UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

UMA ANÁLISE DA QoS NA TRANSMISSÃO EM REDES 802.11G SOB PRESENÇA DE INTERFERÊNCIA BLUETOOTH

Elaborado por:

Leandro Cabral Figueiredo

Recife, Dezembro de 2008.

UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

UMA ANÁLISE DA QoS NA TRANSMISSÃO EM REDES 802.11G SOB PRESENÇA DE INTERFERÊNCIA BLUETOOTH

por

LEANDRO CABRAL FIGUEIREDO

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Pernambuco como parte dos requisitos para a obtenção do grau de Mestre em Engenharia Elétrica.

ORIENTADOR: RAFAEL DUEIRE LINS, Ph.D.

Recife, Dezembro de 2008.

F475a **Figueiredo, Leandro Cabral.**
Uma análise da QoS na transmissão em redes 802.11G sob presença de interferência Bluetooth / Leandro Cabral Figueiredo. - Recife: O Autor, 2008.
xii, 155 folhas, il : grafs., tabs.

Dissertação (Mestrado) – Universidade Federal de Pernambuco. CTG. Programa de Pós-Graduação em Engenharia Elétrica, 2008.

Inclui bibliografia e Apêndice.

1. Engenharia Elétrica. 2. Redes sem Fio 3. *Bluetooth*. 4. Redes IEEE 802.11g 5. Interferência. I. Título.

UFPE

621.3 **CDD (22. ed.)** **BCTG/2009-009**

Universidade Federal de Pernambuco

Pós-Graduação em Engenharia Elétrica

PARECER DA COMISSÃO EXAMINADORA DE DEFESA DE DISSERTAÇÃO DO MESTRADO ACADÊMICO DE

LEANDRO CABRAL FIGUEIREDO

TÍTULO

"UMA ANÁLISE DA Q_0S NA TRANSMISSÃO EM REDES 802.11G SOB PRESENÇA DE INTERFERÊNCIA BLUETOOTH"

A comissão examinadora composta pelos professores: RAFAEL DUEIRE LINS, DES/UFPE, VALDEMAR CARDOSO DA ROCHA JÚNIOR, DES/UFPE, e CARMELO JOSÉ ALBANEZ BASTOS FILHO, DCC/UPE sob a presidência do primeiro, consideram o candidato **LEANDRO CABRAL FIGUEIREDO APROVADO**.

Recife, 10 de dezembro de 2008.

EDUARDO FONTANA
Coordenador do PPGEE

RAFAEL DUEIRE LINS
Orientador e Membro Titular Interno

CARMELO JOSÉ ALBANEZ BASTOS FILHO
Membro Titular Externo

VALDEMAR CARDOSO DA ROCHA JÚNIOR
Membro Titular Interno

DEDICATÓRIA

Dedico este trabalho a Luiz Carlos de Figueiredo, Telma Maria Cabral Figueiredo, Danielle Cabral Figueiredo e Thaís Chaves de Oliveira.

AGRADECIMENTOS

A Deus, que sempre nos apóia para vencermos novas etapas em nossas vidas; aos meus pais e amigos, por todo apoio que nos foi dado; ao Professor Doutor Rafael Dueire Lins, pela orientação, atenção e tempo dedicado para a realização e desenvolvimento deste trabalho; à banca examinadora por tornar esta dissertação mais completa e aos colegas de trabalho da Celpe e do curso de pós-graduação da UFPE pelo companheirismo e amizade.

Resumo da Dissertação apresentada à UFPE como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica.

UMA ANÁLISE DA QoS NA TRANSMISSÃO EM REDES 802.11G SOB PRESENÇA DE INTERFERÊNCIA BLUETOOTH

Leandro Cabral Figueiredo

Dezembro/2008

Orientador: Rafael Dueire Lins, Ph.D.

Área de Concentração: Redes IEEE 802.11g.

Palavras-chave: interferência, IEEE 802.11g, *Bluetooth*

Número de Páginas: 155.

RESUMO: Esta dissertação de mestrado trata da interferência entre os sistemas de rede sem fio IEEE 802.11g e *Bluetooth*. Mais especificamente, foi analisado o impacto que transmissões *Bluetooth* causam em transmissões de dados IEEE 802.11g. Foram analisados cinco parâmetros: tempo de transmissão, taxa média de transmissão de pacotes, taxa média de transmissão de dados, número de pacotes perdidos e número de pacotes ACK duplicados. Por meio da comparação entre os resultados obtidos para esses parâmetros sem interferência *Bluetooth* e com interferência *Bluetooth*, foi possível determinar que a transmissão *Bluetooth* afetou a transmissão IEEE 802.11g. Os experimentos foram realizados para três distâncias entre os transmissores e receptores *Bluetooth* e *Wi-Fi*. Inicialmente, avaliou-se a interferência para a distância de 1,60 m. Em seguida, para 2,60 m e, por fim, para 4,60 m. Para as distâncias de 1,60 m e 2,60 m não havia obstáculos entre os dispositivos de transmissão e recepção. Para a distância de 4,60 m, havia uma parede de alvenaria. Os resultados obtidos mostraram que a interferência *Bluetooth* na transmissão IEEE 802.11g foi maior com o aumento da distância, sendo mais significativa para a distância de 4,60 m.

Abstract of Dissertation presented to UFPE as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering.

AN ANALYSIS OF THE QoS IN THE TRANSMISSION IN 802.11G NETWORKS IN THE PRESENCE OF BLUETOOTH INTERFERENCE

Leandro Cabral Figueiredo

December/2008

Supervisor(s): Rafael Dueire Lins, Ph.D.

Area of Concentration: IEEE 802.11g Networks.

Keywords: interference, IEEE 802.11g, Bluetooth

Number of Pages: 155.

ABSTRACT: This MSc thesis deals with the interference between the IEEE 802.11g and Bluetooth wireless network systems. More specifically, it was analyzed the impact that Bluetooth transmissions causes on IEEE 802.11g data transmissions. Five parameters were analyzed: transmission time, average packet transmission rate, average data transmission rate, number of lost packets and number of duplicate ACK packets. By comparing the results obtained for these parameters with and without Bluetooth interference, it was possible to determine that the Bluetooth transmission affected the IEEE 802.11g one. The experiments were performed at three distances between the Bluetooth and Wi-Fi transmitters and receivers. Initially, the interference was measured for the distance of 1.60 m. Then, for 2.60 m and, finally, for 4.60 m. For the distances of 1.60 m and 2.60 m, there was no obstacle between the transmitter and receiver devices. For the distance of 4.60 m, there was a wall of bricks covered with plaster. The results obtained showed that the Bluetooth interference on the IEEE 802.11g transmission increased with the distance, being more significant for the distance of 4.60 m.

SUMÁRIO

Lista de Tabelas

Lista de Figuras

Capítulo 1. INTRODUÇÃO

O surgimento da tecnologia VoIP (*Voice over IP*) veio como uma alternativa bastante interessante e de baixo custo, tanto para usuários domésticos quanto empresariais, dos serviços de telefonia convencional para ligações não-locais, sejam nacionais ou internacionais [1].

Existe um interesse cada vez maior por parte das organizações em se estabelecer conexões de voz a baixos custos. Com a grande competição existente no mercado, por exemplo, é muito importante para elas, principalmente as micro e pequenas empresas, reduzirem seus gastos internos com o objetivo de se ter mais recursos para investirem em projetos importantes. Também nesse cenário, o VoIP aparece como uma opção atraente.

O VoIP se refere ao uso de uma rede, que originalmente foi projetada para transmitir dados, para o uso no transporte de amostras de voz. Em virtude do fato de que amostras de voz comprimidas [2] podem consumir pouca largura de banda, pesquisadores têm investigado como a Internet pode prover chamadas telefônicas com boa qualidade.

Paralelamente ao aumento no uso da Internet para tráfego de voz, o uso de redes sem fios para acesso direto do usuário (última milha) aumentou significativamente. Uma das tecnologias de redes sem fios mais conhecidas é a da telefonia móvel celular, cujo número de linhas já ultrapassou o da telefonia fixa.

Com a evolução das redes sem fio e com o aumento no seu uso, diferentes tecnologias foram e ainda estão sendo definidas, testadas e implementadas, tais como:

- WWAN (*Wireless Wide Area Networks*). Um exemplo são as redes celulares GSM (*Global System for Mobile communication*);
- WMAN (*Wireless Metropolitan Area Networks*). O padrão IEEE 802.16, também conhecido por WiMax (*Worldwide Interoperability for Microwave Access*), é um exemplo de WMAN;
- WLAN (*Wireless Local Area Networks*). O padrão IEEE (*Institute of Electrical and Electronics Engineers*) 802.11 é um exemplo.
- WPAN (*Wireless Personal Area Networks*). A tecnologia *Bluetooth* presente em vários dispositivos, como telefones celulares, é um exemplo.

Historicamente, o surgimento das WLANs remontam ao ano de 1997, quando o IEEE publicou o padrão IEEE 802.11 dedicada a redes locais sem fio. A numeração 802 se

refere ao grupo de documentos que descrevem as características recomendadas pelo IEEE para redes locais [3].

Em 1995, surgiu o *Bluetooth*, uma tecnologia da Ericsson desenvolvida para conectar, por meio de ondas de rádio, os telefones móveis a diversos acessórios. Há pouco tempo, surgiu um grupo de estudos formado por fabricantes líderes mundiais no ramo das telecomunicações que estavam interessados nessa tecnologia para aplicá-la a outros dispositivos, como PDAs (*Personal Digital Assistant*), terminais móveis e até eletrodomésticos [3].

Entretanto, o verdadeiro desenvolvimento e disseminação dessas tecnologias de redes sem fio vieram a partir do momento que o FCC (*Federal Communications Commission*), órgão americano responsável por regular as emissões radioelétricas, aprovou o uso civil da técnica de transmissão em banda larga SS (*Spread Spectrum*). Essa técnica de transmissão já era utilizada no âmbito militar desde a Segunda Guerra Mundial devido as suas extraordinárias características no que diz respeito à dificuldade de rastreamento e tolerância a interferências externas.

Um fato que não pode ser ignorado é que as WLANs IEEE 802.11 têm adquirido muito espaço no mercado e na vida das pessoas. Podemos encontrar WLANs em locais públicos, residências, escritórios, universidades e em diversos outros estabelecimentos, tornando o acesso à Internet praticamente onipresente.

As redes sem fio 802.11 possuem uma série de vantagens em relação às redes convencionais cabeadas, pois as primeiras não estão restritas fisicamente à uma conexão. Esse fato lhes concede uma maior mobilidade, facilidade de instalação e liberdade de localização, tornando-as sérias concorrentes das redes Ethernet em locais onde se requer grande mobilidade, tais como fábricas, áreas de armazenagem, congressos ou escritórios temporários. Em áreas com características de mobilidade, a montagem de redes cabeadas, além de demandar uma infra-estrutura fixa, restringe a liberdade de movimentação dos terminais, que é uma condição imprescindível [3]. No entanto, as redes Ethernet podem conseguir taxas de transmissão mais altas.

No cenário atual, no qual se observa a expansão e consolidação simultânea das tecnologias VoIP e 802.11, além da disseminação de aplicativos VoIP como o *MSN messenger*, o *Skype* ou o *GoogleTalk*, as chamadas de voz estabelecidas via Internet [2] estão cada vez mais sendo encaminhadas por meio de redes sem fio.

Todavia, algumas WLANs estão susceptíveis a vários tipos de fontes interferentes por operarem na faixa de freqüência ISM (*Industrial*, *Scientific*, *Medical*) de 2,4 GHz. A banda ISM é não-licensiada, ou seja, os dispositivos que funcionam nessa faixa não precisam de autorização por parte dos órgãos reguladores. Telefones sem fio, fornos de microondas e dispositivos com *Bluetooth* habilitado podem ser citados como alguns exemplos de fontes interferentes das WLANs que operam na mesma banda ISM. Outra faixa de freqüência não-licenciada é a U-NII (*Unlicensed-National Information Infra-structure*) de 5 GHz, sendo que a banda ISM é a mais utilizada.

O propósito desta dissertação é avaliar se a presença de dispositivos com *Bluetooth* ativado próximos à WLAN podem degradar a qualidade do sinal da rede, provocando perda de pacotes e conseqüente queda no desempenho.

1.1 A interferência *Bluetooth* – IEEE 802.11g

Um dos maiores desafios para as WLANs que trabalham na banda ISM é operar em proximidade a dispositivos *Bluetooth* [2]. A tecnologia *Bluetooth* tem o objetivo de substituir cabos para comunicações de curto alcance. O seu uso já está bastante difundido e as pessoas a utilizam, por exemplo, para efetuar troca de arquivos entre telefones celulares e interconectividade entre periféricos de computadores, como *mouse* e impressora.

Já as WLANs IEEE 802.11 têm como objetivo prover comunicação sem fio em uma área de cobertura maior, fornecendo interconectividade de rede aos terminais móveis. Dessa forma, o *Bluetooth* surge como um sistema complementar, e será bastante provável que dispositivos utilizando cada uma dessas tecnologias estejam ativos simultaneamente e próximos uns dos outros em, por exemplo, residências e escritórios.

Em virtude das redes sem fio com acesso à Internet e dos dispositivos com tecnologia *Bluetooth* estarem cada vez mais presentes e disponíveis nas vidas das pessoas, identificou-se a motivação e importância de, nesse trabalho dissertativo, avaliar o desempenho da transmissão em redes sem fio quando submetidas a fontes interferentes *Bluetooth*. Mais especificamente, o estudo desenvolvido aqui trata sobre a degradação da transmissão de uma massa de dados em uma WLAN IEEE 802.11g quando submetida à interferência *Bluetooth*.

A razão de se estudar a interferência *Bluetooth* em redes *Wi-Fi* e não o oposto foi pelo fato de que as transmissões 802.11 são muito mais afetadas pelo *Bluetooth* do que o contrário. Como o sistema *Bluetooth* utiliza uma técnica de transmissão de salto em

freqüência, ele combate interferências simplesmente saltando de uma freqüência para outra [4].

Um outro fato é que a largura de banda de um canal 802.11 é de 22 MHz, maior do que um canal *Bluetooth*, que é de 1 MHz. Com isso, quando comparado com a potência do sinal *Bluetooth*, a potência do sinal 802.11 é distribuída por uma largura espectral maior, reduzindo sua densidade espectral de potência e fazendo com que a intensidade do sinal 802.11 atenue abaixo do sinal *Bluetooth*. Isso torna o sinal 802.11 susceptível as interferências. Além disso, como os pacotes *Bluetooth* são muito menores do que os pacotes 802.11, ao ocorrer uma colisão, poucos dados *Bluetooth* serão perdidos e o sistema retransmitirá o pacote rapidamente em outra freqüência [4].

Visando embasar a presente dissertação, foi efetuada extensa pesquisa bibliográfica na área e foi constatado que este assunto já foi tratado por outros autores [2, 3, 4, 7, 12, 24, 33, 34, 35, 36, 37, 38, 40], alguns com enfoque teórico e outros com resultados experimentais. Este trabalho dissertativo se diferencia de outros estudos na área pois aborda uma topologia de rede diferente para realizar os testes, adota uma metodologia de trabalho também diferenciada e todos os testes foram feitos em um ambiente residencial que pode ser semelhante ao que muitas pessoas possuem em suas casas.

Nesta dissertação, além de tal estudo da literatura técnica existente, efetuamos testes para validar (ou não) os resultados apresentados na bibliografia. O cenário de testes utilizado neste estudo é apresentado na Figura 1.1. Esse cenário é mais detalhado no capítulo 7.

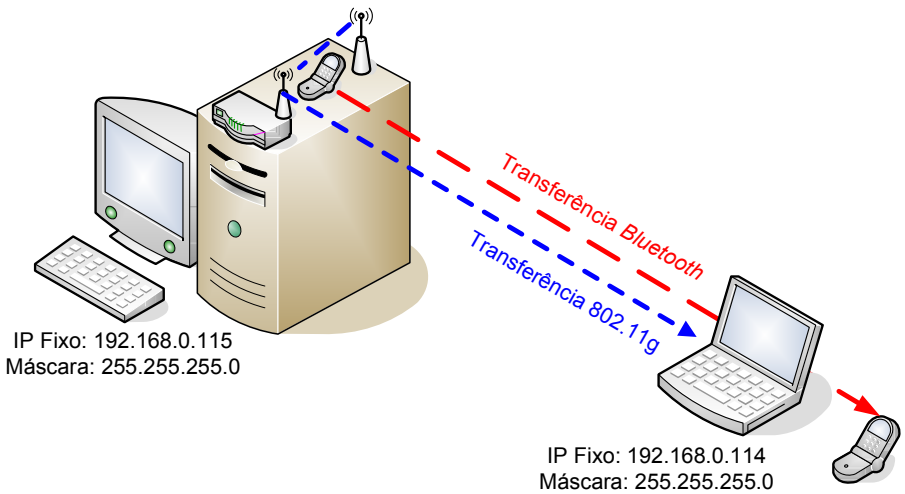


Figura 1.1 – *Cenário de testes.*

1.2 Estrutura desta dissertação

Esta dissertação é formada por nove capítulos adicionais a estes de Introdução. O capítulo 2 apresenta um sumário sobre o padrão EEE 802.11 e o *Bluetooth*. Os capítulos 3 e 4 tratam das camadas físicas dos sistemas 802.11g e *Bluetooth*, respectivamente. O capítulo 5 faz uma modelagem matemática da interferência entre esses dois sistemas de rede sem fio. O capítulo 6 apresenta uma revisão bibliográfica. O capítulo 7 faz uma descrição dos experimentos realizados e dos cenários de testes implementados. O capítulo 8 faz uma análise dos resultados obtidos e o capítulo 9 apresenta as conclusões e sugestões para trabalhos futuros.

Em apêndices a esta dissertação, apresentamos a técnica de modulação OFDM (*Orthogonal Frequeny Division Multiplexing*), a técnica de transmissão FHSS (*Frequency Hopping Spread Spectrum*), as transformadas rápidas de Fourier FFT (*Fast Fourier Transform*) e IFFT (*Inverse Fast Fourier Transform*), técnicas de segurança e criptografia em redes sem fio e as modulações BPSK (*Binary Phase Shift Keying*), QPSK (*Quadrature Phase Shift Keying*) e QAM (*Quadrature Amplitude Modulation*). Todos esses conceitos são necessários ao entendimento dos sistemas aqui descritos.

Capítulo 2. O PADRÃO IEEE 802.11 E O *BLUETOOTH*

As redes sem fio no padrão IEEE 802.11, também chamadas de redes *Wi-Fi*, podem ser encontradas tanto em ambientes corporativos quanto domiciliares, podendo ser encontradas em larga escala nos mais varados ambientes, tais como aeroportos, universidades, *cyber* cafés, etc.

A popularidade dessas redes deve-se, principalmente, ao fato desse tipo de rede dispensar a implementação de uma infra-estrutura de cabeamento, baixo custo, facilidade de instalação e uso [5]. Existem situações onde esses fatores são primordiais, como, por exemplo, em residências, onde a passagem de cabeamento é inconveniente ou em prédios históricos, onde o investimento em infra-estrutura de cabeamento seria oneroso em virtude da abrangência da área de cobertura necessária, dificuldades arquiteturais, etc.

Outro fator que contribuiu para a disseminação das redes sem fio foi a adesão da indústria às normas internacionais do IEEE (família 802.11). Isso proporcionou uma ampla interoperabilidade dos equipamentos e a conseqüente concorrência entre os fabricantes [5].

As WLANs são uma das mais inovadoras tecnologias de acesso à Internet de hoje. Na década de 1990, muitos padrões para LANs sem fio foram desenvolvidos. Entretanto, uma classe particular de padrões se mostrou a campeã: a WLAN IEEE 802.11, também conhecida como *Wi-Fi* [6].

Existem diversos padrões 802.11 para redes sem fio, dentre eles estão o 802.11b, 802.11a, 802.11g e mais recentemente o 802.11n. A Tabela 2.1 ilustra um resumo das principais características desses padrões.

Tabela 2.1 – *Principais características dos Padrões IEEE 802.11a/b/g/n.*

Padrão	Faixa de freqüência de operação	Taxa de dados
802.11b	2,4 – 2,485 GHz	até 11 Mbps
802.11a	5,1 – 5,8 GHz	até 54 Mbps
802.11g	2,4 – 2,485 GHz	até 54 Mbps
802.11n	2,4 – 2,485 e 5,1 – 5,8 GHz	até 300 Mbps

Os quatro padrões apresentados na Tabela 2.1 apresentam algumas características em comum, tais como [6]:

- todos usam o mesmo protocolo de acesso ao meio, o CSMA/CA (*Carrier Sense Multiple Access*/*Collision Avoidance*);
- todos usam a mesma estrutura de quadro de camada de enlace;

- todos têm a capacidade de reduzir sua taxa de transmissão para alcançar distâncias maiores;
- e todos os quatro padrões permitem dois modos de operação: em infra-estrutura ou BSS (*Basic Service Set*) e *ad hoc* ou IBSS (*Independent BSS*).

Como pode ser observado na Tabela 2.1, as WLANs 802.11b e 802.11g operam na faixa de freqüência que não necessita de licença de 2,4 a 2,485 GHz. Por operarem em uma faixa de freqüência superior, as WLANs 802.11a possuem um alcance menor para um dado nível de potência quando comparado ao alcance dos padrões 802.11b e 802.11g. Isso ocorre porque quanto maior for a freqüência de operação do sinal de rádio, mais penalizado ele será por propagação multipercurso e reflexões. O efeito de dispersão também é menor, fazendo com que a capacidade do sinal em contornar obstáculos seja reduzida. Assim, dado que o 802.11a atinge taxas mais altas que o 802.11b, o padrão 802.11g une o melhor dos outros dois padrões, a faixa de freqüência do 802.11b e a taxa de transmissão do 802.11a [6].

Os padrões IEEE 802.11a e 802.11g atingem uma taxa de transmissão na camada física de até 54 Mbps empregando o esquema de multiplexação COFDM (*Coded Orthogonal Frequency Division Multiplexing*). O padrão IEEE 802.11b atinge uma taxa de transmissão de até 11 Mbps na camada física empregando uma técnica denominada CCK (*Complementary Code Keying*). Os padrões que operam na banda ISM de 2,4 GHz facilitam a comunicação nesta faixa de freqüência utilizando as técnicas DS (*Direct Sequence*) e FH (*Frequency Hopped*) *Spread Spectrum* [7].

2.1 A topologia 802.11

A Figura 2.1 ilustra os principais componentes de uma WLAN. A estrutura básica de uma LAN sem fio é o BSS (*Basic Service Set*), onde o roteador 802.11 está conectado a uma rede cabeada, geralmente uma Ethernet padrão 802.3. O BSS é composto por uma ou mais estações sem fio e uma estação base central, denominada de AP (*Access Point*).

A Figura 2.1 mostra dois BSSs conectados entre si por dois AP's via um dispositivo de interconexão (hub, *switch* ou roteador). Esse dispositivo faz também a conexão tanto do BSS 1 quanto do BSS 2 com a Internet. Em uma rede residencial típica, existe apenas um AP com função de roteador conectado a um modem a cabo ou ADSL (*Assymmetric Digital Subscriber Line*), e esse à Internet.

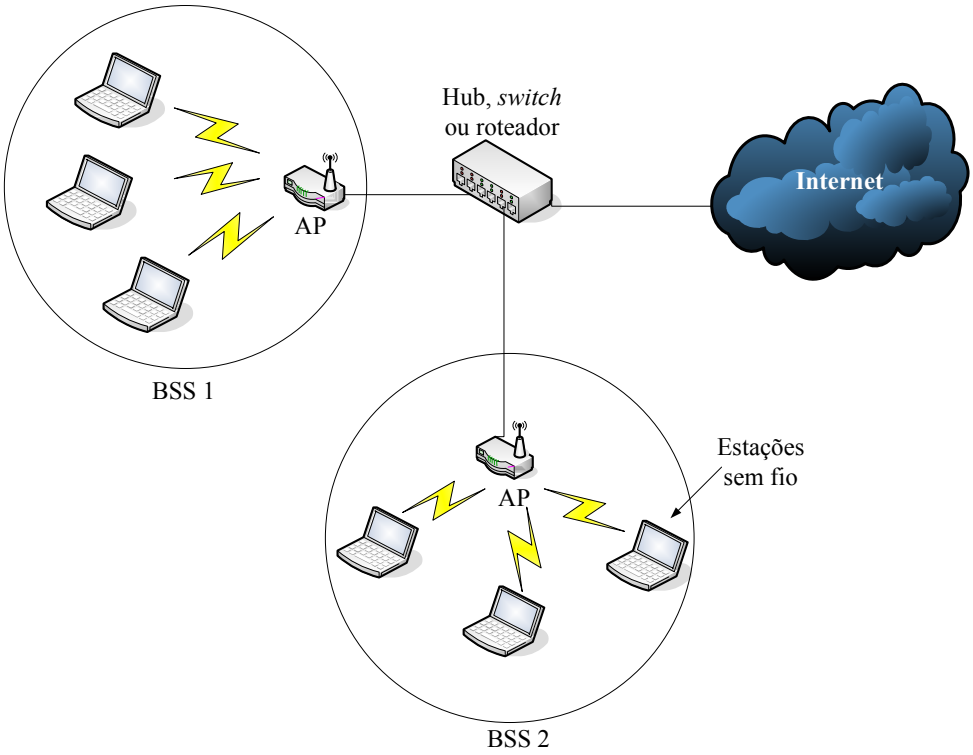


Figura 2.1 – *Estrutura básica de WLAN IEEE 802.11.*

Conforme dito anteriormente, os padrões IEEE 802.11 permitem dois modos de operação: em infra-estrutura e *ad hoc*. No modo de infra-estrutura existe um controle central que é realizado por um AP (roteador sem fio), que conecta o BSS à Internet através de uma rede cabeada. A Figura 2.1 ilustra esse tipo de configuração. Já no modo *ad hoc* não há um controle central, não existe um AP. Nesse caso, a rede é formada conforme a necessidade por equipamentos móveis que, por acaso, estão próximos uns aos outros, não havendo conexão com o "mundo externo". A Figura 2.2 ilustra o modo de operação *ad hoc*.

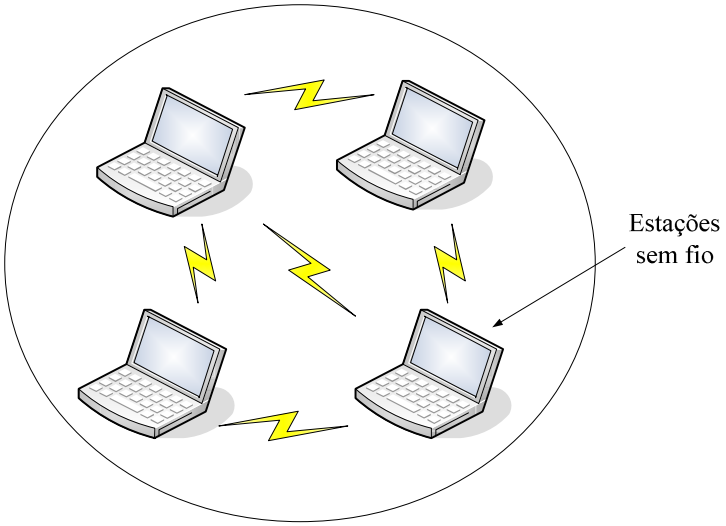


Figura 2.2 – *Modo de operação ad hoc.*

2.2 Canais e associação

No modo de infra-estrutura, cada estação sem fio deve se associar a um AP antes de poder enviar e receber quadros 802.11 contendo dados da camada de rede [6]. Para conseguir acesso à Internet em um aeroporto, por exemplo, um *laptop wireless* deverá se juntar a um BSS específico, ou seja, deverá se associar a um AP. Associar-se a um AP significa que o *laptop* estabelecerá um canal entre ele e o AP. Após concretizado esse canal de comunicação, somente esse AP específico enviará quadros contendo dados para o *laptop*, e esse, por sua vez, enviará quadros à Internet somente via esse AP associado.

Para que um determinado dispositivo sem fio fique ciente de todos os BSSs disponíveis para acesso, o padrão IEEE 802.11 estabelece que os APs devem, periodicamente, enviar quadros de sinalização, cada qual incluindo o SSID (*Service Set Identifier*) e o endereço MAC (*Medium Access Control*) do AP [6]. Ao se instalar uma WLAN, o administrador dessa rede sem fio deve designar um SSID (*Service Set Identifier*) ao AP. O SSID é o nome que identifica um BSS específico.

A estação sem fio, sabendo que os APs estão periodicamente enviando quadros de sinalização, faz uma varredura em todos os 11 canais (os quais serão especificados mais adiante) em busca desses quadros. Ao tomar conhecimento de todos os APs disponíveis, o dispositivo sem fio seleciona um deles para se associar. Após escolher o AP, o hospedeiro sem fio e o AP selecionado dialogam usando um protocolo de associação 802.11. A estação sem fio ficará associada ao AP selecionado se tudo ocorrer bem durante este diálogo. Vale observar aqui que, dependendo do proprietário da WLAN, poderá ser solicitada ou não uma senha para que a estação sem fio possa se associar a um determinado AP. Existem esquemas de criptografia de redes sem fio os quais serão abordados em seções posteriores.

Na banda de freqüência ISM, as normas 802.11b e 802.11g definem 11 canais de operação, cada um com uma largura de banda de 22 MHz e uma guarda entre cada canal de 5 MHz [8]. A Figura 2.3 ilustra um esquema com os 11 canais definidos pelas normas 802.11b e 802.11g na referida banda ISM.

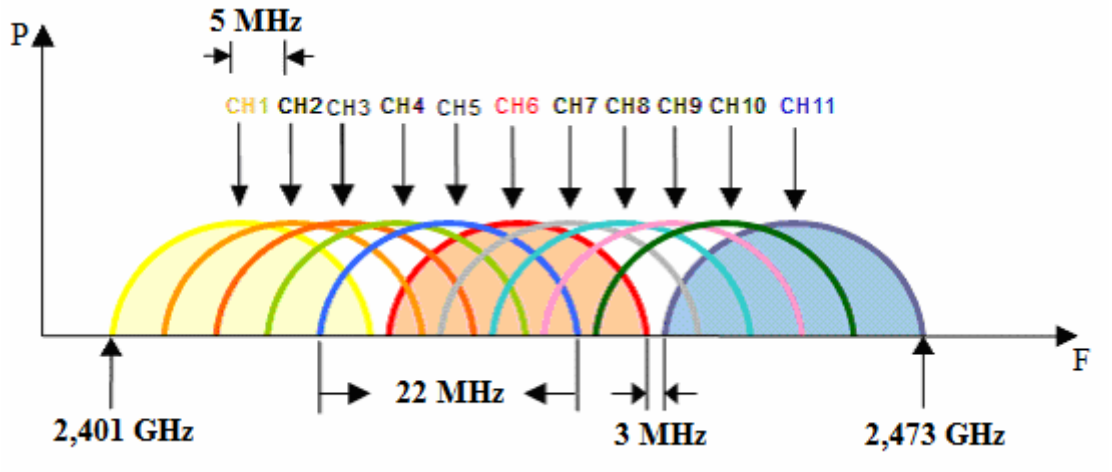


Figura 2.3 – *Canais 802.11 na banda de freqüências ISM (Modificado de [9]).*

Pela Figura 2.3 percebe-se que existe uma sobreposição dos canais nos padrões IEEE 802.11b e 802.11g. Dessa forma, não há uma completa isolação espectral dos canais, e redes 802.11 que operem em canais com algum nível de sobreposição podem sofrer problemas de interferência [8]. Ainda com base na Figura 2.3, observa-se que o único conjunto de canais que não se sobrepõe são os canais 1, 6 e 11. Esses três canais poderiam ser utilizados ao mesmo tempo por redes distintas de uma determinada região sem que ocorram interferências.

O problema de sobreposição espectral gera uma limitação na questão de alocação de canais para as redes 802.11b e 802.11g, visto que o número de canais não interferentes disponíveis para uso é pequeno.

As redes sem fio do padrão IEEE 802.11a também utilizam uma faixa de freqüências não licenciadas para sua operação. Conforme já dito anteriormente, esse padrão utiliza a faixa de 5 GHz. Essa faixa de freqüência é conhecida como U-NII. Esse padrão tem a seu dispor um total de 12 canais distintos de operação, os quais não apresentam sobreposição espectral e podem ser utilizados sem a presença de interferência entre canais adjacentes [8].

Apesar de oferecer mais canais para uso, os dispositivos baseados no padrão IEEE 802.11a não ganharam muito destaque no mercado por basicamente três razões:

- os dispositivos IEEE 802.11a não são compatíveis com os dispositivos IEEE 802.11b, já numerosos no mercado;
- o surgimento do padrão IEEE 802.11g, que oferece a mesma taxa de transmissão do padrão IEEE 802.11a, porém é compatível com o padrão IEEE 802.11b;

- e, pelo fato do padrão 802.11a operar em uma faixa de freqüência mais elevada, seus dispositivos são de custo mais elevado e o alcance de transmissão dos seus rádios são menores.

A Tabela 2.2 apresenta os diversos canais utilizados no padrão IEEE 802.11a e suas respectivas freqüências centrais de operação. Já a Tabela 2.3 apresenta as mesmas informações para os padrões 802.11b e 802.11g [8].

Tabela 2.2 – *Canais do padrão IEEE 802.11a.*

Canal	Freqüência (GHz)
36	5,180
40	5,200
44	5,220
48	5,240
52	5,260
56	5,280
60	5,300
64	5,320
149	5,745
153	5,765
157	5,785
161	5,805

Tabela 2.3 – *Canais dos padrões IEEE 802.11b/g.*

Canal	Freqüência (GHz)
1	2,412
2	2,417
3	2,422
4	2,427
5	2,432
6	2,437
7	2,442
8	2,447
9	2,452
10	2,457
11	2,562

2.3 O protocolo MAC 802.11

Tomando como base o enorme sucesso da Ethernet e seu protocolo de acesso aleatório (o CSMA/CD - *Carrier Sense Multiple Access*/*Collision Detection*), o padrão IEEE 802.11 adotou também um protocolo de acesso aleatório denominado CSMA/CA.

O sentido da sigla CSMA tanto no padrão IEEE 802.3 (Ethernet) quanto no padrão IEEE 802.11 (WLAN) significa acesso múltiplo com detecção de portadora. Isto quer dizer que, quando uma estação sem fio (no caso da WLAN) "deseja" transmitir alguma informação, ela deve primeiramente escutar o meio físico (canal) para descobrir se pode ou não realizar sua transmissão.

É importante neste momento frisar duas diferenças fundamentais entre o CSMA/CD e CSMA/CA [6]:

- o segundo protocolo não detecta colisões, como faz o primeiro. Ao invés de detectar colisões, o CSMA/CA faz o possível para evitá-las.
- diferentemente do Ethernet, o 802.11 utiliza uma técnica de reconhecimento/retransmissão (ARQ – *Automatic Repeat Request*) de camada de enlace de dados.

No padrão IEEE 802.3, ao utilizar o algoritmo CSMA/CD, cada estação monitora constantemente o canal enquanto está transmitindo. O objetivo disso é poder abortar a transmissão ao identificar a ocorrência de uma colisão. Já o algoritmo CSMA/CA não implementa detecção de colisão por dois motivos [6]:

- para que uma estação sem fio detecte colisões, ela precisaria ter a capacidade de enviar o próprio sinal e de, simultaneamente, receber sinais provenientes de colisões. Como a potência do sinal recebido geralmente é inferior quando comparada com a potência do sinal transmitido, seria custoso elaborar um hardware capaz de detectar colisões;
- mesmo que um adaptador pudesse transmitir e ouvir ao mesmo tempo, ainda assim ele não seria capaz de detectar todas as colisões devido ao problema do terminal oculto, que será explicado na seção 2.3.1 desta dissertação.

Dessa forma, é de extrema importância para o padrão IEEE 802.11 evitar colisões, pois, dado que o CSMA/CA não as detecta, e uma vez que um quadro 802.11 colida, todo ele será perdido. Isso implicará em desperdício de largura de banda, já que no protocolo CSMA/CA uma transmissão que se iniciou não será abortada, mesmo em caso de colisão.

Agora, serão vistas as técnicas utilizadas pelo 802.11 para prevenir as colisões.

O protocolo MAC 802.11 introduz dois diferentes métodos para controlar o acesso compartilhado ao meio sem fio: o PCF (*Point Coordination Fuction*) e o DCF (*Distributed Coordination Function*) [10, 11].

O modo PCF só pode ser implementado na configuração de infra-estrutura. O modo PCF foi originalmente destinado para suportar aplicações com tráfego de tempo real. Entretanto, ele raramente é implementado nas redes WLAN disponíveis [10, 11]. Isto se dá em virtude de sua complexidade de implementação e eficiência duvidosa [10].

O modo DCF é aplicável tanto na configuração de infra-estrutura quanto na configuração *ad hoc*. Este modo de operação é baseado no CSMA/CA. Cada estação móvel em um único BSS compete para transmitir o seu pacote de dados no meio físico compartilhado (o ar, para o caso das WLANs).

Como já dito anteriormente, cada estação sem fio deve sondar o meio antes de iniciar uma transmissão de um pacote. Ao monitorar o meio físico, a estação pode encontrar duas situações: (1) o meio está disponível, ou (2) o meio já está ocupado com uma transmissão em andamento.

Suponha que uma determinada estação sem fio de um BSS necessite enviar um pacote de dados. Considerando inicialmente o caso (1), mesmo que a estação móvel detecte que o meio está livre, ela não transmite imediatamente o seu pacote de dados. A estação deve aguardar, ainda monitorando o canal de comunicação, por um período de tempo predeterminado antes de efetivamente iniciar sua transmissão. Este intervalo de tempo é denominado de DIFS (*Distributed Interframe Space*). Se o meio físico permanecer ocioso durante o DIFS, a estação inicia sua transmissão logo após o término deste período de tempo.

Se a transmissão do pacote ocorrer com sucesso, o receptor deve enviar uma confirmação (ACK – *Acknowledgment*) ao transmissor dos dados. Ao receber o pacote, a confirmação deve ser enviada pelo receptor após um intervalo de tempo chamado de SIFS (*Short Interframe Space*). Se, por algum motivo, o transmissor não receber o ACK durante um intervalo de *time-out*, ele assumirá que o pacote de dados transmitido foi perdido, reenviando-o novamente. Entretanto, conforme será explicado mais adiante, essa retransmissão não acontece imediatamente. Nesse caso de retransmissão, pode ter ocorrido que o pacote enviado foi realmente perdido ou houve perda da própria confirmação. A Figura 2.4 ilustra uma transmissão realizada com sucesso após detecção do canal ocioso.

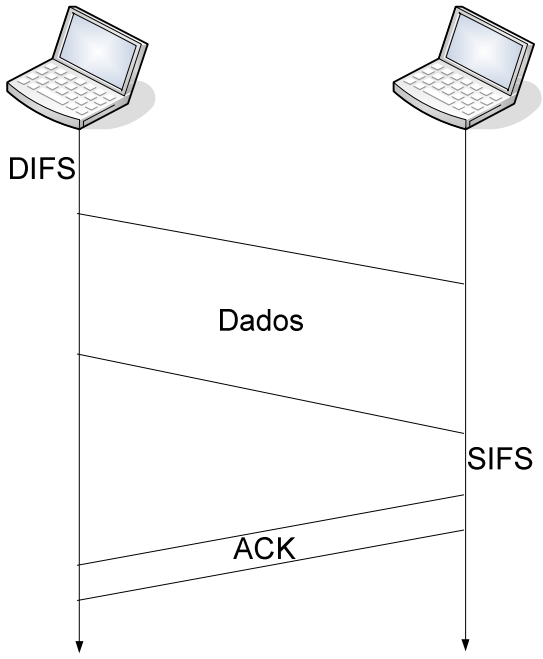


Figura 2.4 – *Situação (1): o meio está disponível (Modificado de [6]).*

Suponha agora o caso (2), ou seja, uma estação deseja transmitir um pacote de dados, porém, ela identifica que o canal de comunicação já está sendo utilizado. Nesse caso, a estação sem fio escolherá um valor aleatório de retardo, realizando, após o término do intervalo de tempo DIFS, uma contagem regressiva a partir daquele valor quando notar que o canal ficou liberado. Assim que o intervalo de tempo DIFS e a contagem de retardo terminar, ela transmite o pacote. A Figura 2.5 ilustra essa situação.

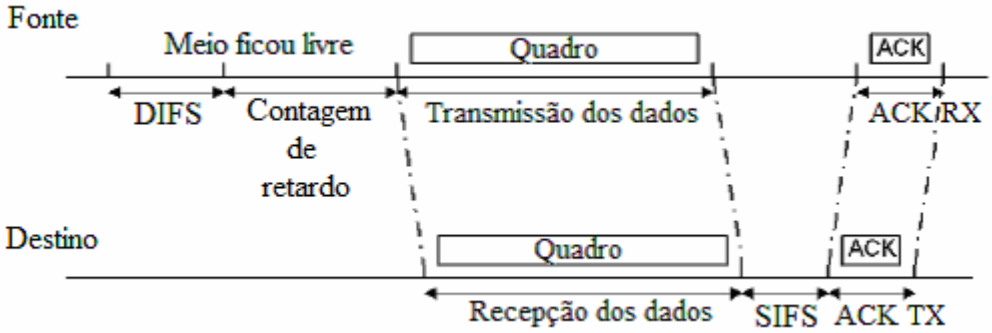


Figura 2.5 – *Situação (2): o meio estava ocupado com uma transmissão (Modificado de [12]).*

Resumindo o processo de transmissão de um pacote por uma estação 802.11, segue uma seqüência de quatro passos [6]:

- **passo 1:** se inicialmente a estação perceber que o canal está ocioso, ela transmitirá seu pacote após um curto período de tempo conhecido como DIFS;

- **passo 2:** caso contrário, a estação escolherá um valor aleatório de retardo e fará a contagem regressiva a partir desse valor assim que perceber que o canal estiver ocioso. Antes de proceder com essa contagem regressiva, a estação sem fio deve ainda aguardar por um intervalo de tempo DIFS. Dessa forma, mesmo o canal ficando livre para transmissões, a estação deverá aguardar por um intervalo de tempo total igual a soma do DIFS e do retardo antes de iniciar o envio do seu quadro. O valor do contador ficará estacionário enquanto o canal estiver sendo utilizado;
- **passo 3:** quando o contador chegar ao zero (isso só pode acontecer quando a estação percebe que o canal está livre), ela transmitirá o pacote inteiro e ficará aguardando um ACK;
- **passo 4:** se o transmissor receber um ACK, ele admitirá que o pacote de dados foi entregue com sucesso ao receptor. Se não receber um ACK dentro de um tempo limite, a estação transmissora entrará em fase de retardo, escolhendo um valor aleatório dentro de um intervalo ainda maior como conseqüência de uma possível colisão.

Analisando o passo 2, vemos que, diferentemente do protocolo CSMA/CD, no protocolo CSMA/CA a transmissão dos dados é atrasada. Isso ocorre em virtude do fato de que, mesmo percebendo que o canal ficou liberado, a estação sem fio deve realizar a contagem regressiva até o final antes de proceder a transmissão.

De acordo com o que foi explicado nos parágrafos anteriores, percebe-se que o protocolo CSMA/CA é mais cauteloso quando comparado ao protocolo CSMA/CD.

Reforçando essa diferença entre esses dois protocolos, considere um cenário com duas estações onde cada uma delas tem um pacote para ser transmitido [6]. Entretanto, nenhuma delas irá transmitir de imediato, pois percebem que uma terceira estação já está transmitindo. Em uma rede 802.3, ou seja, Ethernet e baseada no protocolo 802.3, cada uma das estações transmitiria tão logo detectassem que a terceira estação deixou de transmitir. Como é de se esperar, isto acarretaria em colisão, o que não é muito crítico no CSMA/CD, pois ambas as estações abortariam suas transmissões, evitando desperdício de largura de banda.

Porém, como já explicado anteriormente, a situação com o 802.11 é bem diferente. Em virtude do fato de que o padrão IEEE 802.11 não detecta colisões e nem aborta transmissões, um quadro que sofra colisão será transmitido integralmente. Assim, a meta do 802.11 é evitar colisões sempre que possível. Com o CSMA/CA, se duas estações perceberem que o canal está ocupado, ambas entrarão imediatamente em fase de retardo

aleatório, e, possivelmente, escolherão valores diferentes de retardo. Considerando que esses valores serão diferentes, assim que o canal ficar ocioso, as contagens regressivas se iniciarão e uma estação atingirá o zero antes da outra, iniciando sua transmissão de dados. A estação "perdedora" ouvirá o sinal da estação "vencedora", interromperá seu contador e não transmitirá até que a outra estação conclua sua transmissão. Dessa forma, evita-se uma colisão dispendiosa.

É claro que ainda podem ocorrer colisões em redes 802.11. Esse seria o caso se os valores de retardo aleatório escolhidos pelas estações forem iguais ou se as estações estiverem ocultas uma da outra.

2.3.1 O problema do terminal oculto: RTS e CTS

O protocolo MAC 802.11 inclui uma técnica inteligente e opcional de reserva de recursos de largura de banda que ajuda a evitar colisões, mesmo na presença de terminais ocultos. Para ilustrar o problema do terminal oculto, será analisada a situação apresentada na Figura 2.6, a qual apresenta duas estações sem fio e um ponto de acesso.

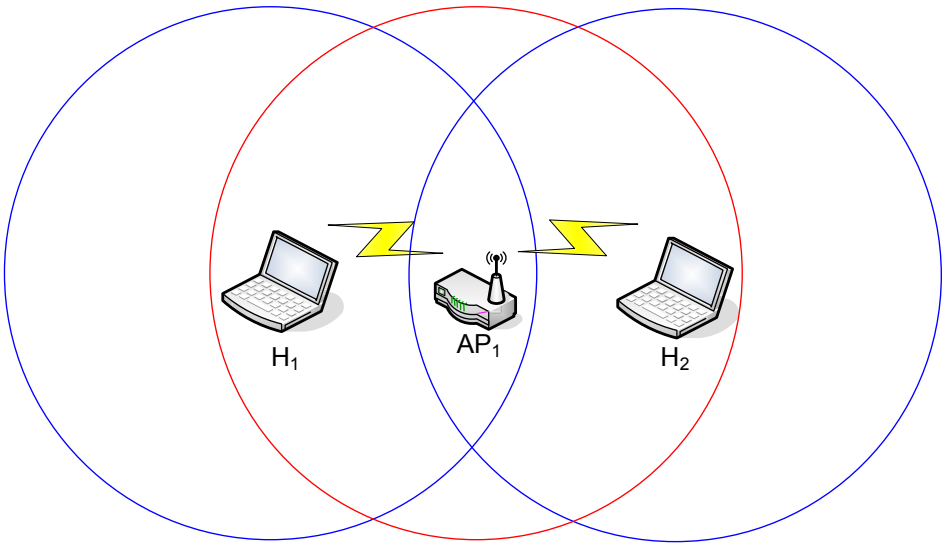


Figura 2.6 – *Exemplo de terminal oculto: H1 está oculto de H2 e vice-versa.*

Como pode ser observado pela Figura 2.6, ambas as estações sem fio estão dentro da área de cobertura do AP_1 (círculo vermelho) e ambas se associaram a ele. Entretanto, em virtude das condições de propagação do sinal e de desvanecimentos causados pelo meio de transmissão, as áreas de cobertura de cada estação sem fio está limitada à área dos círculos azuis. Com isso, vê-se que a área de cobertura de H_1 não alcança H_2, e vice-versa, ou seja, uma estação está oculta em relação a outra.

Para mostrar o por quê de terminais ocultos serem um problema, suponhamos a seguinte situação: a estação H_1 está transmitindo um quadro e, durante a sua transmissão, a camada de rede na estação H_2 passa um quadro de dados para a camada MAC 802.11. Pelo fato de a estação H_2 está fora da área de cobertura da estação H_1, ela não "escuta" a transmissão dessa última, e envia, após o intervalo de tempo DIFS, seu quadro de dados. Considerando que tanto H_1 quanto H_2 estão enviando quadros de dados para o AP_1, ocorrerá uma colisão nesse destino.

Para evitar esse problema, o protocolo IEEE 802.11 permite que uma estação utilize dois quadros curtos de controle, chamados de RTS (*Request To Send*) e CTS (*Clear To Send*). Esses quadros são utilizados para reservar o acesso ao canal compartilhado. Existem duas técnicas de transmissão que utilizam esses quadros, as quais são [13]:

- **MACA (*Multiple Access with Collision Avoidance*):** esta técnica consiste em fazer com que o transmissor estimule o receptor a liberar um quadro curto como saída, permitindo que estações vizinhas possam detectar essa transmissão e evitem transmitir dados.
- **MACAW (*MACA for Wireless*):** é uma otimização do MACA. Sem as confirmações da camada de enlace de dados, os quadros perdidos não eram transmitidos até que a camada de transporte percebesse sua ausência, bem mais tarde. Foi então introduzido um quadro ACK após cada quadro de dados bem sucedido. A Figura 2.7 ilustra todo o processo de transmissão de um quadro de dados utilizando-se os quadros curtos RTS, CTS e ACK.

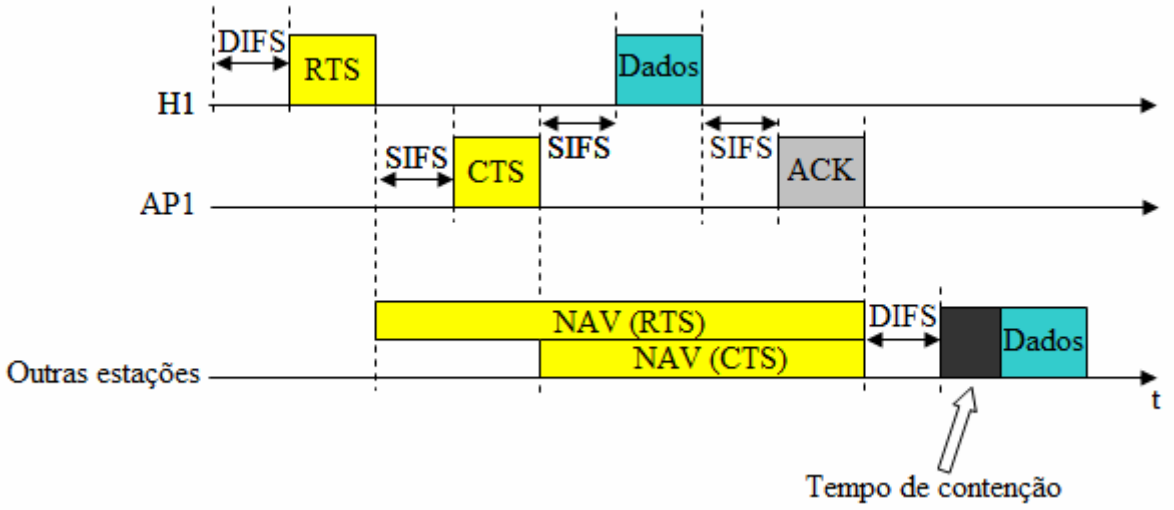


Figura 2.7 – *Prevenção de colisão utilizando-se os quadros RTS e CTS (Modificado de [14]).*

Conforme especificado pelo protocolo MAC 802.11, se a estação H_1 da Figura 2.6 tiver um quadro para transmitir, ela deverá primeiramente sondar o canal. Se o meio estiver ocioso, depois de esperar por um intervalo de tempo DIFS, H_1 poderá emitir um RTS, que não possui nenhuma prioridade em relação às outras mensagens. O pacote RTS inclui informações do tipo: o destinatário do pacote de dados a ser enviado, no caso AP_1, e o tempo previsto para a sua transmissão. Esse tempo previsto inclui o intervalo de tempo necessário para H_1 transmitir integralmente o quadro de dados propriamente dito mais o tempo necessário para ele receber o sinal ACK que será enviado por AP_1. Toda a estação que receber o sinal RTS tem que fixar o seu NAV (*Net Allocation Vector*) de acordo com a duração do tempo previsto especificado no RTS. O NAV serve para que outras estações sem fio fiquem cientes de que uma transmissão está para ocorrer e que elas devem permanecer em silêncio durante todo aquele tempo previsto no RTS [15].

Se o receptor da mensagem que o emissor quer enviar recebe o RTS, ele responde com um CTS, depois de esperar por um intervalo de tempo SIFS. Esse sinal CTS contém novamente o tempo previsto para transmissão da mensagem propriamente dita. Todas as estações que receberem o CTS do receptor têm também que ajustar seus NAV para evitarem transmissões durante o intervalo de tempo especificado nos quadros RTS e CTS. Vale salientar que o conjunto de estações que receberam o CTS não é necessariamente o mesmo conjunto de estações que receberam o RTS [15].

Finalmente, o transmissor pode enviar sua mensagem depois de um intervalo de tempo SIFS após ter recebido o CTS. O receptor recebe a mensagem, espera por SIFS, e envia o sinal ACK se a transmissão estiver correta. Agora, a transmissão está completa e o NAV em cada estação indica que o meio está inativo e o ciclo padrão pode recomeçar.

Assim, utilizando-se os quadros RTS e CTS, todas as estações dentro do raio de ação do transmissor e do receptor foram informadas que vão ter que aguardar mais tempo para tentar acessar o meio. Isso contorna o problema do terminal escondido.

É importante observar que a interferência (colisão) ocorre no destino. Se o destino dos dados enviados pela estação H_2 for o AP_2 ao invés do AP_1, conforme ilustrado na Figura 2.8, a transmissão ocorrerá sem colisões. Nessa Figura, vê-se que a estação H_2 está dentro da área de cobertura tanto de AP_1 (círculo vermelho) quanto de AP_2 (círculo preto) e, assim, H_2 pode se associar a qualquer um desses dois APs.

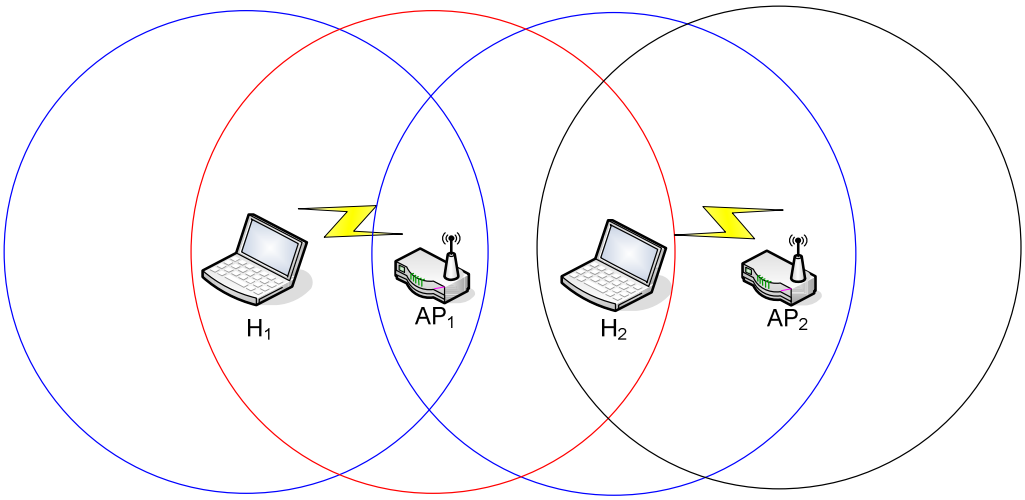


Figura 2.8 – *Transmissões simultâneas em uma WLAN IEEE 802.11.*

Em sistemas cabeados (com fio), todos os sinais se propagam para todas as estações e, portanto, somente uma transmissão pode ocorrer de cada vez em qualquer parte do sistema. Porém, em sistemas baseados em ondas de rádio de pequeno alcance, várias transmissões podem ocorrer simultaneamente, desde que todas essas transmissões obedeçam as duas seguintes condições [13]:

- primeira condição: ter destinos diferentes;
- segunda condição: esses destinos devem estar fora do alcance uns dos outros.

Dessa forma, podem ocorrer na Figura 2.8 duas transmissões simultâneas sem que ocorram colisões. Uma delas seria de H_1 para o AP_1 e a outra seria de H_2 para o AP_2. Conforme observado na Figura 2.8, os destinos dessas duas transmissões seriam diferentes (uma para o AP1 e outra para o AP2), e esses destinos estão fora de alcance um do outro.

2.3.2 O problema do terminal exposto

Um outro problema que existe na transmissão de dados em redes sem fio IEEE 802.11, além do problema do terminal oculto, é o problema do terminal exposto. A Figura 2.9 ilustra o problema do terminal exposto, lembrando que, da mesma forma que nas Figuras 2.6 e 2.8, o círculo vermelho representa o alcance do AP_1 e o círculo azul representa o alcance do H_2.

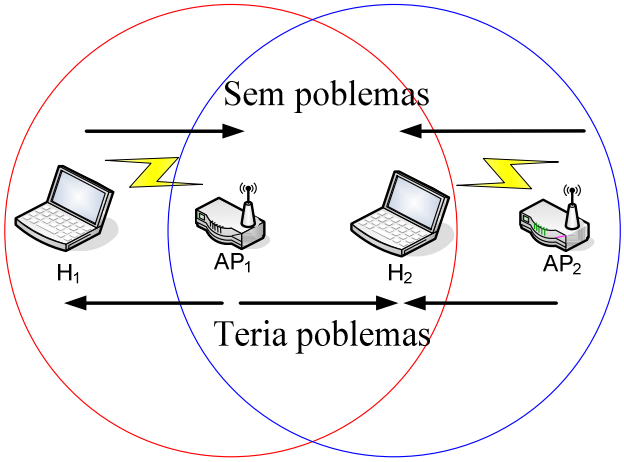


Figura 2.9 – *Problema da estação exposta.*

Na situação ilustrada na Figura 2.9, se H_2 detectar o meio físico, esse *host* ouvirá uma transmissão em andamento, que é a transmissão de AP_1 para H_1, e concluirá, erradamente, que não pode transmitir para o AP_2. Na verdade, essa transmissão de H_2 para o AP_2 só geraria uma recepção de má qualidade na zona entre AP_1 e H_2, onde nenhum dos receptores desejados está localizado.

Vamos considerar a situação onde o AP_2 está localizado na área entre AP_1 e H_2, conforme pode ser visto na Figura 2.10.

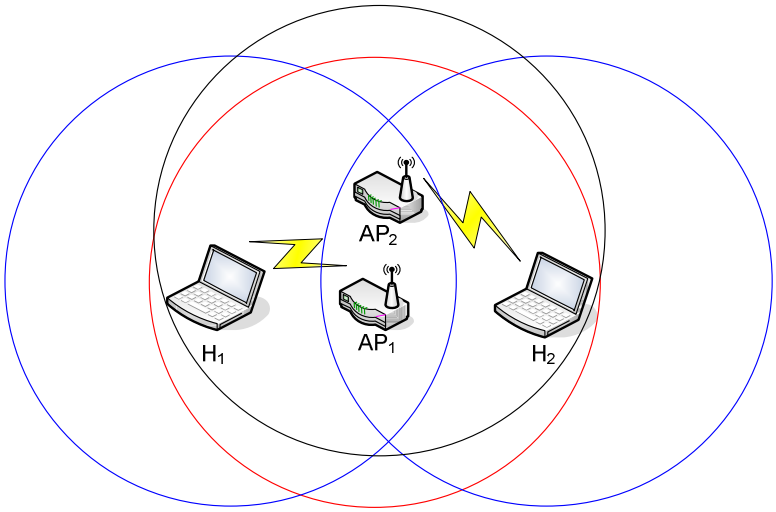


Figura 2.10 – *Recepção de má qualidade.*

Nesta situação, as transmissões simultâneas de AP_1 para H_1 e de H_2 para AP_2 causariam recepções de má qualidade pois, apesar de os destinatários de cada transmissão serem diferentes (um é o H_1 e o outro é o AP_2), eles estão dentro do alcance um do outro. O AP_2 está dentro da área de cobertura de H_1 (círculo azul mais a esquerda da Figura 2.10) e o H_1, por sua vez, está dentro da área de cobertura do AP_2 (círculo preto). Dessa forma, a primeira condição estabelecida anteriormente é respeitada (destinatários das transmissões

são diferentes), mas a segunda condição (os destinatários estarem fora do alcance um do outro) não é respeitada.

2.3.3 O uso do EIFS

Um importante parâmetro de tempo utilizado em redes *wireless* é o EIFS (*Extended Inter Frame Spacing*). De acordo com o protocolo MAC 802.11 baseado no CSMA/CA, [16] uma estação sem fio só pode transmitir um quadro se ela conseguir determinar que o meio de transmissão está fisicamente e virtualmente livre.

Uma estação sem fio pode determinar se o meio está fisicamente livre se sua placa de rede sem fio não detectar sinais cuja potência seja maior do que a sensibilidade do receptor. Caso contrário, o meio é tido como ocupado. Nesse caso, a estação deve receber um quadro (RTS ou CTS), independentemente dos endereços de origem e destino. Para o caso de redes cabeadas, uma estação só receberá um quadro se seu endereço estiver no campo de endereço de destino ou se for um quadro de *broadcast*; no caso de redes sem fio, como conseqüência das características do meio de transmissão, estações podem receber quadros independentemente dos endereços contidos nele.

Após o recebimento de um quadro RTS ou CTS, a estação deve conferir o campo CRC (*Cyclic Redundance Check*) para verificar se o quadro foi ou não corretamente recebido.

Se o CRC estiver correto, a estação sem fio pode extrair o cabeçalho MAC do quadro recebido. O cabeçalho MAC (*Media Access Control*) possui duas informações importantes [16]: o endereço MAC de destino e o campo de duração. Após verificar o campo de endereço MAC de destino do quadro, a estação pode ou descartá-lo, caso ele não seja destinado para a estação recebedora, ou repassá-lo para a camada superior [16]. O campo de duração (ou NAV) contém o tempo estimado para finalização da transmissão que está em andamento. Por meio da análise do NAV é que uma estação pode ter uma estimativa futura de quando o canal estará virtualmente livre.

Caso contrário, ou seja, se o campo CRC estiver incorreto por colisão ou nível de ruído inaceitável no sinal recebido, a estação sem fio não poderá extrair o cabeçalho MAC e o NAV, conseqüentemente, será desconhecido. É aqui que entra o papel do EIFS. Por causa do NAV desconhecido, a estação não poderá determinar exatamente quando o canal estará virtualmente livre. O EIFS é então utilizado como o pior caso de estimativa. O cálculo do EIFS é feito assumindo que o quadro incorreto pode ter sido corretamente

recebido por alguma outra estação, e esta estação responderá com ACK ou CTS utilizando a menor taxa de transmissão [16].

O EIFS também é aplicado quando uma estação transmite um quadro, mas não recebe uma confirmação. A falta da confirmação pode ter sido causada por uma colisão na recepção ou porque a BER (*Bit Error Rate*) está inaceitavelmente elevada. Em ambos os casos, o destinatário da transmissão não pôde receber um quadro válido, logo, não enviou uma confirmação. Frente a essa situação, a estação transmissora faz uso do EIFS como um *timeout*, após o qual o transmissor decide que o quadro enviado foi perdido e que uma retransmissão se faz necessária.

2.4 O quadro IEEE 802.11

O quadro 802.11 possui algumas semelhanças com o quadro Ethernet 802.3, mas também possui campos que são específicos para o uso em redes sem fio. Talvez, a diferença mais marcante no quadro 802.11 é que ele possui quatro campos de endereço. A Figura 2.11 ilustra o quadro MAC 802.11.

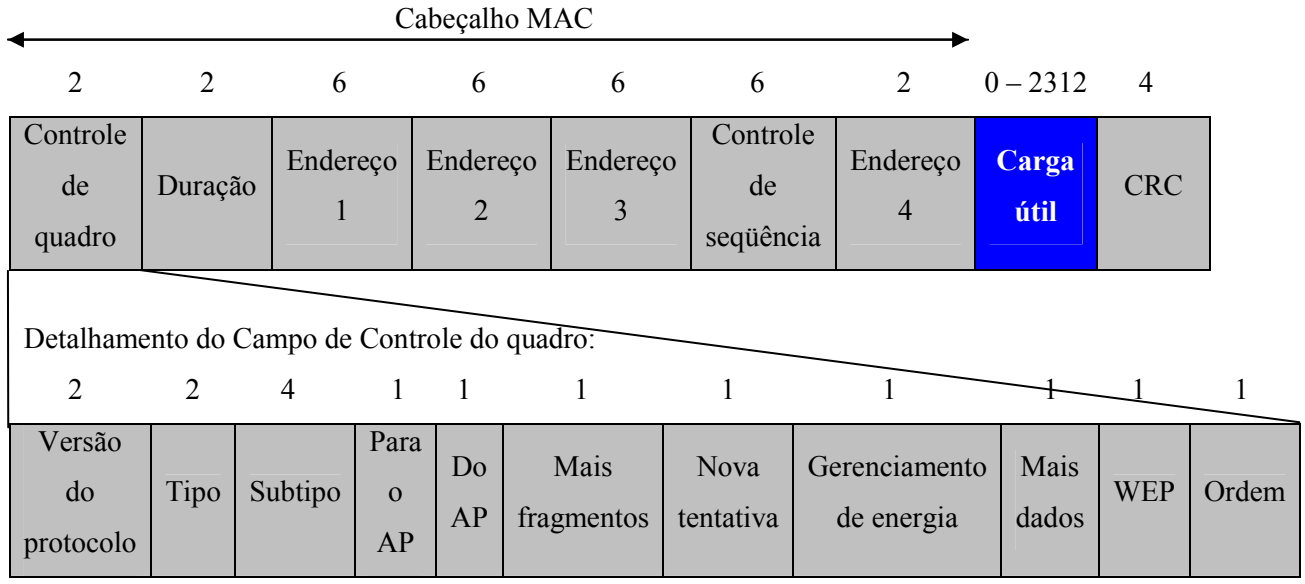


Figura 2.11 – *Quadro 802.11*
(Modificado de [6]).

Como pode ser visto na Figura 2.11, o quadro MAC 802.11 é composto por um cabeçalho MAC, o campo de carga útil e um campo de CRC. Os números em cima de cada campo representam o número de bytes de cada um. Está destacado também na Figura 2.11 o campo Controle de quadro.

A descrição de cada campo é apresentada a seguir [14].

- **Controle de quadro:** esse quadro contém informações de controle usadas para definir o tipo de quadro MAC 802.11. A estrutura desse quadro está ilustrada na Figura 2.11. Nele, temos os campos relacionados abaixo.
  - **Versão do protocolo:** indica a versão corrente do protocolo 802.11 utilizado. As estações receptoras usam esse campo para verificar se a versão do protocolo do quadro recebido é suportada.
  - **Tipo e Subtipo:** servem para determinar a função do quadro. Há três diferentes tipos de quadro: de controle, de dados e de gerenciamento. Existem ainda vários subtipos de quadro para cada um desses tipos, determinando uma função específica desempenhada.
  - **Para o AP e Do AP:** este campo serve para indicar se o quadro está indo para o AP ou se está vindo do AP. Esses campos só são utilizados em quadros do tipo dados de estações associadas a APs.
  - **Mais fragmentos:** indica se mais fragmentos do quadro de dados ou gerenciamento estão a caminho.
  - **Nova tentativa:** indica se a informação (dado ou gerenciamento) está ou não sendo retransmitida.
  - **Gerenciamento de energia:** indica se a estação que transmitiu a informação está em modo ativo ou em modo de economia de energia.
  - **Mais dados:** indica, para uma estação que está operando no modo de economia de energia, que o AP tem mais quadros para enviar. Esse campo é também usado pelo AP para indicar que quadros de *broadcast*/*multicast* serão enviados.
  - **WEP (*Wireless Equivalent Privacy*):** serve para indicar se está ou não sendo usado no quadro o processo de criptografia e autenticação.
  - **Ordem:** indica se todos os quadros recebidos devem ser processados em ordem.
- **Duração:** utilizado tanto em quadros de dados quanto em quadros de controle (RTS e CTS). O protocolo 802.11 permite que uma estação transmissora reserve o canal durante um período que inclui o tempo para transmitir seu quadro de dados e o tempo para transmitir um reconhecimento (quadro ACK) [6]. Esse valor de duração é incluído nesse campo [6].

- **Endereço 1:** é o endereço MAC da estação sem fio que deve receber o quadro. Dessa forma, se uma estação móvel sem fio transmitir o quadro, o endereço 1 armazenará o endereço MAC do AP de destino. De modo semelhante, se um AP transmitir o quadro, o endereço 1 conterá o endereço MAC da estação sem fio destinatária [6].
- **Endereço 2:** é o endereço MAC da estação que transmite o quadro. Se uma estação móvel transmitir um quadro, seu endereço MAC será inserido neste campo. Da mesma forma, se um AP transmitir um quadro, seu endereço MAC será inserido neste campo.
- **Endereço 3:** serve para realizar a interconexão do BSS (que consiste no AP e estações sem fio) com outros segmentos de rede, via alguma interface de roteador.
- **Endereço 4:** é utilizado apenas em redes ad-hoc, não em redes de infra-estrutura.
- **Carga útil:** contém a informação específica de dados ou de gerenciamento sendo transmitida.
- **CRC:** o transmissor do quadro aplica um CRC-32 sobre todos os campos do cabeçalho MAC e sobre a carga útil para gerar o CRC. O receptor do quadro se utiliza do mesmo CRC para determinar o seu próprio valor e então verificar se ocorreu ou não erro durante a transmissão.

Para compreender melhor a função do campo de endereço 3, considere o exemplo de interconexão de rede exemplificado pela Figura 2.12.

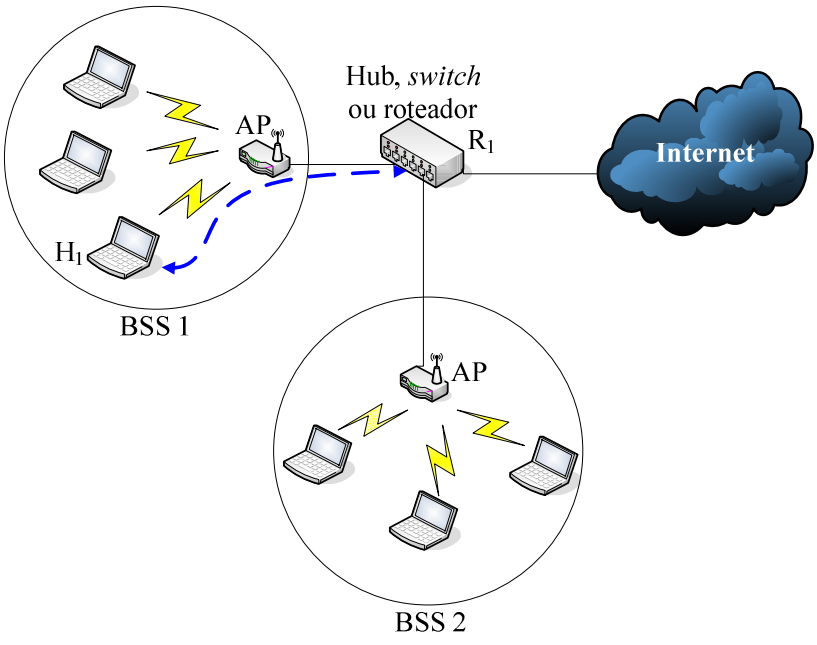


Figura 2.12 – *A utilização dos campos de endereço em quadros 802.11. Movendo um quadro entre H1 e R1.*

Na Figura 2.12, existem dois APs, cada um responsável por um certo número de estações sem fio. Cada um dos APs tem uma conexão direta com o roteador, que, por sua vez, se conecta com rede mundial de computadores, a Internet.

Vamos considerar agora a transmissão de um datagrama da interface de roteador R_1 até a estação sem fio H_1. O roteador não está ciente de que existe um AP entre ele e H_1, ou seja, da perspectiva do roteador, H_1 é apenas um hospedeiro em um dos segmentos de rede aos quais ele está conectado.

Primeiramente, o roteador fica sabendo o endereço IP de H_1 baseado no campo endereço IP de destino que consta no datagrama recebido. O roteador então utiliza um pacote ARP (*Address Resolution Protocol*) para determinar o endereço MAC de H_1, exatamente como aconteceria em uma LAN Ethernet comum. Ao determinar o endereço MAC de H_1, a interface de roteador R_1 encapsula o datagrama em um quadro Ethernet, no qual o campo de endereço de fonte contém o endereço MAC de R_1 e o campo de endereço de destino contém o endereço MAC de H_1.

Ao receber o quadro Ethernet, o AP converte o quadro IEEE 802.3 em um quadro IEEE 802.11 antes de enviá-lo pelo canal sem fio. Baseado nos campo de endereço vistos anteriormente, o AP preenche os campos de endereços 1, 2 e 3 da seguinte maneira: o campo de endereço 1 conterá o endereço MAC do destinatário, ou seja, o endereço MAC de H_1; o campo de endereço 2 armazenará o endereço MAC da origem, que será o endereço MAC do AP; e o campo de endereço 3 conterá o endereço MAC de R_1. Assim, com base nesse campo de endereço 3, H_1 pode determinar o endereço MAC da interface de roteador que enviou o datagrama para o seguimento de rede [6].

Considere agora o caminho inverso, ou seja, quando H_1 responde movendo um datagrama para R_1. A estacão móvel H_1 cria um quadro IEEE 802.11, preenchendo os campos de endereço 1 e 2 com o endereço MAC do AP e com seu próprio endereço MAC, respectivamente. O endereço 3 é preenchido por H_1 com o endereço MAC da interface de roteador R_1.

O AP, ao receber o quadro 802.11, converte-o para um quadro 802.3, onde o campo de endereço MAC de origem é o endereço MAC de H_1 e o campo de endereço MAC de destino é o endereço MAC de R_1. Assim, o campo de endereço 3 do quadro IEEE 802.11 permite que o AP determine o endereço MAC de destino apropriado ao construir o quadro Ethernet.

2.5 A especificação *Bluetooth* e diferenças com o padrão IEEE 802.15

O padrão IEEE 802.11, explanado anteriormente, visa estabelecer comunicação sem fio entre dispositivos separados por distâncias de até 100 metros [6] ou superiores, sem o uso de amplificadores. Redes de telefonia celular, por exemplo, visam estabelecer comunicação por dezenas ou centenas de quilômetros.

O padrão *Bluetooth* foi desenvolvido por um grupo de trabalho formado por cinco empresas líderes mundiais no ramo das telecomunicações (Ericsson, Nokia, IBM, Toshiba e Intel) [17]. Este grupo é conhecido como SIG (*Special Interest Group*). O trabalho conjunto desenvolvido pelo SIG permitiu a criação de um padrão aberto de comunicação sem fio de curto alcance, baixa potência e rádios de baixo custo, garantindo uma ampla aceitação e compatibilidade no mercado. O nome *Bluetooth* foi dado em homenagem a um rei Viking que unificou a Dinamarca e a Noruega [13, 18, 19].

Além dessas duas tecnologias de redes sem fio, existe um padrão de rede sem fio denominado WPAN, cuja tecnologia foi baseada na especificação do *Bluetooth*. Logo, essa especificação [19] serviu como base para o desenvolvimento do padrão IEEE 802.15 para WPANs, cuja finalidade [20] é promover interconectividade entre dispositivos pessoais separados por distâncias de até 10 metros.

No entanto, é importante observar que *Bluetooth* e 802.15 não são exatamente a mesma coisa. A especificação *Bluetooth* se refere a um sistema completo, ou seja, da camada física até a camada de aplicação. Por sua vez, o comitê do IEEE 802.15 padroniza apenas as camadas física e de enlace de dados, deixando o restante da pilha de protocolos fora do escopo de trabalho [13].

O *Bluetooth* é uma tecnologia de rede sem fio de baixa potência, curto alcance e baixa velocidade, criado para interconectar *notebooks*, equipamentos periféricos, telefones celulares e PDAs. Já o padrão IEEE 802.11 é uma tecnologia de acesso sem fio de potência e velocidade mais altas e médio alcance. Ambas as tecnologias de rede *Bluetooth* e IEEE 802.11 (extensões b e g) operam na faixa de freqüências denominada ISM (2,4 GHz). Como visto na Tabela 2.1, a extensão 802.11n opera nas faixas de freqüência de 2,4 GHz e de 5 GHz.

O *Bluetooth* foi projetado para operar em um ambiente com muitos usuários, no qual até oito dispositivos ativos podem se conectar em uma pequena rede, denominada de *piconet*. Dez dessas *piconets* podem coexistir na mesma faixa de cobertura de rádio

Bluetooth [21], e para oferecer segurança, cada enlace é codificado e protegido contra espionagem e interferência.

O *Bluetooth* provê suporte para conectividade sem fio em três áreas de aplicações gerais, são elas [21]:

- **pontos de acesso de voz e dados.** O *Bluetooth* facilita as transmissões de voz e dados em tempo real, fornecendo fácil conexão sem fio entre dispositivos de comunicação portáteis e estacionários;
- **substituição de cabos.** O *Bluetooth* elimina a necessidade de ligações cabeadas. As conexões são instantâneas e mantidas mesmo quando os dispositivos não estão em linha de visada;
- **rede *ad-hoc*.** Um dispositivo *Bluetooth* pode estabelecer comunicação com outro dispositivo *Bluetooth* assim que este último entrar na área de cobertura do primeiro. E não será necessária uma infra-estrutura de rede sem fio com um ponto de acesso.

2.5.1 Arquitetura de protocolo

O *Bluetooth* é estruturado como uma arquitetura de vários protocolos dispostos em camadas. Esses protocolos podem ser subdivididos em três grupos [21]: os protocolos básicos, os protocolos de substituição de cabos e controle de telefonia e os protocolos adotados.

Os protocolos básicos formam uma pilha de cinco camadas [13, 19, 21]:

- **rádio.** Essa camada especifica detalhes da interface aérea, incluindo freqüência de operação, esquema de modulação, a técnica de transmissão e a sua potência;
- **banda básica.** Nessa camada são definidos o estabelecimento da conexão dentro de uma *piconet*, ou seja, como os dispositivos *Bluetooth* procuram e se conectam uns com os outros, o endereçamento, o formato do pacote, a temporização, o controle de energia, o tipo de pacote, os procedimentos para o processamento dos pacotes, as estratégias para detecção e correção de erros, a criptografia e a transmissão e retransmissão de pacotes. Os papéis de mestre e escravo que um dispositivo pode assumir e a seqüência de saltos em freqüência que será utilizada também são definidos aqui;
- **protocolo gerenciador de enlace (LMP – *Link Management Protocol*).** Responsável pela configuração de enlace entre dispositivos *Bluetooth* e pelo gerenciamento de enlace em andamento. A configuração de enlace inclui aspectos de alocação de largura de banda para dados em geral, de reserva de banda para tráfego de

áudio, de segurança (criptografia e autenticação) além do controle e da negociação de tamanhos de pacotes;

- **controle de enlace lógico e protocolo de adaptação (L2CAP - *Logical Link Control and Adaptation Protocol*).** Responsável pela adaptação de protocolos de camadas superiores à camada de banda básica. O L2CAP suporta a multiplexação de vários protocolos de camada superior. Esta propriedade permite múltiplos protocolos e aplicações compartilhem a interface aérea. O L2CAP também é responsável pela fragmentação e remontagem de pacotes, fornecendo serviços não-orientados à conexão e orientados à conexão. Em resumo, [13] o L2CAP "esconde" das camadas superiores os detalhes da transmissão, executando uma função similar à subcamada LLC (*Logical Link Control*) do padrão 802.3;
- **programa de descoberta de serviço (SDP – *Service Discover Program*).** Permite que os dispositivos *Bluetooth* consultem uns aos outros sobre as informações do dispositivo, os serviços disponíveis por cada um e as características desses serviços. Com isto, o SDP tem o objetivo de tornar possível o estabelecimento de uma conexão entre dois ou mais dispositivos *Bluetooth*.

O protocolo de substituição de cabo incluído na especificação *Bluetooth* é o RFCOMM (*RF Communication*) [21]. O RFCOMM apresenta uma porta serial virtual que foi projetada para tornar a substituição das tecnologias de cabo o mais transparentes possível.

As portas seriais são um dos tipos mais comuns de interfaces de comunicação usadas em dispositivos de computação de comunicação. O protocolo RFCOMM permite a substituição dos cabos de portas seriais com o mínimo de modificações nos dispositivos existentes, fornecendo transporte de dados binários e emulando os sinais de controle RS-232 sobre a camada de banda básica do *Bluetooth*. O RS-232 é um padrão de interface de porta serial amplamente usado.

O protocolo de controle de telefonia especificado pelo *Bluetooth* é o TCS BIN (*Telephone Control Specification – Binary*). Esse protocolo é baseado em bits e define a sinalização de controle para o estabelecimento de chamadas tanto de voz quanto de dados entre dispositivos *Bluetooth*. Além disso, o TCS BIN define procedimentos de gerenciamento de mobilidade para manipular dispositivos *Bluetooth*.

Os protocolos adotados são definidos em especificações emitidas por outras organizações criadoras de padrões, sendo incorporados na arquitetura geral do *Bluetooth*.

A estratégia do padrão IEEE 802.15 é de criar apenas protocolos necessários e usufruir, sempre que possível, de padrões existentes [21]. Por esse motivo, os protocolos listados abaixo são classificados no padrão como protocolos adotados:

- **PPP (*Point-to-Point Protocol*).** O PPP é um protocolo padrão da Internet para transportar datagramas IP por meio de um enlace ponto-a-ponto. Este protocolo é bastante empregado na camada de enlace para conexões discadas para a Internet;
- **TCP (*Transmission Control Protocol*) / UDP (*User Datagram Protocol*) / IP (*Internet Protocol*).** Esses são protocolos básicos da pilha de protocolos TCP/IP;
- **OBEX (*Object Exchange*).** O protocolo de troca de objeto é um protocolo que atua em nível de sessão e que foi desenvolvido pela *Infrared Data Association* (IrDA). Ele define uma relação cliente-servidor para a movimentação de dados. O OBEX fornece funcionalidade semelhante à do protocolo HTTP (*Hipertext Transfer Protocol*), porém, de maneira mais simples;
- **WAE (*Wireless Application Environment*) / WAP (*Wireless Application Protocol*).** O *Bluetooth* incorpora o ambiente de aplicação sem fio e o protocolo de aplicação sem fio em sua arquitetura.

O padrão *Bluetooth* possui muitos protocolos agrupados em camadas. A estrutura de camadas não segue nenhum dos modelos conhecidos (OSI, TCP/IP, 802). Entretanto, o IEEE vem trabalhando para modificar o *Bluetooth* de modo que ele se adapte melhor ao modelo 802 [13]. A estrutura básica das camadas *Bluetooth*, da forma como foi modificada pelo comitê do 802, é ilustrada na Figura 2.13.

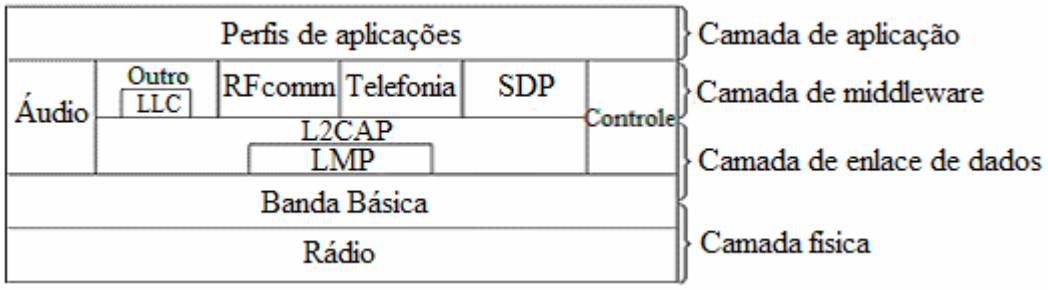


Figura 2.13 – *Versão IEEE 802.15 da pilha de protocolos Bluetooth (Modificado de [13]).*

Os protocolos de áudio e controle, como os próprios nomes sugerem, tratam questões de controle e áudio. As aplicações podem chegar a esses protocolos diretamente, sem a necessidade de passar pelo protocolo L2CAP [13].

O protocolo LLC foi inserido pelo IEEE por questões de compatibilidade com suas outras redes 802.11. O protocolo de telefonia é um protocolo de tempo real utilizado para

os três perfis destinados a voz (esses perfis serão descritos a seguir). Ele também gerencia o estabelecimento e a finalização de chamadas.

A camada de nível mais alto é onde se localizam as aplicações e os perfis, os quais fazem uso dos protocolos das camadas inferiores para realizarem seu trabalho. Cada aplicação possui seu conjunto específico de protocolos, ou seja, um *mouse Bluetooth*, por exemplo, possui apenas os protocolos específicos para a sua finalidade.

2.5.2 Perfis de aplicações Bluetooth

Diferentemente da maioria dos protocolos de redes de computadores, a especificação *Bluetooth* determina treze aplicações específicas para serem suportadas, cada qual com sua pilha de protocolos. O padrão 802.11, por exemplo, não especifica diferentes pilhas de protocolos dependendo do que o usuário for fazer com seu *laptop* móvel (se ele o utilizará para ler *e-mails*, para navegar na Internet ou qualquer outra atividade).

As treze aplicações *Bluetooth*, as quais são chamadas de perfis *Bluetooth*, estão listadas na Tabela 2.4.

Tabela 2.4 – *Os perfis Bluetooth.*

Nome	Descrição
Acesso geral	Procedimentos para gerenciar enlaces
Descoberta de serviços	Protocolo para descoberta de serviços ofertados (Protocolo SDP)
Porta serial	Substituição de um cabo de porta serial (Protocolo RFCOMM)
Troca geral de objetos	Define a relação cliente-servidor para troca de objetos (Protocolo OBEX)
Acesso de LAN	Protocolo entre um *laptop* móvel e uma LAN fixa
Discagem de rede	Permite um *laptop* fazer uma chamada via um telefone móvel
Fax	Permite que um Fax móvel se comunique com um telefone móvel
Telefone sem fio	Conecta um telefone sem fio com sua base
Intercom	Walkie-Talkie digital
Headset	Permite uma comunicação de voz sem necessidade do uso das mãos
Object push	Provê um meio para troca de objetos simples
Transferência de arquivos	Provê uma opção mais geral de transferência de arquivos
Sincronismo	Permite que um PDA sincronize com um outro computador

Na verdade, o perfil de acesso geral não é uma aplicação, mas sim uma base sobre a qual as aplicações são desenvolvidas. O principal objetivo desse perfil [13] é estabelecer e manter um canal seguro entre o dispositivo mestre e os escravos. O perfil de descoberta de serviços (protocolo SDP), conforme já descrito anteriormente, é utilizado pelos dispositivos para descobrir quais serviços outros dispositivos têm a oferecer. É um requisito que todos os dispositivos *Bluetooth* implementem esses dois perfis. O restante é opcional.

O perfil de porta serial é um protocolo de transporte que a maioria dos outros perfis utiliza. O protocolo responsável por emular uma porta serial é o RFCOMM.

O perfil de troca geral de objetos (que faz uso do protocolo OBEX) define a relação cliente-servidor para o transporte de dados. O dispositivo cliente é que dá início à operação de troca de dados e um escravo pode fazer o papel tanto de cliente quanto de servidor [13].

O próximo grupo de três perfis é para conexões de rede. O perfil de acesso de LAN permite a um dispositivo *Bluetooth* se conectar a uma rede fixa. O perfil de discagem de rede permite que um *laptop* se conecte a um telefone móvel sem necessidade de uso de fios. O perfil fax permite a um fax sem fio enviar e receber faxes utilizando um telefone móvel sem a necessidade de fios entre ambos [13].

Os próximos três perfis são para telefonia. O perfil de telefone sem fio provê um meio de conectar o telefone sem fio à sua estação base. O perfil intercom permite que dois telefones sem fio se conectem como *walkie-takies*. O perfil *headset* permite uma conexão de voz entre o *headset* e a estação base para, por exemplo, fazer uma ligação enquanto dirige um carro [13].

Os últimos três perfis são para a troca de objetos entre dois dispositivos sem fio. Esses objetos podem ser fotos ou arquivos de dados, por exemplo.

2.5.3 Piconets e Scatternets

A célula básica de uma rede *Bluetooth* é a *piconet*. A especificação *Bluetooth* [19] define uma *piconet* como um *cluster ad-hoc* formado de maneira espontânea por dispositivos *Bluetooth*. Em uma *piconet*, um dispositivo assume um papel de mestre e o restante assume papel de escravo. No máximo, sete escravos podem estar ativos em um dado momento na *piconet*. Se existem mais de sete escravos, o restante deve estar no estado de baixa potência (*park* – estacionado). Nesse estado, pode haver até 255 nós na *piconet* [13]. Os possíveis estados nos quais um dispositivo *Bluetooth* pode estar serão detalhados mais adiante.

Quando dois dispositivos *Bluetooth* entram no alcance de comunicação um do outro, eles tentarão se comunicar. Se nenhuma *piconet* está disponível no momento, dar-se-á início a um processo de negociação. Nesse processo, um dispositivo é designado como mestre, sendo este responsável pela determinação da seqüência de saltos em freqüência e da temporização que deverão ser utilizados pelo dispositivo escravo. O mestre [19] também é responsável por instruir um escravo a mudar de um estado para outro em períodos de inatividade.

Um dispositivo *Bluetooth* pode estar em um dos estados ilustrados na Figura 2.14.

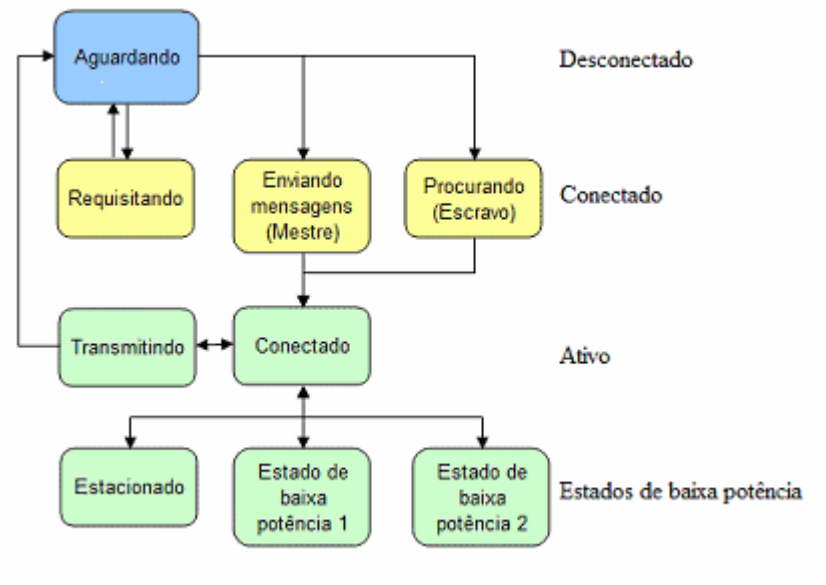


Figura 2.14 – *Estados de conexão Bluetooth*
(Modificado de [19]).

Um dispositivo está no modo aguardando quando ele está ligado mas não está associado ainda a nenhuma *piconet*. Ele entra no estado de requisição quando envia requisições para outros dispositivos com os quais pode se conectar. Um mestre em uma dada *piconet* pode está no estado de envio de mensagens, no qual ele pode convidar outros dispositivos para sua *piconet* [19].

Quando uma comunicação é estabelecida com sucesso entre o mestre e um novo dispositivo, esse assume o papel de escravo e entra no modo conectado. Neste modo, o escravo recebe um endereço ativo. Enquanto conectado, um escravo pode transmitir dados quando o mestre determinar. Durante sua transmissão, o escravo está no modo de transmissão e, no final dela, ele retorna ao modo conectado [19].

O estado de baixa potência 1 é um estado de baixo consumo de potência no qual o escravo “dorme” durante um número pré-determinado de *slots* de tempo. O dispositivo

escravo então "acorda" para transmitir no *slot* de tempo determinado e então retorna novamente ao mesmo estado. O estado de baixa potência 2 é outro de baixa potência no qual o escravo está inativo por uma quantidade de tempo pré-determinada. Diferentemente do primeiro estado de baixa potência, no segundo não há transferência de dados [19].

Quando o dispositivo escravo não tem dados nem para transmitir e nem para receber, o dispositivo mestre pode instruí-lo a entrar no estado estacionado. Ao entrar nesse estado, o escravo libera seu endereço ativo para que o mestre possa alocá-lo para um escravo que ele acabou de reativar do modo estacionado [19].

Dentro de uma mesma *piconet*, um dispositivo escravo só pode se comunicar com o dispositivo mestre e quando este último autorizar. É importante mencionar que um mesmo dispositivo pode fazer parte de uma *piconet* como escravo e de outra *piconet* como mestre. Todavia, enquanto que um dispositivo *Bluetooth* pode ser escravo em várias *piconets* que formam uma rede espalhada ou *scatternet*, ele poderá ser mestre em apenas uma das *piconets* [22]. Uma *scatternet* é uma coleção de *piconets* interconectadas e sobrepostas. A Figura 2.15 ilustra uma *scatternet*.

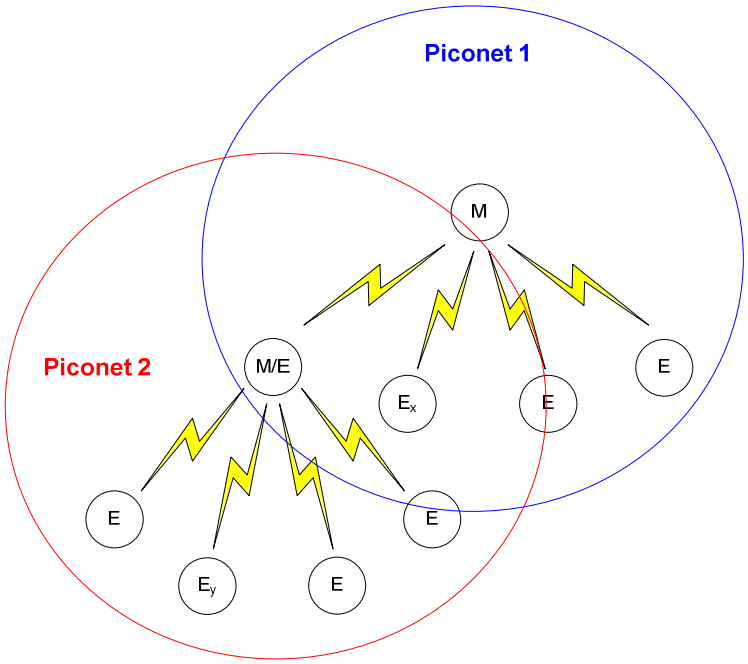


Figura 2.15 – *Relações Mestre/Escravo em uma scatternet.*

Ao fazer parte de uma *scatternet*, um dispositivo *Bluetooth* só pode transmitir e receber em uma *piconet* de cada vez das quais ele faz parte. Dessa forma, sua participação em múltiplas *piconets* tem que ser baseada em uma multiplexação no tempo (TDM – *Time Division Multiplexing*) [22]. Dispositivos *Bluetooth* que fazem parte de diversas *piconets* são chamados de nós de interconexão. Esses nós podem operar como *gateways* entre

piconets, encaminhando o tráfego entre elas. A Figura 2.16 ilustra um cenário onde se tem uma *scatternet* composta por três *piconets*.

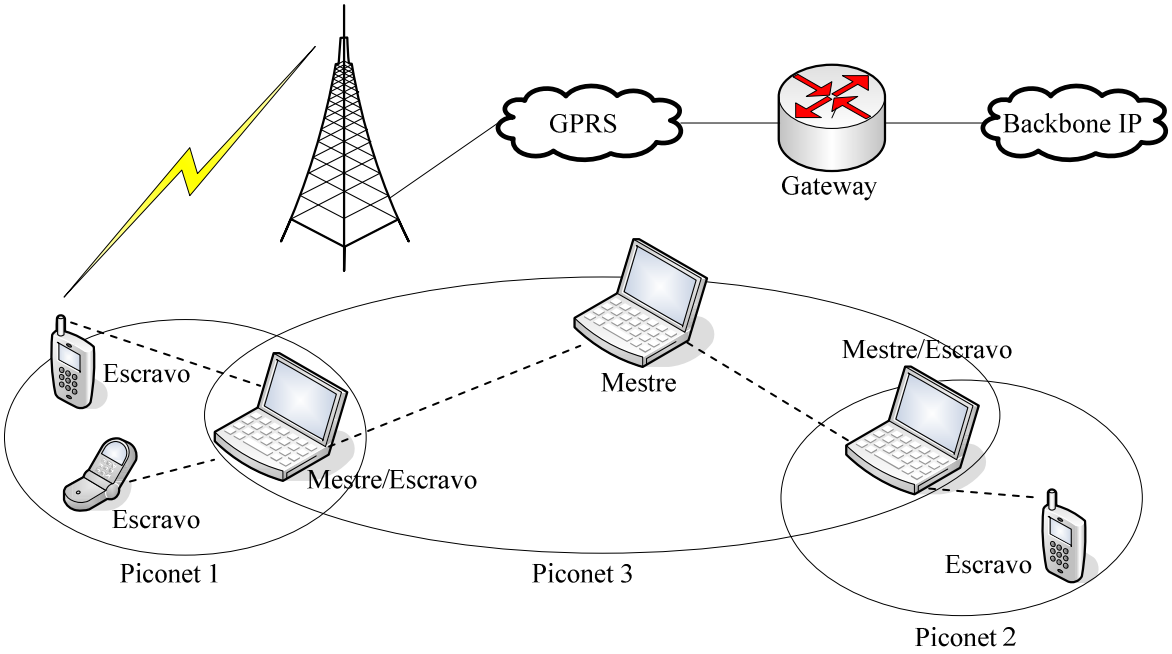


Figura 2.16 – *O mestre da piconet 1 com função de gateway (Modificado de [22]).*

No cenário ilustrado na Figura 2.16, o mestre da *piconet* 1 encaminha pacotes do telefone celular com acesso GPRS (*General Packet Radio Service*) para todos os *laptops*, fornecendo, assim, acesso à Internet a toda *scatternet*. Desse modo, o mestre da *piconet* 1 atua como um *gateway* e como um nó de interconexão entre a *piconet* 1 e a *piconet* 3; o mestre da *piconet* 2 atua como um nó de interconexão entre essa *piconet* e a *piconet* 3.

Um sistema *Bluetooth* usa um esquema de saltos em freqüência denominado FHSS, no qual cada freqüência (canal) tem uma largura de banda de 1 MHz. Como são utilizados até 79 canais [21], a largura de banda total disponível é de 79 MHz.

No padrão IEEE 802.15, um canal lógico é definido pela seqüência de saltos em freqüência. Diferentes canais lógicos, ou seja, diferentes seqüências de salteamento, podem compartilhar simultaneamente a mesma largura de banda de 79 MHz. Colisões ocorrerão quando dispositivos em diferentes *piconets*, ou seja, comandados por mestres diferentes, como é o caso de E_x e E_y da Figura 2.15, usarem a mesma freqüência de salto ao mesmo tempo. À medida que se aumenta o número de *piconets* que formam uma *scatternet* em uma determinada área, o número de colisões aumenta e o desempenho cai. Assim, a área física e a largura de banda total de 79 MHz são compartilhadas por *scatternet*; o canal lógico (seqüência de saltos) e a transferência de dados são compartilhados por *piconet* [21].

Capítulo 3. A CAMADA FÍSICA DO PADRÃO IEEE 802.11g

O padrão inicial IEEE 802.11 especificava a taxa de dados de 1 Mbps e 2 Mbps para três diferentes camadas físicas. Essas camadas são baseadas em DSSS (*Direct Sequence Spread Spectrum*), FHSS e IR (*Infrared*) [23]. Ambas as técnicas de transmissão DSSS e FHSS são especificadas na banda ISM de 2,4 GHz. A camada física baseada na técnica DSSS [23] é a mais empregada, pois, dentre as três, é a que provê maior taxa de transmissão.

O padrão IEEE 802.11b provê taxas de transmissão de até 11 Mbps na banda ISM de 2,4 GHz. Essa taxa de transmissão é atingida utilizando-se a técnica DSSS com modulação CCK ou com o algoritmo PBCC (*Packet Binary Convolutional Coding*). Esse algoritmo [23] foi oficialmente ratificado pelo IEEE como uma alternativa ao CCK.

O padrão IEEE 802.11a especifica uma camada física que faz uso da técnica de multiplexação OFDM. Com OFDM, redes baseadas no padrão IEEE 802.11a provêem taxas de transmissão que podem ir de 6 Mbps até 54 Mbps na banda U-NII de 5 GHz. Esse padrão representou um significativo aumento na taxa de transmissão de dados nas WLANs. Todavia, por operarem em uma faixa de freqüência diferente, os dispositivos IEEE 802.11a são incompatíveis com os dispositivos 802.11b.

A fusão dos padrões IEEE 802.11a e 802.11b veio com a publicação do padrão IEEE 802.11g, cuja camada física também implementa a multiplexação OFDM. Esse padrão combina a taxa de transmissão do 802.11a e a freqüência de operação do 802.11b. As redes sem fio 802.11g transmitem a taxas de 54 Mbps na banda ISM de 2,4 GHz [16], combinação essa que representa uma vantagem em relação aos padrões IEEE 802.11a e 802.11b. Um fato importante é que o padrão IEEE 802.11g opera na faixa ISM de 2,4 GHz que é compatível com o padrão IEEE 802.11b [16].

Em virtude desse aumento substancial na taxa de dados oferecido pelo padrão IEEE 802.11g na faixa de freqüência ISM [24], aplicações em tempo real de áudio e vídeo sob demanda se tornaram uma realidade nas redes sem fio. As novas funcionalidades trazidas pelo padrão IEEE 802.11g são [23]:

- a implementação de quatro opções de camadas físicas diferentes;
- o suporte obrigatório ao preâmbulo curto;
- o atributo de rede ERP (*Extended Rate Physicals*);
- os mecanismos de proteção que tratam questões de interoperabilidade;

- e o mecanismo CTS-*to-self*.

O padrão IEEE 802.11g disponibiliza oito modos de transmissão dependendo do tipo de modulação e da taxa de codificação utilizada. Esses modos estão listados na Tabela 3.1.

Tabela 3.1 – *Modos de transmissão IEEE 802.11g.*

Modo	Modulação	Taxa de codificação	Taxa de transmissão nominal
1	BPSK	1/2	6 Mbps
2	BPSK	3/4	9 Mbps
3	QPSK	1/2	12 Mbps
4	QPSK	3/4	18 Mbps
5	16-QAM	1/2	24 Mbps
6	16-QAM	3/4	36 Mbps
7	64-QAM	2/3	48 Mbps
8	64-QAM	3/4	54 Mbps

Quanto mais sofisticada for a técnica de modulação utilizada, maior será a taxa de transmissão atingida. Porém, para taxas de transmissão mais elevadas, o sinal se torna mais sensível a ruídos, interferências e propagação multi-percurso. O modo de transmissão mais resistente a erros é o primeiro (6 Mbps); enquanto que o oitavo (54 Mbps) é o modo mais susceptível a degradações.

3.1 Camadas físicas definidas no padrão IEEE 802.11g

Diferentemente do padrão IEEE 802.11b, que utiliza apenas a tecnologia DSSS, o padrão IEEE 802.11g pode fazer uso do DSSS, do OFDM ou de ambos, constituindo um sistema híbrido [23]. O que torna possível o uso desse sistema híbrido é a disponibilização de quatro opções de camadas físicas distintas.

No padrão IEEE 802.11g, essas camadas são definidas como ERP e elas coexistem durante uma transmissão em andamento. O transmissor e o receptor podem optar por utilizar uma dessas quatro opções de camadas e ambos devem suportar a escolha feita.

As quatro opções de camadas físicas disponíveis no padrão IEEE 802.11g são [23]:

- **ERP-DSSS/CCK.** Essa é a antiga camada física utilizada pelo padrão IEEE 802.11b, a qual provê taxas de até 11 Mbps;

- **ERP-OFDM.** Essa é uma nova opção de camada física introduzida pelo padrão IEEE 802.11g. A técnica de multiplexação OFDM é utilizada para prover as taxas de transmissão do padrão IEEE 802.11a só que na banda ISM de 2,4GHz. Essa opção de camada física realiza a transmissão de dados no meio sem fio dividindo um sinal serial de informação de taxa elevada em vários sub-sinais de taxa mais baixa, os quais são transmitidos pelo sistema simultaneamente e em diferentes freqüências.
- **ERP-DSSS/PBCC.** Essa opção de camada física foi introduzida pelo padrão IEEE 802.11b e trata-se de uma alternativa à modulação CCK. A taxa de transmissão disponibilizada por esta camada é a mesma da camada ERP-DSSS/CCK. O padrão IEEE 802.11g estendeu as taxas de dados disponíveis no 802.11b, adicionando 22 Mbps e 33 Mbps;
- **DSSS-OFDM.** Essa é uma nova opção de camada física que se constitui de um sistema híbrido, combinando as técnicas DSSS e OFDM. O cabeçalho da camada física do pacote é transmitido usando DSSS e a carga útil, OFDM. Essa opção de camada física é importante para tratar questões de interoperabilidade.

Todo dispositivo de rede sem fio 802.11g deve dar suporte obrigatório às camadas físicas ERP-DSSS/CCK e ERP-OFDM. A implementação das duas outras camadas é opcional.

A Tabela 3.2 apresenta as taxas de transmissão de dados suportadas para cada uma das quatro camadas físicas.

Tabela 3.2 – *Taxas de transmissão para as quatro camadas físicas IEEE 802.11g.*

Camada física – PHY (*Physical*)	**Taxa de dados (Mbps)**
ERP-DSSS/CCK (obrigatória)	1, 2, 5,5, 11
ERP-OFDM (obrigatória)	6, 9, 12, 18, 24, 36, 48, 54
ERP-DSSS/PBCC (opcional)	1, 2, 5,5, 11, 22, 33
DSSS-OFDM (opcional)	6, 9, 12, 18, 24, 36, 48, 54

Analisando a Tabela 3.2, observa-se que as camadas físicas capazes de atingir a maior taxa de dados do padrão (54Mbps) são as que implementam a técnica de multiplexação OFDM.

A camada MAC se comunica com a subcamada PLCP (*Physical Layer Convergence Protocol*) por meio de instruções (conhecidas como primitivas) via pontos de acesso de serviço, chamados de SAP (*Service Access Points*). Quando a camada MAC

envia uma instrução à subcamada PLCP, a primeira prepara uma MPDU (MAC *Protocol Data Unit*) para transmissão [25]. A Figura 3.1 ilustra a arquitetura IEEE 802.11g das camadas física e MAC.

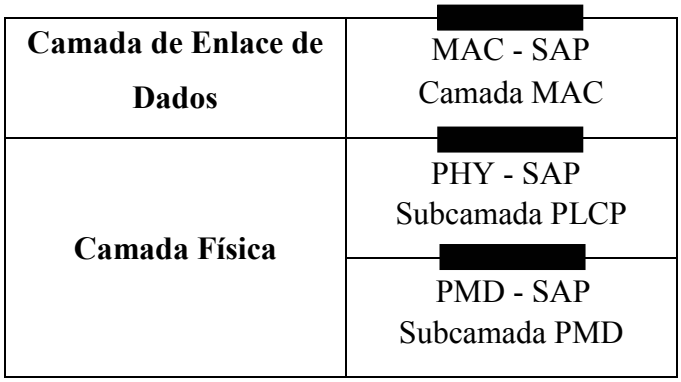


Figura 3.1 – *Arquitetura 802.11g das camadas MAC e física (Modificado de [25]).*

A função da subcamada PLCP é minimizar a dependência da camada MAC em relação à subcamada PMD (*Physical Medium Dependent*). Para isso, a subcamada PLCP faz um mapeamento das MPDUs para um formato de quadro que seja passível de transmissão pela subcamada PMD [25]. A subcamada PLCP também entrega à camada MAC quadros provenientes do meio de transmissão.

A subcamada PLCP anexa à MPDU um preâmbulo e um cabeçalho específicos de camada física, os quais contêm informações necessárias paras as camadas físicas tanto do transmissor quanto do receptor. O preâmbulo PLCP habilita o receptor a fazer o sincronismo com o sinal que está chegando antes de receber o conteúdo do quadro [16, 23, 25] e o cabeçalho PLCP contém informações sobre o quadro relacionadas à camada física [23, 25].

Logo abaixo da subcamada PLCP, a subcamada PMD provê a transmissão e recepção de quadros de dados entre as camadas físicas de duas estações via um meio sem fio. Para dar suporte a esse serviço, a subcamada PMD faz uma interface direta com o meio sem fio, realizando os processos de modulação e demodulação dos quadros transmitidos e recebidos [25]. Em sua operação, a subcamada PMD transforma uma representação binária de um PPDU (PLCP *Protocol Data Unit*) em um sinal de rádio adequado para transmissão. Do mesmo modo que ocorre entre a camada MAC e a subcamada PLCP, a comunicação entre essa última e a subcamada PMD se dá via primitivas através de um SAP.

O grupo de trabalho do padrão IEEE 802.11b percebeu que o preâmbulo PLCP era muito longo e que adicionava uma sobrecarga considerável em um sistema WLAN. Com o

objetivo de melhorar o desempenho do sistema e reduzir a sobrecarga do pacote, foi introduzida uma opção de suporte a um preâmbulo mais compacto. Se ambos o transmissor e o receptor darem suporte a essa opção, a transmissão deve ser realizada utilizando o preâmbulo mais curto. [23] O padrão IEEE 802.11g determina o uso obrigatório da opção de preâmbulo mais curto.

A Tabela 3.3 resume o atraso e o comprimento para ambas as opções de preâmbulo (longo e curto) para cada uma das quatro camadas físicas do padrão IEEE 802.11g.

Tabela 3.3 – *Parâmetros de atraso e comprimento para as opções de preâmbulo.*

Camada física	**Preâmbulo PLCP + cabeçalho PLCP (atraso)**		**Preâmbulo PLCP + cabeçalho PLCP (comprimento)**	
	Longo	**Curto**	**Longo**	**Curto**
ERP-DSSS/CCK	192 µs	96 µs	192 bits	120 bits
ERP-OFDM	20 µs		40 bits	
ERP-DSSS/PBCC	192 µs	96 µs	192 bits	120 bits
DSSS-OFDM	192 µs	96 µs	192 bits	120 bits

Quando o preâmbulo e o cabeçalho PLCP são transmitidos utilizando DSSS (isso ocorre para todas as camadas físicas, com exceção da ERP-OFDM), ambas as opções de preâmbulo longo e curto são definidas. Para a camada ERP-OFDM, só há um tipo de preâmbulo e cabeçalho, [23] cujo formato é quase idêntico ao do padrão IEEE 802.11a.

3.1.1 A estrutura de quadro da camada física ERP-OFDM

O padrão IEEE 802.11 se refere ao quadro formado pelo MPDU adicionado com o preâmbulo e cabeçalho PLCP como PPDU. A MPDU [25] também é conhecida como PSDU (PLCP *Service Data Unit*).

A Figura 3.2 ilustra o quadro da camada física ERP-OFDM PPDU, [25] que é o mais implementado no padrão IEEE 802.11g. Esse quadro dá suporte as taxas de dados de 6, 9, 12, 18, 24, 36, 48 e 54 Mbps.

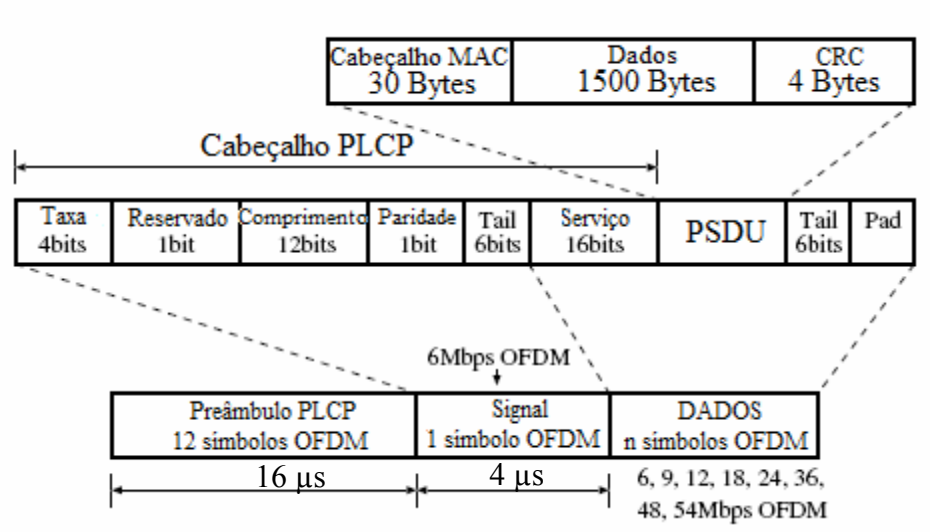


Figura 3.2 – *Estrutura de quadro ERP-OFDM PPDU (Modificado de [26]).*

Conforme observado na Figura 3.2, o quadro ERP-OFDM PPDU é formado por três partes [25]: preâmbulo PLCP, cabeçalho PLCP e dados.

O preâmbulo PLCP é formado por duas partes: o campo SYNC (*Synchronization*) e o campo SDF (*Start Frame Delimiter*). O campo SYNC consiste em uma seqüência de 0s ou 1s, os quais alertam o receptor para um sinal a ser recebido. O receptor iniciará o sincronismo com o sinal que está recebendo após detectar o SYNC. É possível que um receptor não receba todo o campo SYNC, mas apenas uma parte dele. Como o campo SYNC é formado por uma seqüência de 0s ou 1s, não importa em que parte da seqüência o receptor vai "notar" que está recebendo um sinal SYNC, desde que ele consiga fazer o sincronismo antes de receber o sinal SDF. O campo SDF define o início de um quadro. O padrão de bits para esse campo é sempre 1111001110100000 para o caso do uso do preâmbulo longo. Para o caso do preâmbulo curto, o padrão é 0000010111001111 [25]. Como pode ser observado na Figura 3.2, o preâmbulo ERP-OFDM leva 16 µs para "despertar" o receptor sobre um quadro de dados que será recebido.

O cabeçalho PLCP é formado por quatro bits de taxa, um bit reservado, doze bits para comprimento, um bit de paridade, seis bits de *tail* e dezesseis bits de serviço. O campo *signal* é formado por um símbolo OFDM (totalizando 24 bits) e tem as mesmas informações do cabeçalho PLCP, exceto pelo campo serviço. O campo *signal* é sempre transmitido a uma taxa de 6 Mbps, utilizando a modulação BPSK [25].

Os quatro bits destinados ao campo taxa do cabeçalho PLCP servem para indicar o tipo de modulação e a taxa de codificação utilizada no restante do PPDU, iniciando logo

após o campo *signal*. A Tabela 3.4 ilustra o tipo de modulação, a taxa de codificação e a taxa de transmissão de dados em função do valor assumido pelo campo taxa do cabeçalho PLCP.

Tabela 3.4 – *Valores assumidos pelo campo taxa do cabeçalho PLCP.*

Campo – Taxa	**Taxa de transmissão – campo Dados**
1101	6 Mbps
1111	9 Mbps
0101	12 Mbps
0111	18 Mbps
1001	24 Mbps
1011	36 Mbps
0001	48 Mbps
0011	54 Mbps

O campo reservado (1 bit) é estabelecido como zero, pois ele atualmente não é utilizado. O campo comprimento (12 bits) indica o número de octetos dentro do PSDU que a camada MAC está solicitando para a camada física transmitir. O campo paridade é formado por um bit para verificação de paridade par. Essa verificação é baseada nos primeiros dezessete bits, ou seja, nos bits que constam nos campos taxa, reservado e comprimento. O campo *tail* é formado por seis bits, os quais são sempre zero.

O campo de dados é formado pelos campos serviço, PSDU, *tail* e *pad*. O campo serviço consiste em dezesseis bits, dos quais os sete primeiros são fixados em zero para fins de sincronismo do receptor e os nove restantes são reservados para uso futuro, sendo estes também fixados em zero [25]. Como o campo serviço faz parte do campo dados, ele é transmitido na taxa especificada pelo campo taxa do cabeçalho PLCP, o mesmo ocorrendo para os campos PSDU, *tail* e *pad*.

O PSDU constitui efetivamente as informações (dados) que foram enviadas pela camada MAC para transmissão pelo meio sem fio. O PSDU pode ser visualizado mais detalhadamente na Figura 2.11, já sendo previamente detalhado.

O campo *tail* é formado por seis bits iguais a zero, os quais são necessários para retornar o codificador convolucional para o estado zero. O campo *Pad* contém pelo menos seis bits, porém ele é formado por uma quantidade de bits que faça com que o campo de dados seja um múltiplo do número de bits codificados em um símbolo OFDM.

3.1.2 A estrutura de quadro da camada física DSSS-OFDM

O padrão IEEE 802.11g especificou um tipo de PPDU que consiste em um preâmbulo e cabeçalho PLCP transmitidos com DSSS e um PSDU transmitido com OFDM. Esse novo PPDU é chamado como DSSS-OFDM. Relembrando que a opção de camada física DSSS-OFDM permite o uso do preâmbulo PLCP longo ou curto, as Figuras 3.3 e 3.4 ilustram respectivamente a estrutura de quadro com preâmbulo longo e com preâmbulo curto.

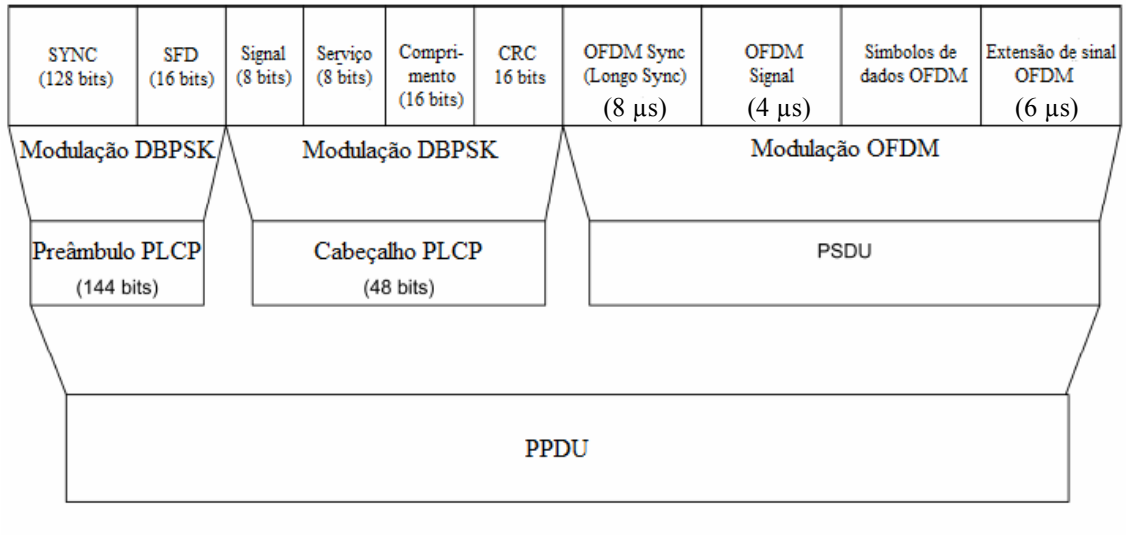


Figura 3.3 – *Estrutura de quadro DSSS-OFDM PPDU para preâmbulo longo (Modificado de [25]).*

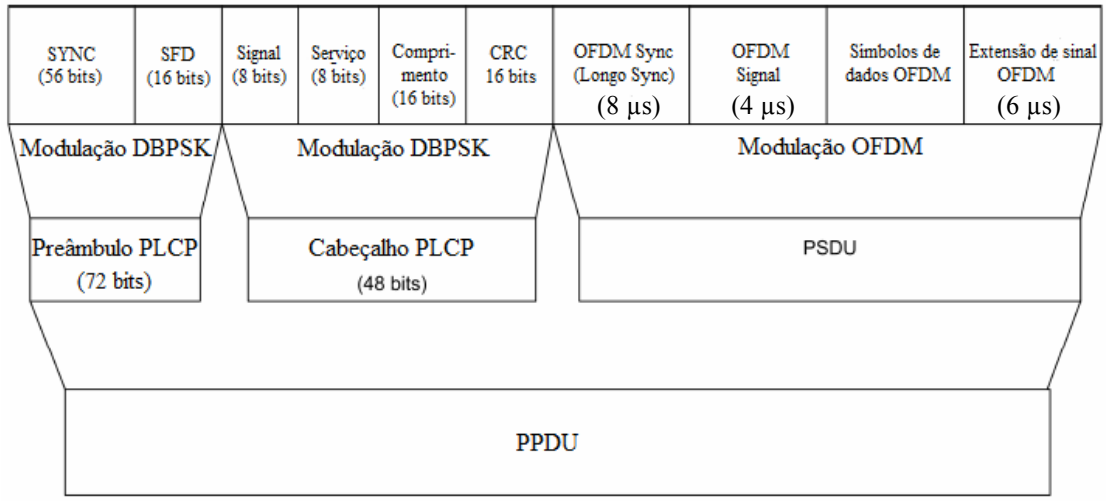


Figura 3.4– *Estrutura de quadro DSSS-OFDM PPDU para preâmbulo curto (Modificado de [25]).*

Comparando-se as Figuras 3.3 e 3.4, verifica-se que a única diferença entre os quadros é o tamanho do preâmbulo PLCP. O restante do quadro é idêntico.

A partir de agora, será detalhada a porção PSDU do quadro DSSS-OFDM PPDU. Conforme pode ser visualizado nas Figuras 3.3 e 3.4, o PSDU é composto por quatro seções principais, a saber: o campo OFDM *Sync*, o OFDM *Signal*, dados OFDM e extensão de sinal OFDM.

O campo OFDM *Sync* é utilizado pelo demodulador OFDM para que este obtenha os parâmetros do receptor. O campo OFDM *Signal* provê ao demodulador informações sobre a taxa de transmissão e o comprimento das informações que constam no campo símbolos de dados OFDM. O campo OFDM *Signal* para o quadro DSSS-OFDM [25] é idêntico ao campo *signal* encontrado no cabeçalho do quadro ERP-OFDM.

Após o campo OFDM *Signal*, vem a seção de dados da porção PSDU. O campo de símbolos de dados OFDM é modulado de forma a se obter as seguintes taxas de transmissão: 6, 9, 12, 18, 24, 36, 48 ou 54 Mbps, as quais são as mesmas taxas de transmissão possíveis para o campo de dados do quadro ERP-OFDM.

Logo em seguida ao campo de dados OFDM, o PSDU do quadro DSSS-OFDM anexa um sinal de extensão de modo a prover um tempo de processamento adicional para o demodulador OFDM. Esse sinal de extensão DSSS-OFDM é um período de 6 μs durante o qual não há transmissão de dados [25]. O tempo adicional inserido permitirá ao demodulador OFDM finalizar a decodificação convolucional dos símbolos de dados recebidos, possibilitando o envio de um reconhecimento ao transmissor após o intervalo de tempo SIFS.

3.2 O atributo de rede ERP

Para redes sem fio 802.11b, os valores padrões de *slot* de tempo e da janela mínima de contenção são, respectivamente, iguais a 20 μs e 31 *slots* [23]. Como o padrão IEEE 802.11g tem compatibilidade com o padrão IEEE 802.11b, o primeiro adotou também esses valores em todas as suas quatro opções de camadas físicas. Esses parâmetros são ajustados de forma a maximizar o desempenho de transmissões DSSS para taxa de dados de até 11 Mbps com preâmbulo PLCP longo (de 192 μs) ou curto (de 96 μs).

Porém, quando estações sem fio transmitem à taxa de dados ERP-OFDM (de 6 a 54 Mbps) com preâmbulo significativamente curto (de 20 μs), os valores anteriormente citados de *slot* de tempo e janela mínima de contenção degradam o desempenho da rede. Nesse caso, os valores mais apropriados para esses parâmetros são 9 μs e 15 *slots* de tempo respectivamente, os quais são os valores definidos para o padrão IEEE 802.11a [23].

O padrão IEEE 802.11g possibilita um ajuste dinâmico do *slot* de tempo e da janela mínima de contenção por meio da definição de um atributo de rede ERP. Esse atributo é uma *flag* publicada para as estações sem fio via um quadro *beacon*, o qual se trata de um quadro de controle contendo informações da rede. O atributo ERP é habilitado se todas as estações associadas a um AP em uma WLAN suportam as taxas de dados ERP-OFDM. Se esse for o caso, os valores de *slot* de tempo e janela mínima de contenção vão depender do modo de operação da WLAN [23].

Para o modo de operação BSS, se o atributo ERP está habilitado, o parâmetro de *slot* de tempo é igual a 9 μs, a janela mínima de contenção é igual a 15 *slots* de tempo e todas as trocas de quadros são efetuadas utilizando as taxas de dados ERP-OFDM. O valor da janela mínima de contenção pode ser ajustado para 15 *slots* de tempo mesmo se o atributo ERP estiver desabilitado, desde que o AP suporte as taxas de transmissão ERP-OFDM.

Para o modo de operação IBSS, se o atributo ERP estiver habilitado, o valor da janela mínima de contenção é ajustado para 15 *slots* de tempo e todos os quadros são trocados utilizando-se as taxas de dados ERP-OFDM. O valor do *slot* de tempo é sempre ajustado para 20 μs.

3.3 Aspectos de interoperabilidade e mecanismos de proteção

Em uma WLAN IEEE 802.11g, as estações sem fio podem escolher uma dentre quatorze diferentes taxas de transmissão e uma dentre quatro camadas físicas disponíveis de modo a transmitir um pacote da maneira mais eficiente. Todavia, essas amplas possibilidades de taxas de transmissão e camadas físicas dão margem a questões de interoperabilidade, podendo coexistir, em uma mesma WLAN IEEE 802.11g, diferentes tipos de estações sem fio, a saber [23]:

- **estações ERP.** São aquelas que suportam a camada física ERP-OFDM. Essas estações são equipadas com uma placa de rede sem fio IEEE 802.11g;
- **estações não-ERP e que suportam o preâmbulo PLCP curto.** São aquelas equipadas com uma placa de rede sem fio IEEE 802.11b de versão mais atual e que suportam taxas de dados de até 11 Mbps. Seu *firmware* pode ser atualizado para dar suporte ao preâmbulo PLCP curto;

- **estações não-ERP e que não suportam a opção de preâmbulo PLCP curto.** São aquelas equipadas com uma placa de rede sem fio IEEE 802.11b de versão mais antiga e que não dão suporte a opção de preâmbulo curto PLCP.

Como conseqüência dessa diversidade, surgem, no ambiente *wireless* 802.11g, diferentes combinações de comunicação. Essas combinações estão contempladas na Tabela 3.5 com seus respectivos parâmetros de camada física.

Tabela 3.5 – *Parâmetros da camada física para diferentes cenários de comunicação.*

Comunicação	**Preâmbulo**	***Slot* de tempo**	**Janela mínima de contenção (em *slots*)**
ERP para ERP	ERP-OFDM	9 μs	15
ERP para não-ERP/C	Curto	20 μs	31
ERP para não-ERP/L	Longo	20 μs	31
não-ERP/C para não-ERP/C	Curto	20 μs	31
não-ERP/L para não-ERP/L	Longo	20 μs	31
não-ERP/C para não-ERP/L	Longo	20 μs	31

Na Tabela 3.5, uma estação não-ERP/C significa uma estação sem fio equipada com placa de rede 802.11b que dá suporte ao preâmbulo curto. Já uma estação não-ERP/L faz alusão a uma estação sem fio equipada com uma placa de rede 802.11b que não dá suporte ao preâmbulo curto, mas apenas ao preâmbulo longo.

Retornando agora para a questão da interoperabilidade, considere uma WLAN composta por estações sem fio ERP e não-ERP. Deve-se ter em mente que estações ERP são aquelas estações IEEE 802.11g que dão suporte às quatro opções de camadas físicas já explicitadas anteriormente; estações não-ERP são estações IEEE 802.11b, as quais transmitem utilizando apenas a tecnologia DSSS. Estações ERP [23] se comunicam entre si utilizando quadros ERP-OFDM (Figura 3.2); todavia, estações não-ERP não são capazes de detectar uma transmissão OFDM. Assim, se uma estação ERP transmite, o meio de comunicação é tido como livre para as estações não-ERP e qualquer tentativa de transmissão feita por essas últimas resultará em colisão.

A primeira solução proposta pelo padrão IEEE 802.11g é o uso da camada física DSSS-OFDM. Com essa camada, todas as estações estarão habilitadas para detectar as transmissões do preâmbulo e do cabeçalho PLCP, pois ambos se dão com a técnica DSSS.

Detectando essas transmissões, uma estação não-ERP poderá reter sua transmissão, mesmo que ela não detecte o envio da carga útil, que se dá utilizando-se a técnica OFDM.

A segunda solução proposta pelo padrão é o uso dos quadros de controle RTS/CTS para proteger os quadros ODFM transmitidos. De acordo com o padrão IEEE 802.11g, quando estações ERP e não-ERP coexistem em uma WLAN, todos os quadros RTS e CTS devem ser transmitidos usando a camada física ERP-DSSS [23]. Dessa forma, todas as estações são informadas sobre as transmissões que estão a caminho, mesmo que o quadro de dados seja transmitido com o uso do OFDM, ou seja, com o uso da camada física ERP-OFDM.

Os quadros de controle RTS e CTS já foram apresentados anteriormente neste estudo (seção 2.3.1). Além deles, o padrão [23] IEEE 802.11g define um modo alternativo de mecanismo de proteção chamado de CTS-to-*self*. Esse mecanismo tem como objetivo evitar colisões em virtude do problema de interoperabilidade DSSS/OFDM.

3.3.1 O mecanismo CTS-to-self

A Figura 3.5 ilustra o mecanismo de proteção CTS-to-*self*.

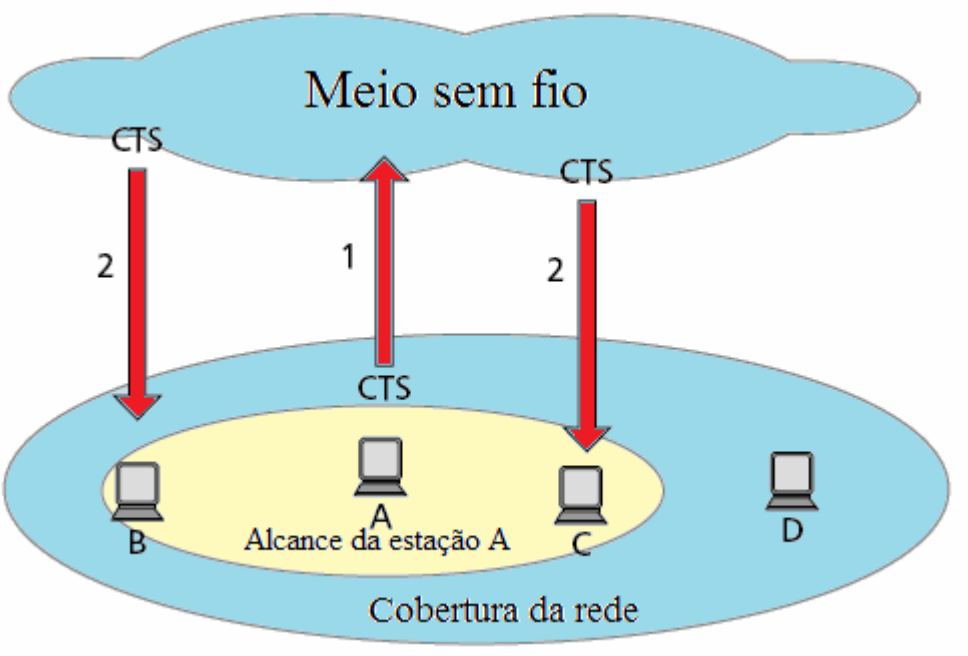


Figura 3.5 – *O mecanismo de proteção CTS-to-self (Modificado de [23]).*

Na Figura 3.5, quando a estação A tem um quadro para transmitir para a estação C, ela primeiro envia um quadro CTS (seta 1), o qual será recebido por ambas as estações B e

C (setas com o número 2). Essas, por sua vez, evitarão transmitir em virtude do recebimento do quadro CTS. Entretanto, a estação D, que está fora da área de cobertura de A, não receberá o quadro CTS e, conseqüentemente, não detectará uma transmissão de A. Assim, a estação D causará uma colisão se ela decidi transmitir. Dessa forma, o mecanismo CTS-to-*self* só é capaz de prevenir colisões acidentais, ou seja, colisões provocadas quando duas ou mais estações iniciam uma transmissão no mesmo *slot* de tempo. Esse mecanismo não é capaz de prevenir colisões causadas pelo problema do terminal oculto. Tendo em vista essa limitação, o mecanismo CTS-to-*self* só deve ser utilizado quando todas as estações presentes na rede sem fio são capazes de detectar as transmissões umas das outras. Em outros casos, o mecanismo RTS/CTS deve ser aplicado.

Capítulo 4. A CAMADA FÍSICA DO *BLUETOOTH*

Em julho de 1999, o grupo de trabalho SIG do *Bluetooth* criou uma especificação composta por 1.500 páginas. Logo após, o grupo de trabalho do IEEE, que pesquisava sobre as WPANs 802.15, adotou a especificação *Bluetooth* como base e começou a trabalhar a partir das proposições desse documento. O fato do IEEE tomar como referência a especificação do *Bluetooth* acabou por ajudar a promover o uso dessa tecnologia. Apesar das versões *Bluetooth* do SIG e 802.15 do IEEE não serem idênticas, é esperado que elas convirjam para um padrão único [13].

4.1 A camada de rádio *Bluetooth*

O *Bluetooth* faz uso em sua interface de rádio de uma técnica de transmissão denominada FHSS e [20] uma técnica de modulação chamada GFSK (*Gaussian Frequency Shift Keying*). Utilizando o FHSS, a transmissão realizada por um dispositivo é distribuída por vários canais diferentes, obedecendo a uma seqüência de saltos pseudo-aleatório que é definida pelo dispositivo mestre [21]. Dessa forma, com o uso do FHSS, a transmissão é espalhada sobre o espectro de freqüência ao longo do tempo [6].

O algoritmo para gerar a seqüência de saltos do FHSS funciona da seguinte maneira [12]: dada uma janela contígua com 32 freqüências dentro do espectro do *Bluetooth*, uma seqüência de salto aleatória é escolhida para essas 32 freqüências. Uma vez que todas as freqüências dentro dessa janela foram visitas uma única vez, uma nova janela com 32 freqüências é selecionada. Essa nova janela inclui 16 freqüências previamente visitadas e 16 novas freqüências.

A camada de rádio move os bits do mestre para o escravo ou vice-versa em uma banda de freqüência que é dividida em 79 canais de 1 MHz cada. A faixa de freqüência utilizada pelo *Bluetooth* tem início e fim, respectivamente, [27] em torno das freqüências de 2,402 a 2,4835 GHz, que corresponde a banda ISM.

Com a modulação GFSK, que provê 1 bit por Hz, alcança-se uma taxa de transmissão por canal de 1 Mbps [13]. Nesse esquema de modulação, um bit 1 é representado por um desvio de freqüência positivo e um bit 0 por um desvio de freqüência negativo. Em um modulador GFSK, tudo é igual a um modulador FSK, [28] exceto que antes de os pulsos em banda básica entrarem no modulador FSK, eles passam por um filtro

Gaussiano para suavizá-los, limitando sua largura espectral. Filtros Gaussianos são uma das formas padrão para reduzir a largura espectral de sinais.

A partir da versão 2 da norma do *Bluetooth*, [27] são definidos dois modos de modulação: o modo em taxa básica (*Basic Rate*) e o modo EDR (*Enhanced Data Rate*). O modo básico utiliza a modulação GFSK; o modo EDR, que é apenas definido na versão 2, utiliza uma técnica M-ária DPSK (*Differential Phase Shift Keying*), que permite taxas de dados mais elevadas, tais como 2 Mbps, para M = 4, e 3 Mbps, para M = 8. A tabela 4.1 relaciona a versão do *Bluetooth* com a taxa de transmissão.

Tabela 4.1 – *Taxas de transmissão Bluetooth.*

Versão	Taxa de transmissão
1.2	1 Mbps
2.0 + EDR	2 Mbps – 3 Mbps
3.0 (Em desenvolvimento)	53 – 480 Mbps (Proposto)

A taxa de saltos aplicada com o FHSS é de 1.600 saltos por segundo (que geram *slots* de tempo de 625 µs) e todos os nós em uma *piconet* saltam simultaneamente, com o mestre ditando a seqüência de salto. Essa taxa de saltos é aplicada quando o dispositivo *Bluetooth* está no estado conectado. Quando o dispositivo se encontra nos estados requisitando ou enviando mensagens [19], a taxa de saltos é de 3.200 saltos por segundo, com um *slot* de tempo de 312,5 µs.

No que diz respeito ao alcance de uma determinada *piconet Bluetooth*, isso dependerá da classe dos dispositivos *Bluetooth* utilizados. A tabela 4.2 relaciona as classes de dispositivos com suas respectivas potências de transmissão e alcance.

Tabela 4.2 – *Classes de dispositivos Bluetooth x potência de transmissão em função do alcance.*

Classe	Potência Máxima (mW/dBm)	Alcance aproximado
1	100 mW (20 dBm)	100 metros
2	2,5 mW (4 dBm)	10 metros
3	1 mW (0 dBm)	1 metro

Quando comparada com o alcance de *piconets Bluetooth* formadas exclusivamente por dispositivos de classe 2, a cobertura de *piconets Bluetooth* compostas por dispositivos de classe 1 e 2 pode ser expandida. Isso acontece em virtude da alta sensibilidade e

potência de transmissão do equipamento *Bluetooth* de classe 1. A elevada potência de transmissão do dispositivo de classe 1 permite a recepção do sinal pelo dispositivo de classe 2 a distâncias maiores. Por sua vez, a alta sensibilidade do dispositivo de classe 1 torna possível a recepção da baixa potência de transmissão do dispositivo de classe 2, mesmo a grandes distâncias [29].

4.2 A camada de banda básica *Bluetooth*

A camada de banda básica do *Bluetooth* é a que se aproxima mais da subcamada MAC. Entretanto, ela inclui alguns elementos da camada física. A camada de banda básica transforma uma seqüência de bits em quadros, definindo alguns formatos.

A temporização (definida nessa camada) diz respeito à quando um dispositivo escravo pode transmitir. O *Bluetooth* utiliza a técnica TDD (*Time Division Duplexing*) [27] para garantir a bidirecionalidade no canal de comunicação, sendo este dividido em *slots* de tempo cada um com duração de 625 μs. A técnica TDD é uma aplicação de multiplexagem no tempo TDM, [30] consistindo em um caso particular de acesso múltiplo por divisão de tempo (TDMA – *Time Division Multiple Access*).

A técnica de comunicação TDD possui vantagens na implementação de ligações assimétricas, nas quais as bandas de *uplink* e *downlink* são diferentes ou podem ser ajustadas dinamicamente. Por exemplo, quando o tráfego de *uplink* aumenta, pode ser alocada a ele a banda não utilizada pelo *downlink*, e vice-versa. Esse ajuste é efetuado apenas pela variação da relação entre os *slots* de tempo atríbuidos a cada enlace de comunicação [30].

Conforme pode observado na Figura 4.1, do intervalo de tempo total de cada *slot,* ou seja, dos 625 μs, uma parcela de tempo no final de cada *slot* é reservada para permitir o salto em freqüência e a sintonização dos rádios do mestre e do escravo. Por isso, o período de tempo do *slot* realmente ocupado pela transmissão de um pacote é de 366 μs, o que corresponde a 59% dos 625 μs.

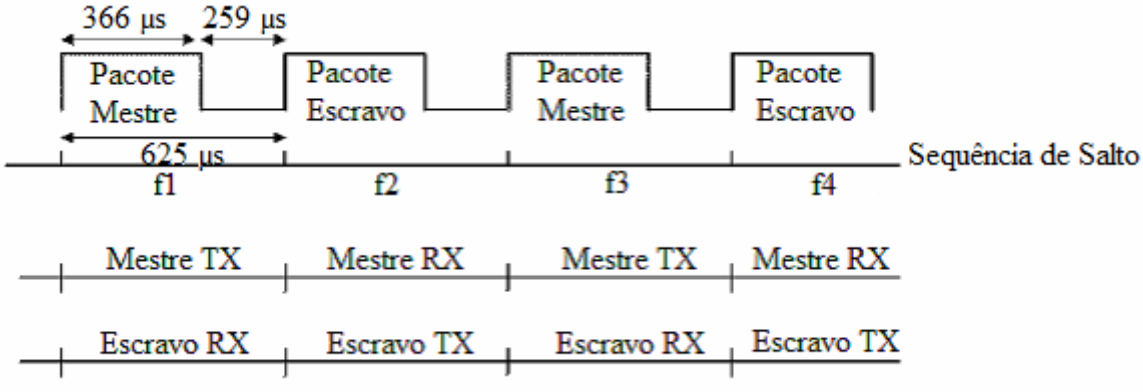


Figura 4.1 – *Seqüência de salto Mestre/Escravo (Modificado de [12]).*

Vê-se na Figura 4.1 que durante cada *slot* de tempo um dispositivo transmite por um dentre os canais disponíveis no sistema *Bluetooth*, sendo a transmissão alternada entre o mestre e um escravo. É importante observar que [7] o sistema salta em freqüência uma vez por pacote e não uma vez por *slot* de tempo. Essa mudança no canal se dá de uma maneira conhecida, porém pseudo-aleatória.

O dispositivo mestre em cada *piconet* define uma série de *slots* de tempo de 625 µs, com as transmissões iniciadas por ele nos *slots* pares e as dos escravos, nos ímpares. Isto nada mais é do que uma transmissão baseada na técnica TDM, com o dispositivo mestre tomando para si metade dos *slots* de tempo e os dispositivos escravos compartilhando a outra metade.

Os quadros podem ser longos o suficiente para ocuparem 1, 3 ou até 5 *slots* de tempo. Para transmissões de pacotes que ocupem apenas 1 *slot* de tempo, a taxa de saltos é de 1.600 saltos/s; para pacotes que ocupem 3 *slots*, a taxa é de aproximadamente 534 saltos/seg; e para pacotes que ocupem 5 *slots*, a taxa é de 320 saltos/s.

Cada quadro é transmitido sobre um canal lógico entre o mestre e o escravo. Existem dois tipos possíveis de canais lógicos [13]:

- ACL (*Asynchronous Connection –Less*) e;
- SCO (*Synchronous Connection Oriented*).

4.2.1 O enlace ACL

O enlace ACL é uma comutação por pacote, utilizado para dados disponíveis em intervalos de tempo irregulares. Esses dados são provenientes da camada L2CAP do transmissor e são entregues à camada L2CAP do receptor. O tráfego ACL é entregue com base no melhor esforço, ou seja, nenhuma espécie de garantia é dada. Conseqüentemente,

quadros podem ser perdidos, sendo necessárias retransmissões. Para o caso de reenvio de pacotes, [12] é utilizado o procedimento ARQ até que um reconhecimento positivo ACK seja recebido pelo transmissor.

Um escravo só pode ter um enlace ACL com seu mestre. Um dispositivo que estabelece um enlace ACL, pode enviar pacotes de tamanhos variados, que ocupem 1, 3 ou 5 *slots* de tempo, [12] denominados, respectivamente, de DM1, DM3 e DM5. Nesse tipo de enlace não há *slots* de tempo reservados para as transmissões [13, 19].

4.2.2 O enlace SCO

O enlace SCO é utilizado para tráfego de tempo real, como conexões de voz, consistindo em um canal ponto-a-ponto simétrico entre o mestre e o escravo [12]. Para esse tipo de canal lógico, *slots* de tempo fixos são determinados em cada direção e os pacotes SCO devem ocupar apenas um *slot* de tempo.

Os pacotes SCO são enviados do mestre para o escravo em intervalos de tempo regulares, definido como T_{SCO}. Esse parâmetro é ajustado em 2, 4 ou 6 *slots* de tempo para, respectivamente, os formatos de pacotes HV1, HV2 ou HV3. Todos os três formatos de pacotes SCO são definidos para transportar tráfego de voz a uma taxa de 64 Kbps, nunca sendo retransmitidos em caso de perda ou erro de pacote [12].

Em virtude das características do tráfego em tempo real, quadros já enviados nunca são retransmitidos. É utilizada a técnica FEC (*Forward Error Correction*), tanto em pacotes SCO quanto em pacotes ACL, para correção de erros no receptor, aumentando a confiabilidade do enlace e reduzindo o número de retransmissões necessárias. Um escravo pode ter até três enlaces *full-duplex* SCO com seu mestre, cada um sendo capaz de transmitir um canal de áudio PCM (*Pulse Code Modulation*) a uma taxa de 64 Kbps. Um dispositivo que estabelece um enlace SCO tem para si reservado determinados *slots* de tempo para uso. Seus pacotes de dados são tratados de forma prioritária e serão atendidos antes dos pacotes ACL [13, 19].

4.3 A estrutura de quadro *Bluetooth*

A estrutura de quadro da especificação *Bluetooth* está ilustrada na Figura 4.2.

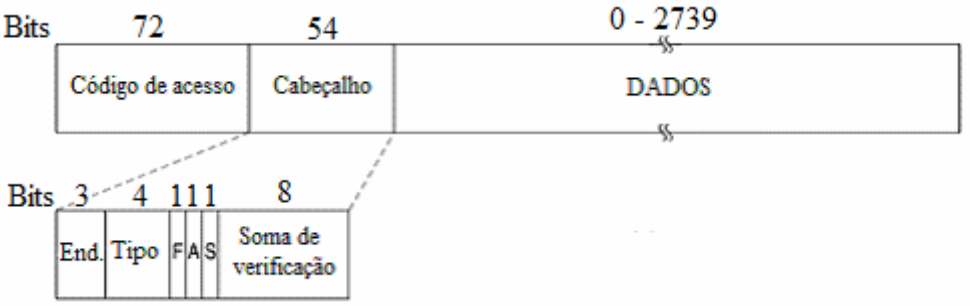


Figura 4.2 – *Estrutura padrão de quadro Bluetooth (Modificado de [13]).*

O primeiro campo é o código de acesso, composto por 72 bits. Esse campo é utilizado para identificar o mestre, de forma que um escravo que esteja dentro do alcance de dois mestres possa indicar nesse campo para qual dos dois o tráfego está destinado. O próximo campo é o de cabeçalho, formado por 54 bits. Esse campo contém informações típicas da subcamada MAC. O último é o campo de dados. Para o caso de pacotes que ocupam cinco slots de tempo em suas transmissões, ele é composto por 2.739 bits; para pacotes que são transmitidos utilizando apenas um slot de tempo, o campo de dados possui 240 bits.

Os campos que formam o cabeçalho são [13]:

- **endereço:** identifica para qual dos oito dispositivos ativos (sete escravos mais o mestre) o quadro está destinado;
- **tipo:** identifica o tipo de quadro, como por exemplo, quadros ACL, SCO, *poll* ou *null*, o tipo de correção de erro utilizado no campo de dados e qual o comprimento do pacote em termos de *slots* de tempo;
- **fluxo (F):** um dispositivo escravo utiliza esse campo para sinalizar que o seu *buffer* está cheio e que não pode receber mais dados. Esse campo é utilizado para controle de fluxo;
- ***acknowledgement* (A):** utilizado para reconhecimento de quadros recebidos. Esse bit de reconhecimento é geralmente enviado de carona (*piggyback*) em um quadro de dados do fluxo reverso. Caso não haja fluxo, um quadro só com o ACK é enviado, evitando *time-out*;
- **seqüência (S):** esse campo é utilizado para numerar os quadros, identificando retransmissões;
- **soma de verificação:** utilizado para detecção de erros.

Para proteger contra erros na transmissão, aplica-se correção de erros tanto no cabeçalho quanto no campo de dados do pacote *Bluetooth*.

Todo o cabeçalho, formado por 18 bits, é codificado por um código FEC com uma taxa de 1/3 para prover alta confiabilidade. Tal código é um código de repetição, no qual cada bit do cabeçalho é transmitido três vezes, totalizando 54 bits [13, 19]. Resumindo, o cabeçalho é transmitido três vezes. O circuito de recepção analisa as três cópias de cada bit. Se elas forem idênticas, o bit é aceito; se houver diferença entre essas três cópias, a mais freqüente será assumida como a correta. Com essa redundância, obtêm-se confiabilidade nas transmissões de dados em um ambiente ruidoso utilizando dispositivos de baixo custo e poder computacional.

Um código de Hamming opcional com taxa de 2/3 pode ser aplicado ao campo de dados (*payload*). Esse código Hamming é um código de bloco (15,10) em que temos n = 15 (tamanho da palavra-código), k = 10 (número de bits da mensagem) e n – k = 5 (bits de redundância).

O *Bluetooth* também suporta a opção ARQ, por meio da qual uma retransmissão pode ser solicitada pelo terminal de recepção se um pacote for recebido incorretamente [31].

Vários formatos são usados para o campo de dados de um quadro ACL. A estrutura de um quadro ACL DM5 está ilustrada na Figura 4.3.

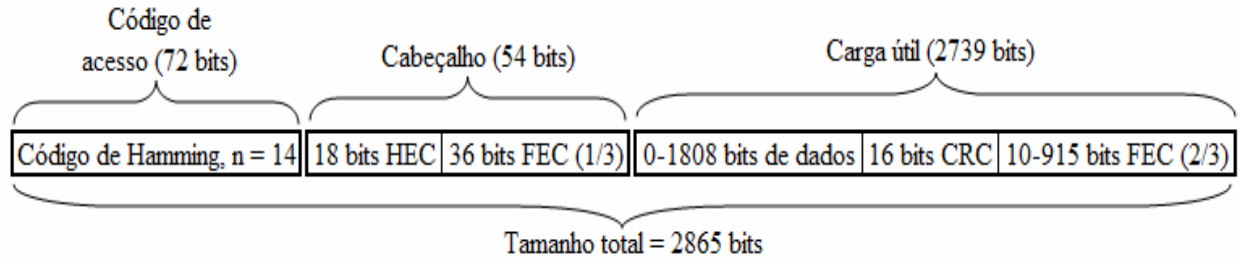


Figura 4.3 – *Estrutura de quadro DM5 Bluetooth (Modificado de [12]).*

Como pode ser visto na Figura 4.3, os pacotes DM5 utilizam um FEC com taxa 2/3 para corrigir erros na carga útil do pacote. Erros no cabeçalho ou no código de acesso são corrigidos, respectivamente, por um FEC com taxa 1/3 e por um código de Hamming.

O campo de dados de quadros SCO é sempre de 240 bits. Três variantes são definidas para quadros SCO, dependendo do tipo de proteção utilizada [13, 19]:

- **primeira variante.** Aplicando um código FEC com taxa de codificação de 1/3, permite-se uma carga útil com 80 bits, os quais são repetidos três vezes, totalizando os

240 bits. Essa técnica é a mesma que é utilizada no cabeçalho. A Figura 4.4 mostra um pacote HV1 da primeira variante.

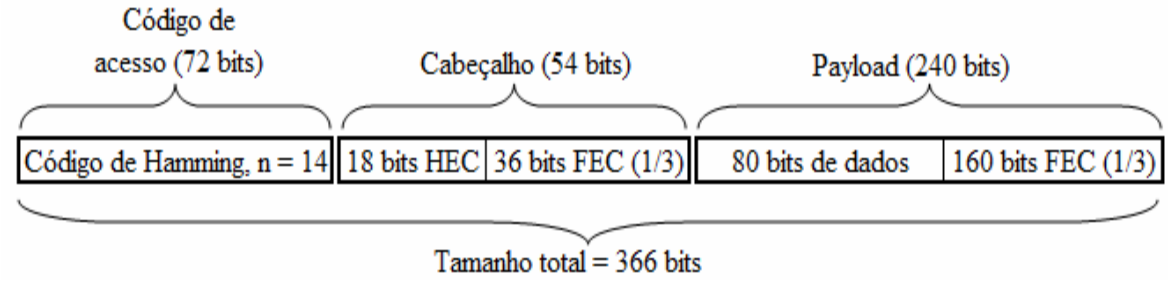


Figura 4.4 – *Estrutura de quadro HV1 Bluetooth (Modificado de [12]).*

Como pode ser visto na Figura 4.4, erros no código de acesso são corrigidos pelo código de Hamming e erros no cabeçalho e na carga útil são corrigidos com o FEC de taxa 1/3.

- **segunda variante.** Utilizando-se um código FEC com taxa de codificação de 2/3, o qual é um código de Hamming encurtado (15,10) capaz de corrigir todos os erros de um único bit e detectar todos os erros de dois bits, permite-se uma carga útil com 160 bits. O restante dos bits são utilizados para correção de erros;
- **terceira variante:** Se não for utilizada a técnica FEC, transmite-se com uma taxa de codificação de 3/3, na qual a carga útil é de 240 bits.

Dessa forma, a versão mais confiável é a de 80 bits de carga útil e a versão menos confiável é a de 240 bits de carga útil, na qual não restam bits no campo de dados para a correção de erros.

A Tabela 4.3 resume a ocorrência de erros no pacote e a ação tomada caso os erros não sejam corrigidos.

Tabela 4.3 – *Ação tomada em virtude da localização do erro.*

Localização do erro	Técnica para correção	Ação tomada
Código de acesso	Código de Hamming, n =14	Pacote é descartado
Cabeçalho	FEC (1/3)	Pacote é descartado
Carga útil do HV1	FEC (1/3)	Pacote é aceito
Carga útil do DM5	FEC (2/3)	Pacote é descartado

Conforme já dito anteriormente, o sistema *Bluetooth* trabalha com uma taxa de 1.600 saltos por segundo. Considerando transmissões de pacotes que ocupem apenas 1 *slot* de tempo e que o sistema salta em freqüência por pacote, uma taxa de 1.600 saltos/s

implica em dizer que estão sendo transmitidos 1.600 pacotes por segundo. Como cada pacote é transmitido em um único *slot* de tempo, temos uma quantidade total de 1.600 *slots* por segundo, onde cada *slot* tem duração de 625 µs.

Dessa forma, como os dispositivos escravos transmitem apenas nos *slots* ímpares, eles tomam para si 800 *slots*/s, assim como o mestre faz. Com uma carga útil de 80 bits, a capacidade do canal a partir do escravo é de 64.000 bps (80 bits/*slot* x 800 *slots*/s); a capacidade do canal a partir do mestre também é de 64 Kbps. Isso já é suficiente para um canal de voz PCM *full-duplex*. A Figura 4.5 ilustra uma conexão SCO *full-duplex* entre um mestre e um escravo para uma carga útil de 80 bits.

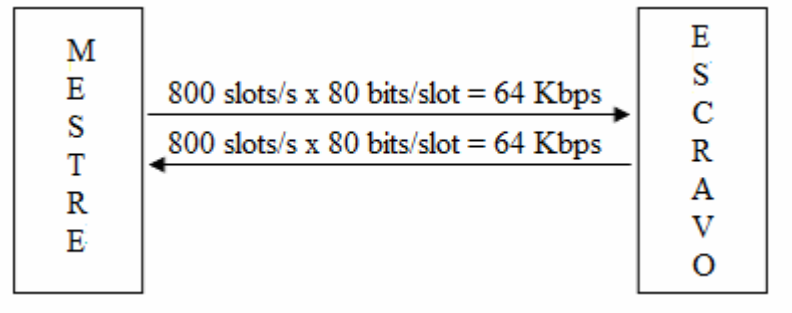


Figura 4.5 – *Canal SCO full-duplex para tráfego de voz.*

Para a variante do quadro SCO menos confiável (a com 240 bits de carga útil), três canais de voz *full-duplex* podem coexistir simultaneamente, razão pela qual um máximo de três enlaces SCO são permitidos por escravo [13]. A Figura 4.6 ilustra três enlaces SCO *full-duplex*, cada um com uma capacidade de 64 Kbps.

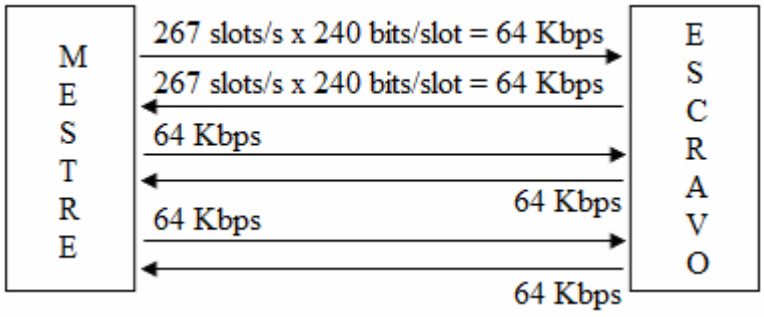


Figura 4.6 – *Máximo de três canais SCO full-duplex.*

Na situação ilustrada na Figura 4.6, a quantidade total de 1.600 *slots*/s é dividida por entre os seis canais de voz PCM, ficando cada um com aproximadamente 267 *slots*/s.

Capítulo 5. MODELAGEM MATEMÁTICA DE INTERFERÊNCIA

O uso simultâneo de sistemas *Bluetooth* em ambientes onde existam tráfego de redes IEEE 802.11g resulta em interferência, causando degradação no desempenho da rede *Wi-Fi*.

Para que uma WLAN 802.11g possa ser submetida à interferência *Bluetooth*, ou seja, para que uma colisão possa ocorrer, ambos os sinais devem coincidir tanto na freqüência quanto no tempo [7]. A Figura 5.1 ilustra a relação no domínio do tempo e da freqüência dos sinais *Bluetooth* e 802.11g.

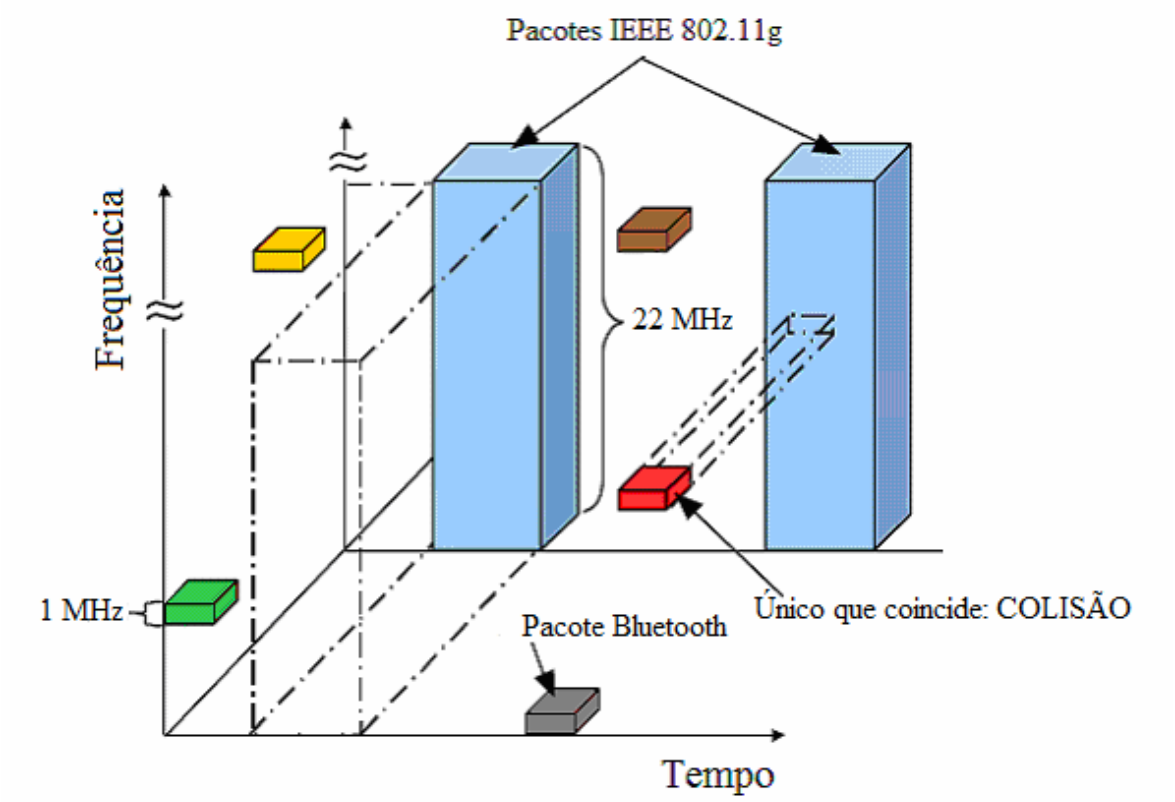


Figura 5.1 – *Relação dos sinais Bluetooth e 802.11g nos domínios do tempo e freqüência (Modificado de [32]).*

Os pacotes *Bluetooth* sinalizados na Figura 5.1 têm largura de banda de 1 MHz e os pacotes 802.11g, 20 MHz. No sistema 802.11g, [33] a largura de banda utilizada é de 22 MHz, sendo que o sinal OFDM ocupa 20 MHz.

Conforme pode ser observado pela Figura 5.1, o sistema *Bluetooth* se baseia na técnica FHSS, na qual cada pacote é transmitido em um canal diferente ao longo do tempo. O sinal 802.11g, por sua vez, utiliza um canal fixo durante o tempo. Dessa forma, existe uma probabilidade não nula de os pacotes *Bluetooth* coincidirem no tempo e na freqüência com os pacotes 802.11g, ocasionando uma colisão. Analisando a Figura 5.1, observa-se

que o pacote *Bluetooth* que ocasiona uma colisão com o pacote 802.11g é aquele sinalizado na cor vermelha. Abaixo está listada a coincidência dos pacotes *Bluetooth* com os pacotes 802.11g nos domínio do tempo e da freqüência:

- **o pacote verde**. Coincide no domínio da freqüência, mas não do tempo;
- **o pacote amarelo**. Não coincide nem no domínio da freqüência nem no do tempo;
- **o pacote cinza**. Coincide no domínio da freqüência, mas não do tempo;
- **o pacote marrom**. Coincide no tempo, mas não na freqüência;
- **o pacote vermelho**. Coincide em ambos os domínios, ocasionando uma colisão.

5.1 Colisão no domínio da freqüência

A Figura 5.2 mostra o espectro de um canal IEEE 802.11g e de um canal *Bluetooth*.

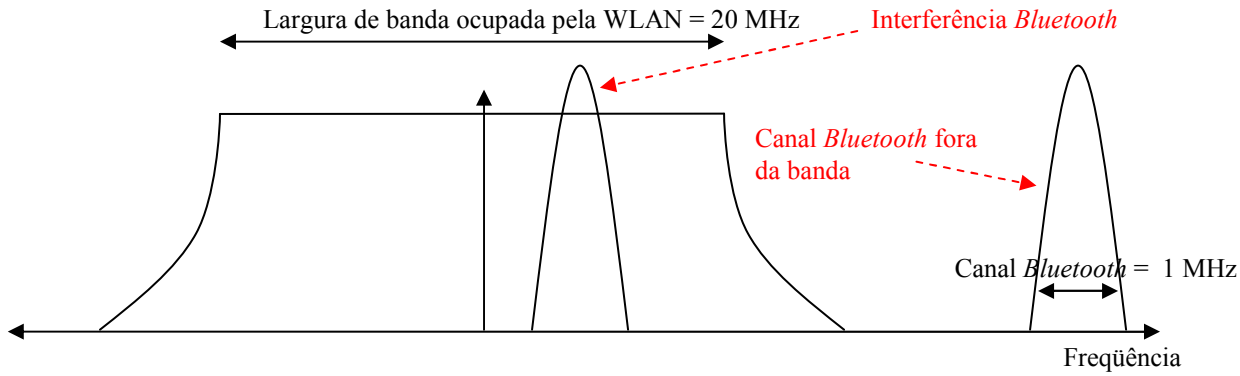


Figura 5.2 – *Interferência Bluetooth em uma WLAN 802.11g (Modificado de [7]).*

O sinal *Bluetooth*, utilizando FHSS, ocupa 1 dentre 79 canais diferentes a cada *slot* de tempo. Conforme visto anteriormente, cada um desses 79 canais tem uma largura de banda de 1 MHz, totalizando 79 MHz. Por sua vez, o sinal 802.11g ocupa uma largura de banda de aproximadamente 20 MHz [7].

A probabilidade de os sinais 802.11g e *Bluetooth* se sobreporem no domínio da freqüência [34] é igual à probabilidade da rede *Bluetooth* saltar para dentro da banda da rede 802.11g.

Dado o procedimento utilizado pelo *Bluetooth* para gerar a seqüência de saltos, pode-se aproximar o salto do *Bluetooth* para dentro da banda da rede *Wi-Fi* por um processo i.i.d (independente e identicamente distribuído) com parâmetro h_f [34]. Quando não é aplicado nenhum mecanismo de coexistência, pode-se escrever [34]:

$$h_f = \frac{22}{79} \cdot 100 \approx 27{,}8\% , \tag{5.1}$$

onde 22 MHz e 79 MHz são, respectivamente, a largura de banda da rede 802.11g e do *Bluetooth*.

É importante observar que para pacotes *Bluetooth* maiores, que ocupem, por exemplo, 5 *slots* de tempo, a probabilidade de colisão com um dado pacote 802.11g é menor, visto que a taxa de salto em freqüência do sistema *Bluetooth* será reduzida [7].

5.2 Colisão no domínio do tempo

A probabilidade de que os pacotes 802.11g e *Bluetooth* coincidam no tempo é função do comprimento de ambos os pacotes [7]. As Figuras 5.3 (a), (b) e (c) ilustram a ocupação no domínio do tempo do *Bluetooth* e do 802.11g.

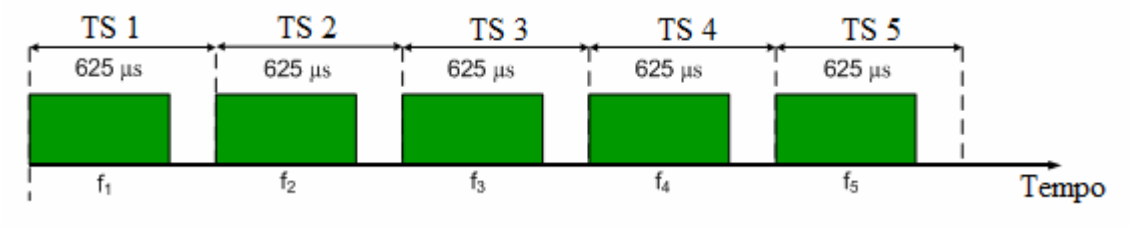


Figura 5.3a – *Efeito do tamanho do pacote na probabilidade de colisão: transmissão de pacotes Bluetooth com comprimento de 1 slot de tempo (Modificado de [7]).*

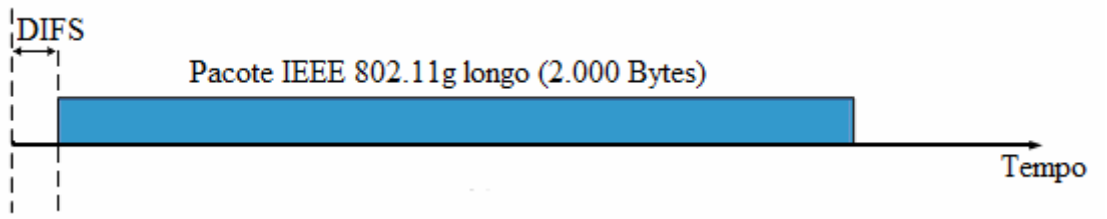


Figura 5.3b – *Efeito do tamanho do pacote na probabilidade de colisão: transmissão de um pacote 802.11g longo (2000 bytes) (Modificado de [7]).*

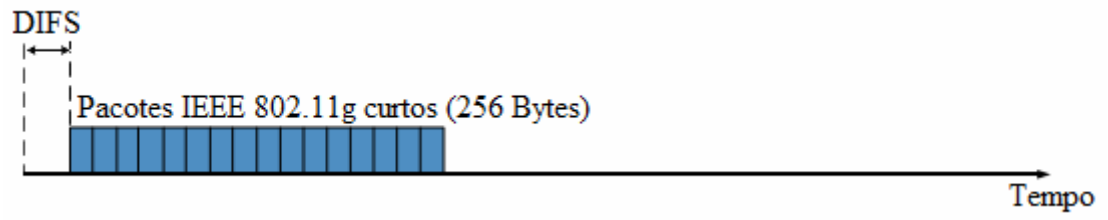


Figura 5.3c – *Efeito do tamanho do pacote na probabilidade de colisão: transmissão de pacotes 802.11g curtos (256 bytes) (Modificado de [7]).*

Na Figura 5.3 (a), cinco pacotes *Bluetooth* são transmitidos, sendo que cada um é transmitido em uma freqüência diferente e em *time slots* (TS) distintos. Considerando a

transmissão de um pacote 802.11g longo, conforme ilustrado na Figura 5.3 (b), vê-se que é praticamente certo que ocorrerá uma colisão entre esse pacote longo e os cinco pacotes *Bluetooth* em virtude da coincidência no tempo. Para o primeiro e último pacote *Bluetooth*, a colisão seria parcial. Entretanto, para os três pacotes do centro, a colisão seria total.

Para o caso da Figura 5.3 (c), onde ocorre a transmissão de pacotes 802.11g curtos, a probabilidade de colisão diminui. Isso ocorre em virtude do fato de que alguns pacotes 802.11g são transmitidos durante o intervalo de contenção entre os pacotes *Bluetooth*. A probabilidade de pacotes 802.11g serem completamente transmitidos durante o intervalo de tempo entre dois pacotes *Bluetooth* depende da proporção de tempo em que o *Bluetooth* não está transmitindo. Visto que pacotes *Bluetooth* maiores têm um intervalo de contenção menor, a probabilidade de colisão com um dado pacote 802.11g será maior [7].

Conforme ilustra a Figura 5.4, denotaremos o *slot* de tempo total do *Bluetooth* por T_{BI} (625 µs), a parcela de tempo do *slot* que realmente é ocupada por um pacote *Bluetooth* por T_{BP} (366 µs), e o tempo de duração de um pacote 802.11 por T_W.

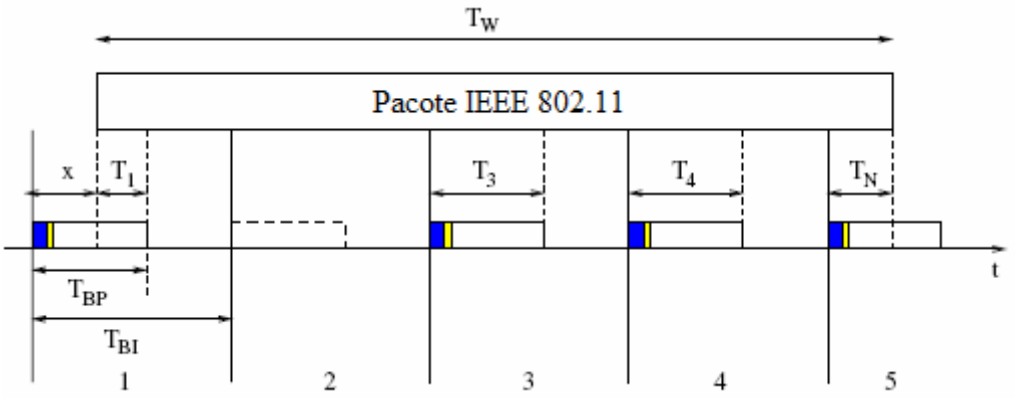


Figura 5.4 – *Sobreposição temporal entre um pacote IEEE802.11 e pacotes Bluetooth (Modificado de [34]).*

Conforme mostrado na Figura 5.4, denota-se x o intervalo de tempo que vai do início do primeiro *slot Bluetooth*, que se sobrepõe ao pacote IEEE 802.11, até o início deste pacote. Assim [35, 36], x é uma variável aleatória uniformemente distribuída entre 0 e T_{BI} (0, 625 µs). Chama-se $N(x)$ o número de *slots* de tempo *Bluetooth* que coincidem no tempo com o pacote IEEE 802.11. $N(x)$ depende de x e pode ser expresso por [34]:

$$N(x) = \begin{cases} \left\lceil \frac{T_W}{T_{BI}} \right\rceil & \text{se} \quad x \le T_{BI} \cdot \left\lceil \frac{T_W}{T_{BI}} \right\rceil - T_W \\ \left\lceil \frac{T_W}{T_{BI}} \right\rceil + 1 & \text{caso contrário} \end{cases} \tag{5.2}$$

A Figura 5.4 ilustra um exemplo com *N(x)* = 5 e pacotes *Bluetooth* de comprimento igual a 1 *slot* de tempo. A variável T_i (*i* = 1, 2,..., *N(x)*) indica a porção do i-ésimo pacote *Bluetooth* que realmente interfere com o pacote IEEE 802.11. Para um *slot* de tempo *i* (*i* = 1, 2,..., *N(x)*) qualquer, se nenhuma transferência *Bluetooth* ocorrer, tem-se que $T_i = 0$ (o que acontece no *slot* de tempo 2 da Figura 5.4); caso contrário [34]:

$$T_i = \begin{cases} \max(T_{BP} - x, 0) & i = 1 \\ T_{BP} & i = 2,\ldots,N(x) - 1 \\ \min(x + T_W - (N(x)-1)\cdot T_{BI}, T_{BP}) & i = N(x) \end{cases} \quad (5.3)$$

Fixando o valor de *x*, para *i* = 1, 2,..., *N(x)* define-se δ_i como a probabilidade de que tráfego *Bluetooth* seja gerado no *slot* de tempo *i*.

De acordo com [34], o número médio de símbolos que são degradados como conseqüência de uma colisão entre os sistemas *Bluetooth* e 802.11 pode ser escrito por:

$$\eta_x = h_f\left(T_1^{(s)}\delta_1 + \sum_{i=2}^{N(x)-1} T_i^{(s)}\delta_i + T_{N(x)}^{(s)}\delta_{N(x)}\right), \quad (5.4)$$

onde $T_i^{(s)}$ é a razão T_i / T_s (*i* = 1, 2,..., *N(x)*), com T_s sendo o tempo de duração de um símbolo. Da equação acima, verifica-se que para reduzir a interferência mútua entre os sistemas *Bluetooth* e 802.11, torna-se necessário reduzir um dos seguintes parâmetros: *N(x)*, h_f ou δ_i. Um *N(x)* pequeno pode ser obtido utilizando pacotes WLAN curtos. Reduzir o h_f implica em fazer com que a probabilidade de sobreposição na freqüência entre as transmissões *Bluetooth* e 802.11 seja pequena. Obter um pequeno δ_i implica em reduzir a probabilidade de colisão no domínio do tempo [34].

Como será detalhado no capítulo 6, os experimentos desta dissertação foram feitos para três distâncias entre os transmissores e receptores *Bluetooth* e *Wi-Fi*. (1,60 m, 2,60 m e 4,60 m). Apesar de o modelo analítico apresentado por F. Chiasserini e R. Rao em [34] não levar em consideração a distância e os obstáculos entre os transmissores e receptores *Wi-Fi* e *Bluetooth* e não especificar qual padrão IEEE 802.11 estudado, será utilizado aqui a equação (5.4) para prever o número de símbolos corrompidos devido a uma colisão entre os dois sistemas.

Conforme será visto na seção 7.6, a média da taxa de transmissão de pacotes para a distância de 1,60 m com interferência *Bluetooth* é de 686,247 pacotes por segundo. Isso dá um T_W igual a 1.457,2 µs. Admitindo que para a distância de 1,60 m os pacotes *Bluetooth* são do tipo DM5, que ocupam cinco *slots* de tempo, temos que T_{BI} é igual a T_{BP}. Utilizando a equação (5.2) tem-se:

$$x \leq T_{BI} \cdot \left\lceil \frac{T_W}{T_{BI}} \right\rceil - T_W \therefore$$
$$x \leq 625\mu s \cdot \left\lceil \frac{1457,2\mu s}{625\mu s} \right\rceil - 1457,2\mu s \therefore$$
$$x \leq 417,8\mu s$$

Logo, obtém-se utilizando a equação (5.2) *N(x)* = 3 para $x \leq 417,8\mu s$ e *N(x)* = 4 caso contrário. Fazendo uso da equação (5.3) e fixando o valor de *x* em 417,8μs, tem-se:

$$T_1 = \max(625\mu s - 417,8\mu s, 0) \therefore$$
$$T_1 = 207,2\mu s$$

$$T_2 = 625\mu s$$

$$T_3 = \min(417,8\mu s + 1457,2\mu s - (3-1) \cdot 625\mu s, 625\mu s) \therefore$$
$$T_3 = 625\mu s$$

Conforme será visto também na seção 7.6, a taxa média de transmissão de dados para a distância de 1,60 m com interferência *Bluetooth* é de 5,475 Mbps. Isso implica em um tempo médio de símbolo (T_s) de aproximadamente 0,183 μs. Como foi utilizado tráfego *Bluetooth* constante em todos os testes realizados nesta dissertação, podemos assumir sempre δ_i = 1. Substituindo na equação (5.4) os parâmetros obtidos do experimento realizado para 1,60 m, tem-se:

$$\eta_x = 0,278 \cdot \left(\frac{207,2\mu s}{0,183\mu s} \cdot 1 + \frac{625\mu s}{0,183\mu s} \cdot 1 + \frac{625\mu s}{0,183\mu s} \cdot 1 \right) \therefore$$
$$\eta_x = 2.214$$

Assim, quando houver uma colisão entre os sistemas *Bluetooth* e 802.11 para a distância de 1,60m, segundo o modelo analítico proposto por F. Chiasserini e R. Rao em [34], encontra-se uma quantidade de símbolos corrompidos igual a 2.214. Como o tamanho médio dos pacotes nos testes realizados na distância de 1,60m com interferência *Bluetooth* é de 997,159 Bytes (7.977,272 bits), ao ocorrer uma colisão, aproximadamente 27,75% dos bits serão degradados, acarretando perda de pacotes.

Da mesma maneira que para os testes realizados na distância de 1,60 m, é assumido que os pacotes *Bluetooth* nos testes realizados na distância de 2,60 m são do tipo DM5. Como será visto na seção 7.6, temos, para 2,60 m, os seguintes parâmetros com interferência *Bluetooth*:

- média da taxa de transmissão de pacotes: 653,257 pacotes por segundo, o que implica T_W = 1.530,79 μs;

- média da taxa de transmissão de dados: 5,212 Mbps, o que implica T_S = 0,192 μs.

Utilizando a equação (5.2), tem-se:

$$x \leq 625 \cdot \left\lceil \frac{1530,79\mu s}{625\mu s} \right\rceil - 1530,79\mu s \therefore$$
$$x \leq 344,21\mu s$$

Para $x \leq 344,21$ μs, temos $N(x) = 3$. Caso contrário, $N(x) = 4$. Fixando $x = 344,21$μs e utilizando a equação (5.3), tem-se:

$$T_1 = \max(625\mu s - 344,21\mu s, 0) \therefore$$
$$T_1 = 280,79\mu s$$

$$T_2 = 625\mu s$$

$$T_3 = \min(344,21\mu s + 1530,79\mu s - (3-1) \cdot 625\mu s, 625\mu s) \therefore$$
$$T_3 = 625\mu s$$

Substituindo os valores na equação (5.4) obtém-se:

$$\eta_x = 0,278 \cdot \left(\frac{280,79\mu s}{0,192\mu s} \cdot 1 + \frac{625\mu s}{0,192\mu s} \cdot 1 + \frac{625\mu s}{0,192\mu s} \cdot 1 \right) \therefore$$
$$\eta_x = 2.216$$

Dessa forma, quando houver uma colisão entre os sistemas *Bluetooth* e 802.11 para a distância de 2,60 m, encontra-se uma quantidade de símbolos corrompidos igual a aproximadamente 2.216. Considerando que o tamanho médio dos pacotes nos testes realizados na distância de 2,60 m com interferência *Bluetooth* é de 997,402 Bytes (7.979,216 bits), ao ocorrer uma colisão, aproximadamente 27,77% dos bits serão degradados, acarretando perda de pacotes.

Para os testes realizados na distância de 4,60 m, foi assumido que os pacotes *Bluetooth* são do tipo DM1. Tem-se, como será visto na seção 7.6, para essa distância com interferência *Bluetooth*, os parâmetros T_W e T_S abaixo:

- média da taxa de transmissão de pacotes: 385,738 pacotes por segundo, o que implica T_W = 2.592,43 μs;
- média da taxa de transmissão de dados: 3,059 Mbps, o que implica T_S = 0,327 μs.

Utilizando a equação (5.2), obtém-se:

$$x \leq 625 \cdot \left\lceil \frac{2592{,}43\mu s}{625\mu s} \right\rceil - 2592{,}43\mu s \therefore$$
$$x \leq 532{,}57\mu s$$

Com base na equação (5.2), para $x \leq 532{,}57$ µs, temos $N(x) = 5$. Caso contrário, $N(x) = 6$. Fixando $x = 532{,}57$ µs e utilizando a equação (5.3), tem-se:

$$T_1 = \max(366\mu s - 532{,}57\mu s, 0) \therefore$$
$$T_1 = 0\mu s$$

$$T_2 = T_3 = T_4 = 366\mu s$$

$$T_5 = \min(532{,}57\mu s + 2592{,}43\mu s - (5-1) \cdot 625\mu s, 366\mu s) \therefore$$
$$T_5 = 366\mu s$$

Substituindo os valores na equação (5.4) obtém-se:

$$\eta_x = 0{,}278 \cdot \left(\frac{0\mu s}{0{,}327\mu s} \cdot 1 + \frac{366\mu s}{0{,}327\mu s} \cdot 1 + \frac{366\mu s}{0{,}327\mu s} \cdot 1 + \frac{366\mu s}{0{,}327} \cdot 1 + \frac{366\mu s}{0{,}327} \right) \therefore$$
$$\eta_x = 1.245$$

Para os testes realizados em 4,60 m, quando houver uma colisão entre os sistemas *Bluetooth* e 802.11, teremos uma quantidade de símbolos corrompidos igual a aproximadamente 1.245. Considerando que o tamanho médio dos pacotes nos testes realizados na distância de 4,60 m com interferência *Bluetooth* é de 990,815 Bytes (7.926,52 bits), ao ocorrer uma colisão, aproximadamente 15,7% dos bits serão degradados, acarretando perda de pacotes.

Enquanto que o modelo analítico exposto por F. Chiasserini e R. Rao em [34] calcula o número médio de bits corrompidos quando há uma colisão entre um pacote *Bluetooth* e um pacote IEEE 80211, o modelo descrito por Hsu, Wei e C.C. em [37] estabelece a taxa de erro de pacotes de uma estação sem fio 802.11 sob interferência *Bluetooth*. A equação (5.5) descreve o modelo analítico exposto em [37]:

$$PER_{Wi-Fi} = 1 - (1 - h_f \cdot \delta_i \cdot \sigma)^N , \qquad (5.5)$$

onde h_f é a probabilidade de os sinais *Bluetooth* e 802.11 coincidirem na freqüência, δ_i é a probabilidade de ser gerado tráfego *Bluetooth* no *slot* de tempo i, σ é a razão entre o tempo de atividade em um *slot* de tempo *Bluetooth* (T_{BP}) e o tempo total de um *slot Bluetooth* (T_{BI}) e N é o número de *slots* de tempo *Bluetooth* que coincidem no tempo com o pacote 802.11.

Para as três distâncias de testes analisadas nesta dissertação, tem-se h_f = 0,278 e δ_i=1. Para as distâncias de 1,60 m e 2,60 m, temos que σ = 1, pois T_{BP} é igual a T_{BI} . Para a distância de 4,60m, σ = 0,5856 (366 μs/625 μs). Conforme já calculado anteriormente, para as distâncias de 1,60 m e 2,60 m temos *N(x)* = 3; para a distância de 4,60 m, tem-se *N(x)* = 5.

Substituindo os valores na equação (5.5), calcula-se a $PER_{Wi\text{-}Fi}$ (*Packet Error Rate*) para cada uma das distâncias:

$$PER_{Wi-Fi(1,60m)} = 1-(1-0{,}278\cdot 1\cdot 1)^3 \therefore$$
$$PER_{Wi-Fi(1,60m)} = 62{,}36\%$$

$$PER_{Wi-Fi(2,60m)} = 1-(1-0{,}278\cdot 1\cdot 1)^3 \therefore$$
$$PER_{Wi-Fi(2,60m)} = 62{,}36\%$$

$$PER_{Wi-Fi(4,60m)} = 1-(1-0{,}278\cdot 1\cdot 0{,}5856)^5 \therefore$$
$$R_{Wi-Fi(4,60m)} = 58{,}87\%$$

O modelo analítico exposto tanto por Jo e Jayant em [38] quanto por Jim Zyren em [35] para cálculo da probabilidade de colisão entre pacotes *Wi-Fi* e *Bluetooth* são para redes sem fio do padrão IEEE 802.11b e para pacotes *Bluetooth* do tipo DM1. Entretanto, achou-se válido usar também esse modelo para tentar prever a probabilidade de perda de pacotes devido a uma colisão entre os sistemas *Bluetooth* e IEEE 802.11g para a distância de 4,60 m, pois os pacotes *Bluetooth* são do tipo DM1. De acordo com [38] e [35], a probabilidade total de colisão $P_{total}(N)$ entre um pacote WLAN IEEE 802.11b e um pacote *Bluetooth* é calculado como:

$$P_{total}(N) = (P_{N-1}\cdot P_{coll}(N-1) + (P_N \cdot P_{coll}(N)) \tag{5.6}$$

onde,

$$P_{N-1} = \frac{T_{BI}\cdot(N-1) + T_{BI} - T_{BP} - T_W}{T_{BI}} \tag{5.7}$$

$$P_{coll}(N-1) = 1-(1-(h_f\cdot\delta_i))^{N-1} \tag{5.8}$$

$$P_N = 1 - P_{N-1} \tag{5.9}$$

$$P_{coll}(N) = 1-(1-(h_f\cdot\delta_i))^{N} \tag{5.10}$$

Utilizando as equações (5.6) a (5.10) obtém-se, para a distância de 4,60 m com *N(x)*=5:

$$P_{5-1} = P_4 = \frac{625\mu s \cdot (5-1) + 625\mu s - 366\mu s - 2592{,}43\mu s}{625\mu s} = 26{,}65\%$$

$$P_{coll}(5-1) = P_{coll}(4) = 1 - (1 - (0{,}278 \cdot 1))^4 = 72{,}82\%$$

$$P_5 = 1 - P_4 = 1 - 0{,}2665 = 73{,}35\%$$

$$P_{coll}(5) = 1 - (1 - (0{,}278 \cdot 1))^5 = 80{,}38\%$$

$$P_{total}(5) = (0{,}2665 \cdot 0{,}7282) + (0{,}7335 \cdot 0{,}8038) = 78{,}36\%$$

Logo, para a distância de testes de 4,60 m, encontra-se uma probabilidade de colisão de 78,36%.

No capítulo 7 serão expostos os resultados dos testes práticos e comparados com os resultados teóricos obtidos aqui. Vale salientar que os modelos analíticos encontrados na literatura não são, a princípio, muito adequados à aplicação nos testes realizados nesta dissertação, pois eles foram desenvolvidos tomando como base o padrão IEEE 802.11b e tráfego *Bluetooth* com pacotes do tipo DM1. Outra razão para a possível não adequação é que os modelos analisados não levam em conta o incremento da distância e a presença de obstáculos.

Capítulo 6. Revisão bibliográfica

Este capítulo apresenta uma revisão bibliográfica de trabalhos co-relacionados ao tema desta dissertação, encontrados na literatura. São sumariamente descritos os trabalhos de Doufexi, Arumugam e Armour [7], Abukharis e O'Farell [24], Mckay e Masuda [2], Wong e O'Farell [39], Rukh [40] e Jo e Jayant [38].

Doufexi, Arumugam e Armour [7] investigaram o impacto de um sinal interferente *Bluetooth* sobre a taxa de erro de pacotes (PER – *Packet Error Rate*) de uma rede *Wi-Fi* 802.11g. O estudo também analisa o efeito causado em função do tamanho dos pacotes *Bluetooth* e 802.11g. Nesse estudo, foi constatado que a PER da rede *Wi-Fi* é aumentada quando se eleva o nível de interferência *Bluetooth*, considerando, para a rede *Wi-Fi*, o modo de transmissão 5 (24 Mbps) e um tamanho de PSDU de 500 bytes. Para o sistema *Bluetooth*, foi assumido que ele gerava pacotes de comprimento igual a 1 *slot* de tempo. Foi verificado também que aumentando o tamanho da PSDU resulta em uma PER mais elevada. Todavia, esse desempenho também depende do modo de transmissão 802.11g utilizado. Para um mesmo modo de transmissão e nível de interferência *Bluetooth*, constatou-se que o desempenho da rede *Wi-Fi* é melhor para PSDUs menores.

Como conclusão, relata-se em [7] que a probabilidade de colisão entre os dois sistemas depende da razão entre o tempo de transmissão de um pacote 802.11g e de um pacote *Bluetooth*, como também da razão entre o espectro de freqüência ocupado pelos dois padrões de rede sem fio. É mencionada, em caráter informativo, uma estratégia que está sendo proposta para possibilitar a coexistência dos dois sistemas. Trata-se de utilizar no *Bluetooth* a técnica AFH (*Adaptive Frequency Hopping*), que habilita o dispositivo a reduzir o número de canais nos quais o sistema salta, deixando alguns deles livres para outros dispositivos (802.11g, por exemplo).

6.1 Streaming MPEG2 sob interferência *Bluetooth*

Abukharis e O'Farell em [24] realizam um estudo avaliando a transmissão de *streaming* de vídeo MPEG2 (*Moving Picture Experts Group*) quando submetida à interferência *Bluetooth*. Nesse trabalho, é proposto um mecanismo que opera na camada física para reduzir os efeitos da interferência, restabelecendo a QoS (*Quality of Service*). Segundo esse estudo, a qualidade de transmissões de vídeo depende de vários parâmetros, como por exemplo, codificação e compressão, perda de pacotes, atraso, quantidade de movimento na cena, cor, contraste e tamanho da imagem. Logo, métricas convencionais de

rede, tais como PER e BER, não dão uma medida consistente da qualidade do vídeo. Por esse motivo, foi utilizada no estudo uma métrica específica para medir a qualidade do vídeo recebido. Essa métrica é baseada em uma medida objetiva da qualidade percebida.

Os vídeos MPEG2 codificados com taxas de 2, 4 e 6 Mbps eram passados para a camada física 802.11g utilizando pacotes com tamanho de 100 bytes. Os pacotes MPEG eram encapsulados em pacotes IEEE 802.11g e transmitidos pelo canal sem fio. Uma fonte interferente *Bluetooth* foi localizada a uma certa distância da fonte transmissora do vídeo. A distância entre o AP e a fonte transmissora do vídeo foi mantida inalterada.

Quando o sinal *Bluetooth* salta para dentro da largura de banda utilizada pelo sinal 802.11g, ele causa interferência de banda estreita, que afeta um pequeno número de subportadoras OFDM. De acordo com o estudo realizado em [24], a relação sinal/ruído em uma determinada sub-portadora OFDM é determinada instantaneamente pela potência tanto do sinal OFDM quanto do sinal *Bluetooth* que foi transmitido dentro da banda correspondente àquela subportadora. Dessa forma, os símbolos transportados nas subportadoras afetadas, ou seja, aqueles com baixa relação sinal/ruído, são removidos. Essa remoção ocorre por meio da inserção do que a literatura chama de apagamentos (*erasures*), onde valor do símbolo é zerado.

Ainda de acordo com [24], as vantagens de se utilizar o mecanismo de apagamentos na camada física são que ele não causa impacto nas especificações 802.11g e *Bluetooth*, podendo ser utilizado para tratar qualquer tipo de interferência e não requer uma colaboração explícita entre os dispositivos 802.11g e *Bluetooth*.

Nos experimentos realizados em [24], foram utilizadas três taxas de codificação de vídeo MPEG2 (2, 4 e 6 Mbps) para o mesmo vídeo clip. Esse conteúdo era então enviado pelo canal 802.11g à taxa de dados de 24 Mbps e 54 Mbps. O dispositivo *Bluetooth* foi localizado a 10 metros da estação móvel. A distância entre esta e o AP foi de 46 metros para uma taxa de dados de 24 Mbps e 24 metros para 54 Mbps. Foi constatado nesse trabalho que, apesar de o sistema 802.11g operar a uma potência de transmissão mais elevada (20 dBm) do que o sistema *Bluetooth* (0 dBm), uma substancial degradação na qualidade do vídeo foi causada em virtude da interferência. Os resultados também mostraram que a qualidade do vídeo recebido depende da taxa de codificação de vídeo utilizada, da taxa de transmissão da rede *Wi-Fi* e do número de apagamentos aplicados.

6.2 VoIP e interferência *Bluetooth*

Mckay e Masuda [2] estudaram os efeitos da interferência *Bluetooth* na qualidade de transmissão VoIP em redes 802.11b. Nesse trabalho, foram realizadas medições da qualidade de transmissão de voz sobre o enlace 802.11b em dezesseis condições diferentes. O nível de sinal do enlace 802.11b foi variado de -75 dBm a -85 dBm (em passos de 5 dB), utilizando-se materiais que absorvem a radiação e objetos metálicos para atenuar o sinal entre o AP e o *laptop*. O ruído médio observado foi de -95 dBm. Assim, trabalhou-se em [2] com relações sinal/ruído de 20, 15 e 10 dB. A Figura 6.1 ilustra a topologia aplicada no experimento.

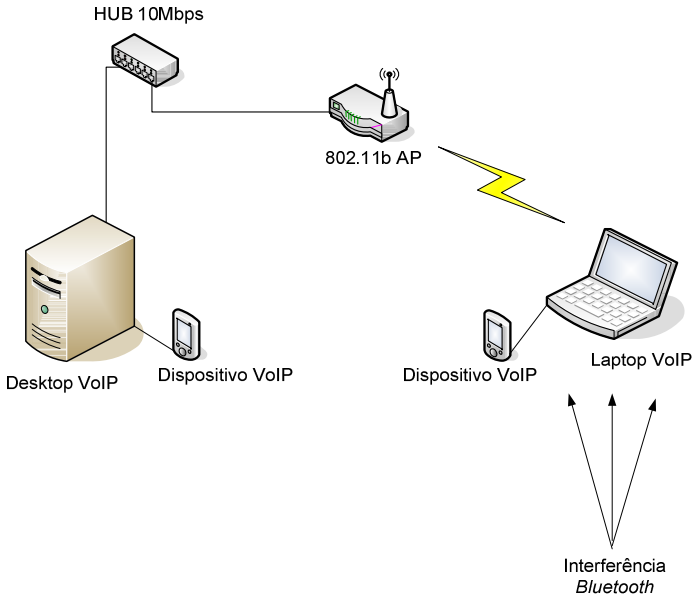


Figura 6.1– *Topologia do experimento realizado por Mckay e Masuda [2] para medição da qualidade do tráfego VoIP.*

A cada nível de sinal, testes de qualidade de voz foram realizados sem interferência *Bluetooth* e com um, dois, três e quatro fontes interferentes *Bluetooth* ativas. Foram realizados também testes utilizando um enlace cabeado Ethernet entre o *desktop* e o *laptop*. Os resultados desses testes foram utilizados como base de comparação.

Os dispositivos VoIP conectados aos computadores na Figura 6.1, fazem a amostragem e compressão de uma fonte de áudio analógica conectada em suas entradas. Os computadores, por sua vez, executando *softwares* VoIP, empacotam os dados recebidos dos dispositivos VoIP e os enviam pela rede até o seu destino. Nos experimentos realizados em [2], todo o tráfego VoIP era originado no *desktop* e destinado ao *laptop*, apesar de o enlace estabelecido ser bidirecional. Para realizar os testes, quarenta transmissões foram efetuadas para cada cenário de teste. A entrada de áudio fornecida ao

dispositivo VoIP conectado ao *desktop* era simultaneamente gravada. Da mesma forma, era gravada também a saída do dispositivo VoIP conectado ao *laptop*. Assim, obteve-se quarenta pares de entrada e saída do sistema VoIP para cada condição de interferência.

Para realizar as avaliações da qualidade do tráfego VoIP, foi utilizado em [2] um algoritmo chamado de PESQ (*Perceptual Evaluation of Speech Quality*). De acordo com [2], o PESQ consegue, com um alto grau de precisão, emular a percepção humana de qualidade de voz na escala MOS (*Mean Opinion Score*). O PESQ foi aplicado para cada par de entrada/saída com o objetivo de gerar um MOS para cada sentença.

Para criar uma fonte interferente *Bluetooth* de maneira constante, uma transferência de arquivo foi realizada entre um par de dispositivos *Bluetooth*. O dispositivo que enviava o arquivo foi colocado a 1 metro de distância do laptop. Foram utilizados até quatro pares de transceptores *Bluetooth*. No experimento, todas as opções de criptografia foram desabilitadas para permitir que os dispositivos operassem em suas taxas máximas.

O experimento realizado em [2] demonstrou que a QoS do tráfego VoIP na rede 802.11b variou bastante com o nível de sinal da rede e com o número de dispositivos *Bluetooth* interferentes. Sem nenhuma interferência *Bluetooth*, o índice MOS da transmissão VoIP para níveis de relação sinal/ruído de 20 dB e 15 dB ficou praticamente idêntico, ficando próximo do índice obtido para a rede Ethernet. Para uma relação sinal/ruído de 10 dB, os resultados mostraram que mesmo sem interferência *Bluetooth* a QoS ficou prejudicada.

Para um ou dois dispositivos *Bluetooth* interferentes, a QoS do tráfego VoIP ficou bastante similar para as relações sinal/ruído de 20 dB e 1 dB. Porém, quando três ou quatro fontes interferentes foram ativadas, a SNR (*Signal to Noise Ratio*) de 20 dB apresentou os melhores resultados. Os resultados das transmissões VoIP para uma SNR de 10 dB foram piores quando comparados as obtidos para as SNRs de 20 dB e 15 dB.

Para as relações sinal/ruído de 15 e 20 dB e para dois ou mais dispositivos *Bluetooth* ativos, os resultados obtidos em [2] demonstraram que a qualidade da comunicação VoIP da rede 802.11b apresentou queda significativa, ao ser comparada com a qualidade para a rede cabeada. O mesmo se aplica quando qualquer fonte interferente *Bluetooth* estava ativa para uma SNR de 10 dB.

Dessa forma, de acordo com [2], as redes 802.11b suportam o tráfego VoIP quando não existe interferência *Bluetooth*. A queda no desempenho da rede se tornou um

fato a ser considerado à medida que iam sendo ativadas as fontes *Bluetooth* interferentes, tornando o tráfego VoIP inviável ao se aumentar o nível de interferência.

6.3 Interferência 802.11g e *Bluetooth*

A coexistência entre redes 802.11g e dispositivos *Bluetooth* interferentes é analisada em Wong e O'Farell [39]. O objetivo desse estudo é verificar a cobertura do AP da WLAN 802.11g quando sua estação móvel está submetida à interferência *Bluetooth*, sendo proposto o uso de apagamentos para reduzir o impacto dessa interferência. De acordo com [39], o impacto da interferência depende da utilização e da distância entre ambos os dispositivos. Para se estudar o problema da coexistência entre os dois sistemas, a topologia ilustrada na Figura 6.2 foi proposta.

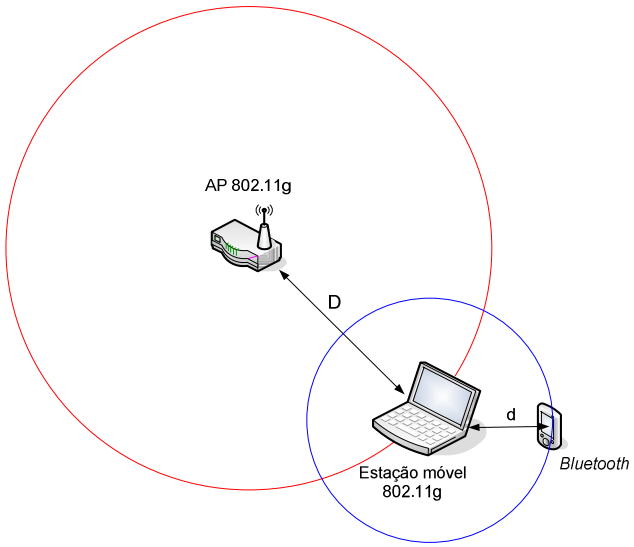


Figura 6.2 – *Topologia de teste implementada por Wong e O'Farell [39].*

No cenário mostrado na Figura 6.2, a estação móvel 802.11g está associada a um AP fixo e é assumido que existe uma *piconet Bluetooth* causando interferência na WLAN. O nível de interferência *Bluetooth* ao qual a estação 802.11g está submetida depende da distância entre esta e o AP, D (m), como também da distância entre o dispositivo *Bluetooth* e a estação móvel 802.11g, d (m).

Segundo [39], mesmo a WLAN tendo potência de transmissão mais elevada (20 mW) do que o dispositivo *Bluetooth* (0 mW), o sinal WLAN cai consideravelmente devido a perdas por propagação quando a estação 802.11g está uma distância suficiente do AP. Dessa forma, o sinal WLAN pode ser severamente degradado pelo sinal *Bluetooth*.

Primeiramente, foi analisada a cobertura do AP quando este estava transmitindo pacotes de 100 bytes sem interferência *Bluetooth*. A potência média de transmissão era de

20 mW e a meta era obter uma PER de 0,01. A intensidade do sinal recebido era medida em função da distância, e à medida que a estação móvel 802.11g se distanciava do AP, a SNR decrescia devido às perdas por propagação. Foi verificado que a cobertura do AP caía 50% à medida que a taxa de transmissão aumentava de 6 Mbps até 54 Mbps.

Após essa primeira avaliação, foi verificada a cobertura do AP quando submetido à interferência *Bluetooth*, com potência média interferente era de 1 mW. Uma transmissão de voz (enlace SCO) foi estabelecida entre o mestre e o escravo. Este é o pior cenário, pois um enlace SCO utiliza toda a capacidade *Bluetooth*, sendo requerida para a transmissão 100% dos *slots* de tempo e uma taxa de saltos de 1.600 saltos/s.

Na análise realizada em [39], a interferência *Bluetooth* foi medida em termos da SNR na estação móvel 802.11g, que implicitamente leva em consideração as distâncias *D*, entre o roteador *Wi-Fi* e o receptor *Wi-Fi* (*laptop*) e *d*, entre o *laptop* e o equipamento *Bluetooth*. Foi determinada a menor SNR requerida para a estação móvel 802.11g de modo que seja atingida uma PER de 0,01. Então, a cobertura do AP, com a distância *d* sendo um parâmetro, foi avaliada. À medida que o dispositivo *Bluetooth* se distancia ou se aproxima da estação móvel, a cobertura do AP se expande ou se reduz de maneira a manter a relação sinal/ruído mínima.

Os resultados obtidos em [39] demonstraram que a cobertura do AP varia de acordo com a distância entre a estação móvel e o dispositivo *Bluetooth* interferente. Foi constatado que quanto maior a distância *d* entre a estação 802.11g e o dispositivo *Bluetooth*, maior era a cobertura do AP. Outro fato foi observado em relação ao número de apagamentos empregados. Verificou-se que quanto maior o número de apagamentos, mais rapidamente o AP atinge sua cobertura máxima para uma dada taxa de transmissão. Entretanto, existe um limite com relação ao número de apagamentos implementados, acima do qual não se verificam melhorias em relação à área de cobertura do AP, podendo ocorrer até mesmo uma queda nessa área.

6.4 Evitando interferências WLAN/*Bluetooth*

Algumas soluções propostas e implementadas para eliminar o problema da interferência entre os sistemas WLAN e *Bluetooth* são descritas por Rukh [40]. Essas soluções são classificadas nas duas seguintes categorias:

- mecanismos colaborativos;
- mecanismos não-colaborativos.

Nos mecanismos colaborativos, os dispositivos *Bluetooth* e WLAN são forçados a conversarem uns com os outros de modo a determinar qual rede tem permissão para transmitir, em qual freqüência e canal. Dessa forma, os sistemas *Bluetooth* e WLAN precisam trocar mensagens ao acessarem o meio de comunicação. As soluções colaborativas incluem PTA (*Packet Traffic Arbitration*) e AWMA (*Alternating Wireless Medium Access*).

O PTA é um algoritmo dinâmico de agendamento de tráfego que é mais bem implementado quando ambas as tecnologias *Bluetooth* e WLAN estão localizadas no mesmo dispositivo. O PTA utiliza uma entidade de controle que implementa um *handshake* entre as camadas MAC dos dois sistemas. O PTA previne colisões utilizando algo parecido com um sinal de trânsito para o tráfego de pacotes 802.11 e *Bluetooth*. Segundo [40], esse algoritmo são seria uma boa escolha quando os sistemas estão implementados em dispositivos distintos. A Figura 6.3 ilustra o controle de tráfego do PTA.

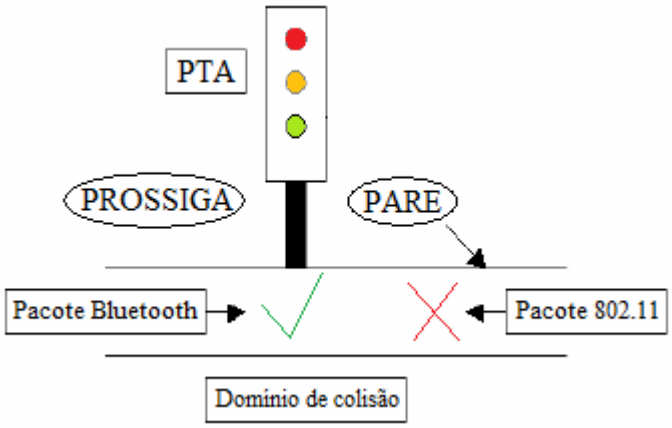


Figura 6.3 – *Controle de tráfego do PTA para o Bluetooth e para a WLAN (Modificada de [40]).*

O método AWMA aloca períodos de tempo para os sistemas WLAN e *Bluetooth*, forçando cada um a transmitir somente nos seus respectivos intervalos de tempo. De acordo com [40], esse método funciona bem mesmo se os rádios dos dois sistemas estiverem no mesmo dispositivo. Os rádios também podem estar separados por apenas alguns centímetros. A Figura 6.4 mostra o método AWMA.

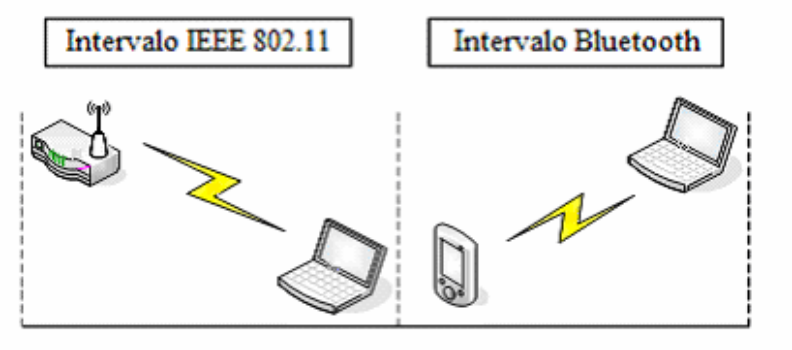


Figura 6.4 – *WLAN e Bluetooth utilizando AWMA (Modificada de [40]).*

Segundo [40], o AWMA aplica-se apenas a enlaces ACL, não sendo útil para enlaces SCO. Isso se deve ao fato de que enlaces SCO são regulares e têm um período bastante curto (3,75 ms), tomando toda a capacidade do meio para si. Isso dificulta o encaixe de pacotes 802.11 entre os pacotes SCO. No sistema WLAN, o mecanismo AWMA pode ser utilizado enviando-se, por meio do AP, um sinal CTS ao final do intervalo de transmissão da rede 802.11. Com isso, todos os rádios 802.11 não irão transmitir durante o intervalo de tempo especificado no quadro CTS.

Os mecanismos não-colaborativos requerem que os dispositivos *Bluetooth* e WLAN tomem medidas independentes para evitar interferência. As soluções não-colaborativas incluem AFH (*Adaptive Frequency Hopping*), LBT (*Listen-Before-Talk*) e *Adaptive Packet Selection and Scheduling*.

O algoritmo AFH muda dinamicamente a seqüência de saltos do dispositivo, restringindo o número de canais pelos quais o nó *Bluetooth* transmite. O mecanismo AFH para o *Bluetooth* pode ser dividido em quatro componentes principais:

- **classificação de canal.** Classifica os canais em bom ou ruim de acordo com nível de interferência;
- **gerenciamento de enlace.** Coordena e distribui as informações AFH para todos os dispositivos *Bluetooth* da rede;
- **modificação da seqüência de salto.** Reduz, seletivamente, o número de canais pelos quais o sistema saltará de modo a evitar canais ruins;
- **manutenção de canal.** Reavalia periodicamente a qualidade dos canais para lidar com situações não previstas.

A Figura 6.5 ilustra o mecanismo AFH, no qual os canais sendo utilizados pela WLAN são considerados ruins.

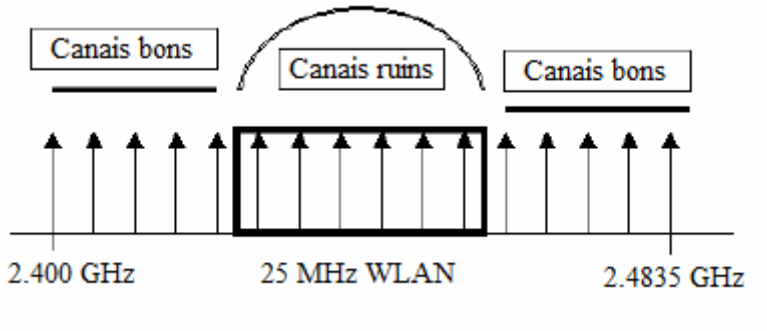


Figura 6.5 – *Mecanismo AFH*
(Modificada de [40]).

O mecanismo LBT requer que o dispositivo *Bluetooth* escute o canal antes de transmitir. Ao aguardar que transmissões de outras fontes terminem ou verificando que o canal está ocupado, o LBT é capaz de evitar colisões. A Figura 6.6 mostra o mecanismo LBT.

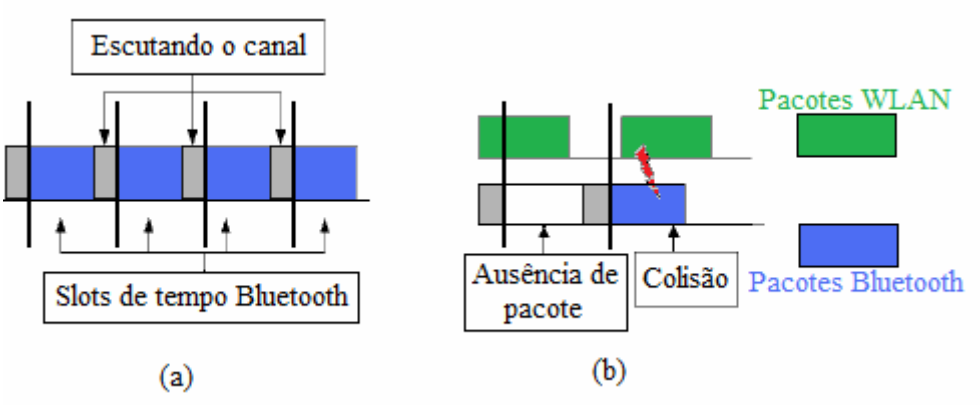


Figura 6.6 – *(a) Mecanismo LBT com Bluetooth (b) Colisão devido a uma previsão incorreta por parte do LBT (Modificada de [40]).*

A Figura 6.6 (a) mostra os *slots* de tempo do sistema *Bluetooth*, os quais possuem duração de 625 µs. Destes, 366 µs são utilizados para transmissão de pacotes *Bluetooth* (sinalizados na cor azul), restando 259 µs para sintonização dos rádios (sinalizados na cor cinza). O mecanismo LBT utiliza justamente esses 259 µs para escutar o canal.

Conforme consta em [40], a falha do LBT reside no fato de que ele não prevê o futuro, podendo ocorrer colisões entre os sistemas *Bluetooth* e WLAN. A Figura 6.6 (b) ilustra uma situação de colisão entre os dois sistemas. Primeiramente, o dispositivo *Bluetooth* identificou, no final de um período de tempo anterior, que já havia uma transmissão WLAN em andamento, abstendo-se de transmitir no período seguinte. No final desse *slot* de tempo, o dispositivo *Bluetooth* verificou que o canal estava ocioso e iniciou uma transmissão no próximo *slot*. Entretanto, o LBT não contava que o sistema 802.11 iria

iniciar outra transmissão logo após o início da transmissão *Bluetooth*, resultando em colisão.

O último mecanismo não-colaborativo descrito em [40] para evitar colisões é o *Adaptive Packet Selection and Scheduling*. Esse mecanismo é um melhoramento na camada MAC *Bluetooth* que utiliza uma tabela para fazer um registro estatístico dos canais nos quais ocorreram interferências. Essa tabela é acessada freqüentemente de modo que as transmissões ocorram apenas quando o salto para um bom canal tenha acontecido.

6.5 Interferência WLAN 802.11b/*Bluetooth*

No trabalho realizado por Jo e Jayant [38], foi avaliado o desempenho de uma WLAN 802.11b composta por um AP e múltiplas estações sem fio sob a presença de interferência *Bluetooth*. O *throughput* do sistema 802.11b foi estimado não apenas como função dos parâmetros da WLAN, mas também como função dos parâmetros do sistema interferente *Bluetooth*. Para poder avaliar o nível de degradação causada pela interferência *Bluetooth* na rede 802.11b, em [38] foi obtido o *throughput* teórico máximo da WLAN em condições perfeitas de canal para posterior comparação com o *throughput* em outros cenários. Nas análises realizadas, foi assumido que a *piconet Bluetooth* transmitia pacotes com comprimento de apenas 1 *slot* de tempo, pois esse é o pior caso, já que pacotes que ocupem múltiplos *slots* fazem com que a *piconet* tenha uma taxa de saltos mais baixa.

Segundo [38], o número máximo de *slots* de tempo *Bluetooth* que podem colidir com um pacote 802.11b são três. Colisões na freqüência ocorrem quando os canais de transmissão utilizados pela *piconet Bluetooth* causam erro de pacote na WLAN. Porém, ainda de acordo com [38], se a relação sinal/ruído do sistema for maior do que 10 dB, a rede 802.11b pode fornecer um serviço confiável mesmo sob interferência *Bluetooth*.

Com o objetivo de obter resultados numéricos, em [38] foi considerado um número finito de estações sem fio distribuídas de maneira aleatória na BSS. A topologia de estudo implementada foi conforme ilustra a Figura 6.7.

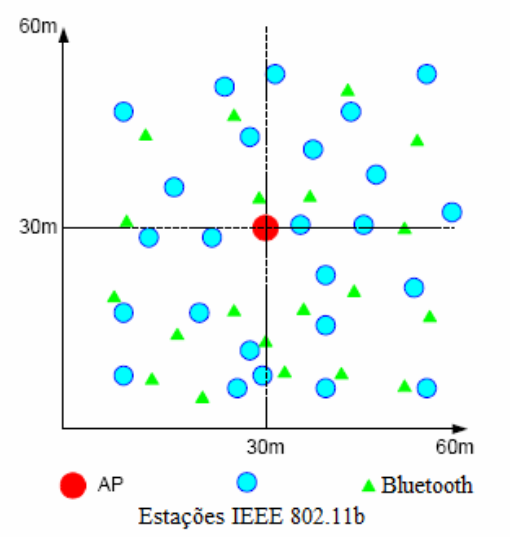


Figura 6.7 – *Topologia de análise implementada Jo e Jayant [38].*

Conforme ilustrado na Figura 6.7, foi utilizado um AP, 25 estações sem fio IEEE 802.11b e várias *piconets Bluetooth* distribuídas de maneira uniforme por uma BSS com área de cobertura igual a 60 m^2.

Dois cenários de interferência *Bluetooth* foram implementados: um com tráfego *Bluetooth* menos intenso e outro com tráfego mais intenso, inclusive com uma densidade maior de *piconets* por m^2.

Os resultados obtidos em [38] mostraram que para uma taxa de transmissão de 2 Mbps e uma carga útil de 100 bytes, o *throughput* do sistema 802.11b é maior quando não se utilizam os quadros de controle RTS/CTS. Isso acontece para ambos os cenários *Bluetooth* implementados. Para uma carga útil de 1.000 bytes ocorreu justamente o oposto, ou seja, a rede 802.11b apresentou um *throughput* mais elevado quando foram utilizados os quadros RTS e CTS para os dois cenários *Bluetooth*.

Para uma taxa de transmissão de 11 Mbps, os resultados mostraram que, tanto para uma carga útil de 1.000 bytes quanto para uma de 1.500 bytes, o *throughput* da rede 802.11b foi um pouco melhor no modo básico (sem os quadros RTS/CTS) do que no modo com os quadros de controle.

Como resultado final, constatou-se em [38] que o *throughput* da WLAN 802.11b degradou-se em 25% e 66% para um cenário de tráfego *Bluetooth* menos e mais intenso respectivamente, independentemente da taxa de transmissão da WLAN.

Capítulo 7. DESCRIÇÃO DOS EXPERIMENTOS REALIZADOS

O estudo apresentado neste trabalho dissertativo tem o objetivo de analisar quantitativamente a QoS da transmissão de uma massa de dados em uma rede IEEE 802.11g quando submetida a interferência *Bluetooth*.

Para isso, foi realizada a transmissão de um arquivo de vídeo com tamanho aproximado de 715 MB entre uma fonte e um destino na rede 802.11g em dois cenários distintos: um livre da interferência *Bluetooth* e outro sob interferência *Bluetooth*. A interferência *Bluetooth* era causada pela transferência de uma massa de dados (também um vídeo) entre dois telefones celulares posicionados próximos um ao transmissor 802.11g e outro ao receptor 802.11g.

A primeira bateria de testes foi feita para uma distância entre os transmissores e receptores 802.11g e *Bluetooth* de 1,60 m, alternando-se entre uma transmissão *Wi-Fi* sem interferência e outra com interferência *Bluetooth*. O motivo de se ter alternado entre os dois cenários foi para fazer com que cada um deles presenciasse condições de rede as mais próximas possíveis.

A segunda bateria de testes foi feita para uma distância de 2,60 m entre os transmissores e receptores 802.11g e *Bluetooth*, alternando-se entre uma transmissão sem interferência e outra com interferência *Bluetooth*.

A terceira bateria de testes foi realizada a uma distância de 4,60 m entre os transmissores e receptores 802.11g e *Bluetooth*. Porém, diferentemente da primeira e segunda bateria de testes, agora existia uma parede de alvenaria entre os transmissores e receptores 802.11g e *Bluetooth*. Além do incremento da distância, o objetivo de se testar a comunicação com a obstrução da parede é analisar o desempenho da rede *Wi-Fi* sem visada direta. Não havia nenhum obstáculo entre os dispositivos de comunicação nas baterias de testes anteriores.

Para cada uma das distâncias analisadas, foram realizados um total de 100 transmissões, sendo 50 sem interferência e 50 com interferência. Dessa forma, no total foram realizadas 300 transmissões na rede 802.11g.

Ao final de todos os testes realizados, foram levantados cinco parâmetros para avaliação da QoS, que foram:

- tempo de transmissão;
- taxa média de transmissão de pacotes (em pacotes/segundo);

- taxa média de transmissão de dados (em Mbps);
- quantidade de pacotes perdidos e
- quantidade de reconhecimentos (ACKs) duplicados.

Por meio da comparação entre os resultados obtidos desses parâmetros, foi possível avaliar o nível de degradação da transmissão de dados em uma rede sem fio 802.11g causada pela fonte interferente *Bluetooth*.

7.1 Ambiente de testes

Para a avaliação da QoS da transmissão de dados na rede IEEE 802.11g, foram implementadas nesta dissertação as topologias de rede ilustradas nas Figuras 7.1 e 7.2.

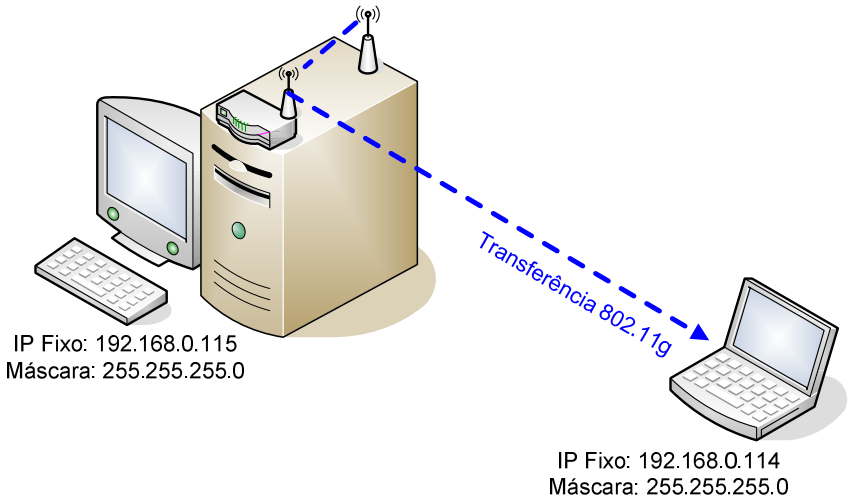


Figura 7.1 – *Topologia de análise sem interferência Bluetooth.*

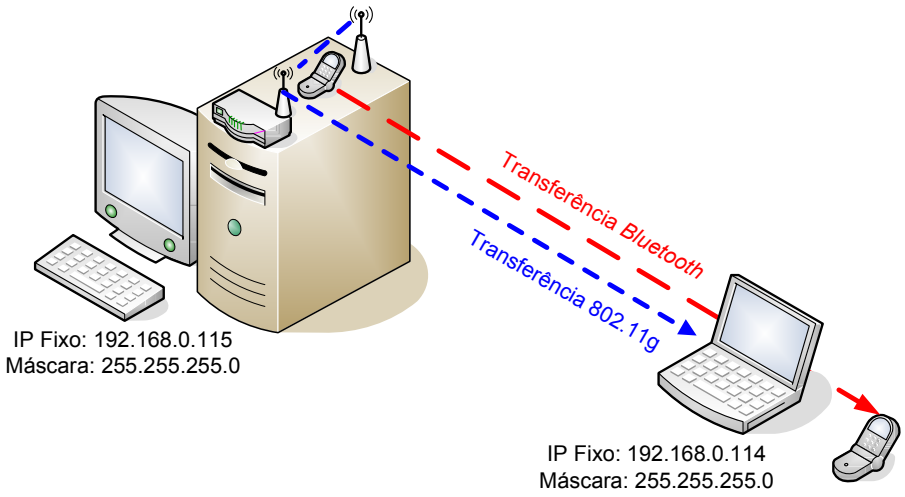


Figura 7.2 – *Topologia de análise com interferência Bluetooth.*

Conforme pode ser visualizado na Figura 7.1, não há interferência *Bluetooth* na rede *Wi-Fi* durante a transmissão da massa de dados do *desktop* para o *laptop*. Já na Figura 7.2, existe uma fonte interferente *Bluetooth*.

Para gerar a interferência *Bluetooth* constante durante toda a transmissão 802.11g entre o *desktop* e o *laptop*, transferiu-se, conforme ilustra a Figura 7.2, uma massa de dados entre dois telefones celulares com *Bluetooth* habilitado. Em todos os testes realizados, a transmissão 802.11g ocorre do *desktop* para o *laptop* por meio de um enlace totalmente sem fio.

A Tabela 7.1 mostra as especificações das máquinas associadas ao BSS.

Tabela 7.1 – *Configurações dos computadores envolvidos nos testes.*

	Desktop	***Laptop***
Sistema Operacional	*Windows* XP *Professional*	*Windows Vista Home Premium*
Processador	AMD Athlon (tm) 1,10 GHz	Intel Core 2 Duo T5450 1,67 GHz
Adaptador de rede sem fio	Adaptador USB DWL – G122	Adaptador PCMCIA Atheros AR5007EG

Conforme pode ser observado pela Tabela 7.1, o *laptop* possui maior capacidade de processamento e de armazenamento de dados. Por esse motivo, essa máquina foi a escolhida como a receptora da massa de dados 802.11g, sendo realizado nela todo o processamento para a obtenção dos resultados.

Durante todas as 300 transmissões realizadas, procurou-se manter as mesmas condições de testes. Para tanto, os dispositivos envolvidos nas transferências foram sempre dispostos nas mesmas posições, os móveis dentro do ambiente sempre estavam também nos mesmos lugares e nenhum outro dispositivo, além dos envolvidos nos experimentos, estava ativo dentro do ambiente de testes.

Ambas as estações e o AP pertencem a um mesmo BSS. O modelo, a potência de transmissão e as configurações básicas utilizadas pelo AP são as seguintes:

- modelo: D-Link DI – 524 802.11g/2,4 GHz *Wireless Router*;
- potência de transmissão: 15 dBm (aproximadamente 32 mW);
- SSID: Vitória;
- canal: 1 (2,412 GHz);
- criptografia: WPA-PSK / WPA2-PSK;
- intervalo *beacon*: 100 ms (conforme configuração padrão);

- taxa de transmissão: 54 Mbps;
- *broadcast* de SSID: habilitado (conforme configuração padrão).

A escolha do canal para ser utilizado pela rede *Wi-Fi* foi baseada no fato de que, conforme já exposto nesta dissertação na subseção 2.2, os únicos três canais que não se sobrepõe são os canais 1, 6 e 11. Como o canal 6 já vêm configurado como padrão nos roteadores, decidiu-se não utilizá-lo para evitar ao máximo possíveis interferências de outros BSSs vizinhos que já estivessem utilizando esse canal. Com isso, sobraram os canais 1 e 11, sendo escolhido o canal 1, pois verificou-se que era o menos utilizado por redes *Wi-Fi* vizinhas.

Conforme dito no início deste capítulo, os testes foram realizados para três distâncias distintas entre os transmissores e receptores 802.11 e *Bluetooth*. A Figura 7.3 mostra o posicionamento dos dispositivos no apartamento para as distâncias de 1,60 m, 2,60 m e 4,60 m.

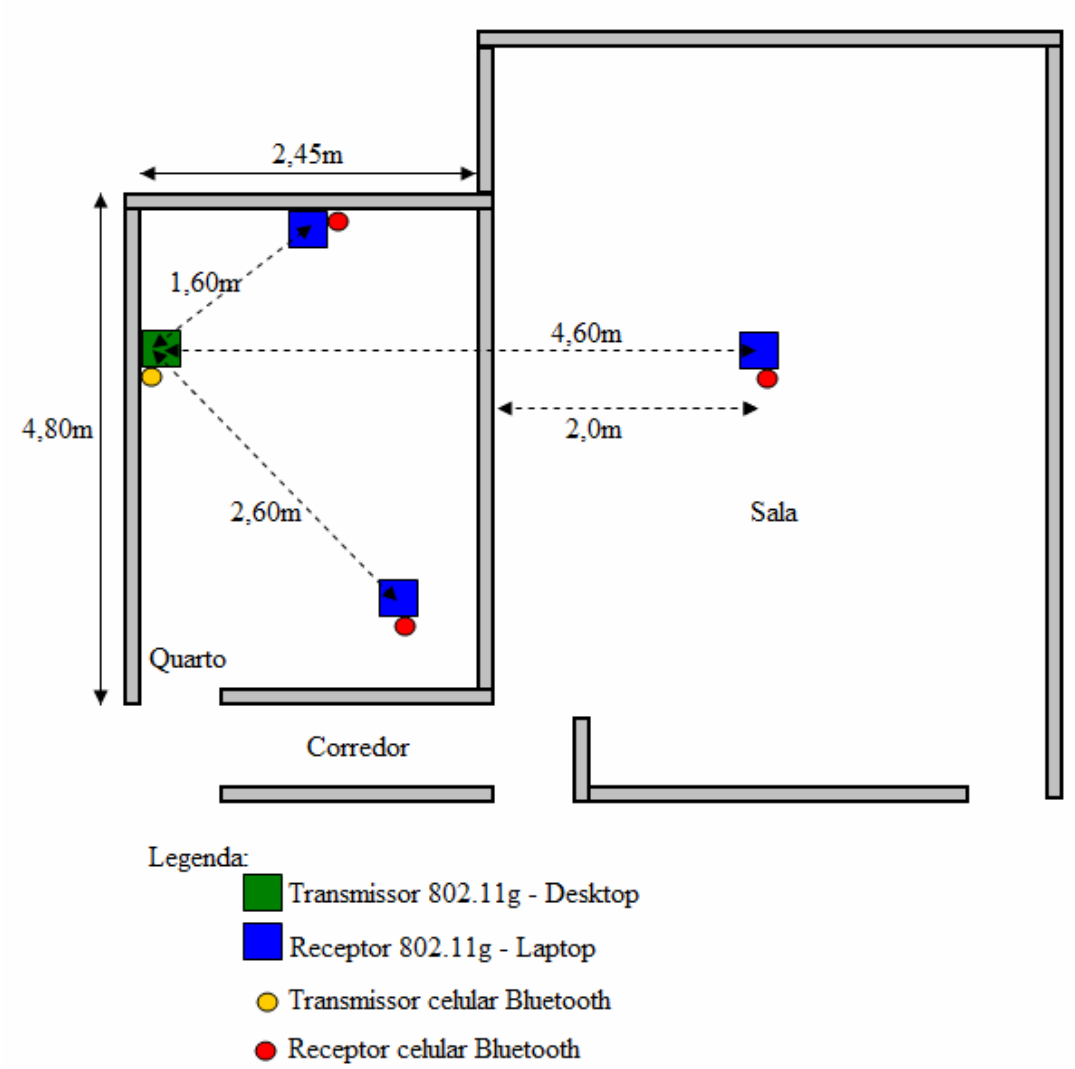


Figura 7.3 – *Posicionamento dos dispositivos no ambiente de testes.*

Como pode ser visto na Figura 7.3, para a distância de testes de 4,60 m, existe uma parede de alvenaria entre os dispositivos de comunicação. A parede entre o quarto e a sala tem aproximadamente 15 cm de espessura.

A Figura 7.4 mostra o posicionamento dos dispositivos transmissores de dados *Wi-Fi* e *Bluetooth*.

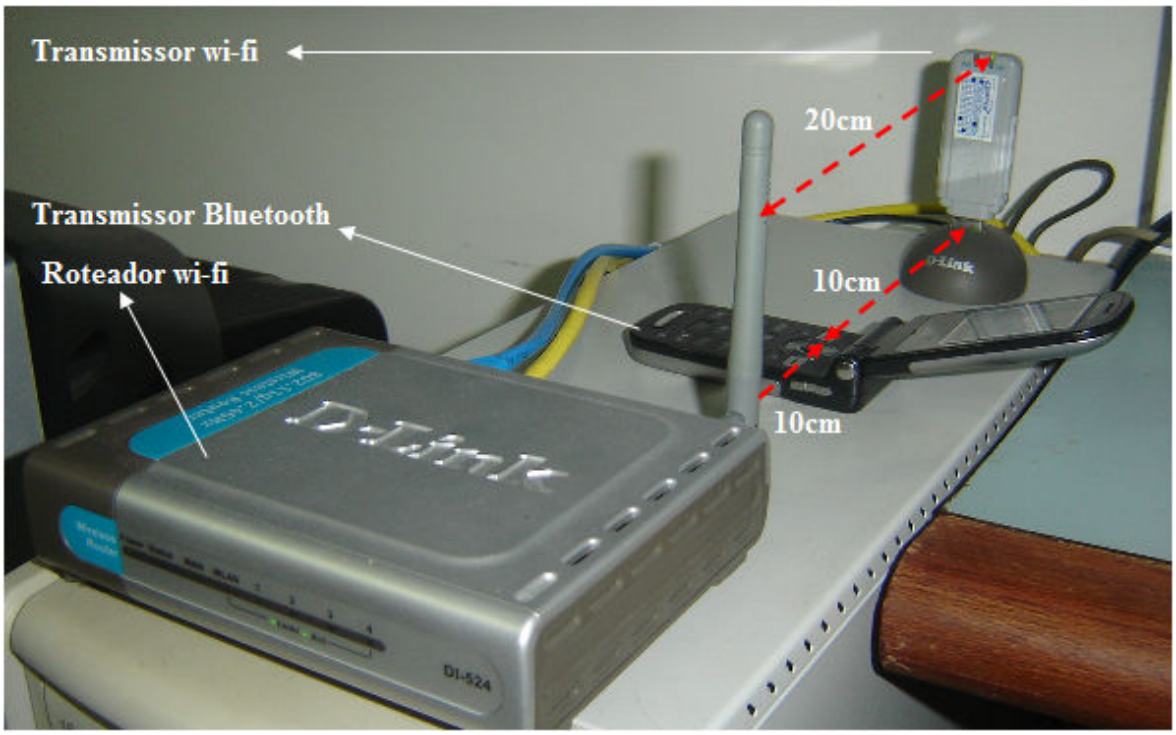


Figura 7.4 – *Ilustração do posicionamento dos dispositivos de transmissão.*

A distância entre o transmissor *Wi-Fi* e o roteador é de 20 cm. O dispositivo transmissor *Bluetooth* foi posicionado entre o transmissor *Wi-Fi* e o roteador, a 10 cm de distância de cada um. Esse posicionamento foi mantido para todas as distâncias testadas.

A Figura 7.5 mostra a disposição dos dispositivos de comunicação para a distância de 1,60 m.

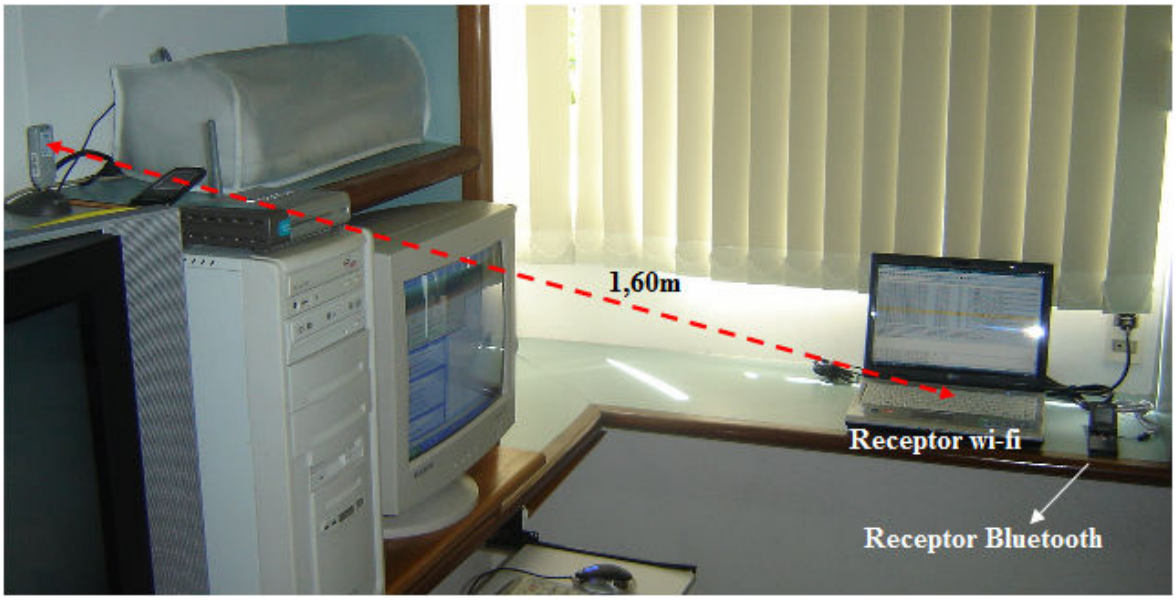


Figura 7.5 – *Ilustração do posicionamento dos dispositivos de comunicação para 1,60 m.*

A Figura 7.6 mostra com mais detalhe o posicionamento dos receptores *Wi-Fi* e *Bluetooth* para a distância de 1,60m.

Figura 7.6 – *Ilustração do posicionamento dos dispositivos de recepção para 1,60 m.*

O posicionamento ilustrado na Figura 7.6 foi mantido para as distâncias de 2,60 m e 4,60 m.

A Figura 7.7 mostra a disposição dos dispositivos para a distância de 2,60 m.

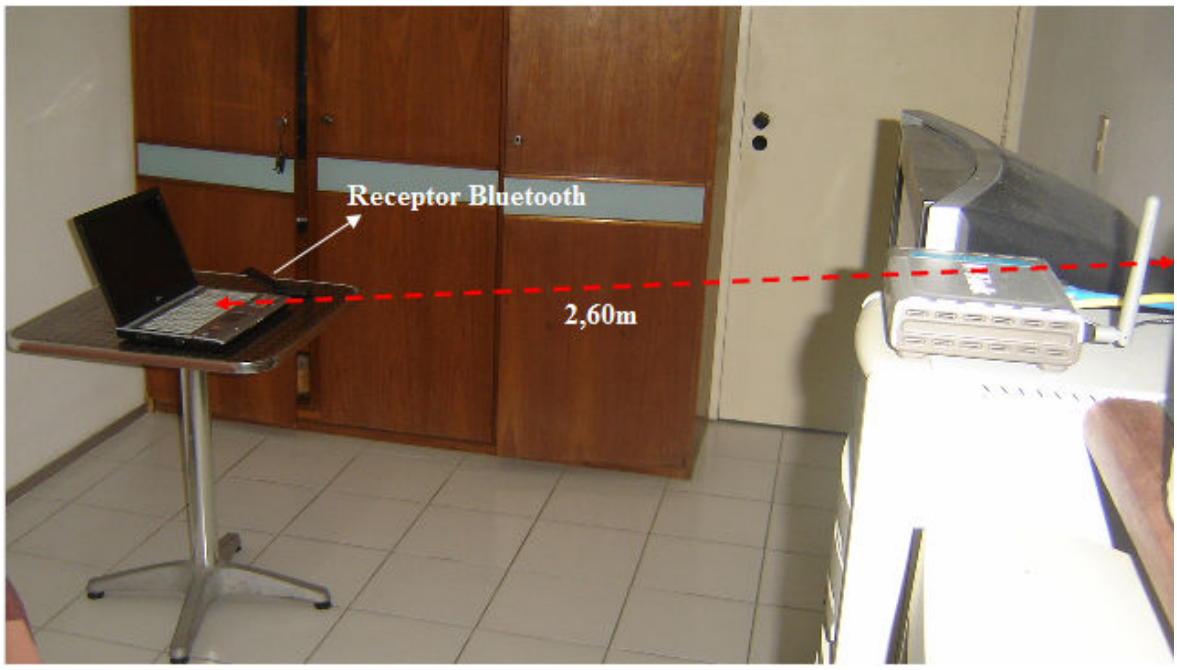


Figura 7.7 – *Ilustração do posicionamento dos dispositivos de comunicação para 2,60 m.*

Conforme pode ser visto nas Figuras 7.5 e 7.7, não há obstáculos posicionados entre os transmissores e os receptores, estando em visada direta um do outro.

A Figura 7.8 mostra o posicionamento dos equipamentos de recepção para a distância de 4,60 m.

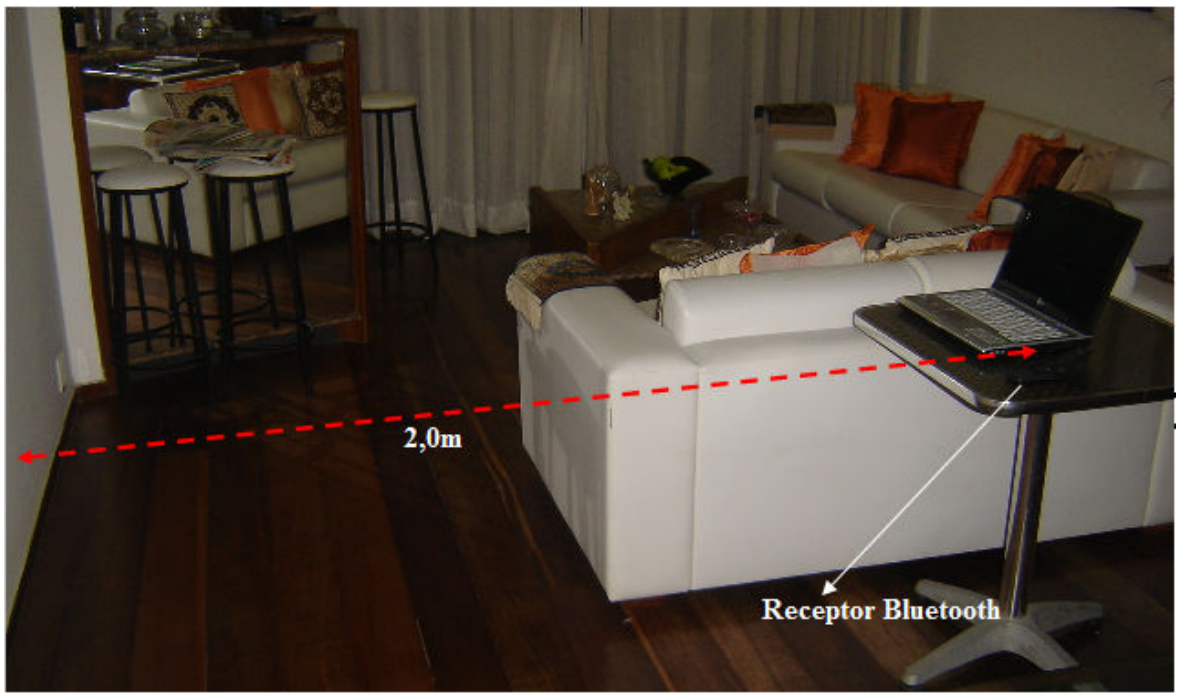


Figura 7.8 – *Ilustração do posicionamento dos dispositivos de recepção para 4,60 m.*

Conforme mostrado na foto da Figura 7.8, existe uma parede entre os equipamentos de transmissão e recepção, não havendo visada direta entre eles. Os sinais são recebidos via reflexões e propagação multi-percurso.

Nos testes realizados para as distâncias de 1,60 m e 2,60 m, foi assumido que os pacotes *Bluetooth* possuem comprimento máximo de 5 *slots* de tempo. Para a distância de 4,60m, foi assumido que os pacotes *Bluetooth* têm comprimento mínimo de 1 *slot* de tempo. Essas suposições foram baseadas no tempo de transmissão do arquivo de vídeo *Bluetooth*. Enquanto que para as distâncias de 1,60 m e 2,60 m o tempo de transmissão era cerca de 30 a 35 minutos, para a distância de 4,60 m, esse tempo era de aproximadamente 1 hora e 30 minutos.

Como se trata de uma transmissão que não é de tempo real, o enlace estabelecido entre o mestre e o escravo *Bluetooth* é um ACL. Vale salientar aqui que o pior cenário de interferência para a rede *Wi-Fi* é para a distância de 4,60 m. Isso porque, além do obstáculo da parede, [7] os pacotes *Bluetooth* com comprimento de 1 *slot* de tempo fazem com que a freqüência de saltos do sistema FHSS seja maior (1.600 saltos/s), aumentando a probabilidade de colisões. No próximo capítulo são apresentados os resultados obtidos para tais experimentos.

Capítulo 8. ANÁLISE DOS RESULTADOS DOS EXPERIMENTOS

Conforme exposto no início do capítulo 7, foram levantados cinco parâmetros para possibilitar a avaliação da QoS da transmissão de dados em uma rede 802.11g quando submetida a interferência *Bluetooth*. Relembrando, esses parâmetros são:

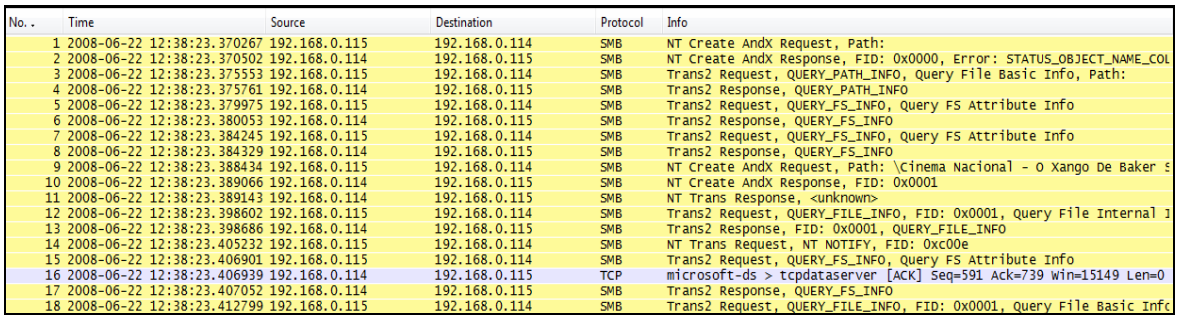
- tempo de transmissão;
- taxa média de transmissão de pacotes (em pacotes/segundo);
- taxa média de transmissão de dados (em Mbps);
- quantidade de pacotes perdidos e
- quantidade de reconhecimentos (ACKs) duplicados.

Os dados especificados acima foram extraídos por meio da captura dos pacotes realizada utilizando-se o *software Wireshark* versão 1.0.2 instalado no *laptop*. Esse software é gratuito e pode ser encontrado em [41].

Para que as estatísticas sobre todos os testes nos dois cenários fossem geradas sempre sobre a mesma quantidade de pacotes, o *Wireshark* foi configurado de tal maneira que quando ele atingisse a quantidade de 775.211 pacotes capturados, a captura fosse interrompida. Foi estipulado esse número porque se verificou que essa quantidade era suficiente para concluir a transferência do arquivo de vídeo selecionado para os testes.

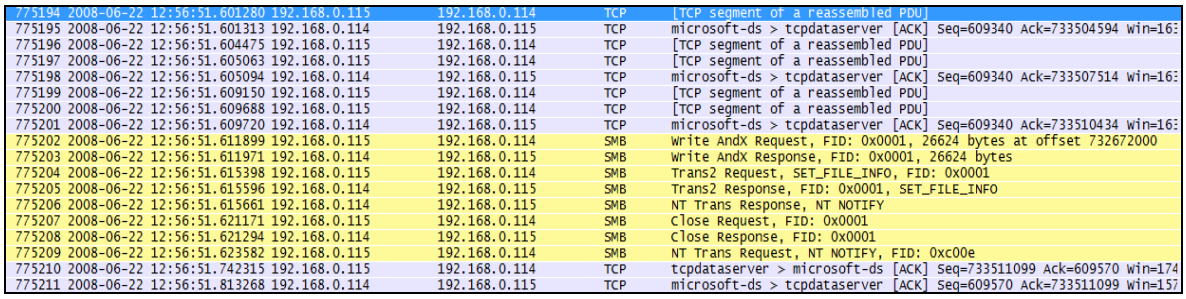
Outra configuração necessária na utilização do *Wireshark* para realizar os testes foi criar um filtro para que o *software* capture apenas os pacotes que tenham como origem (ou destino) o IP 192.168.0.115. Dessa maneira, evitou-se que fossem capturados pacotes de *broadcast*, como, por exemplo, pacotes ARP.

As Figuras 8.1 e 8.2 ilustram o início e o término, respectivamente, de uma das 50 capturas realizadas no cenário sem interferência *Bluetooth*.



No. .	Time	Source	Destination	Protocol	Info
1	2008-06-22 12:38:23.370267	192.168.0.115	192.168.0.114	SMB	NT Create AndX Request, Path:
2	2008-06-22 12:38:23.370502	192.168.0.114	192.168.0.115	SMB	NT Create AndX Response, FID: 0x0000, Error: STATUS_OBJECT_NAME_COL
3	2008-06-22 12:38:23.375553	192.168.0.115	192.168.0.114	SMB	Trans2 Request, QUERY_PATH_INFO, Query File Basic Info, Path:
4	2008-06-22 12:38:23.375761	192.168.0.114	192.168.0.115	SMB	Trans2 Response, QUERY_PATH_INFO
5	2008-06-22 12:38:23.379975	192.168.0.115	192.168.0.114	SMB	Trans2 Request, QUERY_FS_INFO, Query FS Attribute Info
6	2008-06-22 12:38:23.380053	192.168.0.114	192.168.0.115	SMB	Trans2 Response, QUERY_FS_INFO
7	2008-06-22 12:38:23.384245	192.168.0.115	192.168.0.114	SMB	Trans2 Request, QUERY_FS_INFO, Query FS Attribute Info
8	2008-06-22 12:38:23.384329	192.168.0.114	192.168.0.115	SMB	Trans2 Response, QUERY_FS_INFO
9	2008-06-22 12:38:23.388434	192.168.0.115	192.168.0.114	SMB	NT Create AndX Request, Path: \Cinema Nacional - O Xango De Baker S
10	2008-06-22 12:38:23.389066	192.168.0.114	192.168.0.115	SMB	NT Create AndX Response, FID: 0x0001
11	2008-06-22 12:38:23.389143	192.168.0.114	192.168.0.115	SMB	NT Trans Response, <unknown>
12	2008-06-22 12:38:23.398602	192.168.0.114	192.168.0.115	SMB	Trans2 Request, QUERY_FILE_INFO, FID: 0x0001, Query File Internal I
13	2008-06-22 12:38:23.398686	192.168.0.114	192.168.0.115	SMB	Trans2 Response, FID: 0x0001, QUERY_FILE_INFO
14	2008-06-22 12:38:23.405232	192.168.0.115	192.168.0.114	SMB	NT Trans Request, NT NOTIFY, FID: 0xc00e
15	2008-06-22 12:38:23.406901	192.168.0.115	192.168.0.114	SMB	Trans2 Request, QUERY_FS_INFO, Query FS Attribute Info
16	2008-06-22 12:38:23.406939	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=591 Ack=739 Win=15149 Len=0
17	2008-06-22 12:38:23.407052	192.168.0.114	192.168.0.115	SMB	Trans2 Response, QUERY_FS_INFO
18	2008-06-22 12:38:23.412799	192.168.0.115	192.168.0.114	SMB	Trans2 Request, QUERY_FILE_INFO, FID: 0x0001, Query File Basic Info

Figura 8.1 – *Início de captura dos pacotes.*



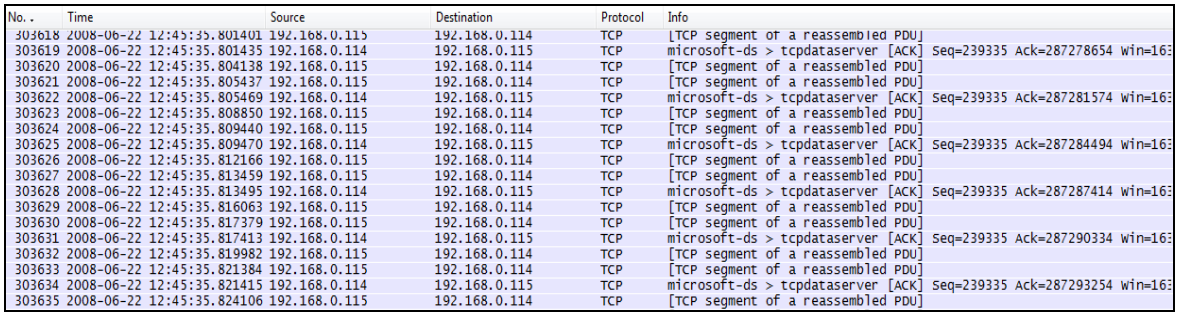
775194	2008-06-22 12:56:51.601280	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
775195	2008-06-22 12:56:51.601313	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=609340 Ack=733504594 Win=16
775196	2008-06-22 12:56:51.604475	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
775197	2008-06-22 12:56:51.605063	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
775198	2008-06-22 12:56:51.605094	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=609340 Ack=733507514 Win=16
775199	2008-06-22 12:56:51.609150	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
775200	2008-06-22 12:56:51.609688	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
775201	2008-06-22 12:56:51.609720	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=609340 Ack=733510434 Win=16
775202	2008-06-22 12:56:51.611899	192.168.0.115	192.168.0.114	SMB	Write AndX Request, FID: 0x0001, 26624 bytes at offset 732672000
775203	2008-06-22 12:56:51.611971	192.168.0.114	192.168.0.115	SMB	Write AndX Response, FID: 0x0001, 26624 bytes
775204	2008-06-22 12:56:51.615398	192.168.0.115	192.168.0.114	SMB	Trans2 Request, SET_FILE_INFO, FID: 0x0001
775205	2008-06-22 12:56:51.615596	192.168.0.114	192.168.0.115	SMB	Trans2 Response, FID: 0x0001, SET_FILE_INFO
775206	2008-06-22 12:56:51.615661	192.168.0.115	192.168.0.114	SMB	NT Trans Response, NT NOTIFY
775207	2008-06-22 12:56:51.621171	192.168.0.115	192.168.0.114	SMB	Close Request, FID: 0x0001
775208	2008-06-22 12:56:51.621294	192.168.0.114	192.168.0.115	SMB	Close Response, FID: 0x0001
775209	2008-06-22 12:56:51.623582	192.168.0.115	192.168.0.114	SMB	NT Trans Request, NT NOTIFY, FID: 0xc00e
775210	2008-06-22 12:56:51.742315	192.168.0.115	192.168.0.114	TCP	tcpdataserver > microsoft-ds [ACK] Seq=733511099 Ack=609570 Win=174
775211	2008-06-22 12:56:51.813268	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=609570 Ack=733511099 Win=157

Figura 8.2 – *Término de captura dos pacotes.*

O protocolo SMB (*Server Message Block*), que aparece tanto no início quanto no final da transferência dos dados, é um protocolo de redes cujo uso mais comum é o compartilhamento de arquivos. Esse protocolo permite que o cliente manipule arquivos como se estes estivessem em sua máquina local. O protocolo SMB dá suporte a ações como leitura, escrita, criação, apagamento e renomeação, considerando que os arquivos que estão sendo tratados não estão no computador local, mas sim em um servidor remoto [42]. No caso desta dissertação, o cliente é o *desktop* com IP 192.168.0.115 e o servidor é o *laptop* com IP 192.168.0.114.

Analisando a Figura 8.2, pode-se verificar que o cliente solicita encerramento da conexão (*Close Request*) no pacote número 775.207. O servidor, por sua vez, responde a essa solicitação (*Close Response*) no pacote logo em seguida. A captura é encerrada de maneira automática, com todo o arquivo já transferido, ao se atingir 775.211 pacotes.

A Figura 8.3 ilustra um trecho intermediário durante a transferência do arquivo de vídeo.



No. .	Time	Source	Destination	Protocol	Info
303618	2008-06-22 12:45:35.801401	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303619	2008-06-22 12:45:35.801435	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=239335 Ack=287278654 Win=16
303620	2008-06-22 12:45:35.804138	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303621	2008-06-22 12:45:35.805437	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303622	2008-06-22 12:45:35.805469	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=239335 Ack=287281574 Win=16
303623	2008-06-22 12:45:35.808850	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303624	2008-06-22 12:45:35.809440	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303625	2008-06-22 12:45:35.809470	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=239335 Ack=287284494 Win=16
303626	2008-06-22 12:45:35.812166	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303627	2008-06-22 12:45:35.813459	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303628	2008-06-22 12:45:35.813495	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=239335 Ack=287287414 Win=16
303629	2008-06-22 12:45:35.816063	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303630	2008-06-22 12:45:35.817379	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303631	2008-06-22 12:45:35.817413	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=239335 Ack=287290334 Win=16
303632	2008-06-22 12:45:35.819982	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303633	2008-06-22 12:45:35.821384	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]
303634	2008-06-22 12:45:35.821415	192.168.0.114	192.168.0.115	TCP	microsoft-ds > tcpdataserver [ACK] Seq=239335 Ack=287293254 Win=16
303635	2008-06-22 12:45:35.824106	192.168.0.115	192.168.0.114	TCP	[TCP segment of a reassembled PDU]

Figura 8.3 – *Transferência do arquivo em andamento.*

Na Figura 8.3, os pacotes com origem no IP 192.168.0.115 e destino 192.168.0.114 contêm os dados do arquivo utilizado para os testes. Já os pacotes com sentido inverso, ou seja, com origem no IP 192.168.0.114 e destino 192.168.0.115 são os reconhecimentos (ACKs).

Com exceção da quantidade de pacotes perdidos, todos os outros quatro parâmetros foram obtidos de forma direta do *Wireshark*. Para se determinar o número de pacotes

perdidos em cada transferência efetuada, foi necessário analisar, por meio dos arquivos de *log* gerados pelo *Wireshark*, o número de seqüência dos pacotes que geraram a seguinte mensagem de alerta: TCP *Previous segment lost*.

De acordo com [43], essa mensagem de alerta somente ocorre quando realmente existiu perda de pacotes. Esse fato pôde ser comprovado por meio da análise do número de seqüência de cada um dos pacotes que provocaram essa mensagem de alerta e compará-lo com o número de seqüência que era esperado pelo receptor.

Dois dos mais importantes campos do cabeçalho TCP são o número de seqüência e o número de reconhecimento ACK. Esses dois campos constituem parte fundamental do serviço de transferência confiável de dados no TCP [6].

O número de seqüência para um determinado segmento é o número do primeiro byte desse segmento [6]. Considere, como é o caso deste trabalho dissertativo, que um processo no hospedeiro A queira transmitir uma cadeia de dados para um processo no hospedeiro B por uma conexão TCP. O TCP do hospedeiro A vai, então, implicitamente numerar cada byte dessa cadeia de dados.

Suponha que os dados a serem transmitidos consistam em um arquivo formado por 500.000 bytes, que a carga útil de cada segmento seja de 1.000 bytes e que seja atribuído o número 0 ao primeiro byte da cadeia de dados. Conforme pode ser visualizado na Figura 8.4, o TCP constrói 500 segmentos para a seqüência de informações.

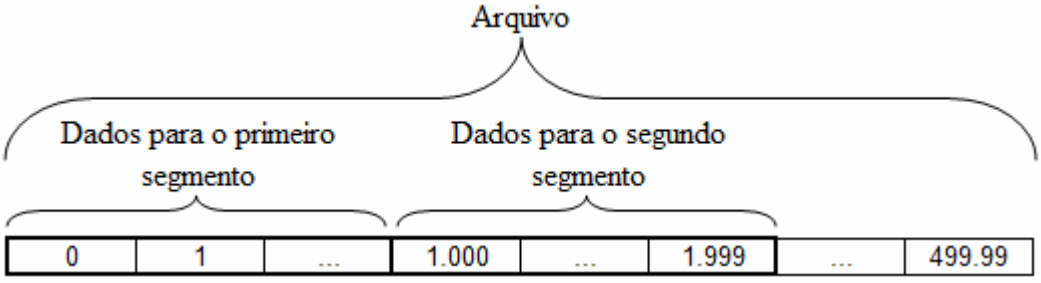


Figura 8.4 – *Dividindo os dados do arquivo em segmentos TCP (Modificado de [6]).*

O primeiro segmento recebe o número de seqüência 0; o segundo, o número de seqüência 1.000; o terceiro, o número de seqüência 2.000, e assim sucessivamente. Verifica-se, então, que o número de seqüência do próximo segmento é sempre igual ao do segmento anterior somado do número de bytes de carga útil deste último.

Os números de reconhecimentos são utilizados por um hospedeiro para informar ao outro qual o próximo número de seqüência que ele espera receber [6]. Por exemplo, no caso da Figura 8.4, após o hospedeiro B receber o segmento com número de seqüência 1.000, ele retornará um quadro ACK ao hospedeiro A no qual o número de reconhecimento

será igual ao número 2.000, que é justamente o próximo número de seqüência que B está aguardando receber de A. Se o próximo segmento que B receber contiver o número de seqüência 5.000, por exemplo, isso implica que houve perda de informações, ou seja, perda de pacotes. Nesse caso, foram perdidos 3 pacotes, os de número de seqüência 2.000, 3.000 e 4.000.

Com esse exemplo, verifica-se que o número de pacotes perdidos pode ser determinado subtraindo-se o número de reconhecimento do número de seqüência do último pacote recebido e dividir esse resultado pelo número de bytes de dados do segmento:

$$\frac{(5.000 - 2.000)}{1.000} = 3 \text{ pacotes}$$

A Figura 8.5 (a) mostra com mais detalhes dois pacotes com dados do arquivo de vídeo utilizado nos experimentos desta dissertação sendo transferidos do IP 192.168.0.115 para o IP 192.168.0.114. O pacote de reconhecimento ACK para eles também pode ser observado. É importante notar que o reconhecimento parte do IP 192.168.0.114 para o IP 192.168.0.115 seguindo, portanto, o sentido oposto ao do fluxo de dados.

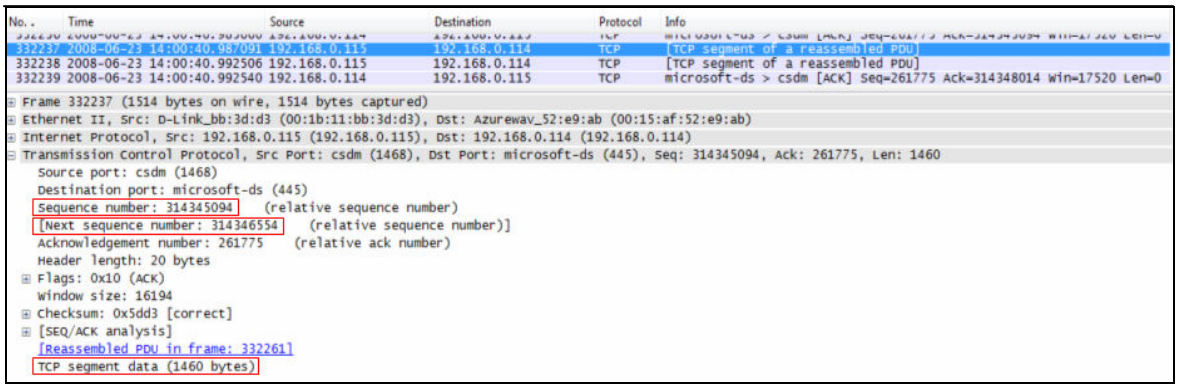


Figura 8.5 a – *Pacote contendo carga útil sendo transferido do desktop para o laptop: número de seqüência 314345094.*

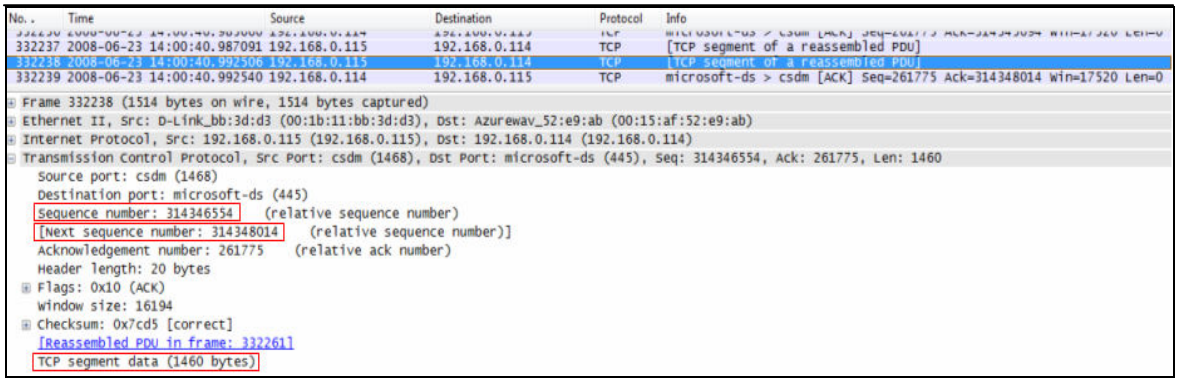


Figura 8.5 b – *Pacote contendo carga útil sendo transferido do desktop para o laptop: número de seqüência 314346554.*

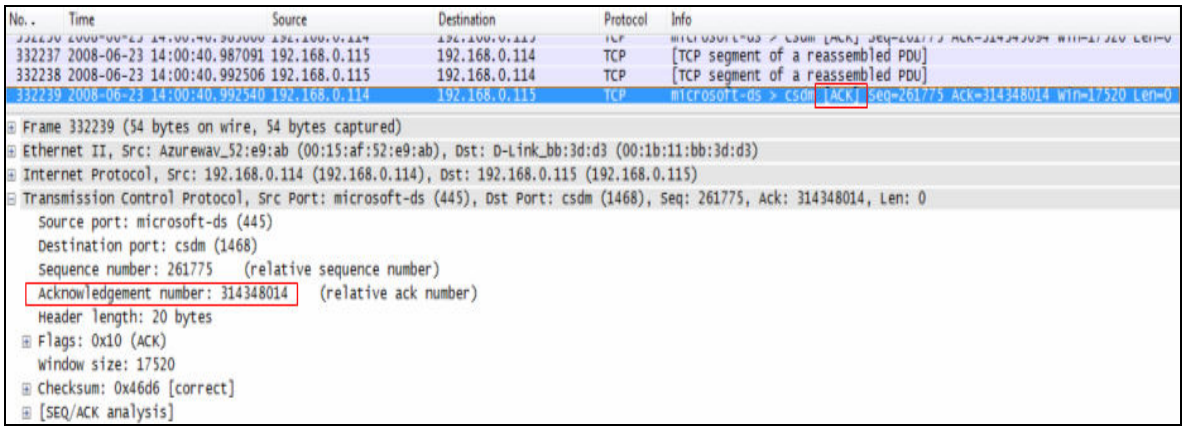


Figura 8.5 c – *Confirmação de recebimento enviado pelo laptop ao desktop.*

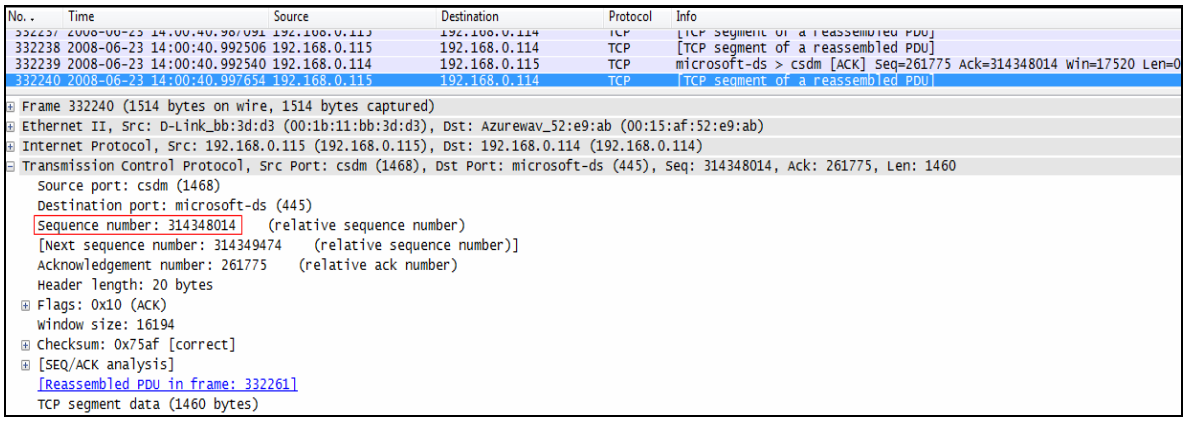


Figura 8.5 d – *Continuação natural do fluxo de dados.*

Na Figura 8.5 (a) pode ser observado que o número de seqüência (campo *Sequence number*) do segmento TCP é 314345094. O próximo número de seqüência (campo *Next sequence number*) é o número 314346554, pois o segmento é formado por 1.460 bytes, como mostrado pelo *TCP segment data*.

A Figura 8.5 (b) mostra que o próximo pacote transmitido tem 314346554 como número de seqüência do segmento TCP, que confere com o número informado no campo *Next sequence number* do pacote imediatamente anterior. Como esse segmento TCP é formado por 1.460 bytes, o próximo número de seqüência será o 314348014.

A Figura 8.5 (c) mostra um pacote de reconhecimento ACK. Conforme pode ser visualizado no campo *Acknowledgement number* (Número de reconhecimento), o receptor do fluxo de dados está informando ao transmissor que o próximo número de seqüência que ele (o receptor) está esperando é o 314348014. É importante notar que esse é justamente o número que veio no campo *Next sequence number* do último segmento recebido. Dessa forma, o receptor está confirmando o recebimento do último pacote e, por tabela, do penúltimo.

De acordo com a expectativa do receptor, o pacote de dados que se segue ao ACK tem 314348014 como número de seqüência do segmento TCP. Esse fato pode ser constatado pela Figura 8.5 (d).

No caso exemplificado pelas Figuras 8.5 (a), (b), (c) e (d) não houve perda de pacotes. A seguir, será exemplificado um outro trecho desse mesmo arquivo de *log* no qual houve perda de pacotes.

A Figura 8.6 mostra um trecho do arquivo de captura onde houve perda de pacotes. Analisando a primeira linha da Figura 8.6, podemos verificar que o próximo número de seqüência que o receptor está aguardando é o 311934602. Esse valor é o que aparece no campo ACK marcado com um retângulo vermelho. As últimas linhas da Figura 8.6 (pacotes sinalizados como TCP *Fast Retransmission* e TCP *Retransmission*) sinalizam as retransmissões dos pacotes perdidos.

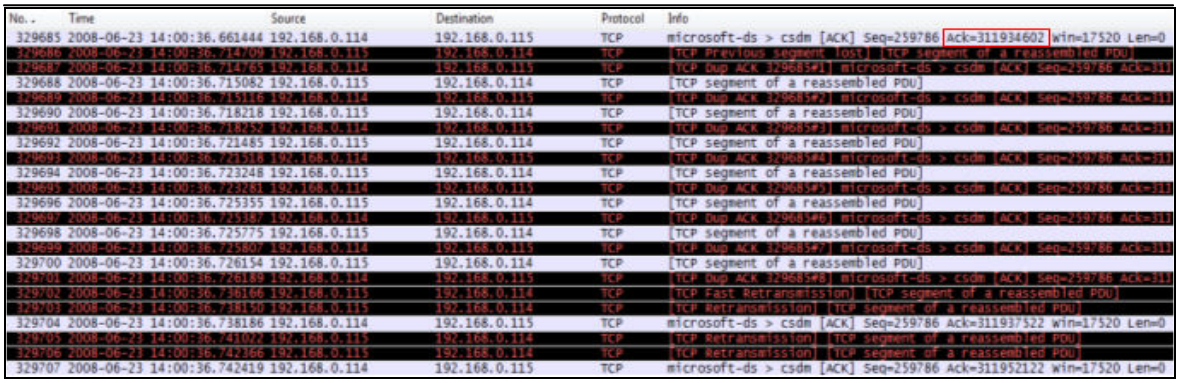


Figura 8.6 – *Próximo número de seqüência esperado.*

Apesar de o receptor está esperando um pacote com número de seqüência 311934602, o próximo pacote recebido, como mostra a Figura 8.7, possui o número de seqüência 311940442, diferente, portanto, do que era esperado. Com isso, uma mensagem de alerta TCP *Previous segment lost* é gerada para sinalizar que houve perda de pacotes.

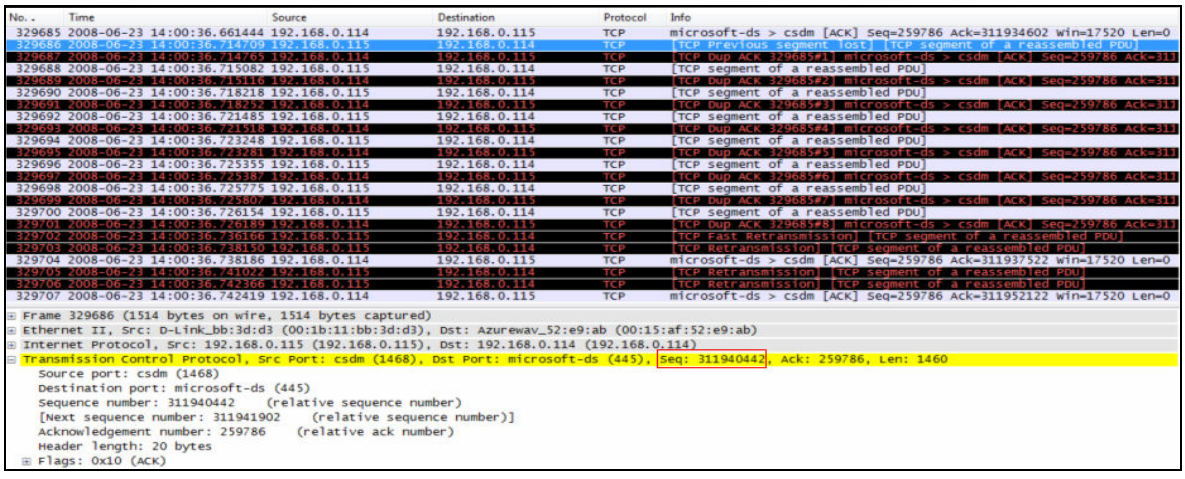


Figura 8.7 – *Conflito entre número de seqüência esperado e recebido.*

Subtraindo 311934602 de 311940442 e dividindo o resultado por 1.460, determina-se que 4 pacotes foram perdidos nesse trecho.

Observando as Figuras 8.6 e 8.7, pode-se verificar que logo em seguida ao pacote com o alerta TCP *Previous segment lost*, surge uma seqüência de pacotes com a informação TCP Dup ACK. Esses pacotes constituem uma reação imediata do receptor aos pacotes que foram perdidos. De acordo com [43], um número elevado de ACKs duplicados pode ser um indicador de latência na rede.

Os quatro pacotes perdidos nesse trecho da captura foram os de números de seqüência 311934602, 311936062, 311937522 e 311938982. As Figuras 8.8 (a), (b), (c) e (d) mostram suas respectivas retransmissões.

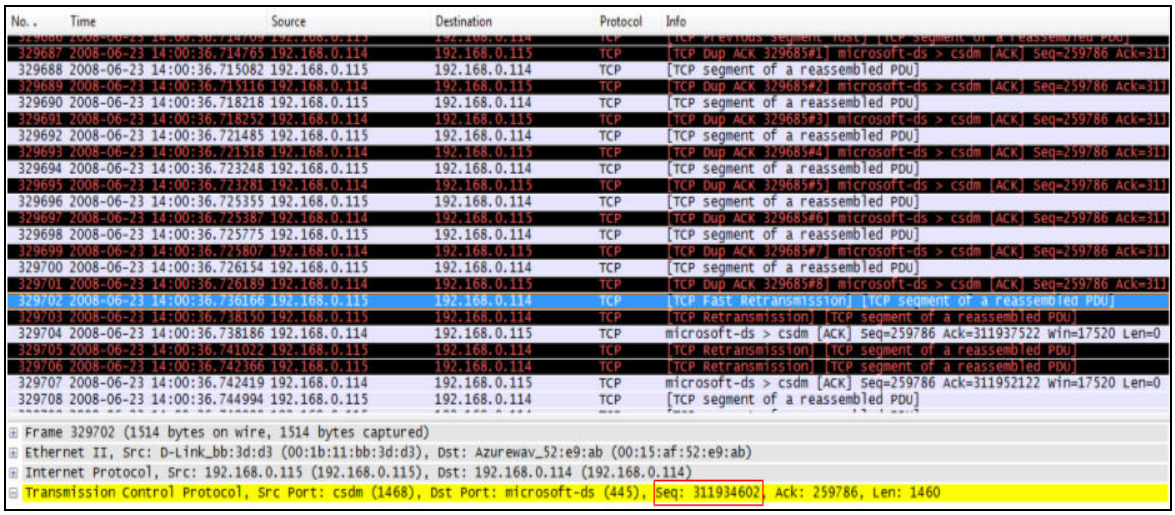


Figura 8.8 a – *Retransmissão do pacote com número de seqüência 311934602.*

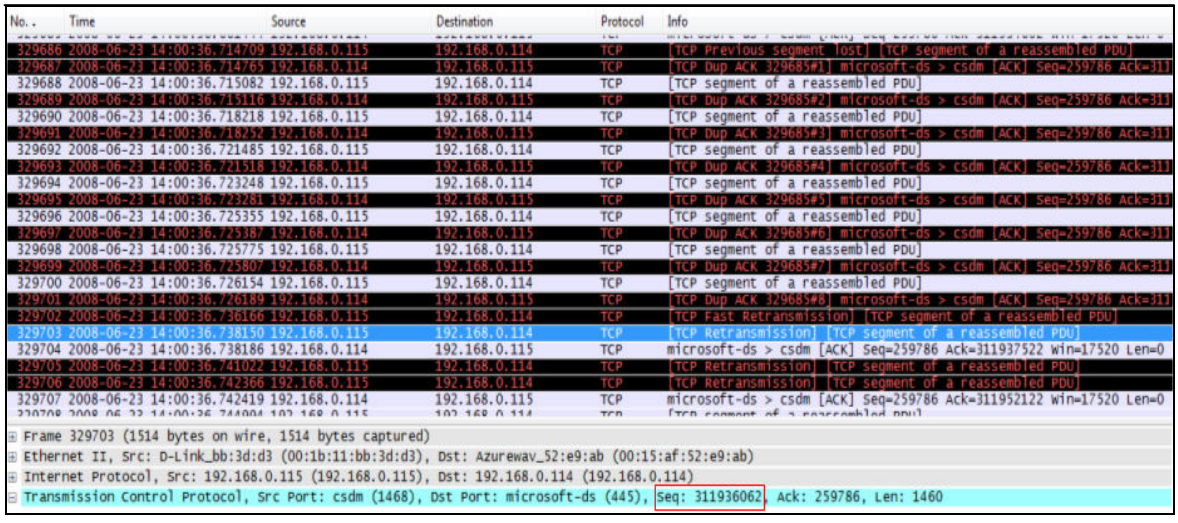


Figura 8.8 b – *Retransmissão do pacote com número de seqüência 311936062.*

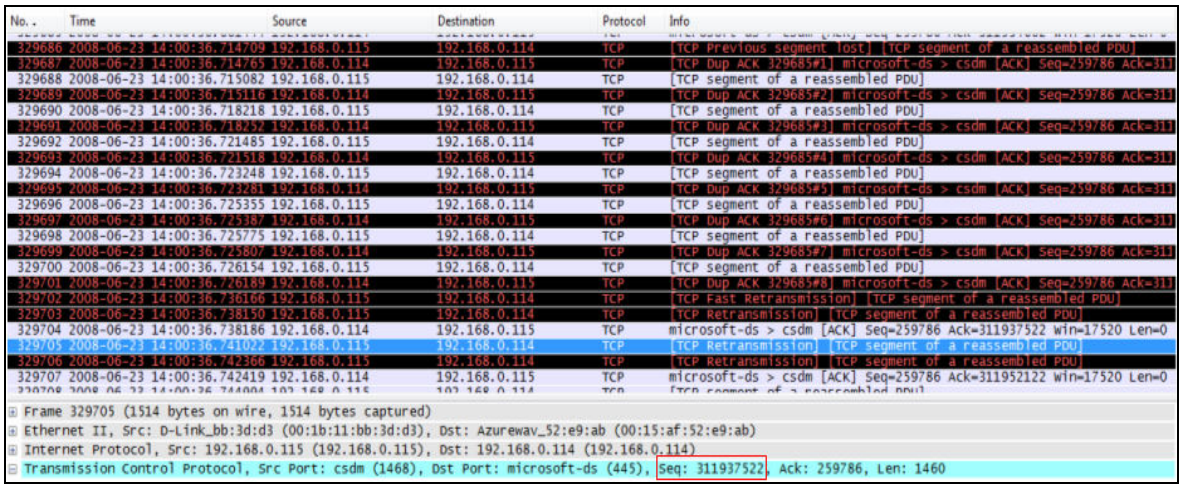


Figura 8.8 c – *Retransmissão do pacote com número de seqüência 311937522.*

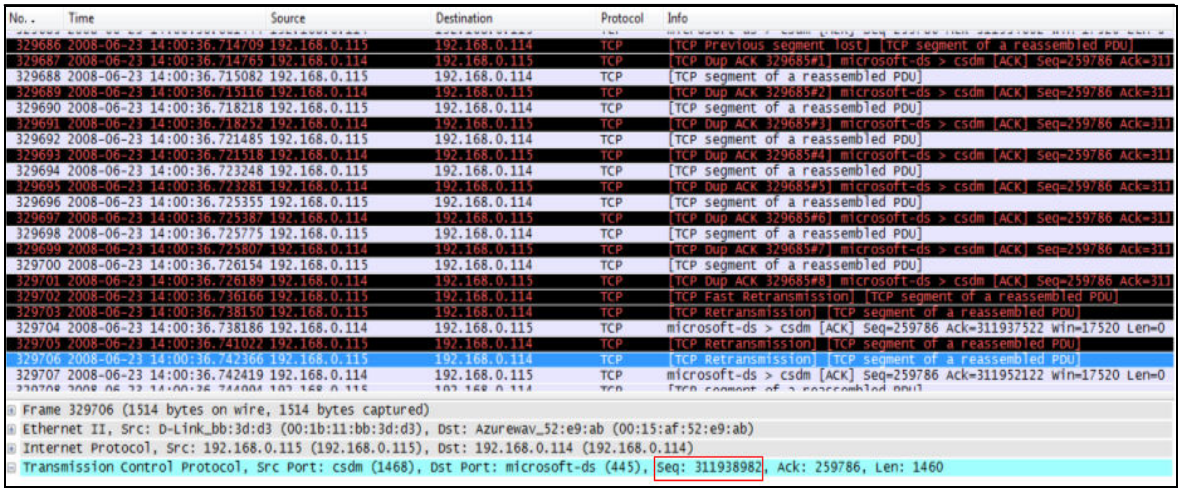


Figura 8.8 d – *Retransmissão do pacote com número de seqüência 311938982.*

Assim, baseado na explicação realizada nos parágrafos anteriores, foi possível determinar a quantidade total de pacotes perdidos em cada transferência por meio da análise dos números de seqüência de todos os pacotes que geraram a mensagem de alerta TCP *Previous segment lost*.

A partir de agora, serão expostos os resultados dos experimentos. Para todos os gráficos que se seguem, a curva em verde representa o cenário sem interferência *Bluetooth* e a vermelha, o cenário com interferência *Bluetooth*.

8.1 Tempo de transmissão

O primeiro resultado ilustrado aqui diz respeito ao parâmetro tempo de transmissão do arquivo na rede IEEE 802.11g. As Figuras 8.9, 8.10 e 8.11 mostram as curvas de tempo para cada uma das três distâncias trabalhadas e para cada um dos dois cenários de testes.

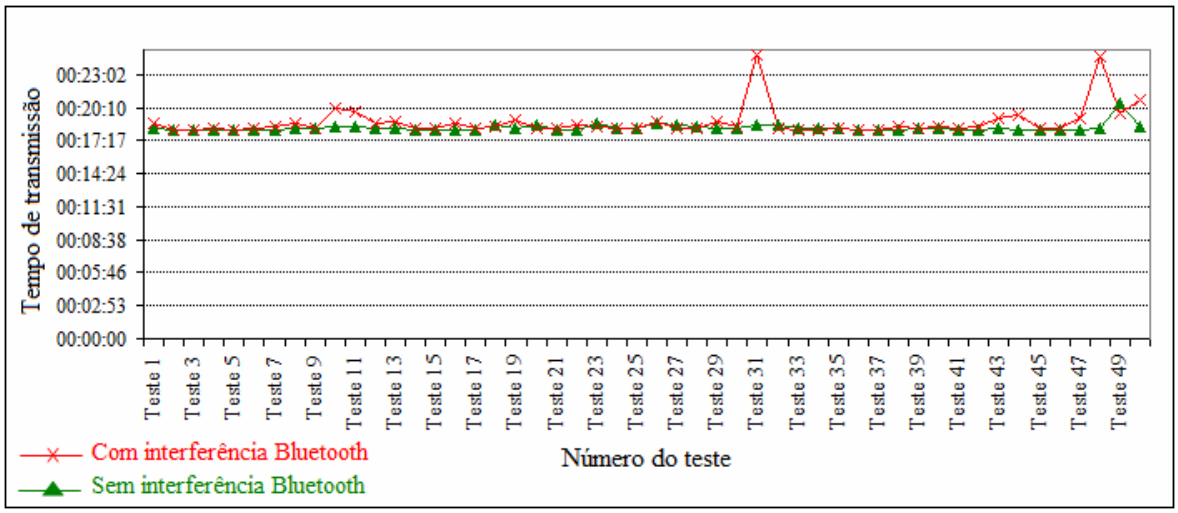


Figura 8.9 – *Tempo de transmissão: distância de 1,60 m.*

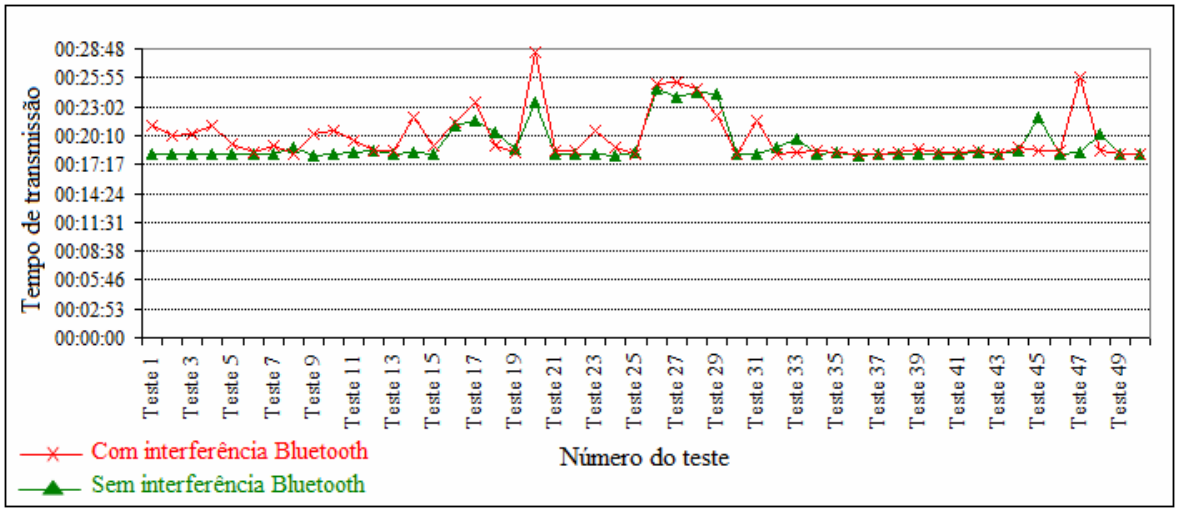


Figura 8.10 – *Tempo de transmissão: distância de 2,60 m.*

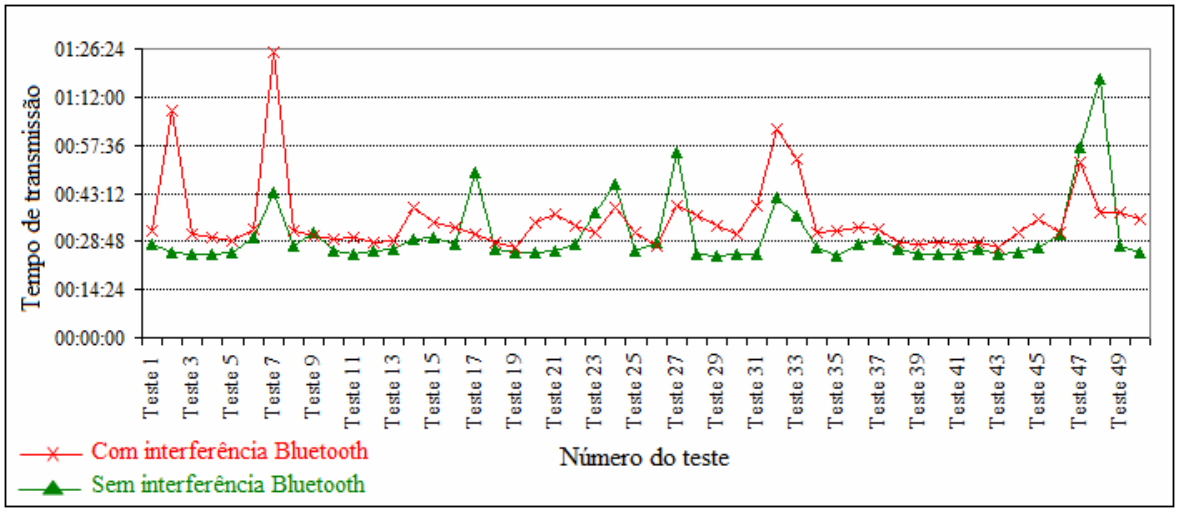


Figura 8.11 – *Tempo de transmissão: distância de 4,60 m.*

Analisando a Figura 8.9, observa-se que as curvas de tempo de transmissão sem e com interferência *Bluetooth* praticamente se sobrepõem. Em alguns pontos, a curva em

vermelho claramente supera a curva em verde. Portanto, nesses pontos a transmissão do arquivo levou mais tempo para ser concluída. O maior pico registrado para a distância de 1,60m foi no momento em que havia interferência *Bluetooth*, com 24 minutos e 50 segundos; o menor tempo registrado foi quando não havia interferência *Bluetooth*, com 18 minutos e 13 segundos.

A Figura 8.10 já mostra uma diferença um pouco mais clara entre as duas curvas. É interessante notar que, em geral, as duas curvas seguem aproximadamente a mesma tendência, ou seja, quando uma tende a piorar (ou melhorar), a outra acompanha esse comportamento. Esse fato pode ser observado, por exemplo, nos testes 25 ao 30. Já para a a distância de 2,60 m, o maior pico foi registrado quando havia interferência Bluetooth, com 28 minutos e 32 segundos; o menor tempo registrado foi sem interferência *Bluetooth*, com 18 minutos e 13 segundos.

Para a distância de 4,60 m, a diferença entre os tempos de transmissão com e sem interferência *Bluetooth* é mais evidente. Conforme pode ser observado na Figura 8.11, para a grande maioria dos testes realizados, o desempenho com interferência *Bluetooth* foi pior do que sem interferência *Bluetooth*. Para alguns trechos, as duas curvas apresentaram aproximadamente o mesmo comportamento. Um bom exemplo disso é o trecho que vai do teste 25 ao teste 47.

Analisando a Figura 8.11, pode-se verificar que em seis testes o tempo de transmissão sem interferência *Bluetooth* foi maior do que com interferência. Isso pode ser explicado por três possíveis razões: (1) pelo motivo de, no momento desses testes sem a interferência *Bluetooth*, uma quantidade maior de redes *Wi-Fi* vizinhas estavam ativas em freqüências próximas à utilizada; (2) dispositivos *Bluetooth* localizados na vizinhança poderiam estar ativos e (3) a probabilidade de colisão entre os pacotes dos dois tipos de redes sem fios não é de 100%. Se assim fosse, o desempenho *Wi-Fi* sob interferência *Bluetooth* seria obrigatoriamente pior do que sem a interferência.

Seguindo a mesma característica das distâncias de 1,60 m e 2,60 m, o maior pico para a distância de 4,60 m foi registrado quando havia interferência *Bluetooth*, com 1 hora 25 minutos e 15 segundos; o menor tempo registrado foi sem interferência *Bluetooth*, com 24 minutos e 37 segundos.

Comparando agora entre si os resultados obtidos para as distâncias de 1,60 m, 2,60 m e 4,60 m, é possível perceber que o sistema 802.11g fica mais instável com relação ao tempo de transmissão à medida que a distância aumenta. Essa instabilidade acontece tanto

para a curva de tempo sem interferência *Bluetooth* quanto para a curva com interferência *Bluetooth*.

Além dessa instabilidade, verificou-se que com o aumento da distância, o tempo de transmissão foi gradativamente aumentando, indiferentemente se havia interferência *Bluetooth* ou não. As Figuras 8.12 e 8.13 ilustram essa situação.

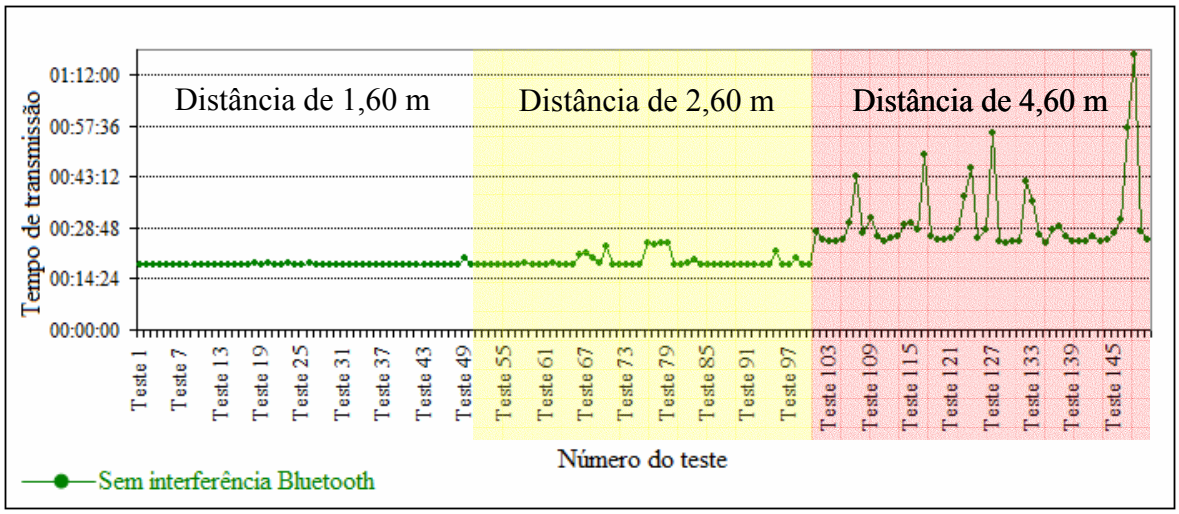


Figura 8.12 – *Evolução da curva de tempo de transmissão: sem interferência Bluetooth.*

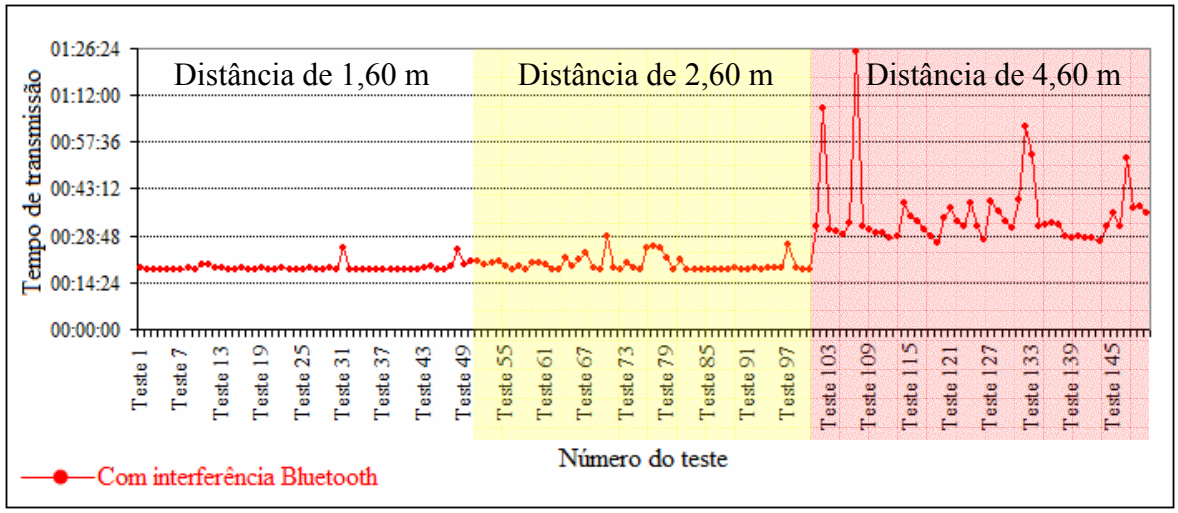


Figura 8.13 – *Evolução da curva de tempo de transmissão: com interferência Bluetooth.*

8.2 Taxa média de transmissão de pacotes (em pacotes/s)

O próximo parâmetro a ser analisado é a taxa média de transmissão de pacotes. As Figuras 8.14, 8.15 e 8.16 ilustram os resultados obtidos para os experimentos realizados com as diferentes distâncias.

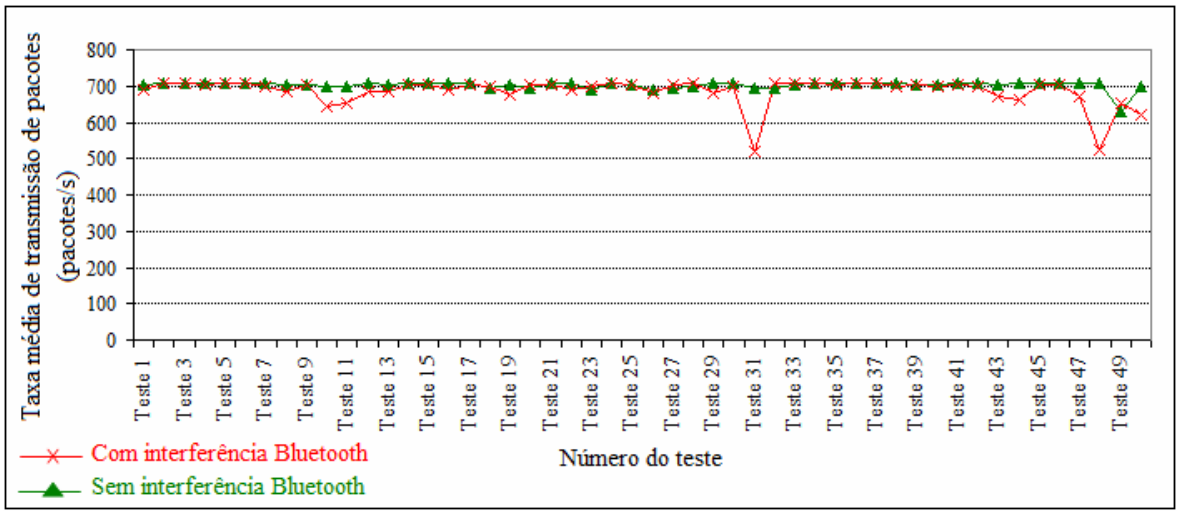


Figura 8.14 – *Taxa media de transmissão de pacotes: distância de 1,60 m.*

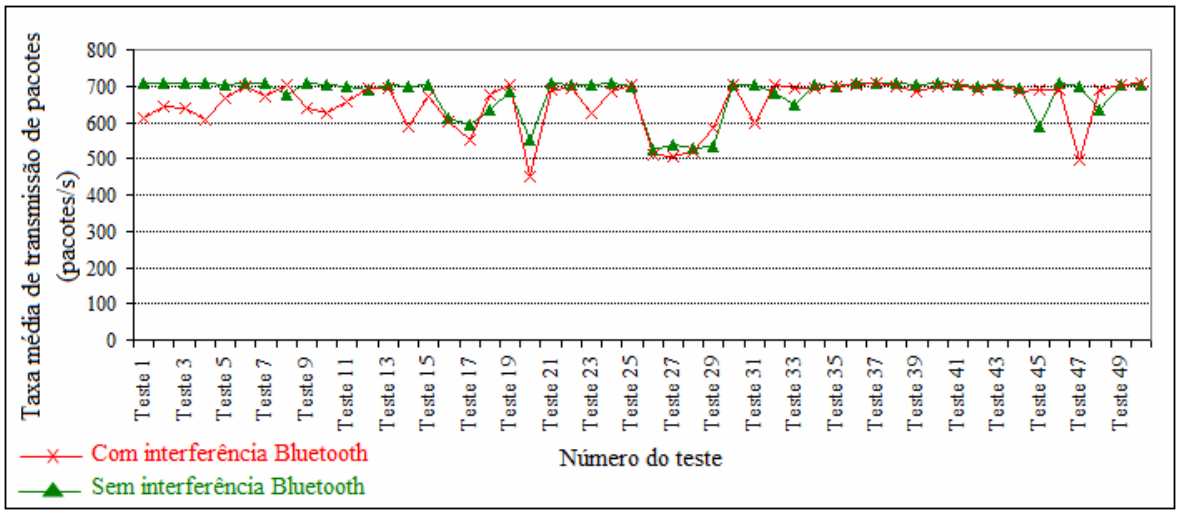


Figura 8.15 – *Taxa media de transmissão de pacotes: distância de 2,60 m.*

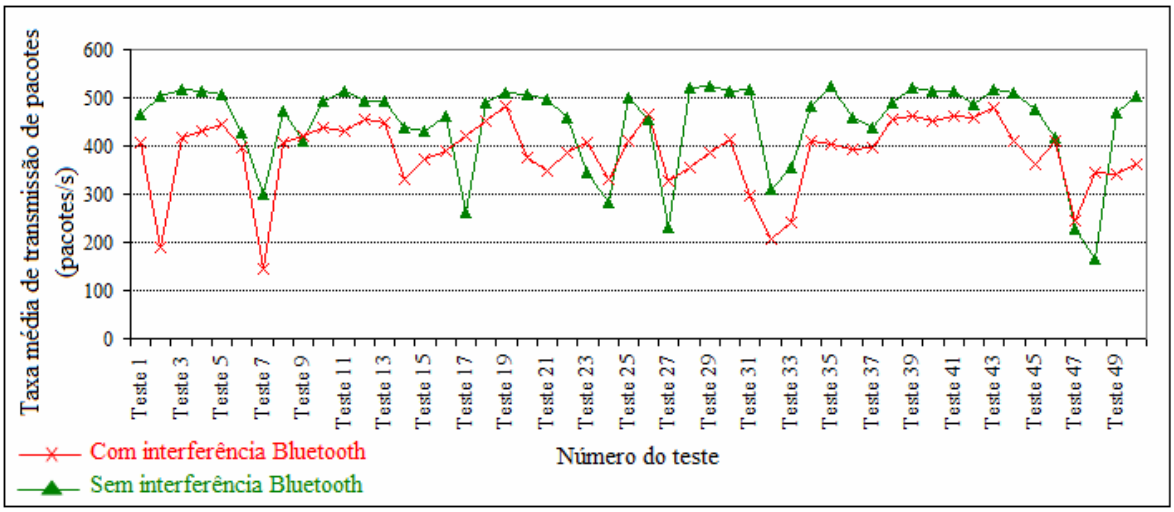


Figura 8.16 – *Taxa media de transmissão de pacotes: distância de 4,60 m.*

Uma primeira característica interessante de ser observada é que o comportamento das curvas ilustradas nas Figuras 8.14, 8.15 e 8.16 é semelhante ao das curvas das Figuras

8.9, 8.10 e 8.11, porém com tendência oposta. Por exemplo, enquanto a tendência do trecho que vai do teste 25 ao 29 da Figura 8.10 é crescer, se manter em um patamar e depois decrescer, a tendência para esse mesmo trecho de testes na Figura 8.15 é decrescer, se manter em um patamar e depois crescer. Isso acontece pelo simples fato de que quanto mais tempo uma transmissão levar para ser concluída, menor será a taxa de transmissão de pacotes e vice versa.

Analisando a Figura 8.14, observa-se que as taxas de transmissão sem e com interferência *Bluetooth* se mantêm muito próximas para a maioria dos testes realizados. Porém, para alguns experimentos, é possível verificar uma discrepância maior entre as taxas médias de transmissão de pacotes sem e com interferência *Bluetooth*. A menor taxa de transmissão registrada foi no momento em que havia interferência *Bluetooth*, com 520,038 pacotes/s; a maior foi sem interferência *Bluetooth*, com 709,226 pacotes/s.

A diferença entre as taxas médias de transmissão de pacotes para os dois cenários de testes fica mais evidente para a distância de 2,60 m, como mostrado na Figura 8.15. Para esse caso, a menor taxa de transmissão registrada foi com interferência *Bluetooth*, computando-se 452,640 pacotes/s; a maior foi sem interferência *Bluetooth*, com 708,869 pacotes/s.

Conforme pode ser visualizado na Figura 8.16, para a maioria dos testes realizados para a distância de 4,60 m, a taxa média de transmissão de pacotes sem interferência *Bluetooth* foi superior a com interferência. A menor taxa de transmissão registrada para a distância de 4,60 m foi na existência de fonte interferente *Bluetooth*, com 145,679 pacotes/s; a maior foi sem interferência *Bluetooth*, com 524,778 pacotes/s. Da mesma maneira que para o tempo de transmissão (Figura 8.11), analisando-se a Figura 8.16, verifica-se que em seis testes a taxa média de transmissão de pacotes sem interferência *Bluetooth* foi menor. Isso pode ser explicado pelas mesmas possibilidades mencionadas anteriormente para o tempo de transmissão.

Comparando-se as Figuras 8.14, 8.15 e 8.16, verifica-se que as curvas de taxa média de transmissão de pacotes sem e com interferência *Bluetooth* apresentam maior oscilação à medida que a distância aumenta.

Analisando-se separadamente as situações em que não havia interferência *Bluetooth* e que havia, verifica-se que a taxa média de transmissão de pacotes vai decrescendo à medida que a distância entre o transmissor e o receptor vai aumentando. As Figuras 8.17 e 8.18 ilustram essas situações.

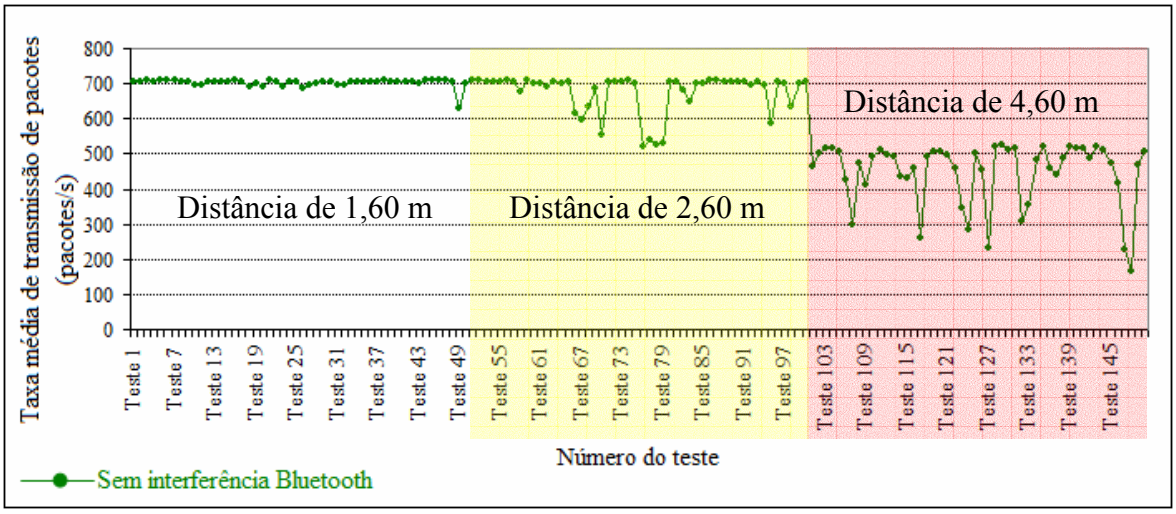


Figura 8.17 – *Evolução da taxa media de transmissão de pacotes: sem interferência.*

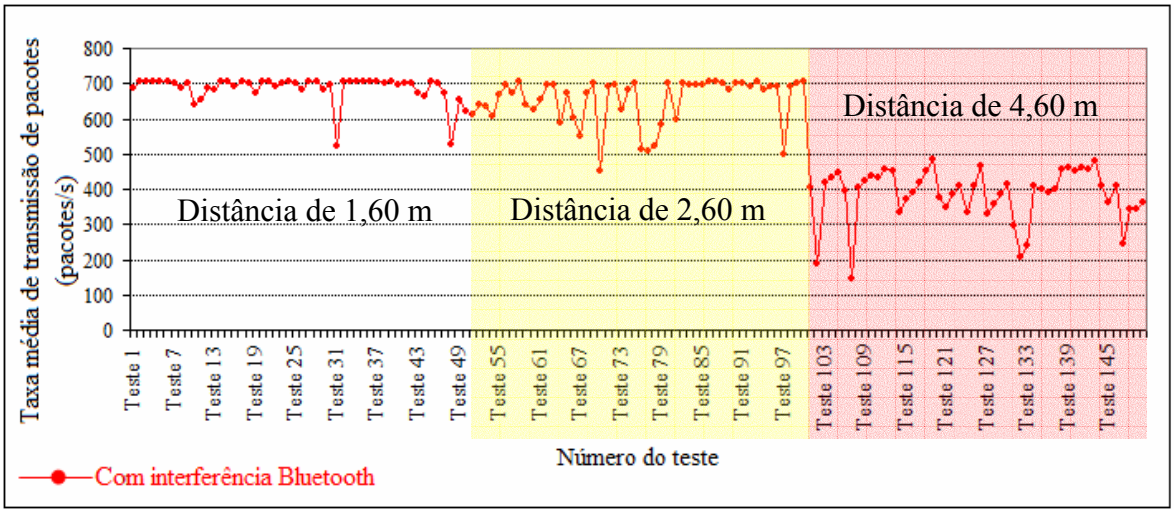


Figura 8.18 – *Evolução da taxa media de transmissão de pacotes: com interferência.*

8.3 Taxa média de transmissão de dados (em Mbps)

As Figuras 8.19, 8.20 e 8.21 ilustram os resultados obtidos para esse parâmetro para as situações em que havia interferência *Bluetooth* e em que não havia interferência *Bluetooth,* considerando cada uma das três distâncias analisadas.

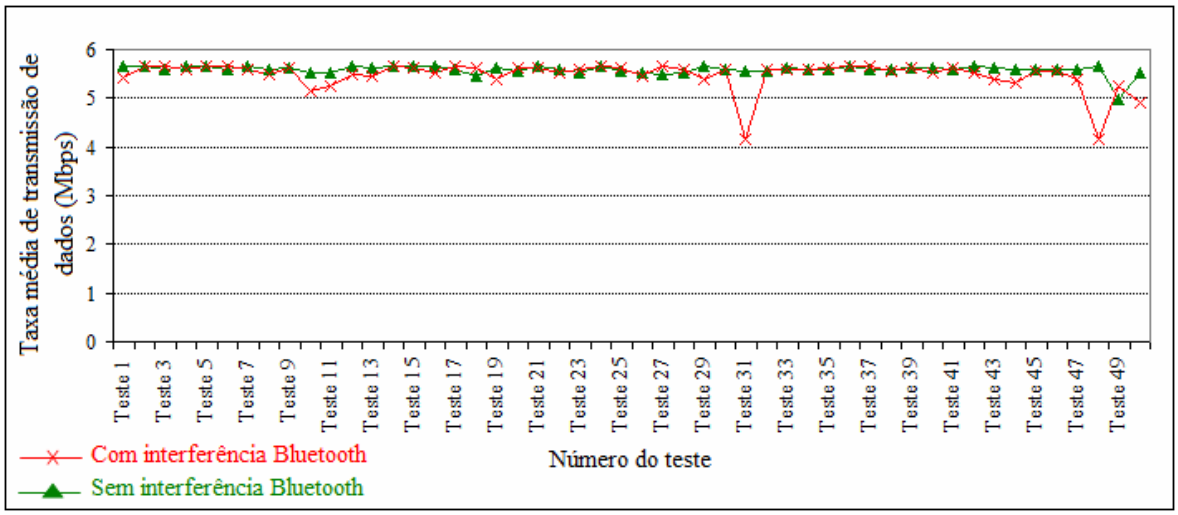


Figura 8.19 – *Taxa media de transmissão de dados: distância de 1,60 m.*

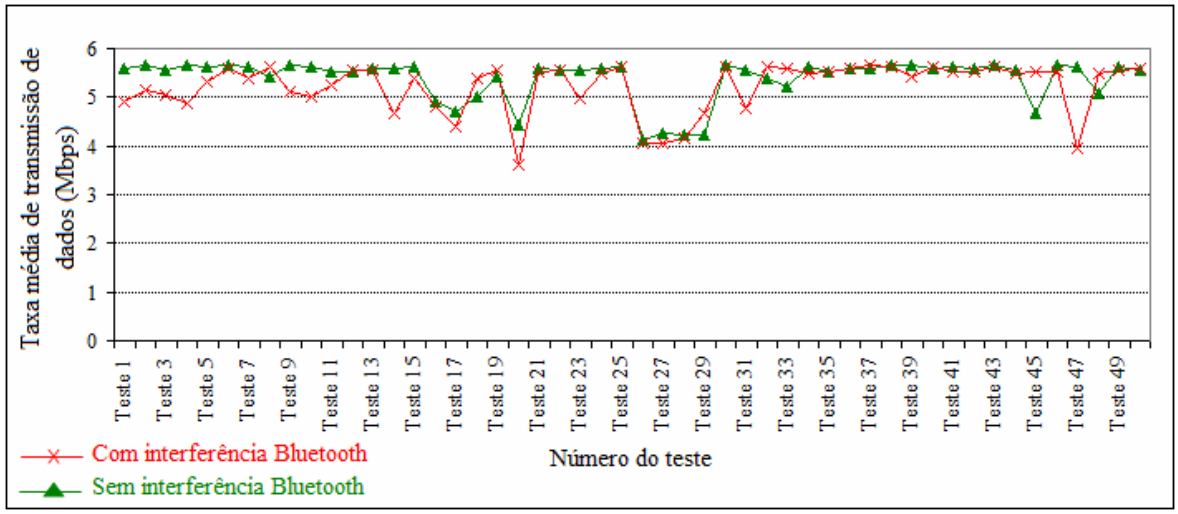


Figura 8.20 – *Taxa media de transmissão de dados: distância de 2,60 m.*

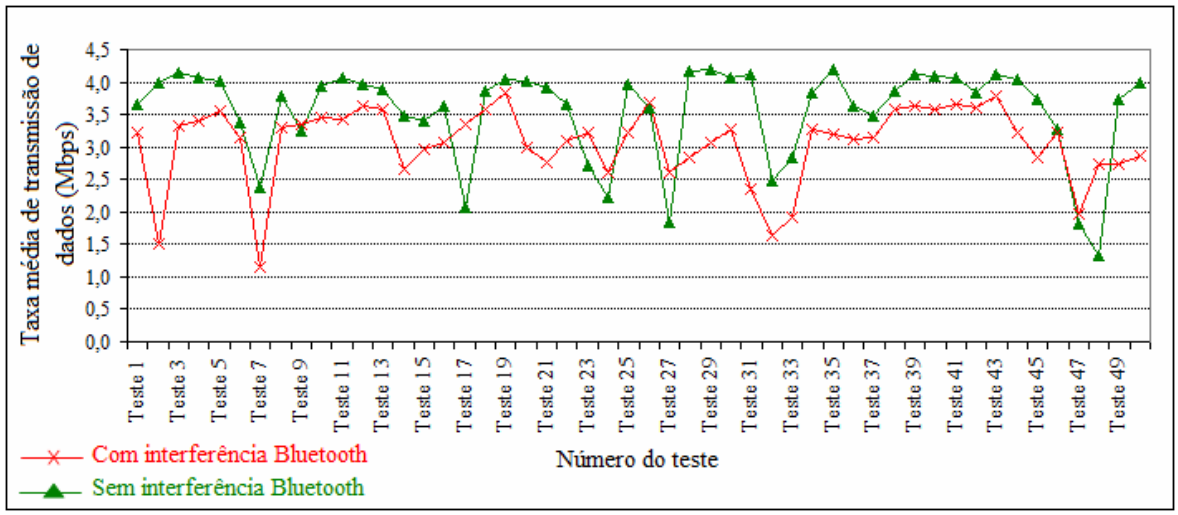


Figura 8.21 – *Taxa media de transmissão de dados: distância de 4,60 m.*

Comparando-se as Figuras 8.19 com 8.14, 8.20 com 8.15 e 8.21 com 8.16, constata-se que o comportamento das curvas para a taxa média de transmissão de dados foi praticamente igual ao das curvas de taxa média de transmissão de pacotes.

Outro ponto importante é que, da mesma maneira que as curvas de taxa média de transmissão de pacotes, as curvas de taxa média de transmissão de dados têm um comportamento semelhante, porém com tendência oposta, ao das curvas de tempo de transmissão.

Da mesma forma que para os outros parâmetros analisados até aqui, vê-se que com o aumento da distância, a diferença de desempenho entre os dois cenários de testes torna-se mais evidente. E novamente o cenário sem a presença do *Bluetooth* apresentou melhores resultados, principalmente para a distância de 4,60 m. Como pode ser observado na Figura 8.21, a curva que representa o cenário sem interferência *Bluetooth* ficou acima da curva que representa o cenário com interferência *Bluetooth* para a maioria dos testes. O comportamento observado nas Figuras 8.11 e 8.16 se repete na Figura 8.21, com seis testes sem *Bluetooth* apresentando taxa média de transmissão de dados inferior. Os prováveis motivos desse comportamento são os mesmo explicitados para as Figuras 8.11 e 8.16.

Verificou-se que a taxa média de transmissão de dados da rede 802.11g ficou mais instável à medida que se afastava o receptor do transmissor. Além dessa instabilidade, pôde ser observada uma queda na taxa para os dois cenários de testes à medida que se aumentava a distância, conforme ilustrado nas Figuras 8.22 e 8.23.

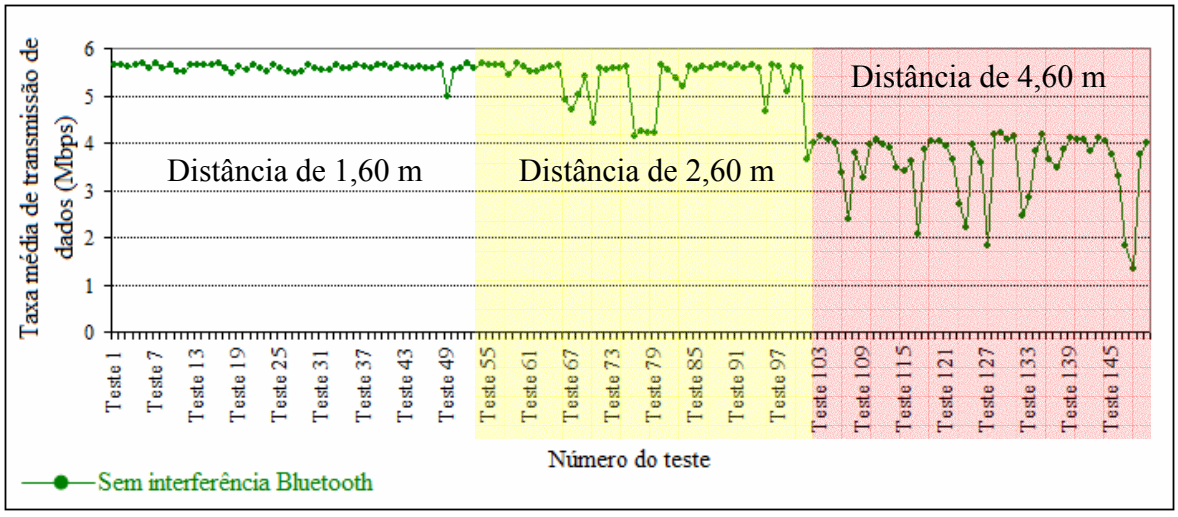


Figura 8.22 – *Evolução da taxa media de transmissão de dados: sem interferência Bluetooth.*

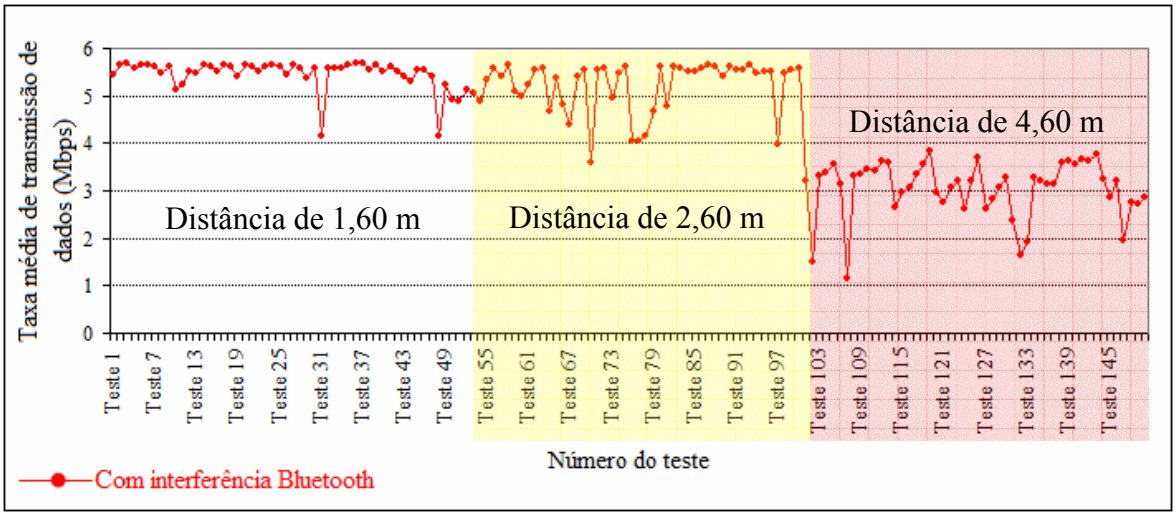


Figura 8.23 – *Evolução da taxa media de transmissão de dados: com interferência Bluetooth.*

Um fato importante de ser pontuado aqui é que, apesar da rede 802.11g estar configurada para trabalhar a 54 Mbps, a maior taxa de transmissão de dados atingida pelo sistema, considerando todos testes realizados, não chegou a 6 Mbps. A razão disso é que essa taxa de transmissão depende do tamanho da janela TCP negociada entre o transmissor e o receptor.

Para a distância de 1,60 m, a menor taxa de transmissão atingida foi quando havia fonte interferente *Bluetooth*, com 4,153 Mbps; a maior foi quando não havia fonte interferente *Bluetooth*, com 5,675Mbps. Para a distância de 2,60 m, a menor taxa de transmissão obtida foi quando havia interferência, com 3,607 Mbps; a maior foi quando não havia interferência, com 5,671 Mbps. E para a distância de 4,60 m, a menor taxa de transmissão foi também para o cenário no qual existia fonte interferente *Bluetooth*, com 1,144 Mbps; a maior foi no cenário sem interferência, com 4,201 Mbps.

8.4 Quantidade de pacotes perdidos

O parâmetro de avaliação de desempenho mais importante é a quantidade de pacotes perdidos em cada transmissão realizada. As Figuras 8.24, 8.25 e 8.26 ilustram os resultados obtidos para esse parâmetro.

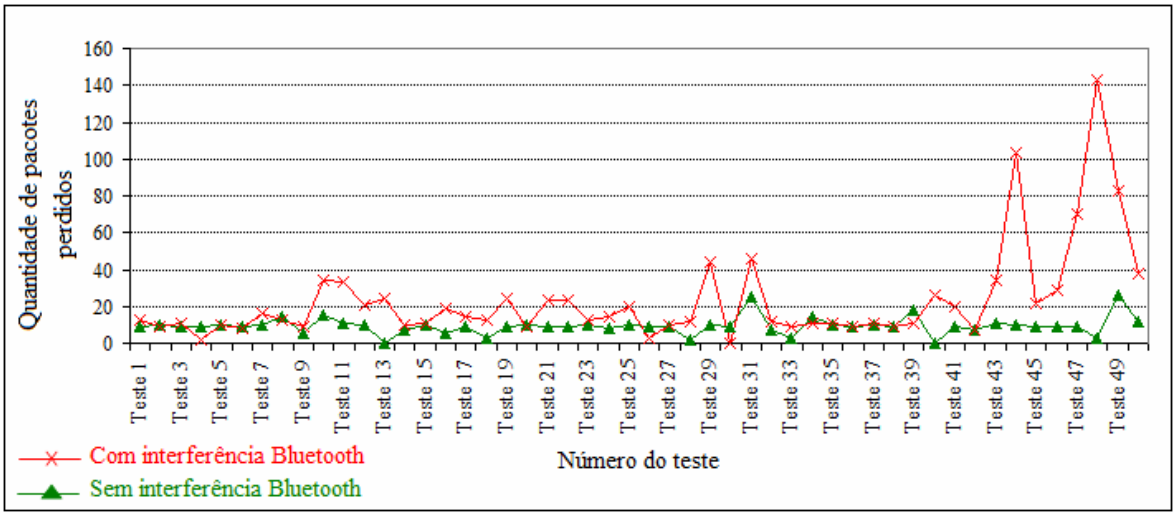


Figura 8.24 – *Quantidade de pacotes perdidos: distância de 1,60 m.*

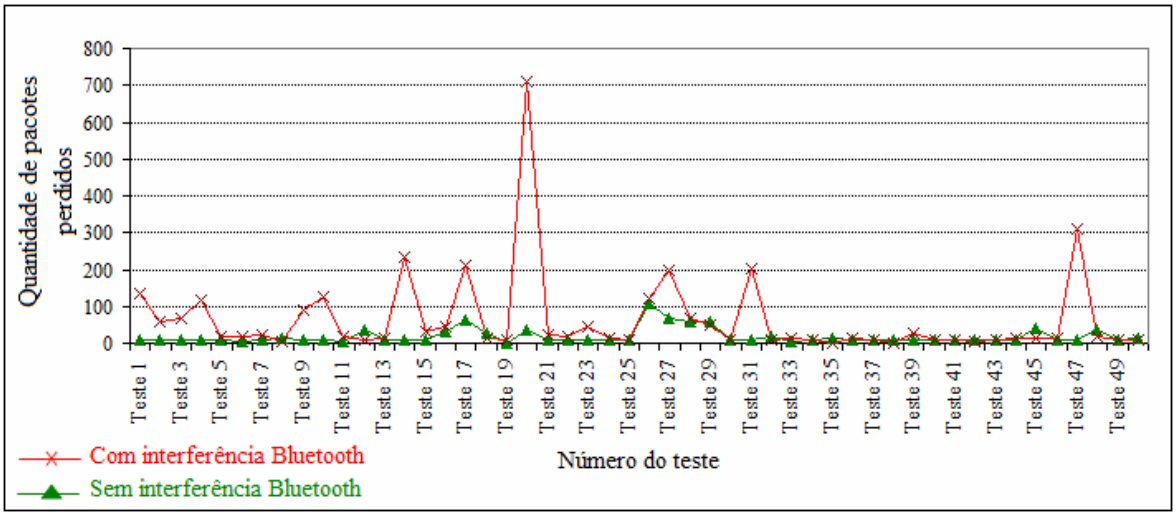


Figura 8.25 – *Quantidade de pacotes perdidos: distância de 2,60 m.*

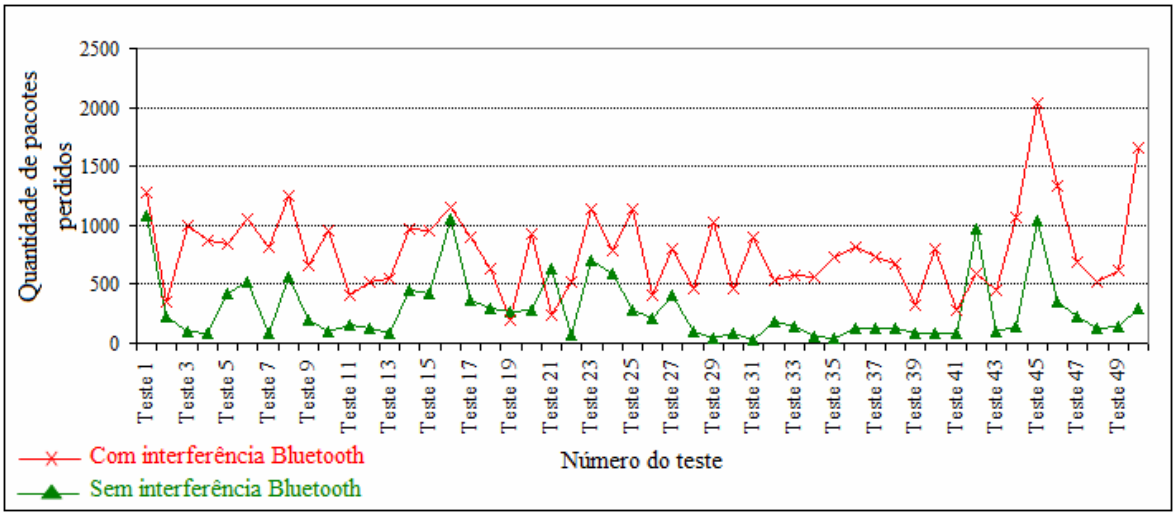


Figura 8.26 – *Quantidade de pacotes perdidos: distância de 4,60 m.*

Analisando-se a Figura 8.24, verifica-se que a quantidade de pacotes perdidos no cenário com interferência *Bluetooth* foi superior ao do cenário sem interferência em cerca

de metade dos testes realizados. Essa diferença entre a quantidade de pacotes perdidos dos dois cenários se acentuou mais do teste 43 ao teste 50.

Para os resultados da Figura 8.25, observa-se que, de uma maneira geral, a quantidade de pacotes perdidos foi maior quando existia fonte interferente *Bluetooth*. Nesse ponto, é importante lembrar que a probabilidade de colisão entre um pacote IEEE 802.11g e um pacote *Bluetooth* não é de 100%. Se assim fosse, obrigatoriamente o número de pacotes perdidos seria sempre maior quando a transmissão *Bluetooth* estava ocorrendo simultaneamente à transmissão 802.11g. Por isso, para vários testes da Figura 8.25, a quantidade de pacotes perdidos para os dois cenários de testes é bastante próximo ou levemente diferente. Inclusive, em alguns momentos, o cenário com interferência *Bluetooth* pode apresentar um desempenho até um pouco melhor do que o cenário sem interferência.

A diferença na quantidade de pacotes perdidos entre os dois cenários de testes ficou mais gritante para última distância de 4,60 m, como pode ser constatado na Figura 8.26. Em 94% dos testes realizados para essa distância, a quantidade de pacotes perdidos quando havia interferência *Bluetooth* foi superior.

Considerando a distância de 1,60 m, o maior número de pacotes perdidos ocorreu quando existia interferência *Bluetooth*, com 143 pacotes perdidos. Para essa distância, os dois cenários chegaram a apresentar uma quantidade mínima de pacotes perdidos igual a zero.

Para a distância de 2,60 m, a maior quantidade de pacotes perdidos foi também no cenário com interferência, com 710 pacotes. A menor quantidade de pacotes perdidos foi nesse mesmo cenário de testes, com zero perda.

Já para a distância de 4,60 m, o maior número de pacotes perdidos foi computado quando havia interferência *Bluetooth*, com 2.039 pacotes; o menor número foi quando não havia interferência, com 35 pacotes.

Por meio da análise individual de cada cenário, as Figuras 8.27 e 8.28 mostram que a quantidade de pacotes perdidos aumentou bruscamente quando se iniciaram os testes na distância de 4,60 m, tanto para o cenário com interferência quanto para o cenário sem interferência.

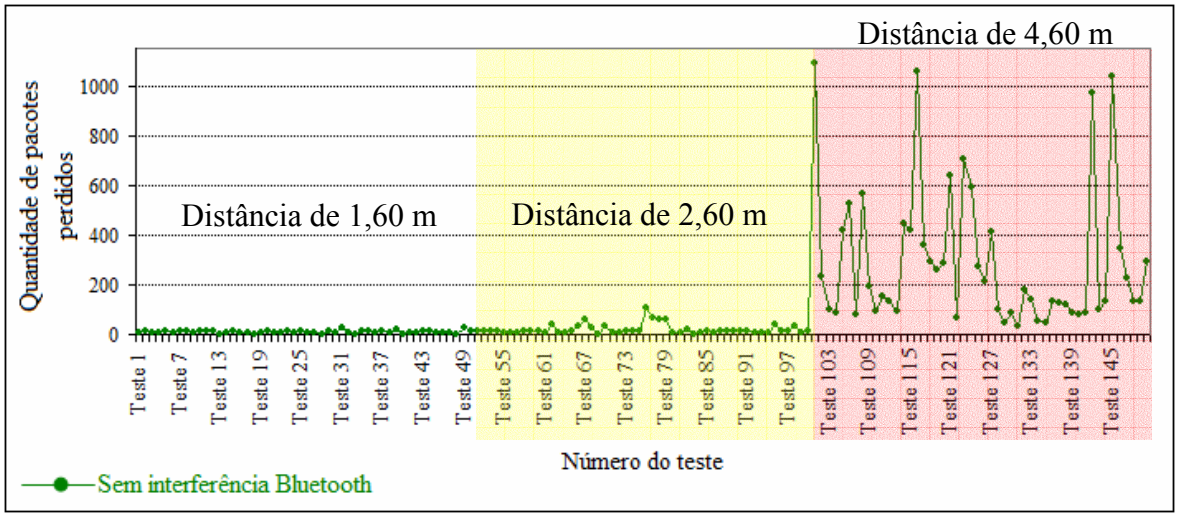


Figura 8.27 – *Evolução da quantidade de pacotes perdidos: sem interferência Bluetooth.*

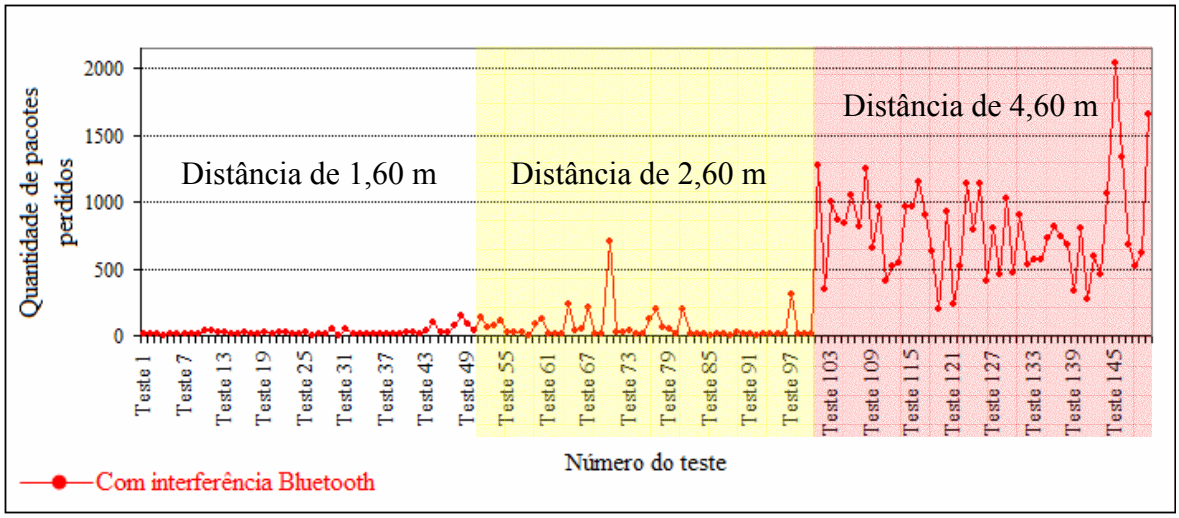


Figura 8.28 – *Evolução da quantidade de pacotes perdidos: com interferência Bluetooth.*

8.5 Quantidade de ACKs duplicados

O receptor dos dados rastreia os números de seqüência dos pacotes que estão chegando do transmissor. Conforme exposto no início deste capítulo, esses números de seqüência devem ser incrementados com o número de bytes de dados contidos em cada pacote.

Relembrando, se um pacote chega com um número de seqüência 10.000 e com 1.460 bytes de dados, o próximo número de seqüência esperado será o 11.460. Se o próximo número de seqüência recebido for o 12.920, o seguinte for o 14.380 e assim por diante, o receptor notificará o transmissor sobre a falta do número de seqüência 11.460. O receptor faz esse alerta enviando vários pacotes ACKs com esse número de seqüência faltante no campo *Acknowledgment Number* [43].

Nesse processo de notificação, o receptor primeiramente envia um pacote ACK normal, e, em seguida, envia os ACKs duplicados. O transmissor deve receber três pacotes ACKs, dos quais um é normal e dois são duplicados, antes de retransmitir o pacote que não foi recebido. Dessa forma, um elevado número de ACKs duplicados pode ser um indicador de problemas de latência na rede. Uma seqüência contínua de ACKs duplicados será enviada até a retransmissão ser recebida [43].

As Figuras 8.29, 8.30 e 8.31 ilustram os resultados obtidos para esse parâmetro de avaliação de desempenho.

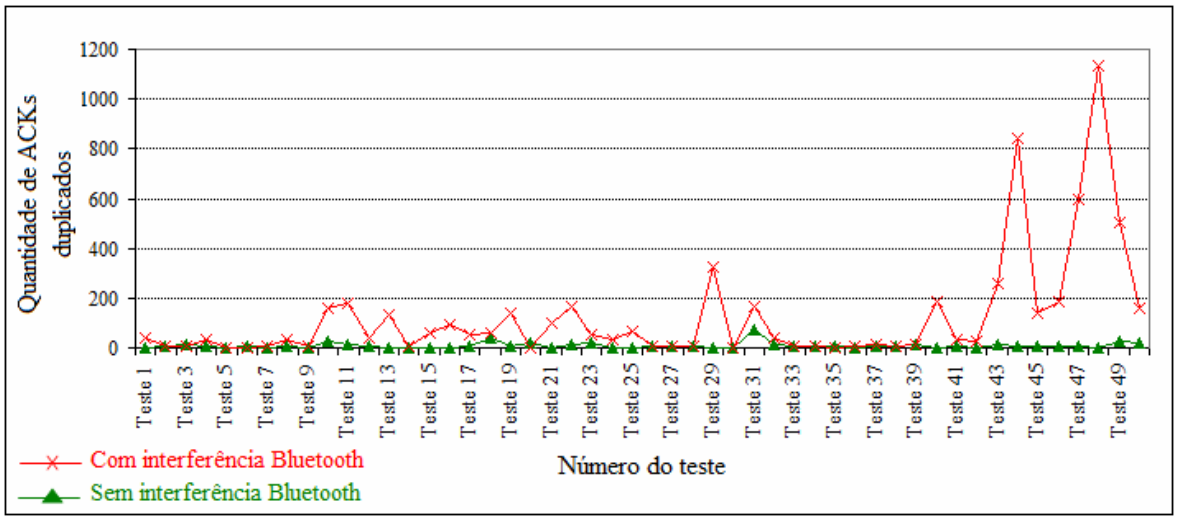


Figura 8.29 – *Quantidade de ACKs duplicado: distância de 1,60 m.*

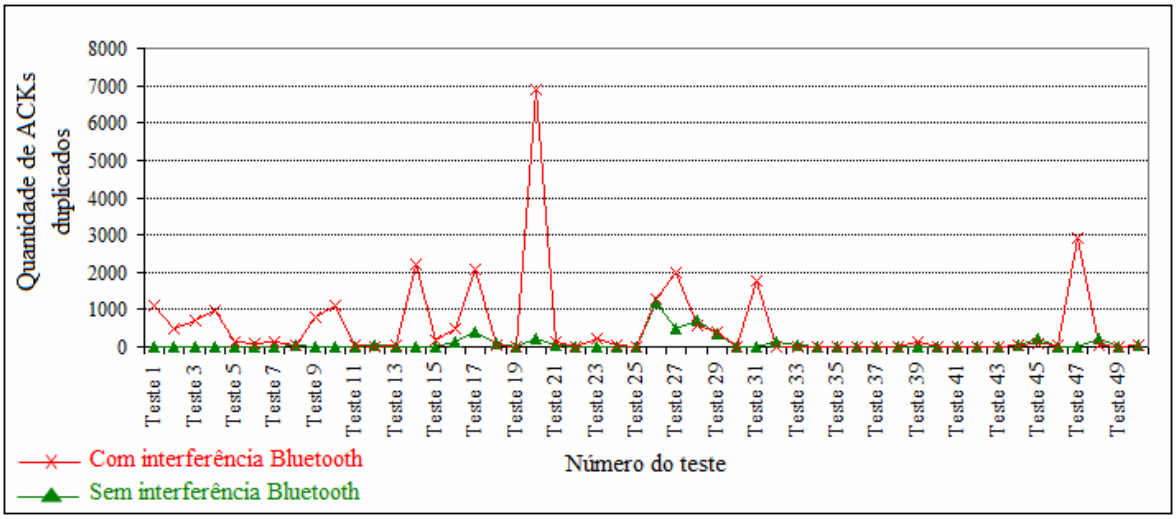


Figura 8.30 – *Quantidade de ACKs duplicado: distância de 2,60 m.*

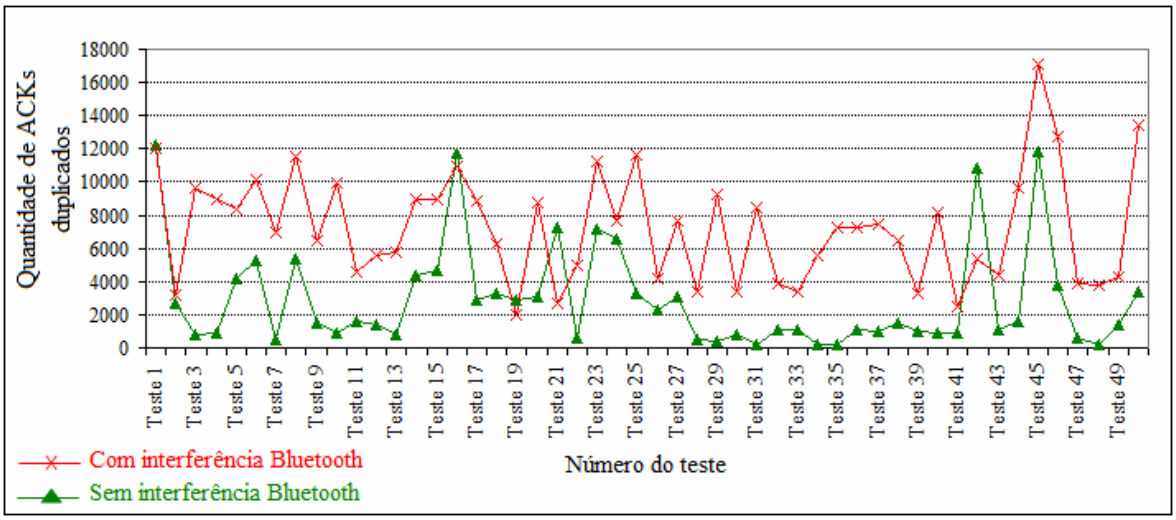


Figura 8.31 – *Quantidade de ACKs duplicado: distância de 4,60 m.*

Analisando as Figuras 8.29 e 8.30, verifica-se que a quantidade de ACKs duplicados para o cenário com interferência *Bluetooth* é superior em relação ao cenário sem interferência, apesar de em alguns trechos os dois ambientes de testes apresentarem uma quantidade de ACKS duplicados semelhante. A diferença de desempenho é maior para os testes realizados na distância de 4,60 m, conforme pode ser visualizado na Figura 8.31. Nesse caso, fica mais evidente que a interferência *Bluetooth* gerou uma latência alta na rede 802.11g.

Comparando-se as Figuras 8.29, 8.30 e 8.31, verifica-se que para a distância de 4,60 m, o sistema 802.11g apresentou maior instabilidade.

Um fato interessante de ser notado é que, ao se comparar as Figuras 8.29 com 8.24, 8.30 com 8.25 e 8.31 com 8.26, verifica-se que as curvas apresentam um comportamento bastante semelhante.

Analisando-se agora cada cenário de teste individualmente, foi verificado que a latência da rede 802.11g foi aumentando à medida que se incrementava a distância entre o transmissor e o receptor, como pode ser observado nas Figuras 8.32 e 8.33.

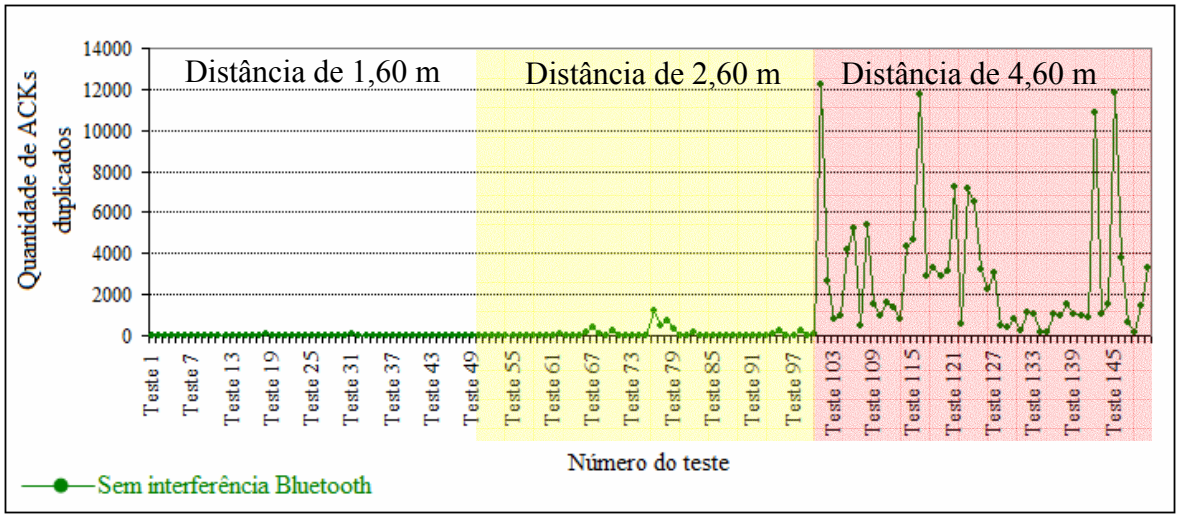


Figura 8.32 – *Evolução da quantidade de ACKs duplicados: sem interferência Bluetooth.*

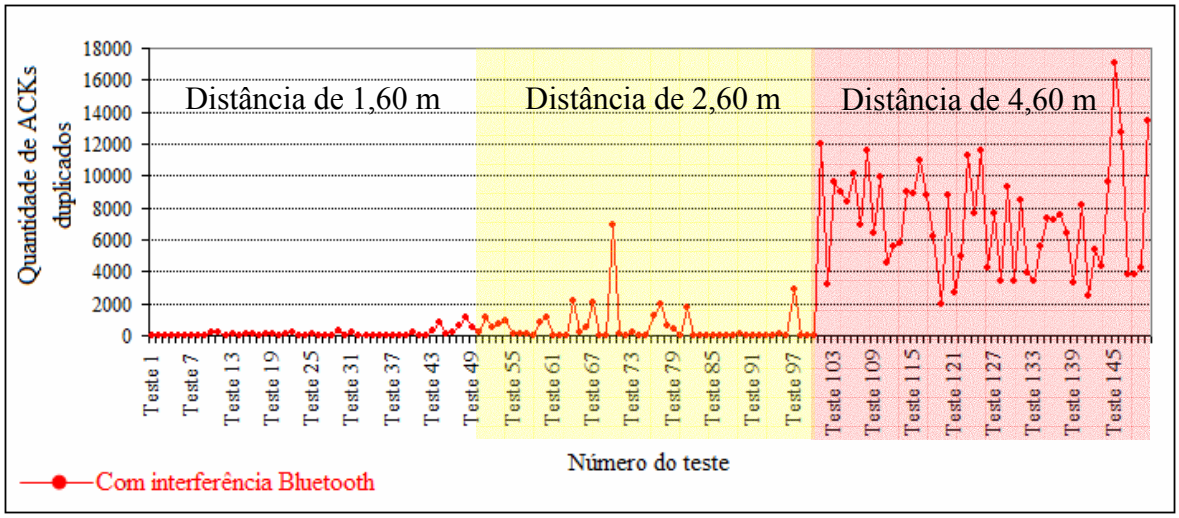


Figura 8.33 – *Evolução da quantidade de ACKs duplicados: com interferência Bluetooth.*

Observando as Figuras 8.32 e 8.33, nota-se que a latência da rede *Wi-Fi* aumenta significativamente para a distância de 4,60m, independentemente do cenário de testes. Fazendo-se uma análise comparativa entre as Figuras 8.32, 8.17 e 8.12 e entre as Figuras 8.33, 8.18 e 8.13, verifica-se que uma maior quantidade de pacotes ACK duplicados gerou uma taxa média de transmissão de pacotes baixa que, por sua vez, causou um elevado tempo de transmissão.

8.6 Box Plot e valores totais

Nesta subseção, será apresentado, por meio do uso de Box Plot, um resumo estatístico para os cinco parâmetros analisados neste trabalho dissertativo. A estatística foi gerada com base nos resultados experimentais obtidos para as distâncias de 1,60 m, 2,60 m

e 4,60 m, tanto para o cenário com interferência *Bluetooth* quanto para o cenário sem interferência.

Também será exposto aqui o total acumulado para os seguintes parâmetros: tempo de transmissão, quantidade de pacotes perdidos e quantidade de ACKs duplicados.

O Box Plot é uma ferramenta estatística desenvolvida por John W. Tukey que resume, em um único gráfico, as seguintes informações: o valor máximo, o terceiro quartil, a mediana, o primeiro quartil e o valor mínimo. O máximo é o maior valor da distribuição; o terceiro quartil é o valor abaixo do qual estão 75% dos dados; a mediana é o valor abaixo do qual estão 50% dos dados; o primeiro quartil é o valor abaixo da qual estão 25% dos dados; e o valor mínimo é o menor valor da distribuição.

As Figuras 8.34, 8.35, 8.36, 8.37 e 8.38 mostram, respectivamente, os gráficos Box Plot do tempo de transmissão, taxa de transmissão de pacotes, taxa de transmissão de dados, pacotes perdidos e ACKs duplicados.

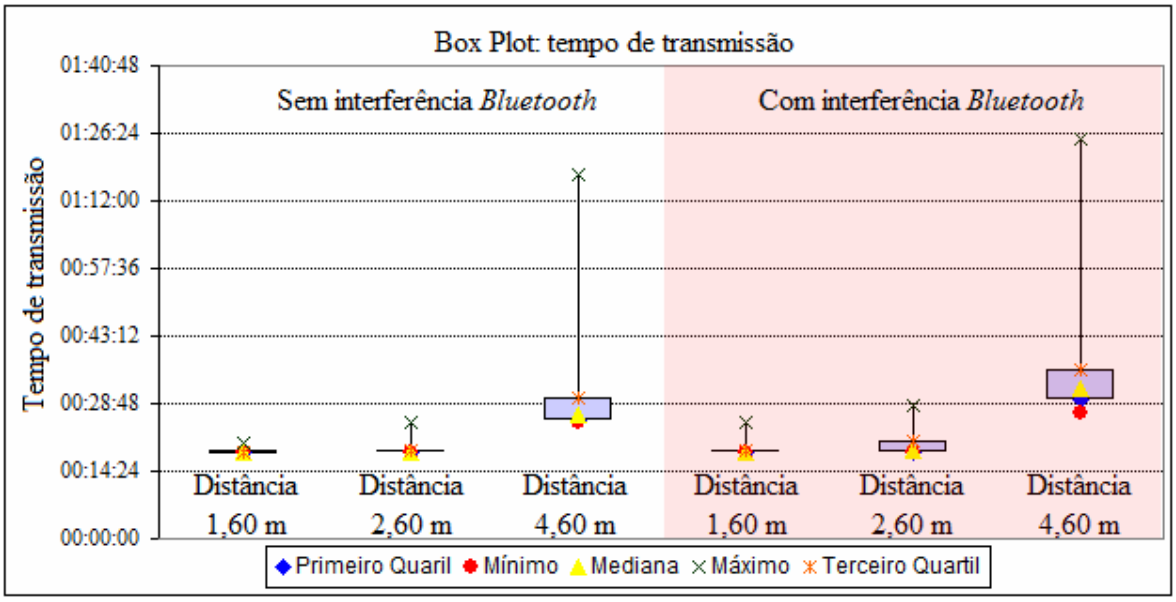


Figura 8.34 – *Box Plot: tempo de transmissão.*

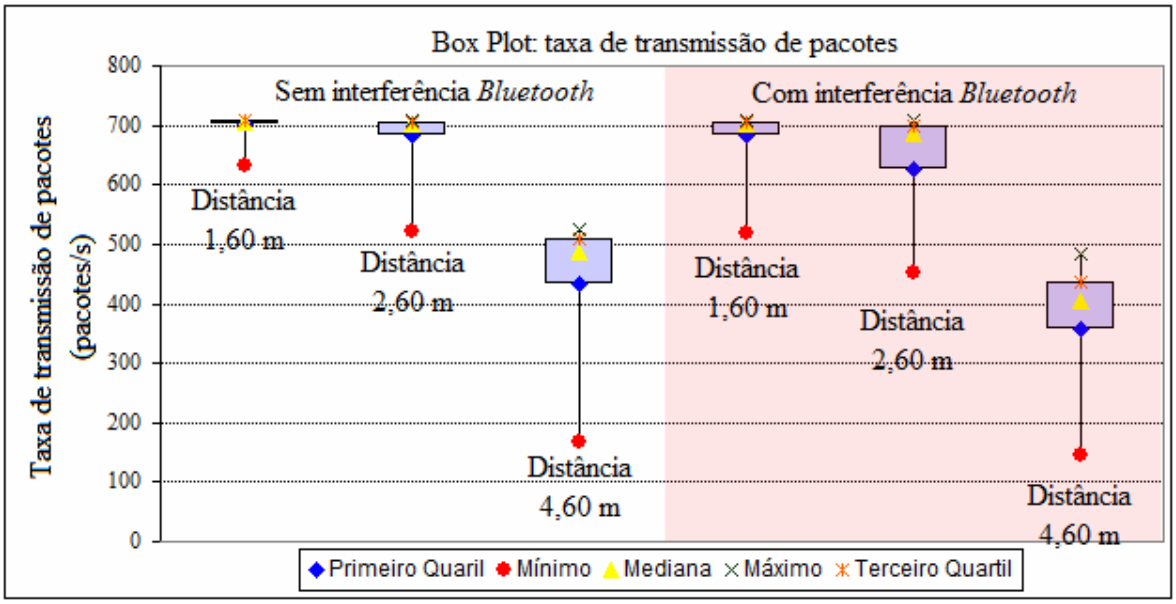


Figura 8.35 – *Box Plot: taxa de transmissão de pacotes.*

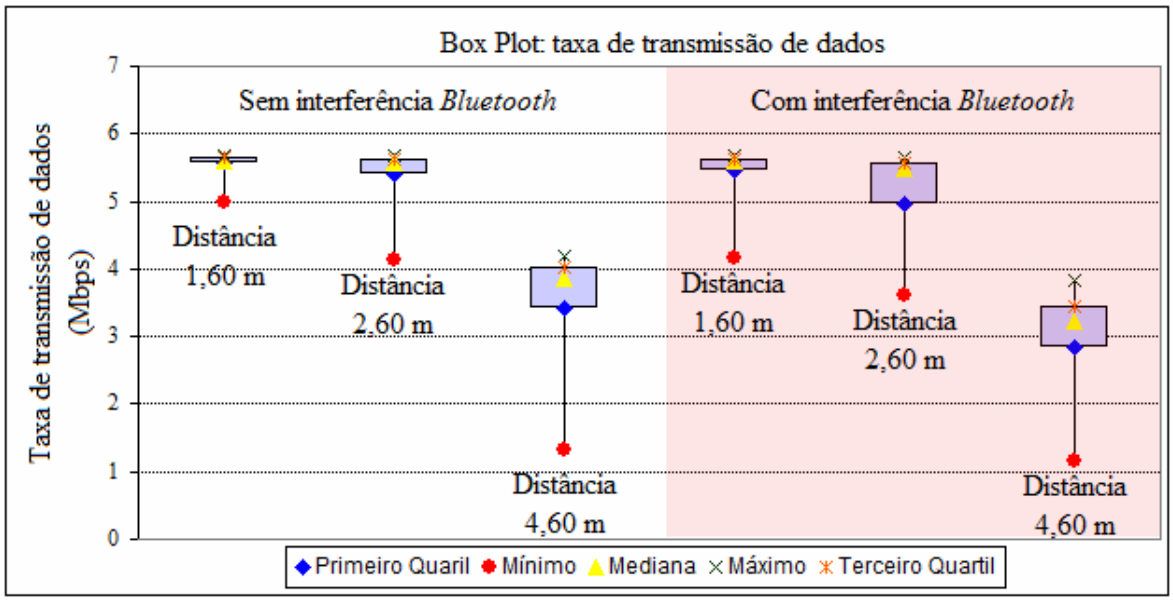


Figura 8.36 – *Box Plot: taxa de transmissão de dados.*

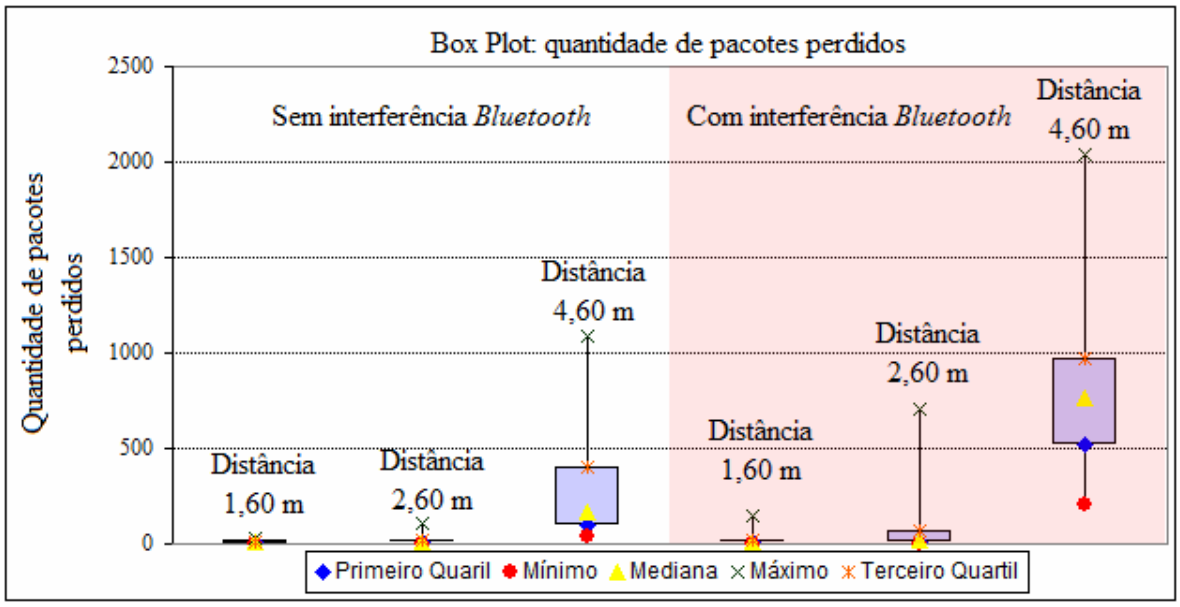


Figura 8.37 – *Box Plot: pacotes perdidos.*

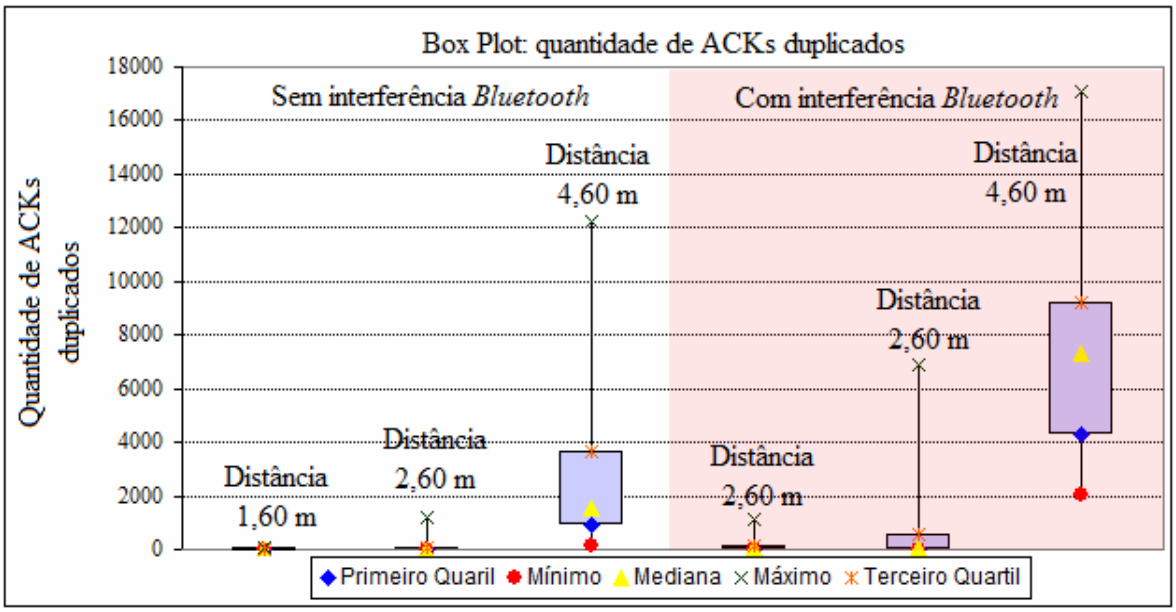


Figura 8.38 – *Médias:ACKs duplicados.*

Na Figura 8.34, pode-se observar que com o aumento da distância, o valor máximo da distribuição foi se tornando cada vez maior, tanto para o cenário com quanto para o cenário sem interferência *Bluetooth*. Para a distância de 4,60 m, para os dois cenários de testes, verifica-se que os resultados experimentais apresentam maior dispersão. Em contrapartida, os resultados experimentais obtidos para os testes realizados sem interferência *Bluetooth* a 1,60 m apresentam baixíssima dispersão.

Analisando a Figura 8.35, observa-se, para os dois cenários de testes, uma queda na taxa de transmissão de pacotes e aumento na dispersão dos dados com o incremento da distância. É possível verificar também que os piores resultados experimentais foram

obtidos para a distância de 4,60 m, independentemente de se ter ou não interferência *Bluetooth*. A mesma análise se aplica para a taxa de transmissão de dados, ilustrada na Figura 8.36.

Como pode ser visto na Figura 8.37, para todas as distâncias, a quantidade máxima de pacotes perdidos quando havia interferência *Bluetooth* foi muito superior ao sem interferência, com essa diferença sendo mais acentuada para as distâncias de 2,60 m e 4,60 m. Para essa última distância de testes, a dispersão dos resultados foi bastante acentuada. Por exemplo, para a distância de 4,60 m com interferência *Bluetooth*, temos os seguintes valores de Box Plot:

- máximo: 2.039;
- terceiro quartil: 966,5;
- mediana: 763;
- primeiro quartil: 519,750;
- mínimo: 203.

Por esses resultados, vê-se uma grande diferença entre os valores máximo e mínimo registrados.

Em se tratando da quantidade de ACKs duplicados, como pode ser constatado na Figura 8.38, o valor máximo contabilizado desses pacotes foi bastante superior no cenário com interferência *Bluetooth* do que no cenário sem interferência para todas as distâncias. A Figura 8.38 mostra que os resultados obtidos para a quantidade de pacotes ACKs duplicados para a distância de 4,60 m apresentaram grande dispersão quando comparados com aqueles obtidos para as distâncias de 1,60 m e 2,60 m para ambos os cenários de testes.

Serão mostradas agora as quantidades totais para o tempo de transmissão, quantidade de pacotes perdidos e quantidade de ACKs duplicados. Esses totais foram obtidos para as distâncias de 1,60 m, 2,60 m e 4,60 m tanto para o cenário de testes sem interferência *Bluetooth* quanto para o cenário com interferência *Bluetooth*.

As Figuras 8.39, 8.40 e 8.41 mostram, respectivamente, as quantidades totais para o tempo de transmissão, pacotes perdidos e ACKs duplicados.

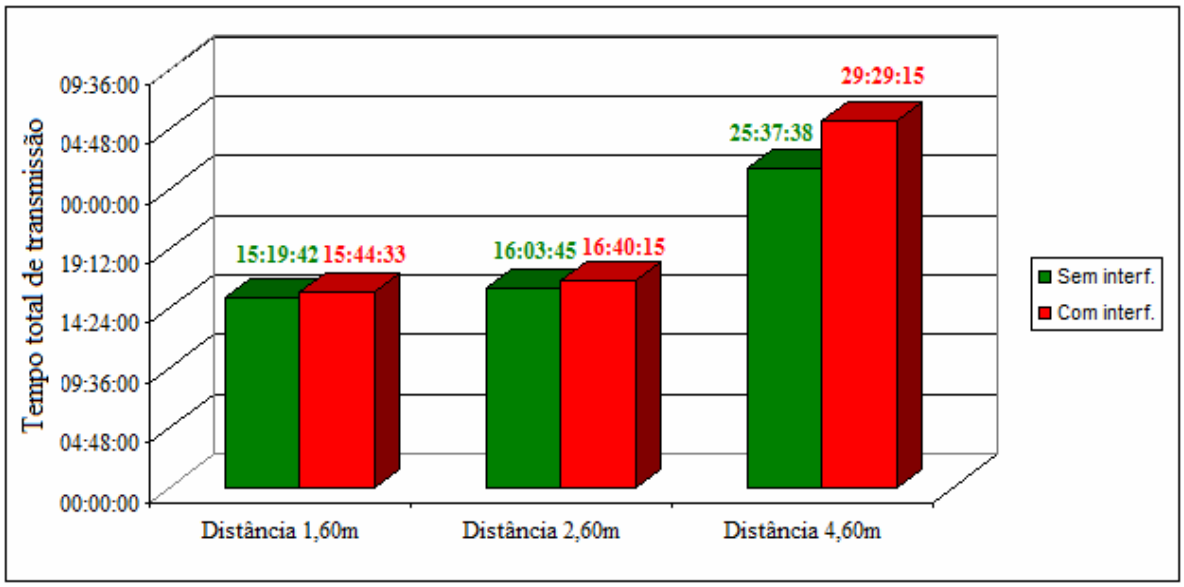


Figura 8.39 – *Total acumulado: tempo de transmissão.*

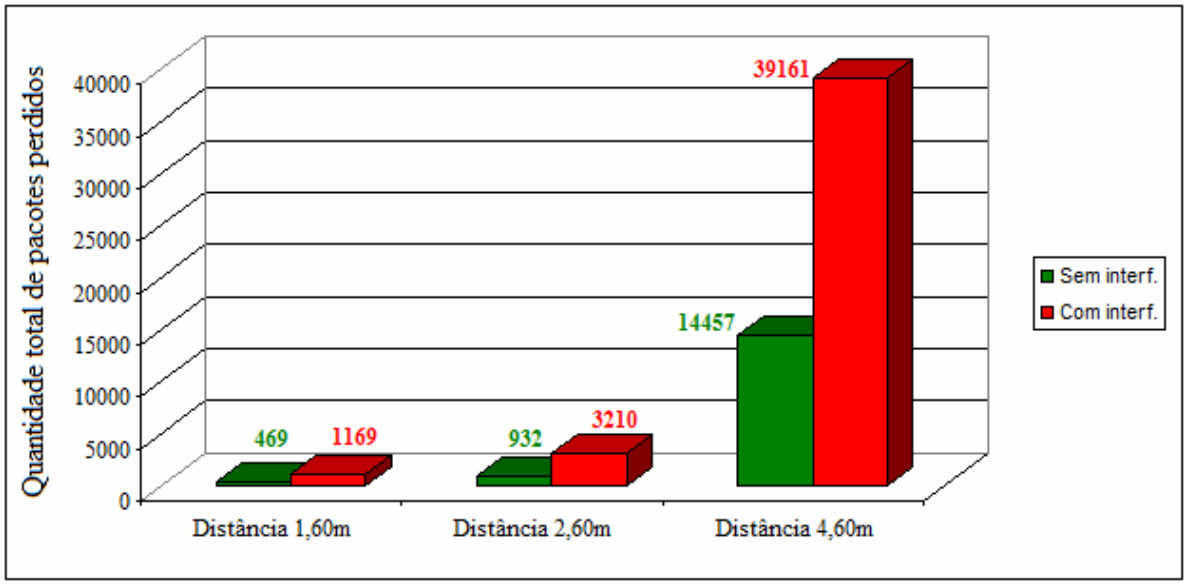


Figura 8.40 – *Total acumulado: pacotes perdidos.*

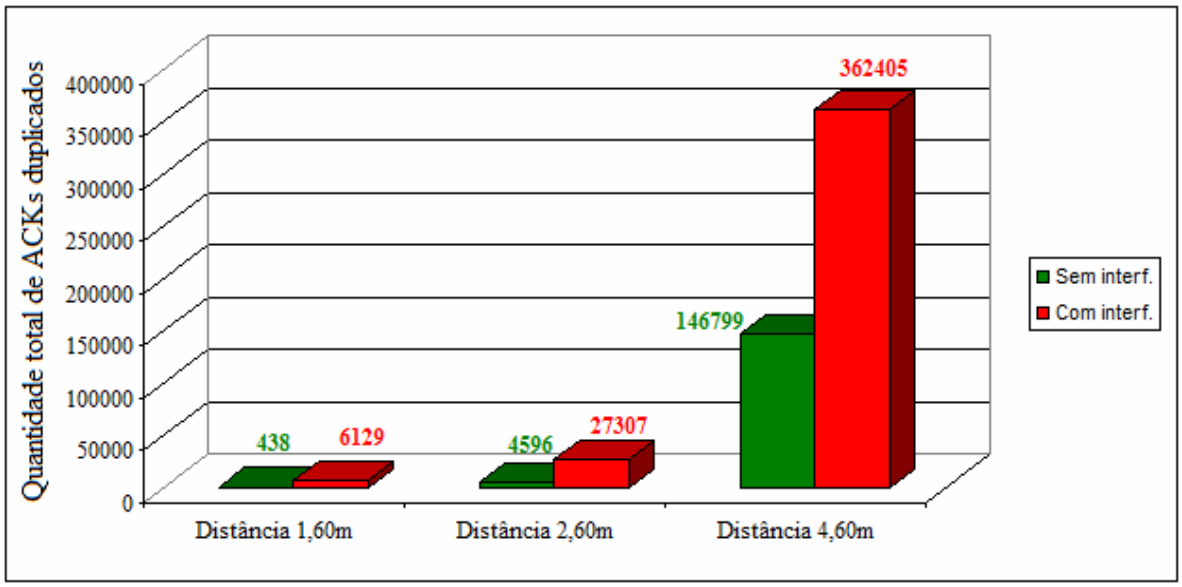


Figura 8.41 – *Total acumulado: ACKs duplicados.*

Analisando a Figura 8.39, vemos que, para todas as distâncias, o tempo total de transmissão sob a presença de interferência *Bluetooth* foi consideravelmente superior. Para a distância de 4,60 m, a diferença entre o cenário com interferência *Bluetooth* e sem interferência *Bluetooth* foi maior do que para as distâncias de 1,60 m e 2,60 m. Vemos pela Figura 8.39 que, mesmo quando não havia fonte interferente *Bluetooth*, o tempo total de transmissão na distância de 4,60 m foi muito superior ao tempo total de transmissão para as demais distâncias com interferência *Bluetooth*.

Partindo-se agora para a análise da Figura 8.40, constata-se que a quantidade total de pacotes perdidos sob a presença de interferência *Bluetooth* para todas as distâncias foi significativamente superior quando comparado com a quantidade total sem interferência. Na Figura 8.40, verifica-se que mesmo para o caso sem interferência *Bluetooth* da distância de 4,60 m, a quantidade total de pacotes perdidos é muito maior do que para os casos com interferência *Bluetooth* das distâncias de 1,60 m e 2,60 m.

Para o caso da quantidade total de ACKs duplicados ilustrado na Figura 8.41, verifica-se que, para todas as distâncias, a quantidade total desses pacotes foi muito superior quando havia, simultaneamente a transmissão 802.11g, a transmissão *Bluetooth*. A quantidade total de pacotes ACKs duplicados para ambos os cenários na distância de 4,60 m foi largamente superior às quantidades para as distâncias de 1,60 m e 2,60 m.

8.7 Nível de sinal da rede 802.11g

O estudo do comportamento da propagação de ondas eletromagnéticas é um fenômeno físico complexo [44]. De Souza e D. Lins realizaram em [44] um estudo sobre

os vários modelos encontrados na literatura que fazem uma estimativa do comportamento do sinal de radiofreqüência de redes sem fio IEEE 802.11 na faixa de 2,4 GHz. Nesse trabalho, foi investigada a influência da temperatura e da umidade relativa do ar na atenuação do sinal, sendo proposto um novo modelo de propagação. De Souza e D. Lins concluíram em [44] que os efeitos da umidade relativa do ar sobre a propagação do sinal têm efeitos consideráveis.

Em virtude do estudo realizado em [44] e de a distância entre o transmissor e o receptor ter sido variada nos experimentos realizados, achou-se de fundamental importância identificar o nível de sinal da rede 802.11g que chega ao receptor para as distâncias de 1,60 m, 2,60 m e 4,60 m.

As medições do nível de sinal recebido em cada uma das distâncias foram realizadas primeiro por aproximadamente 10 minutos sem interferência *Bluetooth*, seguidamente por cerca de mais 10 minutos com interferência *Bluetooth*. Da mesma maneira, foram feitas medições do nível de sinal no transmissor, primeiramente sem interferência *Bluetooth* e depois com interferência *Bluetooth*.

As medições do nível de sinal no transmissor foram realizadas utilizando o *software Netstumbler* versão 0.4.0. As medições no receptor foram realizadas com o software inSSIDer versão 1.0.13.0926. O *Netstumbler* e o inSSIDer são *softwares* gratuitos e podem ser encontrados, respectivamente, em [45] e [46].

A Figura 8.42 mostra o nível de sinal no transmissor 802.11g (*desktop*) antes e após ser ativado o *Bluetooth* no transmissor.

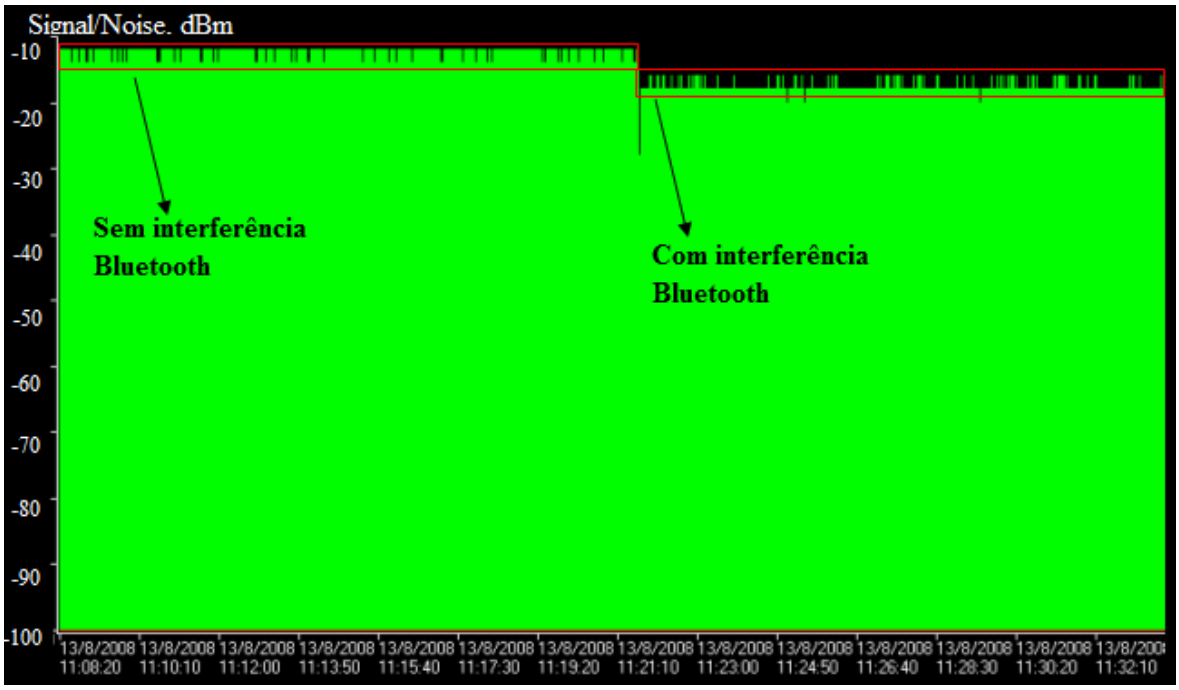


Figura 8.42 – *Nível de sinal no transmissor Wi-Fi antes e depois de ser ativado o Bluetooth.*

Antes de ser ativado o *Bluetooth*, o nível de sinal variava entre -12 dBm e -14 dBm. Após ser ativado o *Bluetooth*, o sinal variava em torno de -18 dBm e -20 dBm. Como poder ser visto na Figura 8.42, observa-se uma queda no nível de sinal com a ativação do *Bluetooth*.

As Figuras 8.43, 8.44 e 8.45 mostram a evolução, para as distâncias de 1,60 m, 2,60 m e 4,60 m, do nível de sinal no receptor *Wi-Fi* (*laptop*) antes e depois de ser ativado o *Bluetooth*.

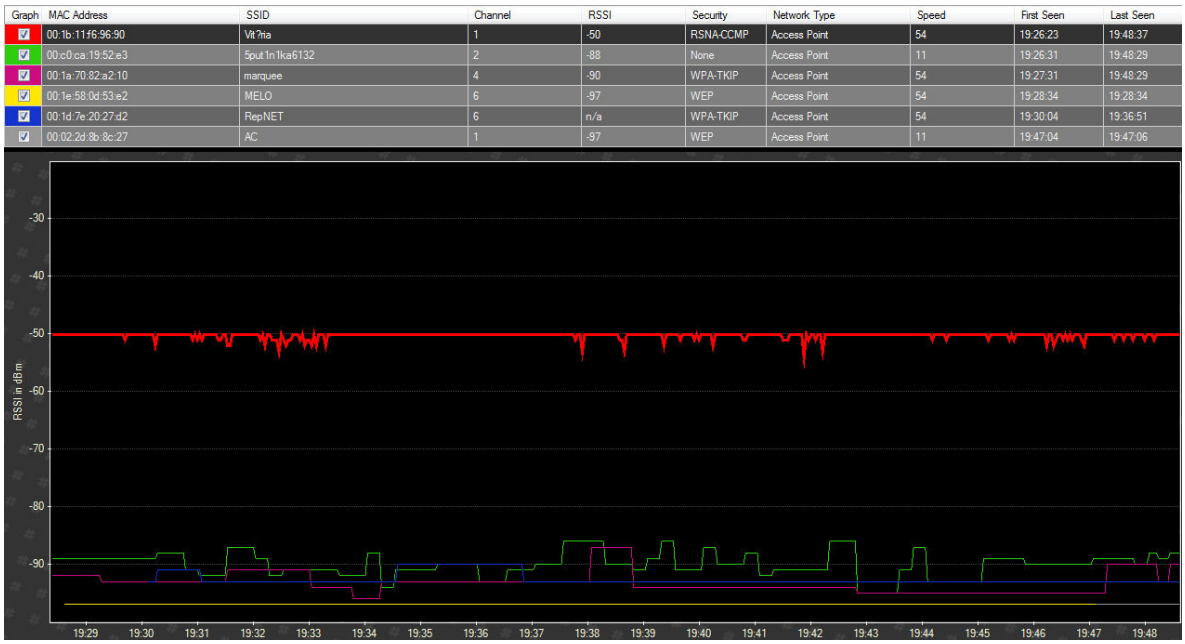


Figura 8.43 – *Nível de sinal no receptor Wi-Fi: distância de 1,60 m.*

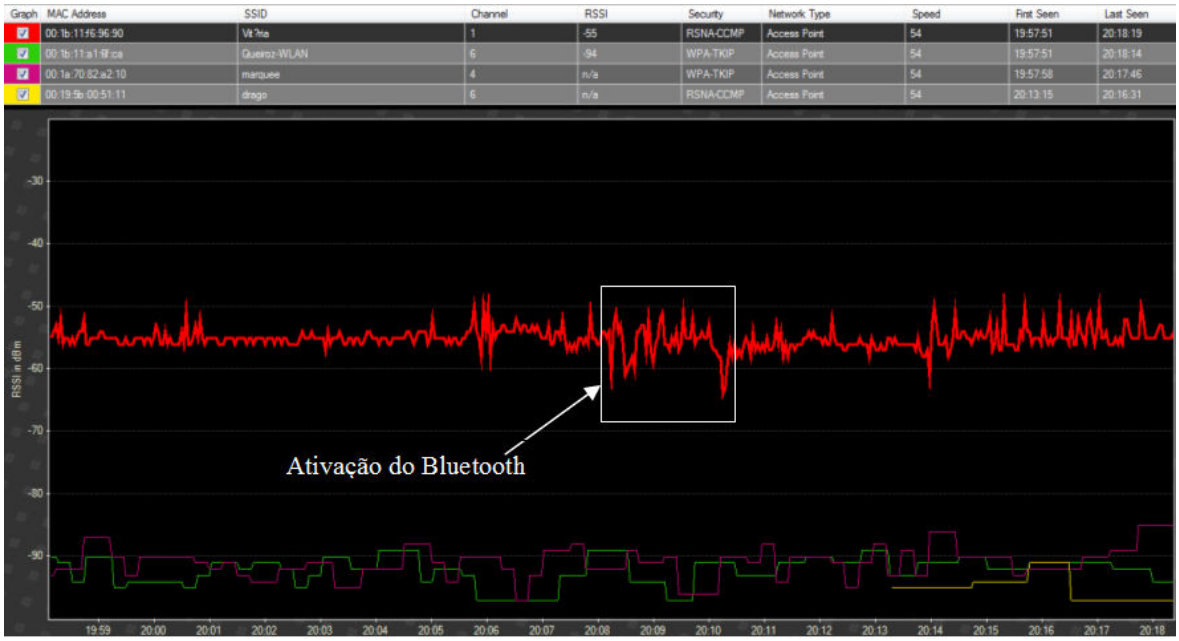


Figura 8.44 – *Nível de sinal no receptor Wi-Fi: distância de 2,60 m.*

Figura 8.45 – *Nível de sinal no receptor Wi-Fi: distância de 4,60 m.*

Como pode ser visto na Figura 8.43, o nível de sinal recebido da rede sem fio para a distância de 1,60 m ficou bastante estável em torno de -50 dBm, não sofrendo mudança no comportamento após a ativação do *Bluetooth*.

Analisando a Figura 8.44, observa-se que o nível de sinal recebido apresenta maior oscilação quando comparado com o da Figura 8.43, variando em torno da faixa de -55 dBm. Verifica-se pela Figura 8.44 que houve uma perturbação no nível de sinal recebido no momento em que o *Bluetooth* foi ativado, chegando a um valor em torno de -65 dBm.

Observa-se também na Figura 8.44 que, após a ativação do *Bluetooth*, a amplitude de variação do sinal ficou maior.

Para a distância de 4,60 m, verifica-se na Figura 8.45 que o nível de sinal caiu em relação às Figuras 8.43 e 8.44, variando agora em torno do valor de -60 dBm. Para essa distância, verificou-se que o nível de sinal recebido apresentou maiores amplitudes em sua variação, atingindo, em dois momentos do monitoramento, valores abaixo de -70 dBm. No momento da ativação do *Bluetooth* sinalizado na Figura 8.45, observa-se que ocorreu uma perturbação no nível de sinal recebido. Ao final do período de monitoramento, o nível de sinal retorna a apresentar uma maior instabilidade.

No momento em que se estava monitorando o nível de sinal recebido na distância de 1,60 m, identificaram-se cinco outras redes sem fio, das quais, uma estava ativada também no canal 1, duas no canal 6, uma no canal 2 e uma no canal 4.

Durante o monitoramento na distância de 2,60 m, identificaram-se três outras redes, das quais duas estavam no canal 6 e uma no canal 4. E durante o monitoramento para a distância de 4,60 m, foram identificadas quatro outras redes. Uma delas estava no canal 1, uma no canal 7, uma no 6 e outra no 5. Para isolar totalmente o cenário de testes de interferências externas, os experimentos teriam que ter sido realizados em um ambiente que não tivesse nenhuma outra rede sem fio ativa.

Capítulo 9. CONCLUSÕES E TRABALHOS FUTUROS

Nas Tabelas 9.1, 9.2 e 9.3 que se seguem, é feita, respectivamente para as distâncias de 1,60 m, 2,60 m e 4,60 m, uma comparação entre o maior valor registrado sem *Bluetooth* com o maior valor registrado com *Bluetooth*, e entre o menor valor registrado sem *Bluetooth* com o menor valor registrado com *Bluetooth*.

A Tabela 9.1 mostra, para os testes realizados na distância de 1,60 m, o valor máximo e mínimo registrado para cada um dos cinco parâmetros de desempenho sem e com interferência *Bluetooth*.

Tabela 9.1 – *Valores máximos e mínimos registrados na distância de 1,60 m.*

	Sem *Bluetooth*		**Com *Bluetooth***	
Parâmetro de desempenho	Máximo	Mínimo	Máximo	Mínimo
Tempo de TX	00:20:28	00:18:13	00:24:50	00:18:14
Taxa de TX de pacotes (pacotes/s)	709,226	630,961	708,204	520,038
Taxa de TX de dados (Mbps)	5,675	4,988	5,671	4,153
Pacotes perdidos	26	0	143	0
ACKs duplicados	76	0	1.133	1

Analisando a Tabela 9.1, podemos verificar que as diferenças mais significativas entre os cenários sem e com *Bluetooth* ocorrem entre os valores mínimos da taxa de transmissão de pacotes e taxa de transmissão de dados, e entre os valores máximos de pacotes perdidos e ACKs duplicados.

A Tabela 9.2 mostra os valores máximos e mínimos registrados de cada parâmetro sem e com interferência *Bluetooth* para a distância de 2,60 m.

Tabela 9.2 – *Valores máximos e mínimos registrados na distância de 2,60 m.*

	Sem *Bluetooth*		**Com *Bluetooth***	
Parâmetro de desempenho	Máximo	Mínimo	Máximo	Mínimo
Tempo de TX	00:24:44	00:18:13	00:28:32	00:18:16
Taxa de TX de pacotes (pacotes/s)	708,869	522,244	707,104	452,640
Taxa de TX de dados (Mbps)	5,671	4,140	5,662	3,607
Pacotes perdidos	107	1	710	0
ACKs duplicados	1.184	2	6.909	2

Da mesma maneira que nos resultados apresentados na Tabela 9.1, as diferenças mais significativas observadas na Tabela 9.2 são entre os valores mínimos da taxa de transmissão de pacotes e de dados e entre os valores máximos de pacotes perdidos e ACKs duplicados.

A Tabela 9.3 mostra os valores máximos e mínimos registrados de cada parâmetro sem e com interferência *Bluetooth* para a distância de 4,60 m.

Tabela 9.3 – *Valores máximos e mínimos registrados na distância de 4,60 m.*

	Sem *Bluetooth*		**Com *Bluetooth***	
Parâmetro de desempenho	Máximo	Mínimo	Máximo	Mínimo
Tempo de TX	01:17:36	00:24:37	01:25:15	00:26:44
Taxa de TX de pacotes (pacotes/s)	524,778	166,468	483,050	145,679
Taxa de TX de dados (Mbps)	4,201	1,333	3,828	1,144
Pacotes perdidos	1.088	35	2.039	203
ACKs duplicados	12.229	162	17.093	2.010

Analisando a Tabela 9.3, podemos verificar que apenas a diferença entre os valores mínimos do tempo de transmissão sem e com *Bluetooth* não foi muito significativa. A diferença entre os demais valores máximos sem e com *Bluetooth* e mínimos sem e com *Bluetooth* possui maior relevância.

Os resultados experimentais obtidos nesta dissertação permitem concluir que a interferência *Bluetooth* é mais prejudicial à rede 802.11g para distâncias maiores quando não há visada direta entre o transmissores e receptores *Wi-Fi* e *Bluetooth*. Esses resultados mostraram também que, mesmo para distâncias mais próximas entre transmissores e receptores e com visada direta, a presença do sistema *Bluetooth* provocou alguma perturbação na rede *Wi-Fi*.

Com relação à rede 802.11g, com o aumento da distância, mesmo sem uma obstrução direta entre o transmissor e o receptor, como nos casos das distâncias de 1,60 m e 2,60 m, o sinal sofre atenuação de espaço livre.

Considerando o obstáculo da parede de alvenaria, o que ocorreu para a distância de 4,60 m, o sinal sofre, além da atenuação de espaço livre, atenuação devido à obstrução imposta pelo obstáculo. Esse foi um fato que contribuiu para que o desempenho da rede *Wi-Fi* para a distância de 4,60 m fosse pior quando comparado ao desempenho para as distâncias de 1,60 m e 2,60 m. Outro fator que causou maior impacto no tráfego 802.11g

para a distância de 4,60 m foi o tamanho reduzido dos pacotes *Bluetooth*. Pacotes *Bluetooth* de tamanhos menores (tipo DM1) fazem com que a taxa de saltos em freqüência seja maior, aumentando a probabilidade de colisões.

Em cada teste realizado, foram transmitidos 775.211 pacotes entre a estação transmissora e receptora *Wi-Fi*. Dessa maneira, como foram realizados 50 testes com interferência *Bluetooth* em cada uma das três distâncias, tivemos uma quantidade total de 38.760.550 pacotes IEEE 802.11g transmitidos por distância sob interferência *Bluetooth*.

Baseado nos resultado práticos obtidos na seção 8.6, concluí-se que a probabilidade de colisão entre os pacotes dos dois sistemas foi de, respectivamente para as distâncias de 1,60 m, 2,60 m e 4,60 m, 0,003%, 0,008% e 0,101%. Esses resultados mostram também que existir interferência não necessariamente implica em erro. Ao compararmos essas probabilidades com as calculadas no capítulo 5, de 62,36% para as distâncias de 1,60 m e 2,60 m e de 58,87% para 4,60 m considerando o modelo de Hsu, Wei e C.C. [37] e de 78,36% para 4,60 m considerando o modelo de Jo e Jayant [38] e de Jim Zyren [35], confirmamos a inadequação dos modelos analíticos encontrados na literatura para prever a probabilidade de colisão entre pacotes IEEE 802.11g e *Bluetooth*, tanto do tipo DM1 quanto do tipo DM5.

Como sugestões de trabalhos futuros ficam:

- estudar a interferência *Bluetooth* em redes 802.11g em função do aumento do número de *piconets Bluetooth*;
- estudar a interferência entre os sistemas 802.11g e *Bluetooth* para um número maior tanto de estações 802.11 quanto de dispositivos *Bluetooth*;
- estudar o impacto na transmissão VoIP em redes 802.11g quando submetida à interferência *Bluetooth*, analisando o desempenho da rede em função do aumento de *piconets Bluetooth*;
- analisar o desempenho da rede 802.11g quando submetida a várias fontes interferentes na banda ISM, como fornos microondas e telefones sem fio.
- Determinar um modelo analítico para prever a probabilidade de colisão entre pacotes 802.11g e *Bluetooth* que leve em conta o incremento da distância, a presença de obstáculos e a quantidade de *piconets Bluetooth* interferentes.

REFERÊNCIAS

[1]BARBOSA, Douglas C. P.; GONDIM, Marcos A.A.; LINS, Rafael D.; DE SOUZA, Rafael S. *Uma Análise Comparativa da QoS do Skype, Yahoo! Messenger e Google Talk*. In: SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES (XXV : Set. 2007 : Recife). *Anais*. Recife, 2007.

[2]MCKAY, C.; MASUDA, F. *Empirical Studies of Wireless VoIP Speech Quality in the Presence of Bluetooth Interference*. Disponível em: http://www.ieee.org/portal/site. Acessado em 29/01/2008.

[3]http://www.uily.com/2006/saibamaiswireless.php. Acessado em 24/05/2008.

[4]FAINBERG, Michael.; GOODMAN, David. *Analysis of the Interference Between IEEE 802.11b and Bluetooth Systems*. Polytechnic University Electrical Engineering. Brooklyn, NY, 11201, USA.

[5]http://www.cert.pt/download/REC-WIFI.pdf. Acessado em 13/03/2008.

[6]KUROSE, James F.; ROSS Keith W. *Redes de computadores e a Internet: uma abordagem top-down*. 3ª ed. São Paulo: Pearson Addison Wesley, 2006.

[7]DOUFEXI, A.; ARUMUGAM, A.; ARMOUR, S. et. al. *An Investigation of the Impact of Bluetooth Interference on the Performance of 802.11g Wireless Local Área Networks*. **IEEE**, 2003. Disponível em: http://www.ieee.org/portal/site. Acessado em 29/01/2008.

[8]ROCHA DA SILVA, M. W. *Alocação de canal em redes sem fio IEEE 802.11 independentes*. Rio de Janeiro, 2006. Dissertação (Mestrado em Engenharia Elétrica) – Ciências em Engenharia Elétrica, Universidade Federal do Rio de Janeiro.

[9]Material de curso de especialização do CPdD (Centro de Pesquisa e Desenvolvimento em Telecomunicações) cedido via e-mail: O serviço VoIP em Wi-Fi.

[10]KIM, J.; TODE, H.; MURAKAMI, K. *Friendly Coexistance of Voice and Data Traffic in IEEE 802.11 WLANS*. **IEEE Transactions on Consumer Electronics**, v. 52, n. 2, Mai. 2006. Disponível em: http://www.ieee.org/portal/site. Acessado em 29/01/2008.

[11]MEDEPALLI, K.; GOPALAKRISHNAN, P.; FAMOLARI, D. et al. *Voice Capacity of IEEE 802.11b, 802.11a and 802.11g Wireless LANs*. **IEEE Communications Society Globecom**, 2004. Disponível em: http://www.ieee.org/portal/site. Acessado em 29/01/2008.

[12]GOLMIE, N.; VAN DYCK, R. E.; SOLTANIAN, A. *Interference of Bluetooth and IEEE 802.11: Simulation Modeling and Performance Evaluation*. National Institute of Standards and Technology .Gaithersburg, Maryland.

[13]TANENBAUM, Andrew S. *Computer Networks*. 4ª ed. New Jersey: Prentice Hall, 2003.

[14]http://www.teleco.com.br/tutoriais/tutorialrwlanman2/pagina_4.asp. Acessado em 08/03/2008.

[15]http://www.gta.ufrj.br/grad/00_2/ieee/CSMARTS.htm. Acessado em 27/02/2008.

[16]DAO, N. T.; MALANEY, R. A. *Throughput Performance of Saturated 802.11g Networks*. **The 2nd International Conference on Wireless Broadband and Ultra Wideband Communications**, 2007. Disponível em: http://www.ieee.org/portal/site. Acessado em 01/04/2008.

[17]http://www.gta.ufrj.br/~rezende/cursos/eel879/trabalhos/bluetooth/introducao.htm. Acessado em 07/12/2008.

[18]EDUARDO BILLO, Afonso. *Uma pilha de protocolos Bluetooth adaptável à aplicação*. Florianópolis, 2003. Disponível em: http://www.lisha.ufsc.br/~guto/teaching/theses/billo.pdf. Acessado em 22/04/2008.

[19]MCDERMOTT-WELLS, P. *What is Bluetooth?* Dez. 2004 / Jan. 2005. Disponível em: http://www.ieee.org/portal/site. Acessado em 01/04/2008.

[20]CONTI, M.; MORETTI, D. *System Level Analysis of the Bluetooth Standard*. **Proceedings of the Design, Automation and Test in Europe Conference and Exhibition**, 2005. Disponível em: http://www.ieee.org/portal/site. Acessado em 01/04/2008.

[21]STALLINGS, William. *Redes e Sistemas de Comunicação de Dados: teoria e aplicações corporativas.* 5ª ed. Rio de Janeiro: Elsevier, 2005.

[22]JOHANSSON, P.; KAZANTZIDIS, M.; KAPOOR, R. et al. *Bluetooth: An Enabler for Personal Area Networking*. **IEEE Network**, Set./Out. 2001. Disponível em: http://www.ieee.org/portal/site. Acessado em 01/04/2008.

[23]VASSIS, D.; KORMENTZAS, G.; ROUSKAS, A. et al. *The IEEE 802.11g Standard for High Data Rate WLANs*. **IEEE Network**, Mai./Jun. 2005. Disponível em: http://www.ieee.org/portal/site. Acessado em 01/04/2008.

[24]ABUKHARIS, S.; O'FARELL, T. *MPEG-2 Video Streaming over IEEE 802.11g WLAN in the Presence of Bluetooth Interference*. Disponível em: http://www.ieee.org/portal/site. Acessado em 01/04/2008.

[25]http://searchmobilecomputing.techtarget.com/searchMobileComputing/downloads/CWAP_ch8.pdf. Acessado em 06/04/2008

[26]MIYAMOTO, S.; HARADA, S.; MORINAGA, N. *Performance of 2.4 GHz-band Wireless LAN System using Orthogonal Frequency Division Multiplexing Scheme under Microwave Oven Noise Environment*. **IEEE**, 2005. Disponível em: http://www.ieee.org/portal/site. Acessado em 01/04/2008.

[27]ALBINO, José; SERRADOR, António. *Análise de Desempenho de Transações sobre Bluetooth*. Instituto Superior de Engenharia de Lisboa. Disponível em www.deetc.isel.ipl.pt/jetc05/JETC05/Artigos/Telecomunicacoes/Poster%20T/211.doc. Acessado em 20/08/2008.

[28]http://en.wikipedia.org/wiki/GFSK. Acessado em 10/06/2008.

[29]http://pt.wikipedia.org/wiki/Bluetooth. Acessado em 30/08/2008.

[30]http://pt.wikipedia.org/wiki/Time_division_duplex. Acessado em 30/08/2008.

[31]HAYKIN, Simon; MOHER, Michael. *Sistemas Modernos de Comunicações Wireless*. 1ª ed. Porto Alegre: Bookman, 2008.

[32]HOWITT, I.; HAM, S. UY. *Site Specific WLAN and WPAN Coexistance Evaluation*. **IEEE**, 2003. Disponível em: http://www.ieee.org/portal/site. Acessado em 06/05/2008.

[33]GHOSH, M.; GADDAM, V. *Bluetooth interference cancellation for 802.11g WLAN receivers*. **IEEE**, 2003. Disponível em: http://www.ieee.org/portal/site. Acessado em 06/05/2008.

[34]F. CHIASSERINI, Carla; R. RAO, Ramesh. *Coexistance Mechanisms for Interference Mitigation between IEEE 802.11 WLANs and Bluetooth*. **IEEE Infocom 2002**.

[35]ZYREN, Jim. *Reliability of IEEE 802.11 Hi Rate DSSS WLANs in a High Density Bluetooth Environment*. Junho de 1999. Disponível em http://www.seattlewireless.net/docs/99098r0P802-15_TG1-Reliability-of-P802-11-DS-BT_JZyren-Intersil.pdf. Acessado em 15/11/2008.

[36]GOLMIE, Nada; MOUVEAUX, Frederic. *Interference in the 2.4 GHz ISM Band: Impact on the Bluetooth Access Control Performance*. National Institute of Standards and Technology. Gaithersburg, Maryland.

[37]HSU, Alex Chia-Chun; WEI, David S.L.; C.C., Jay Kuo. *Coexistance Mechanism Using Dynamic Fragmentation for Interference Mitigation between Wi-Fi and Bluetooth*. Disponível em http://hsuchiachun.googlepages.com/DF_final-rev.pdf. Acessado em 15/11/2008.

[38]JO, J. H.; JAYANT, N. *Peformance Evaluation of Multiple IEEE 802.11b WLAN Stations in the Presence of Bluetooth Radio Interference*. **IEEE**, 2003. Disponível em: http://www.ieee.org/portal/site. Acessado em 06/05/2008.

[39]WONG, K. K.; O'FARELL, T. *Coverage of 802.11g WLANS in the Presence of Bluetooth Interference*. **The 14th IEEE 2003 International Symposium on Personal , Indoor and Mobile Radio Communication Proceedings**, 2003. Disponível em: http://www.ieee.org/portal/site. Acessado em 06/05/2008.

[40]RUKH, M. *Solutions to the WLAN & Bluetooth Interference*. **IEEE Computer Society**, 2008. Disponível em: http://www.ieee.org/portal/site. Acessado em 06/05/2008.

[41]WIRESHARK.EXE. Versão 1.0.2. Programa analisador de protocolo de rede. Gerald Combs e colaboradores. Linguagem Microsoft Visual C++ 6.0. Disponível em http://www.wireshark.org/. Acessado em 16/07/2008.

[42]http://www.gta.ufrj.br/grad/01_2/samba/smbcifs.htm. Acessado em 16/07/2008.

[43]http://www.novell.com/connectionmagazine/2007/q3/tech_talk_9.html. Acessado em 16/07/2008.

[44]DE SOUZA, Rafael S.; LINS, Rafael D. *Um Novo Modelo de Propagação para Redes Wi-Fi em 2,4GHz*. In: SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES (XXVI : Set. 2008 : Rio de Janeiro). *Anais*. Rio de Janeiro, 2008.

[45]NETSTUMBLER.EXE. Versão 0.4.0. Programa medidor de nível de sinal em redes Wi-Fi. Marius Milner. Disponível em http://www.vivasemfio.com/blog/downloads-vsf/. Acessado em 20/08/2008.

[46]INSSIDER.EXE. Versão 1.0.13.0926. Programa medidor de nível de sinal em redes Wi-Fi. Disponível em http://www.metageek.net/products/inssider. Acessado em 23/10/2008.

[47]PINTO, E. L.; ALBUQUERQUE, C. P. *A técnica de transmissão OFDM*. Revista Científica Periódica – Telecomunicações, v. 5, n. 1, Jun. 2002. Disponível em: http://revista.inatel.br/revista/. Acessado em 15/02/2008.

[48]http://www.efagundes.com/artigos/Arquivos_pdf/Wireless_LAN.PDF. Acessado em 29/03/2008.

[49]CAVALCANTI, Francisco R. P.; JR. FREITAS, Walter C.; DOS SANTOS, Ricardo B. et al. *Algoritmos de Alocação de Recursos de Rádio em Sistemas OFDMA*. In: SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES (XXV : Set. 2007 : Recife). *Anais*. Recife, 2007.

[50]http://cict.inatel.br/nova2/docentes/lucianol/Artigos/Incitel/constelacao_16QAM.pdf. Acessado em 03/06/2008.

[51]DE OLIVEIRA, HÉLIO M. *Fundamentos da Engenharia de Telecomunicações*. 1. ed. Recife: publicação do Departamento de Eletrônica e Sistemas – UFPE, 1998.

[52]GOES, Adriano A.; YAMAMOTO, J. S.; BRANQUINHO, Omar C. *Análise da Capacidade e Desempenho da Tecnologia de Transmissão de Rádio IP-OFDMA*. In: SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES (XXV : Set. 2007 : Recife). *Anais*. Recife, 2007.

[53]http://www.eletrica.ufpr.br/artuzi/apostila/cap5/16QAM-C.GIF. Acessado em 03/06/2008.

[54]http://en.wikipedia.org/wiki/Frequency-hopping_spread_spectrum. Acessado em 01/05/08.

[55]PERES, André; WEBER, Raul F. *Considerações sobre Segurança em Redes Sem Fio*. Disponível em: http://www.ppgia.pucpr.br/~maziero/pesquisa/ceseg/wseg03/07.pdf. Acessado em 13/03/2008.

[56]http://www.teleco.com.br/tutoriais/tutorialrwlanman3/pagina_3.asp. Acessado em 06/08/2007.

[57]http://en.wikipedia.org/wiki/Wired_Equivalent_Privacy. Acessado em 15/08/2008.

[58]http://en.wikipedia.org/wiki/RC4. Acessado em 15/08/2008.

[59]http://www.abc.org.br/~verissimo/textos/802117.jpg. Acessado em 25/05/2008

[60]http://en.wikipedia.org/wiki/Extensible_Authentication_Protocol. Acessado em 15/03/2008

[61]GUIMARÃES, Alexandre G.; LINS, Rafael D.; OLIVEIRA, Raimundo. *Segurança com Redes Privadas Virtuais*. 1ª ed. Recife: Brasport, 2007.

[62]http://pt.wikipedia.org/wiki/IEEE_802.11i. Acessado em 15/03/2008.

[63]http://pt.wikipedia.org/wiki/RSN. Acessado em 15/03/2008.

[64]http://en.wikipedia.org/wiki/CCMP. Acessado em 15/03/2008.

[65]http://www.microsoft.com/brasil/technet/Colunas/cableguy/cg0805.mspx#EFB. Acessado em 17/03/2008.

[66]http://en.wikipedia.org/wiki/BPSK. Acessado em 01/06/2008.

.

.

APÊNDICE A – A TÉCNICA DE MULTIPLEXAÇÃO OFDM

A técnica de multiplexação OFDM surgiu como uma evolução da técnica tradicional de multiplexação FDM (*Frequency Division Multiplexing*). Com FDM, existe uma banda de guarda entre as portadoras, porém, no OFDM, existe uma sobreposição espectral entre as subportadoras [47]. A Figura A.1 ilustra ambas as técnicas de multiplexação.

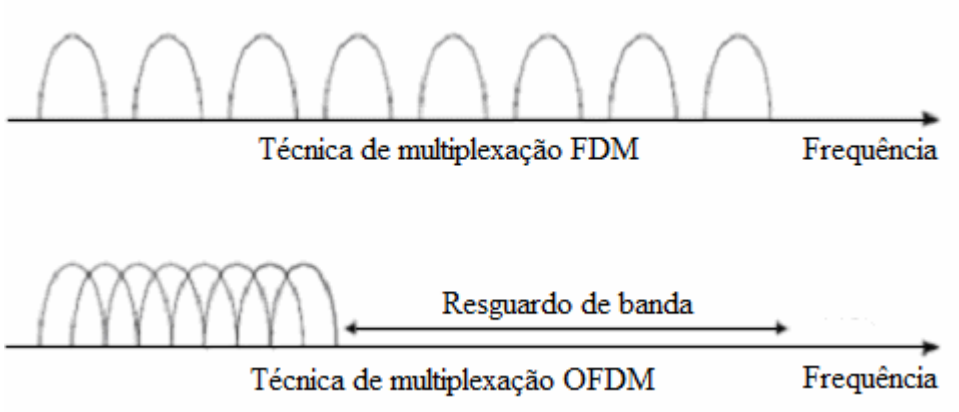


Figura A.1 – *Espectro gerado nas écnicas de multiplexação FDM e OFDM.*

Em sistemas de transmissão convencionais, os símbolos, ou seja, as palavras binárias, são enviados de uma maneira seqüencial através do uso de uma única portadora. Essa portadora é então modulada na taxa de símbolos da fonte de informação.

O OFDM consiste na transmissão paralela de dados em diversas subportadoras através da segmentação de uma portadora de freqüência elevada em várias subportadoras de freqüências mais baixas [47, 48]. Dessa forma, a informação a ser enviada é distribuída por entre as várias subportadoras utilizadas [49]. A taxa de transmissão por subportadora será menor quanto maior for o número de subportadoras empregadas.

A redução na taxa de transmissão por subportadora (o que causa um aumento do tempo de duração de cada símbolo) implica em uma diminuição da sensibilidade em relação à seletividade em freqüência [50]. Isso significa que a transmissão dos dados se torna mais robusta e menos sensível aos desvanecimentos devido a propagações multipercurso. Esse tempo de símbolo maior faz também com que a transmissão OFDM sofra menos com a ISI (*Inter-Symbol Interference*) [47, 49]. A largura de faixa dos subcanais de um sistema OFDM é dada pela divisão da largura de faixa total destinada ao sistema pelo número de subportadoras empregadas [47]. São utilizadas um total de 54 subportadoras, das quais 4 são subportadoras pilotos, utilizadas para sincronismo, e 48 são para dados [13].

Como a transmissão de um sinal OFDM ocorre em várias freqüências simultaneamente, essa técnica pode ser considerada uma forma de espalhamento espectral, porém, diferente do FHSS [13].

Dividir o sinal de transmissão em várias faixas estreitas apresenta vantagens quando comparado ao uso um único canal mais largo [13]. Pode-se obter, por exemplo, um grande benefício no que diz respeito ao desvanecimento seletivo em freqüência. Isso decorre do fato de que um possível desvanecimento seletivo em freqüência apresentado por um canal para uma transmissão de portadora única pode ser revertido em um desvanecimento plano apresentado pelas frações desse canal. Isso elimina ou reduz significativamente a necessidade de equalização.

Em um sistema OFDM, o espaçamento entre as subportadoras é cuidadosamente selecionado de forma que cada uma esteja centrada nos pontos em que o espectro das demais cruza o zero. Esse posicionamento está ilustrado na Figura A.2.

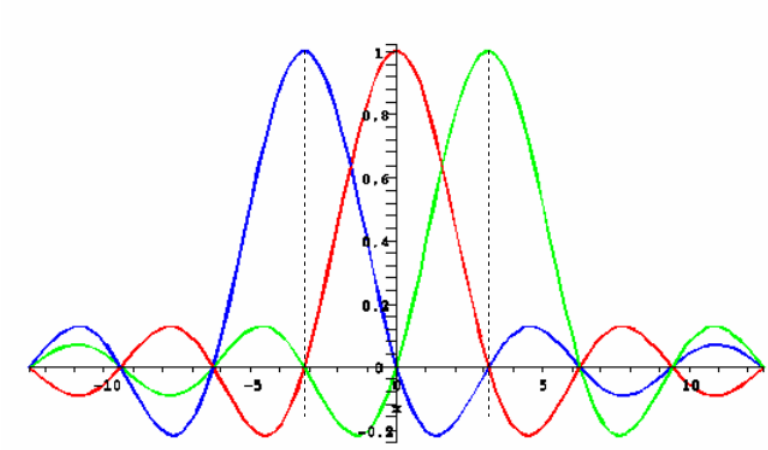


Figura A.2 – *Espaçamento entre as subportadoras na técnica OFDM (Extraído de [51]).*

Pela Figura A.2 vê-se que existe uma sobreposição espectral entre as subportadora e que cada uma delas está centrada nos pontos onde o espectro das demais cruza o zero. Embora exista essa sobreposição espectral entre as subportadoras moduladas, a informação conduzida por cada uma delas poderá ser isolada das demais através da projeção do sinal OFDM recebido sobre a subportadora a ele associado. A projeção desse sinal OFDM recebido sobre as demais subportadoras será zero. Isso ocorre pelo fato de existir ortogonalidade entre as subportadoras, a qual se deve ao espaçamento de freqüência empregado.

Vale salientar aqui que, conforme pode ser observado na Figura A.1, essa sobreposição espectral implementada pela técnica OFDM [47] gera uma economia de banda de aproximadamente 50%.

A implementação típica de um transmissor OFDM é ilustrada na Figura A.3.

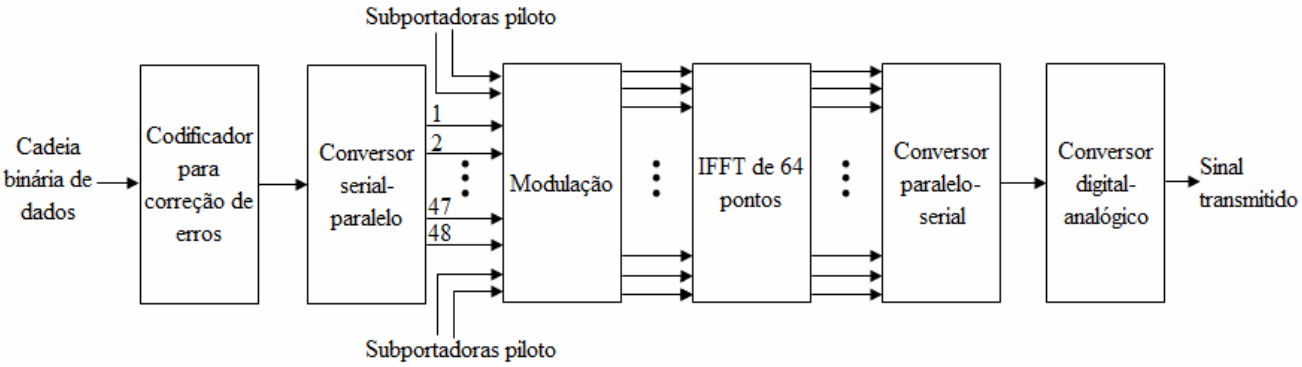


Figura A.3 – *Diagrama em blocos de um transmissor OFDM (Modificado de [31]).*

Primeiramente, a seqüência binária de dados que chega ao transmissor é codificada para correção de erros. Após essa codificação, a seqüência de dados passa por um dispositivo conversor serial-paralelo para criar as 48 cadeias de dados independentes que irão modular cada uma das 48 subportadoras de dados.

Após o processo de modulação, essas cadeias de dados independentes são submetidas a uma implementação eletrônica computacionalmente eficiente do IDFT (*Discrete Fourier Transform*), conhecida como IFFT. A IDFT converte amostras no domínio freqüência para amostras no domínio do tempo [31].

A saída do IFFT consiste em amostras no domínio do tempo a serem transmitidas pelo canal [50, 52]. Além das 48 subportadoras de dados, a Figura A.3 mostra as subportadoras piloto adicionais usadas pelo receptor para propósitos de sincronização. Na Figura A.3 é utilizado um dispositivo IFFT de 64 pontos, pois [31] a implementação computacionalmente eficiente dos algoritmos IFFT e FFT requer que o número de amostras seja um múltiplo de uma potência de 2.

A saída do bloco IFFT de 64 pontos consiste em 64 amostras no domínio do tempo, que serão convertidas de paralelo para série. Por fim, efetua-se uma conversão digital-analógico para facilitar a transmissão da seqüência de dados de entrada pela interface aérea.

A Figura A.4 apresenta a implementação correspondente ao receptor.

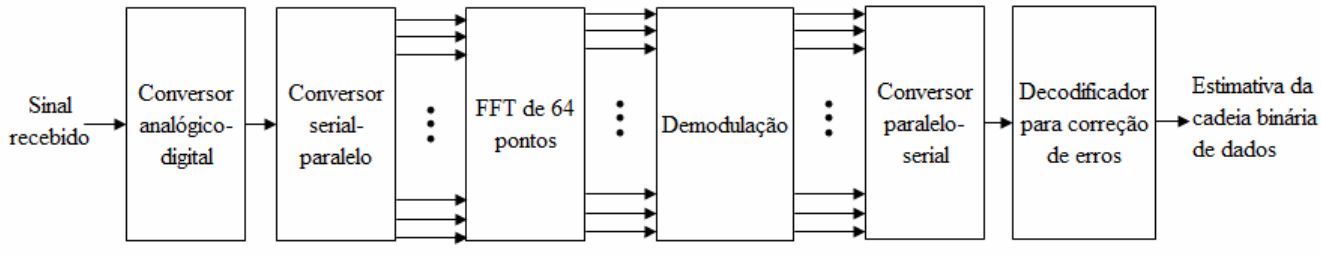


Figura A.4 – *Diagrama em blocos de um receptor OFDM (Modificado de [31]).*

Para recuperar a seqüência de dados de entrada original, o sinal recebido passa por uma seqüência de operações na ordem inversa daquela realizada no transmissor [31]:

- conversor analógico-digital;
- conversor serial-paralelo;
- algoritmo FFT de 64 pontos;
- demodulação;
- conversor paralelo-serial;
- decodificador para correção de erros.

As Figuras A.3 e A.4 são instrutivas de um ponto de vista computacional. A Figura A.5 ilustra um outro ponto de vista sobre o mesmo sistema da Figura A.3.

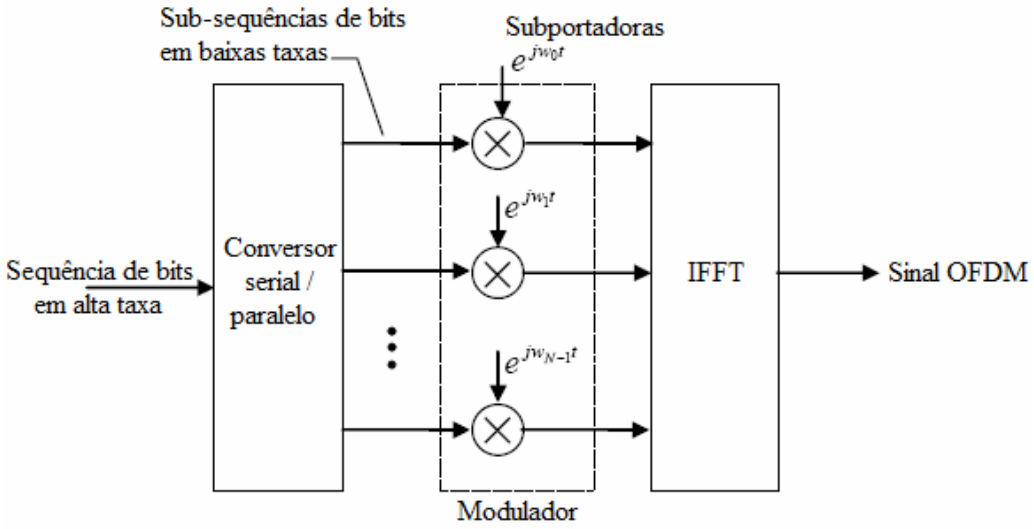


Figura A.5 – *Arquitetura básica de um sistema de transmissão OFDM (Modificado de [52]).*

O conjunto de subportadoras da Figura A.5 constituem um sistema ortogonal. A Figura A.6 apresenta outro ponto de vista sobre o mesmo sistema ilustrado na Figura A.4.

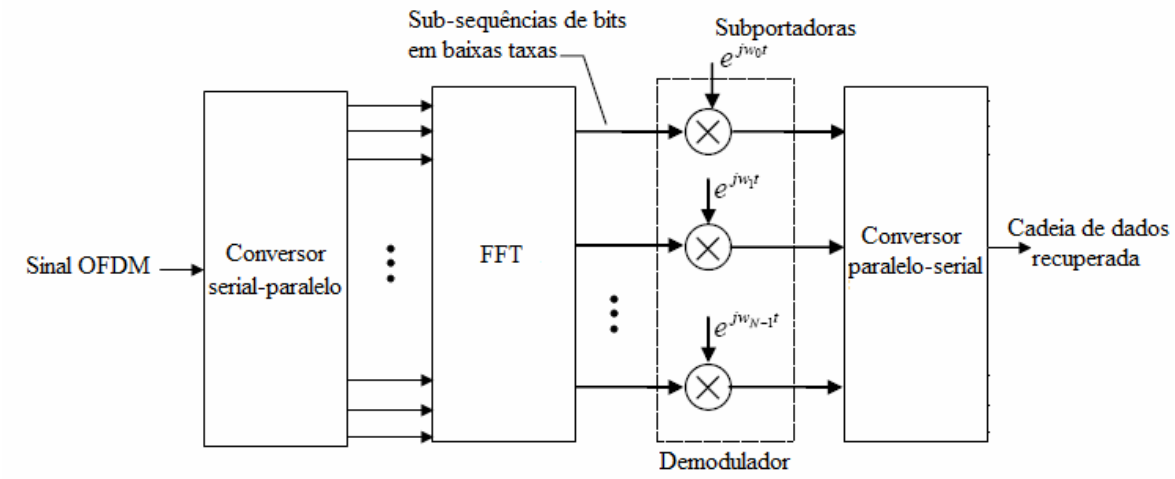


Figura A.6 – *Arquitetura básica de um sistema de recepção OFDM (Modificado de [31]).*

Bandas de segurança ou bandas de guarda são inseridas na seqüência de dados no transmissor OFDM de modo a reduzir ou eliminar totalmente a ISI. Para isso, um prefixo de guarda é adicionado ao começo de cada símbolo. Esse prefixo de guarda é uma cópia do final do símbolo que é inserida no começo de cada símbolo OFDM [49, 50, 54]. Naturalmente, a inclusão de prefixos resulta em aumento da largura de banda de transmissão do sinal OFDM. Em virtude de serem adicionados prefixos como bandas de guarda no conversor paralelo-serial do transmissor OFDM, eles deverão ser removidos no conversor serial-paralelo do receptor OFDM [31].

APÊNDICE B – ESPALHAMENTO ESPECTRAL FHSS

Em virtude do fato de os dispositivos *Bluetooth* operarem na compartilhada banda ISM de 2,4 GHz, algumas técnicas devem ser aplicadas para evitar problemas de interferência. As duas técnicas mais conhecidas que fazem o espalhamento espectral do sinal a ser transmitido amenizando interferências são a DSSS e a FHSS. O sistema *Bluetooth* utiliza FHSS.

A Figura B.1 ilustra de maneira simples a técnica FHSS.

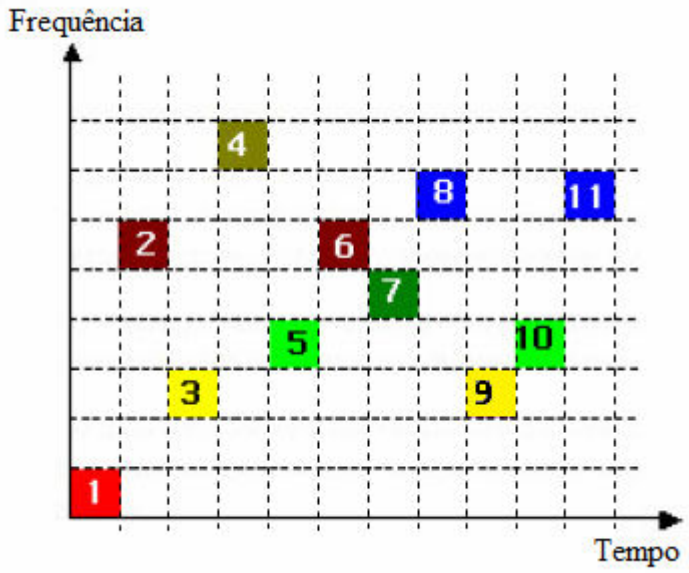


Figura B.1 – *A técnica FHSS*

A Figura B.1 ilustra o tempo divido em 11 *slots* de tempo e cada cor representa um canal (freqüência) distinto. No primeiro *slot* de tempo, a informação é transmitida em um determinado canal; no segundo, em outro, e assim por diante. As freqüências podem se repetir em diferentes *slots* de tempo, como acontece nos *slot*s 2 e 6, 3 e 9, 5 e 10 e 8 e 11.

Como mostra a Figura B.1, de um *slot* de tempo para outro acontecem saltos em freqüência. A ordem em que esses saltos ocorrem, ou seja, a ordem com que as freqüências são escolhidas ao longo do tempo, forma uma espécie de código. Conforme comentado anteriormente, esse código é pseudo-aleatório, diferente de transmissão para transmissão, e definido pelo mestre. Com o uso dessa técnica, além de se obter robustez no que diz respeito à interferência, consegue-se oferecer um certo nível de segurança. Afinal, só se consegue entender o que está sendo transmitido a partir do momento que se tiver conhecimento da maneira como são feitos os saltos em freqüência [18].

A largura de banda total necessária para uma transmissão FHSS é maior do que a que seria requerida para transmitir a mesma informação utilizando apenas uma portadora. Enquanto que essa técnica de transmissão não é muito eficaz contra interferências causadas por sinais de banda larga, ela reduz a degradação causada por interferências de sinais de banda estreita [54].

Um dos desafios no uso de sistema FHSS é o sincronismo entre o transmissor e o receptor. Uma abordagem é garantir que o transmissor usará todos os canais disponíveis durante um período fixo de tempo. O receptor pode então encontrar o transmissor selecionando um canal aleatório e monitorá-lo para detectar dados válidos sendo transmitidos nele. Os dados enviados pelo transmissor são identificados por uma seqüência especial de dados que não é normal que ocorra no canal escolhido. Uma outra abordagem consiste em o transmissor e o receptor utilizarem tabelas com seqüência de canais. Uma vez sincronizados, ambos podem manter a comunicação seguindo a tabela [54].

Um diagrama simplificado de um transmissor FHSS é apresentado na Figura B.2.

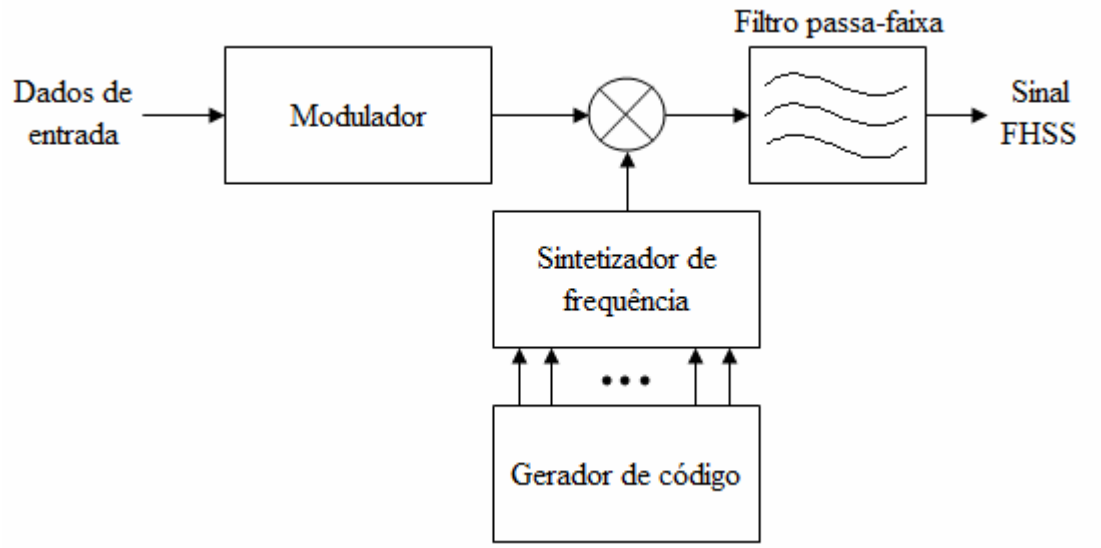


Figura B.2 – *Transmissor FHSS (Modificado de [31]).*

Na Figura B.2, o sintetizador de freqüência usa a saída do gerador de código e a saída do modulador para gerar um sinal na freqüência de transmissão desejada. Conforme exposto em [31], o sinal FHSS na saída da Figura B.2 pode ser expresso matematicamente como:

$$s(t) = \sqrt{E_s}\,\mathrm{Re}\{\exp(j(2\pi(f_i + \zeta_k + f_c)t + \phi_k))\} \qquad \text{(B.1)}$$

Na Equação B.1, E_s é a energia do símbolo transmitido, f_i é a representação de freqüência do símbolo atual, ξ_k é a freqüência do salto atual, f_c é a freqüência central da faixa e $\square_k$ é a fase.

A seqüência de saltos é escolhida em um padrão pseudo-aleatório e freqüentemente o número de saltos é selecionado como sendo uma potência de 2^m. Essa seqüência de saltos é gerada por um registrador de deslocamento. No FHSS, os bits do registrador são usados em uma quantidade m de cada vez, sendo que esses m bits formam um índice para a seleção da freqüência de salto. Se o registrador de deslocamento tem comprimento m, então utilizando m bits de cada vez equivale a usar o estado do registrador para a seleção da freqüência de salto [31].

No receptor, uma vez que a freqüência de um salto é determinada, o estado do registrador é conhecido e, assim, o padrão de salto pode ser predito. Se dois transmissores querem compartilhar o mesmo espectro, eles podem usar o mesmo padrão de salto, porém ambos têm que garantir que existe um deslocamento temporal entre suas transmissões.

Considerando que T_{salto} é o período de salto, ou seja, o intervalo de tempo em que o sistema permanece em uma determinada freqüência, e que T é o tempo de transmissão de um símbolo de dados, [31] pode-se ter um sistema FHSS com salto de freqüência lento ou um sistema FHSS com salto de freqüência rápido.

No FHSS com salto de freqüência lento, o período de salto T_{salto} é maior do que o tempo necessário para se transmitir um símbolo, ou seja, $T_{salto} > T$. Nessa situação, vários símbolos de dados são transmitidos na mesma freqüência de salto [31].

Em contrapartida, se o período de salto T_{salto} for menor do que o tempo de transmissão de um símbolo, ou seja, se $T_{salto} < T$, o mesmo símbolo de dados é transmitido utilizando diferentes freqüências de salto [31]. Nesse caso, tem-se um sistema FHSS com salto de freqüência rápido.

APÊNDICE C – A FFT E A IFFT

A avaliação da transformada de Fourier em casos práticos não é feita utilizando-se os procedimentos algébricos convencionais de cálculo. Isso se deve ao fato de que na maioria das vezes não se dispõe de uma expressão analítica para a função que se deseja analisar o espectro. Para tanto, a DFT (*Discrete Fourier Transform*) é bastante utilizada para se estudar o espectro de sinais, sendo determinada com o auxílio de um computador digital [51].

Tomando-se N amostras de um determinado sinal no domínio do tempo, denotadas por $f(k)$, com $k = 0, 1, 2,..., N-1$, a DFT é dada por um conjunto de N amostras desse sinal no domínio da freqüência, denotadas por $F(n)$, com $n = 0, 1, 2,..., N-1$. Conforme exposto em [51], a DFT é definida por:

$$F(n) := \frac{1}{N}\sum_{k=0}^{N-1} f(k)\exp\left(-\frac{j2\pi kn}{N}\right). \tag{C.1}$$

A obtenção do sinal no domínio do tempo pode se feita utilizando-se a IDFT (*Inverse Discrete Fourier Transform*), definida na Equação C.2:

$$f(k) := \sum_{k=0}^{N-1} F(n)\exp\left(\frac{j2\pi kn}{N}\right). \tag{C.2}$$

Logo, diz-se que $f(k)$ e $F(n)$ formam um par de transformadas.

Diversas transformadas, como a de Fourier, Gabor, Hilbert, *Wavelets*, etc vêm assumindo um papel de grande importância na Engenharia. Um bom exemplo diz respeito ao uso da DFT em diversos campos do conhecimento, principalmente em Engenharia Elétrica [51].

Em 1965, J.W. Cooley da IBM em conjunto com J.W. Tukey do Bell Labs desenvolveram um método revolucionário no tratamento digital de sinais, denominado de Transformada Rápida de Fourier (em inglês FFT – *Fast Fourier Transform*). A FFT realiza de maneira bastante eficiente o reagrupamento dos cálculos dos coeficientes de uma DFT [51].

Ao invés de calcular a DFT diretamente pela definição da Equação C.1, a FFT utiliza um algoritmo que viabiliza o cálculo da DFT com o menor esforço computacional. Segundo [51], o esforço computacional para avaliar uma DFT é da ordem de N^2. Já para o algoritmo FFT, esse esforço é da ordem de $2N\log_2 N$. A Tabela C.1 ilustra a eficiência do algoritmo FFT quando comparado a DFT para vários números de amostras.

Tabela C.1 – *Comparação entre a DFT e o FFT em termos de esforço computacional.*

N	N^2 (DFT)	$N \log_2 N$ (FFT)	Vantagem
2	4	2	2
4	16	8	2
8	64	24	2,67
16	256	64	4
32	1.024	160	6,4
64	4.096	384	10,67
128	16.384	896	18,29
256	65.536	2.048	32
512	262.144	4.068	56,89
1024	1.048.576	10.240	102,4
2048	4.194.304	22.528	186,18
4096	16.777.216	49.512	341,33
8192	671.088.964	106.496	630,15

Analisando a Tabela C.1, verifica-se a economia de esforço computacional ao se implementar a FFT.

É importante salientar que a FFT não é um tipo diferente de transformada, mas sim [51] uma técnica que possibilita determinar a DFT de maneira mais rápida e econômica.

Conforme dito anteriormente, vários sinais práticos de interesse, como voz, vídeo, entre outros, não possuem expressões analíticas para descrevê-los. A maneira mais usual de lidar com sinais dessa natureza é “calcular” a transformada através de um analisador de espectro. Contudo, [51] com o desenvolvimento das técnicas de processamento digital de sinais, a DFT surge como uma alternativa prática muito atrativa.

A grande redução no custo dos chips capazes de processar sinais digitais, o aumento da capacidade de processamento e o aparecimento de novas técnicas mais eficientes, como a FFT, vêm permitindo trabalhar em tempo real com vários sinais.

Para calcular uma FFT inversa, ou seja, uma IFFT, [51] usa-se praticamente o mesmo algoritmo da FFT direta, com algumas modificações.

APÊNDICE D – SEGURANÇA EM REDES SEM FIO

Ao contrário das redes cabeadas, as redes sem fio são caracterizadas por transmissões não-guiadas, em um meio comum e acessível a todos, dentro do raio de alcance das antenas de transmissão [5]. Imersas nesse cenário, caso as redes sem fio não estejam configuradas com mecanismos básicos de segurança, o acesso à rede fica disponível para qualquer usuário que esteja dentro do raio de cobertura do AP e com um terminal móvel compatível com o padrão utilizado.

Uma rede sem fio sem qualquer configuração de segurança mínima em seu AP se encontra em um estado que pode ser caracterizado por rede aberta [5]. Redes abertas podem ser acessadas por qualquer terminal móvel que tenha uma placa de rede *Wi-Fi*, desde que esteja dentro da área de cobertura do AP. Um exemplo simples de acesso a uma rede aberta é a de um usuário ingênuo que simplesmente liga o seu computador e acessa a Internet, sem perceber que está usando a conexão de uma entidade com uma rede sem fios aberta. Uma rede que possua configurações mínimas de segurança habilitadas exigirá de um possível intruso um maior esforço, tanto mental quanto computacional, para tentar invadir a rede e usar a conexão.

Um intruso que consiga entrar em uma rede sem fios ficará com acesso privilegiado para lançar ataques aos elementos constituintes dessa rede, uma vez que é muito mais difícil atacar um sistema a partir de um local remoto da Internet do que lançar esse ataque a partir de dentro da rede. A dificuldade de um ataque remoto aumenta se existirem, entre o atacante e o alvo do ataque, barreiras, como por exemplo, um servidor NAT (*Network Address Translation*). O objetivo de um NAT é esconder do atacante o alvo do ataque e, para isso, o servidor NAT faz a tradução de endereços IPs. Essa tradução consiste na conversão de uma determinada faixa e IPs em uma outra.

Uma vez que um intruso invadiu a rede, ele pode causar os seguintes danos:

- furto ou violação de integridade de informação e serviços da rede;
- abuso de conexão à Internet;
- utilização dessa conexão para prática de atos ilícitos;
- destruição de dados e interferência ao normal funcionamento da rede, etc.

Com o objetivo de tornar uma rede 802.11 mais segura, foram criados alguns protocolos de segurança. O objetivo desses protocolos é dificultar o acesso de usuários não-autorizados à rede sem fio e executar a criptografia dos dados que estão trafegando. Esses protocolos são os relacionados abaixo:

- WEP (*Wired Equivalent Privacy*);
- WPA (*Wi-Fi Protected Access*);
- Padrão IEEE 802.11i ou WPA2 (*Wi-Fi Protected Access*2).

O WEP é um protocolo que faz parte do padrão IEEE 802.11, que, como o próprio nome sugere, tem por objetivo fornecer integridade e confidencialidade equivalentes às das redes com fios. O conceito de integridade diz respeito à não alteração do conteúdo da mensagem enviada; confidencialidade, diz respeito a quem pode ler a mensagem. A integridade dos dados é obtida por meio do uso do CRC-32 (*Cyclic Redundancy Check*-32) e a confidencialidade, por meio da cifra de seqüência RC4 (*Rivest Cipher* 4) [55]. Ambos serão explicados mais adiante.

O protocolo WEP fornece autenticação e criptografia de dados entre um hospedeiro e um AP utilizando uma abordagem de chaves simétricas compartilhadas, não especificando um algoritmo de gerenciamento de chaves [6]. Em um esquema de criptografia de chaves simétricas, a mesma chave é utilizada tanto para cifrar quanto para decifrar os dados transmitidos.

A forma de autenticação utilizada no WEP é a *shared key*. Nesse esquema de autenticação, um segredo é utilizado como semente para o algoritmo de criptografia. Esse segredo compartilhado é a chave secreta. Toda a troca de informações durante o funcionamento normal da rede é realizada por meio da utilização do protocolo WEP, cifrando os dados com essa chave secreta compartilhada. As mensagens trocadas durante a fase de autenticação estão ilustradas na Figura D.1 [56].

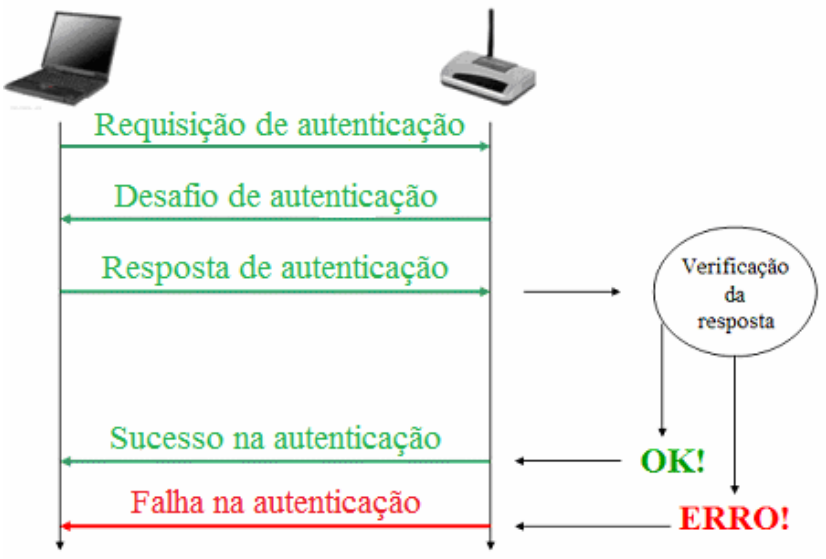


Figura D.1 – *Autenticação no WEP*
(Modificado de [51]).

Com base na Figura D.1, vê-se que o processo de autenticação *shared key* do WEP ocorre em quatro etapas:

- **primeira etapa:** um hospedeiro sem fio requisita autenticação a um AP;
- **segunda etapa:** o AP responde à requisição de autenticação com um quadro de desafio de 128 bits;
- **terceira etapa:** o hospedeiro sem fio cifra esse quadro de desafio usando a chave secreta que compartilha com o AP;
- **quarta etapa:** o ponto de acesso então decifra esse quadro que foi cifrado pela estação móvel, utilizando a mesma chave.

Se o AP conseguir decifrar a reposta enviada pela estação móvel utilizando a chave simétrica compartilhada, e essa resposta corresponder ao quadro inicialmente enviado pelo AP, a autenticação é validada. Caso contrário, o hospedeiro sem fio não é autenticado.

Existem alguns padrões WEP de acordo com o tamanho da chave secreta utilizada. O padrão 64-bits WEP utiliza uma chave secreta de 40 bits (10 caracteres hexadecimais) que é concatenada com um IV (*Initialization Vector*) de 24 bits para formar a chave seqüencial (*Keystream*) RC4. O padrão 128-bits WEP utiliza uma chave de 104 bits de comprimento (26 caracteres hexadecimais) que também é concatenada com um IV de 24 bits de comprimento. Existe também o sistema 256-bits WEP [57].

O algoritmo RC4 funciona como um algoritmo de fluxo, ou seja, é utilizado para enviar um conjunto de bits cifrados em um fluxo contínuo [58]. Nesse tipo de algoritmo, não se pode esperar o acúmulo de um certo número de bits para transmitir e ele é classificado como sendo de chave simétrica.

O RC4 cria bytes pseudo-aleatórios a partir de uma semente formada pela chave secreta e pelo IV. Esses bytes formam a chave de cifragem (chamada de *Keystream*) que será utilizada para cifrar uma mensagem, por meio de operações XOR bit a bit [58]. Ao receber esta mensagem cifrada, o destinatário deve executar o algoritmo da mesma maneira (realizando XOR bit a bit com a mesma chave), recuperando, assim, a mensagem.

A utilização adequada do algoritmo RC4 requer que o mesmo valor de *keystream* de, por exemplo, 64 bits, nunca seja utilizado mais de uma vez [6]. Dado que a chave secreta não muda, ou raramente muda, é o IV que é modificado quadro a quadro. Como o ele é formado por 24 bits, teremos 2^{24} possibilidades diferentes de *keystreams*. E essas possibilidades se esgotam rapidamente com as altas taxas de transmissão disponíveis.

A criação da chave RC4 funciona da seguinte maneira [58]:

- o RC4 recebe uma semente *K* de *n* bits (entre 1 e 2.048). A partir dessa semente, cria um vetor *S* de 256 bytes. Esse vetor tem suas posições permutadas, de acordo com o valor da semente;
- com o vetor formado, o algoritmo utiliza seus dados para criar uma seqüência de números pseudo-aleatórios para cifrar a mensagem. Conforme a mensagem vai sendo enviada, o vetor *S* tem seu conteúdo alterado.

A permutação dos bytes do vetor *S* é realizado pelo algoritmo KSA (*Key Scheduling Algorithm*). O algoritmo PRNG (*Pseudo-Random Number Generation*) é utilizado para gerar os números pseudo-aleatórios.

O protocolo WEP funciona utilizando o gerador de números PRNG do RC4. A semente para geração da chave é uma combinação do segredo compartilhado com um vetor aleatório de 24 bits chamado IV. Para cada quadro, o protocolo WEP deve selecionar um IV diferente, permitindo que a chave secreta permaneça a mesma. Como o destinatário da mensagem deve criar a chave de decifragem a partir da mesma semente, o remetente envia o IV escolhido sem criptografia com o quadro. Dessa forma, o destinatário pode unir o segredo compartilhado com o IV escolhido e utilizar essas informações como semente no PRNG [58].

Para verificar se os dados não foram alterados, ou seja, a integridade deles, é calculado um CRC-32 sobre os dados a serem enviados. O CRC-32 é um código cíclico para detecção de erros. Ele se baseia em um polinômio gerador de grau 32. Existem alguns polinômios geradores padronizados. Um exemplo é o polinômio $G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$, padronizado pelo IEEE 802.3 [57].

O cálculo do CRC-32 segue os seguintes passos:

- **primeiro passo:** multiplica-se o polinômio mensagem, denotado por u(x), por $x^{(n-k)}$, onde (n-k) é o grau do polinômio gerador. A saída dessa etapa é $u(x). x^{(n-k)}$;
- **segundo passo:** dividi-se $u(x). x^{(n-k)}$ por G(x), que é o polinômio gerador;
- **terceiro passo:** concatena-se o resto da divisão do passo anterior à mensagem. Esse resto constitui o CRC.

Para o receptor identificar se houve erros durante a transmissão, ele deve dividir a palavra-código recebida por G(x). Se o resto for igual a zero, então o receptor assume que não ocorreram erros na mensagem.

A Figura D.2 ilustra o processo de cifragem dos dados a serem transmitidos utilizando o WEP.

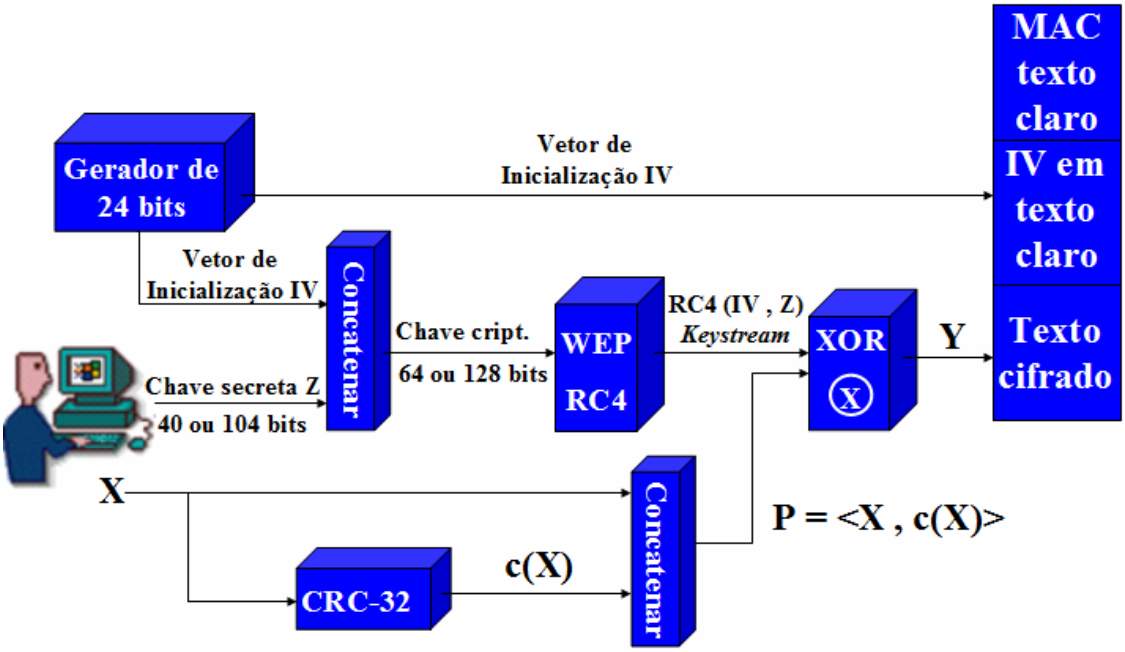


Figura D.2 – *Processo de cifragem no WEP*

(Extraído de [51]).

Primeiramente, é calculado um CRC-32, denotado por c(X), sobre a mensagem a ser enviada. Esse resultado é concatenado ao texto claro X, formando o elemento *P = <X , c(X)>*. Em seguida, escolhe-se um IV. Em função do IV escolhido e da chave secreta Z, o algoritmo RC4 cria uma seqüência de bytes pseudo-aleatório (*keystream*), denotada por RC4 (IV , Z). Finalmente, é feita a cifragem dos dados através de uma operação XOR entre o texto P e a *keystream* RC4, ou seja, gera-se $Y = P \oplus RC4\ (IV , Z)$.

A Figura D.3 mostra todo o processo de decifragem do WEP.

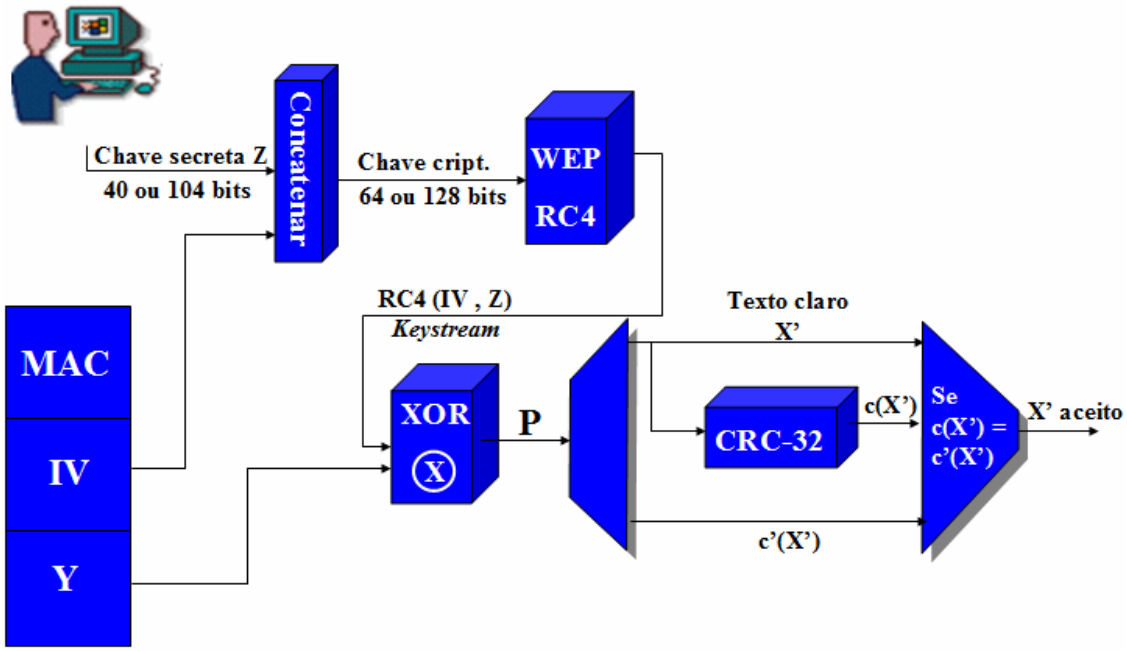


Figura D.3 – *Processo de decifragem no WEP*

(Extraído de [51]).

Por meio do IV recebido e da chave secreta Z, o receptor gera a mesma *keystream* RC4 (IV , Z). Em seguida, efetua-se a operação XOR entre o texto cifrado Y e o RC4 (IV , Z). Após essa operação, obtém-se o elemento P = < X' , c'(X')>, onde X' é uma estimativa do texto claro enviado e c'(X') é o cálculo do CRC-32 sobre a mensagem que foi recebida. Faz-se a separação entre o texto claro recebido X' e o CRC-32 recebido c' (X'). Calcula-se, então, um novo CRC-32 para X', ou seja, c (X'), e compara-se com o CRC-32 recebido c'(X'). Se c(X') for igual a c'(X'), ou seja, se o novo cálculo do CRC-32 sobre a mensagem recebida for igual ao CRC-32 recebido, o texto pleno X' é aceito como válido.

O WEP possui algumas vulnerabilidades com relação a questões de segurança. Uma dessas vulnerabilidades é em relação ao IV. Conforme visto anteriormente, tem-se 2^{24} possibilidades diferentes de IV para serem utilizados na geração da *keystream*. Se esses IVs forem escolhidos aleatoriamente, demonstra-se que a probabilidade de ser escolhido o mesmo valor de IV (e, por conseguinte, a mesma *keystream* de 64 bits) é mais de 99% após apenas 12 mil quadros transmitidos. Com quadros de 100 bytes de tamanho e taxa de transmissão de 11Mbps (padrão 801.11b), bastam apenas alguns segundos para que os 12 mil quadros sejam transmitidos. Visto que o IV é transmitido em texto claro, um bisbilhoteiro saberá sempre que for utilizado um valor de IV duplicado [6].

A seguir são listadas algumas medidas de segurança recomendadas ao se utilizar o protocolo WEP:

- habilite o WEP como um nível mínimo de segurança;
- altere a chave WEP freqüentemente;
- prefira o padrão 128-bits WEP;
- desabilite broadcast de SSID (*Server Set ID*);
- altere o SSID;
- utilize protocolos de tunelamento (IPsec);
- utilize um padrão mais seguro.

Depois que o WEP foi exaustivamente criticado, o IEEE criou o TG1 (*Task Group* 1) para desenvolver um padrão que deveria atender aos objetivos para os quais o WEP fora criado, como também resolver todas as críticas direcionadas a ele.

A *Wi-Fi Alliance* é uma associação sem fins lucrativos, formada em 1999 para certificar os produtos baseados no padrão IEEE 802.11 quanto a sua interoperabilidade. Hoje, há mais de 180 empresas afiliadas, e já certificaram mais de 600 produtos. A *Wi-Fi Alliance*, junto com o IEEE, foram os responsáveis pelo desenvolvimento do WPA na tentativa de resolver as vulnerabilidades do WEP e atender o mercado enquanto o TG1 não lançava um novo padrão [56].

O WPA foi desenvolvido para ser executado nos mesmos *hardwares* que rodavam o WEP. Dessa forma, toda a base de interfaces de rede já instalada, que permite o *upgrade* de *firmware*, foi aproveitada.

O WPA usa o padrão 802.1x para resolver os problemas do WEP no que tange à autenticação. O 802.1X foi desenvolvido para redes cabeadas, mas pode ser aplicado às redes sem fio. O padrão provê controle de acesso baseado em porta e autenticação mútua entre os clientes e os pontos de acesso, através de um servidor de autenticação.

Existem três participantes em uma transação usando o 802.1X [56]:

- **o suplicante**. Um usuário ou um cliente que quer ser autenticado. Ele pode ser qualquer dispositivo sem fio;
- **o servidor de autenticação**. Um sistema de autenticação, tipo RADIUS, que faz a autenticação dos clientes autorizados;
- **o autenticador**. Dispositivo que age como um intermediário na transação entre o suplicante e o servidor de autenticação. É o ponto de acesso, na maioria dos casos.

A Figura D.4 ilustra o processo de autenticação no WPA.

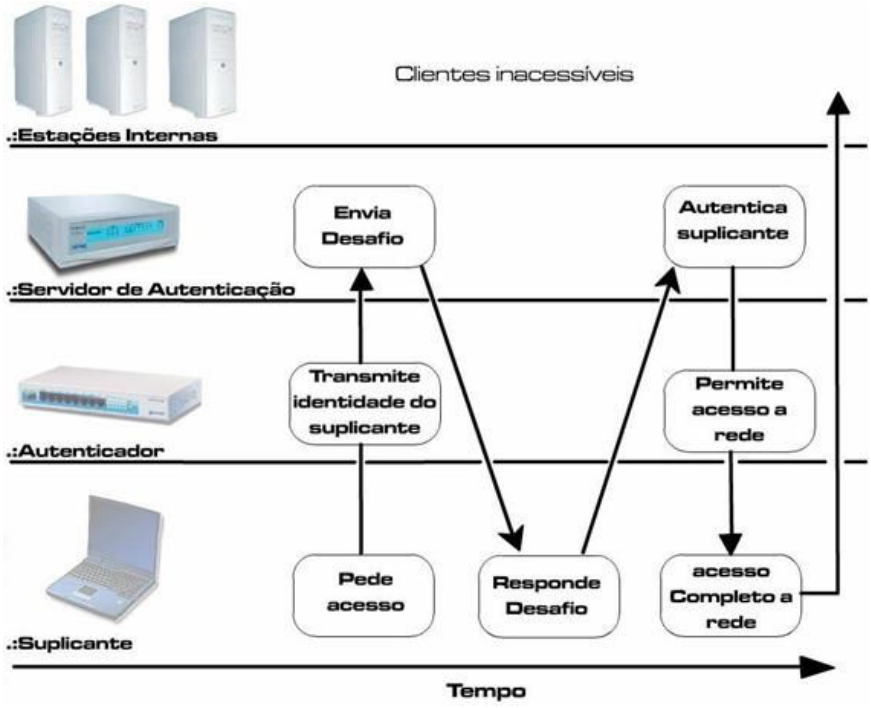


Figura D.4 – *Autenticação no WPA*

(Extraído de [59[]).

A autenticação mútua no 802.1X ocorre segundo os passos descritos abaixo:

- um suplicante inicia uma conexão com o autenticador. O autenticador detecta a inicialização e abre a porta para o suplicante. Contudo, todo o tráfego, exceto o relativo à transação 802.1X, é bloqueado;
- o autenticador pede a identidade ao suplicante;
- o suplicante responde com a sua identidade;
- o autenticador passa a identidade a um servidor de autenticação;
- o servidor de autenticação verifica a identidade do suplicante e envia uma mensagem de *ACCEPT* ao autenticador;
- o autenticador então abre o tráfego ao suplicante;
- o suplicante pede a identidade do servidor de autenticação;
- o servidor de autenticação responde com a sua identidade;
- o suplicante autentica o servidor de autenticação e só então os dados começam a trafegar.

O EAP (*Extensible Authentication Protocol*) é o protocolo que o padrão 802.1X usa para gerenciar a autenticação mútua. O protocolo provê um *framework* generalizado para o sistema de redes sem fio escolher um método específico de autenticação. O método de

autenticação pode ser uma senha, o certificado PKI (*Public Key Infrastructure*), um envio de um desafio ou outro *token* de autenticação. Com um EAP padrão, um autenticador não precisa entender em detalhes o método de autenticação. O autenticador simplesmente age como um intermediário que recebe e repassa pacotes EAP entre o suplicante e o servidor de autenticação, que nesse caso fará a autenticação [56].

Existem vários tipos de protocolos EAP que são utilizados [60[]:

- **EAP-LEAP** (*Lightweight* EAP). Esse padrão foi desenvolvido pela CISCO. O EAP-LEAP usa um par login/senha para transmitir a identidade do suplicante para o servidor RADIUS para autenticação.
- **EAP-TLS** (*Transport Layer Security*). Esse padrão foi descrito na RFC 2716 (*Request For Comments* 2716). O EAP-TLS usa um certificado X.509 para autenticação.
- **EAP-TTLS** (*Tunneled TLS*). Esse é um padrão desenvolvido pela Funk *Software*. O EAP-TTLS é uma opção alternativa ao EAP-TLS. Enquanto o servidor de autenticação se identifica ao cliente com um certificado de servidor, o suplicante usa um par login/senha para se identificar.
- **EAP-PEAP** (*Protect EAP*). Outro padrão desenvolvido para prover autenticação mútua segura. O padrão foi elaborado para superar algumas vulnerabilidades existentes em outros métodos EAP.

A Figura D.5 ilustra as etapas de negociação e autenticação EAP.

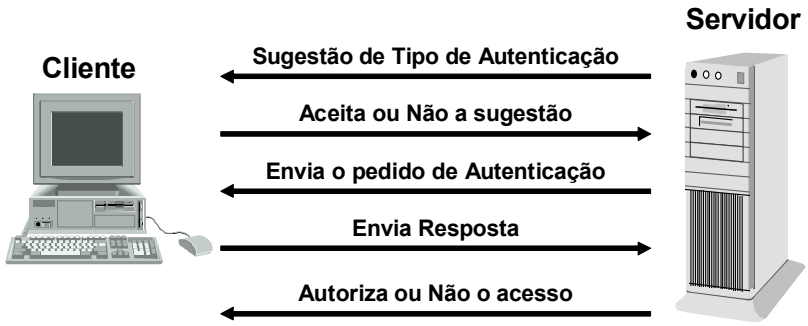


Figura D.5 – *Autenticação EAP*
(Extraído de [61]).

Inicialmente, o servidor envia uma sugestão de autenticação, que pode ser, por exemplo, o uso de um par login/senha (EAP-LEAP). O cliente então aceita ou não essa sugestão. Caso seja aceita, o servidor envia o pedido de autenticação e o cliente devolve uma resposta logo em seguida. Ao receber a resposta do cliente, o servidor autoriza ou não o acesso.

A melhor proposta para trazer segurança ao tão criticado WEP, e que seja de aplicação imediata, é o TKIP (*Temporal Key Integrity Protocol*) [56]. O algoritmo de escalonamento de chaves do TKIP surgiu a partir da idéia proposta ao IEEE por Russ Housley e Doug Whiting, chamada de TKH (*Temporal Key Hash*). TKH é uma função *hash* geradora de chaves para o WEP. A utilização de uma função *hash* para derivar uma outra chave a partir da chave-base foi sugestão de Ron Rivest, que citou como exemplo o MD5 (*Message-Digest algorithm* 5). Entretanto, os autores do TKH preferiram não utilizar o MD5, por ser muito custoso. No lugar disso, eles propuseram um algoritmo muito mais simples e que exige menos processamento. Submeter a chave à função *hash* resolve parte do problema. O resultado da função *hash* deve ser combinado ao resultado da função de integridade para prevenir a alteração e o reenvio de mensagens. O gerenciamento de chaves ainda precisa ser implementado. Em 2002, Niels Ferguson juntou-se à Housley e Whiting, e propuseram um alternativa ao TKH, o ATKH (*Alternate Temporal Key Hash*) [56].

No TKIP, agora conhecido como WPA, o dispositivo começa com uma chave-base secreta de 128 bits, chamada de TK (*Temporal Key*). Essa chave é então combinada com o TA (*Transmitter Address*), ou seja, o endereço MAC do transmissor, criando a chave chamada de TTAK (*Temporal and Transmitter Address Key*). A TTAK é então combinada com o IV para criar as chaves que variam a cada pacote, chamadas de RC4KEY. Cada chave é utilizada pelo RC4 para cifrar somente um pacote.

O TKIP faz com que cada estação da mesma rede utilize uma chave diferente para se comunicar com o ponto de acesso. O problema da colisão de chaves do RC4 é resolvido com a substituição da TK antes que o IV assuma novamente um valor que já assumiu. Isso quer dizer que a cada vez que o IV assumir o seu valor inicial, o TK deve assumir um valor distinto, dado que o endereço MAC do transmissor não muda.

Há um caso especial na implementação do IEEE 802.1X. Em ambientes pequenos (ambientes domésticos ou pequenas empresas), um servidor de autenticação pode não estar disponível. Então uma chave pré-estabelecida é usada. A chave é de conhecimento do suplicante e do autenticador. Uma autenticação, parecida com a que acontece no WEP, então é feita entre esses dois participantes.

Em junho de 2004, a *Wi-Fi Alliance* lança a segunda geração de segurança WPA, conhecida como *Wi-Fi Protected Access 2* ou padrão 802.11i, totalmente compatível com o WPA.

O padrão IEEE 802.11i é um conjunto de padrões e especificações para redes sem fio [626] que apresenta mecanismos de segurança mais robustos, permitindo a implementação de um sistema completo e seguro, mantendo a compatibilidade com sistemas anteriores. Enquanto o WEP oferecia criptografia relativamente fraca, somente um único modo de realizar autenticação e nenhum mecanismo de distribuição de chaves, o IEEE 802.11i fornece formas de criptografia muito mais robustas, um conjunto extenso de mecanismos de autenticação e um sistema de distribuição de chaves [6].

O 802.11i funciona utilizando um sistema de criptografia conhecido por AES (*Advanced Encription Standard*). Esse sistema é mais complexo, fazendo uso de uma arquitetura dos componentes 802.1X para autenticação, RSN (*Robust Security Network*) para acompanhar a associação [63] e CCMP (*Counter Mode with Cipher Block Chaining Message Authentication Code Protocol*) [64] para prover confidencialidade, integridade e autenticidade de origem. O CCMP é um novo modo de operação para uma cifra de bloco que permite que uma única chave seja usada tanto para a cifragem quanto para a decifragem.

Como o WPA, o WPA2 fornece aos usuários empresariais e SOHO (*Small Office and Home Office*) altos níveis de segurança aos seus dados garantindo que apenas usuários autorizados possam ter acesso as suas redes sem fio. Também, da mesma forma que o WPA, o WPA 2 fornece suporte a autenticação 802.1X/EAP [56].

Para resolver o problema do WEP com relação à fraca integridade dos dados, o WPA 2 substitui o cálculo do CRC-32 do WEP pelo algoritmo CCMP do AES. Esse algoritmo foi criado com a finalidade de fornecer uma integridade de dados mais robusta. O algoritmo CCMP calcula um valor de 128 bits, e o WPA 2 usa os 64 bits de maior ordem como um código de integridade da mensagem, realizando, sobre esse código, uma cifragem com a criptografia do modo de contador do AES [65]. Esse código de integridade da mensagem é conhecido como MIC (*Message Integrity Chek*) ou Michael.

Ao contrário do WEP, que usa uma única chave para a criptografia de dados em *unicast* e normalmente uma chave separada para a criptografia de dados em *multicast* e em *broadcast*, o WPA 2 usa um conjunto de quatro chaves diferentes para cada par AP/estação móvel para o tráfego em *unicast* e um conjunto de duas chaves diferentes para o tráfego em *multicast* e *broadcast* [65]. Esse conjunto de quatro chaves é conhecido como chaves temporais emparelhadas.

As quatro chaves emparelhadas para o tráfego de dados em *unicast* são [65]:

- **chave de criptografia de dados**. Consiste em uma chave de 128 bits usada para realizar a cifragem de quadros em *unicast*;
- **chave de integridade de dados**. É uma chave de 128 bits utilizada para calcular o MIC das mensagens em *unicast*;
- **chave de criptografia EAPoL (*Extensible Authentication Protocol over LAN*).** É uma chave de 128 bits usada para a cifrar mensagens da chave EAPoL;
- **chave de integridade EAPoL.** Uma chave de 128 bits usada para calcular o MIC das mensagens da chave EAPoL.

Essas quatro chaves temporais do WPA 2 são derivadas utilizando-se um processo de *handshake* de quatro vias. A Figura D.6 ilustra o processo de geração das chaves temporais (*Temporal Key* – TK):

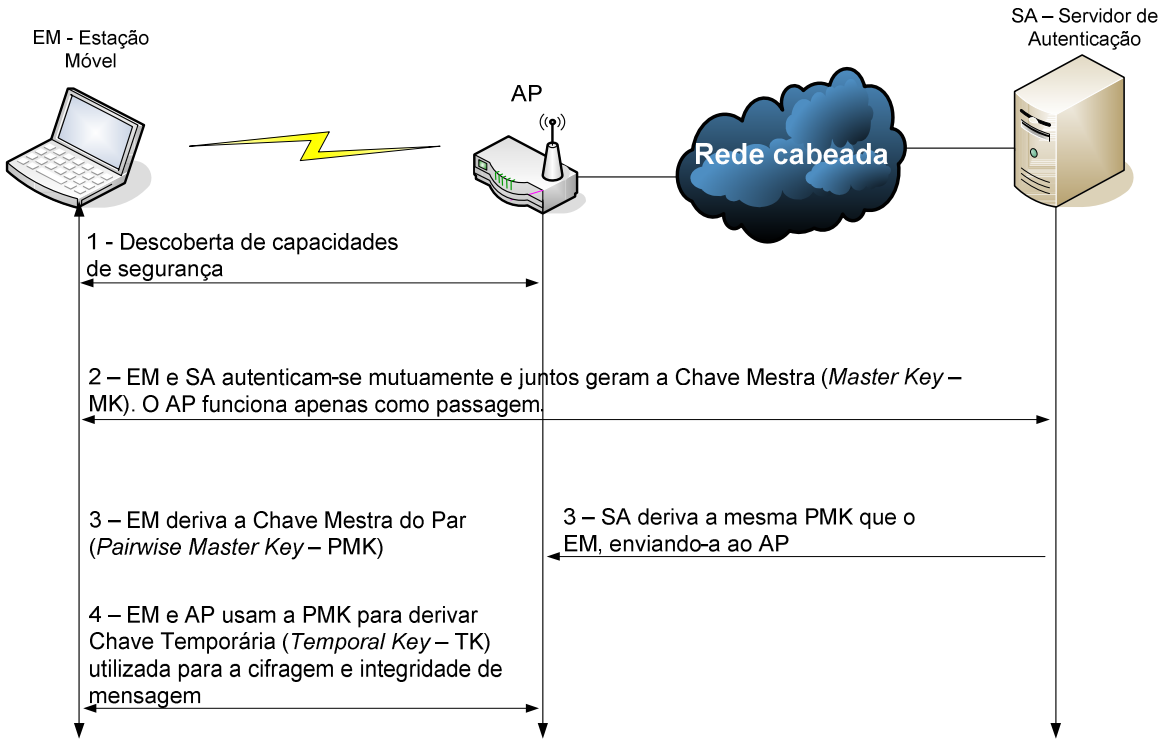


Figura D.6 – *Padrão 802.11i: handshake de quatro vias (Modificado de [6]).*

O funcionamento do *handshake* de quatro vias do padrão 802.11i se baseia nas seguintes fases [6]:

- **Descoberta.** Na fase de descoberta (fase 1 da Figura D.6), o AP anuncia sua presença e todas as opções disponíveis de autenticação e criptografia que podem ser oferecidas a uma estação móvel sem fio. O dispositivo móvel requisita, então, as opções de segurança que deseja. É importante observar que, apesar de haver uma comunicação entre

o cliente e o AP já nessa fase, o primeiro ainda não foi autenticado e não possui uma chave de criptografia.

- **Autenticação mútua e geração da Chave Mestra (*Master Key* – MK).** Essa etapa corresponde à fase 2 da Figura D.6. É nesse momento que ocorre a autenticação entre o cliente sem fio e o servidor de autenticação. O AP age simplesmente como um ponto de passagem, transmitindo mensagens entre o cliente e o servidor. As mensagens fim-a-fim que são trocadas entre o cliente e o servidor são definidas pelo protocolo EAP, já abordado anteriormente. Essas mensagens são interações baseadas em requisições/respostas. As mensagens EAP são encapsuladas usando o EAPoL pelo enlace sem fio entre o cliente e o ponto de acesso. Ao receber as mensagens EAP, o ponto de acesso realiza o desencapsulamento e logo em seguida reencapsula essas mensagens utilizando o protocolo RADIUS para transmissão ao servidor de autenticação por UDP/IP. Com o EAP, o servidor de autenticação pode escolher um dentre os vários modos para realizar autenticação. O esquema de autenticação EAP-TLS é bastante utilizado. O EAP-TLS utiliza técnicas de chaves públicas para permitir que o cliente e o servidor de autenticação se autentiquem mutuamente e também para derivar uma Chave Mestra (*Master Key* – MK) conhecida por ambos os participantes.

- **Geração da Chave Mestra de Par (*Pairwise Master Key* – PMK).** A chave MK é um segredo compartilhado apenas entre o cliente sem fio e o servidor de autenticação, e que eles usam para derivar uma segunda chave: a chave PMK. O cliente já possui sua chave PMK, pois ele a derivou com base em sua chave MK. O AP obtém essa mesma chave PMK através do servidor de autenticação, que a envia para ele (o AP). Essa etapa corresponde à fase 3 da Figura D.6. O cliente e o AP possuem agora uma chave compartilhada e autenticam-se mutuamente.

- **Geração de Chave Temporária (*Temporal Key* – TK).** Com base na chave PMK, tanto o cliente sem fio quanto o AP podem agora gerar chaves adicionais as quais serão utilizadas na comunicação pelo enlace sem fio. A chave TK é de particular interesse, pois ela será utilizada para realizar toda a criptografia na camada de enlace de dados na interface aérea quando um hospedeiro sem fio desejar transmitir informações a outro hospedeiro remoto arbitrário. Essa última etapa corresponde à fase 4 da Figura D.6.

APÊNDICE E – MODULAÇÃO BPSK, QPSK e QAM

O protocolo IEEE 802.11g, para baixas taxas de transmissão, emprega o esquema de modulação BPSK é implementado. Com o uso da modulação BPSK, cada subportadora (sub-canal) é capaz de transmitir a uma taxa de 125 Kbps. Como são utilizadas 48 subportadoras para dados, totaliza-se 6000 Kbps ou 6 Mbps.

Utilizando a modulação QPSK, cada subcanal terá uma capacidade de transmissão igual a 250 Kbps, totalizando 12 Mbps. Para taxas de transmissão iguais ou superiores a 24 Mbps, são utilizadas técnicas de modulação QAM.

A modulação BPSK [31, 50] é a técnica mais simples de modulação PSK, ou seja, de modulação de fase digital. Com a técnica BSPK, duas fases distintas, separadas entre si por 180°, são utilizadas para modular a informação digital. [31] O símbolo binário 1 é representado pelo conjunto de portadoras de fase $\theta(t) = 0$ radianos, e o símbolo binário 0 é representado pelo conjunto de portadoras de fase $\theta(t) = \pi$ radianos. Sendo assim, têm-se:

$$c(t) = \begin{cases} A_c \cos(2\pi f_c t) & \text{para o símbolo binário 1} \\ A_c \cos(2\pi f_c t + \pi) & \text{para o símbolo binário 0.} \end{cases} \quad \text{(E.1)}$$

Na Equação (E.1), $c(t)$ se refere à onda portadora (*carrier*), A_c é a sua amplitude e f_c a freqüência. Lembrando que $\cos(\theta(t) + \pi) = -\cos(\theta(t))$ para todo t, pode-se reescrever a Equação (E.1) como:

$$c(t) = \begin{cases} A_c \cos(2\pi f_c t) & \text{para o símbolo binário 1} \\ -A_c \cos(2\pi f_c t) & \text{para o símbolo binário 0} \end{cases} \quad \text{(E.2)}$$

Considere o caso da modulação de um sinal digital que está na forma de uma seqüência binária de informação. Seja $p(t)$ a forma do pulso básico usado na construção dessa seqüência, conforme mostra a Figura E.1. Seja T o tempo de duração do bit, ou seja, o tempo de duração de um símbolo 0 ou 1. Logo, segundo [31], uma seqüência binária de dados formada por 0s e 1s, pode ser descrita por:

$$m(t) = \sum_k b_k \, p(t - kT) \,, \quad \text{(E.3)}$$

onde:

$$b(k) = \begin{cases} +1 & \text{para o símbolo binário 1} \\ -1 & \text{para o símbolo binário 0.} \end{cases} \quad \text{(E.4)}$$

Por exemplo, para o caso de um pulso retangular, têm-se:

$$p(t) = \begin{cases} +1 & \text{para } 0 \le t \le T \\ 0 & \text{caso contrário.} \end{cases} \tag{E.5}$$

O que é mostrado na Figura E.1.

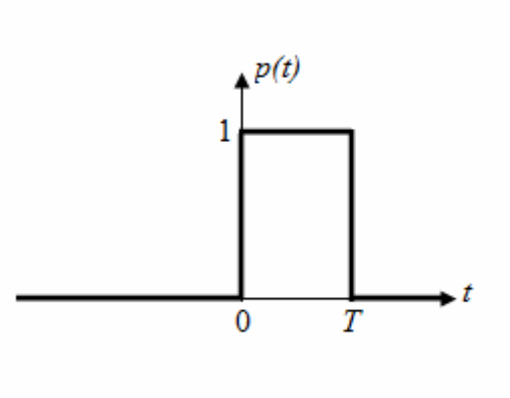


Figura E.1 – *Pulso retangular.*

De acordo com [31], levando em conta as Equações (E.2), (E.4) e (E.5), pode-se expressar o sinal BPSK da seguinte forma:

$$s(t) = c(t)m(t)\,, \tag{E.6}$$

onde *m(t)* é a informação na forma digital definida pela Equação (E.3).

A modulação BPSK também é conhecida por modulação 2-PSK, pois são utilizados dois estados de modulação (duas fases distintas), cada um com apenas 1 bit. Não é importante [66] saber exatamente onde os pontos na constelação estão posicionados, desde que eles estejam separados entre si por 180°. A Figura E.2 ilustra o diagrama de constelação da modulação BPSK.

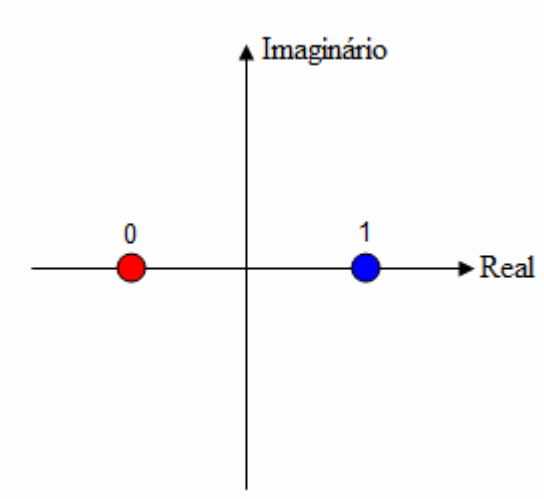


Figura E.2 – *Diagrama de constelação da modulação BPSK.*

No diagrama de constelação da Figura E.2, os pontos BPSK são posicionados no eixo real em 0° e 180°.

Essa técnica de modulação é a forma mais robusta de modulação PSK. Uma grande distorção é necessária para fazer com que o demodulador tome uma decisão incorreta na recepção do sinal. Contudo, essa técnica só é capaz de modular 1 bit por símbolo, como pode ser visto na Figura E.2, não sendo muito adequada para aplicações que requerem altas taxas de transmissão [66].

Também conhecida como quaternária, quadrifase PSK ou 4-PSK, a modulação QPSK utiliza quatro pontos no diagrama de constelação, eqüidistantes em torno de um círculo [66]. A Figura E.3 ilustra o diagrama de constelação QPSK.

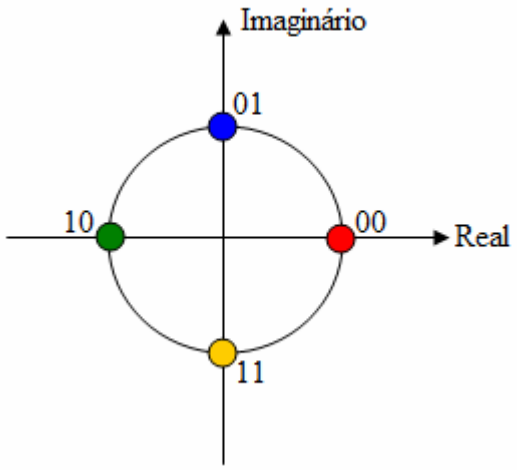


Figura E.3 – *Diagrama de constelação da modulação BPSK.*

Como pode ser visualizado na Figura E.3, com quatro fases, a modulação QPSK é capaz de codificar dois bits por símbolo, dobrando a taxa de transmissão em relação à modulação BPSK. Análises demonstram [66] que a modulação QPSK pode ser utilizada para dobrar a taxa de transmissão BPSK, mantendo a largura de banda ocupada pelo sinal, ou para manter a taxa de transmissão BPSK, porém reduzindo pela metade a largura de banda necessária.

Como sugere o nome, [31] a fase da portadora QPSK assume um dos quatro valores de fase que estão igualmente espaçados, dependendo da composição de cada dibit, ou grupo de dois bits adjacentes na seqüência binária de dados original. Como pode ser observado na Figura E.3, pode-se utilizar 0, 90, 180 e 270 graus como o conjunto de quatro valores disponíveis para o chaveamento de fase da portadora. Outra alternativa seria o conjunto de 30, 120, 210 e 300 graus.

Na técnica QPSK, a seqüência de dados binária original, $m(t)$, é demultiplexada em duas subseqüências $m_1(t)$ e $m_2(t)$. Em particular, a subseqüência $m_1(t)$ chaveia a fase da

portadora entre os valores 0 e 180 graus. A subseqüência $m_2(t)$ chaveia a fase da portadora entre os outros dois valores restantes, ou seja, entre 90 e 270 graus [31].

O diagrama em bloco de um modulador QPSK está ilustrado na Figura E.4.

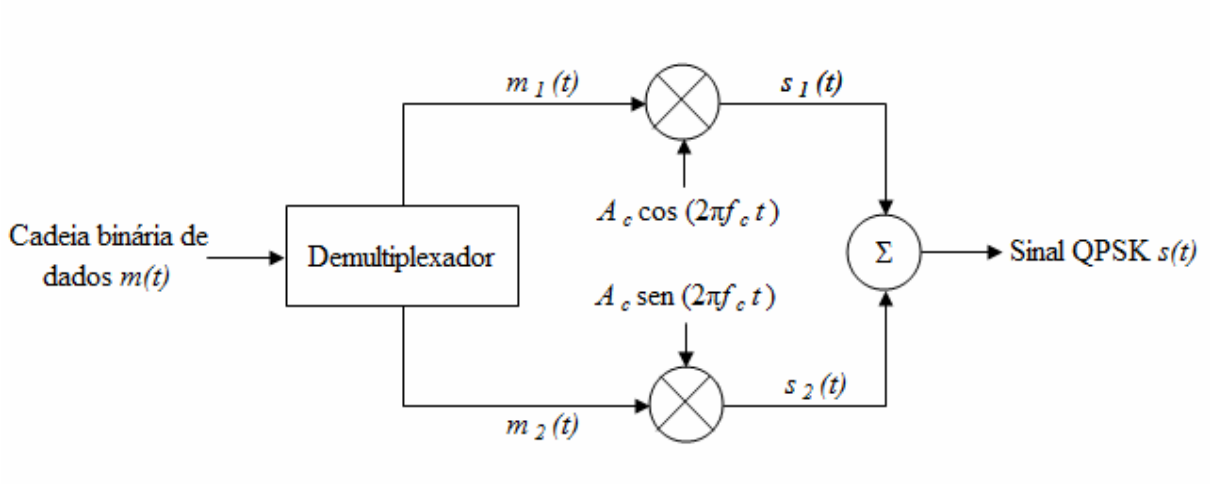


Figura E.4 – *Diagrama em bloco de um gerador QPSK usando um par de portadoras em qudratura. (Extraído de [31])*

Baseando-se na estrutura da Figura E.4, pode-se descrever um modulador QPSK como uma combinação paralela de dois moduladores BPSK que operam em quadratura de fase entre si. Por quadratura de fase, entende-se que o arranjo das fases entre as portadoras é tal que a da portadora no percurso inferior sempre está 90° defasada em relação à da portadora do percurso superior.

Conforme sinalizado anteriormente, $m_1(t)$ e $m_2(t)$ são duas subcadeias binárias que resultaram da demultiplexação da seqüência binária original $m(t)$. Segundo [31], transcrevendo a Equação (E.3) para a presente situação, pode-se descrever $m_1(t)$ e $m_2(t)$ tomando:

$$m_i(t) = \sum_k b_{k,i}\, p(t - kT) \qquad \text{para } i = 1, 2. \tag{E.7}$$

Para $i = 1, 2$ têm-se:

$$b_{k,i} = \begin{cases} +1 & \text{para o símbolo 1} \\ -1 & \text{para o símbolo 0.} \end{cases} \tag{E.8}$$

Para o caso de um pulso retangular,

$$p(t) = \begin{cases} +1 & \text{para } 0 \le t \le 2T \\ 0 & \text{caso contrário.} \end{cases} \tag{E.9}$$

Então, para o sinal BPSK produzido no percurso superior da Figura E.4, têm-se:

$$s_1(t) = c_1(t)m_1(t) \therefore$$
$$s_1(t) = A_c m_1(t)\cos(2\pi f_c t) \quad \text{(E.10)}$$

O sinal BPSK produzido no percurso inferior da Figura E.4 é:

$$s_2(t) = c_2(t)m_2(t) \therefore$$
$$s_2(t) = A_c m_2(t)\sin(2\pi f_c t) \quad \text{(E.11)}$$

Logo, o sinal QPSK na saída do diagrama em bloco da Figura E.4 é igual à soma de $s_1(t)$ e $s_2(t)$:

$$s(t) = s_1(t) + s_2(t) \therefore$$
$$s(t) = A_c m_1(t)\cos(2\pi f_c t) + A_c m_2(t)\sin(2\pi f_c t) \quad \text{(E.12)}$$

A técnica de modulação implementada determina o número de bits por símbolo enviado [13]. Umas das técnicas de modulação mais eficiente é a QAM. Essa técnica combina a modulação PSK, na qual a informação digital é transmitida por meio da variação da fase da portadora analógica, e da modulação ASK, na qual a informação digital é transmitida por meio da variação da amplitude da portadora. Dessa maneia, na modulação QAM tanto a fase quanto a amplitude da portadora variam de acordo com a informação digital a ser transmitida [50].

Na modulação 16-QAM, dispõe-se de dezesseis estados distintos com quatro bits cada, ou seja, [13] cada símbolo transmitido é formado por quatro bits. A Figura E.5 ilustra o diagrama de constelação da modulação 16-QAM.

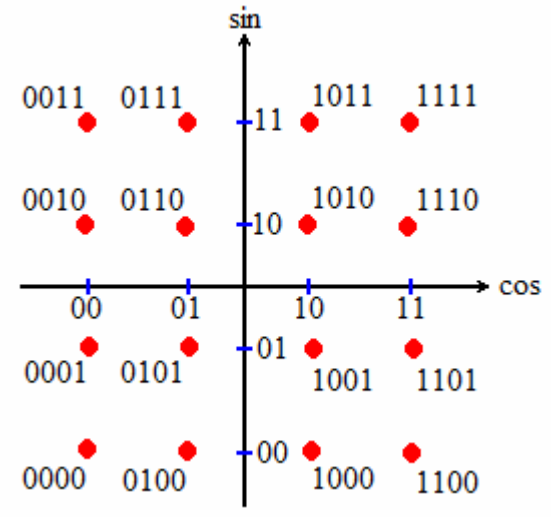


Figura E.5 – *Diagrama de constelação da modulação 16-QAM.*

(Modificado de [53])

Analisando a Figura E.5, constata-se que existem dezesseis estados, doze níveis de fase e três níveis de amplitude. Um esquema de modulação de ordem mais elevada é o 64-

QAM. Nele são disponíveis sessenta e quatro estados com seis bits cada, ou seja, cada símbolo é formado por seis bits. A Figura E.6 ilustra o diagrama de constelação da modulação 64-QAM.

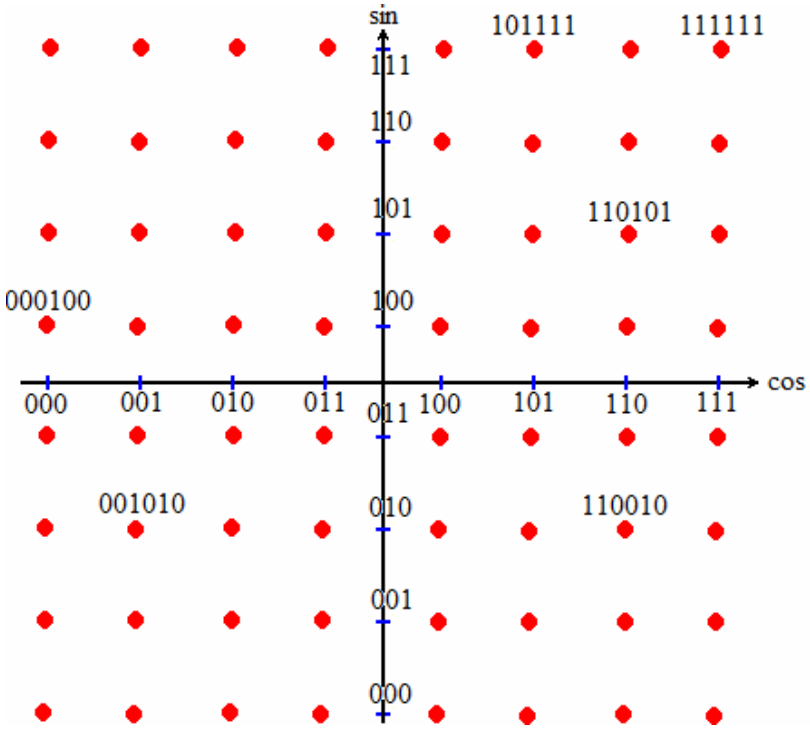


Figura E.6 – *Diagrama de constelação da modulação 64-QAM.*

A Figura E.6 ilustra alguns símbolos na constelação formados por seis bits. Existem ordens mais altas de modulação QAM, como, por exemplo, a modulação 256-QAM.

A modulação QAM fornece taxas de transmissão mais elevadas do que as vistas anteriormente, porém a um custo de ser mais sensível às interferências. Quanto mais elevada for a ordem de modulação, mais sensível às interferências será o sinal, pois os pontos na constelação ficam cada vez mais próximos entre si.