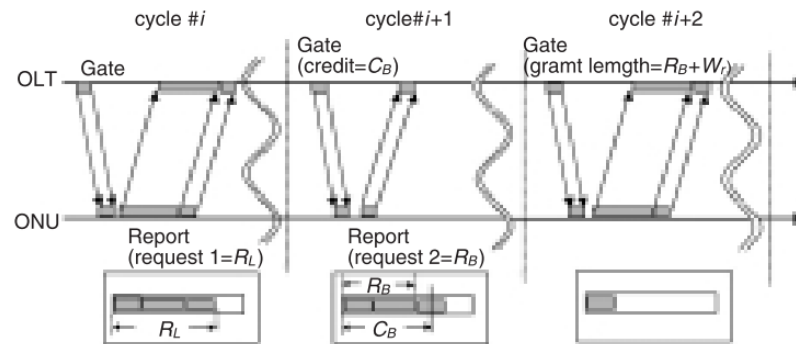


Figura 2.9 - Etapas do algoritmo D-DRED [11]

O algoritmo está ilustrado na Figura 2.9, e é descrito da seguinte forma:

- I. No ciclo $\#i$, a ONU envia uma trama para a OLT informando o tamanho da sua fila, R_L ;
- II. No ciclo $\#i+1$, a OLT envia para a ONU os créditos concebidos, C_B , e que são calculados tendo em conta R_L e as condições gerais de tráfego;
- III. Ainda no mesmo ciclo, a ONU define um limite R_B para transmissão, que pode ser igual ou inferior a C_B , e informa a OLT;
- IV. No ciclo $\#i+2$, a OLT envia para a ONU o tempo de início e o tamanho atribuído para a ligação;
- V. Ainda no mesmo ciclo a ONU inicia a transmissão de dados [11].

Os resultados da simulação deste algoritmo permitiram verificar a sua fiabilidade para uma implementação com equipamento real, dado que foram alcançados resultados para o *throughput* acima do esperado teoricamente, juntamente com uma redução do atraso nas filas, especialmente para tráfego sensível a atrasos [11].

2.7 Protocolos do Tipo *Selective Repeat*

Serão apresentados nesta secção alguns exemplos de protocolos do tipo *Selective Repeat*, referindo os objetivos e as suas principais características.

2.7.1 “*Highly Efficient Selective Acknowledgement (SACK) for TCP*”

A perda de múltiplos pacotes de dados pode ter um efeito devastador no *throughput* do protocolo TCP devido ao uso do *cumulative acknowledgment*, que implica que todos os pacotes que não são confirmados “à esquerda” do limite passado na trama ACK, ou são retransmitidos, mesmo que alguns tenham sido corretamente recebidos, ou então o emissor tem de esperar um RTT para saber mais sobre os pacotes perdidos [12].

O uso do mecanismo de *Selective Acknowledgment (SACK)* permite que o emissor tenha conhecimento dos pacotes que foram corretamente enviados, retransmitindo aqueles que efetivamente foram perdidos [12].

No documento [12] são propostas modificações ao SACK, com a implementação de um parâmetro designado *transmit state*, que é adicionado a cada pacote transmitido, e retornado cada vez que um *acknowledgment* é enviado [12].

O uso deste parâmetro permite que o emissor saiba que pacotes foram transmitidos após a informação contida no *acknowledgment*, de forma que apenas retransmita os pacotes que realmente não são confirmados, excluindo aqueles que possivelmente ainda estarão em transmissão [12].

Também foram propostas alterações na forma como os SACKs são reportados, para que o mais antigo e portanto mais crítico seja transmitido rapidamente sem que haja repetição múltipla de informação [12].

2.7.2 *Compressed Block Ack* em redes IEEE 802.11

O mecanismo de *Compressed Block Ack* (do tipo *selective repeat*) foi proposto juntamente com o método de agregação de tramas em redes IEEE 802.11 [13].

Como já referido, no *standard* IEEE 802.11e foi proposto o mecanismo de *Block Ack*; contudo, o mecanismo de fragmentação era suportado, pelo que cada MSDU podia ser fragmentada num máximo de 16 MPDUs. Com isto, o *bitmap* enviado em cada *Block Ack* requeria por cada MSDU enviada o uso de 16 bits de confirmação. Dado que o tamanho dos *Block Ack* era de 128 bytes era possível a confirmação de 64 MSDUs [13].

Com o surgimento do *standard* IEEE 802.11n, em que foi proposto o mecanismo de agregação de tramas, surgiu a necessidade de realizar uma adaptação ao *Block Ack* usado no *standard* IEEE 802.11e, uma vez que se concluiu que o uso de fragmentação não era desejável dado o aumento do *overhead*. Deste modo, o *bitmap* usado no *Block Ack* necessitaria apenas

de 1 bit para confirmar cada MSDU, levando a uma redução no tamanho do *bitmap* do *Block Ack* de 128 bytes para 8 bytes, mantendo-se a capacidade de confirmação de 64 MSDUs [13].

Simulações realizadas ao mecanismo de *Compressed Block Ack* juntamente com agregação de tramas levaram a um aumento do *throughput* de cerca de 10% [13].

2.8 Ferramentas de Simulação

Nesta secção será feita uma breve referência a alguns dos simuladores de rede mais comuns, evidenciando algumas das suas características e justificando a escolha do ns-3 para a realização da presente dissertação.

Existem no mercado diversas ferramentas de simulação que podem ser usadas como instrumento auxiliar no desenvolvimento de novos protocolos, no aperfeiçoamento de protocolos existentes, bem como na conceção de arquiteturas de rede.

O uso destas ferramentas tem assumido um papel cada vez mais preponderante, já que permitem antecipar comportamentos e avaliar o desempenho da rede ou do protocolo em estudo, e consequentemente evitar custos adicionais de uma possível implementação errada em cenário real.

Reveste-se portanto de elevada importância a escolha de um simulador de rede adequado ao problema a tratar, pelo que deverão ser considerados diversos critérios nessa escolha, entre eles o tipo de licença que possui, a adaptabilidade ao tipo de rede que se está a desenvolver, a capacidade de emulação da realidade, o tipo de linguagem de programação usada na implementação, o tipo de plataforma de desenvolvimento, a interface com o utilizador disponibilizada e por fim mas não menos importante a quantidade e qualidade da documentação disponibilizada.

Destacam-se de seguida alguns dos simuladores de redes mais comuns.

2.8.1 OPNET (*Optimized Network Engineering Tool*)

O OPNET é um simulador de redes baseado em eventos discretos e orientado a objetos. Fornece um ambiente de desenvolvimento abrangente para a especificação, simulação e análise de desempenho de redes de comunicação [14].

Trata-se de um *software* de simulação extenso e poderoso capaz de simular uma variedade de redes heterogéneas com diferentes protocolos [15].

É fornecido em pacote comercial que está disponível tanto para uso em ensino como para investigação em instituições académicas que estão abrangidas pelo “*OPNET university academic program*”. Possui diversos módulos e ferramentas e é programável em C e C++. É amplamente usado na indústria de redes quer para análise de desempenho de redes locais quer de redes de área alargada [14].

Tem como principais vantagens o facto de possuir uma biblioteca abrangente, organizada em módulos com elevado nível de detalhe; além disto possui interface com o utilizador - GUI (*Graphical User Interface*) - e possui uma apresentação personalizada dos resultados de simulação. Como principal desvantagem tem o facto de ser muito caro [14].

2.8.2 QualNet (*QualNet Developer*)

O QualNet é um simulador de redes distribuído, que representa a realidade com fidelidade, e que pode ser usado para projetar e simular grandes redes com muito tráfego. É capaz de simular redes com e sem fios bem como arquiteturas mistas. É programável em C++ e constitui a versão comercial do simulador *open source* *GloMoSim* [14, 15].

O QualNet possui ferramentas para visualização, para estruturação do protocolo, para análise estatística em tempo real e ainda ferramenta para *debugging* [14].

As suas principais qualidades prendem-se com o facto de permitir milhares de nós e de poder ser corrido numa variedade de máquinas e sistemas operativos, enquanto a maior desvantagem consiste em não possuir modelos pré-definidos que possam ser usados para construção [14].

2.8.3 GloMoSiM (*Global Mobile Information System Simulator*)

O GloMoSiM é um simulador *open source* baseado em eventos discretos. É usado para simular redes sem fios de grande escala, sendo capaz de simular milhares de nós [16].

Tem capacidade para simular comunicações *multi-hop*, que podem ser assimétricas, com suporte para os protocolos de rede mais comuns [16].

Em termos de estrutura interna, está organizado em módulos, onde cada um simula um protocolo de comunicação sem fios específico da pilha protocolar [16].

2.8.4 OMNeT++ (*Objective Modular Network Testbed in C++*)

O OMNeT++ é um simulador *open source* baseado em eventos discretos, programável em C++ e com um elevado número de utilizadores. Tem uma estrutura modular, possuindo dois tipos de módulos designados simples e compostos. Os módulos simples são usados para definir algoritmos, sendo portanto componentes ativos do simulador. Os módulos compostos consistem numa agregação de módulos simples que interagem entre si [14, 16, 17].

Ao contrário do ns-2 e ns-3, abordados de seguida, o OMNeT++ não foi projetado exclusivamente para simulação de redes, sendo no entanto este o seu principal uso [16].

Os pontos fortes do simulador estão relacionados com o facto de incluir GUI, por ser possível inspecionar o estado de cada elemento durante a simulação e pela implementação detalhada dos módulos e protocolos. Contudo, tem como desvantagens o facto de ser um pouco lento devido às longas simulações e ao grande consumo de memória, para além da dificuldade em usar [14].

2.8.5 ns-2 (*network simulator-2*)

O ns-2 foi um dos simuladores mais usados a nível global. É *open source*, orientado a objetos e baseado em eventos discretos. Foi inicialmente desenvolvido para simular algoritmos de encaminhamento, *multicast* e protocolos TCP/IP, sendo mais tarde estendido para suportar mobilidade dos nós. É escrito em C++ e usa OTcl para *scripting*. É capaz de simular tanto redes com fios como redes sem fios [14, 16].

As maiores vantagens deste simulador são o facto de não ter custo, de possuir uma visão gráfica da rede através do NAM (*Network Animator*), e devido ao seu amplo uso e pelo código desenvolvido em investigações ser em muitas situações publicado, levar a que a sua evolução seja constante, permitindo que outros investigadores desenvolvam os seus projetos sobre projetos já desenvolvidos [14].

Há, no entanto, problemas no ns-2 que foram identificados e que estão relacionados com a elevada abstração nas camadas implementadas, o que leva a que não haja correspondência entre o comportamento na simulação e numa implementação real. Outros dos problemas estão relacionados com os módulos desatualizados, com a ausência ou desatualização da documentação e ainda com o facto de o sistema de *tracing* ser difícil de utilizar [18].

Pelos motivos supracitados foi desenvolvido um novo simulador, o ns-3, que será de seguida analisado, e que constitui um substituto e não uma extensão do ns-2.

2.8.6 ns-3 (*network simulator-3*)

O projeto ns-3 foi iniciado em meados de 2006 e encontra-se em constante expansão devido à grande comunidade de desenvolvimento, o que contribui para o seu progresso e para a sua manutenção, levando ao seu rápido crescimento [16, 19].

Tinha como objetivos iniciais desenvolver um ambiente de simulação aberto, tendo como alvo o uso no ensino e em investigação, sendo uma ferramenta perfilada com as necessidades de simulação de redes modernas [18].

O ns-3 é um simulador *open source*, baseado em eventos discretos, desenvolvido em linguagem C++, com interface opcional em *Python* [16].

Com a criação do ns-3 foi dada resposta aos problemas encontrados no ns-2, tendo sido projetado com atenção ao realismo, ou seja, tentando encurtar a distância entre o ambiente de simulação e o ambiente real. A sua implementação possibilita a integração com outro *software open source*, nomeadamente na geração de ficheiros de *packet trace* com extensão *.pcap*, que podem ser analisados, por exemplo, com o *Wireshark*, e que podem ser usados para *debugging*. É ainda de salientar a boa organização do código fonte, bem como de realçar a quantidade e a qualidade da documentação disponível [16, 18].

A sua estrutura é baseada em módulos, sendo estes distribuídos como ilustrado na Figura 2.10, existindo dependências dos módulos superiores em relação aos módulos inferiores.

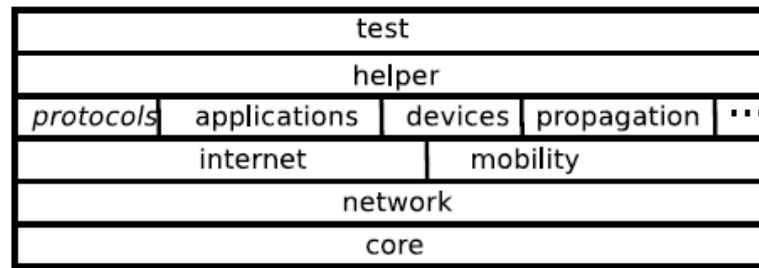


Figura 2.10 - Ilustração da organização em módulos do simulador ns-3 [20]

Pelas vantagens já mencionadas, de que se realça o facto de ser *open source*, de possuir um código fonte organizado e atualizado, pela qualidade da documentação existente e, não desmerecendo, pelo facto de se tratar de um simulador que é amplamente aceite pela comunidade científica, foi esta a ferramenta de simulação usada para a elaboração da presente dissertação.

De entre os vários módulos disponíveis para o ns-3, apresenta-se de seguida uma breve descrição do módulo *Wi-Fi* disponível, que servirá como base para a implementação que constitui o objeto de estudo da presente dissertação.

2.8.6.1 Módulo *Wi-Fi* existente no ns-3

O simulador ns-3 possui um módulo designado *wifi*, baseado no *standard* IEEE 802.11, que possui capacidade para suportar tanto redes infraestruturadas como redes *ad-hoc*.

Estão já implementadas diferentes versões do *standard* IEEE 802.11, entre elas o 802.11a, 802.11b, 802.11g e 802.11n.

Alguns dos mecanismos mais recentes encontram-se já implementados, nomeadamente o mecanismo de agregação, tanto A-MSDU como A-MPDU.

Tem capacidade para suportar diferentes modelos de perdas devido à propagação e diferentes modelos de atrasos de propagação, juntamente com vários algoritmos de controlo de débito.

A implementação é modular e está organizada como mostrado na Figura 2.11.

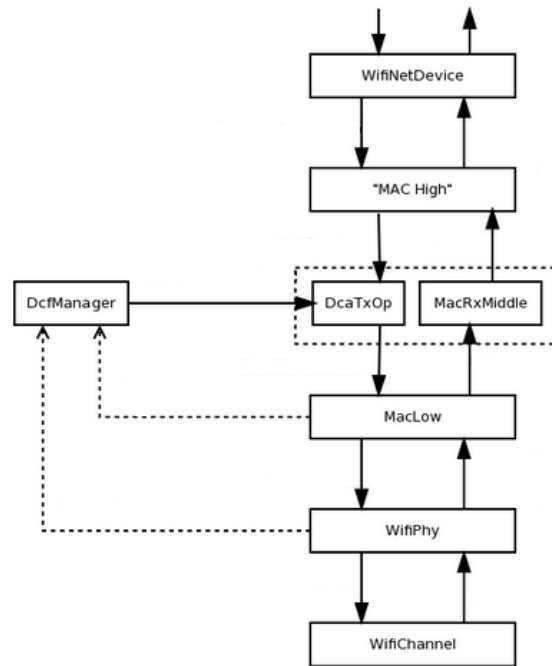


Figura 2.11 - Arquitetura do módulo *Wi-Fi* existente no ns-3 [1]

- O módulo *WifiNetDevice* é responsável pela introdução do cabeçalho LLC (*Logical Link Control*), pela passagem dos pacotes da camada de rede para o *MAC High* e, em sentido inverso, pela entrega das tramas do *MAC High* à camada de rede;
- Há três tipos de *MAC High*, que definem o tipo de elemento da topologia *Wi-Fi* em uso:
 - o *ns3::ApWifiMac*, que define um *Access Point* (AP), que realiza a geração de *beacons* e as respostas a *Association Request*;
 - o *ns3::StaWifiMac*, que define uma estação (STA), que implementa o *active probing* e o envio de *Association Request*;
 - a implementação que permite criar nós para uma rede *ad-hoc*, que se designa *ns3::AdhocWifiMac*.
- O módulo *DcaTxOp* é o responsável pela gestão da fila, pela fragmentação dos pacotes e pela retransmissão dos mesmos. O *DcaTxOp* é usado em MACs que não têm o *QoS* ativo. Como alternativa o *EdcaTxopN* é usado em MACs em que o *QoS* está ativo, sendo também este o módulo responsável por realizar a A-MSDU;
- O módulo *DcfManager* implementa as funcionalidades de DCF referidas anteriormente;
- O módulo *MacLow* é o responsável por passar as tramas ao módulo *WifiPhy*, sejam elas de dados ou de controlo, sendo também aqui que se realiza a agregação de MPDUs;
- O módulo *WifiPhy* é o responsável pelo envio e receção de tramas, consultando o modelo de erros para obter a probabilidade de uma trama ser recebida com base no SINR (*Signal to interference plus noise ratio*);

- Por fim o módulo *WifiChannel* é o responsável por interligar diferentes dispositivos *Wi-Fi*. Para a transmissão das tramas entre dispositivos são usados os modelos quer de perda, quer de atraso [21].

Capítulo 3

Protocolo TPAP

O protocolo TPAP (*TDMA Polling with Ack Piggyback*) [22] opera numa camada protocolar normalmente designada 2.5, uma vez que se situa entre a camada MAC e a camada de Rede, como ilustrado na Figura 3.1, em que é apresentada a pilha protocolar TCP/IP sem e com o protocolo TPAP.

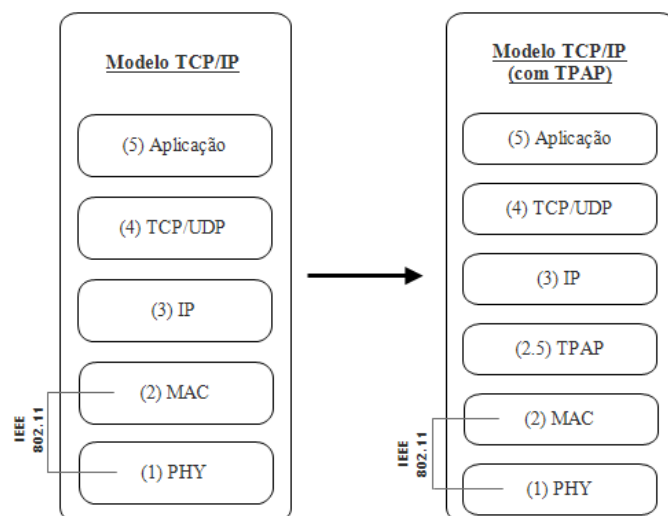


Figura 3.1 - Modelo TCP/IP com protocolo TPAP

3.1 Considerações Preliminares

O protocolo TPAP foi concebido para operar sobre a camada MAC de redes IEEE 802.11, com o objetivo de suportar a comunicação entre uma Estação Base (EB) e um conjunto de Estações Clientes (ECs), numa configuração ponto-multiponto, ilustrada na Figura 3.2. A comunicação é efetuada sobre um canal *wireless half-duplex* partilhado e envolve, para além

das tramas de controlo que suportam os mecanismos protocolares, a troca de tramas de dados nos dois sentidos.

O TPAP combina controlo centralizado de acesso ao meio baseado em *polling* e um mecanismo de recuperação de erros do tipo janela deslizante, baseado em confirmação seletiva de blocos de tramas (*Selective Repeat*). A implementação do controlo de erros e o suporte a retransmissões implementados no TPAP pressupõe a eliminação na camada MAC de confirmações individuais (com recurso a tramas ACK) e de retransmissões de tramas (responsáveis por *overhead* considerável). A janela constitui igualmente uma forma de controlo de fluxo; numa versão simples os créditos atribuídos às ligações unidirecionais podem ser fixos (e eventualmente diferentes), mas numa versão mais avançada os créditos podem variar dinamicamente para adaptação a condições de tráfego variáveis com o objetivo de tornar o mecanismo de *polling* mais eficiente (constituindo uma forma de *smart polling*).

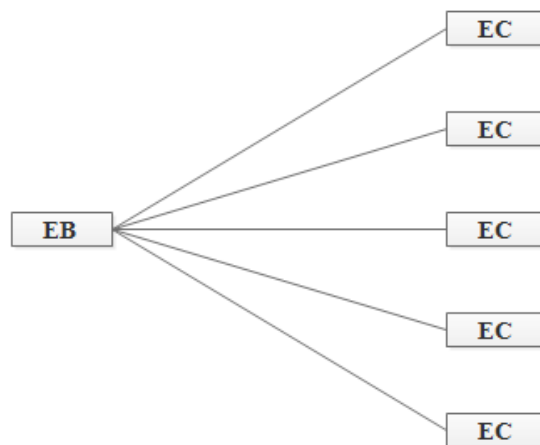


Figura 3.2 - Configuração ponto-multiponto

A combinação dos mecanismos referidos constitui a versão completa do protocolo, que designamos simplesmente por TPAP.

O TPAP é adequado para as variantes do *standard* IEEE 802.11 que não suportam agregação de tramas na camada MAC. A operação do protocolo emula agregação de tramas, pois cada estação pode enviar um *burst* de tramas (se disponíveis) até ao limite da janela; no entanto, cada trama transporta o seu próprio cabeçalho PHY, para além dos cabeçalhos MAC e TPAP e, por omissão, os tempos de guarda entre tramas são mantidos (veremos mais adiante que é possível melhorar a eficiência reduzindo o intervalo entre tramas).

Como visto anteriormente, as variantes mais recentes do *standard* IEEE 802.11, como o IEEE 802.11n e o IEEE 802.11ac suportam agregação de tramas e um mecanismo de confirmação seletiva de tramas em bloco, baseado no mesmo princípio de funcionamento adotado no TPAP.

À medida que aumenta o débito binário do canal, os ganhos conseguidos com agregação de tramas em A-MPDUs compensam largamente as vantagens inerentes à eliminação de ACKs

individuais de tramas não agregadas. Esta agregação pressupõe a utilização do mecanismo nativo de confirmação e retransmissão seletiva de tramas na camada MAC, pelo que o mecanismo do TPAP deixa de fazer sentido, por constituir uma duplicação.

Contudo, tanto o mecanismo de *polling* como o mecanismo de controlo de fluxo por meio da janela devem ser mantidos, uma vez que a sua combinação permite por um lado controlar o acesso ao meio, reduzindo ou mesmo eliminando as colisões, e por outro permite distribuir de forma controlada os recursos da rede pelas ligações unidirecionais estabelecidas, limitando o número de tramas que são enviadas em cada ciclo de *polling* por cada estação lógica. Designamos por TPAP simples esta versão reduzida do protocolo.

3.2 Funcionamento do protocolo TPAP

O mecanismo de *polling* permite, como já referido, realizar o controlo da transmissão e tem como objetivo garantir que em cada momento apenas uma das estações pode enviar tramas TPAP. Este controlo de transmissão é liderado pela EB, que fornece às estações clientes, através do envio de uma trama de *polling* (POLL), o direito a transmitir as suas tramas de dados, sendo devolvido no final o controlo à EB por parte da EC, através do envio uma trama designada RET (de *return*).

Dado que o protocolo é do tipo *Selective Repeat*, isto exige que cada trama de dados seja numerada de forma a permitir identificar as tramas para efeito de confirmação e retransmissão. Para além disso, uma vez que o protocolo opera em ciclos, as retransmissões ocorrem em ciclos diferentes do da primeira transmissão, pelo que a sequência de numeração deve ser preservada enquanto a ligação respetiva estiver ativa, para garantir coerência.

O tamanho da janela deslizante que limita o número de tramas de dados transmitidas em cada ciclo é definido com base no módulo de numeração $M=2^k$, onde k é o número de bits usados para numerar as tramas. Para que seja possível distinguir novas tramas de duplicados resultantes de retransmissões, o tamanho máximo possível da janela W_{max} não pode exceder $M/2$; no entanto o tamanho configurado da janela W pode ser inferior a W_{max} .

O uso do *Block Acknowledgment* (BACK) tem como objetivo reduzir o *overhead* quando comparado com o envio de confirmações individuais; para além disso evita retransmissões de tramas corretamente recebidas (ao contrário do que acontece em protocolos do tipo *GoBackN*). Cada trama BACK transporta um *bitmap*, que abrange todas as tramas pertencentes à janela atual do recetor, antes de uma possível rotação, e é usado para sinalizar as tramas que foram corretamente recebidas e implicitamente as tramas que não sendo confirmadas positivamente terão de ser retransmitidas. Em cada posição do *bitmap*, tramas recebidas corretamente são assinaladas com o valor 1; o valor 0 numa posição do *bitmap* significa que a trama correspondente não foi recebida corretamente, ou não foi ainda transmitida (o emissor é capaz de distinguir as duas situações).

A trama BACK transporta ainda o limite inferior da janela do recetor, designada R_i , para que o emissor tenha uma referência para identificar as tramas cobertas pelo *bitmap*.

Apresenta-se de seguida um exemplo para que se torne mais claro o uso do *bitmap*, o número de bits necessários para o representar, bem como o número de bits necessários para representar R_i .

Se optarmos por $k=4$, o módulo de numeração será $M=2^4=16$, e portanto $W_{max}=8$. Considerando $W=8$, são necessários 4 bits para representar as tramas de dados e 8 bits para representar o *bitmap*.

Se considerarmos que a janela atual do emissor abrange as tramas entre 5 e 12, tal como a janela do recetor (neste caso estão sincronizadas), e que o emissor enviou as tramas 5 a 9, tendo as tramas 6 e 9 sido recebidas com erro (descartadas na camada MAC), a trama BACK enviada pelo recetor para o emissor deveria indicar um *bitmap* e um R_i tal como representado na Figura 3.3.

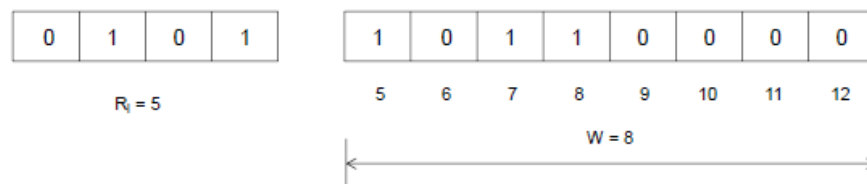


Figura 3.3 - Exemplo ilustrativo do *bitmap*

Vemos na Figura 3.3 os 4 bits enviados para indicar o $R_i=5$, bem como os 8 bits enviados para o *bitmap*. As tramas 5, 7 e 8 estão assinaladas com 1, indicando que foram corretamente recebidas; as tramas 6 e 9 estão marcadas com 0, competindo ao emissor a sua retransmissão. Uma vez que as tramas 10 a 12 não foram transmitidas, o valor 0 nessas posições do *bitmap* não implica qualquer ação; no entanto, se alguma dessas posições fosse assinalada com 1, o emissor deveria interpretar esse facto como um erro de protocolo.

A rotação da janela do emissor depende do *bitmap* recebido, o qual pode obrigar a retransmissões de tramas antigas, o que leva a que estas tenham de ser mantidas dentro da janela.

No exemplo mostrado anteriormente, a janela do emissor pode avançar uma posição, já que a trama com número 5 foi corretamente recebida pelo recetor; no entanto como a trama 6 tem de ser retransmitida a janela do emissor permanecerá entre 6 e 13, sendo possível transmitir 4 novas tramas (10 a 13), para além da retransmissão das tramas 6 e 9.

Dado que o protocolo exige a entrega à camada superior de tramas de dados recebidas corretamente e estritamente em sequência, a rotação da janela do recetor obedece a este princípio.

A Figura 3.4 ilustra a sequência de tramas trocada entre a EB e cada uma das ECs.

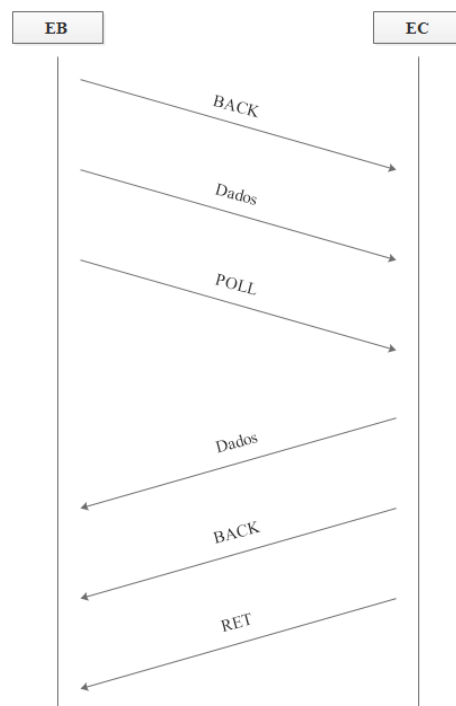


Figura 3.4 - Sequência de mensagens trocadas entre a EB e cada uma das ECs

A EB começa por enviar a trama de BACK para a EC, que confirma as tramas de dados enviadas pela EC no ciclo anterior, de seguida envia as suas tramas de dados até ao máximo permitido pela janela deslizante e por fim envia a trama de POLL conferindo autorização à EC para transmitir as suas tramas. A EC começa por enviar as suas tramas de dados, até ao máximo permitido pela janela deslizante, posteriormente envia a trama BACK que confirma as tramas de dados enviadas pela EB neste ciclo e por fim devolve o controlo à EB através da trama RET. Esta sequência tem como principal objetivo minimizar o tempo de processamento nas ECs aquando da inversão do sentido de transmissão.

Caso ocorra a perda da trama BACK, esta é acomodada pelo protocolo; no entanto, de forma a contornar possíveis perdas das tramas POLL ou RET, que são críticas dado que a EB ficaria bloqueada à espera que a EC lhe devolvesse o controlo, é implementado um temporizador na EB que é ativado após o envio da trama POLL. Caso ocorra perda da trama POLL ou da trama RET (subsequente à receção correta da trama POLL), após o temporizador expirar (*time-out*), a EB inicia uma nova interação com a EC seguinte.

3.3 Modelo Teórico

Como complemento à especificação do protocolo TPAP foi desenvolvido um modelo teórico ideal [23], impondo condições de tráfego particulares, mas que apesar disso procura caracterizar o sistema através de um conjunto limitado de parâmetros e identificar os fatores com maior impacto no desempenho (incluindo os relativos ao próprio protocolo) e a sua importância relativa.

No desenvolvimento do modelo foi considerada a influência de fatores alheios ao protocolo TPAP, destacando-se o *overhead* associado ao mecanismo de acesso ao meio do *standard* IEEE 802.11. O número de tramas de dados transmitidas é igualmente relevante; uma vez que o *overhead* associado ao protocolo TPAP é fixo em cada interação entre a EB e uma EC, independentemente do número de tramas de dados enviadas, o seu impacto relativo diminui com o aumento do número de tramas de dados enviadas. Deste ponto de vista o protocolo TPAP é particularmente eficiente para débitos elevados dos fluxos em competição, situação em que o desempenho de protocolos do tipo CSMA/CA é comprometido com a ocorrência de colisões, o que é agravado com a retransmissão de tramas. Por outro lado, o *overhead* fixo pode ser penalizador do ponto de vista do atraso no envio de pacotes em situações de pouco tráfego e com um número elevado de estações, pelo que é necessário quantificar esse *overhead* e identificar maneiras de reduzir o seu impacto.

O modelo teórico está organizado em três secções, sendo a primeira correspondente ao modelo básico desenvolvido para o protocolo TPAP completo, em que não há recurso ao mecanismo de agregação, e onde é considerada a ausência de erros de transmissão.

Posteriormente o modelo básico foi expandido de forma a considerar-se o protocolo TPAP completo, sem agregação mas considerando a ocorrência de erros em tramas de dados.

Por fim foi criada uma extensão do modelo de forma a analisar-se o protocolo TPAP simples com recurso ao mecanismo de agregação na camada MAC.

3.3.1 Modelo básico para TPAP completo, sem agregação e sem erros

O modelo teórico é caracterizado tendo como base um sistema com uma estação base (EB) e N_c estações clientes (ECs); dada a natureza bidirecional da comunicação entre a EB e cada EC e a topologia ponto-multiponto, existem N_e estações lógicas, sendo $N_e = 2N_c$.

Dada a existência de N_e estações lógicas, o número máximo de tramas de dados, n_{max} , que é possível enviar por ciclo de *polling* é dado por

$$n_{max} = W \cdot N_e \quad (3.1)$$

onde W representa o tamanho da janela de transmissão.

A fração de créditos utilizados pelas estações num ciclo de *polling* é definida pelo parâmetro α , e obtido através de

$$\alpha = \frac{n}{n_{max}} \quad (3.2)$$

onde n representa o número de tramas de dados efetivamente enviadas num ciclo de *polling* por todas as estações lógicas (independentemente da sua distribuição pelas estações). Quando as estações lógicas gerarem fluxos com débitos tais que, num ciclo de *polling*, cada estação esgote a sua janela (estação saturada), n será igual a n_{max} e α será igual a 1.

A versão inicial do protocolo foi desenvolvida para o caso em que não se utiliza o mecanismo de agregação de tramas, pelo que se assume que a transmissão de cada trama tem um *overhead* diretamente associado. Esse *overhead* varia de trama para trama (devido ao intervalo de contenção variável) mas considerou-se que o seu valor médio ao longo de vários ciclos é praticamente constante. Esse *overhead* está incorporado em T_p , que representa o tempo total de ocupação do meio com a transmissão de uma trama de dados. Para o cálculo da eficiência máxima considerou-se um tamanho de tramas constante, mas as fórmulas mantêm-se válidas se o tamanho das mesmas for variável, obtendo-se neste caso um valor de T_p médio.

O protocolo TPAP está inevitavelmente sujeito ao *overhead* próprio; contudo o *overhead* devido às camadas inferiores, MAC e PHY, implica que a capacidade útil disponível para o protocolo, designada por C , seja inferior à capacidade C_c do canal físico, e igual a

$$C = \frac{1}{T_p} \quad (3.3)$$

onde T_p é definido considerando duas componentes (*payload* e *overhead*) e obtido por

$$T_p = T_d + T_o \quad (3.4)$$

T_d corresponde ao tempo de transmissão do *payload* de uma trama de dados e T_o é o *overhead* das camadas inferiores associado à transmissão da trama (inclui cabeçalhos PHY e MAC, *gaps* entre tramas e períodos de contenção). T_d é dado por

$$T_d = \frac{1}{C_c} \quad (3.5)$$

onde C_c representa o débito do canal físico.

O *overhead* associado ao protocolo TPAP, definido como O_{TPAP} , corresponde à soma dos *overheads* associados a cada interação da estação base com as estações clientes. Assim, o tempo de um ciclo de *polling*, T , é obtido através de

$$T = O_{TPAP} + n.T_p \quad (3.6)$$

ou seja, T é obtido somando o *overhead* associado ao protocolo TPAP com o tempo total gasto, por todas as estações lógicas, com a transmissão de tramas de dados.

De forma a expressar o impacto relativo do *overhead* associado ao protocolo TPAP, face ao tempo máximo gasto num ciclo de *polling* com a transmissão de tramas de dados, foi definido um parâmetro k , dado por

$$k = \frac{O_{TPAP}}{n_{max}.T_p} \quad (3.7)$$

Das expressões 3.6 e 3.7 podemos derivar

$$T = (k + \alpha).W.N_e.T_p \quad (3.8)$$

O débito útil total, R , é definido em pacotes por segundo e dado por

$$R = \frac{n}{T} = C \cdot \frac{\alpha}{k + \alpha} \quad (3.9)$$

A expressão da eficiência, S , é dada por

$$S = \frac{R}{C} = \frac{\alpha}{k + \alpha} \quad (3.10)$$

De notar que S representa a eficiência do protocolo e não a do sistema. Para obter a eficiência global seria necessário multiplicar S por T_d/T_p .

A eficiência máxima do protocolo TPAP, S_{max} , é expressada com base no parâmetro k , e obtida da expressão 3.10 considerando $\alpha=1$ (ocupação total das janelas de todas as estações)

$$S_{max} = \frac{1}{k+1} \quad (3.11)$$

O débito útil total máximo, R_{max} , é obtido através de

$$R_{max} = C.S_{max} \quad (3.12)$$

Quando algumas estações estão saturadas e outras não, é útil exprimir o débito das estações saturadas em função do débito das estações não saturadas. Considerando que

existem N_s estações saturadas, o seu débito útil será inferior ou no limite igual ao débito gerado. O débito útil global das estações saturadas, R_o , é obtido por

$$R_o = (C - R_1) \cdot \frac{N_s}{N_s + k \cdot N_e} \quad (3.13)$$

onde R_1 representa o débito útil global das estações não saturadas, que por sua vez é igual ao débito total gerado pelas mesmas.

Nestas condições não se atinge a eficiência máxima; para melhorar a eficiência poderia aumentar-se o tamanho da janela das estações saturadas (aumentando assim o respetivo débito), sem violar o tempo máximo de ciclo. Na prática corresponde a atribuir mais créditos a essas estações, porque as restantes não usaram todo o crédito disponível. Isto é a base para a atribuição dinâmica de créditos, como um possível instrumento de *smart polling*.

3.3.2 Extensão do modelo básico para TPAP completo, sem agregação, mas considerando erros

As equações apresentadas anteriormente para o modelo teórico pressupõem condições ideais, sem ocorrência de perda de pacotes e consequentemente sem retransmissões.

Contudo, quando ocorrem erros é necessária uma adaptação das equações anteriormente apresentadas de forma a incluírem a probabilidade de perda de pacotes, designada p_e , e representarem a influência dessas perdas no número de pacotes retransmitidos.

Para se distinguir os parâmetros utilizados para o cenário sem erros dos parâmetros utilizados no cenário com erros, será utilizada a nomenclatura n' , α' e T' , para designar, respetivamente, o número de pacotes transmitidos por ciclo, o seu valor relativo à ocupação máxima da janela e o tempo por ciclo.

Dada a ocorrência de retransmissões, o número médio de pacotes transmitidos por ciclo é obtido por

$$n' = n'_u + n'_r \quad (3.14)$$

onde n'_u representa o número de pacotes úteis e n'_r o número de pacotes retransmitidos.

Expressando n'_u e n'_r em função da probabilidade de perda de pacote e do número total de pacotes transmitidos por ciclo, obtemos

$$n'_r = p_e \cdot n' \quad (3.15)$$

$$n'_u = (1 - p_e) \cdot n' \quad (3.16)$$

Dada a relação de α com n , o parâmetro α' obtém-se de forma semelhante, dado por

$$\alpha' = \alpha'_u + \alpha'_r \quad (3.17)$$

onde α'_r e α'_u se obtêm através de

$$\alpha'_r = p_e \cdot \alpha' \quad (3.18)$$

$$\alpha'_u = (1 - p_e) \cdot \alpha' \quad (3.19)$$

Quer para o protocolo TPAP, quer para as camadas inferiores, o facto de num ciclo de transmissão ocorrerem apenas transmissões de pacotes de dados novos e/ou retransmissões é irrelevante do ponto de vista do *overhead*, pelo que foi assumido que os valores dos parâmetros k e C são iguais aos relativos ao cenário homólogo em que não ocorrem erros.

Daqui resulta que

$$T' = (k + \alpha') \cdot W \cdot N_e \cdot T_p \quad (3.20)$$

e

$$R' = \frac{n'}{T'} = C \cdot \frac{\alpha'}{k + \alpha'} \quad (3.21)$$

R' representa o débito total oferecido pelo TPAP à camada MAC, uma vez que n' inclui quer os pacotes novos, quer os pacotes retransmitidos, pelo que o débito útil é dado por

$$R'_u = \frac{n'_u}{n'} \cdot R' = C \cdot \frac{\alpha'_u}{k + \alpha'} = C \cdot (1 - p_e) \cdot \frac{\alpha'}{k + \alpha'} \quad (3.22)$$

Para garantir o mesmo débito que na ausência de erros (o que é possível enquanto não ocorrer saturação), o fator $(1 - p_e)$ tem de ser compensado com o aumento de α' em relação a α , com o inerente aumento do tempo médio de ciclo. Por outro lado, quando é necessário retransmitir uma ou mais tramas, a rotação da janela é parcial e a sua amplitude depende da posição na janela da trama mais antiga a retransmitir. Deste modo o valor máximo de α' é inferior a 1, o que constitui um fator adicional de redução do débito útil máximo, para além do correspondente a $(1 - p_e)$.

3.3.3 Modelo para TPAP simples, considerando agregação na camada MAC

Quando as tramas enviadas num ciclo por uma estação são agregadas numa única A-MPDU, o *overhead* associado à camada PHY é constante e inclui o respetivo cabeçalho. De forma a facilitar a análise, decidiu-se incluir também nesse *overhead* o tempo relativo ao envio de confirmações BACK, uma vez que existe apenas uma trama BACK por cada trama MAC

agregada. Esta inclui os cabeçalhos das sub-tramas MAC bem como os delimitadores entre sub-tramas, constituindo o *overhead* da camada MAC.

Nesta modelização designamos por O_{PHY} o *overhead* total da camada física considerando todas as estações lógicas e que em todas se realiza agregação (condição necessária para que ocorra eficiência máxima). Por outro lado, passamos a designar por T_{MAC} o tempo total de ocupação do meio com a transmissão de uma sub-trama de dados, excluindo como se disse o tempo relativo à trama BACK enviada por cada trama agregada.

Deste modo, o tempo de ciclo, T , é dado por

$$T = O + n \cdot T_{MAC} \quad (3.23)$$

onde o *overhead* associado à camada MAC está incluído em T_{MAC} e O representa o *overhead* total conjunto da camada PHY e do protocolo TPAP, considerando as N_e estações, isto é

$$O = O_{TPAP} + O_{PHY} \quad (3.24)$$

O parâmetro k será agora designado por k^* , uma vez que, nesta formulação, traduz o impacto de O (e não apenas de O_{TPAP}) em relação a T_{MAC} , sendo definido por

$$k^* = \frac{O}{n_{max} \cdot T_{MAC}} \quad (3.25)$$

A capacidade útil disponível se apenas existisse o *overhead* da camada MAC é designada por C^* e dada por

$$C^* = \frac{1}{T_{MAC}} \quad (3.26)$$

Poderá agora escrever-se

$$R = C^* \cdot \frac{\alpha}{k^* + \alpha} = C_c \cdot \frac{T_d}{T_{MAC}} \cdot \frac{\alpha}{k^* + \alpha} \quad (3.27)$$

em que se evidenciam os fatores que contribuem para a degradação de desempenho, um relativo à camada MAC e outro à camada PHY e ao protocolo TPAP em conjunto.

O valor de R_{max} obtém-se fazendo $\alpha = 1$. Pode ainda obter-se a eficiência máxima global (referida a C_c) ou apenas a relativa à camada PHY e ao protocolo TPAP em conjunto (referida a C^*), embora se conheça o peso relativo dos componentes O_{PHY} e O_{TPAP} em O .

Capítulo 4

Metodologia

Neste capítulo será apresentada a metodologia usada para a implementação do protocolo TPAP na pilha protocolar existente no ns-3, nomeadamente como esta foi adaptada para suportar o novo protocolo e todos os mecanismos que o integram.

De seguida será explicado como foi realizada a extração de dados da simulação, como foram introduzidos erros ao nível do protocolo TPAP, de forma a verificar o seu comportamento na ocorrência de perda de tramas, sendo posteriormente referida a adaptação do protocolo TPAP completo para o protocolo TPAP simples.

Por fim será revisto o ficheiro de testes desenvolvido de forma a testar o protocolo implementado e a extrair resultados de simulação que permitam retirar conclusões quanto ao seu desempenho.

4.1 Implementação do protocolo TPAP no ns-3

De forma a incluir o protocolo TPAP no módulo *Wi-Fi* apresentado anteriormente, decidiu-se adaptar o módulo *WifiNetDevice* para suportar todo o mecanismo subjacente ao protocolo a implementar, ao qual foi dado o nome *WifiTPAP*, passando o módulo *wifi* do ns-3 a ter o aspeto ilustrado na Figura 4.1.

O módulo *DcfManager* foi ocultado da Figura 4.1 por uma questão de simplicidade, mantendo-se no entanto na implementação.

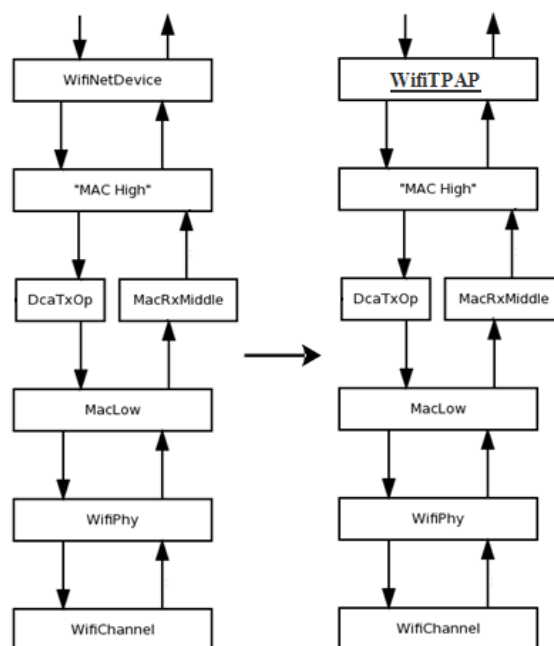


Figura 4.1 - Módulo *Wi-Fi* do ns-3 antes e depois da implementação do protocolo TPAP

A escolha do módulo *WifiNetDevice* para a implementação do protocolo TPAP deve-se ao facto de este módulo estar localizado acima da camada MAC, como assumido para o protocolo TPAP, e em segundo lugar por possuir os métodos que fazem a ligação entre as camadas de Rede e MAC: o método *Send*, através do qual os pacotes de nível 3 são passados para o nível 2, e o método *ForwardUp*, através do qual as tramas de nível 2 são passadas para o nível 3, pelo que constitui uma boa base para a implementação do protocolo TPAP.

4.1.1 Alterações efetuadas ao módulo *WifiNetDevice*

De forma a implementar o protocolo TPAP no módulo *WifiTPAP*, que como referido consistiu numa adaptação do módulo *WifiNetDevice*, foi necessário alterar os dois métodos já existentes, o *Send* e o *ForwardUp*, para além da criação de novos métodos que permitam implementar tanto o mecanismo de *polling*, como o mecanismo de *selective repeat* e de janela deslizante.

O esquema da Figura 4.2 permite ilustrar o ciclo de processamento que é executado no módulo *WifiTPAP*.

4.1.1.1 Métodos para controlo do acesso ao meio (implementação do mecanismo de *polling*)

- **Método para EB enviar trama POLL** - Neste método é criado um novo pacote, ao qual é acrescentado um cabeçalho MAC, que identifica o pacote como sendo do tipo POLL. De seguida este pacote é enviado para o “MAC HIGH”, mais precisamente para o *ApWifiMac*, uma vez que se trata da EB. No “MAC HIGH”, o pacote é recebido num método que foi criado para receber este tipo de pacotes, e que se limita a reencaminhá-lo para o nível inferior, o *DcaTxOp*. Após o envio do pacote é ativado o temporizador, salvaguardando possíveis falhas de transmissão quer da trama POLL, quer da trama RET da EC para a EB.
- **Método para EC receber trama POLL** - No “MAC HIGH” das ECs (o *StaWifiMac*) foi necessário realizar uma alteração para que o método onde são recebidas as tramas possa identificar uma trama POLL e notificar o método responsável pela receção da trama POLL no *WifiTPAP*, onde será chamado o método responsável pelo envio de tramas de dados nas ECs.
- **Método para EC enviar trama RET** - É semelhante ao método responsável pelo envio da trama POLL pela EB, ou seja, há a criação de um pacote, ao qual é acrescentado um cabeçalho MAC, que identifica o pacote como sendo do tipo RET. De seguida o pacote é passado ao “MAC HIGH”, neste caso o *StaWifiMac*, para um método que foi implementado para receber este tipo de pacotes, e que os reencaminha para o nível inferior, para o *DcaTxOp*. Neste caso não há ativação do temporizador.
- **Método para EB receber trama RET** - Tal como para o envio da trama RET, este método é em tudo semelhante ao método para receber a trama POLL. No *ApWifiMac* procedeu-se a uma alteração para que o método onde se recebem as tramas faça a identificação de uma trama RET e notifique o método em questão. Uma vez notificado, é chamado o método para a EB enviar as tramas de dados.

4.1.1.2 Métodos para controlo de erros (implementação do *Selective Repeat*)

- **Método para enviar trama BACK** - Os métodos para envio de tramas BACK pela EB e pelas ECs são semelhantes pelo que a apresentação da sua implementação será realizada em conjunto. Para o envio de tramas BACK foi criado um novo tipo de cabeçalho, capaz de transportar quer o *bitmap*, quer o limite inferior atual da janela

do recetor, R_i . Posto isto, para o envio de uma trama BACK é criada uma nova trama, na qual é introduzido o cabeçalho novo que transporta o *bitmap* e o R_i , bem como um cabeçalho MAC que identifica a trama como sendo um BACK. Uma vez criada a trama, esta é enviada para o nível inferior, para o “MAC HIGH”, para um método que se limita a passar a trama ao módulo *DcaTxOp*. No final é atualizado o *bitmap* enviado, de forma a proceder-se à rotação da janela do recetor, se for o caso.

- **Método para receber trama BACK** - Tal como no método anterior, também este é igual para a EB e para as ECs pelo que a análise é igual para ambos. Tal como em casos anteriores, também aqui foi necessário realizar alterações no “MAC HIGH”, para que o método onde são recebidas as tramas consiga identificar tramas do tipo BACK e notificar o respetivo método. Uma vez notificado o método, é inicialmente removido o cabeçalho onde é transportado o *bitmap* e o R_i . De seguida é realizada a atualização da janela do emissor, sendo também atualizado o *bitmap* recebido para ser usado no processo de transmissão, conforme representado na Figura 4.3.



Figura 4.3 - Atualização do *bitmap* recebido

Por inspeção do *bitmap* recebido, e assumindo $R_i=1$, vemos que as tramas 2, 5 e 6 precisam de ser retransmitidas. O *bitmap* é então atualizado e guardado para que, no processo de envio de tramas, seja usado para identificar as tramas a (re)enviar, já que a ordem na fila de espera é a mesma presente no *bitmap*. Neste caso apenas a trama 1 é eliminada da fila; por facilidade de implementação optou-se por manter as tramas 3, 4 e 7 na fila de espera, apesar de já terem sido corretamente transmitidas e apagar apenas blocos seguidos de tramas, mantendo-se a sequência do *bitmap* igual à da fila de espera. Neste caso após se apagar a trama 1, a fila de espera será composta pelas tramas entre 2 e 7, para além de outras entretanto guardadas. Quando se iniciar o processo de envio, retira-se o pacote da fila e verifica-se se este deve ser enviado, com base no *bitmap* recebido, e posteriormente atualizado; nos

casos em que o *bitmap* possui a posição correspondente assinalada com 1, que indica que a trama já foi corretamente transmitida, a trama não é enviada.

No final deste método são apagadas as tramas da fila de espera, de acordo com o explicado anteriormente.

4.1.1.3 Métodos para envio e receção de tramas de dados

- **Método para EB enviar tramas de dados** - Neste método, em primeiro lugar a EB começa por identificar qual o endereço MAC da próxima EC com a qual iniciará uma nova interação. Posteriormente é chamado o método responsável pelo envio da trama BACK, para o endereço MAC respetivo. De seguida acede ao valor atual do limite inferior da janela do emissor, E_i , para a respetiva EC, que será usado para a numeração das tramas a enviar. Depois são enviadas tramas até ao máximo permitido pela janela deslizante, ou caso o número de tramas seja inferior ao tamanho da janela, são enviadas as tramas existentes. Para o envio das tramas de dados, inicialmente é extraída a trama da fila de espera da EC atual, verifica-se se a mesma deve ser enviada, com base no *bitmap* recebido, tal como referido anteriormente, e em caso afirmativo é colocado o número de sequência correspondente e enviada a trama para o nível inferior, para o *ApWifiMac*. Uma vez enviada a trama, é incrementado o número de transmissões da mesma. Por fim é chamado o método para envio da trama POLL.
- **Método para EC enviar tramas de dados** - Este método é semelhante ao apresentado anteriormente, com as devidas exceções que são de seguida mencionadas. Em primeiro lugar são enviadas as tramas de dados, e o processo é em tudo semelhante ao caso da EB. A maior diferença reside no facto de o método responsável pelo envio da trama BACK ser chamado após todas as tramas de dados serem enviadas, ao contrário do caso da EB em que é chamado antes. Por fim é chamado o método para o envio da trama RET.
- **Método para receber tramas de dados** - O método para as ECs e a EB receberem as tramas de dados é o já referido *ForwardUp*, que já estava implementado no *WifiNetDevice*, e ao qual foi necessário introduzir alterações para satisfazer as exigências do protocolo TPAP. Neste método, tal como estava definido, apenas era removido o cabeçalho LLC, sendo a trama de seguida passada para o nível superior. Foi necessário introduzir alterações para que ao receber uma trama, caso esta esteja dentro da janela do recetor seja assinalado no *bitmap*, identificando que a trama foi corretamente recebida. Outra alteração resultou da necessidade de verificar se todas as tramas anteriores foram já recebidas, para

que a nova trama possa ser entregue ao nível superior, uma vez que o protocolo TPAP exige a entrega de forma ordenada das tramas ao nível superior. Caso a trama esteja fora de ordem, é então guardada numa outra fila, até que as tramas em falta sejam recebidas.

4.1.1.4 Métodos auxiliares

Para além dos métodos já referidos, foram implementados outros que servem como suporte ao protocolo TPAP. Além disto, também houve necessidade de alterar outros módulos, como o *WifiHelper*, de forma a permitir instalar o *WifiTPAP* nos nós. Era através do *WifiHelper* que era realizada a instalação do *WifiNetDevice* nos nós, através do método *Install*, que foi alterado para permitir a instalação do *WifiTPAP* ou do *WifiNetDevice*, conforme pretendido.

Foi implementado no *WifiTPAP* um método para definir o tamanho da janela deslizante e, de forma a este método poder ser invocado através do ficheiro de testes, foi necessário implementar no módulo *WifiHelper* um método que receba o tamanho da janela e que, quando realiza a instalação do *WifiTPAP* nos nós, passe para o método respetivo do *WifiTPAP* o tamanho da janela definido.

De referir ainda dois métodos que foram criados no *WifiTPAP*, um para a EB e outro para as ECs. No primeiro caso, para que a EB saiba o endereço MAC das ECs que estão associadas, permitindo também inicializar todas as variáveis necessárias, por exemplo, a fila onde são guardadas as tramas recebidas do nível superior, uma por destino. O outro método tem o mesmo objetivo, mas é aplicado às ECs, sendo necessário para que cada uma saiba o endereço MAC da EB à qual estão associadas, permitindo realizar as inicializações necessárias.

Para o funcionamento destes métodos foi necessário alterar no “MAC HIGH”. No caso da EB, foi alterado o *ApWifiMac*, para que quando recebe a confirmação que o *Association Response* foi corretamente enviado, notifique o *WifiTPAP* do endereço MAC da EC para o qual o *Association Response* foi enviado. No caso das ECs, foi alterado *StaWifiMac* que, quando recebe a trama *Association Response*, notifica o método do *WifiTPAP*, informando o endereço MAC da EB à qual está associado.

Outros métodos implementados no *WifiTPAP* e que já foram implicitamente referidos anteriormente são o método para introduzir o número de sequência nas tramas, os métodos implementados para gerir a fila de espera, quer a das tramas que se recebem do nível superior para enviar para o nível inferior, quer das tramas que são recebidas fora de ordem e portanto necessitam de ser armazenadas antes de serem passadas ao nível superior, bem como os métodos para gerir a rotação da janela.

Por fim, e para o caso do TPAP completo, tiveram de ser alterados os módulos do ns-3 de forma a proceder-se à remoção dos ACKs de nível MAC que, como já referido, não estão incluídos no funcionamento do protocolo TPAP.

4.1.1.5 Extração de dados do *WifiTPAP*

O ns-3 possui mecanismos que permitem a extração de resultados de simulação, entre eles o *FlowMonitor*; no entanto, dadas as características específicas do protocolo TPAP, há parâmetros que é necessário monitorizar e que as ferramentas disponíveis não permitem, como por exemplo o tempo de ciclo *polling*, ou no caso da existência de perdas, o número de tramas POLL, RET ou BACK perdidas.

Para ultrapassar este problema foi implementada no *WifiTPAP* uma estrutura de forma a extrair os resultados pretendidos por fluxo de ligação, como se mostra na Figura 4.4.

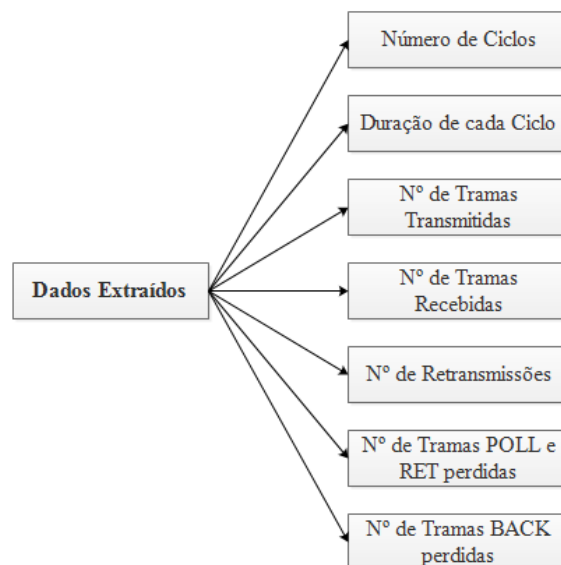


Figura 4.4 - Dados extraídos do módulo *WifiTPAP*

Para que estes dados possam ser acedidos através do ficheiro de testes foi necessário criar um *callback*.

4.1.1.6 Criação de Erros

Para testar o protocolo TPAP na presença de erros foi introduzido no *WifiTPAP*, nos métodos responsáveis pela receção das tramas de dados, das tramas POLL e de RET, e das tramas BACK, uma taxa de erros e uma variável gerada aleatoriamente entre 0 e 1.

Quando a variável aleatória é inferior à taxa de erros assume-se que ocorre erro, por exemplo, no caso da receção da trama POLL levará a que a EC esteja um ciclo sem transmitir, e por outro lado levará ao expirar do temporizador (*time-out*) que é ativado aquando do envio da trama POLL, uma vez que a trama RET não será gerada pela EC. Caso a variável aleatória seja superior à taxa de erros então assume-se a correta receção da trama.

A criação de erros reveste-se de elevada importância, dado que numa implementação real o protocolo TPAP estará inevitavelmente sujeito à ocorrência de erros de transmissão. Por este motivo é necessário analisar o seu comportamento na presença de erros de forma a poder aferir-se o respetivo impacto no seu funcionamento (validação funcional do mecanismo de confirmação e retransmissão seletiva) e no seu desempenho (face ao previsto na teoria para protocolos do tipo *Selective Repeat*).

4.1.1.7 Protocolo TPAP com mecanismo de agregação e BACK de nível MAC

Como já mencionado, o protocolo TPAP completo é útil para os *standards* IEEE 802.11 que não possuem os mecanismos de agregação de tramas nem de confirmação de tramas em bloco, como por exemplo o *standard* IEEE 802.11a.

Contudo, e dado que o ns-3 possui implementação destes mecanismos no nível MAC (no caso, de acordo com o *standard* IEEE 802.11n), foi decidido implementar e simular também a versão simples do TPAP de forma a efetuar-se uma comparação do desempenho das duas versões sobre IEEE 802.11n (versão completa em condições idênticas às relativas a IEEE 802.11a, isto é, inibindo a agregação de tramas e ACKs no nível MAC, e versão simples com agregação de tramas e BACKs no nível MAC).

Para a implementação do TPAP simples foram removidos do TPAP inicialmente implementado os mecanismos de *Selective Repeat* e de rotação da janela (isto é, não são geradas tramas BACK), mantendo-se no entanto a janela de forma a limitar o número de tramas que são passadas ao nível inferior, bem como o mecanismo de *polling*, de forma a disciplinar o acesso ao meio.

Os resultados obtidos para as duas versões do protocolo TPAP serão discutidas no capítulo dedicado à simulação dos vários cenários e à análise dos respetivos resultados.

4.1.1.8 Ficheiro de Testes

De forma a poder testar-se o protocolo TPAP implementado, foi necessário criar um ficheiro de testes onde foi configurada uma topologia ponto-multiponto, por corresponder ao cenário alvo para o protocolo TPAP na implementação real. Foram atribuídos os modelos de perdas de propagação e de atraso de propagação, e foi ainda criado um módulo de

mobilidade para cada nó. Por fim foram criados os geradores de tráfego necessários, neste caso, um por EC e, no caso da EB, um número de geradores igual ao número de ECs.

É também neste ficheiro que se define o tipo de *standard* em uso, bem como a utilização ou não do mecanismo RTS/CTS (importante no caso em que o protocolo TPAP é removido, uma vez que em qualquer das versões do TPAP, o mecanismo RTS/CTS é inibido), de fragmentação e de agregação de tramas e confirmação em bloco das mesmas.

Como as simulações têm como objetivo avaliar o comportamento do protocolo, é necessário variar alguns parâmetros, sendo portanto definidos de forma a poder-se proceder à sua alteração de forma simples, nomeadamente o módulo de numeração que servirá para calcular o tamanho da janela deslizante, o número de ECs, o tamanho dos pacotes de dados a criar pelos geradores de tráfego, a distância entre as ECs e a EB e o débito a usar por gerador de tráfego (tanto simétrico como assimétrico).

Por fim foi ainda criado um ficheiro de saída, com extensão *.csv*, compatível com Excel, onde são escritos os resultados obtidos da simulação através do *callback* criado no *WifiTPAP*, de forma a obter-se os resultados organizados por fluxo e a poder realizar-se a contabilização de resultados finais como por exemplo o cálculo do *goodput* ou a ocupação média da janela de transmissão.

Capítulo 5

Simulação de cenários de teste e análise de resultados

No presente capítulo serão apresentados os resultados obtidos através de um vasto conjunto de simulações efetuadas considerando diferentes cenários de estudo.

O objetivo genérico passa pela caracterização do protocolo TPAP sobre canais IEEE 802.11, considerando duas variantes, uma que não suporta agregação, o *standard* IEEE 802.11a, e outra que suporta o mecanismo de agregação, o *standard* IEEE 802.11n.

A caracterização do protocolo TPAP, em termos específicos, pode ser dividida em três objetivos:

- Avaliação do protocolo TPAP completo sobre as duas variantes IEEE 802.11 (inibindo a agregação no caso de IEEE 802.11n), residindo a diferença nos débitos binários possíveis; por um lado, é possível avaliar o impacto do aumento do débito binário do canal na componente do *overhead* atribuível ao protocolo TPAP e por outro usar os resultados relativos ao IEEE 802.11n para comparação com a versão simplificada do protocolo;
- Avaliação da versão simplificada do protocolo TPAP sobre IEEE 802.11n, com agregação de tramas e confirmação de blocos de tramas na camada MAC, o que permite avaliar os ganhos da agregação e eventuais desvantagens do *overhead* associado ao mecanismo ACK/BACK na camada MAC;
- Comparação dos cenários anteriores com o envio de pacotes de dados diretamente sobre MAC IEEE 802.11, considerando os três cenários referidos, ou seja, usando quer o *standard* IEEE 802.11a quer o *standard* 802.11n com e sem agregação.

Relativamente às simulações efetuadas, de referir que foram realizadas inicialmente simulações do protocolo TPAP sobre IEEE 802.11a a 54 Mbit/s e apenas posteriormente foi possível realizar simulações sobre IEEE 802.11n a 65 Mbit/s, através da utilização de uma nova versão do ns-3, designada ns-3.23.

Os testes iniciais realizados sobre IEEE 802.11a foram mais extensivos, cobrindo múltiplos aspetos com vista a uma caracterização do protocolo em diferentes condições e à validação do modelo teórico, referidos no Capítulo 3. A confirmação dos resultados teóricos dispensou uma simulação tão exaustiva sobre IEEE 802.11n, sendo neste caso o foco colocado na questão da agregação.

Em ambas as versões do protocolo TPAP a troca de tramas RTS/CTS foi inibida, uma vez que a utilização de tramas de controlo (POLL e RET) evita colisões entre tramas TPAP. Contudo, nos casos em que, para efeito de comparação, o protocolo TPAP não foi implementado, foram realizadas simulações com ativação e inibição de RTS/CTS.

É ainda de referir que o protocolo TPAP completo implica a utilização de tramas BACK, para além de tramas POLL e RET, ao passo que as tramas BACK não são usadas na versão simplificada; no entanto nesta versão as tramas POLL e RET são confirmadas na camada MAC. O impacto destas diferenças será evidenciado nos resultados de simulação relativos ao tempo mínimo de um ciclo de *polling* (em que não são transmitidas tramas de Dados).

De modo a sistematizar a análise do protocolo e a avaliação do impacto da variação de parâmetros críticos ou das condições de teste, bem como para facilitar a comparação dos vários casos em estudo, a apresentação é organizada em três grupos, tendo em atenção a variante 802.11 em uso e as opções de agregação possíveis.

Todas as simulações realizadas envolvem uma estação base e cinco estações clientes, o que resulta em dez estações lógicas e consequentemente dez ligações unidirecionais, a que corresponde dez geradores de tráfego independentes. A interação entre a estação base e uma estação cliente envolve duas ligações unidirecionais, ocorrendo o *turnaround* com o envio da trama de POLL e o fim da interação com o envio da trama de RET.

Os resultados obtidos correspondem a 19 segundos de tempo real, antecidos de 3 segundos para estabilização de valores. Durante os 3 segundos iniciais não é realizada qualquer contabilização de resultados.

5.1 Resultados obtidos com o *standard* IEEE 802.11a

Nesta secção serão apresentados os resultados obtidos usando o *standard* IEEE 802.11a, que não possui suporte ao mecanismo de agregação e de *block acknowledgment* na camada MAC, sendo portanto utilizada a versão completa do protocolo TPAP, considerando diferentes cenários.

Como termo de comparação serão ainda apresentados resultados obtidos sem o protocolo TPAP, com o envio de tramas diretamente sobre a camada MAC, considerando a ativação e a inativação do mecanismo opcional de RTS/CTS.

Por fim será efetuada a comparação entre as duas soluções apresentadas.

5.1.1 TPAP completo com parâmetros MAC *default*

Como primeiro cenário para a caracterização do protocolo TPAP completo, para além das considerações já mencionadas, considerou-se a utilização dos parâmetros MAC com os valores *default*, nomeadamente, não houve alteração ao nível dos intervalos de tempo entre tramas.

A tabela A.1 do anexo A apresenta os resultados obtidos a partir das simulações realizadas.

Relativamente às colunas da tabela é de referir que a primeira se refere ao débito dos geradores de tráfego de cada estação lógica, a segunda diz respeito ao *goodput* total obtido a partir da soma dos *goodput* individuais por fluxo. A coluna referente à ocupação da janela representa o número médio de tramas enviadas por ciclo e por fluxo, enquanto a coluna referente aos pacotes transmitidos corresponde ao número total de pacotes transmitidos, em média, por cada fluxo. Por fim, a coluna relativa ao ciclo corresponde ao tempo médio de ciclo de *polling*, o número de ciclos corresponde ao número médio de ciclos por estação lógica e n representa o número total de tramas de dados enviadas num ciclo de *polling* (o valor de α obtém-se dividindo n por 160).

Os resultados foram obtidos considerando tamanho de pacotes de 1500 bytes e uma janela de transmissão, W , de tamanho 16, que consideraremos como valores *default*, sendo posteriormente considerados valores diferentes para efeito de comparação. As figuras seguintes ilustram o comportamento do protocolo, que será analisado a seguir.

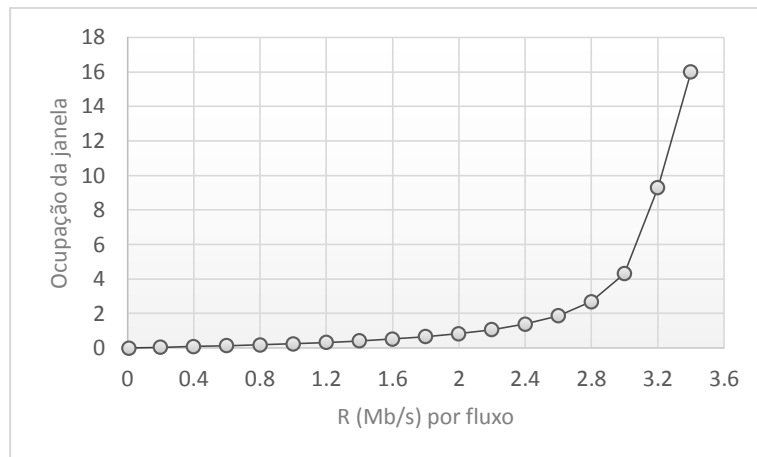


Figura 5.1 - Variação da ocupação da janela com o aumento do débito por fluxo

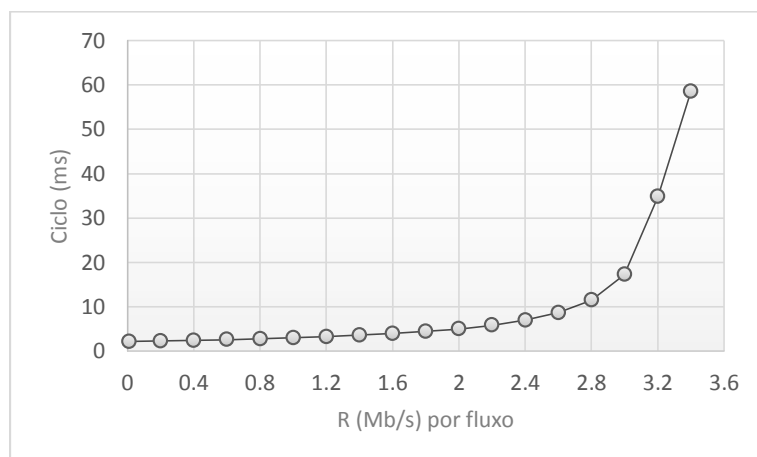


Figura 5.2 - Variação do tempo de ciclo com o aumento do débito por fluxo

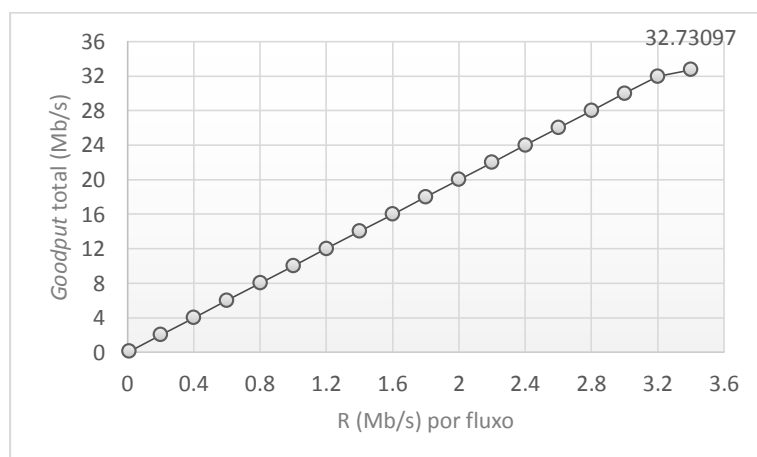


Figura 5.3 - Variação do *goodput* total com o aumento do débito por fluxo

Da análise das Figuras 5.1 e 5.2 conclui-se que inicialmente um crescimento rápido de R se traduz em pequenos aumentos da ocupação da janela e do tempo de ciclo, contrastando com os valores próximos da saturação (isto é, ocupação plena da janela) onde pequenas

variações de R dão origem a grandes variações quer do valor da ocupação da janela, quer do tempo de ciclo, que apresentam comportamentos semelhantes.

Da Figura 5.3 pode-se concluir que para débitos por fluxo até aos 3.2Mbit/s, o *goodput* obtido corresponde à soma dos débitos individuais, ou seja, considerando por exemplo o caso do débito individual de 2.0Mbit/s, o débito total gerado pelas dez estações lógicas é igual a 20Mbit/s, o que corresponde aproximadamente ao *goodput* total obtido. A saturação ocorre para um débito individual de cerca de 3,27Mbit/s a que corresponde um *goodput* igual a 32,73Mbit/s; para valores superiores deixa de ser possível transmitir todos os pacotes gerados, por força do controlo de fluxo. O aumento do tamanho da janela permitiria aumentar o *goodput*, mas a Figura 5.1 indicia que os ganhos seriam marginais (como se comprovará), dado o crescimento rápido da ocupação da janela para valores de R elevados. A mesma conclusão se pode tirar da Figura 5.2, que evidencia ainda que a fixação do tamanho da janela implica um compromisso entre *goodput* e atraso no envio de pacotes (estando este naturalmente relacionado com o tempo de ciclo).

5.1.1.1 Validação do modelo teórico

O modelo teórico apresentado no Capítulo 3 prevê um crescimento linear do tempo de ciclo T com o parâmetro n (ou, analogamente, com a). Sendo válida esta aproximação, como se verificará a seguir, os parâmetros do modelo poderão ser obtidos por meio de uma regressão linear sobre os pontos da função $T(n)$.

Serão apresentados de seguida todos os passos utilizados para a obtenção dos parâmetros, sendo que em futuras análises apenas serão apresentados os resultados obtidos, dado que o método utilizado é idêntico ao descrito a seguir.

I. Obtenção dos valores de n e α

O parâmetro n é obtido multiplicando o valor da coluna “Ocupação da janela” da tabela de resultados, neste caso a tabela A.1 do anexo A, pelo número de estações lógicas, neste caso dez.

Posteriormente obtém-se o parâmetro a através da equação 3.2 apresentada no Capítulo 3, na secção referente ao modelo teórico; neste caso $\alpha = n/160$, pois $n_{max} = 160$.

II. Obtenção do valor de T_p

O cálculo de T_p foi realizado efetuando a regressão linear sobre os pares de valores (T, n) obtidos por simulação (tabela A.1). De facto a Figura 5.4, que representa esses pontos, comprova o crescimento linear do tempo de ciclo T com n .

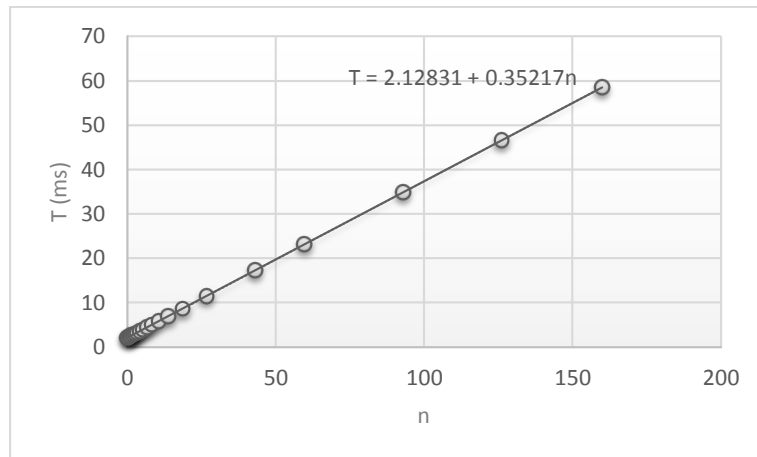


Figura 5.4 - Figura obtida através do método de regressão linear, considerando o cenário com o *standard* IEEE 802.11a, com *gaps*

O tempo mínimo de ciclo ocorre quando não há transferência de dados, mas apenas tramas de controlo, e representa por isso o *overhead* associado ao protocolo TPAP, O_{TPAP} . Uma vez que este caso foi simulado ($R=0$), para efeito da regressão linear fixou-se o ponto da reta que coincide com o eixo dos yy , isto é, $T = 2.12831\text{ms}$ e $n = 0$.

A equação da reta obtida é $T = 2.12831 + 0.35217n$ e constitui a aproximação à equação 3.6 do modelo teórico, o que permite obter diretamente o tempo médio de ocupação do meio com um pacote de dados $T_p = 0.35217\text{ ms}$.

III. Obtenção do valor de T_o

Uma vez obtido o valor de T_p , e sabendo que $T_d = \frac{\text{tamanho do pacote}}{C_c}$, onde o tamanho do pacote corresponde a 1500 bytes, ou seja, 12000 bits, $C_c = 54\text{Mbit/s}$, pelo que T_d é aproximadamente igual a 0.222 ms. Através da equação 3.4 obtemos T_o aproximadamente igual a 0.13 ms.

IV. Obtenção do parâmetro k

A obtenção do parâmetro k é realizada recorrendo à equação 3.7, obtendo-se neste caso o valor aproximado de 0.038.

V. Obtenção dos parâmetros C , S_{max} e R_{max}

O parâmetro C pode ser obtido através do parâmetro T_p e considerando o tamanho dos pacotes de dados utilizado, sendo o seu valor de 34.07Mbit/s. O valor de S_{max} é obtido pela equação 3.11, sendo o seu valor de 96.36%, resultando num R_{max} obtido através da expressão 3.12, igual a 32.83Mbit/s.

T_d (ms)	T_p (ms)	T_o (ms)	O_{TPAP} (ms)	k	C (Mb/s)	S_{max} (%)	R_{max} (Mb/s)
0.222	0.352	0.13	2.128311	0.038	34.07	96.36	32.83

Tabela 5.1 - Valor dos parâmetros obtidos para o modelo teórico considerando o TPAP completo e *standard* IEEE 802.11a, com parâmetros MAC *default*

Relativamente aos resultados obtidos com o modelo teórico podemos concluir que se trata de uma boa aproximação, já que o valor obtido para o R_{max} é muito próximo do valor obtido através das simulações realizadas para o *goodput* máximo (32.731 Mbit/s).

Podemos concluir que a degradação de desempenho se deve essencialmente ao *overhead* associado às camadas MAC/PHY (expresso pela relação $T_d/T_p=0.63$), dado que o valor de k é muito pequeno, refletindo assim a necessidade de atuar ao nível das camadas inferiores.

5.1.2 TPAP completo com eliminação de *gaps* no nível MAC

Como segundo cenário de simulação, e com o objetivo de reduzir o *overhead* associado à camada MAC, decidiu-se eliminar os intervalos de guarda entre tramas sucessivas, dado que num acesso ao meio controlado por *polling* não existe risco de colisão, pelo que estes intervalos não são necessários. Contudo foi mantida a janela de contenção com o valor de $CW_{min} = 15$, mas com *slot time* igual a $1\mu s$, por ser impossível eliminá-lo.

Os resultados obtidos são apresentados na tabela A.2 do anexo A e a comparação dos dois cenários é apresentada na Figura 5.5 (tempos médios de ciclo) e na Figura 5.6 (ocupação média da janela), para diferentes valores de R e na Figura 5.7 (*goodput* total máximo).

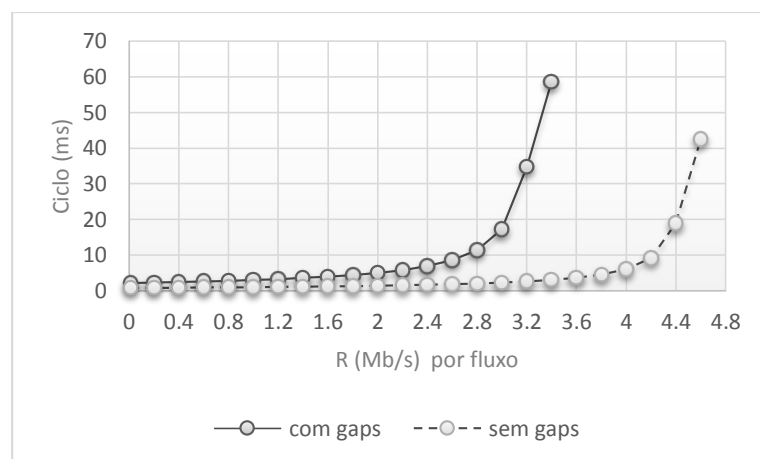


Figura 5.5 - Comparação dos tempos de ciclo para os cenários com e sem *gaps*

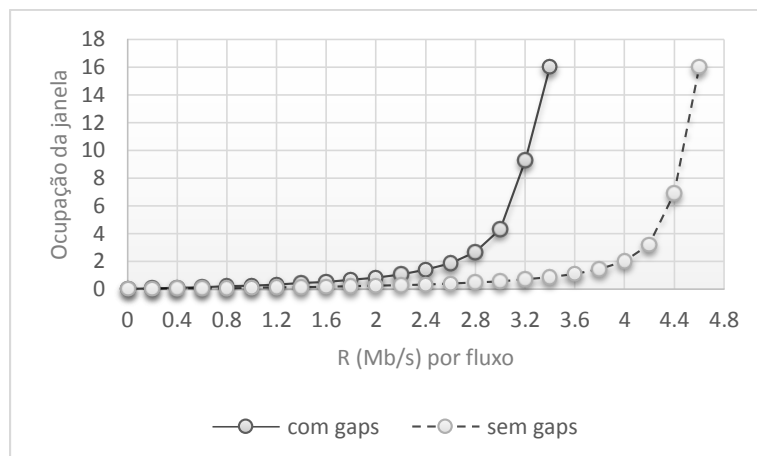


Figura 5.6 - Comparação da ocupação da janela para os cenários com e sem *gaps*

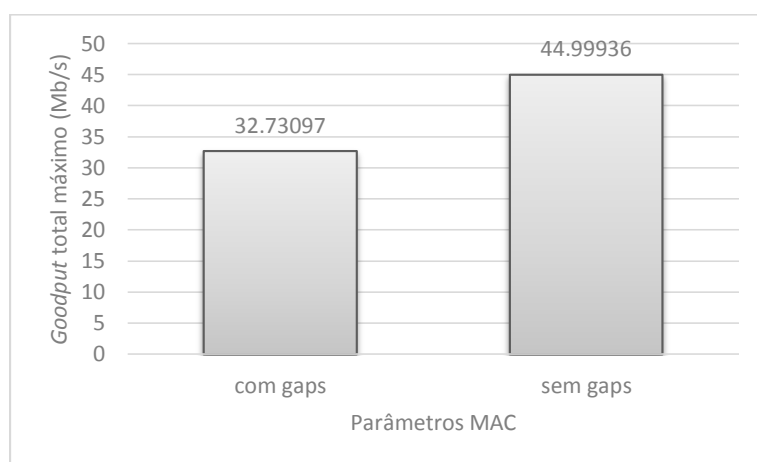


Figura 5.7 - Comparação do *goodput* total máximo obtido para os cenários com e sem *gaps*

Comparando o cenário atual com o discutido na secção 5.1.1, há várias conclusões que se podem retirar. Por um lado há uma redução nos tempos de ciclo, ilustrada na Figura 5.5, devido à inibição dos intervalos entre tramas no nível MAC; por outro lado, esta diminuição do tempo de ciclo origina uma menor ocupação média da janela de transmissão, como ilustrado na Figura 5.6, e um *goodput* máximo superior (Figura 5.7), aproximadamente 45Mbit/s em vez de 32.7Mbit/s alcançados no cenário anterior.

A análise realizada na secção 5.1.1 mantém-se no que diz respeito ao crescimento de *R* com o aumento da ocupação da janela.

5.1.2.1 Validação do modelo teórico

Para o cenário de simulação sem *gaps*, a aproximação linear continua a ser válida, conforme se ilustra na Figura 5.8.

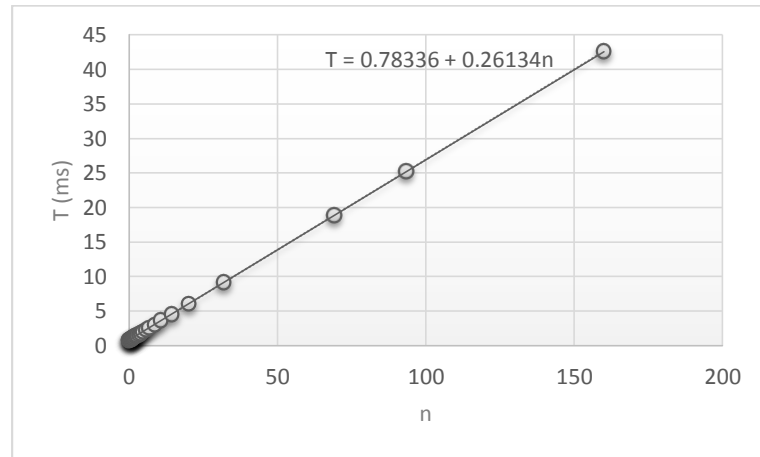


Figura 5.8 - Figura obtida através do método de regressão linear, considerando o cenário com o *standard* IEEE 802.11a, sem *gaps*

Com base no resultado da regressão linear, derivaram-se os valores relativos ao modelo teórico, que são apresentados na tabela 5.2.

<i>Gaps</i>	T_d (ms)	T_p (ms)	T_o (ms)	O_{TPAP} (ms)	k	C (Mb/s)	S_{max} (%)	R_{max} (Mb/s)
Sim	0.222	0.352	0.13	2.128311	0.038	34.07	96.36	32.83
Não	0.222	0.261	0.039	0.783358	0.019	45.92	98.16	45.07

Tabela 5.2 - Valor dos parâmetros obtidos para o modelo teórico considerando o TPAP completo e *standard* IEEE 802.11a, com eliminação de *gaps* no nível MAC

Em comparação com a situação com *gaps*, vemos uma diminuição no valor do tempo total de ocupação do meio com a transmissão de uma trama de dados, T_p , dado que houve uma redução no *overhead* associado à camada MAC, T_o , atribuída à eliminação dos *gaps*. A redução do valor da relação T_d/T_p (agora igual a 0.85) traduz-se numa maior capacidade útil disponível para o protocolo TPAP (45.92 em vez de 34.07 Mbit/s).

Por outro lado, também o valor de k diminuiu, comparativamente à situação com *gaps*, o que tem como consequência um aumento (ainda que pequeno) da eficiência máxima. Embora a diminuição de T_p induza um aumento de k , esse aumento é compensado com a diminuição mais significativa de O_{TPAP} , uma vez que, como já referido, o *overhead* associado ao protocolo TPAP depende de fatores alheios, entre eles o *overhead* das camadas inferiores. Ao reduzir-se o *overhead* associado à camada MAC, reduz-se por sua vez o *overhead* associado ao protocolo TPAP. A degradação de desempenho, embora menos acentuada, continua a dever-se maioritariamente à camada MAC/PHY, pelo que se pode concluir que é ao nível das camadas inferiores que se deve intervir para melhorar o desempenho, nomeadamente com exploração da agregação a nível MAC (que permite reduzir o *overhead* da camada PHY).

A eficiência máxima, S_{max} , é igual a 98.16%, conduzindo a um R_{max} de 45.07Mbit/s, muito próximo do valor obtido, através das simulações, para o *goodput* máximo (44.999Mbit/s).

5.1.3 TPAP completo com variação do tamanho dos pacotes (P) e do tamanho da janela (W)

Nesta secção procura-se aferir o impacto da variação do tamanho dos pacotes de dados e do tamanho da janela de transmissão no desempenho global do protocolo TPAP e nos parâmetros medidos pelo modelo teórico, nomeadamente no que diz respeito quer ao *overhead* associado ao protocolo TPAP quer ao *overhead* associado às camadas inferiores.

Relativamente à camada MAC, manteve-se a inibição dos *gaps*, pelo que a comparação deve ser feita com o segundo cenário anterior, com $P = 1500$ bytes e $W = 16$.

Foram considerados os seguintes cenários para análise:

- $P = 800$ bytes e $W = 16$;
- $P = 1500$ bytes e $W = 8$;
- $P = 800$ bytes e $W = 8$;
- $P = 1500$ bytes e $W = 32$;
- $P = 800$ bytes e $W = 32$.

Os resultados são apresentados nas tabelas A.3 a A.7 do anexo A e nas figuras seguintes. A validação do modelo teórico foi também feita, mas é omitida por não acrescentar valor.

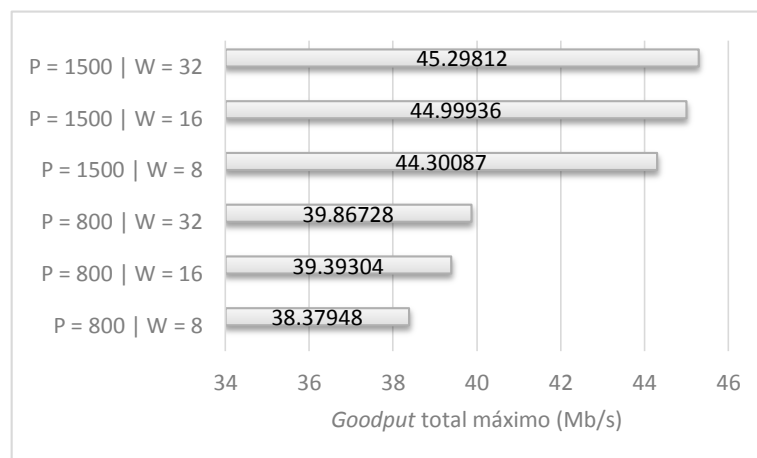


Figura 5.9 - Variação do *goodput* máximo com a variação do tamanho dos pacotes e da janela

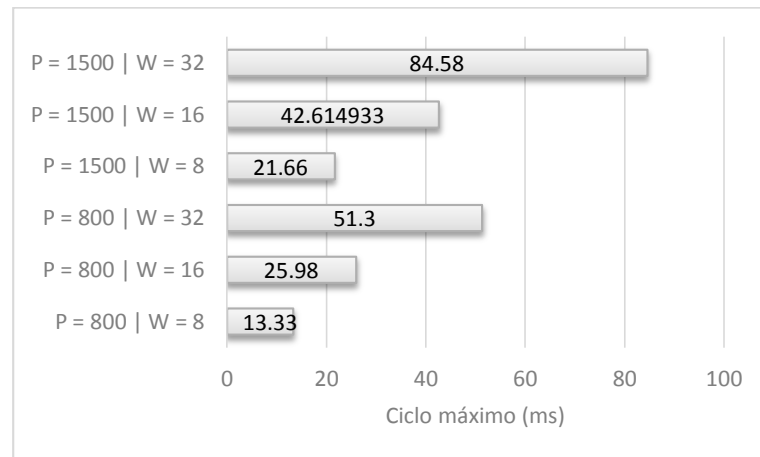


Figura 5.10 - Variação do tempo de ciclo máximo com a variação do tamanho dos pacotes e da janela

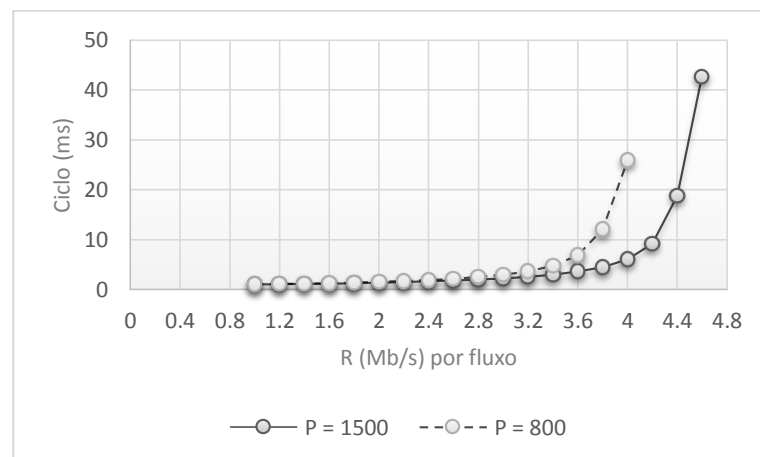


Figura 5.11 - Variação do tempo de ciclo considerando tamanho de pacotes diferentes, com $W = 16$

Os resultados ilustrados na Figura 5.9 permitem concluir que, para o mesmo tamanho de pacotes, o aumento do tamanho da janela permite ganhos no *goodput* máximo que são, no entanto, marginais. Esta conclusão já tinha sido antecipada quando se analisou a Figura 5.1 e é agora comprovada.

Por outro lado, a diminuição do tamanho dos pacotes (neste caso de 1500 para 800 bytes) tem um impacto significativo na diminuição do *goodput* máximo (um valor médio de 800 bytes pode representar uma situação em que coexistem pacotes de comprimento máximo e pacotes muito pequenos, como é o caso de ACKs TCP). Ao reduzir o tamanho dos pacotes aumenta o *overhead* relativo das camadas PHY e MAC (diminuição de C) e também do protocolo TPAP (aumento de k), o que explica a diminuição do *goodput* máximo pelo efeito cumulativo dos dois fatores.

Igualmente, como representado na Figura 5.10, com o aumento do tamanho da janela ocorre o aumento do tempo de ciclo máximo, como é natural, dado que é possível transmitir um maior número de pacotes para o valor de R que permite atingir o *goodput* máximo.

A Figura 5.11 mostra de forma mais detalhada a variação do tempo de ciclo para vários valores de débito R por fluxo, para o caso $W = 16$ e os dois valores de tamanho dos pacotes. Embora o tempo máximo de ciclo seja superior para $P = 1500$ bytes (42,61 vs. 25,98 ms), para o mesmo valor de R o tempo de ciclo para $P = 800$ bytes é maior que para $P = 1500$ bytes. De notar que no primeiro caso são gerados quase duas vezes mais pacotes (de facto 1,875 vezes) e uma vez que para a mesma quantidade de informação útil o *overhead* é maior, conforme referido, o tempo de ciclo é estendido (de forma equivalente, e pela mesma razão, o aumento relativo da ocupação média da janela é superior àquele fator). Por outro lado, ao comparar-se os tempos máximos de ciclo, deve ter-se em atenção que correspondem a valores de R , ou seja, de *goodput* máximo, diferentes (44,999 vs. 39,393 Mbit/s), pelo que tal não contradiz o que se verifica para cada valor particular de R .

5.1.4 Tráfego Assimétrico

Nesta secção pretende-se avaliar o comportamento do protocolo quando o tráfego é assimétrico; ao contrário dos cenários considerados até agora em que todas as estações lógicas geravam fluxos com o mesmo débito, neste caso haverá um ou mais fluxos em que o débito gerado será superior ao dos restantes, mantendo-se fixo o débito destes e aumentando o débito dos primeiros até se atingir a saturação.

As situações analisadas estão divididas em três partes:

- Na primeira situação começamos com nove fluxos com um débito baixo, de 1Mbit/s, e um fluxo com débito muito alto, de 28Mbit/s, aumentando-se a seguir o débito deste fluxo até à saturação. De seguida diminuiu-se o número de fluxos com baixo débito e aumentou-se o número de fluxos com débito alto, aferindo-se o respetivo comportamento na zona de saturação;
- Num segundo cenário procurou-se analisar o comportamento considerando fluxos com débito médio fixo, mantendo-se os fluxos com débito alto variável até se atingir a saturação;
- No último cenário analisado todas as estações clientes foram configuradas para gerar o mesmo débito, enquanto todas as estações lógicas na estação base foram também configuradas com o mesmo débito, mas diferente do débito das estações clientes, pelo que o tráfego em cada ligação bidirecional é assimétrico.

Os resultados obtidos são apresentados nas tabelas A.8, A.9 e A.10 do anexo A.

Analisando a primeira situação conclui-se que, apesar de inicialmente existir uma estação com débito muito superior ao das restantes, o resultado obtido para o *goodput* total se mantém igual à soma dos débitos gerados pelos diferentes fluxos, até que a estação com débito alto sature.

Verifica-se igualmente que, à medida que se aumenta o débito da estação configurada com débito elevado, a ocupação da janela das estações com menor débito se vai adaptando (aumenta devido ao aumento do tempo de ciclo induzido pelo aumento do débito da estação com débito mais elevado); deste modo, as estações com menor débito garantem um *goodput* igual ao débito gerado, não sendo prejudicadas pelo aumento do débito da outra estação, mesmo quando esta atinge a saturação.

Após entrar em saturação, o *goodput* da estação atinge o valor máximo, deixando de acompanhar um eventual aumento do débito gerado. Deve assinalar-se que, nesta situação, o *goodput* total (40,094 Mbit/s) é inferior ao *goodput* máximo da situação em que o tráfego é simétrico (44,999 Mbit/s). Isto deve-se ao facto de as estações com menor débito não atingirem a saturação (não usam todos os créditos disponíveis), pelo que a eficiência global do protocolo é menor. No entanto, a estação com débito elevado obtém um *goodput* máximo superior ao que obteria no caso de tráfego simétrico, pois beneficia do menor débito das restantes estações (tempo de ciclo menor). Estes resultados constituem uma base para a implementação de um mecanismo de *smart polling* baseado numa gestão dinâmica do tamanho da janela (créditos); à estação saturada poderia ser concedido um maior número de créditos (aumento da janela), isto é, reutilizaria créditos não usados pelas estações com menor débito que, como se comprova pelos resultados, não são capazes de usar todos os créditos que têm disponíveis. Tal pode ser feito sem comprometer o tempo máximo de ciclo e na prática isto traduz-se num aumento da eficiência do protocolo.

A segunda situação analisada tem um comportamento em tudo semelhante ao cenário anterior mantendo-se a mesma análise. De referir ainda que, com o aumento do número de estações com tráfego elevado, na situação de saturação destas o *goodput* total aumenta, uma vez que a eficiência do protocolo também aumenta (é menor o número de estações que não utilizam a totalidade dos créditos).

Quanto ao terceiro cenário em que o tráfego é assimétrico nas ligações bidirecionais, pode concluir-se que o *goodput* total é igual à soma dos débitos gerados, enquanto nenhum fluxo saturar, garantindo-se os débitos dos fluxos não saturados. Tal como nos casos anteriores o débito de um fluxo não saturado não é afetado pela eventual saturação de outros fluxos.

5.1.4.1 Validação do modelo teórico

Dado que as condições utilizadas para estes cenários são as da secção 5.1.2, os valores obtidos para os parâmetros do modelo teórico mantêm-se, nomeadamente o parâmetro k e o parâmetro C que têm, respetivamente, os valores 0.018734 e 45.9172 Mbit/s.

Pretende-se aqui validar a equação 3.13, e os resultados obtidos para R_0 são apresentados na tabela 5.3.

Cenário		R_o (total) (Mbit/s)	Resultados de Simulação (Mbit/s)
1	9 Estações a 1 Mbit/s e 1 Estação com alto débito	31.09	31.096
	8 Estações a 1 Mbit/s e 2 Estações com alto débito	34.67	34.657
	7 Estações a 1 Mbit/s e 3 Estações com alto débito	36.63	36.612
2	9 Estações a 3 Mbit/s e 1 Estação com alto débito	15.93	15.942
	8 Estações a 3 Mbit/s e 2 Estações com alto débito	20.04	20.039
	7 Estações a 3 Mbit/s e 3 Estações com alto débito	23.45	23.434
	6 Estações a 3 Mbit/s e 4 Estações com alto débito	26.67	26.649
	5 Estações a 3 Mbit/s e 5 Estações com alto débito	29.80	29.773
3	Tráfego assimétrico entre Estações Clientes e Estação Base	24.98	24.96

Tabela 5.3 - Valores dos parâmetros do modelo teórico considerando tráfego assimétrico

Os valores obtidos para o parâmetro R_o , que representa o débito global das estações saturadas, estão muito próximos dos obtidos através dos resultados de simulação.

5.1.5 Introdução de atrasos de processamento

Com o objetivo de avaliar o impacto da introdução de atrasos de processamento (presentes numa implementação real) associados à execução do protocolo TPAP, que até ao momento não foram considerados, dado que no ns-3 o tempo de processamento é nulo, foram realizados testes em três cenários diferentes. Foram considerados diferentes tempos de atraso de processamento com o objetivo de analisar as variações no *goodput* total, na ocupação da janela, nos tempos de ciclo e por fim nos parâmetros do modelo teórico.

Nos cenários implementados foram considerados atrasos de 0.1ms, 0.5ms e 1ms.

Os resultados obtidos são apresentados nas tabelas A.11, A.12 e A.13 do anexo A e a análise comparativa é feita com base nas figuras seguintes que sintetizam os resultados.

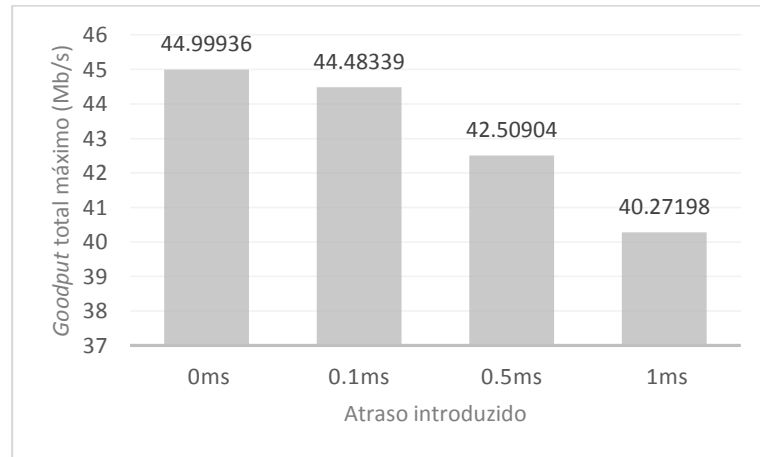


Figura 5.12 - Variação do *goodput* total máximo com o aumento do atraso de processamento introduzido

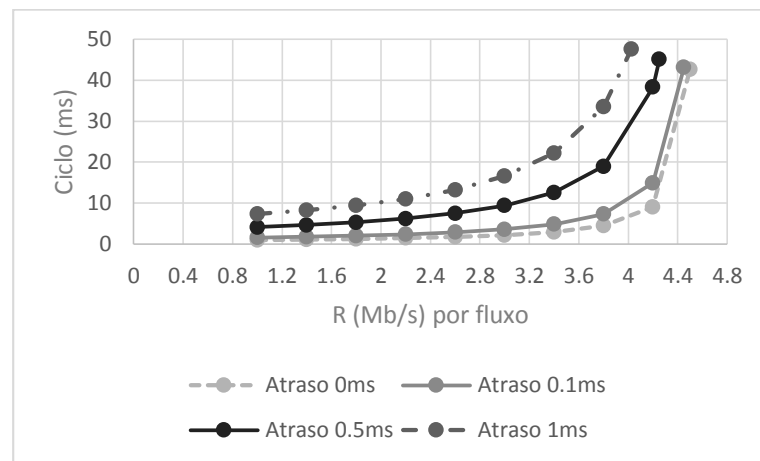


Figura 5.13 - Variação do tempo de ciclo com o aumento do atraso de processamento introduzido

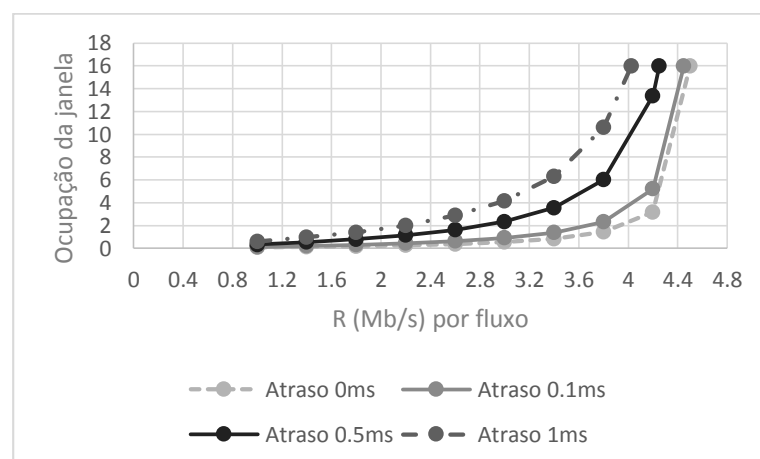


Figura 5.14 - Variação da ocupação da janela com o aumento do atraso de processamento introduzido

O tempo de atraso de 0ms apresentado nas figuras refere-se ao cenário analisado na secção 5.1.2, para $P = 1500$ e $W = 16$, com eliminação dos *gaps* de nível MAC, mas sem atrasos de processamento.

Da Figura 5.12 conclui-se que, com a introdução do atraso de processamento, e à medida que este vai aumentando, o *goodput* total máximo vai diminuindo, e embora para o atraso de 0.1 ms essa degradação seja ligeira, para os valores de 0.5 ms e 1 ms constitui uma perda de aproximadamente 5.5% e 10.5%, respetivamente.

Relativamente ao comportamento do tempo de ciclo, ilustrado na Figura 5.13, apesar de o comportamento nos quatros cenários ilustrados ser semelhante, constata-se, como seria de esperar, que à medida que o atraso introduzido é maior, o tempo de ciclo também aumenta.

Comportamento semelhante apresenta a variação da ocupação da janela, ilustrada na Figura 5.14, onde à medida que o atraso introduzido é maior, a variação da ocupação da janela também aumenta.

5.1.5.1 Validação do modelo teórico

Os parâmetros que caracterizam o modelo teórico são apresentados na tabela 5.4, com base na aproximação linear da função $T(n)$, que se mantém válida.

Atraso (ms)	T_d (ms)	T_p (ms)	T_o (ms)	O_{TPAP} (ms)	k	C (Mb/s)	S_{max} (%)	R_{max} (Mb/s)
0	0.222	0.261	0.039	0.783358	0.019	45.92	98.16	45.07
0.1	0.222	0.261	0.039	1.265219	0.030	45.89	97.06	44.55
0.5	0.222	0.261	0.039	3.265045	0.078	45.93	92.76	42.60
1	0.222	0.261	0.039	5.764792	0.138	45.90	87.89	40.34

Tabela 5.4 - Valores dos parâmetros do modelo teórico considerando atrasos de processamento

Os valores dos parâmetros estão de acordo com o esperado, ou seja, relativamente ao tempo para transmitir o *payload*, T_d , bem como ao tempo para transmitir a tramas, T_p e ao tempo de *overhead* associado às camadas MAC/PHY, T_o , não há alterações, dado que os atrasos são introduzido na camada correspondente ao protocolo TPAP e não interferem com estes parâmetros. Também a capacidade útil disponível, C , se mantém mais ou menos constante, dado que depende de T_p que não sofre alterações.

Relativamente aos parâmetros que estão diretamente relacionados com o protocolo TPAP, dada a introdução do atraso de processamento, verifica-se um aumento do *overhead* associado ao TPAP, O_{TPAP} , e consequentemente um aumento do parâmetro k refletindo um maior impacto na degradação de desempenho introduzido pelo TPAP, que se reflete na diminuição da eficiência máxima. A diminuição da eficiência máxima leva a que o débito

máximo alcançável vá diminuindo, sendo os valores obtidos pelo modelo teórico muito próximos dos obtidos através da simulação.

Dos resultados obtidos pode concluir-se que o atraso de processamento a que o protocolo TPAP estará sujeito numa implementação real é crítico, e deve ser minorado, de forma a obter-se a maior eficiência possível.

5.1.6 Introdução de erros

Dado que numa implementação real o protocolo TPAP estará inevitavelmente sujeito a erros de transmissão (eventuais colisões residuais são tratadas como erros de transmissão) foram realizados testes com introdução de erros emulados na camada correspondente ao protocolo TPAP (visto não ser possível induzir erros de forma controlada no canal físico).

Pelas razões já apontadas no Capítulo 3, foi necessária a utilização de um temporizador, que foi definido com o tempo de 10 ms, para salvaguardar o cenário da ocorrência de perda de tramas de controlo.

Os testes realizados podem dividir-se em dois cenários diferentes:

- Num primeiro cenário foram considerados erros apenas em tramas de dados e foram simuladas taxas de erros de 1% e de 10%;
- Num segundo cenário foram simulados erros, quer para tramas de dados quer para tramas de controlo, com taxas de erros de 10% e 1%, respetivamente.

Os resultados obtidos são apresentados nas tabelas A.14, A.15 e A.16 do anexo A. Nestas tabelas é introduzida uma coluna nova que diz respeito ao número médio de tramas de dados que são retransmitidas por cada estação lógica. Por outro lado, quando a taxa de erros para tramas de controlo é diferente de zero é apresentado o número total de tramas POLL e RET perdidas, bem como o número médio de tramas BACK perdidas por estação lógica. Para as tramas POLL e RET, foi contabilizada uma perda sempre que ocorreu *time-out* no temporizador, pois este indica que ou se perdeu a trama POLL ou se perdeu a trama RET. Quanto às tramas BACK perdidas, são incluídas aquelas que não foram geradas porque a trama POLL não foi recebida.

Nas figuras seguintes compara-se o comportamento dos cenários com e sem erros.

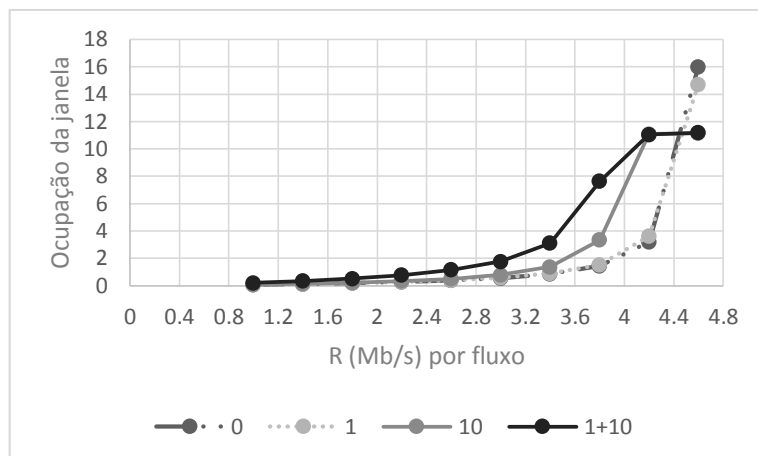


Figura 5.15 - Variação da ocupação da janela com a variação na taxa de erros

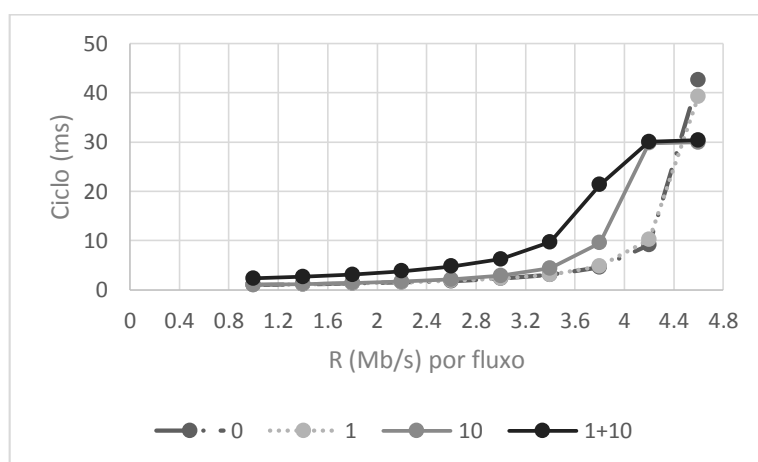


Figura 5.16 - Variação do tempo de ciclo com a variação na taxa de erros

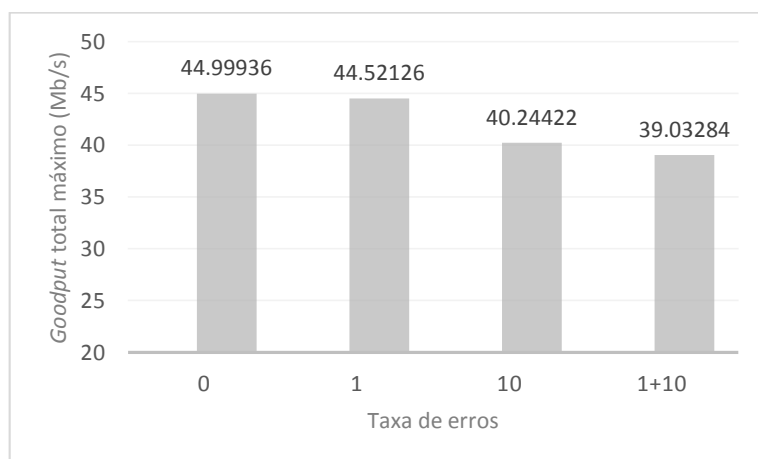


Figura 5.17 - Variação do *goodput* total máximo com a variação na taxa de erros

Nas figuras, os valores 0, 1, 10 e 1+10 representam, respetivamente, taxa de erros de 0%, taxa de erros para tramas de dados de 1%, taxa de erros para tramas de dados de 10%, e taxa de erros para tramas de controlo de 1% mais taxa de erros para tramas de dados de 10%.

Começando a análise com a taxa de erros para tramas de dados de 1%, verifica-se que, para o mesmo débito gerado, a ocupação da janela é ligeiramente superior à do caso em que

não há erros (Figura 5.15), dado que o número de retransmissões de tramas é reduzido. A saturação ocorre para um valor de ocupação da janela abaixo do valor máximo; até ocorrer a saturação, é possível garantir um *goodput* igual ao débito gerado, embora com um aumento do tempo de ciclo, devido às retransmissões.

A mesma análise pode ser realizada para o tempo de ciclo (Figura 5.16), ou seja, nesta situação o tempo de ciclo é ligeiramente superior aos valores obtidos sem erros; no entanto, na situação de saturação o valor é menor, mas tal deve-se ao facto de neste caso haver ciclos de menor duração (por rotação parcial da janela) e o *goodput* máximo ocorrer para um débito gerado inferior, ou seja, os tempos máximos de ciclo com e sem erros ocorrem para valores diferentes do débito gerado.

O *goodput* máximo (Figura 5.17) é inferior ao caso em que não há erros, embora a diminuição seja pequena (de facto um pouco acima de 1%, como seria de esperar). Tal deve-se a duas razões; por um lado devido às retransmissões (mesmo que fosse possível ocupar a janela na totalidade) e por outro porque quando é necessário retransmitir uma trama a janela roda parcialmente (no limite pode não haver sequer rotação). Uma vez que a ocupação da janela na situação de saturação é próxima do valor máximo, o impacto deste efeito é também residual.

Para a taxa de erros em tramas de dados de 10%, o aumento da ocupação da janela é mais acentuado, especialmente para débitos acima dos 3.4Mbit/s, ocorrendo a saturação com uma ocupação da janela de apenas 70% da capacidade máxima, justificado pelo aumento do número de retransmissões relativamente à situação com taxa de erros de 1%. Relativamente aos tempos de ciclo verifica-se um aumento do tempo de ciclo para débitos que não causam saturação, sendo inferior quando ocorre a saturação pelos motivos já referidos. A diminuição do *goodput* máximo é ligeiramente superior a 10% e não é mais acentuada porque a eficiência do protocolo ainda é relativamente elevada para aquele nível de ocupação, devido ao valor baixo do parâmetro k .

Para a última situação analisada, em que foi considerada uma taxa de erros para tramas de dados de 10% e uma taxa de erros para tramas de controlo de 1%, a degradação introduzida é ainda mais acentuada. A ocupação da janela tem um aumento significativo, bem como a duração do ciclo, devido ao aumento do número de retransmissões, dado que a perda de tramas BACK atrasa a rotação da janela do emissor levando à retransmissão de todas as tramas enviadas no ciclo anterior, mesmo que essa retransmissão não seja necessária. Por outro lado a perda de uma trama POLL leva a que uma estação cliente esteja um ciclo sem transmitir (e a consequente não geração da trama RET origina um *time-out*), enquanto a perda de uma trama RET origina também um *time-out* na estação base, que atrasa significativamente o início da interação seguinte.

5.1.6.1 Validação do modelo teórico

A validação do modelo teórico em cenários em que ocorrem erros foi realizada para os dois casos em que apenas se introduziu perda de tramas de dados, não sendo realizada no caso da perda de tramas de controlo, uma vez que as condições de ativação do *time-out* não são de modelização fácil.

Os valores usados para validar o modelo teórico são apresentados nas tabelas 5.5 e 5.6, respetivamente para os cenários com taxas de erros de 1% e 10% para tramas de dados. Dado que os valores dos parâmetros k e C não se alteram com a introdução de erros, foram usados, respetivamente, os valores 0.018734 e 45.9172 Mbit/s, calculados na secção 5.1.2.

Com base nos números de pacotes transmitidos e retransmitidos calculou-se a probabilidade de erros real (equação 3.15), o que permitiu calcular α'_u (equação 3.19) e R'_u (equação 3.22). Da análise dos resultados conclui-se que o débito útil previsto pelo modelo, R'_u , é aproximadamente igual ao *goodput* total, o que valida o modelo.

Enquanto a ocupação da janela não atingir o valor máximo (que corresponde à saturação) é possível transmitir as tramas geradas, embora com maior atraso. Isto significa que R'_u e o *goodput* total são aproximadamente iguais à soma dos débitos gerados por fluxo, apesar da existência de erros.

R (Mb/s) por fluxo	Sem Erros	Com erros			
	α	p_e	α'	α'_u	R'_u
1	0.00521	0.00939	0.00527	0.00523	9.99353
1.4	0.00821	0.01027	0.00833	0.00824	13.98649
1.8	0.01207	0.00973	0.01226	0.01214	17.98531
2.2	0.01721	0.00910	0.01751	0.01735	21.97963
2.6	0.02440	0.00962	0.02496	0.02472	25.97915
3	0.03522	0.01001	0.03625	0.03588	29.96824
3.4	0.05319	0.00957	0.05525	0.05472	33.96140
3.8	0.08941	0.00939	0.09456	0.09368	37.96504
4.2	0.19994	0.00983	0.22574	0.22352	41.98173
4.6	1.00000	0.00997	0.91906	0.90990	44.55132

Tabela 5.5 - Valores dos parâmetros do modelo teórico considerando taxa de erros para tramas de dados de 1%

R (Mb/s) por fluxo	Sem Erros	Com erros			
	α	p_e	α'	α'_u	R'_u
1	0.00521	0.10261	0.00600	0.00538	9.99323
1.4	0.00821	0.09992	0.00959	0.00863	13.98999
1.8	0.01207	0.10038	0.01445	0.01300	17.98914
2.2	0.01721	0.09884	0.02122	0.01912	21.97784
2.6	0.02440	0.09932	0.03163	0.02849	25.97427
3	0.03522	0.10057	0.04953	0.04455	29.96601
3.4	0.05319	0.09940	0.08610	0.07754	33.96338
3.8	0.08941	0.09861	0.20920	0.18857	37.98769
4.2	0.19994	0.09898	0.69333	0.62470	40.28374
4.6	1.00000	0.09881	0.69701	0.62814	40.29692

Tabela 5.6 - Valores dos parâmetros do modelo teórico considerando taxa de erros para tramas de dados de 10%

O aumento do valor de α' em relação ao valor de α (no cenário sem erros) para o mesmo valor de R deve-se não só à necessidade da retransmissão de tramas mas também ao aumento do tempo de ciclo, o que implica uma acumulação média de um maior número de pacotes por ciclo e justifica que $\alpha' > \alpha'_u > \alpha$.

Porém, quando ocorre saturação, não é possível atingir a ocupação total da janela de transmissão, o que se pode comprovar nas tabelas 5.5 e 5.6, com o valor máximo de α' inferior a 1, pois a necessidade de retransmissão de tramas leva a que a rotação da janela de transmissão ocorra de forma parcial, ou no pior dos casos, não ocorra. O valor máximo de α' (e portanto da ocupação máxima da janela) diminui com o aumento da taxa de erros. De acordo com a equação 3.22, o *goodput* máximo tem uma redução determinada pela probabilidade de erro e um agravamento adicional (em relação ao cenário sem erros) pelo facto de o valor máximo de α' ser inferior a 1; no entanto, nos exemplos, este fator tem um peso reduzido, visto que k é pequeno comparado com o valor máximo de α' .

5.1.7 Outros modelos de tráfego (débito não constante)

Todos os testes realizados até agora tinham como base a utilização de um modelo de tráfego com débito constante por estação. Com o objetivo de verificar o comportamento do protocolo com outros modelos de tráfego foram realizadas simulações utilizando três tipos diferentes de distribuições de tráfego:

- Distribuição ON/OFF em que o gerador está ativo (estado ON) por um período constante de 0.7s e é inibido (estado OFF) por um período constante de 0.3s;

- Distribuição ON/OFF em que o gerador está ativo por um período constante de 0.7s e é inibido por um período dado por uma distribuição exponencial com média 0.1s;
- Distribuição ON/OFF em que o gerador está ativo por um período dado por uma distribuição normal com média 8s e variância 2s, sendo o intervalo de inatividade constante e igual a 0.2s.

Os resultados obtidos são apresentados na tabela A.17, A.18 e A.19 do anexo A.

Dos resultados obtidos conclui-se que para as duas primeiras situações há uma redução quer na ocupação da janela quer nos tempos de ciclo quando comparados com a situação da secção 5.1.2, dado que a interrupção do gerador leva a que haja um menor número de pacotes produzidos e consequentemente um menor número de pacotes entregues ao protocolo TPAP, levando a uma menor ocupação de janela e consequentemente a um menor tempo de ciclo.

No terceiro caso dado que o valor atribuído à média da distribuição normal é muito superior ao valor constante definido para o intervalo OFF, os valores de ocupação da janela e de tempo de ciclo encontram-se mais próximos dos obtidos na secção 5.1.2 apesar de se manter a análise anterior.

Concluindo para as três situações, o débito máximo alcançado e o tempo de ciclo máximo correspondem ao obtido na secção 5.1.2, quando considerado tráfego constante, apesar da utilização de modelos de tráfego em que o débito não é constante.

5.1.8 Resultados obtidos sem o protocolo TPAP

Nesta secção pretende-se avaliar o comportamento do mecanismo nativo do *standard* IEEE 802.11, considerando diferentes distâncias entre as estações clientes e a estação base, com utilização e inativação do mecanismo opcional de RTS/CTS. De referir ainda que os parâmetros MAC foram mantidos com os valores *default* e foram ativadas as tramas de reconhecimento de nível MAC (ACKs) e as retransmissões.

Os cenários considerados foram:

- Distância de 20m com mecanismo RTS/CTS inativo e ativo;
- Distância de 200m com mecanismo RTS/CTS inativo e ativo;
- Distância de 1500m com mecanismo RTS/CTS inativo e ativo.

Os resultados obtidos para a distância de 20m são apresentados nas tabelas A.20 e A.21 do anexo A. Os resultados obtidos para a distância de 200m são apresentados nas tabelas A.22 e A.23 do anexo A. Os resultados obtidos para a distância de 1500m são apresentados nas tabelas A.24 e A.25 do anexo A.

Nas tabelas mencionadas é fornecido o *goodput* por fluxo, quer das estações clientes para a estação base quer da estação base para cada uma das estações clientes, de forma a verificar-se se ocorre uma distribuição uniforme do *goodput* pelas estações ou se se verifica alguma assimetria (*unfairness*) entre as mesmas.

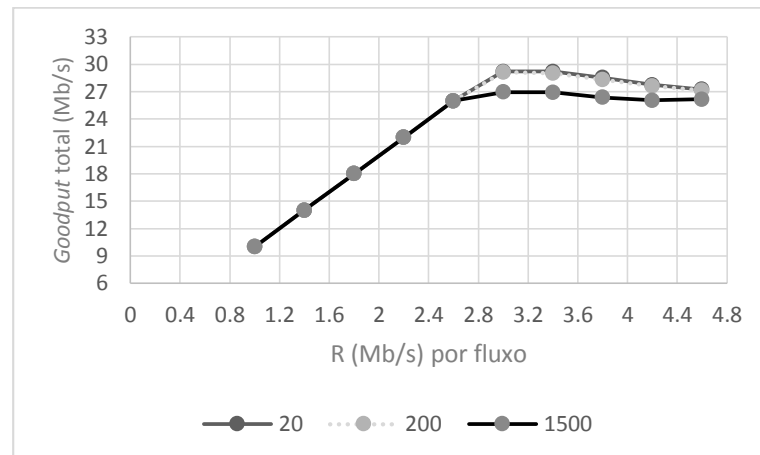


Figura 5.18 - Variação do *goodput* total com o mecanismo nativo do *standard* IEEE 802.11, com RTS/CTS inativo

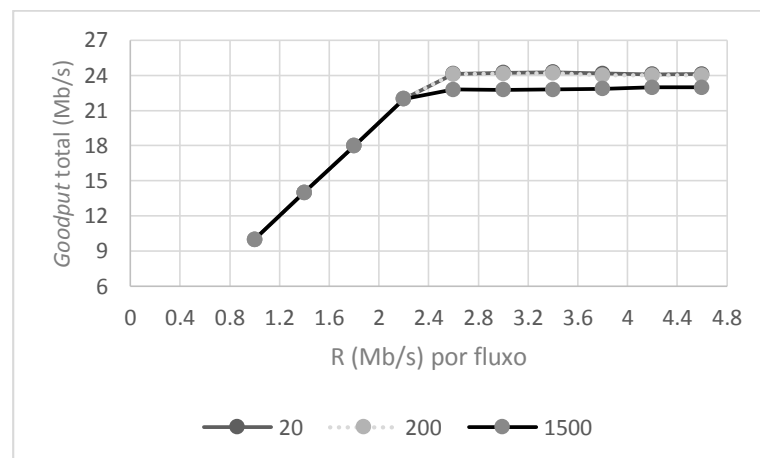


Figura 5.19 - Variação do *goodput* total com o mecanismo nativo do *standard* IEEE 802.11, com RTS/CTS ativo

Analisando os valores de *goodput* na Figura 5.18 (RTS/CTS inativo) e na Figura 5.19 (RTS/CTS ativo) verifica-se que para distâncias de 20 e 200 metros são muito próximos, sendo ligeiramente superiores para a distância de 20 metros, em qualquer dos casos.

Qualquer que seja a distância verifica-se por simples inspeção das figuras que os resultados obtidos com RTS/CTS inativo são melhores do que com a sua ativação.

Da Figura 5.18 podemos concluir igualmente que para as distâncias de 20 e 200 metros, à medida que se aumenta o débito por fluxo, o *goodput* total tende para cerca de 27Mbit/s, enquanto para distâncias de 1500 metros o valor do *goodput* total tende para 26Mbit/s.

Com a ativação do mecanismo RTS/CTS, para as distâncias de 20 e 200 metros o *goodput* total tende para os 24Mbit/s, enquanto para a distância de 1500 metros o valor do *goodput* total estabiliza nos 23Mbit/s.

Outra conclusão a retirar dos resultados obtidos tem que ver com o tratamento *unfair* da estação base a partir de um certo valor de débito por fluxo de 2.6Mbit/s no caso da inativação do mecanismo RTS/CTS e de 2.2Mbit/s no caso da ativação do mecanismo RTS/CTS. Conclui-se que o grau de *unfairness* aumenta, por um lado com o aumento do débito por fluxo, por outro com o aumento da distância para 1500 metros.

Além do já referido, de notar também a existência de *unfairness* entre os próprios fluxos da estação base, o que pode ser atribuído à posição relativa das estações clientes.

5.1.9 Conclusões

Dos resultados obtidos para as duas soluções conclui-se que a utilização do TPAP completo para a arquitetura ponto-multiponto requerida no projeto é vantajosa face à utilização do mecanismo nativo do *standard* IEEE 802.11, uma vez que obtemos uma distribuição equitativa do *goodput* por todos os fluxos, sendo também os valores de *goodput* muito superiores aos obtidos sem a utilização do protocolo TPAP.

Mesmo considerando a existência do tempo de processamento e de erros associados ao protocolo TPAP, que numa implementação real terão de ser considerados, pela análise realizada dos dois casos mencionados nas secções 5.1.5 e 5.1.6, respetivamente, verificou-se mesmo assim um *goodput* total muito superior ao obtido no caso da não utilização do protocolo TPAP.

5.2 Resultados obtidos com o *standard* IEEE 802.11n, com agregação de nível MAC inibida

Nesta secção serão apresentados os resultados obtidos usando o *standard* IEEE 802.11n, que possui suporte ao mecanismo de agregação e de *block acknowledgment* na camada MAC, mas que não serão ainda utilizados, pelo que consideramos mais uma vez a versão completa do protocolo TPAP.

Pretende-se deste modo, antes de explorar os mecanismos avançados do IEEE 802.11n, fazer uma comparação com os resultados obtidos com o IEEE 802.11a, em situações idênticas e assim averiguar o impacto do aumento da capacidade do canal nos parâmetros do modelo e portanto a importância relativa dos *overheads* associados ao TPAP e às camadas PHY e MAC.

Serão também apresentados os resultados obtidos sem o protocolo TPAP, utilizando o mecanismo nativo do *standard* IEEE 802.11, e por fim serão retiradas as conclusões relativas à comparação dos resultados obtidos entre as duas soluções discutidas nesta secção, bem como com os resultados obtidos na secção 5.1.

5.2.1 TPAP completo com eliminação de *gaps* no nível MAC

Nesta secção será realizada a análise aos resultados obtidos com o uso do protocolo TPAP completo sobre IEEE 802.11n, mantendo-se a inibição dos ACKs, retransmissões e *gaps* no nível MAC, e a distância entre estações clientes e a estação base de 1500 metros. Também o tamanho dos pacotes foi mantido nos 1500 bytes e janela com tamanho 16, ou seja, as mesmas condições da secção 5.1.2, com alteração do *standard* utilizado e consequentemente da capacidade de canal disponível.

Os resultados obtidos são apresentados na tabela A.26 do anexo A, o que permite comparar a ocupação da janela (Figura 5.20), o tempo de ciclo (Figura 5.21) e o *goodput* máximo (Figura 5.22) nas duas variantes do protocolo.

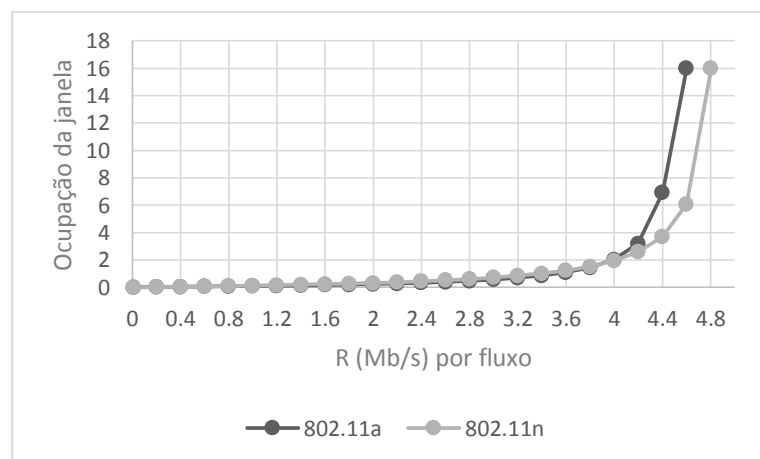


Figura 5.20 - Variação da ocupação da janela considerando o TPAP completo e os *standards* IEEE 802.11a e IEEE 802.11n

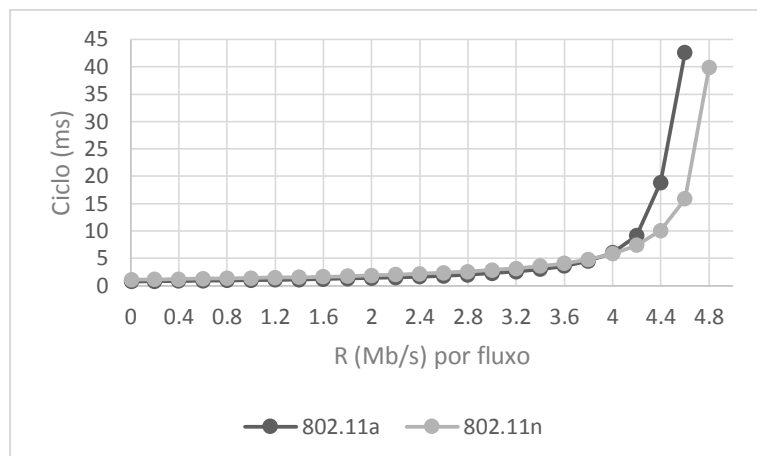


Figura 5.21 - Variação do tempo de ciclo considerando o TPAP completo e os *standards* IEEE 802.11a e IEEE 802.11n

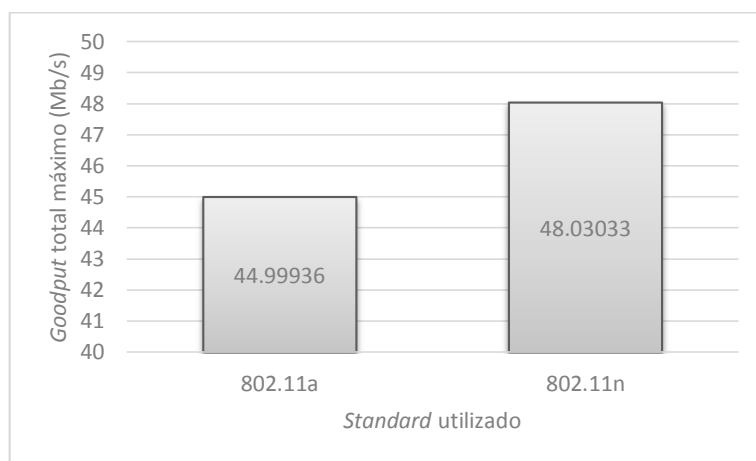


Figura 5.22 - *Goodput* total máximo obtido considerando o protocolo TPAP completo e os *standards* IEEE 802.11a e IEEE 802.11n

Comparando os resultados obtidos com o uso do TPAP completo sobre IEEE 802.11a, verifica-se que até ao débito por fluxo de 3.8Mbit/s, tanto a ocupação da janela como o tempo de ciclo são superiores quando se utiliza o *standard* IEEE 802.11n; contudo, para débitos superiores a 3.8Mbit/s esse facto inverte-se (uma vez que no caso do IEEE 802.11a se está próximo da saturação), havendo uma redução no tempo de ciclo e na ocupação da janela (para o mesmo valor de R). O *goodput* máximo é naturalmente maior (devido à maior capacidade do canal físico); o tempo de ciclo máximo é ligeiramente inferior ao obtido aquando do uso do *standard* IEEE 802.11a (mas em condições de *goodput* máximo diferentes).

5.2.1.1 Validação do modelo teórico

Na Figura 5.23 que representa a função $T(n)$ verifica-se que o crescimento linear do tempo de ciclo com n se mantém, pelo que o método de regressão linear utilizado para a obtenção dos parâmetros do modelo teórico continua válido para a análise desta situação.

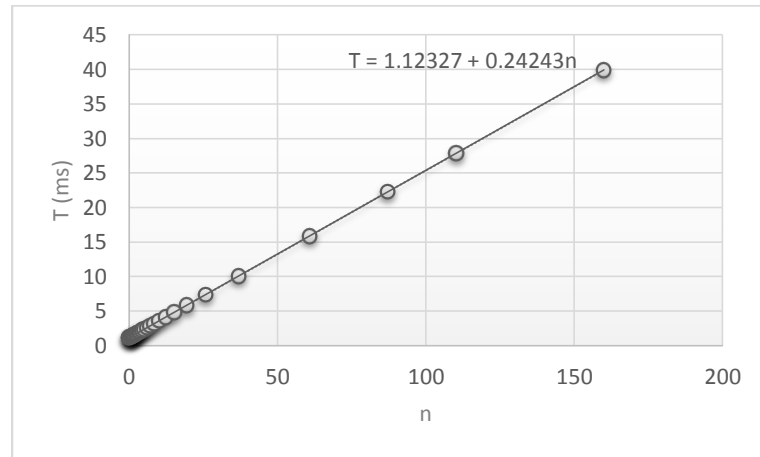


Figura 5.23 - Figura obtida através do método de regressão linear, considerando o cenário com TPAP completo e *standard* IEEE 802.11n, com eliminação dos *gaps* no nível MAC

Os valores obtidos para os parâmetros ilustrativos do modelo teórico são apresentados na tabela 5.7, juntamente com os obtidos anteriormente para o *standard* IEEE 802.11a, nas mesmas condições.

<i>Standard</i> IEEE	T_d (ms)	T_p (ms)	T_o (ms)	O_{TPAP} (ms)	k	C (Mb/s)	S_{max} (%)	R_{max} (Mb/s)
802.11a	0.222	0.261	0.039	0.783358	0.019	45.92	98.16	45.07
802.11n	0.185	0.242	0.058	1.123274	0.029	49.50	97.19	48.11

Tabela 5.7 - Valores dos parâmetros do modelo teórico considerando o protocolo TPAP completo e os *standards* IEEE 802.11a e IEEE 802.11n

Em relação ao *standard* IEEE 802.11a verifica-se uma diminuição do tempo para transportar o *payload* de uma trama de dados, T_d , e um aumento do *overhead* associado às camadas MAC/PHY, T_o . Contudo o tempo total de ocupação do meio com a transmissão de uma trama de dados, T_p , diminuiu.

Relativamente ao *overhead* associado ao protocolo TPAP, O_{TPAP} , verifica-se um aumento do mesmo, o que pode ser justificado pelo aumento já referido do *overhead* do nível MAC/PHY a que as tramas do protocolo TPAP também estão sujeitas.

O aumento do *overhead* associado ao TPAP aliado à diminuição de T_p implica um aumento do parâmetro k ; por este motivo o impacto na degradação de desempenho do protocolo TPAP é maior, como se comprova pela diminuição da eficiência máxima, S_{max} . Um aumento de 11Mbit/s na capacidade do canal físico traduziu-se num ganho de cerca de 3Mbit/s no *goodput* máximo. É de prever que um aumento ainda maior da capacidade do canal acentue esta diminuição da eficiência.

Apesar da diminuição da eficiência, dada a maior capacidade útil disponível, C , o débito máximo alcançável é superior e encontra-se próximo do valor obtido através das simulações para o valor do *goodput* total máximo.

5.2.2 Resultados obtidos sem o protocolo TPAP

Tal como no caso do *standard* IEEE 802.11a, pretende-se agora avaliar o comportamento do mecanismo nativo do *standard* IEEE 802.11, considerando diferentes distâncias entre as estações clientes e a estação base, com a ativação e inativação do mecanismo alternativo de RTS/CTS, de forma a comparar os resultados obtidos quer com o protocolo TPAP sobre IEEE 802.11n, quer com os resultados obtidos com IEEE 802.11a nativo. Os parâmetros MAC foram mantidos com os valores *default* e foram ativados os ACKs e as retransmissões.

Os cenários considerados foram os mesmos da secção 5.1.8.

Os resultados obtidos para a distância de 20m são apresentados nas tabelas A.27 e A.28 do anexo A. Os resultados obtidos para a distância de 200m são apresentados nas tabelas A.29 e A.30 do anexo A. Os resultados obtidos para a distância de 1500m são apresentados nas tabelas A.31 e A.32 do anexo A.

Os resultados obtidos com o *standard* IEEE 802.11n apresentados nas Figuras 5.24 e 5.25 revelam um comportamento semelhante ao apresentado para o *standard* IEEE 802.11a, discutido na secção 5.1.8. Relativamente às distâncias de 20 e 200 metros, os resultados são muito próximos mas ligeiramente melhores para a distância de 20 metros, quer se use o mecanismo RTS/CTS, quer quando o mesmo se encontra inativo.

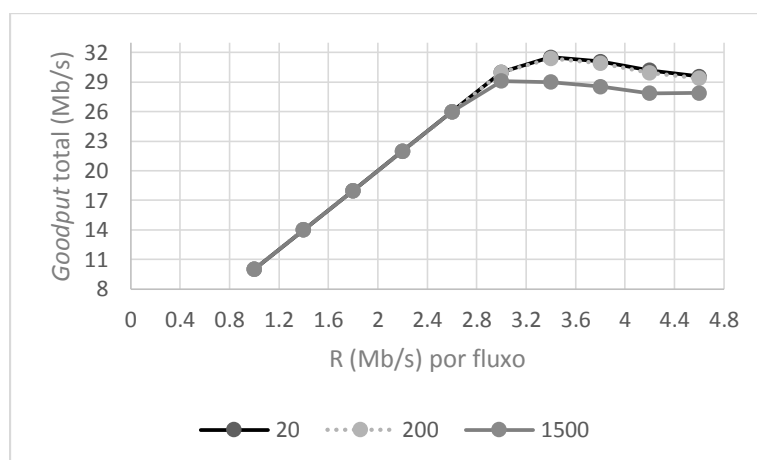


Figura 5.24 - Variação do *goodput* total com o mecanismo nativo do *standard* IEEE 802.11, com RTS/CTS inativo

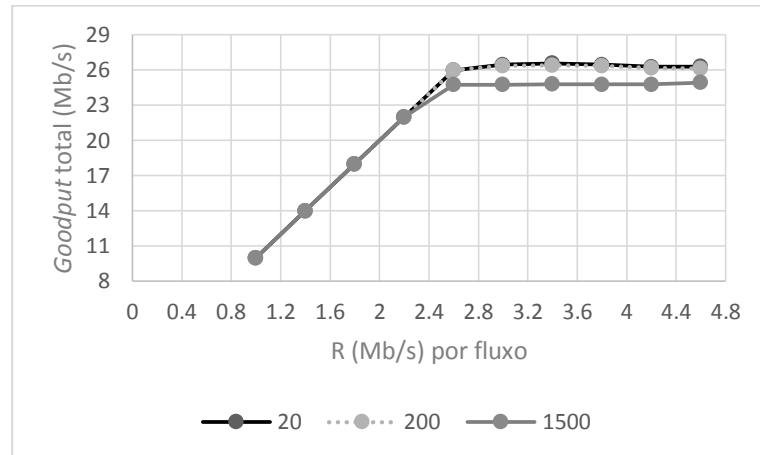


Figura 5.25 - Variação do *goodput* total com o mecanismo nativo do *standard* IEEE 802.11, com RTS/CTS ativo

Também aqui se conclui que os resultados obtidos sem o mecanismo RTS/CTS são melhores do que os obtidos com a ativação do mesmo. Com o mecanismo RTS/CTS inativo, à medida que se aumenta o débito por fluxo, o *goodput* para as distâncias de 20 e 200 metros tende para 29Mbit/s enquanto para a distância de 1500 metros o seu valor tende para 28Mbit/s.

Com a ativação do mecanismo RTS/CTS o valor do *goodput* para as distâncias de 20 e 200 metros tende para 26Mbit/s, enquanto para a distância de 1500 metros o seu valor é próximo dos 25Mbit/s.

O tratamento *unfair* da estação base mantém-se e verifica-se a partir dos 3Mbit/s sem o uso do mecanismo RTS/CTS e a partir dos 2.6Mbit/s com o uso do mecanismo RTS/CTS, para as distâncias de 20 e 200 metros. Para a distância de 1500 metros a situação agrava-se e com o mecanismo de RTS/CTS inativo o fenómeno verifica-se a partir dos 2.6Mbit/s e com a ativação do mecanismo RTS/CTS verifica-se a partir dos 2.2Mbit/s.

Por fim é de referir que continua a verificar-se *unfairness* na distribuição do *goodput* entre os fluxos da estação base, pelo motivo referido anteriormente, e que está relacionado com a posição relativa das estações clientes.

5.2.3 Conclusões

Dos resultados obtidos para o *standard* IEEE 802.11n, quer com a utilização do TPAP completo, quer sem o uso do protocolo TPAP, conclui-se também aqui que a utilização do TPAP completo é vantajosa face à utilização do mecanismo nativo do *standard* IEEE 802.11, uma vez que a estação base não é sujeita ao fenómeno de *unfairness*, havendo uma equidade na distribuição dos recursos, conduzindo a valores de *goodput* muito superiores aos obtidos sem a utilização do protocolo TPAP.

Dos resultados obtidos para o protocolo TPAP completo, considerando as duas variantes do *standard* IEEE 802.11 utilizadas, podemos constatar que para débitos por fluxo até aos

3.8Mbit/s o standard IEEE 802.11n não oferece vantagens; contudo, a partir deste débito ocorre uma diminuição da ocupação da janela e dos tempos de ciclo a que se junta um *goodput* máximo superior ao alcançado com o *standard* IEEE 802.11a. Em termos quantitativos e analisando os valores dos parâmetros do modelo teórico, verificamos um aumento dos *overheads*, tanto associado às camadas MAC/PHY como ao protocolo TPAP, o que resulta num maior impacto na degradação introduzido pelo protocolo TPAP, levando a uma menor eficiência máxima. Contudo e dada a redução no tempo para transmitir uma trama de dados, a capacidade útil disponível é superior e portanto o débito máximo alcançável é superior ao obtido para o *standard* IEEE 802.11a.

Comparando por fim as duas situações em que não foi utilizado o protocolo TPAP, verificamos que o comportamento é semelhante, obtendo-se melhores resultados sem a utilização do mecanismo RTS/CTS. Verifica-se em ambos o fenómeno de *unfairness* na estação base, o que justifica a utilização do protocolo TPAP na versão completa. Verifica-se também que com a utilização do *standard* IEEE 802.11n ocorre uma melhoria em termos de *goodput* máximo, justificada pelo aumento da capacidade do canal de transmissão.

5.3 Resultados obtidos com o *standard* IEEE 802.11n, com agregação e BACK de nível MAC

Nesta secção serão apresentados os resultados obtidos usando o *standard* IEEE 802.11n fazendo uso do mecanismo de agregação e de *block acknowledgment* na camada MAC.

Dada a utilização do mecanismo de agregação e de BACK de nível MAC, a utilização do protocolo TPAP na sua versão completa não fazia sentido, pelo que foi usada a versão simples do mesmo, considerando duas alternativas. Na primeira continuam a ser usadas tramas POLL e RET explícitas, enquanto na segunda as tramas POLL e RET são substituídas por uma *flag* na última trama de dados na direção correspondente, caso a trama de dados exista; caso contrário mantém-se a utilização da trama POLL ou RET.

Após a apresentação dos resultados correspondentes ao protocolo TPAP simples, serão apresentados os resultados obtidos sem o protocolo TPAP, à semelhança dos casos anteriores.

Por fim será realizado um sumário das conclusões a retirar desta secção.

5.3.1 TPAP simples

Para a análise do protocolo TPAP na sua versão simples foi mantida a distância entre as estações clientes e a estação base em 1500 metros, o tamanho dos pacotes em 1500 bytes e a janela com tamanho 16. Em relação ao nível MAC, o mecanismo de RTS/CTS foi inibido, foram usados os valores *default* dos parâmetros e, como referido, foram utilizadas tramas BACK nativas (e tramas ACK quando não é possível agregar tramas de Dados).

Os resultados com tramas POLL e RET explícitas são apresentados na tabela A.33 do anexo A, ao passo que os resultados em que estas tramas são substituídas, quando possível, por uma *flag* na última trama de dados são apresentados na tabela A.34 do anexo A.

As situações em análise são comparadas relativamente ao tempo de ciclo (Figura 5.26), ocupação da janela (Figura 5.27) e *goodput* máximo (Figura 5.28).

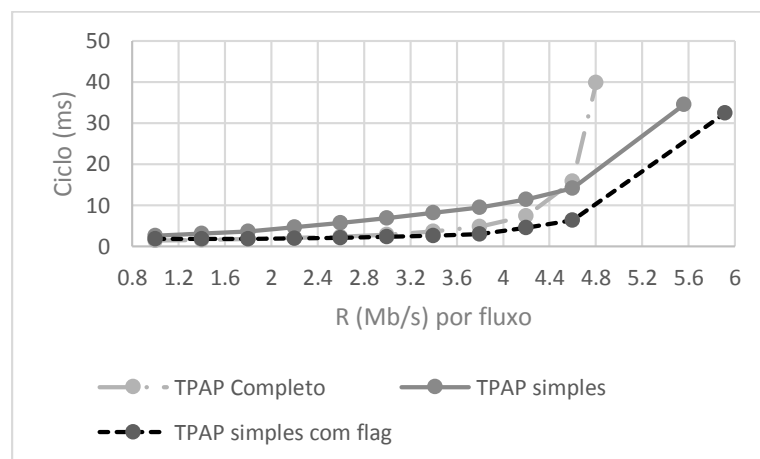


Figura 5.26 - Variação do tempo de ciclo com o aumento do débito por fluxo, considerando as diferentes versões do protocolo TPAP com o *standard* IEEE 802.11n

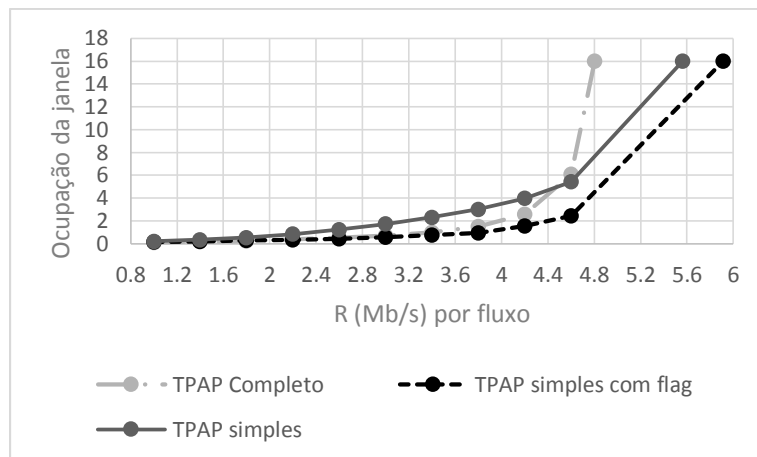


Figura 5.27 - Variação da ocupação da janela com o aumento do débito por fluxo, considerando as diferentes versões do protocolo TPAP com o *standard* IEEE 802.11n

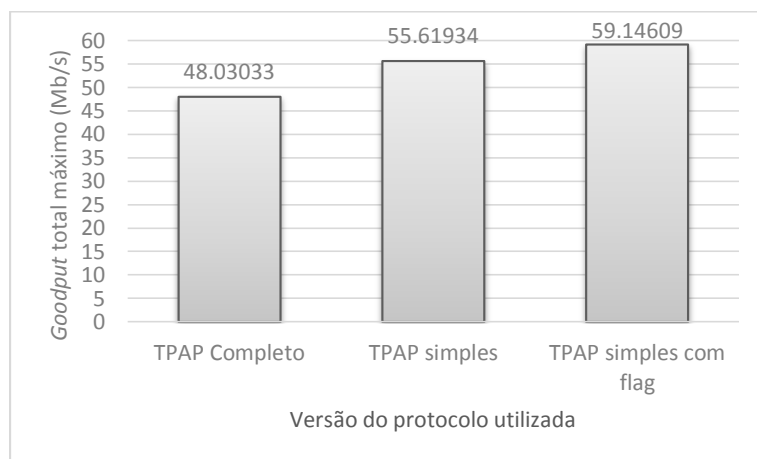


Figura 5.28 - Variação do *goodput* total máximo, considerando as diferentes versões do protocolo TPAP com o *standard* IEEE 802.11n

Considerando em primeiro lugar o TPAP simples com tramas POLL e RET explícitas, referido nas figuras como TPAP simples, verifica-se relativamente à situação com protocolo TPAP completo e *standard* IEEE 802.11n, um aumento do tempo de ciclo e da ocupação da janela até ao débito por fluxo de 4.4Mbit/s, invertendo-se depois a situação. Este aumento está associado ao facto de a ocupação da janela ser baixa e haver ciclos em que apenas se transmite uma trama de dados, levando à geração de uma trama ACK individual por cada trama de dados enviada, o que aumenta o *overhead* introduzido na camada MAC. À medida que a ocupação da janela aumenta para valores superiores a duas tramas por ciclo, a diferença nos tempos de ciclo começa a diminuir, uma vez que se passa a tirar partido do mecanismo de agregação e de BACK do nível MAC.

Apesar dos aumentos referidos de tempo de ciclo e ocupação da janela até aos 4.4Mbit/s, o débito máximo alcançado é bastante superior ao obtido sem a agregação de nível MAC, e o tempo de ciclo máximo obtido é inferior.

De referir que se verificou na simulação que as tramas POLL e RET não são agregadas em conjunto com as tramas de dados, sendo gerada uma trama ACK individual por cada trama de controlo associada ao protocolo TPAP, o que leva a uma diminuição do *goodput* máximo alcançável; esta situação não é fácil de explicar, uma vez que para a camada MAC uma trama de controlo do protocolo não se distingue de uma trama de Dados TPAP (ambas são encapsuladas em tramas MAC de Dados). Dada a situação referida decidiu-se introduzir uma *flag* na última trama de dados, caso exista, de forma a sinalizar o fim da transmissão no sentido respetivo (em substituição das tramas POLL e RET explícitas). Caso não exista nenhuma trama de dados para enviar mantém-se o envio de uma trama POLL ou de uma trama de RET, conforme se trate da estação base ou das estações clientes, respetivamente.

Os resultados obtidos, em comparação com o primeiro cenário, refletem uma melhoria tanto na ocupação da janela como na duração dos ciclos, levando a um ganho de *goodput* máximo significativo.

Comparando com o protocolo TPAP completo verifica-se que até aos 1.8Mbit/s tanto a ocupação da janela como o tempo de ciclo são superiores, causado possivelmente pelo facto de haver confirmações individuais das tramas e de não haver tramas de dados suficientes para se retirar vantagem do facto de a última trama de dados transportar a *flag*, dado que na maioria dos ciclos a inexistência de tramas de dados leva a que se mantenha o envio das tramas POLL e RET de forma explícita. Contudo para valores de débito por fluxo superiores a 1.8Mbit/s, tanto a ocupação da janela como o tempo correspondente de duração do ciclo são inferiores aos obtidos com o protocolo TPAP completo.

5.3.1.1 Validação do modelo teórico

Nesta secção será realizada a análise aos parâmetros do modelo teórico para o caso do TPAP simples com tramas POLL e RET explícitas.

Neste caso, dada a utilização do mecanismo de agregação, temos uma divisão em três zonas com comportamento diferente. Da função $T(n)$ apresentada na Figura 5.29, verificamos uma primeira zona definida para valores de ocupação da janela por fluxo inferiores a um (ou seja, $n < 10$), que corresponde a valores em que não é possível agregação. Uma segunda zona intermédia corresponde a valores de ocupação da janela por fluxo entre um e dois (ou seja, n entre 10 e 20); nesta zona a função não é linear, dado que, conforme os ciclos, tanto pode ocorrer agregação de tramas como não. Uma terceira zona é definida para valores de ocupação da janela por fluxo superiores a dois (ou seja $n > 20$), onde a agregação é realizada.

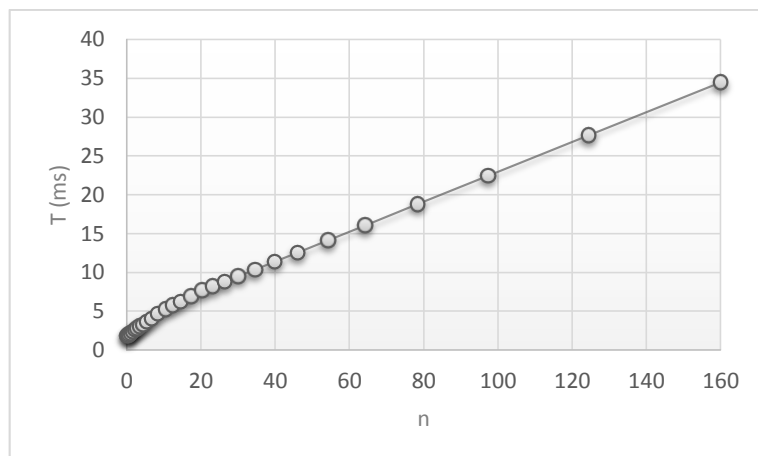


Figura 5.29 - Relação do tempo de ciclo com o aumento de n , considerando o TPAP simples com *standard* IEEE 802.11n

A Figura 5.30 apresenta uma visão expandida da primeira zona, em que cada estação lógica transmite no máximo uma trama de dados por ciclo, não ocorrendo agregação. A função $T(n)$ pode ser aproximada por uma reta, embora se verifiquem alguns desvios, que não podem ser diretamente imputados à agregação, pelo facto de nesta zona cada estação transmitir no máximo uma trama de dados por ciclo. No entanto, o mecanismo de agregação está ativo, o que implica que em transmissões de tramas sucessivas, o intervalo DIFS seja substituído por RIFS (*Reduced Inter-Frame Spacing*). Este intervalo mais pequeno ocorre antes do envio de POLL ou RET sempre que uma destas tramas seja antecedida duma trama de dados. Com o aumento de n , o número de tramas de dados aumenta e, consequentemente, também aumenta o número de ciclos em que as tramas POLL e RET são precedidas de uma trama de dados, implicando a utilização de RIFS em vez de DIFS (o que não aconteceria se a agregação fosse inibida, tal como se comprovou através de simulação, caso em que não ocorreram os desvios referidos).

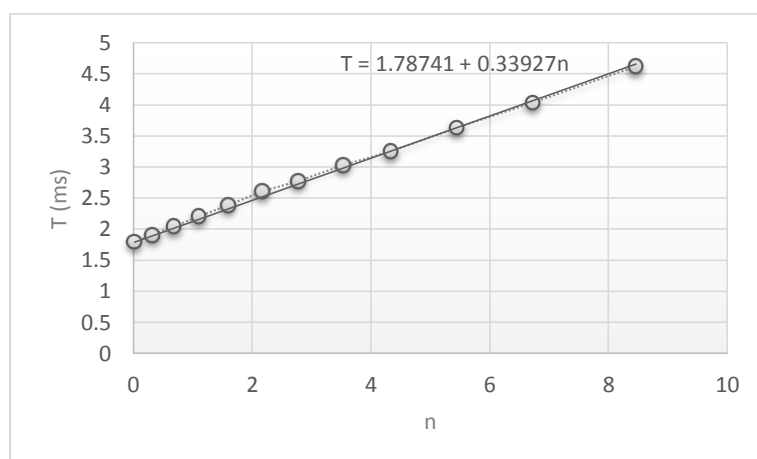


Figura 5.30 - Figura obtida através do método de regressão linear, considerando o cenário com *standard* IEEE 802.11n, com protocolo TPAP simples (primeira zona de $T(n)$)

Nesta zona aplica-se o modelo básico, pelo que seria possível calcular k , sendo o seu valor igual a 0,033, ligeiramente superior ao valor obtido com TPAP completo. No entanto, este

valor tem um interesse relativo, pois o comportamento nesta zona não é extrapolável para toda a gama de valores de R mas apenas se aplica para valores de R inferiores a cerca de 2.2Mbit/s. O valor de O_{TPAP} é também superior ao do TPAP completo devido ao aumento do tempo associado à transmissão de uma trama de dados (sem agregação), que inclui para além dos cabeçalhos e *gaps*, a geração de ACKs na camada MAC. Estes factos explicam os valores de tempos de ciclo e ocupação da janela superiores aos obtidos com TPAP completo nesta zona para o mesmo valor de R .

Passando à análise da terceira zona mencionada ($n > 20$) verificamos, pela Figura 5.31, a linearidade da função $T(n)$ obtida através do método de regressão linear.

Neste caso, o declive da reta igual a 0.19233 representa T_{MAC} , ou seja, o tempo total de ocupação do meio com a transmissão de uma sub-trama de dados, como definido, enquanto o valor de intersecção da reta com o eixo dos yy, igual a 3.71964, representa o parâmetro O , ou seja, o *overhead* total devido ao protocolo TPAP e à camada PHY.

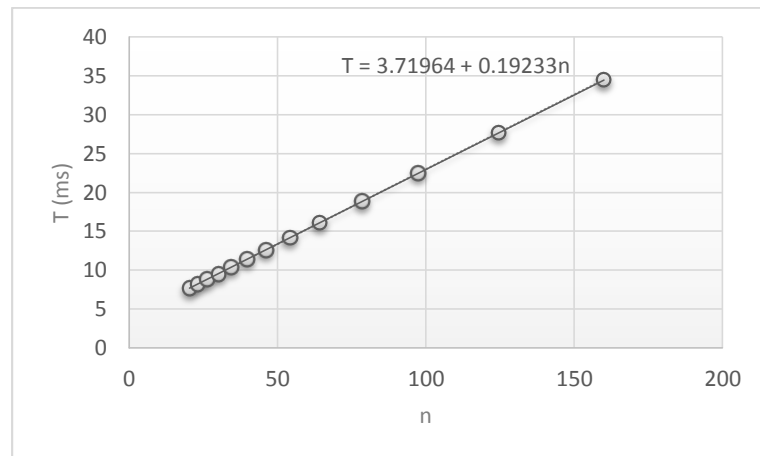


Figura 5.31 - Figura obtida através do método de regressão linear, considerando o cenário com *standard* IEEE 802.11n, com protocolo TPAP simples (terceira zona de $T(n)$)

Os valores dos parâmetros obtidos para este cenário são apresentados na tabela 5.8.

T_d (ms)	T_{MAC} (ms)	O (ms)	k^*	C^* (Mb/s)	S_{max} (%)	R_{max} (Mb/s)
0.185	0.192	3.720	0.121	62.39	89.22	55.66

Tabela 5.8 - Valores dos parâmetros do modelo teórico considerando o protocolo TPAP simples, com *standard* IEEE 802.11n, para a terceira zona de $T(n)$

Verifica-se que a degradação de desempenho devido ao *overhead* de uma sub-trama MAC é pequeno, visto T_{MAC} ser apenas cerca de 4% superior a T_d , o que se traduz num valor elevado de C^* . A degradação deve-se essencialmente ao protocolo TPAP e à camada PHY e o seu impacto é traduzido pelo parâmetro k^* ; o seu valor relativamente elevado em comparação com cenários anteriores tem como consequência uma eficiência máxima prevista

de 89.22%, em relação a C^* (e de 85.64% em relação a C_c). Contudo, visto que o valor de C^* é elevado, o débito máximo previsto é de 55.66 Mbit/s, muito próximo do obtido através das simulações.

Nesta zona mantém-se a componente O_{TPAP} pelo facto de as tramas POLL e RET não serem agregadas a tramas de dados, o que implica a geração de tramas ACK individuais na camada MAC. A substituição de tramas POLL e RET por *flags*, quando possível, elimina o *overhead* do protocolo TPAP, o que se manifestou nas simulações correspondentes (Tabela A.34), obtendo-se um *goodput* máximo igual a 59,146Mbit/s e uma eficiência máxima global de cerca de 91%, e portanto um ganho relativamente aos valores 55.66 Mbit/s e 85,64% acima referidos.

5.3.2 Resultados obtidos sem o protocolo TPAP

Tal como nas secções anteriores, também para o *standard* IEEE 802.11n com utilização de agregação e BACK de nível MAC foram realizadas simulações utilizando o mecanismo nativo do *standard* IEEE 802.11.

Apenas serão apresentados os resultados obtidos para as distâncias de 20, 200 e 1500 metros considerando o mecanismo de RTS/CTS inativo, dado que surgiram problemas com o simulador quando se ativava o mecanismo de RTS/CTS nas condições atuais.

Os resultados obtidos são apresentados nas tabelas A.35, A.36 e A.37 do anexo A, respetivamente para as distâncias de 20, 200 e 1500 metros.

Como ilustrado na Figura 5.32, verifica-se que, para as três distâncias consideradas, até ao débito por fluxo de 2.2Mbit/s, o *goodput* total obtido é igual ao débito total gerado.

Contudo, para débito entre os 2.2 e os 3.8Mbit/s observa-se uma instabilidade nos valores, com degradação no *goodput*, apresentando derivada negativa. Posteriormente para os débitos por fluxo de 4.2 e 4.6Mbit/s observa-se uma recuperação, que poderá estar relacionada com um ganho de eficiência associado ao mecanismo de agregação, dado o maior número de pacotes que é possível agregar.

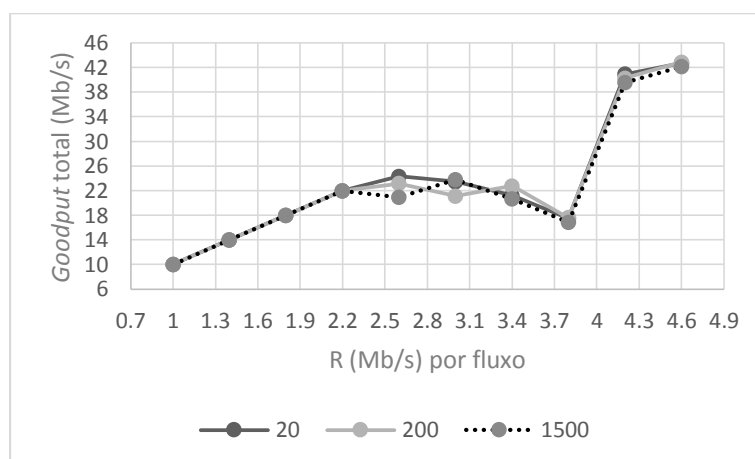


Figura 5.32 - Variação do *goodput* total com o mecanismo nativo do *standard* IEEE 802.11, com RTS/CTS inativo

Outra conclusão que se retira por inspeção dos resultados obtidos é que para esta situação a distribuição do débito pelos fluxos é mais ou menos equitativa, obtendo a estação base quase 50% do *goodput* total, e tendo cada um dos fluxos individuais da estação base sensivelmente o mesmo débito. Esta diferença para as análises realizadas anteriormente sem o protocolo TPAP deverá ser atribuída ao uso do mecanismo de agregação, dado que é a única alteração significativa.

5.3.3 Conclusões

Da análise dos resultados obtidos com IEEE 802.11n nos vários cenários conclui-se que, dada a degradação do desempenho numa gama intermédia de valores de débito por fluxo no caso da não utilização do protocolo TPAP, esta não é aceitável, pelo que esta solução não é recomendável. Significa isto que, a utilização do protocolo TPAP na versão simplificada, de forma a poder ser explorado o mecanismo de agregação, é altamente vantajosa, pois para além do *fairness*, permite valores de *goodput* muito superiores. Mesmo admitindo algum *overhead* adicional de processamento, o *goodput* máximo deve continuar muito acima do possível sem TPAP. Para além disso, com a utilização do protocolo TPAP, é possível através da configuração do valor da janela, por cada ligação unidirecional, controlar a distribuição de débitos pelos fluxos.

Capítulo 6

Conclusões e Trabalho Futuro

Na presente dissertação foram implementadas no simulador ns-3 duas versões do protocolo TPAP, concebido para uma solução de comunicação bidirecional numa configuração ponto-multiponto, em redes de acesso sem fios baseadas no *standard* IEEE 802.11. Para a realização das simulações foram consideradas duas variantes do *standard* IEEE 802.11. Na variante sem suporte ao mecanismo de agregação, o IEEE 802.11a, foi utilizada a versão completa do protocolo TPAP, que permite emular a agregação de tramas, dado que cada estação pode enviar na sua vez um *burst* de tramas até ao limite da janela (embora sem a eficiência do mecanismo de agregação na camada MAC). Relativamente ao *standard* IEEE 802.11n, que suporta a agregação na camada MAC, foi implementada a versão simples do protocolo TPAP, que apenas mantém o controlo de acesso ao meio através do mecanismo de *polling* e o controlo de fluxo por meio da janela, de forma a realizar uma distribuição controlada dos recursos da rede pelos diferentes fluxos.

Com o objetivo de, através de comparações adicionais, avaliar as vantagens de implementar TPAP sobre IEEE 802.11 foram realizadas simulações em que o protocolo TPAP foi eliminado, considerando apenas os mecanismos nativos dos *standards* IEEE 802.11 (incluindo a agregação no caso de IEEE 802.11n), juntamente com a ativação e inativação do mecanismo opcional de RTS/CTS.

Em relação ao *standard* IEEE 802.11a, uma vez que não suporta agregação nem confirmação de tramas em bloco e retransmissão seletiva, só faz sentido considerar o protocolo TPAP completo, que oferece várias vantagens em relação à opção de não o utilizar. Para além de um valor mais elevado do *goodput* máximo (mesmo considerando alguma penalização adicional devido a atrasos de processamento), de uma maior eficiência na recuperação de tramas perdidas devido a erros de transmissão e da estabilidade pelo facto de evitar colisões, permite assegurar uma distribuição equitativa dos recursos pelos fluxos, ao contrário do que acontece quando é utilizado o mecanismo nativo, com ou sem RTS/CTS.

Neste caso verifica-se *unfairness* na estação base que, pela sua natureza, gera um número de fluxos igual ao número de estações clientes, enquanto cada uma destas gera apenas um fluxo.

Relativamente à utilização do *standard* IEEE 802.11n, não considerando a utilização da agregação e *block acknowledgment* do nível MAC, os resultados obtidos para o protocolo TPAP completo revelam um aumento natural no *goodput* máximo, mas pouco significativo tendo em atenção o aumento da capacidade do canal de 11 Mbit/s. Este facto sugere que para valores muito maiores da capacidade do canal, o maior *overhead* relativo quer das camadas inferiores quer do protocolo TPAP completo implica uma utilização ineficiente da capacidade disponível (o que é confirmado pelo modelo teórico). Torna-se assim obrigatório usar o mecanismo de agregação de tramas na camada MAC (aliás concebido para lidar com este problema) e por inerência o mecanismo de confirmação de tramas em bloco e retransmissão seletiva, dispensando por isso o mecanismo equivalente presente no protocolo TPAP completo. Considerando o mecanismo nativo neste cenário (IEEE 802.11n sem agregação), os resultados obtidos revelaram a mesma tendência dos obtidos com o IEEE 802.11a.

Pelas razões apontadas, foi implementada a versão simples do protocolo TPAP, explorando o mecanismo de agregação de tramas e de *block acknowledgment* disponível na variante IEEE 802.11n (e também em variantes mais recentes, como o IEEE 802.11ac).

Os resultados revelaram uma diminuição significativa no *overhead* associado às camadas inferiores, juntamente com um aumento expressivo no *goodput* máximo. As simulações realizadas sem TPAP revelaram possíveis problemas de implementação do mecanismo de confirmação e retransmissão de tramas na versão usada do ns-3. No caso de o mecanismo RTS/CTS ser ativado foi mesmo impossível obter resultados para volumes de tráfego acima dum valor crítico. No caso de inativação deste mecanismo, observou-se uma degradação de desempenho em débitos intermédios não aceitável (traduzida na derivada negativa do *goodput* em relação ao débito oferecido), para além de um *goodput* máximo bastante inferior ao obtido com a versão simples do TPAP. Apesar de o fenómeno de *unfairness* não se ter revelado de forma significativa, as razões indicadas justificam mais uma vez a superioridade do protocolo TPAP, neste caso na versão simples.

Podemos concluir que a versão completa do protocolo TPAP constitui uma solução válida e vantajosa para *standards* sem suporte ao mecanismo de agregação, enquanto a versão simples do protocolo TPAP combinada com o mecanismo de agregação disponível nas versões mais recentes do *standard* IEEE 802.11 constitui uma solução bastante eficiente, estável e equitativa no tratamento dos fluxos e relativamente imune à diferença de distâncias das estações clientes em relação à estação base. O aumento do número de estações (e portanto de fluxos), embora se traduza no aumento do *overhead* do protocolo TPAP (e portanto do atraso), constitui também uma vantagem do protocolo, uma vez que evita colisões que, de outro modo, tenderiam a aumentar e a potenciar um grau mais elevado de *unfairness*.

Em relação ao trabalho realizado foram cumpridos todos os objetivos propostos inicialmente. O desenvolvimento do trabalho correu sem percalços até à obtenção de resultados com o mecanismo de agregação do nível MAC. Há duas situações a relatar neste caso, que embora sejam distintas poderão estar relacionadas com o mesmo problema e revelam uma limitação atual do simulador, possivelmente devido a um *bug*, dada a implementação recente e ainda em curso do mecanismo de agregação. No primeiro caso, devido à ocorrência de um erro durante as simulações foi impossível, como já referido, obter resultados com o *standard* IEEE 802.11n sem TPAP e com ativação do mecanismo de RTS/CTS. Por outro lado, quando se tentou introduzir perda de tramas na versão com o TPAP simples, para testar o mecanismo nativo de recuperação na camada MAC, verificou-se a perda das tramas, como pretendido, mas a sua recuperação revelou problemas. No entanto, o mecanismo de agregação não revelou problemas sérios, à parte da impossibilidade de agregar tramas POLL e RET com tramas de Dados (o que deveria ser transparente para a camada MAC).

Em termos de trabalho futuro, e partindo do trabalho já realizado e que permitiu caracterizar detalhadamente e validar o protocolo TPAP e justificar as vantagens da sua utilização, seria de considerar a implementação dum mecanismo de *smart polling*, especialmente útil em situações com condições de tráfego variáveis. Este mecanismo pode basear-se na adaptação dinâmica da janela de transmissão às necessidades das estações, atribuindo créditos adicionais a estações saturadas ou em vias de saturação, desde que se verifique que algumas estações não usam todos os créditos de que dispõem e não seja violado o tempo máximo de ciclo definido como alvo. De salientar que a janela do protocolo TPAP tem assim um objetivo regulador, não sendo incompatível com a janela implementada na camada MAC e que apenas está associada ao mecanismo de confirmação e retransmissão seletiva de tramas. Com esta adaptação será de esperar um acréscimo no *goodput* máximo, devido a distribuição inteligente dos recursos da rede (neste caso sob a forma de créditos adicionais). Os resultados apresentados na Secção 5.1.4 (tráfego assimétrico) constituem apenas uma verificação inicial e rudimentar desta possibilidade. O modelo teórico foi já estendido de forma a contemplar uma estratégia de *smart polling*, pelo que existem condições para a breve prazo iniciar o processo de simulação.

Anexo A

A. 1 - Resultados com TPAP completo e *standard* IEEE 802.11a, com parâmetros MAC *default*

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	<i>n</i>
		Ocupação da janela	Pacotes Transmitidos			
0	-	-	-	2.1283	-	-
0.01	0.1017	0.0018	16	2.1326	8909	0.0181
0.2	1.9958	0.0375	316	2.2569	8418	0.3754
0.4	3.9992	0.0800	633	2.4017	7911	0.8005
0.6	5.9968	0.1285	950	2.5714	7389	1.2851
0.8	7.9958	0.1841	1266	2.7628	6877	1.8410
1	9.9979	0.2490	1583	2.9888	6356	2.4904
1.2	11.9969	0.3247	1899	3.2480	5849	3.2473
1.4	13.9971	0.4159	2216	3.5657	5329	4.1591
1.6	15.9979	0.5253	2533	3.9399	4822	5.2528
1.8	17.9981	0.6623	2850	4.4158	4302	6.6234
2	19.9971	0.8341	3166	5.0041	3796	8.3409
2.2	21.9979	1.0651	3483	5.8090	3270	10.6511
2.4	23.9937	1.3921	3799	6.9602	2729	13.9205
2.6	25.9926	1.8724	4116	8.6414	2198	18.7244
2.8	27.9903	2.6804	4432	11.4868	1653	26.8036
3	29.9836	4.3134	4748	17.2521	1101	43.1344
3.1	30.9783	5.9697	4905	23.1035	822	59.6970
3.2	31.9693	9.2896	5062	34.8296	545	92.8959
3.25	32.4537	12.6146	5139	46.5649	407	126.1464
3.4	32.7310	16	5184	58.5524	323	160

A. 2 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	<i>n</i>
		Ocupação da janela	Pacotes Transmitidos			
0	-	-	-	0.7834	-	-
0.01	0.0998	0.0007	16	0.7851	24201	0.0065
0.2	2.0021	0.0137	317	0.8195	23184	0.1367
0.4	4.0011	0.0286	634	0.8580	22143	0.2861
0.6	5.9994	0.0450	950	0.9007	21093	0.4504
0.8	7.9990	0.0632	1267	0.9481	20039	0.6320
1	9.9985	0.0834	1583	1.0010	18980	0.8342
1.2	12	0.1060	1900	1.0601	17923	1.0601
1.4	13.9983	0.1314	2217	1.1263	16869	1.3140
1.6	16.0004	0.1602	2533	1.2013	15815	1.6019
1.8	17.9987	0.1931	2850	1.2870	14762	1.9305
2	19.9996	0.2310	3166	1.3862	13706	2.3102
2.2	21.9998	0.2753	3483	1.5017	12652	2.7530
2.4	23.9969	0.3277	3800	1.6385	11595	3.2772
2.6	25.9983	0.3905	4116	1.8021	10543	3.9047
2.8	27.9916	0.4671	4433	2.0022	9489	4.6708
3	29.9981	0.5635	4750	2.2538	8430	5.6347
3.2	31.9971	0.6868	5066	2.5756	7376	6.8684
3.4	33.9966	0.8511	5383	3.0037	6325	8.5106
3.6	35.9962	1.0818	5700	3.6057	5269	10.8177
3.8	37.9939	1.4305	6016	4.5172	4206	14.3051
4	39.9909	2.0148	6332	6.0441	3143	20.1477
4.2	41.9880	3.1991	6648	9.1402	2078	31.9911
4.4	43.9762	6.9104	6964	18.8448	1008	69.1039
4.45	44.4707	9.3477	7042	25.1859	753	93.4768
4.6	44.9994	16	7126	42.6149	445	160

A. 3 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 800 e W = 16

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	<i>n</i>
		Ocupação da janela	Pacotes Transmitidos			
1	9.9995	0.1621	2969	1.04	18311	1.6213
1.2	12	0.2081	3563	1.11	17121	2.0808
1.4	13.9992	0.2608	4156	1.19	15933	2.6084
1.6	15.9983	0.3222	4750	1.29	14741	3.2219
1.8	17.9992	0.3943	5343	1.4	13553	3.9426
2	19.9986	0.4801	5937	1.54	12366	4.8013
2.2	21.9989	0.5845	6531	1.7	11174	5.8452
2.4	23.9983	0.7136	7125	1.9	9984	7.1357
2.6	25.9985	0.8777	7718	2.16	8794	8.7772
2.8	27.9976	1.0936	8312	2.5	7601	10.9357
3	29.9975	1.3910	8906	2.97	6402	13.9102
3.2	31.9966	1.8257	9499	3.65	5203	18.2566
3.4	33.9948	2.5230	10093	4.75	4000	25.2296
3.6	35.9926	3.8203	10686	6.79	2797	38.2028
3.8	37.9911	7.1592	11279	12.06	1576	71.5919
4	39.3930	16	11696	25.98	731	160

A. 4 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 8

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	<i>n</i>
		Ocupação da janela	Pacotes Transmitidos			
1	9.9979	0.0791	1583	0.95	20008	0.7912
1.2	11.9969	0.1005	1900	1.01	18893	1.0054
1.4	13.9958	0.1246	2216	1.07	17781	1.2463
1.6	15.9979	0.1520	2533	1.14	16669	1.5196
1.8	17.9969	0.1831	2850	1.22	15563	1.8309
2	19.9964	0.2192	3166	1.32	14447	2.1915
2.2	21.9979	0.2612	3483	1.42	13337	2.6116
2.4	23.9969	0.3109	3800	1.55	12219	3.1094
2.6	25.9971	0.3705	4116	1.71	11111	3.7046
2.8	27.9979	0.4433	4433	1.9	10000	4.4330
3	29.9969	0.5342	4750	2.14	8890	5.3423
3.2	31.9971	0.6514	5066	2.44	7778	6.5136
3.4	33.9973	0.8077	5383	2.85	6665	8.0770
3.6	35.9962	1.0264	5700	3.42	5553	10.2644
3.8	37.9945	1.3569	6016	4.28	4434	13.5691
4	39.9922	1.9108	6333	5.73	3314	19.1078
4.2	42.3335	3.3731	6703	9.56	1987	33.7314
4.4	44.1581	7.2993	6992	19.82	958	72.9930
4.6	44.3009	8	7014	21.66	877	80

A. 5 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 800 e W = 8

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	<i>n</i>
		Ocupação da janela	Pacotes Transmitidos			
1	9.9995	0.1538	2968	0.98	19298	1.5384
1.2	12	0.1974	3563	1.05	18045	1.9742
1.4	13.9992	0.2475	4156	1.13	16790	2.4753
1.6	15.9983	0.3057	4750	1.22	15536	3.0571
1.8	17.9985	0.3741	5344	1.33	14284	3.7409
2	19.9986	0.4556	5937	1.46	13031	4.5562
2.2	21.9992	0.5548	6531	1.61	11773	5.5477
2.4	23.9983	0.6772	7125	1.81	10521	6.7720
2.6	25.9981	0.8329	7718	2.05	9267	8.3293
2.8	27.9980	1.0373	8312	2.37	8013	10.3729
3	29.9975	1.3188	8906	2.81	6753	13.1876
3.2	31.9966	1.7314	9499	3.46	5486	17.3144
3.4	33.9948	2.3931	10093	4.5	4217	23.9309
3.6	35.9936	3.6266	10686	6.45	2947	36.2657
3.8	38.0510	6.9172	11297	11.63	1633	69.1716
4	38.3795	8	11394	13.33	1424	80

A. 6 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 32

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	n
		Ocupação da janela	Pacotes Transmítidos			
1	9.9979	0.0962	1583	1.15	16462	0.9616
1.2	11.9969	0.1222	1900	1.22	15545	1.2220
1.4	13.9958	0.1515	2216	1.3	14631	1.5146
1.6	15.9979	0.1847	2533	1.39	13718	1.8465
1.8	17.9969	0.2226	2850	1.48	12802	2.2258
2	19.9971	0.2663	3166	1.6	11888	2.6633
2.2	21.9979	0.3174	3483	1.73	10974	3.1739
2.4	23.9969	0.3778	3800	1.89	10058	3.7777
2.6	25.9971	0.4501	4116	2.08	9144	4.5015
2.8	27.9979	0.5386	4433	2.31	8231	5.3859
3	29.9969	0.6495	4750	2.6	7313	6.4950
3.2	31.9977	0.7916	5066	2.97	6400	7.9156
3.4	33.9966	0.9815	5383	3.46	5484	9.8149
3.6	35.9950	1.2483	5699	4.16	4566	12.4829
3.8	37.9939	1.6508	6016	5.21	3644	16.5083
4	39.9916	2.3266	6332	6.98	2722	23.2661
4.2	41.9861	3.6958	6648	10.56	1799	36.9576
4.4	43.9731	8.2778	6963	22.59	841	82.7781
4.6	45.2981	32	7174	84.58	224	320

A. 7 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 800 e W = 32

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	n
		Ocupação da janela	Pacotes Transmitidos			
1	9.9995	0.1870	2969	1.2	15877	1.8698
1.2	12	0.2400	3563	1.28	14846	2.3996
1.4	13.9999	0.3009	4156	1.38	13814	3.0087
1.6	15.9983	0.3715	4750	1.49	12786	3.7146
1.8	17.9988	0.4547	5344	1.62	11752	4.5469
2	19.9986	0.5539	5937	1.77	10719	5.5388
2.2	21.9992	0.6741	6531	1.96	9689	6.7408
2.4	23.9990	0.8230	7125	2.19	8657	8.2297
2.6	25.9982	1.0120	7718	2.49	7627	10.1198
2.8	27.9973	1.2619	8312	2.88	6587	12.6195
3	29.9972	1.6056	8905	3.43	5546	16.0562
3.2	31.9960	2.1063	9499	4.21	4510	21.0626
3.4	33.9944	2.9131	10092	5.48	3464	29.1312
3.6	35.9899	4.4152	10685	7.85	2420	44.1519
3.8	37.9853	8.1960	11278	13.8	1376	81.9600
4	39.8673	32	11837	51.3	370	320

A. 8 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, estações com muito baixo débito e estações com débito elevado

	R (Mb/s)	<i>Goodput</i> total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos
	por fluxo		Ocupação da janela	Pacotes Transmitidos		
9 Estações	1	8.9981	0.3353	1583	---	---
1 Estação	28	28.0017	9.3898	44329	---	---
Total	---	36.9998	---	---	4.02	4721
9 Estações	1	8.9981	0.4346	1583	---	---
1 Estação	30	30.0309	13.0513	47533	---	---
Total	---	39.0290	---	---	5.22	3643
9 Estações	1	8.9981	0.5144	1583	---	---
1 Estação	32	31.0958	16	49232	---	---
Total	---	40.0939	---	---	6.17	3077
8 Estações	1	7.9971	0.3034	1583	---	---
2 Estações	14	28.0611	4.2556	22200	---	---
Total	---	36.0582	---	---	3.64	5217
8 Estações	1	7.9958	0.5136	1583	---	---
2 Estações	16	32.0873	8.2399	25387	---	---
Total	---	40.0831	---	---	6.17	3081
8 Estações	1	7.9977	0.9232	1583	---	---
2 Estações	18	34.6572	16	27432	---	---
Total	---	42.6549	---	---	11.08	1715
7 Estações	1	6.9985	0.2502	1583	---	---
3 Estações	9	26.9994	2.2522	14248	---	---
Total	---	33.9979	---	---	3	6327
7 Estações	1	6.9979	0.5042	1583	---	---
3 Estações	11	32.9987	5.5477	17412	---	---
Total	---	39.9966	---	---	6.05	3139
7 Estações	1	6.9960	1.3109	1583	---	---
3 Estações	13	36.6115	16	19312	---	---
Total	---	43.6075	---	---	15.73	1207

A. 9 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, estações com débito médio e estações com débito elevado

	R (Mb/s)	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos
	por fluxo		Ocupação da janela	Pacotes Transmitidos		
9 Estações	3	26.9969	1.2927	4750	---	---
1 Estação	12	11.9975	5.1702	18990	---	---
Total	---	38.9944	---	---	5.17	3674
9 Estações	3	26.9956	1.8264	4750	---	---
1 Estação	14	14.0109	8.5277	22172	---	---
Total	---	41.0065	---	---	7.31	2600
9 Estações	3	26.9874	3.0118	4749	---	---
1 Estação	16	15.9417	16	25232	---	---
Total	---	42.9291	---	---	12.05	1577
8 Estações	3	23.9937	1.5112	4749	---	---
2 Estações	8	16.0004	4.0299	12664	---	---
Total	---	39.9941	---	---	6.04	3143
8 Estações	3	23.9804	4.7362	4748	---	---
2 Estações	10	20.0116	15.7949	15827	---	---
Total	---	43.9920	---	---	18.94	1002
8 Estações	3	23.9811	4.7918	4748	---	---
2 Estações	12	20.0387	16	15856	---	---
Total	---	44.0198	---	---	19.16	991
7 Estações	3	20.9975	0.9016	4750	---	---
3 Estações	5	14.9987	1.5027	7915	---	---
Total	---	35.9962	---	---	3.6	5268
7 Estações	3	20.9893	2.5541	4750	---	---
3 Estações	7	21.3903	6.0642	11273	---	---
Total	---	42.3796	---	---	10.22	1860
7 Estações	3	20.9779	6.1442	4748	---	---
3 Estações	9	23.4341	16	12357	---	---
Total	---	44.4120	---	---	24.57	773
6 Estações	3	17.9975	1.1287	4750	---	---
4 Estações	5	19.9983	1.8812	7915	---	---
Total	---	37.9958	---	---	4.51	4208
6 Estações	3	17.9855	2.3239	4752	---	---
4 Estações	6	24.0461	4.6494	9502	---	---
Total	---	42.0316	---	---	9.29	2044
6 Estações	3	17.976	7.2059	4749	---	---
4 Estações	7	26.6495	16	10536	---	---
Total	---	44.6255	---	---	28.82	659
5 Estações	3	14.9981	0.8189	4750	---	---
5 Estações	4	19.9996	1.0918	6333	---	---
Total	---	34.9977	---	---	3.28	5800
5 Estações	3	14.9842	1.5764	4753	---	---
5 Estações	5	25.2240	2.6446	7972	---	---
Total	---	40.2082	---	---	6.3	3015
5 Estações	3	14.9773	8.0608	4749	---	---
5 Estações	6	29.7726	16	9418	---	---
Total	---	44.7499	---	---	32.23	589

A. 10 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, estações clientes com débito assimétrico relativamente à estação base

	R (Mb/s)	<i>Goodput</i> total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos
	por fluxo		Ocupação da janela	Pacotes Transmitidos		
EB	1	4.9990	0.0968	1583	---	---
ECs	2	9.9985	0.1936	3166	---	---
Total	---	14.9975	---	---	1.16	16354
EB	1.4	6.9979	0.1555	2216	---	---
ECs	2.4	12	0.2667	3799	---	---
Total	---	18.9979	---	---	1.33	14247
EB	1.8	8.9975	0.2349	2850	---	---
ECs	2.8	13.9990	0.3654	4433	---	---
Total	---	22.9964	---	---	1.57	12132
EB	2.2	10.9990	0.3474	3483	---	---
ECs	3.2	15.9992	0.5054	5066	---	---
Total	---	26.9981	---	---	1.9	10025
EB	2.6	12.9979	0.5201	4116	---	---
ECs	3.6	17.9994	0.7201	5699	---	---
Total	---	30.9973	---	---	2.4	7914
EB	3	14.9981	0.8189	4750	---	---
ECs	4	19.9996	1.0918	6332	---	---
Total	---	34.9977	---	---	3.28	5800
EB	3.4	16.9945	1.4635	5383	---	---
ECs	4.4	21.9992	1.8939	6965	---	---
Total	---	38.9937	---	---	5.17	3678
EB	3.8	18.9796	3.9507	6020	---	---
ECs	4.8	24.0152	4.9854	7593	---	---
Total	---	42.9947	---	---	12.47	1523
EB	4	19.9573	12.8242	6333	---	---
ECs	5	24.96	16	7888	---	---
Total	---	44.9173	---	---	38.46	493

A. 11 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, com atraso de processamento de 0.1ms

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	n
		Ocupação da janela	Pacotes Transmitidos			
0	-	-	-	1.265	-	-
1	9.9979	0.1348	1583	1.618	11746	1.3477
1.4	13.9964	0.2123	2216	1.82	10437	2.1233
1.8	17.9969	0.3122	2850	2.082	9127	3.1222
2.2	21.9979	0.4456	3483	2.431	7817	4.4557
2.6	25.9977	0.6324	4116	2.919	6509	6.3240
3	29.9969	0.9135	4750	3.654	5199	9.1349
3.4	33.9960	1.3838	5383	4.884	3890	13.8383
3.8	37.9901	2.3300	6016	7.358	2582	23.2997
4.2	41.9810	5.2252	6647	14.93	1272	52.2522
4.6	44.4834	16	7045	43.1	440	160

A. 12 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, com atraso de processamento de 0.5ms

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	n
		Ocupação da janela	Pacotes Transmitidos			
0	-	-	-	3.265	-	-
1	9.9979	0.3479	1583	4.175	4551	3.4787
1.4	13.9983	0.5482	2216	4.699	4043	5.4818
1.8	17.9969	0.8058	2850	5.372	3536	8.0576
2.2	21.9954	1.1499	3483	6.272	3029	11.4989
2.6	25.9933	1.6323	4116	7.533	2521	16.3235
3	29.9918	2.3575	4749	9.43	2014	23.5750
3.4	33.9878	3.5693	5382	12.597	1508	35.6931
3.8	37.9781	6.0101	6014	18.97	1001	60.1009
4.2	41.9558	13.3963	6643	38.27	496	133.9625
4.6	42.5090	16	6732	45.06	421	160

A. 13 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, com atraso de processamento de 1ms

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	n
		Ocupação da janela	Pacotes Transmitidos			
0	-	-	-	5.765	-	-
1	9.9985	0.6142	1583	7.371	2577	6.1423
1.4	13.9964	0.9679	2216	8.296	2290	9.6790
1.8	17.9956	1.4229	2849	9.486	2003	14.2288
2.2	21.9916	2.0303	3482	11.07	1715	20.3026
2.6	25.9889	2.8818	4115	13.3	1428	28.8179
3	29.9886	4.1614	4748	16.65	1141	41.6144
3.4	33.9789	6.3051	5380	22.25	853	63.0505
3.8	37.9598	10.5995	6011	33.47	567	105.9954
4.2	40.2720	16	6378	47.6	399	160

A. 14 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, com taxa de erros para tramas de dados de 1%

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	Retransmissões
		Ocupação da janela	Pacotes Transmitidos			
1	9.9979	0.0844	1598	1.004	18931	15
1.4	13.9958	0.1333	2239	1.131	16798	23
1.8	17.9969	0.1961	2877	1.295	14669	28
2.2	21.9979	0.2801	3515	1.514	12548	32
2.6	25.9983	0.3994	4157	1.826	10407	40
3	29.9975	0.5799	4797	2.297	8272	48
3.4	33.9979	0.8839	5435	3.09	6149	52
3.8	37.9958	1.5130	6073	4.733	4014	57
4.2	41.9880	3.6118	6714	10.22	1859	66
4.6	44.5213	14.7050	7122	39.19	484	71

A. 15 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, com taxa de erros para tramas de dados de 10%

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	Retransmissões
		Ocupação da janela	Pacotes Transmitidos			
1	9.9979	0.0960	1764	1.034	18377	181
1.4	13.9958	0.1534	2462	1.184	16052	246
1.8	17.9981	0.2312	3168	1.387	13700	318
2.2	21.9979	0.3396	3865	1.669	11381	382
2.6	25.9990	0.5061	4571	2.104	9030	454
3	29.9981	0.7925	5280	2.852	6662	531
3.4	33.9966	1.3776	5976	4.379	4338	594
3.8	37.9920	3.3472	6673	9.528	1994	658
4.2	40.2335	11.0932	7072	29.78	638	700
4.6	40.2442	11.1522	7074	29.93	634	699

A. 16 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, com taxa de erros para tramas de dados de 10% e taxa de erros para tramas de controlo de 1%

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	Retransmissões
		Ocupação da janela	Pacotes Transmitidos			
1	9.9985	0.2172	1785	2.3	8217	202
1.4	13.9952	0.3483	2502	2.632	7182	286
1.8	17.9987	0.5234	3203	3.09	6120	353
2.2	21.9979	0.7799	3915	3.765	5020	433
2.6	25.9971	1.1622	4644	4.733	3996	528
3	29.9956	1.7590	5349	6.218	3041	600
3.4	33.9853	3.1098	6076	9.666	1954	694
3.8	37.9882	7.6409	6767	21.35	886	751
4.2	38.9236	11.0518	6931	30.06	627	768
4.6	39.0328	11.1833	6954	30.38	622	773

R (Mb/s) por fluxo	Tramas POLL e RET perdidas	Tramas BACK perdidas
1	806	125
1.4	702	108
1.8	603	92
2.2	503	75
2.6	396	59
3	287	44
3.4	184	30
3.8	90	13
4.2	66	10
4.6	56	9

A. 17 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, com modelo de tráfego com tempo ON constante de 0.7s e tempo OFF constante de 0.3s

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos
		Ocupação da janela	Pacotes Transmitidos		
1	6.9979	0.0539	1108	0.923	20566
1.4	9.7958	0.0813	1551	0.995	19085
1.8	12.5969	0.1133	1995	1.079	17608
2.2	15.3979	0.1511	2438	1.178	16133
2.6	18.1983	0.1966	2881	1.296	14659
3	21	0.2523	3325	1.442	13179
3.4	23.7985	0.3219	3768	1.623	11706
3.8	26.5958	0.4121	4211	1.859	10219
4.2	29.4568	0.5374	4662	2.192	8676
4.6	32.1423	0.7380	5090	2.75	6898
7	44.9886	16	7124	42.61	445

A. 18 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, com modelo de tráfego com tempo ON constante de 0.7s e tempo OFF dado por uma distribuição exponencial com média igual a 0.1s

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos
		Ocupação da janela	Pacotes Transmitidos		
1	8.6640	0.0697	1372	0.965	19683
1.4	12.1333	0.1076	1921	1.064	17856
1.8	15.5987	0.1541	2470	1.185	16026
2.2	19.0661	0.2126	3019	1.338	14197
2.6	22.5329	0.2884	3568	1.536	12371
3	25.9971	0.3905	4116	1.802	10541
3.4	29.4638	0.5353	4665	2.18	8715
3.8	32.9293	0.7576	5214	2.76	6882
4.2	36.3960	1.1422	5763	3.766	5046
4.6	39.8432	1.9717	6309	5.935	3200
7	44.9987	16	7126	42.61	445

A. 19 - Resultados com TPAP completo e *standard* IEEE 802.11a, com eliminação dos *gaps* no nível MAC, P = 1500 e W = 16, com modelo de tráfego com tempo ON dado por uma distribuição normal com média 8s e variância 2s e tempo OFF constante igual a 0.2s

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos
		Ocupação da janela	Pacotes Transmitidos		
1	9.8766	0.0822	1562	1	19022
1.4	13.7021	0.1274	2170	1.116	17025
1.8	17.6179	0.1865	2790	1.27	14959
2.2	21.5122	0.2639	3406	1.472	12906
2.6	25.4248	0.3712	4026	1.752	10844
3	29.3653	0.5303	4650	2.167	8767
3.4	33.2811	0.7863	5270	2.835	6702
3.8	37.1943	1.2724	5889	4.104	4629
4.2	41.1044	2.5555	6509	7.458	2547
4.6	44.8693	13.6653	7103	36.5	520
5.5	44.9987	16	7126	42.61	445

A. 20 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11a, para distância de 20m, com P = 1500, com ACKs e retransmissões de nível MAC e mecanismo de RTS/CTS inativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4116	2.5996	2.6002	2.5996	2.5996	2.5996
3	4750	3	2.9994	2.9994	2.9994	3
3.4	5383	3.3998	3.4002	3.3998	3.4002	3.3998
3.8	6017	3.799	3.799	3.8002	3.7996	3.7996
4.2	6650	4.2	4.2	4.2	4.1994	4.2
4.6	7283	4.5556	4.572	4.5903	4.4943	4.5973

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.3996	1.3996	1.3996	1.3996	13.996
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.5996	2.5996	2.5996	2.5996	2.5996	25.9966
3	2.928	2.9021	2.8958	2.8295	2.6678	29.2214
3.4	2.0703	0.9430	3.275	2.9217	2.9924	29.2019
3.8	1.4836	2.2225	1.4968	2.9179	1.4192	28.5374
4.2	1.6952	0.7162	2.5838	0.7181	1.0251	27.7378
4.6	0.9505	0.0979	0.2381	1.1198	2.0406	27.2564

A. 21 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11a, para distância de 20m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS ativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4116	2.5989	2.5996	2.5996	2.5996	2.5989
3	4750	3	3	3	3	3
3.4	5383	3.3998	3.3998	3.3992	3.3998	3.3998
3.8	6017	3.804	3.7888	3.8065	3.7794	3.7895
4.2	6650	3.7648	3.9789	4.0926	4.0623	4.0813
4.6	7283	4.2063	3.9644	3.8116	3.9335	4.0036

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.1474	2.256	2.4158	1.9851	2.3362	24.1371
3	2.4019	2.3109	2.1676	1.4779	0.8615	24.2198
3.4	0.6189	0.132	2.82	1.7583	1.932	24.2596
3.8	0.7497	1.152	0.7301	1.7722	0.7876	24.1598
4.2	0.9947	0.444	1.7867	0.4187	0.468	24.092
4.6	0.8987	0.0891	0.1863	0.9714	2.0343	24.0992

A. 22 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11a, para distância de 200m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS inativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4117	2.5996	2.5996	2.5996	2.5996	2.5996
3	4750	3	3	3	2.9994	2.9994
3.4	5383	3.3992	3.4002	3.3985	3.3998	3.3992
3.8	6017	3.8002	3.7844	3.799	3.8004	3.7996
4.2	6650	4.1994	4.1987	4.2	4.2	4.2
4.6	7283	4.3844	4.5038	4.5758	4.4419	4.5682

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.5996	2.5996	2.5996	2.6002	2.5996	25.9966
3	2.9185	2.8882	2.8838	2.8181	2.6312	29.1386
3.4	1.9958	0.9	3.2697	2.8857	2.9672	29.0153
3.8	1.4438	2.1682	1.4444	2.856	1.4394	28.3354
4.2	1.6541	0.7276	2.5118	0.7465	0.9846	27.6227
4.6	0.9941	0.096	0.2659	1.1785	2.1531	27.1617

A. 23 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11a, para distância de 200m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS ativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.1998	2.1998	2.1998
2.6	4117	2.6002	2.599	2.5996	2.599	2.5996
3	4750	2.9987	2.9994	2.9962	2.9994	3
3.4	5383	3.3985	3.4002	3.3998	3.4	3.4004
3.8	6017	3.799	3.7958	3.7983	3.7699	3.7952
4.2	6650	4.0162	4.1476	4.0333	4.0023	3.9568
4.6	7283	3.9518	4.0011	3.8981	3.9739	4.0244

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.7994	1.8	17.9964
2.2	2.1992	2.1992	2.1998	2.1998	2.1998	21.9968
2.6	2.107	2.2364	2.4518	1.9611	2.3261	24.0798
3	2.388	2.3078	2.1644	1.4375	0.8349	24.1263
3.4	0.6019	0.1194	2.7998	1.7198	1.9257	24.1655
3.8	0.7149	1.1577	0.7175	1.6731	0.792	24.0134
4.2	0.9316	0.4213	1.6813	0.3815	0.4389	24.0108
4.6	0.8912	0.0733	0.1642	1.0067	2.0135	23.9982

A. 24 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11a, para distância de 1500m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS inativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4117	2.5996	2.6002	2.5996	2.6002	2.5996
3	4750	3	3	2.9987	3.0006	3
3.4	5383	3.3998	3.3998	3.3998	3.3998	3.4002
3.8	6017	3.7983	3.7996	3.799	3.7895	3.7996
4.2	6650	4.1987	4.2006	4.1975	4.2	4.1994
4.6	7283	4.5884	4.5954	4.5998	4.5985	4.584

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.599	2.5996	2.6002	2.6002	2.6002	25.9984
3	2.6867	2.7076	2.6368	2.1954	1.7318	26.9576
3.4	1.4192	0.4465	3.036	2.4101	2.6122	26.9234
3.8	1.0528	1.783	1.1255	2.3716	1.044	26.3629
4.2	1.3194	0.5419	1.975	0.5293	0.6903	26.0521
4.6	0.6467	0.0726	0.1585	0.8122	1.5019	26.158

A. 25 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11a, para distância de 1500m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS ativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.1998	2.1998	2.1998
2.6	4117	2.6002	2.5996	2.5996	2.599	2.5996
3	4750	2.9994	2.9994	2.9994	2.9987	3
3.4	5383	3.3998	3.4004	3.3998	3.3998	3.3992
3.8	6017	3.7996	3.8002	3.7926	3.8002	3.7996
4.2	6650	4.1091	4.1356	4.1533	4.14	4.1356
4.6	7283	4.2234	4.1356	4.1526	4.2278	4.2613

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.7994	1.8	17.9964
2.2	2.1992	2.1992	2.1998	2.1998	2.1998	21.9968
2.6	1.7994	2.0097	2.2491	1.6838	2.0457	22.7857
3	2.0141	2.1107	1.9472	1.0554	0.6291	22.7534
3.4	0.4055	0.0688	2.412	1.3686	1.5417	22.7956
3.8	0.5375	0.8583	0.528	1.3421	0.5949	22.853
4.2	0.5583	0.2798	0.9916	0.2192	0.2425	22.965
4.6	0.4225	0.0474	0.0872	0.4415	0.9606	22.9599

A. 26 - Resultados com TPAP completo e *standard* IEEE 802.11n, com eliminação dos *gaps* no nível MAC

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	n
		Ocupação da janela	Pacotes Transmitidos			
0	-	-	-	1.1233	-	-
0.01	0.0998	0.0009	16	1.1261	16871	0.0094
0.2	1.9990	0.0195	317	1.1703	16234	0.1950
0.4	3.9998	0.0407	633	1.2217	15552	0.4072
0.6	6	0.0639	950	1.2779	14868	0.6390
0.8	8.0021	0.0893	1267	1.3392	14187	0.8930
1	9.9979	0.1172	1583	1.407	13505	1.1721
1.2	11.9994	0.1482	1900	1.4817	12822	1.4817
1.4	13.9958	0.1825	2216	1.565	12141	1.8252
1.6	15.9985	0.2211	2533	1.6578	11460	2.2106
1.8	17.9981	0.2645	2850	1.763	10775	2.6446
2	19.9983	0.3137	3167	1.8821	10095	3.1367
2.2	21.9979	0.3700	3483	2.018	9414	3.6998
2.4	23.9969	0.4352	3800	2.1758	8732	4.3515
2.6	25.9971	0.5113	4116	2.36	8050	5.1134
2.8	27.9966	0.6018	4433	2.5790	7367	6.0177
3	29.9969	0.7105	4750	2.842	6685	7.1050
3.2	31.9933	0.8436	5066	3.1637	6005	8.4360
3.4	33.9966	1.0117	5383	3.57	5321	10.1165
3.6	35.9950	1.2296	5699	4.0988	4635	12.2960
3.8	37.9939	1.5236	6016	4.81	3949	15.2355
4	39.9929	1.9411	6332	5.82	3262	19.4114
4.2	41.9906	2.5824	6649	7.38	2575	25.8242
4.4	43.9857	3.6915	6966	10.07	1887	36.9152
4.6	45.9771	6.0843	7283	15.87	1197	60.8433
4.7	46.9617	8.7170	7437	22.2531	853	87.1703
4.75	47.4575	11.0345	7515	27.8725	681	110.3452
6	48.0303	16	7606	39.92	475	160

A. 27 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11n, para distância de 20m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS inativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4116	2.5996	2.6002	2.5996	2.5996	2.5996
3	4750	3	3	2.9987	2.9994	3
3.4	5383	3.3998	3.3998	3.3998	3.3998	3.4004
3.8	6017	3.8002	3.7996	3.8002	3.7996	3.8002
4.2	6650	4.2	4.2	4.2	4.2006	4.2
4.6	7283	4.5992	4.6	4.5985	4.6	4.5897

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.5996	2.5996	2.6002	2.5996	2.5996	25.9972
3	2.9994	3	3	3	3	29.9975
3.4	2.7979	1.9213	3.3316	3.2204	3.2463	31.5171
3.8	2.04	2.7493	2.0248	3.3556	1.9194	31.0889
4.2	2.3387	1.0257	3.0891	1.164	1.5758	30.1939
4.6	0.9556	0.2008	0.6076	1.8916	2.9444	29.5874

A. 28 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11n, para distância de 20m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS ativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4116	2.5996	2.6002	2.5996	2.599	2.5996
3	4750	3	2.9987	2.9994	3	2.9994
3.4	5383	3.3998	3.4004	3.3992	3.4004	3.3992
3.8	6017	3.7971	3.799	3.799	3.8008	3.7996
4.2	6650	4.1994	4.199	4.1634	4.2	4.2
4.6	7283	4.4526	4.2764	4.3794	4.4211	4.3983

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.5996	2.5996	2.5996	2.6002	2.5996	25.9966
3	2.731	2.6691	2.5768	2.0956	1.3971	26.4671
3.4	1.0743	0.3196	3.1731	2.4594	2.5592	26.5846
3.8	1.2221	1.6295	0.9909	2.4613	1.171	26.4703
4.2	1.2208	0.6644	1.8461	0.708	0.8728	26.2739
4.6	0.5842	0.0992	0.3044	1.1905	2.1985	26.3046

A. 29 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11n, para distância de 200m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS inativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4116	2.5996	2.5996	2.5996	2.5996	2.5996
3	4750	3	3	3	2.9994	3
3.4	5383	3.3998	3.3998	3.3998	3.3992	3.4004
3.8	6017	3.799	3.7996	3.7983	3.8002	3.7996
4.2	6650	4.2006	4.1968	4.2	4.1987	4.2
4.6	7283	4.5998	4.59853	4.59284	4.5998	4.5973

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.5996	2.5996	2.5996	2.5996	2.5996	25.996
3	2.9994	2.9994	3	3	3	29.9982
3.4	2.7524	1.891	3.3246	3.1945	3.223	31.3845
3.8	1.9977	2.7133	2.0034	3.3246	1.8783	30.914
4.2	2.2086	1.0333	3.0493	1.1419	1.5	29.9292
4.6	0.9196	0.1989	0.6032	1.8006	2.8775	29.3881

A. 30 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11n, para distância de 200m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS ativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4116	2.5996	2.5996	2.5996	2.5996	2.5996
3	4750	3	2.9981	2.9994	2.9994	2.9994
3.4	5383	3.3998	3.3998	3.399	3.3998	3.3998
3.8	6017	3.799	3.7996	3.8008	3.8002	3.799
4.2	6650	4.1867	4.2	4.188	4.2	4.2
4.6	7283	4.4615	4.4659	4.2467	4.4419	4.147

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.599	2.5996	2.5996	2.6002	2.5996	25.996
3	2.7379	2.6356	2.5566	2.0931	1.3472	26.3667
3.4	1.0282	0.2722	3.1661	2.4341	2.4916	26.3904
3.8	1.2366	1.5758	0.9177	2.4114	1.2095	26.3496
4.2	1.147	0.6411	1.8013	0.7383	0.8842	26.1866
4.6	0.5697	0.1011	0.2893	1.1792	2.2446	26.1469

A. 31 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11n, para distância de 1500m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS inativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4116	2.5996	2.5996	2.5996	2.5996	2.5996
3	4750	3.0006	2.9994	2.9994	2.9994	2.9994
3.4	5383	3.3998	3.4004	3.3992	3.3998	3.3998
3.8	6017	3.799	3.7971	3.799	3.8002	3.7983
4.2	6650	4.1987	4.1987	4.1994	4.2	4.2
4.6	7283	4.5979	4.5998	4.5954	4.5998	4.5998

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.599	2.5996	2.6002	2.5996	2.5996	25.996
3	2.8939	2.9027	2.892	2.7947	2.6286	29.1101
3.4	2.0343	1.0131	3.2027	2.82	2.9425	29.0116
3.8	1.4223	2.311	1.5316	2.9381	1.3547	28.5513
4.2	1.6074	0.8912	2.2017	1.0131	1.1684	27.8786
4.6	0.6417	0.1427	0.4402	1.404	2.2844	27.9057

A. 32 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11n, para distância de 1500m, com P = 1500, com ACKs e retransmissões de nível MAC e com mecanismo de RTS/CTS ativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4116	2.5996	2.6002	2.5996	2.599	2.599
3	4750	3	2.9994	2.9994	3	3
3.4	5383	3.3998	3.3998	3.4004	3.3998	3.3992
3.8	6017	3.7971	3.7996	3.7996	3.8002	3.8002
4.2	6650	4.1987	4.2	4.1994	4.2	4.1987
4.6	7283	4.5998	4.5998	4.4495	4.4905	4.579

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.304	2.3754	2.4777	2.2118	2.3962	24.7625
3	2.4272	2.4663	2.335	1.6093	0.9227	24.7593
3.4	0.6581	0.1762	2.9261	2.0274	2.0166	24.8034
3.8	0.9531	1.1792	0.6227	2.0021	1.0358	24.7896
4.2	0.8602	0.4699	1.2701	0.5387	0.6429	24.7786
4.6	0.2545	0.0467	0.1219	0.5741	1.2145	24.9303

A. 33 - Resultados com o protocolo TPAP simples, com *standard* IEEE 802.11n, com agregação e BACK de nível MAC e com tramas POLL e RET explícitas

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	<i>n</i>
		Ocupação da janela	Pacotes Transmitidos			
0	-	-	-	1.7874	-	-
0.01	0.1017	0.0015	16	1.7932	10595	0.0152
0.2	1.9958	0.0317	316	1.9084	9955	0.3174
0.4	3.9979	0.0681	633	2.0441	9294	0.6811
0.6	5.9968	0.1102	950	2.2045	8618	1.1018
0.8	7.9958	0.1592	1266	2.3890	7953	1.5919
1	9.9979	0.2173	1583	2.6078	7285	2.1729
1.2	11.9969	0.2775	1900	2.7753	6846	2.7748
1.4	13.9971	0.3535	2216	3.0304	6269	3.5349
1.6	15.9979	0.4339	2533	3.2547	5838	4.3392
1.8	17.9969	0.5444	2850	3.6295	5234	5.4441
2	19.9971	0.6728	3166	4.0368	4706	6.7278
2.2	21.9973	0.8463	3483	4.6161	4115	8.4631
2.4	23.9962	1.0501	3800	5.2506	3618	10.5014
2.6	25.9952	1.2375	4116	5.7116	3326	12.3751
2.8	27.9960	1.4557	4433	6.2384	3045	14.5573
3	29.9937	1.7264	4749	6.9056	2751	17.2644
3.2	31.9920	2.0376	5066	7.6408	2486	20.3765
3.4	33.9916	2.3191	5382	8.185	2321	23.1912
3.6	35.9906	2.6387	5699	8.7954	2160	26.3873
3.8	37.9895	3.0068	6015	9.495	2000	30.0675
4	39.9866	3.4541	6332	10.3616	1833	34.5409
4.2	41.9893	3.9830	6648	11.38	1669	39.8304
4.4	43.9838	4.6124	6965	12.5781	1510	46.1239
4.6	45.9821	5.4192	7281	14.14	1344	54.1917
4.8	47.9779	6.4347	7597	16.0852	1181	64.3470
5	49.9737	7.8465	7913	18.8301	1009	78.4650
5.2	51.9657	9.7350	8228	22.4634	845	97.3497
5.4	53.9507	12.4592	8543	27.6836	686	124.5921
6	55.6193	16	8808	34.48	550	160

A. 34 - Resultados com o protocolo TPAP simples, com *standard* IEEE 802.11n, com agregação e BACK de nível MAC e com tramas POLL e RET substituídas por uma *flag* na última trama de dados, caso exista

R (Mb/s) por fluxo	Goodput total (Mb/s)	Pacotes por fluxo		Ciclo (ms)	Número de ciclos	<i>n</i>
		Ocupação da janela	Pacotes Transmitidos			
0.01	0.1017	0.0014	16	1.627	11674	0.0138
0.2	1.9958	0.0271	316	1.632	11643	0.2714
0.4	3.9979	0.0562	633	1.686	11267	0.5618
0.6	5.9975	0.0872	950	1.745	10887	0.8722
0.8	7.9958	0.1204	1266	1.808	10511	1.2045
1	9.9979	0.1471	1583	1.766	10758	1.4715
1.2	11.9969	0.1774	1900	1.774	10708	1.7739
1.4	13.9958	0.2087	2216	1.789	10619	2.0868
1.6	15.9979	0.2313	2533	1.735	10952	2.3128
1.8	17.9969	0.2726	2850	1.817	10453	2.7259
2	19.9964	0.3155	3166	1.893	10034	3.1554
2.2	21.9979	0.3527	3483	1.924	9874	3.5274
2.4	23.9969	0.3886	3800	1.943	9779	3.8856
2.6	25.9971	0.4450	4116	2.054	9250	4.4500
2.8	27.9985	0.5040	4433	2.160	8796	5.0400
3	29.9969	0.5742	4750	2.297	8271	5.7423
3.2	31.9977	0.6486	5066	2.432	7811	6.4864
3.4	33.9973	0.7380	5383	2.605	7294	7.3802
3.6	35.9969	0.8344	5700	2.781	6831	8.3441
3.8	37.9971	0.9533	6016	3.010	6311	9.5330
4	39.9960	1.1596	6333	3.479	5461	11.5958
4.2	41.9937	1.5778	6649	4.508	4214	15.7779
4.4	43.9920	2.0666	6966	5.636	3371	20.6664
4.6	45.9916	2.4227	7282	6.320	3006	24.2268
4.8	47.9912	2.8762	7599	7.190	2642	28.7619
5	49.9870	3.4791	7915	8.350	2275	34.7908
5.2	51.9859	4.3069	8231	9.939	1911	43.0688
5.4	53.9754	5.5291	8547	12.286	1546	55.2911
5.6	55.9712	7.6027	8863	16.291	1166	76.0273
5.8	57.9562	11.4632	9177	23.714	801	114.6316
6	59.1461	16	9366	32.430	585	160

A. 35 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11n, para distância de 20m, com P = 1500, com agregação e BACK de nível MAC e com mecanismo de RTS/CTS inativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4117	2.5996	2.4461	1.884	2.5996	2.5996
3	4750	3	1.8568	1.9244	1.8152	1.7305
3.4	5383	3.3998	1.7975	1.8259	1.8246	1.8827
3.8	6017	1.5802	1.7817	1.6743	1.86	1.9693
4.2	6650	3.9171	3.9897	3.9575	4.1463	3.9158
4.6	7283	4.5284	3.9808	4.2512	4.0832	3.7554

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.3996	1.399	1.3996	1.3996	1.3996	13.996
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.5996	2.5996	2.6002	2.5996	1.7697	24.2976
3	2.9987	2.9994	1.8234	2.3154	2.988	23.4518
3.4	1.6939	1.8025	3.4004	1.7406	1.8107	21.1786
3.8	1.836	1.5076	1.6598	1.6977	1.9099	17.4765
4.2	4.2	4.1899	4.1798	4.2	4.2	40.8961
4.6	4.4015	4.4046	4.3907	4.4236	4.4602	42.6796

A. 36 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11n, para distância de 200m, com P = 1500, com agregação e BACK de nível MAC e com mecanismo de RTS/CTS inativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4117	2.5996	1.9636	2.0665	1.9977	2.599
3	4750	3	1.8215	1.8221	1.8688	1.7002
3.4	5383	3.3998	1.7488	1.8884	1.8158	1.8474
3.8	6017	1.6232	1.8044	1.6888	1.8436	1.9352
4.2	6650	3.9505	3.9013	3.8924	4.1779	3.3701
4.6	7283	4.128	4.111	4.1804	3.7857	3.7958

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	2.2225	2.5996	2.6002	2.599	1.8846	23.1323
3	2.6078	2.9994	1.9181	1.6453	1.7406	21.1238
3.4	1.7785	3.3998	3.4004	1.728	1.7337	22.7406
3.8	1.7905	1.5183	1.668	1.7059	2.0229	17.6008
4.2	4.1798	4.2	4.2	4.1949	4.2	40.2669
4.6	4.5543	4.5777	4.5587	4.5726	4.5878	42.852

A. 37 - Resultados sem o protocolo TPAP, com *standard* IEEE 802.11n, para distância de 1500m, com P = 1500, com agregação e BACK de nível MAC e com mecanismo de RTS/CTS inativo

R (Mb/s)	Pacotes Transmitidos	Goodput por fluxo (Mb/s)				
		EC1 → EB	EC2 → EB	EC3 → EB	EC4 → EB	EC5 → EB
1	1583	0.9998	1.0004	0.9992	0.9992	0.9998
1.4	2216	1.399	1.3996	1.4002	1.3996	1.4002
1.8	2850	1.8	1.8	1.7994	1.7994	1.7994
2.2	3483	2.1998	2.1998	2.2004	2.1998	2.1998
2.6	4117	2.5996	2.5996	1.7564	2.1657	2.599
3	4750	3	1.7665	1.8158	1.8303	2.0545
3.4	5383	1.7615	1.6459	1.8196	1.6421	1.836
3.8	6017	1.7861	1.5688	1.6983	1.7419	1.6819
4.2	6650	3.7143	4.1337	4.2	2.892	3.6808
4.6	7283	3.4611	4.0933	4.599	4.1248	3.9575

R (Mb/s)	Goodput por fluxo (Mb/s)					Goodput Total (Mb/s)
	EB → EC1	EB → EC2	EB → EC3	EB → EC4	EB → EC5	
1	0.9992	0.9998	0.9998	0.9998	0.9998	9.9968
1.4	1.399	1.399	1.3996	1.3996	1.3996	13.9954
1.8	1.7994	1.7994	1.8	1.8	1.8	17.997
2.2	2.1992	2.1992	2.1998	2.1998	2.1992	21.9968
2.6	1.9434	2.5996	0.1983	2.2964	2.1139	20.8719
3	2.9994	2.9994	1.89095	3	2.419	23.7759
3.4	1.6958	3.3998	3.4004	1.6244	1.8392	20.6647
3.8	1.7684	1.5505	1.7451	1.5347	1.8057	16.8814
4.2	4.2	4.0958	4.2	4.2	4.2	39.5166
4.6	4.3882	4.2531	4.4432	4.3888	4.4406	42.1496

Referências

- [1] (07/07/2015). *ns-3*. Available: <https://www.nsnam.org/>
- [2] J. H. Schiller, *Mobile communications*: Pearson Education, 2003.
- [3] IEEE, "*IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*," ed, 2012.
- [4] J. Geier, *Wireless LANs, Second Edition*, 2002.
- [5] IEEE. (2015, 10/07/2015). *OFFICIAL IEEE 802.11 WORKING GROUP PROJECT TIMELINES*. Available: http://www.ieee802.org/11/Reports/802.11_Timelines.htm
- [6] D. Skordoulis, Q. Ni, H.-H. Chen, A. P. Stephens, C. Liu, and A. Jamalipour, "IEEE 802.11 n MAC frame aggregation mechanisms for next-generation high-throughput WLANs," *Wireless Communications, IEEE*, vol. 15, pp. 40-47, 2008.
- [7] E. H. Ong, J. Knecht, O. Alanen, Z. Chang, T. Huovinen, and T. Nihtila, "IEEE 802.11 ac: Enhancements for very high throughput WLANs," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2011 IEEE 22nd International Symposium on*, 2011, pp. 849-853.
- [8] C.-Y. Wang and H.-Y. Wei, "IEEE 802.11 n MAC enhancement and performance evaluation," *Mobile Networks and Applications*, vol. 14, pp. 760-771, 2009.
- [9] E. Perahia and M. X. Gong, "Gigabit wireless LANs: an overview of IEEE 802.11 ac and 802.11 ad," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 15, pp. 23-33, 2011.
- [10] G. Kramer, B. Mukherjee, and G. Pesavento, "IPACT a dynamic protocol for an Ethernet PON (EPON)," *Communications Magazine, IEEE*, vol. 40, pp. 74-80, 2002.
- [11] H. Miyoshi, T. Inoue, and K. Yamashita, "D-CRED: Efficient Dynamic Bandwidth Allocation Algorithm in Ethernet Passive Optical Networks," *SEI TECHNICAL REVIEW-ENGLISH EDITION*, pp. 34-41, 2004.
- [12] A. Sabatini, "Highly Efficient Selective Acknowledgement (SACK) for TCP," 2012.
- [13] T. Nakajima, Y. Utsunomiya, Y. Nishibayashi, T. Tandai, T. Adachi, and M. Takagi, "Compressed Block Ack, an efficient selective repeat mechanism for IEEE802. 11n," in *Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on*, 2005, pp. 1479-1483.
- [14] N. I. Sarkar and S. A. Halim, "A review of simulation of telecommunication networks: simulators, classification, comparison, methodologies, and recommendations," 2011.
- [15] S. Siraj, A. Gupta, and R. Badgujar, "Network simulation tools survey," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 1, pp. 199-206, 2012.

- [16] S. M. Bilal and M. Othmana, "A Performance Comparison of Network Simulators for Wireless Networks," *arXiv preprint arXiv:1307.4129*, 2013.
- [17] S. G. Gupta, M. M. Ghonge, P. D. Thakare, and P. Jawandhiya, "Open-Source Network Simulation Tools: An Overview," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 2, pp. 1629-1635, 2013.
- [18] R. Chaudhary, S. Sethi, R. Keshari, and S. Goel, "A study of comparison of Network Simulator-3 and Network Simulator-2," *IJCSIT International Journal of Computer Science and Information Technologies*, vol. 3, pp. 3085-3092, 2012.
- [19] O. T. T. Kim, V. Nguyen, and C. S. Hong, "Which network simulation tool is better for simulating Vehicular Ad-hoc network?," *2014 한국정보과학회 제 41 회 정기총회 및 동계학술발표회*, pp. 930-932, 2014.
- [20] (07/07/2015). *ns-3 Manual*. Available: <https://www.nsnam.org/docs/release/3.23/manual/singlehtml/index.html>
- [21] (17/07/2015). *ns-3 Wi-Fi Module*. Available: <https://www.nsnam.org/docs/models/html/wifi-design.html>
- [22] J. Ruela, M. Lopes, and R. Campos, "Especificações técnicas: Especificação de *software* (protocolo TPAP)," Relatório técnico interno, Projeto HiperWireless, 2014.
- [23] J. Ruela, "Especificações técnicas: Protocolo TPAP - *smart polling*," Relatório técnico interno, Projeto HiperWireless, 2015.