UNIVERSIDADE FEDERAL FLUMINENSE ESCOLA DE ENGENHARIA MESTRADO EM ENGENHARIA DE TELECOMUNICAÇÕES

EDUARDO POMPEO DA SILVA MINEIRO

CONTROLE DE ADMISSÃO PARA PROVISÃO DE QOS EM REDES EM MALHA SEM FIO

EDUARDO POMPEO DA SILVA MINEIRO

CONTROLE DE ADMISSÃO PARA PROVISÃO DE QOS EM REDES EM MALHA SEM FIO

Dissertação apresentada ao Curso de Pós-Graduação "Stricto Sensu" em Engenharia de Telecomunicações da Universidade Federal Fluminense, como requisito parcial para obtenção do Grau de Mestre. Área de concentração: Sistemas de Telecomunicações.

Orientadora: Profa. Dra. DÉBORA CHRISTINA MUCHALUAT SAADE

NITERÓI 2011

Ficha Catalográfica elaborada pela Biblioteca da Escola de Engenharia e Instituto de Computação da UFF

M664 Mineiro, Eduardo Pompeo da Silva

Controle de admissão para provisão de QOS em redes em malha sem fio / Eduardo Pompeo da Silva Mineiro. – Niterói, RJ: [s.n.], 2011.

124 f.

Dissertação (Mestrado em Engenharia de Telecomunicações) - Universidade Federal Fluminense, 2011.

Orientador: Débora Christina Muchaluat Saade.

1. Rede Mesh. 2. Rede sem fio. 3. Controle de admissão (Engenharia de Telecomunicações). 4. Roteamento. 5. Qualidade de serviço. I. Título.

CDD 621.38216

CONTROLE DE ADMISSÃO PARA PROVISÃO DE QOS EM REDES EM MALHA SEM FIO

EDUARDO POMPEO DA SILVA MINEIRO

Dissertação apresentada ao Curso de Pós-Graduação "Stricto Sensu" em Engenharia de Telecomunicações da Universidade Federal Fluminense, como requisito parcial para obtenção do Grau de Mestre. Área de concentração: Sistemas de Telecomunicações.

Aprovada em Dezembro/2011

Profa. Dra. Débora Christina Muchaluat Saade Universidade Federal Fluminense (Orientadora)

Prof. Dr. Luiz Claudio Schara Magalhães Universidade Federal Fluminense

Prof. Dr. José Ferreira de Rezende COPPE / Universidade Federal do Rio de Janeiro

Niterói, 01 de dezembro de 2011

Dedicatória

Dedico esta dissertação a Deus, a meus pais e irmãos e à minha amada esposa.

Agradecimentos

Agradeço a Deus em primeiro lugar pela sabedoria com que me capacitou durante todo este trabalho. A Ele toda honra, glória e louvor.

Agradeço à minha querida esposa, Ruth, pela paciência, compreensão e incentivo. Sem o seu apoio eu não teria chegado até este momento.

Agradeço a meus pais, Alípio e Alina, pela criação e formação que me proporcionaram. Certamente os vôos que hoje alço são consequência da base sólida que eles me ajudaram a construir.

Agradeço a meus irmãos, Fábio e Renato, amigos verdadeiros que Deus colocou em minha vida.

Agradeço a meus chefes da Petrobras, Faustino e Omar, por incentivar o meu desenvolvimento acadêmico, bem como por terem me proporcionado tempo para que eu pudesse me dedicar ao mestrado.

Agradeço também aos colegas que conheci na UFF, em especial os do Midiacom, pela amizade. Particularmente ao Diego Passos, que com enorme presteza sempre me ajudou nas minhas dúvidas de programação.

Agradeço à Professora Débora, orientadora do trabalho, pelas críticas e sugestões de melhorias sempre oportunas com vistas à excelência do trabalho. Sem dúvida, um exemplo de dedicação e comprometimento.

SUMÁRIO

1.	INT	TRODUÇAO	1
	1.1.	MOTIVAÇÃO	3
	1.2.	OBJETIVOS	4
	1.3.	ESTRUTURA DA DISSERTAÇÃO	5
2.	PAE	DRÃO IEEE 802.11E	6
	2.1.	ACESSO AO MEIO	6
	2.1.1.	HCCA	7
	2.1.2.	EDCA	7
	2.2.	CATEGORIAS DE ACESSO	9
	2.3.	PARÂMETROS DE ACESSO AO MEIO DA FUNÇÃO EDCA	10
		PROCEDIMENTO DE BACKOFF	
	2.5.	RESUMO DO CAPÍTULO	12
3.	TRA	ABALHOS RELACIONADOS	13
	3.1.	MECANISMOS INDEPENDENTES DE PROTOCOLO DE	
	ROTE	EAMENTO	13
	3.2.	MECANISMOS ASSOCIADOS A PROTOCOLOS DE ROTEAMEN	TO
	REAT	TIVOS	15
	3.3.	MECANISMOS ASSOCIADOS A PROTOCOLOS DE ROTEAMEN	TO
	PRÓ-	ATIVOS	17
	3.4.	COMPARAÇÃO DOS TRABALHOS RELACIONADOS COM A	
	PROP	POSTA DESTA DISSERTAÇÃO	17
4.	CAC	C-OLSR	19
	4.1.	MEDIÇÃO DA OCUPAÇÃO DO CANAL	19
	4.2.	ESTIMATIVA DE OCUPAÇÃO DO CANAL POR UM NOVO FLUXO	20
	4.2.	.1. CÁLCULO DE "SIFS" E "AIFS"	22
	4.2.	.2. CÁLCULO DE "ACK"	22
	4.2.	.3. CÁLCULO DE "Data"	23
	4.2.	.4. ESTIMATIVA DE "Backoff"	23
	4.3.	TRATAMENTO DE INTERFERÊNCIAS	25
	4.3.	.1. INTERFERÊNCIA INTERFLUXO	25
	4.3.	.2. INTERFERÊNCIA INTRAFLUXO	27
	4.4.	VIOLAÇÃO DE QOS	28

4.5.	POLÍTICA DE ADMISSÃO	30
4.6.	RESUMO DO CAPÍTULO	33
5. CEN	ÁRIOS PARA ANÁLISE DE DESEMPENHO DO CAC-OLSR	34
5.1.	ANÁLISE DE MÉTRICAS DE QOS	34
5.1.1	. SIMULADOR	34
5.1.2	PERFIL DE TRÁFEGO	36
5.1.3	. CENÁRIO CONFIGURADO	37
5.1.4	CONFIGURAÇÕES DE PARÂMETROS DO CAC-OLSR	39
5.1.5	. MÉTRICAS AVALIADAS	39
5.2.	TRANSMISSÃO DE FLUXO DE VÍDEO REAL	40
5.2.1	. SIMULADOR	41
5.2.2	PERFIL DE TRÁFEGO	43
5.2.3	. CENÁRIO CONFIGURADO	44
5.2.4	CONFIGURAÇÕES DE PARÂMETROS DO CAC-OLSR	44
5.2.5	. MÉTRICAS AVALIADAS	44
6. RES	ULTADOS DA ANÁLISE DE DESEMPENHO	46
6.1.	ANÁLISE DE MÉTRICAS DE QOS	46
6.1.1	. FLUXOS ADMITIDOS	47
6.1.2		
6.1.3	. VAZÃO MÉDIA AGREGADA	51
6.1.4	. VAZÃO MÉDIA POR FLUXO	56
6.1.5	. OVERHEAD ADICIONAL DE CONTROLE	58
6.1.6	. REAÇÃO A VIOLAÇÕES DE QOS	59
6.2.	TRANSMISSÃO DE VÍDEO REAL	62
6.2.1	. PSNR MÉDIO	63
6.2.2	MOS MÉDIO	64
6.2.3	. ATRASO MÉDIO FIM-A-FIM	65
6.2.4	. PERDA MÉDIA DE PACOTES	67
6.3.	ANÁLISE CONSOLIDADA DOS RESULTADOS	69
7. CON	CLUSÃO	71
7.1.	CONTRIBUIÇÕES	72
7.2.	TRABALHOS FUTUROS	73
8. REF	ERÊNCIAS BIBLIOGRÁFICAS	75
ANEXO A	4	79

A.1. SCRIPTS TCL PARA O TESTE "ANÁLISE DE MÉTRICAS DE QOS"	. 79
A.1.1. CENÁRIO CAC-OLSR	. 79
A.1.2. CENÁRIO OLSR ORIGINAL	. 82
A.1.3. CENÁRIO SWAN	. 85
A.1.4. CONFIGURAÇÃO DO TRÁFEGO	. 89
A.1.5. REAÇÃO À VIOLAÇÃO DE QOS	. 91
A.2. SCRIPTS TCL PARA O TESTE "TRANSMISSÃO DE VÍDEO REAL" .	. 95
A.2.1. CENÁRIO CAC-OLSR (EVALVID)	. 95
A.2.2. CENÁRIO OLSR ORIGINAL (EVALVID)	. 99
A.2.3. CENÁRIO SWAN (EVALVID)	103
A.2.4. CONFIGURAÇÃO DO TRÁFEGO (EVALVID)	108

LISTA DE FIGURAS

Figura 1: Operação híbrida nos modos HCCA e EDCA	7
Figura 2: IFSs e suas relações	9
Figura 3: Categorias de Acesso do padrão IEEE 802.11e	10
Figura 4: Ocupação do meio na transmissão de um quadro IEEE 802.11	21
Figura 5: Valor Aleatório médio para Backoff	25
Figura 6: Cenário de interferência interfluxo	26
Figura 7: Cenário de interferência intrafluxo	27
Figura 8: Mecanismo de violação de QoS	29
Figura 9: Política de Admissão CAC-OLSR.	30
Figura 10: Cenário para avaliação de violação de QoS	38
Figura 11: Cenário para avaliação de violação de QoS após 300 s	38
Figura 12: Arquitetura do conjunto de ferramentas Evalvid	41
Figura 13: Fluxos de voz admitidos	47
Figura 14: Fluxos de vídeo admitidos	48
Figura 15: Atraso médio fim-a-fim para fluxos de voz	50
Figura 16: Atraso médio fim-a-fim para fluxos de vídeo	50
Figura 17: Vazão média agregada para fluxos de voz	52
Figura 18: Vazão média agregada para fluxos de vídeo	53
Figura19: Vazão média agregada para fluxos sem QoS	55
Figura 20: Vazão média agregada para todos os fluxos	56
Figura 21: Vazão média por fluxo para voz	57
Figura 22: Vazão média por fluxo para vídeo	57
Figura 23: Atraso médio fim-a-fim para voz	60
Figura 24: Atraso médio fim-a-fim para vídeo	60
Figura 25: Vazão média por fluxo para voz	61
Figura 26: Vazão média por fluxo para vídeo	62
Figura 27: PSNR Médio	63
Figura 28: MOS Médio	64
Figura 29: Atraso (PDF)	66
Figura 30: Atraso (CDF)	67
Figura 31: Perda média de pacotes	68

LISTA DE TABELAS

l'abela 1: Mapeamento sugerido entre prioridades e ACs	10
Tabela 2: Valores sugeridos para os parâmetros das categorias de acesso	11
Tabela 3: Protocolos de roteamento e camadas MAC e PHY	35
Tabela 4: Configuração dos parâmetros das categorias de acesso 802.11e	36
Tabela 5: Configuração alternativa dos parâmetros das categorias de acesso	
802.11e	36
Tabela 6: Escala MOS	42
Tabela 7: Conversão PSNR - MOS	43
Tabela 8: Correspondência entre cenários e legenda	46
Tabela 9: Overhead Adicional	59

LISTA DE ACRÔNIMOS

AC Access Category

AIFS Arbitration InterFrame Space

AIFSN Arbitration InterFrame Space Number

AIMD Additive Increase, Multiplicative Decrease

CAC-OLSR Call Admission Control OLSR

CBR Constant Bit Rate

CDF Cumulative Distribution Function

CS Carrier Sensing

CSMA/CA Carrier Sense Multiple Access with Collision Avoidance

CTS Clear to Send

CW Contention Window

DCF Distributed Coordination Function

DIFS DCF InterFrame Space

DSCP Differentiated Services Code Point

EDCA Enhanced Distributed Channel Access

ERP-OFDM Extended Rate PHY - OFDM

GOP Group of Pictures

HCCA HCF Controlled Channel Access

HCF Hybrid Coordination Function

HSPA High Speed Packet Access

IEEE Institute of Electrical and Electronics Engineers

IP Internet Protocol

LTE Long Term Evolution

MOS Mean Opinion Score

MSDU MAC Service Data Unit

OFDM Orthogonal Frequency Division Multiplexing

OLSR Optimized Link State Routing

PCF Point Coordination Function

PDF Probability Density Function

PLCP Physical Layer Convergence Procedure

PSNR Peak Signal-to-Noise Ratio

QoS Quality of Service

RTS Request to Send

SIFS Short InterFrame Space

SWAN Service Differentiation in Stateless Wireless Ad Hoc Networks

TCL Tool Command Language

ToS Type of Service

TXOP Transmission Opportunity

UP User Priority

VoIP Voice over IP

WiMAX Worldwide Interoperability for Microwave Access

RESUMO

Esta dissertação apresenta um mecanismo de controle de admissão para redes em malha sem fio compostas de múltiplos saltos que operam no modo de acesso baseado em contenção do padrão IEEE 802.11 e utilizam o protocolo de roteamento OLSR. A proposta, denominada CAC-OLSR, tem por objetivo garantir que fluxos com qualidade de serviço associada, em especial voz e vídeo, sejam admitidos na rede apenas se esta possuir recursos tais que os requisitos destes sejam atendidos sem, no entanto, prejudicar a qualidade de serviço de fluxos anteriormente aceitos. A proposta foi avaliada através de simulações no NS-2.

Palavras-chaves: Qualidade de Serviço (QoS), redes em malha sem fio, IEEE 802.11e, controle de admissão, OLSR, CAC-OLSR.

ABSTRACT

This work presents an admission control mechanism for multi-hop wireless mesh networks based on the contention-based access mode of IEEE 802.11 and the OLSR routing protocol. This mechanism, called CAC-OLSR, aims at ensuring that traffic flows with quality of service requirements, especially voice and video, are only admitted in the network if it has available resources in order to provide flow requirements. Besides, QoS requirements of previously admitted traffic flows cannot be violated. The proposal was evaluated with NS-2 simulations.

Keywords: Quality of Service (QoS), wireless mesh networks, IEEE 802.11e, admission control, OLSR, CAC-OLSR.

1. INTRODUÇÃO

No decorrer da última década, duas grandes tendências têm sido observadas no setor de telecomunicações. A primeira delas diz respeito à provisão de diferentes serviços através de redes de pacotes baseadas no protocolo IP. De fato, redes IP são hoje empregadas, por exemplo, para acesso a páginas web, correio eletrônico, *Internet banking*, comércio eletrônico, bem como inúmeros serviços multimídia que demandam transmissão de voz e/ou vídeo. Neste último caso, enquadram-se os serviços de voz sobre IP — VoIP, além de aplicações de mensagem instantânea que permitem ao usuário transmitir e receber fluxos de voz e vídeo.

Alguns destes serviços, no entanto, demandam requisitos das redes IP que antes não eram por elas proporcionados, principalmente aqueles relacionados com segurança e qualidade de serviço. Este fato despertou e continua motivando diversas iniciativas de pesquisa na comunidade científica.

Outra tendência a ser destacada refere-se à mobilidade. Muitas tecnologias de redes sem fio têm sido desenvolvidas e implementadas no sentido de proporcionar aos usuários mobilidade nos serviços ofertados. Dentre elas, é possível mencionar as redes sem fio baseadas no padrão IEEE 802.11 [IEEE P802.11, 2007], além das tecnologias WiMAX, HSPA (High Speed Packet Access) e, mais recentemente, o LTE (Long Term Evolution), sendo estas duas últimas utilizadas em redes celulares de terceira e quarta gerações respectivamente.

Em particular, cumpre destacar o vertiginoso crescimento das redes sem fio calcadas no padrão IEEE 802.11. O avanço tecnológico aliado à redução de custos de implementação e o uso de espectro de frequência isento de licenciamento pelos órgãos reguladores têm contribuído sobremaneira para sua popularização.

Ainda com relação às redes sem fio IEEE 802.11, é importante ressaltar o crescimento das chamadas redes mesh [Passos et al., 2006]. Neste cenário um nó sem fio atende não somente a um cliente final, mas pode também encaminhar pacotes oriundos de outros nós sem fio, caracterizando um cenário de múltiplos saltos.

Apesar do benefício da mobilidade, a adoção de redes sem fio traz a reboque maiores dificuldades na provisão dos serviços. O meio físico controlado das redes cabeadas é agora substituído por um canal sujeito a oscilações (desvanecimentos) e interferências. Este fato torna-se ainda mais crítico no caso de redes com possibilidade de múltiplos saltos, como as redes mesh.

Diante deste cenário, um grande desafio se constitui quando as duas tendências ora mencionadas se aglutinam, isto é, quando a provisão de diferentes serviços sobre redes IP, inclusive aqueles que englobam aplicações de voz e vídeo, passam a ser ofertados através de redes sem fio. Faz-se necessário desenvolver e implementar mecanismos apropriados que garantam a qualidade de serviço requerida.

No que diz respeito às redes sem fio baseadas no padrão IEEE 802.11, o tema qualidade de serviço tem sido objeto de muitas pesquisas. Em 2005, o próprio IEEE publicou a emenda 802.11e [IEEE P802.11e, 2005], a qual foi incorporada ao padrão original em 2007. Esta emenda estabelece, em nível de camada de enlace, quatro categorias de tráfego com prioridades distintas de acesso ao canal sem fio mediante a configuração de parâmetros como espaçamento entre quadros, janela de contenção e oportunidade de transmissão.

Entretanto, a mera diferenciação de tráfego com distintas prioridades de acesso ao meio não é suficiente para garantir a qualidade de serviço requerida por determinadas aplicações multimídia. Por exemplo, ela não exime a rede de congestionamentos constituídos por fluxos pertencentes a mesma categoria de acesso, fato que aumentará o atraso, bem como poderá provocar queda na vazão percebida por cada fluxo.

É neste contexto que surge a importância de um mecanismo de controle de admissão de tráfego na rede [Hanzo e Tafazolli, 2009]. Mecanismos de controle de admissão são aqueles que regulam a entrada de novos fluxos na rede, objetivando sua admissão apenas se seus requisitos de qualidade de serviço puderem ser atendidos, sem violar, por outro lado, os requisitos demandados pelos fluxos anteriormente admitidos.

Em geral, tais mecanismos monitoram algum parâmetro da rede, por exemplo atraso fim a fim, vazão ou tempo livre de canal, confrontando-o com a

demanda requerida pelo novo fluxo, seja em termos de atraso, vazão ou estimativa do tempo de canal a ser ocupado. Caso o algoritmo de admissão constate a viabilidade do atendimento ao novo fluxo sem degradar os vigentes, o novo fluxo é aceito. Caso contrário, deve ser rejeitado.

1.1. MOTIVAÇÃO

O sucesso no provimento de serviços em redes sem fio, principalmente os que envolvem voz e vídeo, demanda a implementação de mecanismos capazes de diferenciar o tráfego no acesso ao meio, bem como controlar sua admissão na rede.

A diferenciação de tráfego no acesso ao meio, conforme já mencionado, foi implementada pela emenda 802.11e do IEEE. Com relação ao controle de admissão, o padrão IEEE 802.11 sugere um mecanismo para redes infraestruturadas, isto é, aquelas formadas por nós sem fio que atendem somente a seus clientes finais, sem no entanto encaminhar tráfego de outros nós sem fio intermediários. O modo de implementação, contudo, não é definido. Além disso, nada é mencionado sobre controle de admissão para redes em malha sem fio que funcionem no modo ad-hoc (redes mesh), cenário ainda mais crítico em função da possibilidade de múltiplos saltos.

Alguns trabalhos têm sido publicados no meio acadêmico sobre controle de admissão, a maior parte voltada para redes IEEE 802.11 operando no modo infraestruturado. No caso do modo ad-hoc com múltiplos saltos (redes mesh), por outro lado, ainda se observa grande oportunidade de pesquisa e desenvolvimento de mecanismos que controlem a admissão de tráfego [Hanzo e Tafazolli, 2009].

Tais mecanismos para redes operando no modo ad-hoc devem ser capazes de lidar com questões relacionadas ao meio sem fio como interferências intra e interfluxo, as quais serão posteriormente explicadas neste texto, bem como tratar violações de QoS (*Quality of Service*). A violação de QoS ocorre quando um mais fluxos na rede têm seus requisitos de QoS violados, o que pode ocorrer, dentre outros motivos, pela admissão de novos fluxos inapropriadamente ou devido à degradação do canal em virtude da

mobilidade de algum nó, mudança na vizinhança de um ou mais nós ou pela mudança nas condições de propagação.

Outra questão a ser considerada por um mecanismo de controle de admissão refere-se à reserva de recursos para as categorias de acesso propostas pelo IEEE 802.11e que demandam qualidade de serviço, isto é, as categorias de voz e vídeo. Os mecanismos propostos até então consideram a admissão de fluxos segundo critérios definidos, porém não fazem a distinção dos recursos ocupados por cada categoria. Por exemplo, se a rede for ocupada até o seu limite com fluxos de vídeo, o mecanismo de controle de admissão pode não permitir que um novo fluxo de voz seja admitido, mesmo o padrão IEEE 802.11e definindo voz com prioridade superior a vídeo.

Neste caso, uma possível abordagem seria a de o mecanismo de controle de admissão interromper o último fluxo de vídeo admitido de modo que fossem admitidos tantos fluxos de voz até que se atingisse o limite de recursos reservados para esta categoria. Evidentemente, recursos reservados e não utilizados deveriam ser disponibilizados para qualquer outro tráfego pertencente às demais categorias.

Assim, a ausência de trabalhos, que abordem todas essas questões mencionadas no que tange a um mecanismo de controle de admissão para redes IEEE 802.11 operando no modo ad-hoc, motivam a pesquisa realizada por esta dissertação.

1.2. OBJETIVOS

O presente trabalho tem por objetivo principal propor um mecanismo de controle de admissão voltado para redes em malha sem fio de múltiplos saltos e que opere sob o modo de acesso baseado em contenção do padrão IEEE 802.11. O mecanismo proposto é uma extensão ao protocolo de roteamento OLSR (Optimized Link State Routing) [Clausen e Jacquet, 2003], denominado CAC-OLSR (Call Admission Control OLSR). Cabe ressaltar que o OLSR é um protocolo de roteamento pró-ativo comumente utilizado em redes em malha sem fio [Passos et al., 2006; Sing et al., 2008].

Além de considerar as categorias de acesso do padrão IEEE 802.11e, CAC-OLSR lidará com interferências inter e intrafluxo, violações de QoS e possibilitará a configuração de reserva de recursos para as categorias de voz e vídeo.

Uma vez implementado, o mecanismo permitirá que novos fluxos com demanda de QoS tenham seus requisitos atendidos, sem que aqueles previamente admitidos sejam degradados.

Outro objetivo desta dissertação é avaliar o mecanismo proposto CAC-OLSR através de simulações utilizando o software NS-2 [NS2, 2010].

1.3. ESTRUTURA DA DISSERTAÇÃO

Esta dissertação está organizada da seguinte forma. O Capítulo 2 apresenta um breve descritivo do padrão IEEE 802.11e, com foco no modo de acesso baseado em contenção. Serão apresentadas as categorias de tráfego propostas, bem como os parâmetros configuráveis de cada uma, os quais regulam suas prioridades no acesso ao meio.

O Capítulo 3 aborda trabalhos publicados pela comunidade científica sobre controle de admissão para redes IEEE 802.11 operando no modo adhoc. Nesse capítulo, propostas correlatas à apresentada nesta dissertação são analisadas, ressaltando-se pontos fortes e oportunidades de melhoria.

O Capítulo 4 descreve o mecanismo de controle de admissão proposto, denominado CAC-OLSR. Esse capítulo descreve a política de admissão, além do modo com que a proposta lida com diversos desafios existentes em um cenário sem fio de múltiplos saltos, como as interferências inter e intrafluxo e a violação de QoS.

O Capítulo 5 descreve a metodologia empregada na análise de desempenho do mecanismo em questão, a qual foi baseada em simulações em software.

O Capítulo 6 apresenta os resultados da análise de desempenho do mecanismo CAC-OLSR.

Finalmente, o Capítulo 7 conclui a dissertação, destacando as principais contribuições e indicando os trabalhos futuros.

2. PADRÃO IEEE 802.11E

A emenda 802.11e foi publicada em 2005 pelo IEEE e incorporada ao padrão 802.11 em 2007 [IEEE P802.11, 2007]. Seu principal objetivo foi oferecer um tratamento diferenciado no acesso ao meio sem fio para tráfego com demanda por qualidade de serviço.

Este capítulo destaca as principais características da emenda 802.11e, de acordo com a seguinte estrutura: primeiramente são descritas as formas de acesso ao meio previstas, seguidas da apresentação das categorias de acesso e parâmetros utilizados no modo com contenção. Por fim, uma breve descrição sobre o procedimento de *backoff* é apresentada.

2.1. ACESSO AO MEIO

O acesso ao meio no padrão 802.11 pode ocorrer de duas formas: uma com a presença de um elemento central que mantém o controle sobre o acesso ao meio de cada nó da rede, e outra distribuída, sem elemento central, a qual prevê uso de contenção. Inicialmente, definiram-se as funções PCF (*Point Coordination Function*) para o primeiro caso e DCF (*Distributed Coordination Function*) para o segundo, ambas utilizando o mecanismo CSMA/CA (*Carrier Sense Multiple Access with Colision Avoidance*) como base.

A emenda 802.11e propõe uma nova função, denominada HCF (*Hybrid Coordination Function*). Como o nome sugere, trata-se de uma função híbrida com a previsão de acesso ao meio de modo controlado através do HCCA (HCF *Controlled Channel Access*) ou com contenção através do EDCA (*Enhanced Distributed Channel Access*). Os subitens a seguir apresentam algumas peculiaridades de cada um destes novos modos de acesso ao meio.

Cabe ressaltar a possibilidade de alternância de operação entre os dois modos mencionados, conforme ilustrado na Figura 1.

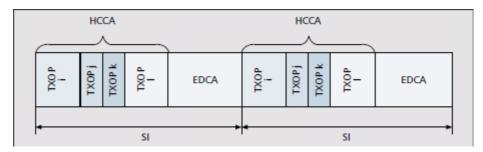


Figura 1: Operação híbrida nos modos HCCA e EDCA.

Fonte [Ramos et al., 2005]

2.1.1. HCCA

O modo HCCA (HCF *Controlled Channel Access*) refere-se ao acesso controlado ao meio e é exclusivo para emprego em redes infraestruturadas, isto é, aquelas com a presença de elementos centrais, ou pontos de acesso, como são denominados no ambiente de redes sem fio 802.11.

Cada ponto de acesso executa as funções previstas no HCCA e, desta forma, controla o acesso ao meio de cada estação sob seu domínio. Dentre estas funções, destacam-se:

- Realização de polling entre as estações, garantindo a todas uma oportunidade de acesso ao meio sem colisão;
- Recebimento de solicitações de reservas de transmissões de estações;
- Alocação de oportunidade de transmissão (TXOP) para estações em resposta a uma solicitação de reserva.

O conceito de oportunidade de transmissão será posteriormente explicado.

2.1.2. EDCA

O EDCA (*Enhanced Distributed Channel Access*) também é conhecido como HCF *Contention Based Channel Access*, isto é, trata-se do modo de acesso baseado em contenção previsto pelo HCF.

Define-se contenção como a ação de conter ou impedir. No caso de redes sem fio, métodos de contenção são aqueles que implementam mecanismos restritivos à forma como cada nó acessará o meio com o objetivo de evitar colisões (*collision avoidance*), conforme explicado em seguida.

Um nó, que possui dados a serem transmitidos, deve aguardar por um determinado período de ociosidade do meio mais um período aleatório antes de iniciar a transmissão. A verificação sobre a utilização do meio se dá via consulta à função CS (*Carrier Sensing*). Se ainda assim houver colisão, física ou mesmo virtual, inicia-se um procedimento de *backoff*, o qual será posteriormente detalhado.

A colisão física ocorre quando dois nós transmitem simultaneamente.

A colisão virtual ocorre quando diferentes categorias de acesso do padrão 802.11e, pertencentes ao mesmo nó, competem pelo acesso ao meio simultaneamente. Neste caso, a função EDCA dá prioridade à categoria preferencial, enquanto a outra entra em procedimento de *backoff*.

O intervalo de tempo em que o nó deve perceber o meio ocioso a fim de iniciar uma transmissão denomina-se IFS (InterFrame Space). Este intervalo não considera o tempo aleatório adicional que o nó deve aguardar para iniciar a transmissão. O padrão IEEE 802.11 define alguns tipos de IFS, porém destacam-se aqui dois de maior relevância para o 802.11e no modo EDCA:

- a) AIFS (*Arbitration InterFrame Space*): trata-se do intervalo em que cada nó deve perceber o meio ocioso para iniciar uma transmissão de dados (excluindo-se o tempo aleatório). Este valor varia para cada uma das categorias de acesso. Categorias de maior prioridade devem possuir valores menores de AIFS, o que lhes garante, estatisticamente, maiores chances de obter acesso ao meio;
- b) SIFS (*Short InterFrame Space*): trata-se do intervalo a ser percebido por cada nó para enviar um reconhecimento ACK, em resposta a um dado transmitido, ou para envio de um quadro CTS (*Clear to Send*) em resposta a um RTS (*Request to Send*). O intervalo SIFS, como o nome já sugere, é o intervalo de menor duração entre os IFSs.

A Figura 2 ilustra a relação entre diferentes IFSs.

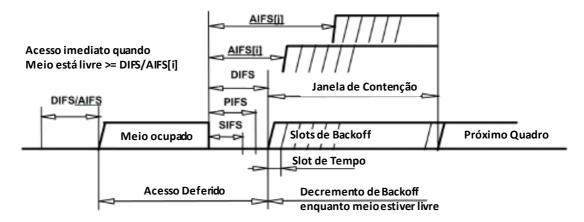


Figura 2: IFSs e suas relações.

Fonte [IEEE P802.11, 2007]

Observa-se na Figura 2 outros valores de IFSs, como o DIFS utilizado pelo DCF e PIFS, pelo PCF [IEEE P802.11, 2007]. Janela de contenção e slots de *backoff* serão alvo de discussão posterior.

Este trabalho tem foco em redes sem fio IEEE 802.11e que funcionam no modo ad-hoc com múltiplos saltos e, por isso, utilizam o modo de acesso baseado em contenção EDCA.

2.2. CATEGORIAS DE ACESSO

A função EDCA do padrão 802.11e provê diferenciação de tráfego mediante o suporte a quatro categorias de acesso (AC – *Access Categories*), conforme listado a seguir:

- a) Background (BK): categoria relativa a tráfego de "fundo". Possui a menor prioridade de acesso ao meio;
- b) *Best-effort* (BE): categoria de melhor esforço. Possui maior prioridade que *Background*;
- c) *Video* (VI): categoria definida para tráfego de vídeo. Possui maior prioridade que *Best-effort*;
- d) *Voice* (VO): categoria definida para tráfego de voz. É a categoria de mais alta prioridade.

A Figura 3 ilustra a arquitetura das quatro categorias de acesso propostas pela emenda 802.11e.

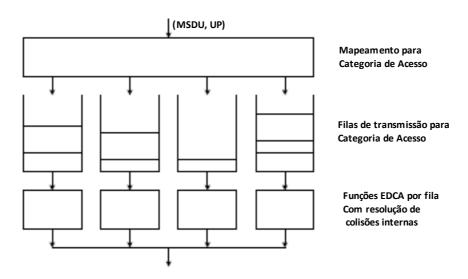


Figura 3: Categorias de Acesso do padrão IEEE 802.11e.

Fonte [IEEE P802.11, 2007]

É importante destacar também que a emenda 802.11e apresenta um mapeamento entre prioridades de usuários definidas no padrão 802.1D [IEEE 802.1D, 2004] com as categorias de acesso definidas, conforme a Tabela 1.

Tabela 1: Mapeamento sugerido entre prioridades e ACs.

Prioridade	UP (conforme 802.1D)	Designação 802.1D	AC	Designação (informativo)
Mais Baixa	1	BK	AC_BK	Background
	2	-	AC_BK	Background
	0	BE	AC_BE	Best Effort
	3	EE	AC_BE	Best Effort
	4	CL	AC_VI	Video
	5	VI	AC_VI	Video
	6	VO	AC_VO	Voice
Mais Alta	7	NC	AC_VO	Voice

Fonte [IEEE P802.11, 2007]

As categorias de acesso diferem entre si em virtude da prioridade que possuem para acessar o meio. Esta prioridade é definida de acordo com o ajuste de alguns parâmetros, conforme será apresentado a seguir.

2.3. PARÂMETROS DE ACESSO AO MEIO DA FUNÇÃO EDCA

A função EDCA prioriza o acesso ao meio das diferentes categorias através da configuração dos seguintes parâmetros:

- a) AIFS: o intervalo AIFS, conforme já explicado, regula o tempo que uma categoria necessitará perceber o meio ocioso antes de iniciar uma transmissão de dados. Quanto maior o valor do intervalo AIFS, menor a prioridade da categoria de acesso;
- b) Janelas de contenção mínima e máxima (CWmin e CWmax): especificam os valores mínimo e máximo para a janela relativa ao procedimento de contenção (*backoff*). Quanto maiores os valores de janela de contenção, menor prioridade possui uma determinada categoria;
- c) Limite de oportunidade de transmissão (TXOPLimit): intervalo em que uma categoria pode enviar quadros em sequência, separados apenas por SIFS para recebimento de reconhecimentos (ACKs).

A Tabela 2 apresenta os valores sugeridos para esses parâmetros pela emenda 802.11e.

	CWmin	CWmax	AIFSN	TXOP Limit		
AC				Para PHYs definido nos capítulos 15 e 18	Para PHYs definido nos capítulos 17 e 19	Outras PHYs
AC_BK	aCWmin	aCWmax	7	0	0	0
AC_BE	aCWmin	aCWmax	3	0	0	0
AC_VI	(aCWmin+1)/2-1	aCWmin	2	6,016 ms	3,008 ms	0
AC_VO	(aCWmin+1)/4-1	(aCWmin+1)/2-1	2	3,264 ms	1,504 ms	0

Tabela 2: Valores sugeridos para os parâmetros das categorias de acesso.

Fonte [IEEE P802.11, 2007]

Cabe ressaltar que o parâmetro AIFS é calculado através da soma de um intervalo de tempo fixo pré-definido mais uma determinada quantidade de slots de tempo determinada pela variável AIFSN. Deste modo, a variável a ser parametrizada é um inteiro denominado AIFSN.

2.4. PROCEDIMENTO DE BACKOFF

O procedimento de *backoff* é implementado em adição aos intervalos entre quadros – IFSs – como mecanismo de contenção da função EDCA. O procedimento prevê a inicialização de um contador a ser decrementado quando da ocorrência de determinados eventos. Uma vez inicializado o referido contador, o nó só adquire direito de acesso ao meio quando a contagem atinge

o valor nulo. Ainda, em caso de reincidência dos eventos mencionados, valores mais altos podem ser configurados para decremento.

Conforme definido na emenda 802.11e, alguns dos eventos que disparam o procedimento de *backoff* são:

- a) transmissão sem sucesso de um determinado quadro;
- b) colisão interna ao próprio nó entre duas categorias de acesso (colisão virtual). Neste caso, a de maior prioridade acessa o meio e outra inicia o procedimento de *backoff*.

Uma vez ocorridos tais eventos, inicializa-se o contador com um valor aleatório tomado entre 0 e o valor definido para janela de contenção da categoria em questão, isto é, [0, CW[AC]]. Na ocorrência do primeiro evento de *backoff*, a janela de contenção possui valor equivalente à janela de contenção mínima da categoria (CWmin). A cada reincidência, o valor máximo da janela é atualizado de acordo com a Equação 1 (extraída da emenda 802.11e) até o valor limite correspondente a CWmax.

$$CW_{i+1}[AC] = (CW_i[AC] + 1)^2 - 1$$
, para $CW_i[AC] \le (CW_{i+1}[AC] = CW_{i+1}[AC]$, para $CW_i[AC] > (CW_{i+1}[AC] = CW_{i+1}[AC]$ (1)

2.5. RESUMO DO CAPÍTULO

O presente capítulo descreveu os principais aspectos da emenda IEEE 802.11e, em especial seu funcionamento sob o modo de contenção. Os parâmetros configuráveis de cada categoria de acesso, os quais regulam a prioridade de cada uma no acesso ao meio, foram destacados.

O próximo capítulo versará sobre os trabalhos publicados na literatura relacionados ao tema controle de admissão.

3. TRABALHOS RELACIONADOS

Os mecanismos de controle de admissão são responsáveis por regular a entrada de tráfego na rede e o fazem com dois principais objetivos:

- Garantir que os requisitos de QoS de um novo fluxo sejam atendidos;
- Zelar para que requisitos de QoS de fluxos previamente admitidos não sejam degradados.

Cumpre destacar que a presença de controle de admissão em uma rede sem fio é de suma importância, visto que apenas a diferenciação de tráfego, conforme previsto na emenda IEEE 802.11e, não é capaz de prevenir congestionamentos causados por excesso de fluxos pertencentes a uma mesma categoria de acesso.

O tema controle de admissão em redes sem fio baseadas no padrão IEEE 802.11 tem sido alvo de algumas iniciativas no meio acadêmico, dada sua relevância. A maior parte delas, no entanto, propõe mecanismos voltados a redes sem fio infraestruturadas, sendo uma menor parcela aquelas que se dedicam a redes em malha com possibilidade de múltiplos saltos, funcionando no modo ad-hoc.

Este capítulo apresenta um resumo dos principais trabalhos relacionados ao tema controle de admissão para redes em malha sem fio. Estes trabalhos foram classificados de acordo com o tipo de protocolo de roteamento associado ao mecanismo de controle de admissão, a saber:

- Mecanismos independentes de protocolo de roteamento;
- Mecanismos associados a protocolos de roteamento reativos;
- Mecanismos associados a protocolos de roteamento pró-ativos.

3.1. MECANISMOS INDEPENDENTES DE PROTOCOLO DE ROTEAMENTO

Ahn et al. propuseram um mecanismo denominado SWAN – *Service Differentiation in Stateless Wireless Ad Hoc Networks* [Ahn et al., 2002a], o qual não depende de um protocolo de roteamento específico e é executado no nó de origem.

O controle de admissão do mecanismo SWAN opera, em resumo, da seguinte maneira: para fluxos sem QoS, o mecanismo apenas regula a taxa de transmissão baseado na aplicação do princípio AIMD (*Additive Increase, Multiplicative Decrease*) sobre a diferença entre o tempo de envio de um pacote e o recebimento de seu reconhecimento (ack). A taxa de um determinado fluxo sem QoS vai aumentando até que a diferença de tempo entre envio de pacote e seu respectivo reconhecimento atinja determinado valor, o que provoca a redução na taxa de transmissão.

Para fluxos com QoS há um esquema de controle de admissão distribuído. Neste caso, o nó fonte de um fluxo novo envia um *probe* em direção ao destino, sendo que esta mensagem armazena a menor disponibilidade de banda dentre todos os nós do caminho. A disponibilidade de banda em cada nó é calculada mediante a diferença entre um limiar de banda predefinido e a banda requerida para transmissão dos pacotes enviados dentro da região de interferência do nó. Ao receber a resposta desse *probe*, o nó fonte decide pela aceitação do tráfego entrante, caso a disponibilidade de banda informada seja maior que a largura de banda solicitada pelo novo fluxo.

Cabe ressaltar que neste cálculo de disponibilidade de banda, SWAN não leva em consideração a banda ocupada entre a região de transmissão e a região de detecção de portadora do nó, fato que pode provocar um estimativa equivocada com relação à banda disponível. Além disso, a definição do limiar de banda de cada nó é outro ponto crítico do mecanismo, dado que na prática este pode variar. O mecanismo também não considera as categorias de acesso da emenda IEEE 802.11e, tampouco o fenômeno de interferência intrafluxo, que será posteriormente explicada.

O trabalho de [Yang e Kravets, 2005], citado em muitos artigos sobre o assunto, propôs um controle de admissão para redes ad-hoc valendo-se do critério de ocupação do canal. Em resumo, cada nó decide sobre a admissão de um novo fluxo mediante a comparação entre a estimativa de ocupação do meio provocada pelo tráfego entrante com a estimativa de recursos disponíveis no canal. Caso a primeira seja menor que a segunda, o fluxo pode ser admitido. O trabalho não levou em consideração as categorias de acesso propostas pela emenda IEEE 802.11e.

3.2. MECANISMOS ASSOCIADOS A PROTOCOLOS DE ROTEAMENTO REATIVOS

No artigo [Chakeres e Belding-Royer, 2007], os autores utilizaram o mesmo princípio de [Yang e Kravets, 2005], porém propuseram a ampliação da região a ser consultada por cada nó na medição de ocupação do meio, visto a possibilidade de haver, por exemplo, "terminais escondidos", fato que pode levar a falsas admissões.

Para cenários com múltiplos saltos, os autores sugeriram associação com protocolos de roteamento reativos, onde a admissão dos novos fluxos se daria por ocasião do processo de descoberta de rotas, porém não mencionaram como o mecanismo poderia ser aplicado em protocolos próativos como o OLSR. Tampouco foram considerados as categorias de acesso da emenda IEEE 802.11e e os fenômenos como a interferência intrafluxo, a qual será posteriormente explicada.

O trabalho de [Luo et al., 2006] também se baseou em [Yang e Kravets, 2005], porém acrescentou suporte a canais com múltiplas taxas, bem como suporte a múltiplos saltos através de protocolos reativos. Além disso, apresentou abordagem para violação de QoS e modelo mais realista de medição da ocupação do canal, considerando-se possibilidade de transmissões paralelas. As categorias de acesso da emenda IEEE 802.11e não foram consideradas em [Luo et al., 2006].

Lindgren e Belding-Royer apresentaram uma abordagem para tratamento da interferência intrafluxo [Lindgren e Belding-Royer, 2004]. Tratase de um fator denominado contador de contenção, o qual deve ser aplicado na estimativa de consumo de recursos de um novo fluxo durante o processo de admissão. Esse trabalho também associou a admissão de fluxos com a descoberta de rotas, fato que exclui a aplicação do mecanismo em redes cujo protocolo de roteamento seja pró-ativo, como o OLSR. As categorias de acesso da emenda IEEE 802.11e também não foram consideradas.

Kettaf et al. apresentaram um método de controle de admissão [Kettaf et al., 2006a] associado com o protocolo de roteamento reativo ACOR (*Admission Control enabled On-demand Routing*) [Kettaf et al., 2006b]. Novamente o controle de admissão é feito em cada nó por ocasião do processo relativo à

descoberta de rotas. Nesta proposta, além da banda disponível, a métrica atraso também é considerada como critério de decisão para admissão de novo fluxo. O artigo também considerou tratamento para casos de violação de QoS, entretanto, a interferência intrafluxo não foi abordada.

Cerveira e Costa em [Cerveira e Costa, 2006] propuseram uma extensão ao protocolo AODV (*Ad hoc On-Demand Distance Vector*) [Perkins e Belding-Royer, 2003], acrescentando um mecanismo de controle de admissão baseado em tempo de ocupação de canal. Porém, a proposta diferencia o tempo de ocupação entre tráfego com QoS e tráfego de melhor esforço, desconsiderando este último no processo de admissão de novos fluxos. Esta estratégia é utilizada para que o tráfego de melhor esforço ocupe somente os recursos de rede não utilizados por fluxos demandantes de QoS. Além da diferenciação do tempo de ocupação, os autores assumiram como premissa a utilização das categorias de acesso previstas na emenda IEEE 802.11e, bem como que o tráfego de melhor esforço utiliza o TCP como protocolo de transporte. Cerveira e Costa apresentaram ainda um tratamento para a questão da interferência intrafluxo, além de tratar a violação de QoS.

O trabalho de [Abdrabou e Zhuang, 2008] também se baseou no protocolo AODV para realizar controle de admissão durante o processo de descoberta de rotas. A diferença deste método reside no critério de decisão, o qual se fundamenta na métrica atraso. Em cada nó estima-se, a partir de modelagem baseada em processos estocásticos, a probabilidade do atraso fima-fim do novo fluxo exceder ao requisitado. Se esta for menor que um limiar, o fluxo pode ser aceito. Caso contrário, deve-se rejeitá-lo.

Prabhakaran em [Prabhakaran, 2010] associou o controle de admissão com o protocolo reativo DSR (Dynamic Source Routing) [Johnson et al., 2007]. Abordou a interferência intrafluxo, violação de QoS e considerou, além de disponibilidade de banda, a confiabilidade do enlace como critério de decisão para controle de admissão. A confiabilidade do enlace é estimada em função da distância do mesmo e velocidade dos nós em questão.

Cheng et al. em [Cheng et al., 2008] também associaram controle de admissão com protocolo de roteamento reativo, porém o critério de decisão adotado foi a banda residual calculada em cada grafo de contenção. Um grafo

de contenção é aquele em que os vértices são os enlaces existentes na rede e os arcos são constituídos entre os enlaces da rede interferentes entre si.

3.3. MECANISMOS ASSOCIADOS A PROTOCOLOS DE ROTEAMENTO PRÓ-ATIVOS

O trabalho de [Nguyen e Minet, 2005] foi o primeiro artigo encontrado a apresentar um mecanismo de controle de admissão específico para protocolo de roteamento pró-ativo. Na verdade, o artigo propõe uma versão modificada do OLSR, a qual considera reserva de banda e interferência do canal para fins de seleção de rotas, além do próprio controle de admissão. Este último é baseado em dois passos: aceitabilidade e viabilidade. No primeiro passo, o nó em questão compara a banda requerida pelo fluxo entrante com a banda disponível em cada nó do caminho (a banda de cada nó é divulgada via mensagens *Hello*). Se aquela for menor do que a menor banda disponível dentre todos os nós em questão, o fluxo é aceito. Em havendo a aceitação, procede-se à etapa de viabilidade. Nesta etapa, ocorre a verificação quanto a interferências do novo fluxo na vizinhança dos nós ao longo do caminho. A proposta não considera a emenda IEEE 802.11e, tampouco aborda a questão da interferência intrafluxo.

Badis e Agha em [Badis e Agha, 2006] propuseram um mecanismo de controle de admissão para o protocolo QOLSR [Badis e Munaretto, 2004]. O esquema baseia-se no envio preliminar de mensagens do tipo Check Request (CREQ) e Check Reply (CREP), as quais carregam os requisitos do fluxo entrante em termos de banda, atraso e jitter. A cada nó, verifica-se se tais requisitos podem ou não ser atendidos, passando-se a mensagem adiante caso positivo. Se o nó originador do CREQ receber o CREP correspondente, o fluxo é aceito. Esta proposta tratou a interferência intrafluxo, porém não considerou o uso das categorias de acesso do IEEE 802.11e.

3.4. COMPARAÇÃO DOS TRABALHOS RELACIONADOS COM A PROPOSTA DESTA DISSERTAÇÃO

Até onde foi possível ao corrente trabalho investigar, o mecanismo de controle de admissão apresentado nesta dissertação é o primeiro a ser

proposto para um protocolo de roteamento pró-ativo que considera as seguintes características: rede em malha sem fio com múltiplos saltos; utilização das categorias de acesso da emenda IEEE 802.11e; tratamento para as interferências inter e intrafluxo e tratamento para eventuais violações de QoS.

Além das características acima mencionadas, o presente trabalho considera a reserva de recurso de tempo de uso do canal para as categorias de acesso de voz e vídeo do padrão IEEE 802.11e. Se não utilizados, estes recursos podem ser empregados por outras categorias. Ressalta-se a importância desta funcionalidade, pois, se não considerada e, por exemplo, uma determinada rede está completamente utilizada com fluxos de vídeo previamente admitidos, o mecanismo de controle de admissão não permitirá a admissão de novos fluxos de voz, categoria de prioridade superior. CAC-OLSR, neste caso, interromperá os últimos fluxos de vídeo admitidos para permitir a admissão de fluxos de voz, até o limite dos recursos reservado para esta categoria. Esta abordagem não é considerada em nenhum dos trabalhos relacionados ao tema encontrados na literatura.

O próximo capítulo descreve o mecanismo proposto, CAC-OLSR.

4. CAC-OLSR

A proposta CAC-OLSR [Mineiro e Saade, 2011], do inglês *Call Admission Control* OLSR, é uma extensão ao protocolo de roteamento OLSR [Clausen e Jacquet, 2003] e tem por objetivo dotá-lo de um mecanismo de controle de admissão. Conforme já anteriormente mencionado, este mecanismo regulará a entrada de novos fluxos na rede de modo a que sejam admitidos apenas se seus requisitos de qualidade de serviço puderem ser atendidos, sem, no entanto, causar degradação naqueles previamente admitidos.

CAC-OLSR baseia-se na medição periódica da ocupação de tempo de uso do canal sem fio e na estimativa de ocupação demandada por um novo fluxo no período considerado. Caso esta seja menor que a disponibilidade temporal de uso do canal, o novo fluxo pode ser aceito. Caso contrário, este deve ser rejeitado.

CAC-OSLR utiliza as categorias de acesso definidas pela emenda IEEE 802.11e, conforme apresentado no Capítulo 2, e trata os problemas das interferências inter e intrafluxo, bem como da violação de QoS, conceitos que serão abordados neste capítulo.

Para melhor entendimento do mecanismo, este capítulo foi estruturado da seguinte forma: a Seção 4.1 apresenta o modo de medição da ocupação do canal no tempo, enquanto a Seção 4.2 mostra como é feita a estimativa de ocupação do meio para um novo fluxo. As Seções 4.3 e 4.4 descrevem, respectivamente, como as interferências inter e intrafluxo e eventuais violações de QoS são tratadas. A Seção 4.5 apresenta a política de admissão do mecanismo e, finalmente, a Seção 4.6 apresenta um resumo do presente Capítulo.

4.1. MEDIÇÃO DA OCUPAÇÃO DO CANAL

A medição da ocupação do canal no tempo se dá mediante a observação, por um determinado período, da indicação de detecção de portadora, resultando em um percentual de ocupação. No padrão IEEE 802.11, esta detecção é implementada através do PHYCS – PHY *Carrier Sense* [IEEE P802.11, 2007].

Cabe ressaltar, no entanto, que CAC-OLSR mede, para fins de controle de admissão, a ocupação provocada por quadros pertencentes às categorias de acesso 0 e 1 da emenda 802.11e, bem como a ocupação percebida por quadros cujo recebimento não pôde ser realizado corretamente. Esta estratégia foi adotada para evitar que fluxos que exigissem QoS deixassem de ser admitidos em virtude do meio estar ocupado por tráfego que não possui qualidade de serviço associada. Considera-se aqui, conforme adotado por [Cerveira e Costa, 2006], que a menor prioridade de acesso ao meio das categorias sem QoS, aliada ao fato de que o tráfego nelas fluente em geral utiliza o TCP como protocolo de transporte, automaticamente reduzirá este tipo de tráfego na presença de fluxos com QoS.

Evidentemente a consideração do tempo de ocupação por quadros que não puderam ser recebidos corretamente, provavelmente transmitidos por nós posicionados entre a região de recepção e a região de interferência de cada nó, teve por objetivo considerar o pior caso, visto que podem pertencer às categorias de acesso sem qualidade de serviço associada.

Outra consideração a ser feita refere-se ao problema conhecido como "terminal exposto". É possível que um dado nó considere o meio ocupado, por conta da transmissão de um quadro de um nó vizinho que não causará interferência em um novo fluxo em admissão. CAC-OLSR assume uma postura conservadora, considerando também esta ocupação na medição do canal.

Em resumo, a medição de ocupação do canal observa por um determinado período (100 ms foi adotado na implementação deste trabalho) a detecção de portadora e retorna a ocupação percentual no tempo provocada por quadros pertencentes às categorias 0 e 1, bem como pelos quadros cujo recebimento não pôde ser efetuado corretamente.

4.2. ESTIMATIVA DE OCUPAÇÃO DO CANAL POR UM NOVO FLUXO

A estimativa de ocupação do canal por um novo fluxo é um aspecto relevante para o controle de admissão, visto que será comparada com a ocupação corrente do meio para a tomada de decisão quanto à aceitação ou rejeição do tráfego entrante. Em caso de superestimativa, poderá haver ociosidade do canal. Por outro lado, se houver subestimativa, a rede estará sujeita a falsas admissões, fato que poderá degradar a qualidade de serviço

dos fluxos. Entende-se por falsa admissão a aceitação de um fluxo que deveria ser rejeitado em virtude das condições de tráfego na rede.

Objetivando facilitar o entendimento da estimativa proposta, faz-se necessário averiguar a forma de ocupação temporal do meio durante a transmissão de um quadro de dados, conforme previsto no padrão IEEE802.11. A Figura 4 ilustra tal ocupação.

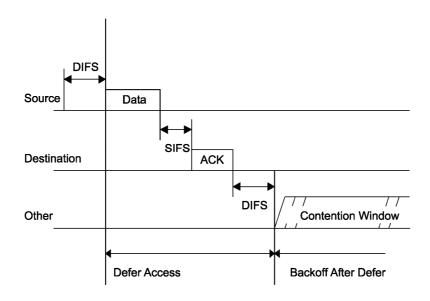


Figura 4: Ocupação do meio na transmissão de um quadro IEEE 802.11.

Fonte [IEEE P802.11, 2007]

Observando-se a Figura 4, chega-se à seguinte conclusão sobre o tempo necessário à transmissão de um quadro de dados, denominado T_{quadro}:

$$T_{quadro} = Data + SIFS + ACK + AIFS + Backoff$$
 (2)

Na Equação (2), "Data" é o tempo necessário à transmissão do quadro de dados; "SIFS" é o espaçamento entre quadros antes da transmissão do reconhecimento do quadro antes enviado; "ACK" é o tempo para transmissão do reconhecimento; "AIFS" é o espaçamento entre quadros antes da transmissão de outro quadro de dados. Na Figura 4, "AIFS" está retratado como "DIFS". Em caso de utilização das categorias de acesso da emenda IEEE 802.11e, o "DIFS" deve ser substituído pelo "AIFS" da categoria correspondente. Por fim, "Backoff" é o tempo necessário à contenção do meio.

As seções que se seguem apresentam a forma de cálculo ou estimativa de cada uma destas parcelas.

4.2.1. CÁLCULO DE "SIFS" E "AIFS"

Os valores de "SIFS" e "AIFS" podem ser facilmente obtidos mediante consulta ao padrão IEEE 802.11. Por exemplo, o valor de SIFS é de 10 µs para estações cuja modulação seja OFDM e que operem com taxas de transmissão de até 54 Mbps, conforme a alínea b do item 19.1.2 do padrão mencionado [IEEE P802.11, 2007].

Já o valor de AIFS é calculado de acordo com a Equação (3), a qual foi extraída do padrão IEEE 802.11.

$$AIFS[AC] = AIFSN[AC] \times aSlotTime + ASIFSTime$$
 (3)

Cada categoria de acesso (AC) possui seu próprio parâmetro de espaçamento entre quadros do tipo AIFS, daí o termo "AC" entre colchetes na Equação (3). A parcela AIFSN é um inteiro configurável para cada categoria. O padrão IEEE 802.11 sugere um AIFSN igual a 2 para as categorias de voz e vídeo, 3 para *Best Effort* e 7 para *Background*, conforme mostrado na Tabela 2.

O valor de aSlotTime equivale a 20 µs para estações com modulação OFDM e taxas de transmissão de até 54 Mbps, de acordo com a alínea b do item 19.1.2 do padrão IEEE 802.11 [IEEE P802.11, 2007].

Por último, ASIFSTime corresponde à parcela "SIFS" cujo valor sugerido de 10 µs já foi mencionado.

Deste modo, chega-se à conclusão que o cálculo para o valor de AIFS para as categorias de voz e vídeo equivale a 50 µs, utilizando-se os valores sugeridos pelo padrão IEEE 802.11 para estações com modulação OFDM.

4.2.2. CÁLCULO DE "ACK"

O cálculo da parcela "ACK" da Equação (2) é realizado simplesmente com base na divisão entre o tamanho do quadro de reconhecimento e a taxa de transmissão do mesmo.

Ressalta-se que o tamanho de "ACK" pode ser obtido mediante consulta ao padrão IEEE 802.11 [IEEE P802.11, 2007]. O item 7.2.1.3 deste documento

estabelece que um quadro "ACK" possui 14 octetos ou 112 bits. A estes devem ser somados os bits de preâmbulo e cabeçalho PLCP (*Physical Layer Convergence Procedure*), que correspondem a 192 bits considerando-se a PLCP longa descrita no item 18.2.2.1 do mesmo padrão.

4.2.3. CÁLCULO DE "Data"

O tempo de transmissão de um quadro de dados deve ser calculado de acordo com a Equação (4).

$$Data = 8 * (tamanho + 48) / taxa$$
 (4)

Na Equação (4), "tamanho" é o tamanho do pacote em bytes, desconsiderando-se os cabeçalhos IP e MAC que já foram estimados em 48 bytes, e "taxa" é a taxa de transmissão de dados em bits/s.

Vale ressaltar que o valor da taxa considerado é fixo. Em caso de mudança automática de taxa para valores menores do que o utilizado, ocorrerá subestimativa dos recursos demandados pelo novo fluxo, problema este a ser corrigido, caso necessário, pelo mecanismo de reação a violações de QoS.

4.2.4. ESTIMATIVA DE "Backoff"

Sabe-se que o valor de "Backoff" corresponde a um valor aleatório escolhido entre 0 e CW multiplicado pela duração de um slot de tempo, sendo CW o parâmetro denominado janela de contenção.

O slot de tempo é conhecido e o valor aleatório mencionado é estimado em CW/2, haja vista a distribuição uniforme preconizada pelo padrão IEEE 802.11. Contudo, é necessário definir qual valor de CW deve ser utilizado, uma vez que segundo a emenda 802.11e, cada categoria de acesso pode variar o valor da janela de contenção entre CWmin e CWmax. O valor de CW inicia-se em CWmin e é incrementado a cada evento de falha da transmissão (colisões ou perda de pacotes) de acordo com a Equação (1).

Dado que o mecanismo de controle de admissão evitará que a rede experimente um estado de congestionamento, os eventos de falha, especialmente os de colisão, tendem a serem reduzidos e, deste modo, CW

aproxima-se de CWmin. Assim, considera-se o tempo gasto com procedimentos de "backoff" como:

Cumpre destacar que, em caso de aumento dos eventos de falha, por exemplo aumento na perda de pacotes devido a variação nas condições de propagação, a estimativa proposta na Equação (5) poderá estar subestimada. Neste caso, o mecanismo de violação de QoS de CAC-OLSR, a ser detalhado posteriormente, entrará em ação de modo a evitar o congestionamento da rede.

Outra observação a ser considerada é que o trabalho de [Cerveira e Costa, 2006] efetuou simulações com diferentes valores de janela de contenção mínima (CWmin), variando-se o número de nós disputando o acesso ao meio, e chegou à conclusão de que o número médio de slots de tempo gasto com o procedimento de "Backoff" tende a se tornar constante, conforme ilustra a Figura 5. Estes valores aproximadamente constantes foram utilizados pelos autores para a estimativa do tempo gasto com "Backoff".

Ressalta-se que, especialmente para as simulações com CWmin iguais a 7 e 15 (sugeridas pelo IEEE 802.11 para as categorias de voz e vídeo respectivamente), a quantidade média de slots de tempo empregadas com o procedimento de "Backoff" foi próxima ao valor de CWmin/2, estimativa proposta por este trabalho.

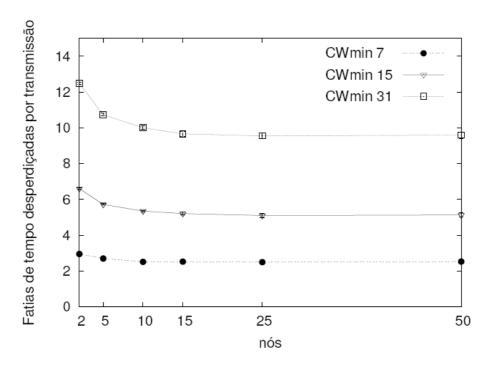


Figura 5: Valor Aleatório médio para *Backoff.*Fonte [Cerveira e Costa, 2006]

De posse do tempo ocupado por um quadro de dados T_{quadro} (Equação 2) e conhecida a periodicidade de envio de pacotes do novo fluxo, T_{pacote} , a estimativa percentual de ocupação temporal do meio provocada por um novo fluxo é dada pela Equação (6):

Ocupação =
$$T_{quadro}/T_{pacote} * 100$$
 (6)

Caso sejam conhecidos valores reais de ocupação, estes podem ser utilizados em substituição ao cálculo aqui sugerido.

4.3. TRATAMENTO DE INTERFERÊNCIAS

No contexto de redes em malha sem fio, dois tipos de interferência precisam ser tratados por um mecanismo de controle de admissão, a saber: interfluxo e intrafluxo. Ambos são tratados nas subseções que se seguem.

4.3.1. INTERFERÊNCIA INTERFLUXO

Considera-se como interferência interfluxo aquela que se dá entre dois ou mais fluxos distintos e concorrentes quanto ao acesso ao meio. Se não considerada no processo de admissão, um novo fluxo pode degradar os requisitos de QoS de outros fluxos vigentes na sua região de interferência. A Figura 6 ilustra um possível caso de interferência interfluxo em redes em malha sem fio.

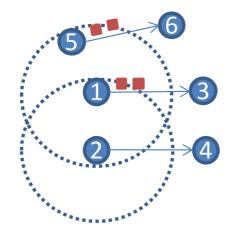


Figura 6: Cenário de interferência interfluxo.

Na Figura 6, há dois fluxos vigentes na rede, um deles do nó 1 para o nó 3 e outro do nó 5 para o nó 6. Por exemplo, cada um desses fluxos ocupa 40% do canal, o que faz com que a ocupação observada pelos nós 1 e 5 seja de 80%. Os círculos tracejados representam a região de interferência dos nós 1 e 2. Como o fluxo do nó 5 para o nó 6 está fora da região de interferência do nó 2, este percebe apenas 40% de ocupação do meio. Caso a interferência interfluxo não seja considerada, um novo fluxo com demanda de 40% de ocupação do meio entre os nós 2 e 4 poderá ser admitido pelo nó 2, visto que o mesmo considera que possui recursos disponíveis. No entanto, a admissão deste novo fluxo provavelmente causará degradação no fluxo entre os nós 1 e 3, em virtude do nó 1 não possuir a totalidade de recursos para acomodar adequadamente os três fluxos citados dentro de sua região de interferência.

CAC-OLSR propõe o combate à interferência interfluxo mediante a divulgação de ocupação do meio, medida por cada nó e por seus vizinhos de um salto, via mensagens Hello do OLSR [Clausen e Jacquet, 2003] e posterior armazenamento desta informação na tabela de vizinhos de um e dois saltos, as quais já são implementadas pelo protocolo em questão. Esta proposta assume a premissa de que a região de interferência de cada nó é composta por seus

vizinhos de até dois saltos de distância, conforme adotado por [Yang e Kravets, 2005].

Deste modo, cada nó ao longo do caminho poderá avaliar se ele próprio, bem como todos os seus vizinhos de um e dois saltos, possuem recursos para atender à nova demanda durante o processo de controle de admissão, garantindo, desta forma, que o novo fluxo poderá ser acomodado sem degradar aqueles previamente admitidos dentro de sua região de interferência.

4.3.2. INTERFERÊNCIA INTRAFLUXO

A interferência intrafluxo ocorre quando nós que encaminham pacotes de um mesmo fluxo são interferentes entre si. Para ilustrar este fenômeno, seja considerado o cenário de transmissão de um fluxo com origem no nó 1 e destino no nó 6, conforme ilustrado na Figura 7.

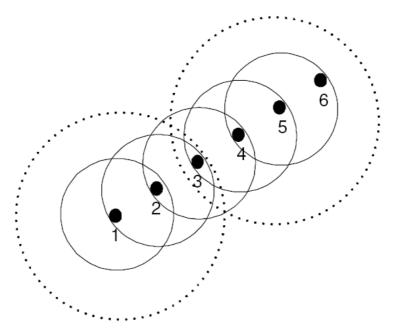


Figura 7: Cenário de interferência intrafluxo.

Fonte [Cerveira e Costa, 2006]

Os círculos cheios representam o alcance de transmissão/recepção de dados para cada nó. Já os círculos tracejados representam o alcance da detecção de portadora dos nós 1 e 5. Observa-se que o nó 3 está na região de detecção dos nós 1, 2, 4 e 5, fato que o impede de transmitir em paralelo com qualquer um deles. Como consequência, seus recursos disponíveis em termos de tempo de canal livre devem ser reduzidos em 5 vezes.

O protocolo CAC-OLSR considera, assim como a proposta de [Lindgren e Belding-Royer, 2004], que a interferência intrafluxo deve ser estimada mediante o cálculo de um fator chamado Contador de Contenção (CC). Este fator soma, em cada nó, a quantidade de nós pertencentes à região de interferência intrafluxo, limitado ao máximo de 5 nós, isto é, os dois anteriores, os dois posteriores, além do próprio nó. Por exemplo, o CC do nó 3 da Figura 7 é igual a 5. Cabe ressaltar que em cenários reais, a quantidade de nós na região de interferência pode ser maior ou menor que dois saltos. Para o primeiro caso, a ocupação do meio poderá ser subestimada, fato que será corrigido pelo mecanismo de reação a violação de QoS. Para o segundo caso, poderá haver superestimativa de ocupação do meio, provocando assim uma postura conservadora do mecanimo CAC-OLSR.

Este fator deve ser aplicado à estimativa de ocupação do meio por um novo fluxo. Deste modo, a Equação (6) deve ser complementada, conforme propõe a Equação (7).

Ocupação
$$_{Total}$$
 = Ocupação * CC (7)

Na Equação (7), "Ocupação_{Total}" corresponde à ocupação provocada por um novo fluxo considerando-se a interferência intrafluxo, "Ocupação" é a ocupação calculada conforme a Equação (6) e CC é o fator denominado Contador de Contenção, conforme já explicado.

4.4. VIOLAÇÃO DE QOS

Considera-se que a qualidade de serviço de um fluxo foi violada quando um ou mais de seus requisitos, por exemplo atraso ou variação do atraso, deixaram de ser atendidos. Violações de QoS podem ocorrer, por exemplo, devido aos seguintes fatores:

- Falsas admissões: ocorrem quando o mecanismo de controle de admissão admite um fluxo que deveria ter sido rejeitado, de modo que seus requisitos de QoS não podem ser atendidos e/ou os requisitos daqueles anteriormente admitidos deixaram de ser satisfeitos;
- Mudanças na vizinhança dos nós: podem ser provocadas em virtude da mobilidade dos nós. Por exemplo, caso um nó cuja ocupação de

canal esteja próxima a 100% se desloque para dentro da região de interferência de um ou mais nós, também com ocupação de canal próxima a 100%, haverá congestionamento e consequente violação de QoS;

 Variações nas condições de propagação: podem modificar consideravelmente os recursos disponíveis em termos de tempo de canal a serem empregados por cada nó, provocando assim degradação na qualidade de serviço de fluxos vigentes na rede sem fio.

O protocolo CAC-OLSR combate possíveis violações de QoS mediante a monitoração da ocupação do canal com fluxos com QoS. Se algum nó detecta que a citada ocupação ultrapassou um limiar denominado "Limiar de Violação", a rede está próxima do congestionamento e, portanto, um ou mais fluxos necessitam ser interrompidos.

Neste caso, CAC-OLSR primeiramente verifica se os recursos de tempo de canal reservados para a categoria de vídeo foram excedidos (conforme mencionado no Capítulo 3, as categorias de voz e vídeo possuem recursos reservados, os quais podem ser utilizados por outras categorias se não estiverem em uso). Caso positivo, uma mensagem do tipo ICMP é enviada ao nó de origem do último fluxo de vídeo admitido, de modo a interrompê-lo. Caso negativo, significa que os recursos reservados para voz foram excedidos e, deste modo, a mensagem ICMP mencionada é enviada para o nó de origem do último fluxo de voz admitido. A Figura 8 ilustra a proposta para tratamento de violações de QoS.

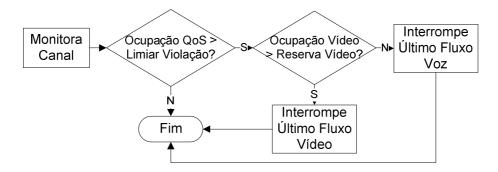


Figura 8: Mecanismo de violação de QoS.

De modo a evitar o efeito de sincronização, cada nó deve iniciar a monitoração do canal em instantes distintos. Além disso, é necessário um intervalo mínimo equivalente a um período de observação (100 ms foi adotado na implementação deste trabalho) entre as medidas para fins de atualização dos recursos da rede.

4.5. POLÍTICA DE ADMISSÃO

A política de admissão engloba os critérios a serem utilizados para tomada de decisão quanto à admissão de um novo fluxo na rede. Esta política considera todos os conceitos explicados neste capítulo, tais como a medição de ocupação de canal, estimativa de ocupação do canal por um novo fluxo, tratamento de interferências interfluxo e intrafluxo e violações de QoS.

A Figura 9 apresenta um fluxograma relativo à política de admissão do protocolo CAC-OLSR.

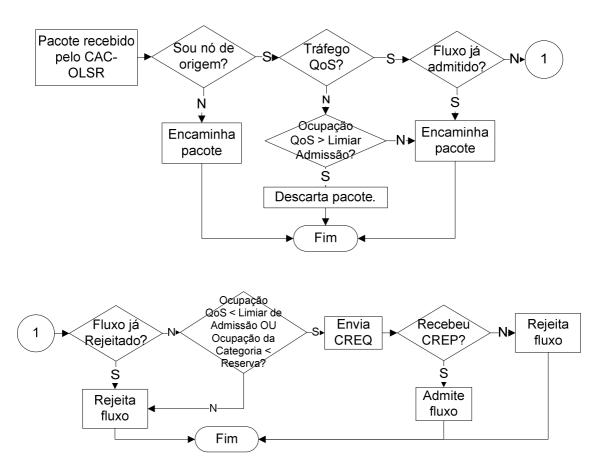


Figura 9: Política de Admissão CAC-OLSR.

Ao receber um pacote de dados, CAC-OLSR verifica se o mesmo é oriundo de um cliente do próprio nó mesh. Em caso negativo, o pacote é encaminhado normalmente como na versão original do OLSR. Embutida nesta ação está a premissa de que se um nó recebe um pacote para ser encaminhado, então este já passou pelo processo de admissão no nó mesh de origem.

Se o pacote foi originado em um cliente do nó em questão, então é verificado se este possui QoS associada, isto é, se será transmitido pelas categorias de acesso de voz ou vídeo. Esta verificação é feita mediante consulta ao campo ToS (Type of Service) ou, mais recentemente, DSCP (Differentiated Services Code Point) do cabeçalho IP, mapeando-se o valor encontrado em uma das quatro categorias de acesso do IEEE 802.11e.

Se o pacote não possuir QoS associada, isto é, se pertencer às categorias 2 ou 3 do IEEE 802.11e, então o pacote deve ser encaminhado, a menos que o nó verifique que a ocupação do meio com tráfego QoS é superior a um limiar de valor configurável, denominado "Limiar de Admissão". Neste caso, o pacote sem QoS deve ser descartado, evitando-se o congestionamento da rede.

Se o pacote pertence a um fluxo com QoS associada (categorias 0 ou 1), então será verificado se o mesmo já foi anteriormente admitido e se ainda possui entrada na tabela de admissão, a ser comentada mais adiante. Em caso positivo, o pacote é imediatamente enviado. Caso contrário, verifica-se se o fluxo já foi rejeitado dentro de um intervalo de tempo mínimo exigido para que um novo processo de admissão possa ser iniciado. Se positivo, este deve ser novamente rejeitado. Se negativo, o processo segue adiante.

É importante ressaltar que, para estas verificações, foi necessário criar uma tabela na base de dados de cada nó mesh com os fluxos em admissão, admitidos e rejeitados. Cada entrada nesta tabela possui um temporizador. Fluxos sem transmissão de pacotes por um determinado tempo serão excluídos e deverão novamente submeter-se à admissão. Fluxos que venham a ser rejeitados permanecem na tabela com status rejeitado, não sendo possível nova tentativa de admissão enquanto a respectiva entrada não for excluída até o estouro do temporizador.

Em seguida, o nó verifica se a ocupação do canal com fluxos com QoS é menor que o "Limiar de Admissão", bem como se a ocupação do canal com fluxos da mesma categoria daquele em admissão é menor que os recursos a ela reservados. Se ambas as condições não forem satisfeitas, o novo fluxo deve ser rejeitado. Por outro lado, se uma delas for atendida, o nó deve enviar uma mensagem do tipo CREQ – Check Request, a qual deve ser respondida pelo destino com um CREP – Check Reply. O atendimento apenas à segunda condição implica que a outra categoria de acesso com QoS excedeu os recursos a ela reservados e, portanto, seu último fluxo admitido será interrompido (violação de QoS).

Cada nó do caminho, ao receber um CREQ verifica a ocupação do meio percebida por si e por todos os seus vizinhos de um e dois saltos. Ressalta-se que esta consulta tem por objetivo combater o efeito da interferência interfluxo e é feita no próprio nó, uma vez que este conhece a ocupação de seus vizinhos, a qual lhe foi informada através de mensagens do tipo Hello do protocolo OLSR.

Esta ocupação é comparada à estimativa de ocupação do meio pelo novo fluxo, já com o fator CC aplicado, relativo ao efeito da interferência intrafluxo. Caso o próprio nó ou algum de seus vizinhos não possua recursos disponíveis, o CREQ deve ser descartado. Por outro lado, se todos possuírem os recursos necessários, o CREQ deve ser encaminhado ao próximo nó do caminho até o destino.

O nó destino, além de processar o CREQ, deve enviar um CREP, o qual percorrerá o caminho no sentido reverso. A verificação de recursos em cada nó é novamente realizada para que o CREP seja encaminhado. Caso o nó origem receba o CREP, deve processá-lo e, por fim, admitir o fluxo se possuir recursos disponíveis. A admissão do fluxo nos demais nós ao longo do caminho é efetuada apenas quando da transmissão de seu primeiro quadro de dados.

O nó de origem possui um temporizador para recebimento do CREP, sendo que, após o estouro deste, uma nova tentativa de admissão é realizada para diminuir casos de rejeição por erros de transmissão do CREQ e CREP. Caso haja reincidência de estouro do temporizador em questão, o fluxo é considerado como rejeitado. Neste caso, um novo processo de admissão só

será permitido após esgotado o temporizador de fluxos rejeitados, conforme anteriormente mencionado.

4.6. RESUMO DO CAPÍTULO

O presente capítulo discorreu sobre o modo de funcionamento da proposta CAC-OLSR, a qual se baseia na comparação entre o tempo de ocupação de canal estimado por um novo fluxo e a ocupação corrente do canal. Além disso, CAC-OLSR considera as categorias de acesso da emenda IEEE 802.11e, bem como implementa reserva de recursos para as categorias de voz e vídeo. As interferências interfluxo e intrafluxo são tratadas, além da violação de QoS.

O próximo capítulo apresenta os cenários constituídos para avaliação do CAC-OLSR.

5. CENÁRIOS PARA ANÁLISE DE DESEMPENHO DO CAC-OLSR

A análise de desempenho do protocolo CAC-OLSR foi realizada mediante simulações em *software* utilizando o NS-2 [NS2, 2010], tendo sido dividida em duas etapas a saber: análise de métricas de QoS e transmissão de fluxo de vídeo real.

Todas as simulações compararam CAC-OLSR com o protocolo OLSR original e com o mecanismo SWAN [Ahn et al., 2002a], previamente descrito no Capítulo 3. Cabe ressaltar que SWAN foi escolhido para fins de comparação com a proposta CAC-OLSR por alguns motivos, dentre os quais destacam-se a relevância deste trabalho, o qual inclusive tornou-se uma proposta de padrão IETF (possui um draft de RFC [Ahn et al., 2002b]), e o fato de seu código fonte ter sido disponibilizado pelos autores para simulações no *software* NS-2.

As Seções 5.1 e 5.2 descrevem respectivamente a análise de métricas de QoS e a transmissão de fluxo de vídeo real para fins de avaliação da proposta CAC-OLSR.

Os scripts de linguagem TCL (*Tool Command Language*) utilizados para implementar todos os cenários utilizados neste trabalho encontram-se descritos no Anexo A desta dissertação.

5.1. ANÁLISE DE MÉTRICAS DE QOS

A análise de métricas de QoS consistiu na montagem de um cenário de uma rede em malha sem fio em que diversos perfis de tráfego foram nela injetados, tendo-se coletado algumas métricas de QoS ao longo das simulações.

5.1.1. SIMULADOR

O *software* utilizado nas simulações foi o NS-2 (*Network Simulator 2*), versão 2.34. O NS-2 é um simulador de eventos discretos, de código aberto, empregado em pesquisas relacionadas a redes de comunicação [NS-2, 2010].

Ressalta-se que o código padrão do NS-2, na versão utilizada, não dispõe do protocolo de roteamento OLSR, tampouco implementa as categorias de acesso conforme previstas na emenda 802.11e. Deste modo, foi necessário

recorrer a implementações já desenvolvidas, como as que são descritas no parágrafo seguinte.

No caso do protocolo OLSR, foi utilizada a implementação para NS-2 desenvolvida pela Universidade de Murcia [Ros, 2009], totalmente aderente à RFC 3626 [Clausen e Jacquet, 2003], a qual descreve o protocolo em questão. No que diz respeito às categorias de acesso da emenda IEEE 802.11e, empregou-se a implementação desenvolvida pelo grupo TKN da Faculdade de Berlim [Wiethölter e Hoene, 2009].

Cabe destacar que a emenda IEEE 802.11e foi utilizada na avaliação do CAC-OLSR e do OLSR original, porém não pôde ser empregada juntamente com o SWAN, visto que este mecanismo, o qual independe de protocolo de roteamento, não considera diferentes categorias de acesso. A Tabela 3 apresenta um resumo dos protocolos de roteamento, camada MAC e camada Física configurados no NS-2 para cada um dos três cenários avaliados.

	Cenários Avaliados		
	CAC-OLSR	OLSR Original	SWAN
Protocolo de Roteamento	CAC-OLSR	OLSR	OLSR
Camada MAC	802.11e	802.11e	802.11
Camada Física	ERP-OFDM	ERP-OFDM	ERP-OFDM

Tabela 3: Protocolos de roteamento e camadas MAC e PHY.

Em todos os cenários, o NS-2 foi configurado para operar com o modelo de propagação *TwoRayGround*. As funções RTS/CTS [IEEE P802.11, 2007] foram desabilitadas e a taxa de dados utilizada foi de 54 Mbit/s, conforme especificações ERP-OFDM (*Extended Rate PHY - Orthogonal Frequency Division Multiplexing*) contidas no padrão IEEE 802.11. O alcance de transmissão dos nós foi configurado para 250 m e o alcance de detecção de portadora, para 550 m.

Em se tratando do CAC-OSLR e do OLSR original, foram configurados os valores contidos na Tabela 4 para os parâmetros das categorias de acesso da camada MAC. Estes valores são os sugeridos pela emenda IEEE 802.11e.

Tabela 4: Configuração dos parâmetros das categorias de acesso 802.11e.

Categoria de Acesso	AIFSN	CWmin	CWmax	TXOPLimit
Voz	2	7	15	3,264 ms
Vídeo	2	15	31	6,016 ms
Best Effort	3	31	1023	0
Background	7	31	1023	0

Além dos valores mostrados na Tabela 4, foram considerados adicionalmente para o CAC-OLSR os valores apontados na Tabela 5, os quais foram sugeridos por [Gerk, 2010]. Deste modo, o cenário CAC-OLSR foi avaliado sob dois conjuntos de parâmetros para as categorias de acesso.

Tabela 5: Configuração alternativa dos parâmetros das categorias de acesso 802.11e.

Categoria de Acesso	AIFSN	CWmin	CWmax	TXOPLimit
Voz	2	3	7	1,504 ms
Vídeo	9	7	15	3,008 ms
Best Effort	10	15	1023	0
Background	14	15	1023	0

O trabalho realizado por [Gerk, 2010] analisou diversas combinações de valores para os parâmetros das categorias de acesso IEEE 802.11e e concluiu que os indicados na Tabela 5 são adequados para um cenário similar ao aqui considerado. Esse trabalho não considerou o uso de mecanismos de controle de admissão na rede em malha.

5.1.2. PERFIL DE TRÁFEGO

Para avaliação dos cenários constituídos, quatro perfis de tráfego foram utilizados [Gerk, 2010], conforme detalhado a seguir:

- Voz: para modelar um tráfego semelhante a um fluxo de voz foi utilizada uma fonte de tráfego CBR (*Constant Bit Rate*), com pacotes de 160 bytes enviados a cada 20 ms. Isto equivale a uma taxa de 64 Kbit/s:
- Vídeo: para modelar um tráfego semelhante a um fluxo de vídeo foi utilizada uma fonte de tráfego CBR, com pacotes de 1280 bytes transmitidos a cada 10 ms. Isto equivale a uma taxa de 1024 Kbit/s.

- Melhor esforço: para modelar tráfego de melhor esforço, adotouse uma fonte de tráfego FTP com tamanho de pacote de 1300 bytes;
- Background: para modelar tráfego do tipo background, adotou-se uma fonte de tráfego FTP com tamanho de pacote de 1500 bytes.

Estes perfis de tráfego foram configurados para serem transmitidos pelas categorias de acesso a eles associadas, a menos da avaliação do mecanismo SWAN, o qual não considera as categorias mencionadas.

Cumpre destacar que os tráfegos de voz e vídeo para este cenário foram configurados como CBR. Sabe-se, no entanto, que na prática as aplicações de voz e vídeo podem demandar um perfil de tráfego variável. Deste modo, uma sugestão de trabalho futuro é a avaliação do CAC-OLSR em redes reais onde sejam injetados tráfegos reais de voz e vídeo.

5.1.3. CENÁRIO CONFIGURADO

Para avaliação do CAC-OSLR e sua comparação com o OSLR original e com o mecanismo de controle de admissão SWAN, foi montado um cenário de uma rede em malha sem fio composta por 10 nós posicionados aleatoriamente em um espaço de 500 m x 500 m. Este posicionamento aleatório foi fornecido pela ferramenta *setdest* do NS-2.

Uma vez montado o cenário descrito, iniciou-se a injeção de tráfego da seguinte maneira: a cada 20 s foi aleatoriamente inserido em um dos nós da rede um novo fluxo de cada tipo (voz, vídeo, melhor esforço e *background*). Os primeiros quatro fluxos, no entanto, iniciaram-se após 31 segundos de simulação, de modo a garantir a correta convergência do protocolo de roteamento. O tempo de simulação foi de 431 segundos.

As simulações em cada um dos três cenários foram repetidas 30 vezes e, ao final, algumas métricas foram avaliadas conforme será mais a frente detalhado.

Um segundo cenário foi configurado especificamente para avaliar o mecanismo de reação a violações de QoS da proposta CAC-OSLR, apresentado na Seção 4.4. Neste caso, analisou-se o CAC-OSLR com e sem o citado mecanismo. A Figura 10 ilustra este cenário.

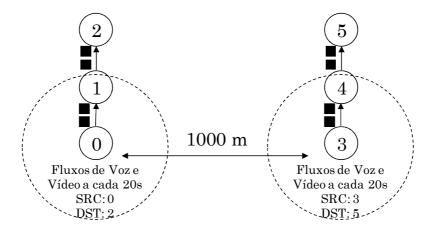


Figura 10: Cenário para avaliação de violação de QoS.

Na Figura 10, fluxos de voz e vídeo com o mesmo perfil de tráfego descrito anteriormente foram iniciados a cada 20 s, dos nós 0 e 3 para os nós 2 e 5, respectivamente. Os primeiros fluxos iniciaram-se com 31 s de simulação para fins de convergência de roteamento. O grupo de nós de 0 a 2 inicialmente foi separado do grupo de nós de 3 a 5 por uma distância de 1000 m (o cenário no NS-2 foi alterado para um espaço de 1000 m x 500 m). A distância entre os nós 0 e 1, 1 e 2, 3 e 4 e 4 e 5 foi configurada para 100 m e mantida constante até o final dos testes. Após 300 s de simulação, o grupo de nós de 3 a 5 deslocou-se horizontalmente, estacionando a uma distância de 10 m do grupo de 0 a 2, conforme mostra a Figura 11.

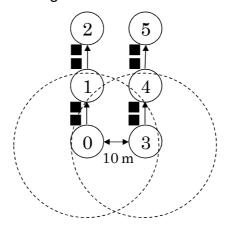


Figura 11: Cenário para avaliação de violação de QoS após 300 s.

Cumpre destacar que este último cenário corresponde a um dos piores casos para a ocorrência de violação de QoS, visto que os dois grupos de nós são inseridos na mesma região de interferência quando a ocupação de tempo de canal de cada um está próximo ao limite.

O tempo de simulação também foi de 431 s, com 30 repetições. As métricas avaliadas serão detalhadas a seguir.

5.1.4. CONFIGURAÇÕES DE PARÂMETROS DO CAC-OLSR

Conforme mencionado no Capítulo 4, CAC-OSLR dispõe de alguns parâmetros, os quais foram configurados conforme abaixo detalhado:

- Limiar de Admissão: 90% do tempo total de ocupação do canal;
- Limiar de violação de QoS: 95% do tempo total de ocupação do canal:
- Percentual de reserva para as categorias de voz e vídeo: foi adotado 55% de tempo de ocupação de canal para voz e 35% para vídeo. Deve-se lembrar que caso estes recursos não estejam sendo utilizados, serão colocados à disposição para as demais categorias de acesso.

Cabe observar que a configuração dos percentuais de reserva deve ser feito com cautela e de acordo com as políticas definidas pelo administrador da rede. Tais percentuais influenciarão significativamente na quantidade máxima de fluxos a serem admitidos em cada categoria. Além disso, deve-se lembrar que a utilização de recursos além daqueles reservados a uma determinada categoria fará com que fluxos admitidos nestas condições estejam sujeitos à interrupções repentinas em caso de reivindicação dos recursos pela categoria que sobre eles tem direito.

CAC-OLSR permite que a funcionalidade de reserva de recursos seja desabilitada, bastando para tal que os recursos destinados para cada categoria sejam configurados para 100%.

5.1.5. MÉTRICAS AVALIADAS

Para cada um dos cenários avaliados, isto é, CAC-OLSR, OLSR e SWAN, as seguintes métricas foram medidas:

- Fluxos Admitidos: corresponde à média de fluxos admitidos na rede, a cada intervalo de 20 s.
- Atraso médio fim-a-fim: corresponde à média do atraso percebido,
 a cada intervalo de 20 s, pelos pacotes transmitidos entre os nós

origem e destino. Esta métrica foi aferida apenas para os fluxos com QoS, separando-se os fluxos de voz e vídeo.

- Vazão média agregada: corresponde à média de bits por segundo recebida pelos nós da rede, a cada intervalo de 20 s. A vazão média agregada foi analisada sob três ópticas: fluxos com QoS, fluxos sem QoS e total de fluxos. No caso de fluxos com QoS, esta métrica foi medida considerando-se todos os fluxos de voz e todos os fluxos de vídeos separadamente. No caso dos fluxos sem QoS, considerou-se uma única medida, tomando-se todos os fluxos de melhor esforço e background em conjunto. Na visão denominada total de fluxos, analisou-se a vazão média agregada considerando-se todos os fluxos vigentes, com ou sem QoS.
- Vazão média por fluxo: corresponde à média de bits por segundo recebida pelos nós da rede, a cada intervalo de 20 s, dividida pelo número de fluxos em vigor. Esta métrica foi aferida apenas para os fluxos com QoS, separando-se os fluxos de voz e vídeo.

Especificamente para o CAC-OLSR, a média de *overhead* de controle adicional inserido por esta proposta foi aferida. Para tanto, mediu-se a quantidade e o tamanho em bytes de todas as mensagens CREQ e CREP, comparando-se os valores encontrados com a quantidade e o tamanho em bytes de todas as mensagens de controle enviadas pelo protocolo de roteamento durantes as simulações. Além disso, também foi avaliado o atraso médio percebido pelo primeiro pacote recebido para os fluxos de voz e vídeo, de modo a se verificar o impacto da troca de mensagens CREQ e CREP nesta métrica.

No caso do cenário montado para fins de avaliação do mecanismo de reação a violação de QoS, foram medidos o atraso médio fim-a-fim e a vazão média por fluxo, conforme definições descritas anteriormente.

5.2. TRANSMISSÃO DE FLUXO DE VÍDEO REAL

Com a finalidade de aproximar as simulações e consequente avaliação do CAC-OSLR de um cenário mais real, foram realizados testes com um vídeo real, avaliando-se sua qualidade após a transmissão no ambiente de uma rede em malha sem fio de múltiplos saltos constituído no NS-2.

5.2.1. SIMULADOR

Para esta avaliação dois simuladores em *software* foram empregados: NS-2 e Evalvid [Evalvid, 2011].

Evalvid é um conjunto de ferramentas desenvolvido para fins de avaliação da qualidade de um vídeo transmitido sobre uma rede de comunicação real ou virtual. Destina-se a pesquisadores que desejem avaliar modelagens e/ou configurações de rede com relação à qualidade de vídeo percebida por um usuário [Evalvid, 2011].

Além de medir algumas métricas como atraso, variação do atraso e perda de pacotes, o conjunto de ferramentas disponibilizado pelo Evalvid possibilita a medição da qualidade do vídeo recebido, aferida através da relação PSNR (*Peak Signal-to-noise Ratio*).

Para melhor entendimento da ferramenta em questão, a Figura 12 ilustra sua arquitetura.

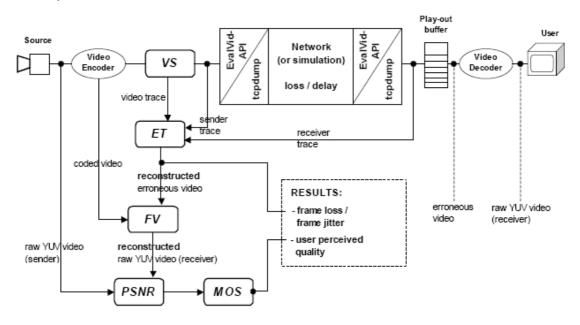


Figura 12: Arquitetura do conjunto de ferramentas Evalvid.

Fonte [Evalvid, 2011]

As ferramentas disponibilizadas pelo Evalvid encontram-se identificadas na Figura 12 através de caixas retangulares. A seguir, descreve-se brevemente cada uma delas, conforme extraído de [Evalvid, 2011].

• VS: trata-se da ferramenta *Video Sender* a qual é responsável por analisar um vídeo comprimido e prepará-lo para a transmissão em uma

rede real ou simulada. Esta preparação inclui a identificação dos quadros, fragmentação de quadros grandes e inserção de uma marca de tempo (*timestamp*) em cada um;

- ET: trata-se da ferramenta *Evaluate Trace* a qual é responsável por reconstruir o vídeo recebido com base no vídeo original e nos arquivos de *trace* transmitido e recebido. ET também calcula métricas como atraso e jitter;
- FV: trata-se da ferramenta *Fix Video*, a qual compara os vídeos transmitido e recebido e insere o último quadro decodificado com sucesso, caso identifique a perda de determinado quadro;
- PSNR: trata-se da ferramenta que, com base nas componentes de luminância dos vídeos original e recebido, não comprimidos, calcula a relação sinal ruído, PSNR, quadro a quadro, conforme a Equação 6.

$$PSNR(n)_{dB} = 20 \log_{10} \left(\frac{V_{peak}}{\sqrt{\frac{1}{N_{col} N_{row}} \sum_{i=0}^{N_{col}} \sum_{j=0}^{N_{row}} [Y_{s}(n,i,j) - Y_{D}(n,i,j)]^{2}}} \right)$$

$$V_{peak} = 2^{k} - 1$$
(8)

Na Equação (8), Y_S é a componente de luminância do vídeo original, Y_D , a do vídeo recebido e k é o número de bits por pixel da componente de luminância.

 MOS: trata-se da ferramenta que analisa o PSNR quadro a quadro do vídeo recebido, atribuindo uma nota segundo a escala de MOS (*Mean Opinion Score*), conforme mostra a Tabela 6.

Escala	Qualidade	Comprometimen to
5	Excelente	Imperceptível
4	Bom	Perceptível, mas não irritante
3	Justo	Pouco irritante
2	Pobre	Irritante
1	Ruim	Muito irritante

Tabela 6: Escala MOS.

A conversão dos valores de PSNR para a escala MOS é feita com base na Tabela 7.

Tabela 7: Conversão PSNR - MOS.

PSNR (dB)	MOS
> 37	5 (Excelente)
31-37	4 (Bom)
25-31	3 (Justo)
20-25	2 (Irritante)
<20	1 (Ruim)

Diante da Figura 12 e do breve explicativo das ferramentas, conclui-se que o Evalvid opera da seguinte maneira: em primeiro lugar, deve-se tomar um vídeo de referência, o qual posteriormente será utilizado para calcular o PSNR quadro a quadro do vídeo recebido. O vídeo selecionado é então preparado para transmissão em rede real ou simulada através da ferramenta VS. No caso deste trabalho, utilizou-se uma rede simulada implementada na ferramenta NS-2. Após as simulações em rede e de posse dos arquivos de trace dos vídeos enviado e recebido, a ferramenta ET reconstrói o vídeo recebido, possivelmente com erros, bem como disponibiliza métricas de atraso, variação de atraso e perda de pacotes. A ferramenta FV insere os últimos quadros codificados com sucesso no lugar de quadros perdidos. Finalmente, utilizam-se as ferramentas PSNR e MOS para, respectivamente, calcular o PSNR quadro a quadro e avaliar a qualidade do vídeo recebido.

5.2.2. PERFIL DE TRÁFEGO

O perfil de tráfego empregado neste teste foi o mesmo descrito na Seção 5.1.2. Adicionalmente foi utilizado um vídeo real denominado "News", obtido de [Evalvid, 2011]. Este vídeo tem formato QCIF (resolução 176 x 144), 30 quadros por segundo, taxa média de 310 kbps e duração de 12 s. Para melhor avaliação de desempenho, o vídeo foi replicado quatro vezes, isto é, a reprodução do mesmo durou 48 s.

5.2.3. CENÁRIO CONFIGURADO

Para fins de avaliação da transmissão de fluxo de vídeo real para os cenários CAC-OSLR, OLSR original e SWAN, simulou-se um cenário de uma rede em malha sem fio composta por 10 nós posicionados aleatoriamente em um espaço de 500 m x 500 m, tal qual descrito anteriormente.

Uma vez montado o cenário descrito, iniciou-se a injeção de tráfego da seguinte maneira: a cada 10 s foi aleatoriamente inserido em um dos nós da rede um novo fluxo de cada tipo (voz, vídeo, melhor esforço e background). Os primeiros quatro fluxos, no entanto, iniciaram-se após 31 segundos de simulação, de modo a garantir a correta convergência do protocolo de roteamento. Vale observar que esta injeção de tráfego na rede é a mesma utilizada no teste anterior.

Adicionalmente, após 50 s de simulação, foi inserido o vídeo "News", sendo os nós de origem e destino escolhidos aleatoriamente. Este vídeo foi codificado no padrão MPEG-4, com GOP (*Group of Pictures*) de 30 quadros utilizando-se quadros P e B.

O tempo de simulação foi de 431 segundos, com 30 repetições. Ao final, foram considerados os arquivos de trace dos vídeos transmitido e recebido, os quais foram tratados pelas ferramentas do Evalvid para análise da qualidade.

5.2.4. CONFIGURAÇÕES DE PARÂMETROS DO CAC-OLSR

As configurações dos parâmetros do CAC-OSLR foram as mesmas descritas no item 5.1.4.

5.2.5. MÉTRICAS AVALIADAS

As métricas avaliadas neste teste foram:

- PSNR médio: nível médio de PSNR quadro a quadro;
- MOS médio: percentual médio de quadros em cada nota da escala MOS;
- Atraso médio fim-a-fim: média do atraso medido entre os quadros transmitidos e recebidos. Diferentemente do teste anterior, o atraso foi medido em termos das curvas de densidade de probabilidade e probabilidade cumulativa, CDF (Cumulative Distribution Function) e PDF (Probability Density Function) respectivamente;

• Perda média de pacotes: média percentual de pacotes perdidos nas simulações efetuadas.

6. RESULTADOS DA ANÁLISE DE DESEMPENHO

Este capítulo apresenta os resultados obtidos após efetuados os testes propostos no Capítulo 5 através de simulações com os softwares NS-2 e Evalvid.

6.1. ANÁLISE DE MÉTRICAS DE QOS

Conforme previsto, foram avaliadas as métricas de atraso médio fim-afim, vazão média agregada, vazão média por fluxo e fluxos admitidos para os
cenários CAC-OLSR, OLSR original e SWAN. Cumpre destacar que o cenário
CAC-OLSR foi avaliado sob duas configurações de parâmetros para as
categorias de acesso do padrão IEEE 802.11, a saber: valores sugeridos pelo
próprio padrão IEEE 802.11 [IEEE P802.11, 2007] e valores sugeridos por
[Gerk, 2010].

Além disso, para o CAC-OSLR foram avaliados o *overhead* adicional introduzido pelo mecanismo de controle de admissão, bem como o mecanismo de reação a violações de QoS. Neste caso, as categorias de acesso da camada MAC foram configuradas apenas com os valores sugeridos pelo padrão IEEE 802.11.

Cabe ressaltar que, para todos os cenários, os resultados a serem apresentados foram obtidos a partir de uma média de 30 simulações. Para cada média calculada, foi computado o intervalo de confiança de 95%, representado nos gráficos por uma barra vertical de erros.

Para fins de clareza, a Tabela 8 registra a correspondência entre os cenários considerados e a legenda que será empregada nos gráficos.

 Cenário
 Legenda

 SWAN
 SWAN

 OLSR Original
 OLSR+802.11e

 CAC-OLSR com parâmetros das categorias de acesso configurados conforme [IEEE P802.11, 2007]
 CAC-OLSR+802.11e

 CAC-OLSR com parâmetros das categorias de acesso configurados conforme [GERK, 2010]
 CAC-OLSR+802.11e_mod

Tabela 8: Correspondência entre cenários e legenda

6.1.1. FLUXOS ADMITIDOS

Esta seção apresenta a média de fluxos admitidos em cada cenário de simulação. As Figuras 13 e 14 apresentam os resultados obtidos para as categorias de voz e vídeo.

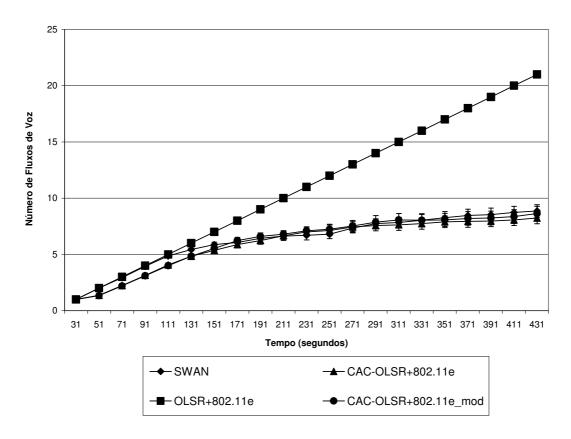


Figura 13: Fluxos de voz admitidos.

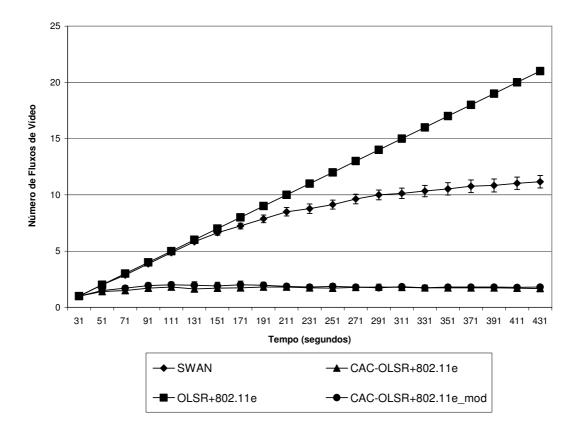


Figura 14: Fluxos de vídeo admitidos

Cumpre destacar que o cenário OLSR original não possui mecanismo de controle de admissão. Deste modo, a quantidade de fluxos de voz e vídeo admitidos na rede aumenta linearmente até o fim da simulação, atingindo 21 fluxos para voz e 21 para vídeo.

Para o cenário SWAN, observa-se que houve admissão de aproximadamente 8 fluxos de voz e 11 de vídeo após 431 s de simulação. Conforme será visto nas métricas de atraso e vazão, SWAN não conseguiu satisfazer os requisitos de QoS dos fluxos, o que indica que permitiu a entrada excessiva de fluxos na rede além da capacidade desta. Atribui-se este excesso de fluxos ao fato de o mecanismo não levar em consideração a ocupação do meio entre a região de transmissão e de detecção de portadora de cada nó, além de não considerar o fenômeno de interferência intrafluxo.

Finalmente, o cenário CAC-OSLR admitiu, em média, aproximadamente 8 fluxos de voz e 2 de vídeo após 431 s de simulação para ambos os casos (categorias de acesso configuradas conforme a emenda IEEE 802.11e e valores modificados). Como os requisitos de QoS foram atendidos (vide

gráficos de atraso e vazão a seguir) e a vazão média agregada de voz e de vídeo superou a das demais propostas, pode-se concluir que a quantidade apropriada de fluxos foi admitida.

A quantidade de fluxos admitidos pelo CAC-OLSR nas categorias de voz e vídeo está associada à reserva de recursos destinada a cada uma. Cabe ressaltar, no entanto, que estes recursos levam em consideração o tempo estimado de ocupação do canal, e não simplesmente a banda requerida por cada fluxo. O tempo de ocupação do canal considera procedimentos de contenção e depende da quantidade de saltos entre origem e destino do fluxo.

6.1.2. ATRASO MÉDIO FIM-A-FIM

Com relação ao atraso médio fim-a-fim, os resultados encontrados para os fluxos de voz e vídeo encontram-se representados, respectivamente, nas Figuras 15 e 16.

Ressalta-se, conforme já explicado no capítulo anterior, que cada valor representado nos gráficos corresponde à média do atraso de pacotes recebidos nos últimos 20 s, considerando-se a média de 30 simulações. Além disso, a cada intervalo de 20 s, um novo fluxo de cada perfil (voz, vídeo, melhor esforço e *background*) é iniciado. No caso do cenário com uso do OLSR original, estes fluxos são injetados diretamente na rede. No caso do CAC-OLSR e SWAN, os mecanismos de controle de admissão regulam a injeção de tráfego na rede.

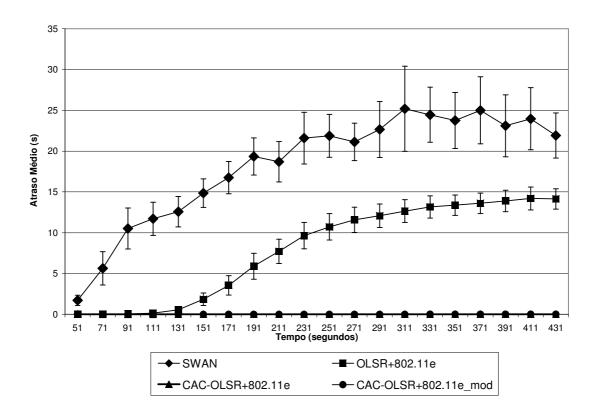


Figura 15: Atraso médio fim-a-fim para fluxos de voz.

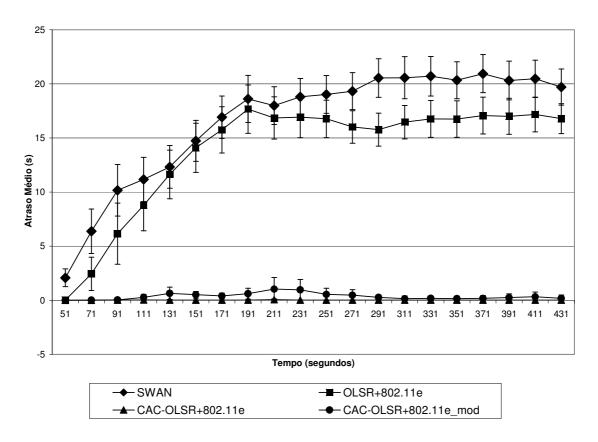


Figura 16: Atraso médio fim-a-fim para fluxos de vídeo.

Conforme mostra a Figura 15, o atraso médio fim-a-fim para os fluxos de voz chegou próximo a 15 s com o OLSR original e atingiu 25 s com SWAN. Por outro lado, CAC-OLSR foi capaz de manter esta métrica abaixo de 150 ms para as duas configurações de parâmetros das categorias de acesso, valor aceitável para aplicações de voz.

Um fato interessante a se observar é que o atraso observado com o uso do SWAN foi maior do que com o uso do OLSR original, o qual não possui controle de admissão. Deve-se lembrar, no entanto, que o cenário com OLSR original contou com a diferenciação de tráfego mediante uso das categorias de acesso da emenda 802.11e, fato não considerado no uso do mecanismo SWAN, uma vez que este não prevê o uso das citadas categorias.

Já para o tráfego de vídeo, CAC-OLSR manteve o atraso médio fim-afim abaixo de 500 ms com parâmetros das categorias de acesso configurados conforme a emenda IEEE 802.11e. Esta mesma métrica chegou a aproximadamente 1 s com a configuração modificada das categorias de acesso, porém se manteve abaixo deste valor.

Para o OLSR original e SWAN, o atraso médio fim-a-fim passou de 15 s para o primeiro cenário e superou 20 s para o segundo.

Conclui-se que o mecanismo de controle de admissão da proposta CAC-OLSR foi capaz de manter a métrica atraso em valores aceitáveis para aplicações de voz e vídeo [Kurose e Ross, 2004], fato que não ocorreu com as demais propostas. Deve-se considerar o alto nível de injeção de tráfego na rede (um fluxo de cada categoria de acesso a cada 20 s). Mesmo assim novos fluxos admitidos e fluxos vigentes não tiveram a métrica atraso degradada com o uso do CAC-OLSR.

6.1.3. VAZÃO MÉDIA AGREGADA

A medição da vazão média agregada foi dividida em fluxos com QoS, fluxos sem QoS e total de fluxos.

6.1.2.1. VAZÃO MÉDIA AGREGADA PARA FLUXOS COM QOS

No caso de fluxos com QoS, a vazão média agregada foi ainda separada em vazão média agregada de voz e de vídeo.

Cabe lembrar que a vazão média agregada considera a média de bit/s recebida a cada 20 s, por todos os nós, do perfil de tráfego em questão (no caso voz ou vídeo).

As Figuras 17 e 18 mostram os resultados para voz e vídeo respectivamente.

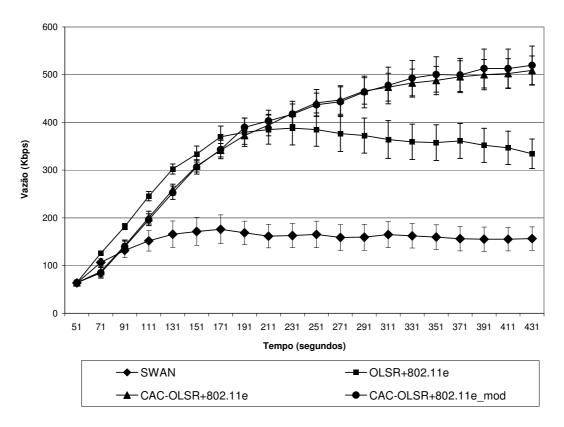


Figura 17: Vazão média agregada para fluxos de voz.

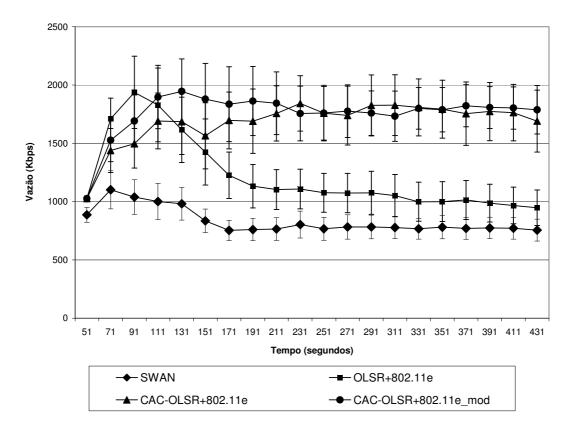


Figura 18: Vazão média agregada para fluxos de vídeo.

A Figura 17 mostra que, para fluxos de voz, o cenário com uso do OLSR original obteve maior vazão média agregada até 191 s. Deve-se lembrar que este cenário não restringe a entrada de fluxos na rede, o que acontece com CAC-OSLR e SWAN.

Contudo, mesmo com uma menor quantidade de fluxos de voz admitidos na rede, CAC-OLSR superou a vazão média agregada percebida pelo cenário OLSR original após 191 s, o qual provavelmente foi impactado pela saturação da rede e, consequentemente, maior tempo despendido com contenção na camada MAC. Nota-se que não houve diferenças significativas entre os cenários CAC-OSLR com categorias de acesso configuradas de acordo com a emenda IEEE 802.11e e com os valores modificados.

No cenário SWAN, além da restrição de fluxos na rede provocada pelo controle de admissão, a não diferenciação e priorização de tráfego em virtude da ausência das categorias de acesso da emenda 802.11e fizeram com que este cenário obtivesse o menor valor de vazão média agregada.

Comportamento semelhante foi observado para os fluxos de vídeo, conforme ilustra a Figura 18. Inicialmente a vazão média agregada apresentada pelo OLSR original foi maior, sendo ultrapassada pelo CAC-OSLR

após 111 s de simulação. SWAN apresentou a menor vazão média agregada, provavelmente pelos mesmos motivos já relatados no parágrafo anterior.

Com relação à comparação entre os cenários CAC-OLSR com parâmetros das categorias de acesso configurados conforme a emenda IEEE 802.11e e com valores modificados, percebe-se que o cenário com valores modificados foi superior até 211 s de simulação. Após este instante, os resultados foram similares. Esta superioridade até os 211 s provavelmente ocorreu em virtude de o cenário original ter permitido uma maior vazão inicial de tráfego sem QoS, conforme mostrará a Figura 19.

6.1.2.2. VAZÃO MÉDIA AGREGADA PARA FLUXOS SEM QOS

No caso de fluxos sem QoS, a vazão média agregada foi medida considerando-se todo o tráfego de categoria melhor esforço e *background*.

A Figura 19 indica que CAC-OLSR também apresentou melhores resultados frente às outras propostas.

O cenário CAC-OLSR com parâmetros das categorias de acesso configurados conforme a emenda IEEE 802.11e foi superior ao CAC-OLSR com valores modificados até 211 s, sendo ultrapassado por este após este instante. Este comportamento provavelmente está associado à própria configuração dos parâmetros, conforme indicado nas Tabelas 4 e 5. O cenário configurado conforme o padrão possui menor AIFS para melhor esforço e background, porém a configuração modificada possui menores valores de janela de contenção mínima. No início das simulações, quando a rede estava com abundância de recursos disponíveis, o menor valor de AIFS fez com que a vazão do cenário original fosse maior para as classes melhor esforço e background. Porém, conforme a rede foi atingindo níveis próximos ao seu limite de capacidade, o que aumentou a quantidade de processos de contenção, o menor valor de janela de contenção fez com que o cenário modificado ultrapassasse o original.

A diminuição na vazão média agregada do tráfego sem QoS para todos os cenários foi provocada principalmente pelo controle proporcionado pelo protocolo TCP.

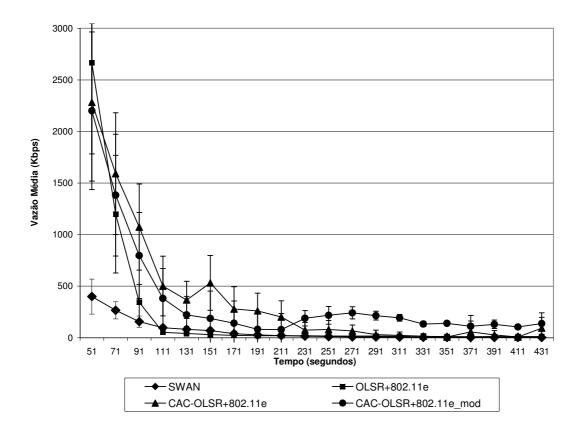


Figura19: Vazão média agregada para fluxos sem QoS.

6.1.2.3. VAZÃO MÉDIA AGREGADA PARA TOTAL DE FLUXOS

Esta seção analisa comparativamente os cenários em avaliação com relação à vazão média agregada considerando todos os fluxos vigentes, com ou sem QoS. A Figura 20 apresenta os resultados.

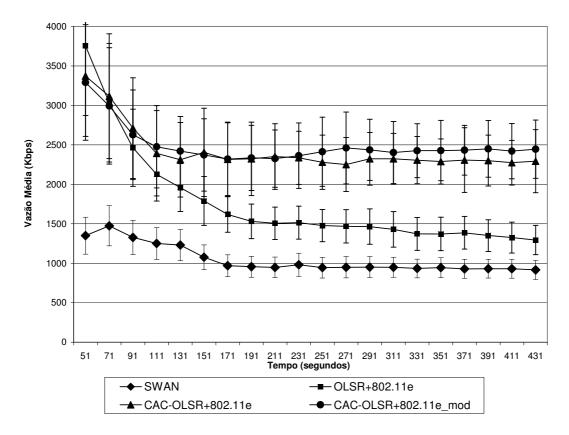


Figura 20: Vazão média agregada para todos os fluxos.

Conforme mostra a Figura 20, CAC-OLSR apresentou a maior vazão média agregada, a qual estabilizou em aproximadamente 2300 kbps para a configuração de parâmetros original e 2400 kbps para a configuração modificada. O cenário OLSR original iniciou com o maior valor de vazão média, porém encerrou as simulações abaixo de 1500 kbps. SWAN apresentou valores próximos a 1000 kbps.

Atribui-se os melhores resultados observados no cenário CAC-OLSR à atuação do controle de admissão. O mecanismo regulou a injeção de tráfego na rede e previniu que esta experimentasse situações de congestionamento, o que demandaria maior tempo a ser despendido com procedimentos de contenção.

6.1.4. VAZÃO MÉDIA POR FLUXO

A vazão média por fluxo considera a média de bit/s recebida de determinada categoria de tráfego a cada 20 s, dividida pela quantidade de fluxos vigentes na rede deste mesmo perfil de tráfego.

As Figuras 21 e 22 mostram os resultados para voz e vídeo respectivamente.

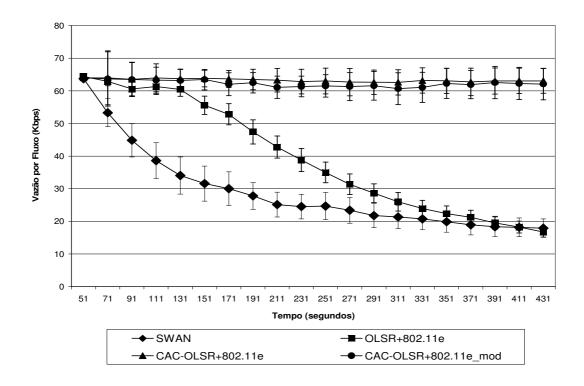


Figura 21: Vazão média por fluxo para voz.

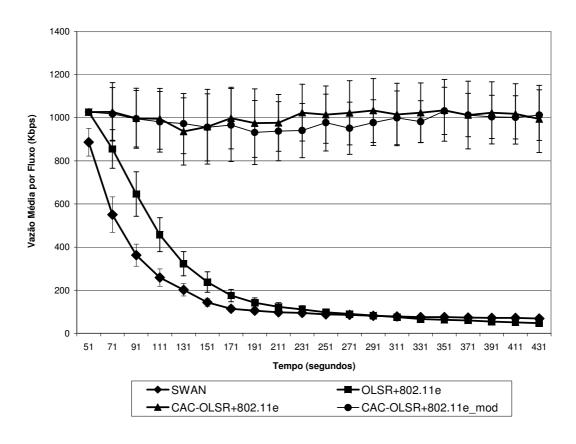


Figura 22: Vazão média por fluxo para vídeo.

Pela análise da Figura 21, pode-se observar que apenas CAC-OLSR foi capaz de atender à vazão de 64 Kbit/s demandada pelos fluxos de voz durante todo o tempo de simulação. Já no cenário OLSR original, observa-se que a diferenciação e priorização do tráfego de voz proporcionadas pela emenda 802.11e foram capazes de atender à vazão de 64 Kbit/s por fluxo somente até 131 s de simulação. Após este instante, houve uma queda acentuada nesta métrica. Finalmente, no caso do mecanismo SWAN, a vazão média por fluxo caiu rapidamente, porém, mesmo sem utilizar a diferenciação de tráfego na camada MAC, SWAN ultrapassou os resultados obtidos pelo uso do OLSR original no final da simulação, tendo em vista a atuação do mecanismo de controle de admissão.

No caso dos fluxos de vídeo, a Figura 22 mostra que mais uma vez apenas CAC-OLSR foi capaz de sustentar a vazão próxima ao valor solicitado, isto é, 1024 Kbit/s. OLSR original só conseguiu manter este mesmo valor para o primeiro fluxo admitido. SWAN apresentou, inicialmente, os menores valores. Porém, observa-se que aos 271 s igualou-se ao OLSR original e ultrapassou-o em seguida.

Atribui-se esta recuperação ao mecanismo de controle de admissão implementado por SWAN. Ainda que SWAN não implemente a diferenciação de tráfego proporcionada pela emenda IEEE 802.11e, a regulação de injeção de tráfego na rede fez com que os resultados de vazão por fluxo obtidos superassem os do cenário OLSR original, o qual se valeu da citada emenda.

Destaca-se que tanto para voz quanto para vídeo, os resultados obtidos para os dois cenários considerados para o CAC-OLSR, considerando valores diferentes para os parâmetros IEEE 802.11e, foram similares.

6.1.5. OVERHEAD ADICIONAL DE CONTROLE

Uma preocupação quanto à adoção do CAC-OLSR é o acréscimo de overhead de controle na rede em função das mensagens CREQ e CREP. No entanto, conforme mostra a Tabela 9, este acréscimo foi, em média, inferior a 6%. A tabela mostra que da média de 8343,10 pacotes de controle enviados pelo CAC-OLSR, 464,57 eram de novas mensagens de controle CREQ ou CREP, o que representa 5,57% do total. Cabe ressaltar que os demais pacotes

são os normalmente enviados pelo OSLR original (Hello, mensagens de topologia – TC, dentre outros). Com respeito à quantidade adicional de bytes injetada na rede por conta de CREQ e CREP, as simulações demonstraram que este acréscimo foi de 4,04%.

Tabela 9: Overhead Adicional

Descrição	CREQ or CREP	Pacotes de Controle	%
Pacotes Enviados	464,57	8343,10	5,57%
Quantidade de bytes	44311,40	1097641,40	4,04%

Ainda com relação ao overhead adicional de controle, também foi avaliado o atraso médio fim-a-fim para a recepção do primeiro pacote de voz e vídeo de cada fluxo. Objetivou-se com esta medição verificar eventuais impactos em função da troca de mensagens CREQ e CREP.

Para os fluxos de voz, observou-se um atraso médio de 32 ms para a recepção do primeiro pacote, sendo que 95% de todas as medidas situaram-se no intervalo de 0 a 87 ms. Estes valores estão dentro dos valores admissíveis para o tráfego de voz [Kurose e Ross, 2004].

Para os fluxos de vídeo, observou-se um atraso médio de 56 ms para a recepção do primeiro pacote, sendo que 95% de todas as medidas situaram-se no intervalo de 0 a 117 ms. Estes valores estão dentro dos valores admissíveis para o tráfego de vídeo [Kurose e Ross, 2004].

6.1.6. REAÇÃO A VIOLAÇÕES DE QOS

O cenário proposto para avaliar a recuperação a violações de QoS é extremamente severo. Conforme descrito no Capítulo 5, os dois grupos de nós são colocados na mesma região de interferência a partir dos 300 s de simulação, quando cada um deles está com quase 100% de seus recursos de canal ocupados por fluxos de voz e vídeo. A avaliação foi feita apenas com a proposta CAC-OLSR, com e sem o mecanismo de reação a violações de QoS.

As Figuras 23 e 24 mostram o atraso médio fim-a-fim para fluxos de voz e vídeo, respectivamente.

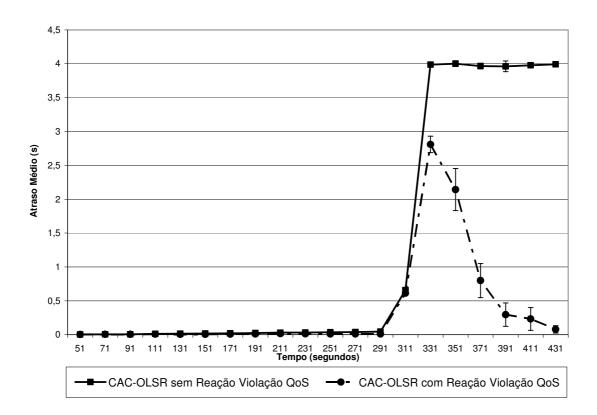


Figura 23: Atraso médio fim-a-fim para voz

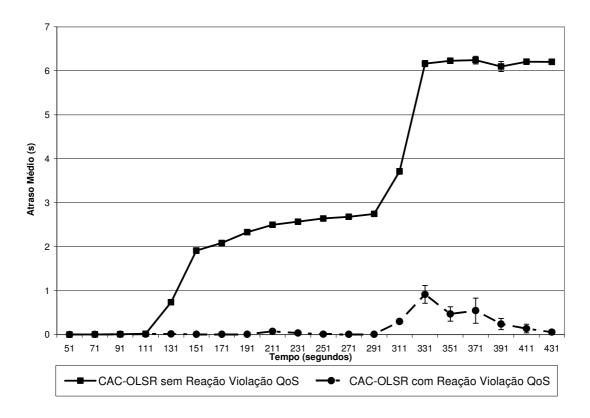


Figura 24: Atraso médio fim-a-fim para vídeo

Conforme pode-se observar, o mecanismo de reação a violação de QoS restabeleceu o atraso médio fim-a-fim para valores inferiores a 100 ms, tanto para voz quanto para vídeo, mesmo diante de uma mudança extrema nas condições de ocupação do canal. A mesma métrica findou em aproximadamente 4 s e 6 s para voz e vídeo respectivamente.

Particularmente para os fluxos de vídeo, observa-se um aumento considerável no atraso médio a partir de 111 s na ausência do mecanismo de reação a violações de QoS. Entretanto, neste instante os dois grupamentos de nós ainda não haviam sido colocados na mesma região de interferência. O aumento de atraso neste caso ocorreu em virtude de os fluxos de vídeo terem inicialmente ocupado recursos de canal que estavam ociosos, porém pertenciam à reserva de recursos da categoria de voz. À medida que novos fluxos de voz foram sendo admitidos, alguns fluxos de vídeo deveriam ter sido interrompidos (vide Seção 4.4), mas não o foram, provocando assim a degradação no atraso médio.

As Figuras 25 e 26 mostram a métrica vazão média por fluxo.

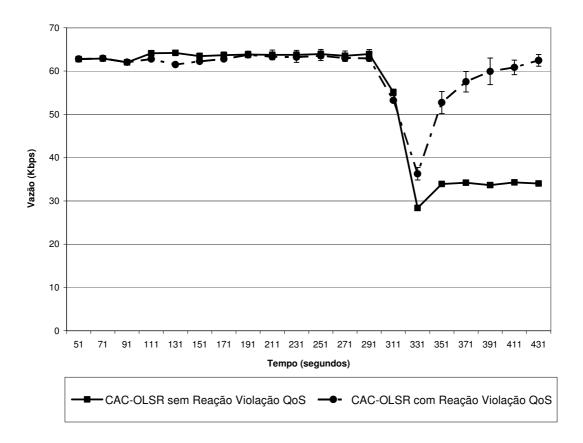


Figura 25: Vazão média por fluxo para voz

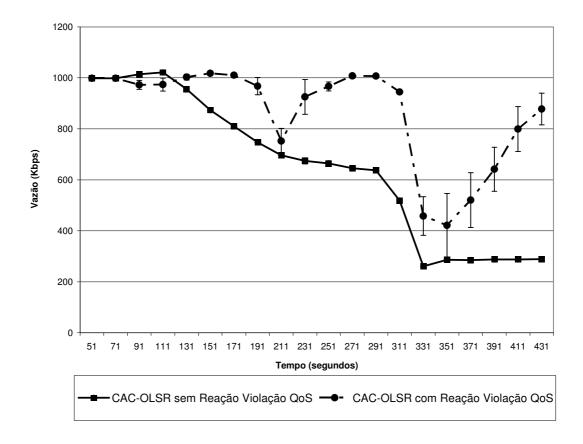


Figura 26: Vazão média por fluxo para vídeo

Nota-se que o mecanismo em avaliação recuperou a vazão média por fluxo para a categoria de voz, retomando a referida métrica a valores próximos dos 64 kbps, conforme originalmente demandado.

Para os fluxos de vídeo, semelhantemente aos resultados para a métrica atraso médio fim-a-fim, a vazão média por fluxo começou a ser degradada a partir dos 111 s na ausência do mecanismo. A explicação é a mesma citada anteriormente, isto é, novos fluxos de voz foram sendo admitidos por fazer jus aos seus recursos reservados de canal, porém não houve interrupção de fluxos de vídeo que estavam ocupando os recursos alocados para voz.

Após os 300 s, há uma queda abrupta na vazão média por fluxo, sendo que sem o mecanismo esta métrica encerrou as simulações com valores próximos a 300 kbps frente aos 1024 kbps solicitados. Já com o mecanismo, houve recuperação da vazão, a qual findou em valores próximos a 900 kbps.

6.2. TRANSMISSÃO DE VÍDEO REAL

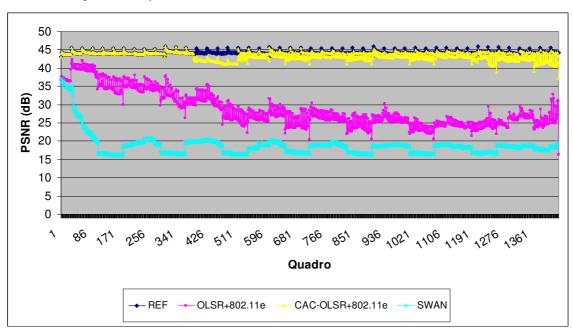
Conforme detalhado no Capítulo 5, a transmissão de vídeo real foi realizada com auxílio da ferramenta Evalvid, além do simulador de redes NS-2.

Novamente foram avaliados três cenários, isto é, CAC-OLSR, OLSR original e SWAN. No caso do CAC-OLSR, tendo em vista a similaridade entre os resultados obtidos no teste anterior para as configurações das categorias de acesso proposta por [Gerk, 2010] e pela emenda IEEE 802.11e, utilizou-se apenas a configuração sugerida pelo padrão mencionado, a qual apresentou ligeira superioridade no que diz respeito ao atraso médio fim-a-fim e vazão média por fluxo para a categoria de vídeo (vide Figuras 14 e 20).

As métricas medidas foram o PSNR médio, MOS médio, atraso médio fim-a-fim e perda média de quadros. Os resultados são apresentados a seguir.

6.2.1. PSNR MÉDIO

O PSNR médio representa a média da relação sinal ruído, quadro a quadro do vídeo transmitido, considerando-se as simulações efetuadas em cada cenário.



A Figura 27 apresenta os resultados obtidos.

Figura 27: PSNR Médio

Na Figura 27, pode-se observar quatro medições de PSNR. A primeira diz respeito ao valor de referência, "REF", o qual corresponde à relação sinal ruído do vídeo original comprimido, antes da transmissão. A segunda refere-se ao cenário CAC-OLSR, a terceira ao OSLR original e a quarta ao cenário SWAN.

Com relação ao CAC-OLSR, constata-se que os níveis obtidos de PSNR ficaram acima de 40 dB, próximos aos do vídeo original.

No OLSR original, o nível de PSNR dos quadros apresentou maior queda com relação ao vídeo de referência. A maior parte dos quadros situou-se entre 20 e 30 dB para PSNR.

Finalmente, SWAN apresentou os piores resultados, sendo que a maior parte dos quadros apresentou PSNR entre 15 e 25 dB.

6.2.2. MOS MÉDIO

O MOS médio representa a média percentual de quadros em cada uma das notas da escala MOS, considerando-se as simulações efetuadas em cada cenário.

Cumpre destacar que esta métrica é mera consequência dos resultados de PSNR médio, levando-se em conta o critério de conversão entre PSNR e MOS apresentados na Tabela 7.

100% Percentual de Quadros por MOS 90% 80% 70% 60% 50% 40% 30% 20% 10% 0% Referência CAC-OLSR+802.11e **SWAN** OLSR+802.11e ■ Ruim (1) ■ Pobre (2) □ Justo (3) □ Bom (4) ■ Excelente (5)

A Figura 28 ilustra os resultados obtidos.

Figura 28: MOS Médio

O vídeo original comprimido, identificado na Figura 28 como "Referência", apresentou 100% dos quadros dentro dos limites de PSNR exigidos para o grau 5 (Excelente) da escala MOS.

Para CAC-OSLR, embora tenha se observado ligeira queda com relação ao vídeo de referência, a média de PSNR de todos os quadros recebidos ficou acima do limite exigido para o grau 5 (Excelente).

No cenário OLSR original, a média das simulações resultou em aproximadamente: 8% dos quadros com grau 5 (Excelente), 20% com grau 4 (Bom), 45% com grau 3 (Justo) e 27% com grau 2 (Pobre).

No cenário SWAN, em média, mais de 85% dos quadros recebidos foram classificados no grau 1 (Ruim), 10 % no grau 2 (Pobre) e apenas 5% nos graus 3 (Justo) e 4 (Bom).

6.2.3. ATRASO MÉDIO FIM-A-FIM

O atraso médio fim-a-fim mensura a média do atraso percebida pelos pacotes do fluxo de vídeo, considerando-se as simulações efetuadas para cada cenário.

Os resultados são apresentados em termos das funções de densidade de probabilidade (PDF) e probabilidade cumulativa (CDF). A função PDF apresenta a probabilidade de um quadro ser recebido com determinado atraso no instante de tempo considerado. A função CDF apresenta a probabilidade de um quadro ser recebido com atraso igual ou inferior ao instante de tempo em questão.

6.2.3.1. PDF

A Figura 29 apresenta a curva densidade de probabilidade para os três cenários sob avaliação.

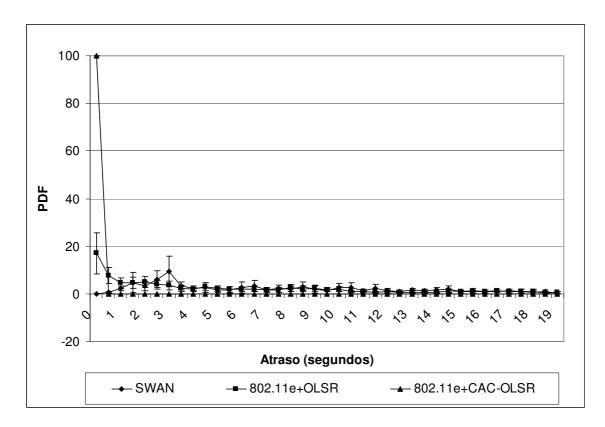


Figura 29: Atraso (PDF)

Conforme pode-se constatar, no cenário CAC-OLSR há uma probabilidade de 100% dos pacotes serem recebidos com atraso de até 0,5 s. Esta mesma probabilidade ficou próxima de 18% para o OLSR original. No caso do mecanismo SWAN, a probabilidade de um quadro ser recebido com atraso inferior a 0,5 s é quase nula.

6.2.3.2. CDF

A Figura 30 apresenta a métrica atraso média fim-a-fim em termos da função de probabilidade cumulativa.

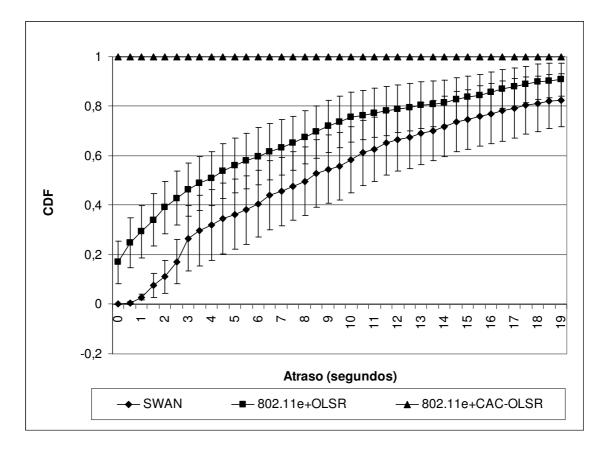


Figura 30: Atraso (CDF)

Coerentemente com a curva anterior, 100% dos pacotes foram recebidos com atraso inferior a 0,5 s no cenário CAC-OSLR. Por outro lado, nota-se que há uma probabilidade de aproximadamente 10% dos quadros serem recebidos com atraso superior a 20 s para o OLSR original. No caso do mecanismo SWAN esta mesma probabilidade chega próxima a 20%.

6.2.4. PERDA MÉDIA DE PACOTES

A perda média de quadros representa a quantidade média de quadros perdidos, considerando-se as simulações efetuadas em cada cenário. A Figura 31 apresenta os resultados obtidos.

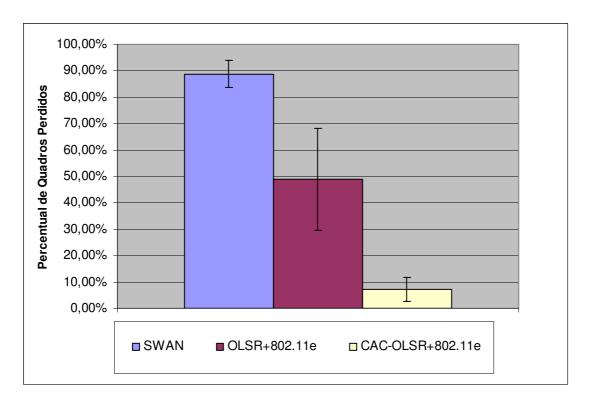


Figura 31: Perda média de pacotes

Conforme pode-se observar, observou-se uma perda média de aproximadamente 7,5% dos quadros no cenário CAC-OSLR. Este mesma métrica chegou a níveis próximos a 48% e 88% para OLSR original e SWAN respectivamente.

A perda de quadros consideravelmente superior no mecanismo SWAN provavelmente está relacionada ao fato de o mecanismo não levar em consideração a diferenciação de tráfego provida pelas categorias de acesso da emenda IEEE 802.11e. Deve-se lembrar de que a cada 10 s de simulação escolhe-se aleatoriamente um nó da rede e nele são injetados 4 novos fluxos, um de cada categoria. No caso dos cenários CAC-OLSR e OLSR original, as categorias de acesso da emenda 802.11e tratam as colisões internas priorizando os fluxos de voz e vídeo em detrimento dos demais. No caso de SWAN, as citadas categorias não são consideradas, ou seja, tráfego com e sem QoS concorrem igualmente pelo acesso ao meio, fato que aumenta a probabilidade de perda de pacotes dos tráfegos com QoS, como o vídeo analisado no Evalvid.

6.3. ANÁLISE CONSOLIDADA DOS RESULTADOS

Com relação ao primeiro teste, análise de métricas de QoS, pode-se constatar que em todos os casos a proposta CAC-OSLR foi superior às demais.

Para atraso média fim-a-fim, CAC-OLSR manteve a métrica abaixo de 150 ms para voz e 500 ms para vídeo, enquanto SWAN e OLSR original obtiveram resultados acima de 10 s.

Com respeito à vazão média agregada, os resultados obtidos no cenário CAC-OLSR foram melhores do que SWAN e OLSR original, tanto para fluxos com QoS quanto para sem QoS. Este fato mostra que o mecanismo de controle de admissão implementado não diminuiu a utilização da rede em termos de quantidade de pacotes transmitidos e recebidos.

Adicionalmente, CAC-OLSR foi o único cenário capaz de preservar a vazão média por fluxo para voz e vídeo, mantendo esta métrica próxima aos valores demandados, isto é, 64 Kbps e 1024 Kbps para voz e vídeo respectivamente. Para os demais cenários, esta mesma métrica findou as simulações com valores inferiores a 20 Kbps para voz e 100 Kbps para vídeo.

Observou-se também que o mecanismo de reação a violações de QoS foi capaz de recuperar os requisitos de QoS dos fluxos vigentes mesmo diante de uma brusca variação nas condições de vizinhança dos nós.

O overhead introduzido pelas novas mensagens do protocolo CAC-OLSR, CREQ e CREP, representaram uma porcentagem inferior a 6% em termos de quantidade de pacotes de controle enviados. Além disso, o atraso para recepção do primeiro pacote de voz e vídeo com o mecanismo prospoto foi em média de 32 ms para voz e 56 ms para vídeo. Tais resultados não impactaram a utilização da rede, tampouco as aplicações em questão.

Finalmente, pôde-se constatar que os resultados obtidos com o cenário CAC-OLSR configurado com parâmetros sugeridos pelo padrão IEEE 802.11 e por [Gerk, 2010] foram próximos para todas as métricas avaliadas. Tal comportamente provavelmente está associado ao fato de o mecanismo de controle de admissão ter prevenido a rede de atingir um estado de congestionamento, situação na qual a configuração sugerida por [Gerk, 2010] apresenta melhores resultados, especialmente para voz [Gerk, 2010].

No segundo teste proposto, transmissão de vídeo real, novamente CAC-OSLR superou os demais cenários.

A avaliação do nível de PSNR dos quadros recebidos mostrou que CAC-OSLR obteve níveis acima de 40 dB, enquanto OLSR original apresentou boa parte dos quadros entre 20 e 30 dB. Já SWAN apresentou a maior parte dos quadros na faixa de 15 a 20 dB.

O resultado da relação PSNR automaticamente refletiu-se na escala MOS, em que CAC-OLSR recebeu grau 5 em quase 100% dos quadros. Nos demais cenários notou-se claramente a degradação da qualidade. OSLR original recebeu aproximadamente 26% dos quadros com grau ruim ou pobre, enquanto que no cenário SWAN, este mesmo percentual chegou a aproximadamente 96%.

Finalmente os resultados apresentados para atraso e perda corroboraram as constatações de qualidade percebida (MOS). A curva de probabilidade cumulativa (CDF) mostrou que 100% dos quadros foram recebidos com atraso inferior a 0,5 s para CAC-OSLR, enquanto que nos demais cenários um percentual próximo de 10% para OLSR original e 20% para SWAN não haviam sido recebidos após 20 s. Para a perda de quadros, CAC-OLSR apresentou uma média inferior a 10%, enquanto que OLSR original e SWAN atingiram aproximadamente 50% e 90% respectivamente.

Como limitações do CAC-OLSR, pode-se citar o aumento de processamento dos nós em virtude do monitoramento da detecção de portadora, além do aumento de consumo de memória nos nós injetores de fluxos na rede, dada a necessidade de armazenamento dos fluxos em admissão, admitidos e rejeitados. Embora dentro dos limites toleráveis para as aplicações de voz e vídeo, o retardo inserido em função da troca de mensagens CREQ e CREP também se configura como um contraponto do CAC-OLSR.

7. CONCLUSÃO

O presente trabalho apresentou um mecanismo de controle de admissão para redes em malha sem fio compostas de múltiplos saltos que operam no modo contenção do padrão IEEE 802.11 e utilizam o protocolo de roteamento OLSR. Esta proposta teve por principal finalidade garantir que novos fluxos com demanda de QoS sejam admitidos na rede apenas se seus requisitos puderem ser satisfeitos sem, por outro lado, violar os requisitos daqueles anteriormente admitidos.

A proposta apresentada, denominada CAC-OSLR, baseou-se na comparação entre a ocupação temporal do meio exigida por um novo fluxo com a disponibilidade do canal. O trabalho considerou as categorias de acesso propostas na emenda IEEE 802.11e e apresentou uma abordagem para tratamento das interferências inter e intrafluxo.

CAC-OLSR também apresentou um mecanismo de recuperação a violações de QoS, bem como implementou reservas de recursos em termos de tempo de canal para as categorias de voz e vídeo. Quando não utilizados, estes recursos podem ser empregados por outras categorias.

A proposta foi avaliada com auxílio das ferramentas NS-2 e Evalvid, sendo todos os resultados comparados com o protocolo OSLR original e com o mecanismo de controle de admissão SWAN.

Na ferramenta NS-2, pôde-se constatar que o mecanismo CAC-OSLR foi capaz de manter o atraso fim-a-fim dos fluxos de voz e vídeo abaixo de 150 ms e 0,5 s, respectivamente. Esta mesma métrica superou os 10 s, tanto para voz quanto para vídeo, com o uso do OLSR original e SWAN, valor considerado não satisfatório para tais aplicações.

Ainda através das simulações no NS-2, pôde-se concluir que apenas o mecanismo CAC-OLSR manteve a vazão média por fluxo demandada pelos tráfegos de voz e vídeo, isto é, 64 Kbps e 1024 Kbps. Esta mesma métrica apresentou queda considerável com o OLSR original e SWAN. Além de manter a vazão média por fluxo, CAC-OLSR apresentou uma melhor vazão agregada para tráfegos com e sem QoS, demonstrando assim que o controle de admissão implementado não provocou menor utilização dos recursos de canal da rede.

Conclui-se também que o mecanismo de reação a violações de QoS proporcionou plena recuperação das métricas atraso e vazão dos fluxos, o que torna a proposta menos suscetível a efeitos de desvanecimento do canal, mudanças na vizinhança dos nós, mobilidade, dentre outros.

Os resultados obtidos na ferramenta NS-2 foram corroborados pelos testes realizados no Evalvid. As medições de PSNR e MOS mostraram que apenas CAC-OSLR obteve êxito em preservar a qualidade do vídeo recebido próxima daquele transmitido. Por outro lado, com o uso do OLSR original houve perceptiva degradação da qualidade, ainda que a maioria dos quadros recebidos tenha recebido grau 3 (justo) ou superior. Já com o uso do mecanismo SWAN, pode-se concluir que o vídeo foi recebido com qualidade insatisfatória, uma vez que mais de 85% dos quadros recebeu classificação equivalente a 1 (ruim).

Em resumo, conclui-se que os resultados proporcionados pelo CAC-OLSR foram plenamente satisfatórios frente àqueles produzidos nas mesmas condições pela proposta SWAN e pelo protocolo OLSR original. Cabe ainda observar que o overhead introduzido pelo mecanismo de admissão proposto foi inferior a 6% com relação à quantidade de pacotes adicionais de controle enviados, percentual considerado baixo frente ao benefício gerado. Além disso, o atraso para recepção do primeiro pacote de voz e vídeo não comprometeram os requisitos demandados pelas aplicações.

7.1. CONTRIBUIÇÕES

Conforme apresentado por este trabalho, a provisão de QoS em redes em malha sem fio passa por pelo menos dois pontos cruciais, a saber: diferenciação de tráfego e controle de admissão.

No que tange às redes sem fio que operam segundo o padrão IEEE 802.11, as categorias de acesso propostas pela emenda 802.11e proporcionaram a diferenciação de tráfego em quatro tipos distintos: voz, vídeo, melhor esforço e *background*.

Com relação ao mecanismo de controle de admissão, o padrão citado apenas orienta como este funcionaria em uma rede infraestruturada, porém não define sua implementação, tampouco aborda seu funcionamento em redes em malha.

Diante deste cenário, diversas iniciativa acadêmicas propuseram mecanismos de controle de admissão para redes sem fio, a maior parte porém voltada para redes infraestruturadas. O presente trabalho vem pois juntar-se ao restrito grupo de mecanismos de controle de admissão com foco em redes em malha sem fio.

No contexto deste grupo a principal contribuição da proposta apresentada foi dotar o protocolo pró-ativo OLSR, comumente empregado em redes em malha sem fio [Passos et al., 2006; Sing et al., 2008], de um mecanismo de controle de admissão que levasse em consideração as seguintes características:

- possibilidade de múltiplos saltos;
- diferenciação de tráfego na camada MAC (IEEE 802.11e);
- interferências inter e intra-fluxo;
- violação de QoS;
- reserva de recursos de canal por categoria de acesso da camada MAC.

Até onde foi possível a este trabalho investigar, nenhuma outra proposta proporciona ao protocolo OLSR um mecanismos onde todos os pontos acima mencionados são considerados.

Particularmente, a reserva de recursos de canal por categoria de acesso na camada MAC não foi encontrada em qualquer outra proposta, independente do protocolo de roteamento adotado, fato que a torna inovadora.

Além da proposta do mecanismo CAC-OLSR, sua implementação no simulador NS-2 pode ser apontada como outra contribuição desta dissertação.

7.2. TRABALHOS FUTUROS

A proposta apresentada por esta dissertação e avaliada por meio de simulações em software, ao mesmo tempo em que apresentou excelentes resultados, abre espaço para trabalhos futuros que venham a aperfeiçoá-la.

A primeira sugestão de trabalho futuro diz respeito à estimativa de ocupação de canal por um novo fluxo. Com relação à estimativa do tempo gasto com contenção (*backoff*), esta dissertação considerou um valor equivalente à metade da janela de contenção mínima multiplicado pela duração de um slot de tempo (Equação 5). Contudo, entende-se que é possível um

tratamento estatístico mais profundo que leve em consideração a variação do parâmetro janela de contenção entre seus valores mínimo e máximo (CWmin e CWmax).

Também há espaço para melhorias no tratamento proposto para as interferências inter e intrafluxo. Ambas consideram um raio de abrangência de nós vizinhos com até dois saltos de distância. Este quantitativo, na prática, pode ser maior ou menor a depender da potência e modulação utilizadas, bem como da própria distância entre os nós.

Outra sugestão de trabalho refere-se à avaliação da proposta com relação a sua escalabilidade. Neste sentido, sugere-se avaliá-la em cenários de redes em malha sem fio com maior número de nós e fluxos.

O desempenho do CAC-OLSR em ambientes cujos nós possuem mobilidade é outro ponto a ser investigado.

Outrossim, identifica-se a necessidade de avaliar o CAC-OSLR quanto ao processamento adicional que demandará em cada nó, bem como no que se refere ao impacto no espaçamento em memória por conta do armazenamento da tabela de fluxos admitidos.

Finalmente, recomenda-se a avaliação do CAC-OLSR em cenários reais. Para tanto, poder-se-ia implementá-lo no firmware openwrt de roteadores sem fio de uma rede em malha.

8. REFERÊNCIAS BIBLIOGRÁFICAS

[Abdrabou e Zhuang, 2008] Abdrabou, A., Zhuang, W. Statistical Call Admission Control for IEEE 802.11 Multi-hop Wireless Ad hoc Network. Globecom. 2008.

[Ahn et al., 2002a] Ahn, G.-S., Campbell, A. T., Veres, A., Sun, L.-H. Supporting Service Differentiation for Real-Time e Best-Effort Traffic in Stateless Wireless Ad Hoc Networks. IEEE Transactions on Mobile Computing, vol. 1, no. 3, pp 192-207. Julho de 2002.

[Ahn et al., 2002b] Ahn, G.-S., Campbell, A. T., Veres, A., Sun, L.-H. SWAN - Internet draft, IETF, (draft-ahn-swan-manet-00.txt). Fevereiro de 2003.

[Badis e Agha, 2006] Badis, H., Agha, K. A. CEQMM: A Complete and Efficient Quality of service Model for MANETs. International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks. 2006.

[Badis e Munaretto, 2004] Badis H., Agha, K. A. e Munaretto A. Quality of Service for Ad hoc Optimized Link State Routing Protocol (QOLSR). IETF-61 Meeting: Internet Engineering Task Force, Washington, DC. Novembro 2004.

[Cerveira e Costa, 2006] Cerveira, C. R., Costa, L. H. M. K., A Time-based Admission Control Mechanism for IEEE 802.11 Ad Hoc Networks. 8th Mobile Wireless Communications Network, pp 217-228, Santiago, Chile. 2006.

[Chakeres e Belding-Royer, 2007] Chakeres, I., Belding-Royer, E. Perceptive Admission Control for Wireless Network Quality of Service. Ad Hoc Networks, vol 5, pp 1129-1148. 2007.

[Cheng et al., 2008] Cheng, X., Mohapatra, P., Lee, S., Banerjee, S. MARIA: Interface-aware Admission Control and QoS Routing in Wireless Mesh Networks. International Conference on Communications. 2008.

[Clausen e Jacquet, 2003] Clausen, T. e P. Jacquet. Optimized Link State Routing Protocol (OLSR). RFC Experimental 3626, Internet Engineering Task Force. Outubro, 2003.

[Evalvid, 2011] Evalvid - A Video Quality Evaluation Tool-set home page http://www.tkn.tu-berlin.de/research/evalvid/, acessado em Agosto/2011.

[Gerk, 2010] L. F. Gerk. Provisão de QoS em Redes em Malha sem Fio Baseada no Padrão IEEE 802.11e. Dissertação de Mestrado – Universidade Federal Fluminense – UFF, Niterói, RJ, Agosto, 2010.

[Hanzo e Tafazolli, 2009] Hanzo II, L., Tafazolli, R. Admission Control Schemes for 802.11-Based Multi-Hop Mobile Ad hoc Networks: A Survey. IEEE Communications Surveys & Tutorials vol 11, No. 4. 2009.

[IEEE 802.1D, 2004] IEEE Standard for Local and metropolitan area networks – Media Access Control (MAC) Bridges. Junho, 2004.

[IEEE P802.11, 2007] IEEE P802.11-2007 - IEEE Standard for Information Technology — Telecommunications and Information Exchange between Systems — Local and Metropolitan Area Networks — Specific Requirements — Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Junho, 2007.

[IEEE P802.11e, 2005] IEEE P802.11e-2005 - IEEE Standard for Information Technology — Telecommunications and Information Exchange between Systems — Local and Metropolitan Area Networks — Specific Requirements — Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications — Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements. Novembro, 2005.

[Johnson et al., 2007] Johnson, D., Hu, Y, Maltz, D. The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4. RFC Experimental 4728, Internet Engineering Task Force. Outubro, 2007.

[Kettaf et al., 2006a] Kettaf, N., Abouaussa, H., Vuduong, T., Lorenz, P. A Cross layer Admission Control On-demand Routing Protocol for QoS Applications. International Journal of Computer Science and Network Security, vol 6, Nr 9B. 2006.

[Kettaf et al., 2006b] N. Kettaf Abouaussa, H. Vuduong, T., Lorenz, P. A. "Admission Control enables Routing Protocol -ACOR-" Internet draft, IETF, (draft-kettafmanet-acor-00.txt). Julho de 2006.

[Kurose e Ross, 2004] Kurose, J. F., Ross, K. W. Computer Networking: A Top-Down Approach Featuring the Internet. 3 ed. Addison Wesley. 2004.

[Lindgren e Belding-Royer, 2004] Lindgren, A., Belding-Royer, E. Multipath Admission Control for Mobile Ad hoc Networks. ACM SIGMOBILE Mobile Computing and Communications, vol 8, pp 68-71. 2004.

[Luo et al., 2006] Luo, L., Gruteser, M., Liu, H., Raychaudhuri, D., Huang, K., Chen, S. A QoS Routing and Admission Control Scheme for 802.11 Ad Hoc Networks. DIWANS'. 2006.

[Mineiro e Saade, 2011] Mineiro, E. P. S., Saade, D. C. M., "Controle de Admissão para Provisão de QoS em Redes em Malha sem Fio" International Information and Telecommunications Technologies Symposium. Florianópolis, 2011.

[Nguyen e Minet, 2005] Nguyen, D., Minet, P. Interference-Aware QoS OLSR for Mobile Ad-hoc Network Routing. Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and First ACIS International Workshop on Self-Assembling Wireless Networks (SNPD/SAWN'05). 2005.

[NS2, 2010] NS2 – http://www.isi.edu/nsnam/ns/, acessado em Dezembro de 2010.

[Passos et al., 2006] Passos, D., Teixeira, D. V., Saade, D. C. M, Magalhães, L. C. S, Albuquerque, C. V. N., "Mesh Network Performance Measurements". International Information and Telecommunications Technologies Symposium. 2006.

[Perkins e Belding-Royer, 2003] Perkins, C., Belding-Royer, E. Ad hoc On-Demand Distance Vector (AODV) Routing – RFC 3561, http://www.ietf.org/rfc/rfc3561.txt. 2003.

[Prabhakaran, 2010] Prabhakaran, M. On Supporting reliable QoS in multi-hop multi-rate mobile ad hoc networks. Wireless Networks, vol 10, pp 813-827. 2010.

[Ramos et al., 2005] Ramos, N., D. Panigrahi e S. Dey. Quality of service provisioning in 802.11e networks: challenges, approaches, and future directions. Network, IEEE. 2005.

[Ros, 2009] Ros, F. J. OLSR protocol for the ns2 network simulator. University of Murcia, Murcia, Espanha, 2009.

[Sing et al., 2008] Sing, S., Sran, S. S., Chand, T., "Performance Comparision of AODV, OLSR and OFLSR em Wireless Mesh Networks", National Conference of Challenges & Opportunities in Information Technology (COIT 2008). Março de 2008.

[Wiethölter e Hoene, 2009] S. Wiethölter e C. Hoene. An IEEE 802.11e EDCA and CFB simulation model for NS2. TKN – Telecommunication Networks Group, Technical University of Berlin, Germany, http://www.tkn.tuberlin.de/research/802.11e_ns2/, acessado em Janeiro/2011.

[Yang e Kravets, 2005] Yang, Y., Kravets, R. Contention-aware Admission Control Protocol for Ad-hoc Networks. IEEE Transactions on Mobile Computing, vol 4, pp 363-377. 2005.

ANEXO A

A.1. SCRIPTS TCL PARA O TESTE "ANÁLISE DE MÉTRICAS DE QOS"

A.1.1. CENÁRIO CAC-OLSR

} elseif {\$opti == "-cwmax0"} {

set val(cwmax0) [lindex \$argv [incr i]]

```
set AKAROA 0
Mac/802 11e set dataRate 54Mb
                       Channel/WirelessChannel
set val(chan)
                       Propagation/TwoRayGround
set val(prop)
set val(netif)
                       Phy/WirelessPhy
set val(mac)
                       Mac/802_11e
                       Queue/DTail/PriQ
set val(ifq)
set val(II)
set val(ant)
                       Antenna/OmniAntenna
set val(x)
                       500
                                        :# coordenada X da topografia
set val(y)
                       500
                                        ;# coordenada Y da topografia
set val(ifglen)
                       1000
                                        ;# max pacotes em ifq
set val(adhocRouting) OLSR
                                        ;# protocolo de roteamento
set val(stop)
                       480.0
                                        ;# tempo de simulacao
set val(txPower)
                       0.0
                                        ;# taxa de transmissao de dados em W
set val(rxPower)
                                        ;# taxa de repecpcao de dados em W
                       0.0
set val(idle)
                       0.0
                                        ;# taxa de escuta ociosa em W
# Programa principal
# Inicia as Variaveis globais
set run 1
for {set i 0} {$i < $argc } {incr i} {
set opti [lindex $argv $i]
if {$opti == "-cenario"} {
       set val(cenario) [lindex $argv [incr i]]
} elseif {$opti == "-seed"} {
       set val(seed) [lindex $argv [incr i]]
} elseif {$opti == "-run"} {
       set run [lindex $argv [incr i]]
} elseif {$opti == "-n"} {
       set val(nn) [lindex $argv [incr i]]
} elseif {$opti == "-trafego"} {
       set val(traf) [lindex $argv [incr i]]
} elseif {$opti == "-cfb"} {
       set val(cfb) [lindex $argv [incr i]]
} elseif {$opti == "-cfb_adapt"} {
       set val(cfb_adapt) [lindex $argv [incr i]]
} elseif {$opti == "-pf0"} {
       set val(pf0) [lindex $argv [incr i]]
} elseif {$opti == "-aifs0"} {
       set val(aifs0) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin0"} {
       set val(cwmin0) [lindex $argv [incr i]]
```

```
} elseif {$opti == "-txopLimit0"} {
       set val(txop0) [lindex $argv [incr i]]
} elseif {$opti == "-pf1"} {
       set val(pf1) [lindex $argv [incr i]]
} elseif {$opti == "-aifs1"} {
       set val(aifs1) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin1"} {
       set val(cwmin1) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax1"} {
       set val(cwmax1) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit1"} {
       set val(txop1) [lindex $argv [incr i]]
} elseif {$opti == "-pf2"} {
       set val(pf2) [lindex $argv [incr i]]
} elseif {$opti == "-aifs2"} {
       set val(aifs2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin2"} {
       set val(cwmin2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax2"} {
       set val(cwmax2) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit2"} {
       set val(txop2) [lindex $argv [incr i]]
} elseif {$opti == "-pf3"} {
       set val(pf3) [lindex $argv [incr i]]
} elseif {$opti == "-aifs3"} {
       set val(aifs3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin3"} {
       set val(cwmin3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax3"} {
       set val(cwmax3) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit3"} {
       set val(txop3) [lindex $argv [incr i]]
} else {
       puts "Opcao invalida: \"$opti\""
       exit 1
remove-all-packet-headers
add-packet-header IP ARP OLSR LL Mac
set ns_ [new Simulator]
                                         ;# Cria instancia do simulador
$ns_ use-scheduler Heap
                                         ;# Seta o objeto topografia
set topo [new Topography]
set trace_file "cenario_mineiro-s$val(seed).tr"
set tracefd [open $trace_file w]
                                         ;# Cria o objeto trace para ns
$ns_ trace-all $tracefd
$topo load_flatgrid $val(x) $val(y)
                                                  ;# Define a topologia
set god_ [create-god $val(nn)]
                                         ;# Cria God
set chan_ [new $val(chan)]
# Define como os nos devem ser criados
# Configurações globais dos nos
$ns_ node-config -adhocRouting $val(adhocRouting) \
       -IIType $val(II) \
       -macType $val(mac) \
       -ifqType $val(ifq) \
```

```
-ifqLen $val(ifqlen) \
       -antType $val(ant) \
       -propType $val(prop) \
       -phyType $val(netif) \
       -channel $chan_\
       -topolnstance $topo \
       -rxPower $val(rxPower) \
       -txPower $val(txPower) \
       -idlePower $val(idle) \
       -agentTrace ON \
       -routerTrace ON \
       -macTrace ON \
       -movementTrace OFF \
if \{\$AKAROA == 1\}
set ak [new Akaroa]
$ak AkDeclareParameters 8
# Cria um especifico numero de nos [$val(nn)] e os associa aos canais
for {set i 0} {$i < $val(nn) } {incr i} {
       set node_($i) [$ns_ node]
       $node_($i) random-motion 0
                                                  ;# Desabilita movimentação aleatória
       set mac [$node_($i) getMac 0]
       set routing [$node_($i) set ragent_]
       $routing addMac $mac
}
# Define a posicao dos nos
source $val(cenario)
#Define padrao de trafego
source $val(traf)
       # Diz aos nos quando a simulação termina
       for \{ \text{set i } 0 \} \{ \text{si} < \text{sval}(nn) \} \{ \text{incr i} \} \{ \}
       $ns_ at $val(stop).0 "$node_($i) reset";
       $ns_ at $val(stop).0002 "puts \"NS EXITING...\"; $ns_ halt"
       puts $tracefd "M 0.0 nn $val(nn) x $val(x) y $val(y) rp $val(adhocRouting)"
       puts $tracefd "M 0.0 prop $val(prop) ant $val(ant)"
       puts "Starting Simulation $run..."
       $ns_ run
```

A.1.2. CENÁRIO OLSR ORIGINAL

set AKAROA 0

```
Mac/802_11e set dataRate_ 54Mb
```

```
Channel/WirelessChannel
set val(chan)
set val(prop)
                        Propagation/TwoRayGround
set val(netif)
                        Phy/WirelessPhy
set val(mac)
                        Mac/802 11e
                        Queue/DTail/PriQ
set val(ifq)
set val(II)
set val(ant)
                        Antenna/OmniAntenna
set val(x)
                        500
                                          ;# coordenada X da topografia
                        500
                                          ;# coordenada Y da topografia
set val(y)
set val(ifqlen)
                        1000
                                          ;# max pacotes em ifq
set val(adhocRouting) OLSR
                                          ;# protocolo de roteamento
                                          ;# tempo de simulacao
set val(stop)
                        480.0
set val(txPower)
                        0.0
                                          ;# taxa de transmissao de dados em W
set val(rxPower)
                        0.0
                                          ;# taxa de repecpcao de dados em W
                                          :# taxa de escuta ociosa em W
set val(idle)
                        0.0
# Programa principal
# Inicia as Variaveis globais
set run 1
for {set i 0} {$i < $argc } {incr i} {
set opti [lindex $argv $i]
if {$opti == "-cenario"} {
       set val(cenario) [lindex $argv [incr i]]
} elseif {$opti == "-seed"} {
       set val(seed) [lindex $argv [incr i]]
} elseif {$opti == "-run"} {
       set run [lindex $argv [incr i]]
} elseif {$opti == "-n"} {
       set val(nn) [lindex $argv [incr i]]
} elseif {$opti == "-trafego"} {
       set val(traf) [lindex $argv [incr i]]
} elseif {$opti == "-cfb"} {
       set val(cfb) [lindex $argv [incr i]]
} elseif {$opti == "-cfb_adapt"} {
       set val(cfb_adapt) [lindex $argv [incr i]]
} elseif {$opti == "-pf0"} {
       set val(pf0) [lindex $argv [incr i]]
} elseif {$opti == "-aifs0"} {
       set val(aifs0) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin0"} {
       set val(cwmin0) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax0"} {
       set val(cwmax0) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit0"} {
       set val(txop0) [lindex $argv [incr i]]
} elseif {$opti == "-pf1"} {
     set val(pf1) [lindex $argv [incr i]]
} elseif {$opti == "-aifs1"} {
       set val(aifs1) [lindex $argv [incr i]]
```

```
} elseif {$opti == "-cwmin1"} {
       set val(cwmin1) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax1"} {
       set val(cwmax1) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit1"} {
       set val(txop1) [lindex $argv [incr i]]
} elseif {$opti == "-pf2"} {
       set val(pf2) [lindex $argv [incr i]]
} elseif {$opti == "-aifs2"} {
       set val(aifs2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin2"} {
       set val(cwmin2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax2"} {
       set val(cwmax2) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit2"} {
       set val(txop2) [lindex $argv [incr i]]
} elseif {$opti == "-pf3"} {
       set val(pf3) [lindex $argv [incr i]]
} elseif {$opti == "-aifs3"} {
       set val(aifs3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin3"} {
       set val(cwmin3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax3"} {
       set val(cwmax3) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit3"} {
       set val(txop3) [lindex $argv [incr i]]
} else {
       puts "Opcao invalida: \"$opti\""
       exit 1
remove-all-packet-headers
add-packet-header IP ARP OLSR LL Mac
set ns [new Simulator]
                                         ;# Cria instancia do simulador
$ns use-scheduler Heap
set topo [new Topography]
                                         ;# Seta o objeto topografia
set trace_file "cenario_mineiro_olsr-s$val(seed).tr"
set tracefd [open $trace_file w]
                                         ;# Cria o objeto trace para ns
$ns_ trace-all $tracefd
$topo load_flatgrid $val(x) $val(y)
                                         ;# Define a topologia
set god_ [create-god $val(nn)] ;# Cria God
set chan_ [new $val(chan)]
# Define como os nos devem ser criados
# Configurações globais dos nos
$ns_ node-config -adhocRouting $val(adhocRouting) \
       -IIType $val(II) \
       -macType $val(mac) \
       -ifqType $val(ifq) \
       -ifqLen $val(ifqlen) \
       -antType $val(ant) \
       -propType $val(prop) \
       -phyType $val(netif) \
       -channel $chan \
       -topoInstance $topo \
```

```
-rxPower $val(rxPower) \
      -txPower $val(txPower) \
      -idlePower $val(idle) \
      -agentTrace ON \
      -routerTrace ON \
      -macTrace ON \
      -movementTrace OFF \
if \{\$AKAROA == 1\} {
set ak [new Akaroa]
$ak AkDeclareParameters 8
}
#
# Cria um especifico numero de nos [$val(nn)] e os associa aos canais
for {set i 0} {$i < $val(nn) } {incr i} {
      set node ($i) [$ns node]
      $node ($i) random-motion 0
                                              ;# Desabilita movimentação aleatória
}
# Define a posicao dos nos
source $val(cenario)
#Define padrao de trafego
source $val(traf)
# Diz aos nos quando a simulacao termina
for {set i 0} {$i < $val(nn) } {incr i} {
$ns_ at $val(stop).0 "$node_($i) reset";
       $ns_ at $val(stop).0002 "puts \"NS EXITING...\"; $ns_ halt"
       puts $tracefd "M 0.0 nn $val(nn) x $val(x) y $val(y) rp $val(adhocRouting)"
       puts $tracefd "M 0.0 prop $val(prop) ant $val(ant)"
       puts "Starting Simulation $run..."
       $ns_ run
```

A.1.3. CENÁRIO SWAN

```
# Opcoes
Mac/802_11 set dataRate_ 54Mb
set opt(chan)
                     Channel/WirelessChannel
set opt(prop)
                     Propagation/TwoRayGround
set opt(netif)
                     Phy/WirelessPhy
                     Mac/802 11
set opt(mac)
                     Queue/DropTail/PriQueue
set opt(ifq)
set opt(II)
                     LL
set opt(ant)
                     Antenna/OmniAntenna
set opt(x)
                     500
                                    ;# coordenada X da topografia
                     500
                                    ;# coordenada Y da topografia
set opt(y)
                                   ;# marcador de progresso
set opt(progress)
                     1000
                                    ;# max pacotes em ifq
set opt(ifglen)
                                    ;# número de nós
set opt(nn)
                     10
set opt(stop)
                     480.0
                                    ;# tempo de simulação
set val(stop)
                     480.0
                     OLSR
set opt(rp)
                                    ;# protocolo de roteamento
"ON"
                                    ;# contralador de taxa ON/OFF
set opt(swan_rc)
                                    ;# controlador de admissão ON/OFF
set opt(swan ac)
                    "ON"
                 "result/test"
                                    ;# diretório destino de resultados
set opt(dir)
set opt(band)
                  "100Kb"
                                   ;# taxa inicial
set opt(ssthresh)
                   "1Mb"
                                    :# limiar do slow start
set opt(segment)
                    "50Kb"
                                    :# incremento de segmento
set opt(mdrate)
                   "50"
                                    :# decremento de taxa
                  "1.2"
set opt(gap)
                                    :# gap de controle
                    "100kb"
set opt(minband)
                                    :# taxa mínima
                  "1000Kb"
                                    ;# taxa para controle de admissão
set opt(acrate)
                  "2000Kb"
set opt(thrate)
                                    :# limiar de taxa
set run 1
for {set i 0} {$i < $argc } {incr i} {
set opti [lindex $argv $i]
if {$opti == "-cenario"} {
      set val(cenario) [lindex $argv [incr i]]
} elseif {$opti == "-seed"} {
      set val(seed) [lindex $argv [incr i]]
} elseif {$opti == "-run"} {
      set run [lindex $argv [incr i]]
} elseif {$opti == "-n"} {
      set val(nn) [lindex $arqv [incr i]]
} elseif {$opti == "-trafego"} {
      set val(traf) [lindex $argv [incr i]]
set AgentTrace
                            ON
set RouterTrace
                            ON
```

```
ON
set MacTrace
LL set mindelay_
                            50us
LL set delay
                            25us
LL set bandwidth_
                            0
LL set off prune
                            0
LL set off_CtrMcast_
                            0
Agent/Null set sport_
Agent/Null set dport_
                            0
Agent/CBR set sport
                            0
Agent/CBR set dport_ 0
Agent/TCPSink set sport
                            0
Agent/TCPSink set dport
Agent/TCP set sport
                            0
Agent/TCP set dport
Agent/TCP set packetSize_
                            1460
Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1
# Antenas omni-directional com ganho unitário e altura de 1,5m.
Antenna/OmniAntenna set X_0
Antenna/OmniAntenna set Y
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Gt 1.0
Antenna/OmniAntenna set Gr_ 1.0
Phy/WirelessPhy set bandwidth_ 54e6
Mac set bandwidth 54e6
#______
proc usage { argv0 } {
      puts "Usage: $argv0"
      puts "\tmandatory arguments:"
      puts "\t\t\[-x MAXX\] \[-y MAXY\]"
      puts "\toptional arguments:"
      puts "\t\t\[-cp conn pattern\] \[-sc scenario\] \[-nn nodes\]"
      puts "\t\t\[-seed seed\] \[-stop sec\] \[-tr tracefile\]\n"
proc getopt {argc argv} {
      global opt
      lappend optlist cp nn seed sc stop tr x y
      for {set i 0} {$i < $argc} {incr i} {
             set arg [lindex $argv $i]
             if {[string range $arg 0 0] != "-"} continue
             set name [string range $arg 1 end]
             set opt($name) [lindex $argv [expr $i+1]]
      }
proc makedir { pathname } {
  if {[file isdirectory $pathname]} {
      return $pathname
  } elseif {[file exists $pathname]} {
      error "Non-directory $pathname already exists."
  } else {
      makedir [file dirname $pathname]
      exec mkdir $pathname
      return $pathname
```

```
}
# Programa Principal
getopt $argc $argv
if \{ \text{sopt}(x) == 0 \mid | \text{sopt}(y) == 0 \} \{
      usage $argv0
      exit 1
}
if \{\text{sopt(seed)} > 0\} {
      puts "Seeding Random number generator with $opt(seed)\n"
      ns-random $opt(seed)
#
# Inicializa Variáveis Globais
remove-all-packet-headers
add-packet-header IP ARP OLSR LL Mac
                     [new Simulator]
set ns
             [new Topography]
set topo
set god_
             [new God]
makedir $opt(dir)
set trace_file "cenario_mineiro_swan-s$val(seed).tr"
set tracefd [open $trace_file w]
                                    ;# Cria o objeto trace para ns
$ns trace-all $tracefd
$topo load_flatgrid $opt(x) $opt(y)
create-god $opt(nn)
set chan_1_ [new $opt(chan)]
#Configurações do nó
$ns node-config -adhocRouting $opt(rp) \
              -IIType $opt(II) \
              -macType $opt(mac) \
              -ifqType $opt(ifq) \
              -ifqLen $opt(ifqlen) \
              -antType $opt(ant) \
              -propType $opt(prop) \
              -phyType $opt(netif) \
              -channel $chan_1_\
              -topoInstance $topo \
              -macTrace ON \
              -rxPower 0.3 \
              -txPower 0.6 \
for {set i 0} {$i < $opt(nn) } {incr i} {
  set node_($i) [$ns_ node]
  $node_($i) random-motion 0
                                            ;# desabilita movimentação aleatória
}
# Define a posicao dos nos
source $val(cenario)
```

```
#Define padrao de trafego
source $val(traf)
# Acionamento das funções shaper e monitor de utilização a cada segundo
for {set i 0} {$i <= $opt(stop)} {incr i 1} {
  $ns_ at $i "$node_($j) monitor 1"
  }
}
#
# Diz aos nós quando a simulação termina
for {set i 0} {$i < $opt(nn) } {incr i} {
  $ns_ at $opt(stop).000000001 "$node_($i) reset";
$ns at $opt(stop).00000001 "puts \"NS EXITING...\"; $ns halt"
for \{\text{set i 1}\}\ \{\text{si <= }\text{sopt(progress)}\}\ \{\text{incr i}\}\
  set t [expr $i * $opt(stop) / ($opt(progress) + 1)]
  $ns_ at $t "puts \"completed through $t secs...\""
}
puts $tracefd "G -t * -prop $opt(prop) -ant $opt(ant)"
puts "Starting Simulation..."
$ns_ run
```

A.1.4. CONFIGURAÇÃO DO TRÁFEGO

```
#Configuração da classe RNG (aleatoriedade)
set sourceRNG [new RNG]
set destRNG [new RNG]
$sourceRNG seed $val(seed)
$destRNG seed $val(seed)
#Seleção aleatória dos nós origem e destino
set source0 [new RandomVariable/Uniform]
$source0 set min 0
$source0_ set max_ 9
$source0_ use-rng $sourceRNG
set dest0 [new RandomVariable/Uniform]
$dest0_ set min_ 0
$dest0 set max 9
$dest0 use-rng $destRNG
for {set t 1} {$t < $val(stop)} {incr t 20} {
set s0_ [expr round([$source0_ value])]
set d0_ [expr round([$dest0_ value])]
while {$d0_ == $s0_} {
     set d0 [expr round([$dest0 value])]
#Perfil de Tráfego
# Trafego de prioridade 0 = Voz
set udp_($t) [new Agent/UDP]
$ns attach-agent $node ($s0 ) $udp ($t)
$udp_($t) set class_ 0
$udp_($t) set prio_ 0
set null ($t) [new Agent/Null]
$ns_ attach-agent $node_($d0_) $null ($t)
set cbr ($t) [new Application/Traffic/CBR]
$cbr ($t) set packetSize 160
$cbr_($t) set interval_ 20e-3
$cbr ($t) attach-agent $udp_($t)
$ns_ connect $udp_($t) $null_($t)
$ns_ at [expr $t + 30.0] "$cbr_($t) start"
# Trafego de prioridade 1 = Video
set udp_([expr $t + 1]) [new Agent/UDP]
$ns attach-agent $node ($s0 ) $udp ([expr $t + 1])
$udp ([expr $t + 1]) set class 1
\ udp ([\exp t + 1]) set prio 1
set null ([expr $t + 1]) [new Agent/Null]
$ns attach-agent $node ($d0 ) $null ([expr $t + 1])
set cbr ([expr $t + 1]) [new Application/Traffic/CBR]
$cbr_([expr $t + 1]) set packetSize 1280
$cbr_([expr $t + 1]) set interval_ 10e-3
$cbr_([expr $t + 1]) attach-agent $udp_([expr $t + 1])
$ns_ connect $udp_([expr $t + 1]) $null_([expr $t + 1])
$ns_ at [expr $t + 30.0] "$cbr_([expr $t + 1]) start"
#$ns_ at [expr $t + 45.0] "[$node_(0) agent 255] print_nbset"
#$ns at [expr $t + 45.0] "[$node (3) agent 255] print nbset"
```

```
# Trafego de prioridade 2
set tcp_($t) [new Agent/TCP]
$ns_ attach-agent $node_($s0_) $tcp_($t)
$tcp_($t) set class_ 2
$tcp_($t) set prio_ 2
set sink_($t) [new Agent/TCPSink]
$ns_ attach-agent $node_($d0_) $sink_($t)
set ftp_($t) [new Application/FTP]
$tcp_($t) set packetSize_ 1300
$ftp_($t) attach-agent $tcp_($t)
$ns_connect $tcp_($t) $sink_($t)
$ns_ at [expr $t + 30.0] "$ftp_($t) start"
# Trafego de prioridade 3
set tcp ([expr $t + 1]) [new Agent/TCP]
n_{so} = 10^{10}  $\text{tach-agent $node_($s0_) $\text{tcp_([expr $t + 1])}
t = 1) set class 3
$tcp_([expr $t + 1]) set prio_ 3
set sink_([expr $t + 1]) [new Agent/TCPSink]
$ns_ attach-agent $node_($d0_) $sink_([expr $t + 1])
set ftp_([expr $t + 1]) [new Application/FTP]
$tcp_([expr $t + 1]) set packetSize_ 1500
ftp_{(expr $t + 1]} attach-agent tcp_{(expr $t + 1]}
n_c = 1 sink_([expr t + 1]) sink_([expr t + 1])
$ns_ at [expr $t + 30.0] "$ftp_([expr $t + 1]) start"
```

A.1.5. REAÇÃO À VIOLAÇÃO DE QOS

\$defaultRNG seed 0

set AKAROA 0

Mac/802_11e set dataRate_ 54Mb

```
set val(chan)
                       Channel/WirelessChannel
set val(prop)
                       Propagation/TwoRayGround
set val(netif)
                       Phy/WirelessPhy
                       Mac/802 11e
set val(mac)
                       Queue/DTail/PriQ
set val(ifq)
set val(II)
                       LL
set val(ant)
                       Antenna/OmniAntenna
set val(x)
                       1000
                                       ;# coordenada X da topografia
                                       ;# coordenada Y da topografia
set val(y)
                       500
                                       ;# max packet in ifq
set val(ifglen)
                       1000
set val(adhocRouting) OLSR
set val(stop)
                       480.0
                                       ;# tempo de simulacao
set val(txPower)
                       0.0
                                       ;# taxa de transmissao de dados em W
                                       ;# taxa de repecpcao de dados em W
set val(rxPower)
                       0.0
set val(idle)
                       0.0
                                       ;# taxa de escuta ociosa em W
# Programa principal
# =========
# Inicia as Variaveis globais
set run 1
for {set i 0} {$i < $argc } {incr i} {
      set opti [lindex $argv $i]
if {$opti == "-cenario"} {
       set val(cenario) [lindex $argv [incr i]]
} elseif {$opti == "-seed"} {
      set val(seed) [lindex $argv [incr i]]
} elseif {$opti == "-run"} {
      set run [lindex $argv [incr i]]
} elseif {$opti == "-n"} {
       set val(nn) [lindex $argv [incr i]]
} elseif {$opti == "-trafego"} {
      set val(traf) [lindex $argv [incr i]]
} elseif {$opti == "-cfb"} {
       set val(cfb) [lindex $argv [incr i]]
} elseif {$opti == "-cfb adapt"} {
       set val(cfb adapt) [lindex $argv [incr i]]
} elseif {$opti == "-pf0"} {
       set val(pf0) [lindex $argv [incr i]]
} elseif {$opti == "-aifs0"} {
       set val(aifs0) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin0"} {
       set val(cwmin0) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax0"} {
       set val(cwmax0) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit0"} {
       set val(txop0) [lindex $argv [incr i]]
} elseif {$opti == "-pf1"} {
     set val(pf1) [lindex $argv [incr i]]
```

```
} elseif {$opti == "-aifs1"} {
       set val(aifs1) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin1"} {
       set val(cwmin1) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax1"} {
       set val(cwmax1) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit1"} {
       set val(txop1) [lindex $argv [incr i]]
} elseif {$opti == "-pf2"} {
       set val(pf2) [lindex $argv [incr i]]
} elseif {$opti == "-aifs2"} {
       set val(aifs2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin2"} {
       set val(cwmin2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax2"} {
       set val(cwmax2) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit2"} {
       set val(txop2) [lindex $argv [incr i]]
} elseif {$opti == "-pf3"} {
       set val(pf3) [lindex $argv [incr i]]
} elseif {$opti == "-aifs3"} {
       set val(aifs3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin3"} {
       set val(cwmin3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax3"} {
       set val(cwmax3) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit3"} {
       set val(txop3) [lindex $argv [incr i]]
} else {
       puts "Opcao invalida: \"$opti\""
       exit 1
remove-all-packet-headers
add-packet-header IP ARP OLSR LL Mac
#set EDCA $val(edca)
set ns_ [new Simulator]
                                                 ;# Cria instancia do simulador
$ns_ use-scheduler Heap
set topo [new Topography]
                                ;# Seta o objeto topografia
set trace_file "cenario_mineiro_qos-s$val(seed).tr"
set tracefd [open $trace_file w]
                                         ;# Cria o objeto trace para ns
$ns_ trace-all $tracefd
$topo load_flatgrid $val(x) $val(y)
                                         ;# Define a topologia
set god_ [create-god $val(nn)]
                                         ;# Cria God
set chan_ [new $val(chan)]
# Define como os nos devem ser criados
# Configurações globais dos nos
$ns node-config -adhocRouting $val(adhocRouting) \
       -IIType $val(II) \
       -macType $val(mac) \
       -ifqType $val(ifq) \
       -ifqLen $val(ifqlen) \
```

```
-antType $val(ant) \
       -propType $val(prop) \
       -phyType $val(netif) \
       -channel $chan_ \
      -topoInstance $topo \
      -rxPower $val(rxPower) \
      -txPower $val(txPower) \
      -idlePower $val(idle) \
      -agentTrace ON \
      -routerTrace ON \
      -macTrace ON \
      -movementTrace OFF \
if \{\$AKAROA == 1\} {
set ak [new Akaroa]
$ak AkDeclareParameters 8
# Cria um especifico numero de nos [$val(nn)] e os associa aos canais
for {set i 0} {$i < $val(nn) } {incr i} {
      set node_($i) [$ns_ node]
      $node_($i) random-motion 0
                                              ;# Desabilita movimentação aleatória
      set mac [$node_($i) getMac 0]
      set routing [$node_($i) set ragent_]
      $routing addMac $mac
}
# Define a posicao dos nos
$node_(0) set X_ 0.0
$node_(0) set Y_ 10.0
$node_(0) set Z_ 0.000000000000
$node_(1) set X_ 0.0
$node_(1) set Y_ 110.0
$node_(1) set Z_ 0.000000000000
$node_(2) set X_ 0.0
$node (2) set Y 210.0
$node_(2) set Z_ 0.000000000000
$node (3) set X 1000.0
$node (3) set Y 10.0
$node_(3) set Z_ 0.000000000000
$node_(4) set X_ 1000.0
$node_(4) set Y_ 110.0
$node_(4) set Z_ 0.0000000000000
$node_(5) set X_ 1000.0
$node_(5) set Y_ 210.0
$node_(5) set Z_ 0.000000000000
#Movimentação dos nós
$ns_ at 300.0 "$node_(3) setdest 20.0 10.0 100.0"
$ns_ at 300.0 "$node_(4) setdest 20.0 110.0 100.0"
$ns_ at 300.0 "$node_(5) setdest 20.0 210.0 100.0"
#Define padrao de trafego
source $val(traf)
# Diz aos nos quando a simulação termina
for {set i 0} {$i < $val(nn) } {incr i} {
$ns_ at $val(stop).0 "$node_($i) reset";
```

 $ns_at \val(stop).0002 "puts \"NS EXITING...\" ; <math display="inline">ns_halt "$ puts $tracefd "M 0.0 nn \val(nn) x \val(x) y \val(y) rp \val(adhocRouting)" puts \tracefd "M 0.0 prop \val(prop) ant \val(ant)" puts "Starting Simulation \run..." \ns_run$

A.2. SCRIPTS TCL PARA O TESTE "TRANSMISSÃO DE VÍDEO REAL"

A.2.1. CENÁRIO CAC-OLSR (EVALVID)

set AKAROA 0

} elseif {\$opti == "-txopLimit0"} {

```
Mac/802 11e set dataRate 54Mb
                      Channel/WirelessChannel
set val(chan)
set val(prop)
                      Propagation/TwoRayGround
                      Phy/WirelessPhy
set val(netif)
set val(mac)
                      Mac/802_11e
                      Queue/DTail/PriQ
set val(ifq)
set val(II)
                      LL
set val(ant)
                      Antenna/OmniAntenna
set val(x)
                      500
                                      ;# coordenada X da topografia
                      500
                                      ;# coordenada Y da topografia
set val(y)
set val(ifqlen)
                      1000
                                      ;# max pacotes em ifq
set val(adhocRouting) OLSR
                                      ;# protocolo de roteamento
                                      ;# tempo de simulacao
                      150.0
set val(stop)
set val(txPower)
                                      ;# taxa de transmissao de dados em W
                      0.0
set val(rxPower)
                                      :# taxa de repecpcao de dados em W
                      0.0
                                      :# taxa de escuta ociosa em W
set val(idle)
                      0.0
# Programa principal
# Inicia as Variaveis globais
set run 1
for {set i 0} {$i < $argc } {incr i} {
set opti [lindex $argv $i]
if {$opti == "-cenario"} {
      set val(cenario) [lindex $argv [incr i]]
} elseif {$opti == "-seed"} {
       set val(seed) [lindex $argv [incr i]]
} elseif {$opti == "-run"} {
       set run [lindex $argv [incr i]]
} elseif {$opti == "-n"} {
      set val(nn) [lindex $argv [incr i]]
} elseif {$opti == "-trafego"} {
       set val(traf) [lindex $argv [incr i]]
} elseif {$opti == "-cfb"} {
      set val(cfb) [lindex $argv [incr i]]
} elseif {$opti == "-cfb_adapt"} {
      set val(cfb_adapt) [lindex $argv [incr i]]
} elseif {$opti == "-pf0"} {
      set val(pf0) [lindex $argv [incr i]]
} elseif {$opti == "-aifs0"} {
      set val(aifs0) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin0"} {
      set val(cwmin0) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax0"} {
      set val(cwmax0) [lindex $argv [incr i]]
```

```
set val(txop0) [lindex $argv [incr i]]
} elseif {$opti == "-pf1"} {
       set val(pf1) [lindex $argv [incr i]]
} elseif {$opti == "-aifs1"} {
       set val(aifs1) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin1"} {
       set val(cwmin1) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax1"} {
       set val(cwmax1) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit1"} {
       set val(txop1) [lindex $argv [incr i]]
} elseif {$opti == "-pf2"} {
       set val(pf2) [lindex $argv [incr i]]
} elseif {$opti == "-aifs2"} {
       set val(aifs2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin2"} {
       set val(cwmin2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax2"} {
       set val(cwmax2) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit2"} {
       set val(txop2) [lindex $argv [incr i]]
} elseif {$opti == "-pf3"} {
       set val(pf3) [lindex $argv [incr i]]
} elseif {$opti == "-aifs3"} {
       set val(aifs3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin3"} {
       set val(cwmin3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax3"} {
       set val(cwmax3) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit3"} {
       set val(txop3) [lindex $argv [incr i]]
} else {
       puts "Opcao invalida: \"$opti\""
       exit 1
remove-all-packet-headers
add-packet-header IP ARP OLSR LL Mac
set ns_ [new Simulator]
                                         ;# Cria instancia do simulador
$ns_ use-scheduler Heap
set topo [new Topography]
                                         ;# Seta o objeto topografia
set trace_file "cenario_mineiro_olsr-s$val(seed).tr"
set tracefd [open $trace_file w]
                                         ;# Cria o objeto trace para ns
$ns_ trace-all $tracefd
set max_fragmented_size 1000
set packetSize 1028
$topo load_flatgrid $val(x) $val(y)
                                         ;# Define a topologia
set god [create-god $val(nn)];# Cria God
set chan_ [new $val(chan)]
# Define como os nos devem ser criados
# Configurações globais dos nos
$ns_ node-config -adhocRouting $val(adhocRouting) \
       -IIType $val(II) \
```

```
-macType $val(mac) \
      -ifqType $val(ifq) \
      -ifqLen $val(ifqlen) \
      -antType $val(ant) \
       -propType $val(prop) \
       -phyType $val(netif) \
       -channel $chan \
      -topolnstance $topo \
      -rxPower $val(rxPower) \
      -txPower $val(txPower) \
      -idlePower $val(idle) \
      -agentTrace ON \
      -routerTrace ON \
      -macTrace ON \
      -movementTrace OFF \
if \{\$AKAROA == 1\}
       set ak [new Akaroa]
      $ak AkDeclareParameters 8
}
# Cria um especifico numero de nos [$val(nn)] e os associa aos canais
for {set i 0} {$i < $val(nn) } {incr i} {
      set node_($i) [$ns_ node]
      $node_($i) random-motion 0
                                              ;# Desabilita movimentação aleatória
        set mac [$node_($i) getMac 0]
       set routing [$node ($i) set ragent ]
      $routing addMac $mac
}
#Define a posicao dos nos
source $val(cenario)
#Define padrao de trafego
source $val(traf)
#Vídeo a ser transmitido
set original file name news gcif
set trace file name "news $run.dat"
set original file id [open $original file name r]
set trace_file_id [open $trace_file_name w]
#set frame_count 0
#set last time 0
set pre_time 0
while {[eof $original_file_id] == 0} {
     gets $original file id current line
     scan $current_line "%d%s%d%d%f" no_frametype_length_tmp1_tmp2_
      set time [expr int(($tmp2_ - $pre_time)*1000000.0)]
     if {$frametype_ == "I"} {
          set type_v 1
          set prio p 1
     if {$frametype_ == "P"} {
          set type v 2
          set prio p 1
     if {$frametype == "B"} {
          set type v 3
```

```
set prio_p 1
     if {$frametype_ == "H"} {
          set type_v 1
          set prio_p 1
     puts $trace_file_id "$time $length_ $type_v $prio_p $max_fragmented_size"
     set pre_time $tmp2_
}
close $original_file_id
close $trace file id
set start_video_time 50.0
set end sim time [expr $tmp2 + $start video time]
puts "$end sim time"
set trace file [new Tracefile]
$trace_file filename $trace_file_name
set video1 [new Application/Traffic/myEvalvid]
$video1 attach-agent $udp1
$video1 attach-tracefile $trace_file
$ns_ at $start_video_time "$video1 start"
$ns_ at $end_sim_time "$video1 stop"
$ns_ at $val(stop).0 "$null1 closefile"
# Diz aos nos quando a simulacao termina
for {set i 0} {$i < $val(nn) } {incr i} {
  $ns_ at $val(stop).0 "$node_($i) reset";
$ns at $val(stop).0002 "puts \"NS EXITING...\"; $ns halt"
puts $tracefd "M 0.0 nn $val(nn) x $val(x) y $val(y) rp $val(adhocRouting)"
puts $tracefd "M 0.0 prop $val(prop) ant $val(ant)"
puts "Starting Simulation $run..."
$ns_run
```

A.2.2. CENÁRIO OLSR ORIGINAL (EVALVID)

set AKAROA 0

```
Mac/802 11e set dataRate 54Mb
set val(chan)
                        Channel/WirelessChannel
set val(prop)
                        Propagation/TwoRayGround
set val(netif)
                        Phy/WirelessPhy
                        Mac/802_11e
set val(mac)
                        Queue/DTail/PriQ
set val(ifq)
set val(II)
                        LL
                        Antenna/OmniAntenna
set val(ant)
                                        ;# coordenada X da topografia
set val(x)
                        500
set val(y)
                        500
                                        ;# coordenada Y da topografia
set val(ifglen)
                        1000
                                        ;# max pacotes em ifq
set val(adhocRouting) OLSR
                                        ;# protocolo de roteamento
set val(stop)
                        150.0
                                        ;# tempo de simulacao
set val(txPower)
                       0.0
                                        ;# taxa de transmissao de dados em W
set val(rxPower)
                                        ;# taxa de repecpcao de dados em W
                       0.0
set val(idle)
                        0.0
                                        ;# taxa de escuta ociosa em W
# Programa principal
# Inicia as Variaveis globais
set run 1
for {set i 0} {$i < $argc } {incr i} {
set opti [lindex $argv $i]
if {$opti == "-cenario"} {
       set val(cenario) [lindex $argv [incr i]]
} elseif {$opti == "-seed"} {
       set val(seed) [lindex $argv [incr i]]
} elseif {$opti == "-run"} {
       set run [lindex $argv [incr i]]
} elseif {$opti == "-n"} {
       set val(nn) [lindex $argv [incr i]]
} elseif {$opti == "-trafego"} {
       set val(traf) [lindex $argv [incr i]]
} elseif {$opti == "-cfb"} {
       set val(cfb) [lindex $argv [incr i]]
} elseif {$opti == "-cfb adapt"} {
       set val(cfb adapt) [lindex $argv [incr i]]
} elseif {$opti == "-pf0"} {
       set val(pf0) [lindex $argv [incr i]]
} elseif {$opti == "-aifs0"} {
       set val(aifs0) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin0"} {
       set val(cwmin0) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax0"} {
       set val(cwmax0) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit0"} {
```

set val(txop0) [lindex \$argv [incr i]]

set val(pf1) [lindex \$argv [incr i]]

} elseif {\$opti == "-pf1"} {

```
} elseif {$opti == "-aifs1"} {
       set val(aifs1) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin1"} {
       set val(cwmin1) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax1"} {
       set val(cwmax1) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit1"} {
       set val(txop1) [lindex $argv [incr i]]
} elseif {$opti == "-pf2"} {
       set val(pf2) [lindex $argv [incr i]]
} elseif {$opti == "-aifs2"} {
       set val(aifs2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin2"} {
       set val(cwmin2) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax2"} {
       set val(cwmax2) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit2"} {
       set val(txop2) [lindex $argv [incr i]]
} elseif {$opti == "-pf3"} {
       set val(pf3) [lindex $argv [incr i]]
} elseif {$opti == "-aifs3"} {
       set val(aifs3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmin3"} {
       set val(cwmin3) [lindex $argv [incr i]]
} elseif {$opti == "-cwmax3"} {
       set val(cwmax3) [lindex $argv [incr i]]
} elseif {$opti == "-txopLimit3"} {
       set val(txop3) [lindex $argv [incr i]]
} else {
       puts "Opcao invalida: \"$opti\""
       exit 1
remove-all-packet-headers
add-packet-header IP ARP OLSR LL Mac
set ns [new Simulator]
                                         :# Cria instancia do simulador
$ns use-scheduler Heap
set topo [new Topography]
                                         ;# Seta o objeto topografia
set trace_file "cenario_mineiro_olsr-s$val(seed).tr"
set tracefd [open $trace_file w]
                                        ;# Cria o objeto trace para ns
$ns_ trace-all $tracefd
set max_fragmented_size 1000
set packetSize 1028
$topo load_flatgrid $val(x) $val(y)
                                         ;# Define a topologia
set god_ [create-god $val(nn)];# Cria God
set chan_ [new $val(chan)]
# Define como os nos devem ser criados
# Configurações globais dos nos
$ns_ node-config -adhocRouting $val(adhocRouting) \
       -IIType $val(II) \
       -macType $val(mac) \
       -ifqType $val(ifq) \
       -ifqLen $val(ifqlen) \
       -antType $val(ant) \
```

```
-propType $val(prop) \
       -phyType $val(netif) \
       -channel $chan \
      -topolnstance $topo \
      -rxPower $val(rxPower) \
      -txPower $val(txPower) \
      -idlePower $val(idle) \
      -agentTrace ON \
      -routerTrace ON \
      -macTrace ON \
      -movementTrace OFF \
if \{\$AKAROA == 1\} {
      set ak [new Akaroa]
      $ak AkDeclareParameters 8
}
# Cria um especifico numero de nos [$val(nn)] e os associa aos canais
for {set i 0} {$i < $val(nn) } {incr i} {
      set node_($i) [$ns_ node]
      $node_($i) random-motion 0
                                              ;# Desabilita movimentação aleatória
}
#Define a posicao dos nos
source $val(cenario)
#Define padrao de trafego
source $val(traf)
#Vídeo a ser transmitido
set original file name news gcif
set trace file name "news $run.dat"
set original file id [open $original file name r]
set trace_file_id [open $trace_file_name w]
#set frame count 0
#set last time 0
set pre time 0
while \{[eof \$original file id] == 0\}
     gets $original_file_id current_line
     scan $current_line "%d%s%d%d%f" no_ frametype_ length_ tmp1_ tmp2_
      set time [expr int(($tmp2_ - $pre_time)*1000000.0)]
     if {$frametype_ == "I"} {
          set type_v 1
          set prio_p 1
     if {$frametype_ == "P"} {
          set type_v 2
          set prio p 1
     if {$frametype_ == "B"} {
          set type v 3
          set prio p 1
     if {$frametype == "H"} {
          set type v 1
          set prio p 1
     puts $trace_file_id "$time $length_ $type_v $prio_p $max_fragmented_size"
```

```
set pre_time $tmp2_
}
close $original_file_id
close $trace_file_id
set start_video_time 50.0
set end_sim_time [expr $tmp2_ + $start_video_time]
puts "$end_sim_time"
set trace_file [new Tracefile]
$trace_file filename $trace_file_name
set video1 [new Application/Traffic/myEvalvid]
$video1 attach-agent $udp1
$video1 attach-tracefile $trace_file
$ns at $start video time "$video1 start"
$ns_ at $end_sim_time "$video1 stop"
$ns_ at $val(stop).0 "$null1 closefile"
# Diz aos nos quando a simulacao termina
for {set i 0} {$i < $val(nn) } {incr i} {
  $ns_ at $val(stop).0 "$node_($i) reset";
$ns_ at $val(stop).0002 "puts \"NS EXITING...\"; $ns_ halt"
puts $tracefd "M 0.0 nn $val(nn) x $val(x) y $val(y) rp $val(adhocRouting)"
puts $tracefd "M 0.0 prop $val(prop) ant $val(ant)"
puts "Starting Simulation $run..."
$ns run
```

A.2.3. CENÁRIO SWAN (EVALVID)

```
# Opcoes
Mac/802 11 set dataRate 54Mb
set opt(chan)
                    Channel/WirelessChannel
set opt(prop)
                    Propagation/TwoRayGround
set opt(netif)
                    Phy/WirelessPhy
                    Mac/802 11
set opt(mac)
                    Queue/DropTail/PriQueue
set opt(ifq)
set opt(II)
                    Antenna/OmniAntenna
set opt(ant)
set opt(x)
                    500
                                   ;# coordenada X da topografia
                    500
                                   ;# coordenada Y da topografia
set opt(y)
set opt(progress)
                    4
                                  ;# marcador de progresso
                    1000
set opt(ifqlen)
                                   ;# max pacotes em ifq
                                   ;# número de nós
set opt(nn)
                    10
set opt(stop)
                    480.0
                                   ;# tempo de simulação
                    480.0
set val(stop)
set opt(rp)
                    OLSR
                                   ;# protocolo de roteamento
                   "ON"
                                   :# contralador de taxa ON/OFF
set opt(swan rc)
                    "ON"
                                   :# controlador de admissão ON/OFF
set opt(swan ac)
                "result/test"
                                   :# diretório destino de resultados
set opt(dir)
                                   ;# taxa inicial
set opt(band)
                  "100Kb"
                   "1Mb"
                                   ;# limiar do slow start
set opt(ssthresh)
                   "50Kb"
set opt(segment)
                                   ;# incremento de segmento
                   "50"
set opt(mdrate)
                                   ;# decremento de taxa
                  "1.2"
set opt(gap)
                                   ;# gap de controle
                   "100kb"
set opt(minband)
                                   ;# taxa mínima
                  "1000Kb"
set opt(acrate)
                                   ;# taxa para controle de admissão
set opt(thrate)
                  "2000Kb"
                                   :# limiar de taxa
set run 1
for {set i 0} {$i < $argc } {incr i} {
set opti [lindex $argv $i]
if {$opti == "-cenario"} {
      set val(cenario) [lindex $argv [incr i]]
} elseif {$opti == "-seed"} {
      set val(seed) [lindex $argv [incr i]]
} elseif {$opti == "-run"} {
      set run [lindex $argv [incr i]]
} elseif {$opti == "-n"} {
      set val(nn) [lindex $argv [incr i]]
} elseif {$opti == "-trafego"} {
      set val(traf) [lindex $argv [incr i]]
```

```
set AgentTrace
                            ON
set RouterTrace
                            ON
                            ON
set MacTrace
LL set mindelay_
                            50us
LL set delay
                            25us
LL set bandwidth_
LL set off_prune_
                            0
LL set off_CtrMcast_
Agent/Null set sport_
                            0
Agent/Null set dport_
                            0
Agent/CBR set sport_
Agent/CBR set dport_ 0
                            0
Agent/TCPSink set sport
Agent/TCPSink set dport_
Agent/TCP set sport
Agent/TCP set dport
Agent/TCP set packetSize_
                            1460
Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1
# Antenas omni-directional com ganho unitário e altura de 1,5m.
Antenna/OmniAntenna set X 0
Antenna/OmniAntenna set Y_
Antenna/OmniAntenna set Z_1 1.5
Antenna/OmniAntenna set Gt 1.0
Antenna/OmniAntenna set Gr_ 1.0
Phy/WirelessPhy set bandwidth 54e6
Mac set bandwidth_ 54e6
#-----
proc usage { argv0 } {
      puts "Usage: $argv0"
      puts "\tmandatory arguments:"
      puts "\t\\[-x MAXX\] \[-y MAXY\]"
      puts "\toptional arguments:"
      puts "\t\t\[-cp conn pattern\] \[-sc scenario\] \[-nn nodes\]"
      puts "\t\t\[-seed seed\] \[-stop sec\] \[-tr tracefile\]\n"
proc getopt {argc argv} {
      global opt
      lappend optlist cp nn seed sc stop tr x y
      for {set i 0} {$i < $argc} {incr i} {
             set arg [lindex $argv $i]
             if {[string range $arg 0 0] != "-"} continue
             set name [string range $arg 1 end]
             set opt($name) [lindex $argv [expr $i+1]]
      }
proc makedir { pathname } {
  if {[file isdirectory $pathname]} {
      return $pathname
  } elseif {[file exists $pathname]} {
      error "Non-directory $pathname already exists."
      makedir [file dirname $pathname]
      exec mkdir $pathname
      return $pathname
  }
```

```
# Programa Principal
getopt $argc $argv
if \{ \text{sopt}(x) == 0 \mid | \text{sopt}(y) == 0 \} \{
      usage $argv0
      exit 1
}
if \{\text{sopt(seed)} > 0\} {
      puts "Seeding Random number generator with $opt(seed)\n"
      ns-random $opt(seed)
# Inicializa Variáveis Globais
remove-all-packet-headers
add-packet-header IP ARP OLSR LL Mac
set ns_
                     [new Simulator]
             [new Topography]
set topo
             [new God]
set god_
makedir $opt(dir)
set trace_file "cenario_mineiro_swan-s$val(seed).tr"
set tracefd [open $trace_file w]
                                    ;# Cria o objeto trace para ns
$ns_ trace-all $tracefd
set max fragmented size 1000
set packetSize 1028
$topo load_flatgrid $opt(x) $opt(y)
create-god $opt(nn)
set chan_1_ [new $opt(chan)]
#Configurações do nó
$ns_ node-config -adhocRouting $opt(rp) \
              -IIType $opt(II) \
              -macType $opt(mac) \
              -ifqType $opt(ifq) \
              -ifqLen $opt(ifqlen) \
              -antType $opt(ant) \
              -propType $opt(prop) \
              -phyType $opt(netif) \
              -channel $chan_1_\
              -topolnstance $topo \
              -macTrace ON \
              -rxPower 0.3 \
              -txPower 0.6 \
for {set i 0} {$i < $opt(nn) } {incr i} {
  set node ($i) [$ns node]
  $node ($i) random-motion 0
                                            ;# desabilita movimentação aleatória
}
```

```
# Define a posicao dos nos
source $val(cenario)
#Define padrao de trafego
source $val(traf)
#Vídeo a ser transmitido
set original_file_name news_qcif
set trace_file_name "news_$run.dat"
set original_file_id [open $original_file_name r]
set trace_file_id [open $trace_file_name w]
#set frame count 0
#set last time 0
set pre time 0
while {[eof $original file id] == 0} {
     gets $original file id current line
     scan $current_line "%d%s%d%d%f" no_frametype_length_tmp1_tmp2_
      set time [expr int(($tmp2_ - $pre_time)*1000000.0)]
     if {$frametype_ == "I"} {
          set type_v 1
          set prio_p 1
     if {$frametype_ == "P"} {
          set type_v 2
          set prio p 1
     if {$frametype_ == "B"} {
          set type_v 3
          set prio_p 1
     if {$frametype_ == "H"} {
          set type_v 1
          set prio_p 1
     puts $trace_file_id "$time $length_ $type_v $prio_p $max_fragmented_size"
     set pre time $tmp2
}
close $original file id
close $trace_file_id
set start_video_time 50.0
set end_sim_time [expr $tmp2_ + $start_video_time]
puts "$end_sim_time"
set trace_file [new Tracefile]
$trace file filename $trace file name
set video1 [new Application/Traffic/myEvalvid]
$video1 attach-agent $udp1
$video1 attach-tracefile $trace file
$ns_ at $start_video_time "$video1 start"
$ns at $end sim time "$video1 stop"
$ns_ at $val(stop).0 "$null1 closefile"
```

```
# Acionamento das funções shaper e monitor de utilização a cada segundo
for {set i 0} {$i <= $opt(stop)} {incr i 1} {
  }
}
#
# Diz aos nós quando a simulação termina
for {set i 0} {$i < $opt(nn) } {incr i} {
  $ns_ at $opt(stop).000000001 "$node_($i) reset";
$ns at $opt(stop).00000001 "puts \"NS EXITING...\"; $ns halt"
for {set i 1} {$i <= $opt(progress)} {incr i} {
  set t [expr $i * $opt(stop) / ($opt(progress) + 1)]
  $ns_ at $t "puts \"completed through $t secs...\""
puts $tracefd "G -t * -prop $opt(prop) -ant $opt(ant)"
puts "Starting Simulation..."
$ns_ run
```

A.2.4. CONFIGURAÇÃO DO TRÁFEGO (EVALVID)

```
#Configuração da classe RNG (aleatoriedade)
set sourceRNG [new RNG]
set destRNG [new RNG]
$sourceRNG seed $val(seed)
$destRNG seed $val(seed)
#Seleção aleatória dos nós origem e destino
set source_[new RandomVariable/Uniform]
$source set min 0
$source_ set max_ [expr $val(nn) - 1]
$source use-rng $sourceRNG
set dest_ [new RandomVariable/Uniform]
$dest_ set min_ 0
$dest_ set max_ [expr $val(nn) - 1]
$dest_ use-rng $destRNG
set s_ [expr round([$source_ value])]
set d_ [expr round([$dest_ value])]
while {$d_ == $s_} {
     set d [expr round([$dest value])]
}
#Vídeo do Evalvid
set udp1 [new Agent/myUDP]
$ns_ attach-agent $node_($s_) $udp1
$udp1 set packetSize $packetSize
$udp1 set filename sd news $run
$udp1 set class 1
$udp1 set prio 1
set null1 [new Agent/myEvalvid Sink]
$ns_ attach-agent $node_($d_) $null1
$ns connect $udp1 $null1
$null1 set filename rd news $run
#Demais fluxos
for {set t 1} {$t < $val(stop)} {incr t 10} {
set s_ [expr round([$source_ value])]
set d [expr round([$dest value])]
while \{\$d == \$s \} \{
     set d [expr round([$dest value])]
}
#Perfil de Tráfego
# Trafego de prioridade 0 = Voz
set udp_($t) [new Agent/UDP]
$ns_ attach-agent $node_($s_) $udp_($t)
$udp_($t) set class_ 0
$udp_($t) set prio_ 0
set null ($t) [new Agent/Null]
$ns attach-agent $node ($d ) $null ($t)
set cbr_($t) [new Application/Traffic/CBR]
$cbr ($t) set packetSize 160
$cbr_($t) set interval_ 20e-3
$cbr_($t) attach-agent $udp_($t)
```

```
$ns_ connect $udp_($t) $null_($t)
$ns_ at [expr $t + 30.0] "$cbr_($t) start"
# Trafego de prioridade 1 = Video
set udp_([expr $t + 1]) [new Agent/UDP]
$ns_ attach-agent $node_($s_) $udp_([expr $t + 1])
$udp_([expr $t + 1]) set class_ 1
$udp_([expr $t + 1]) set prio_ 1
set null_([expr $t + 1]) [new Agent/Null]
$ns_ attach-agent $node_($d_) $null_([expr $t + 1])
set cbr_([expr $t + 1]) [new Application/Traffic/CBR]
$cbr ([expr $t + 1]) set packetSize 1280
$cbr_([expr $t + 1]) set interval_ 10e-3
$cbr_([expr $t + 1]) attach-agent $udp_([expr $t + 1])
$ns connect $udp ([expr $t + 1]) $null ([expr $t + 1])
$ns at [expr $t + 30.0] "$cbr ([expr $t + 1]) start"
# Trafego de prioridade 2
set tcp ($t) [new Agent/TCP]
$ns_ attach-agent $node_($s_) $tcp_($t)
$tcp_($t) set class_ 2
$tcp_($t) set prio_ 2
set sink_($t) [new Agent/TCPSink]
$ns_ attach-agent $node_($d_) $sink_($t)
set ftp_($t) [new Application/FTP]
$tcp_($t) set packetSize_ 1300
$ftp_($t) attach-agent $tcp_($t)
$ns_ connect $tcp_($t) $sink_($t)
$ns_ at [expr $t + 30.0] "$ftp_($t) start"
# Trafego de prioridade 3
set tcp ([expr $t + 1]) [new Agent/TCP]
$ns_ attach-agent $node_($s_) $tcp_([expr $t + 1])
$tcp_([expr $t + 1]) set class_ 3
$tcp_([expr $t + 1]) set prio_ 3
set sink ([expr $t + 1]) [new Agent/TCPSink]
$ns attach-agent $node ($d ) $sink ([expr $t + 1])
set ftp ([expr $t + 1]) [new Application/FTP]
$tcp_([expr $t + 1]) set packetSize_ 1500
ftp_{(expr $t + 1]} attach-agent fcp_{(expr $t + 1]}
$ns_ connect $tcp_([expr $t + 1]) $sink_([expr $t + 1])
$ns_ at [expr $t + 30.0] "$ftp_([expr $t + 1]) start"
```