Demóstenes Zegarra Rodríguez

Algoritmo para determinação da taxa de transmissão em uma rede IP

Dissertação apresentada à Escola
Politécnica da Universidade de São
Paulo para obtenção do Título de
Mestre em Engenharia Elétrica.

Demóstenes Zegarra Rodríguez

Algoritmo para determinação da taxa de transmissão em uma rede IP

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Mestre em Engenharia Elétrica.

Área de concentração:
Sistemas Eletrônicos

Orientador:
Prof. Dr. Miguel Arjona Ramírez

Ficha Catalográfica

Rodríguez, Demóstenes Zegarra

Algoritmo para determinação da taxa de transmissão em uma rede IP. São Paulo, 2009. 85 p.

Dissertação (Mestrado) — Escola Politécnica da Universidade de São Paulo. Laboratório de Processamento de Sinais.

1. Redes de computadores. 2. Redes de comunicações. 3. Processamento de voz. 4. Algoritmos. 5. Comunicações digitais. I. Universidade de São Paulo. Escola Politécnica. Laboratório de Processamento de Sinais. II. Título.

aos meus pais Demóstenes e Julia

Agradecimentos

A Deus, por ter me guiado ao longo dessa caminhada.

Aos meus pais, irmãos e sobrinhas pela paciência, compreensão e amor.

Ao meu orientador, Prof. Dr. Miguel Arjona Ramírez, pelo grande aprendizado tanto no acadêmico quanto no pessoal.

A minha namorada Renata por seu apoio e colaboração.

A todos os demais amigos que me apoiaram e ajudaram na realização deste trabalho.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo apoio financeiro para a realização deste trabalho.

Resumo

A comunicação de Voz sobre IP (VoIP) vem aumentando com o decorrer do tempo e as redes de comunicação estão se tornando cada vez mais congestionadas, ocasionando perda de pacotes e latência nas redes, prejudicando diretamente a qualidade das comunicações de voz.

Neste trabalho, foi estudado em detalhe como a qualidade do sinal de voz transmitido em uma comunicação VoIP é afetada por parâmetros da rede e pelo tipo de codificador utilizado na comunicação. Uma contribuição importante deste trabalho é a apresentação de uma metodologia que serve para prever o comportamento de um codificador de voz em diferentes cenários de redes.

O estabelecimento de um mecanismo de controle que otimize a utilização da rede e ao mesmo tempo garanta a melhor qualidade possível do sinal de voz transmitido vem sendo motivo de pesquisa.

O emprego de codificadores multitaxa nas comunicações de voz possibilita a implementação de algoritmos que controlem a comutação destas taxas de codificação baseados em diferentes fatores de decisão como as características do sinal de voz a ser transmitido ou empregando a informação dos parâmetros de rede.

Este trabalho apresenta um algoritmo de determinação de taxa de codificação com fator de decisão baseado na qualidade do sinal avaliada no ponto da recepção ou em algum ponto intermediário.

Para realização dos testes montou-se um cenário de emulação de rede IP para o estabelecimento de uma chamada VoIP, onde são utilizados codificadores multitaxa (ITU-T G.726 e Speex). Para avaliação da qualidade do sinal de voz foi utilizada a recomendação ITU-T P.563, sendo que o índice MOS obtido no ponto de recepção é transmitido utilizando um laço de transmissão, que forma parte da estrutura do mecanismo de controle apresentado neste trabalho. Este laço de transmissão é implementado via *socket* sobre uma comunicação UDP.

Como as ferramentas utilizadas são todas *freeware*, o cenário de emulação pode ser facilmente implementado por demais pesquisadores.

Os resultados obtidos são confiáveis, já que para cenários onde se mesclam diferentes taxas de codificação, o índice MOS obtido é um valor que está na faixa dos valores obtidos nos cenários onde se utilizou taxa de codificação única.

O correto desempenho do mecanismo do RDA também foi verificado, sendo que a qualidade do sinal de voz decresce, o mecanismo de controle no RDA comuta a uma taxa de codificação menor, garantindo o melhor uso do canal de transmissão. Quando os parâmetros da rede passam a melhores condições, o índice MOS do sinal aumenta e o RDA comuta a uma taxa de codificação maior obtendo uma qualidade mais alta na comunicação.

Abstract

Voice over IP (VoIP) communications are becoming increasingly popular so that data communication networks are ever more prone to degradations like packet losses and rising latency.

In this work, it was studied in detail how the quality of the speech signal transmitted in a VoIP communication is affected by parameters of the network and the type of codec used in the communication. An important contribution of this work is to predict the behavior of a speech codec in different network scenarios.

The main goal of this research has been the development of an algorithm that constrains rate allocation to a speech communication for best quality according to current network resource availability.

The use of multirate codecs in speech communications makes it possible to implement control algorithms for coding rate switching. They are based on speech signal features or network traffic parameters.

This work presents an algorithm for the determination of coding rate with decision factor based on speech quality evaluated at the point of reception or intermediate points.

For accomplishment of the tests, a scenario of emulation of an IP network for the establishment of a VoIP call was built, where multi-rate codecs are used (ITU-T G.726 and Speex). For evaluation of the quality of the speech signal the Recommendation ITU-T P.563 was used, where the MOS index obtained in the reception is transmitted using a transmission loop, as a part in the control mechanism in the proposed algorithm. This transmission loop consists of a UDP message sent on a socket.

As the tools used are all freeware, the simulation scenarios can easily be implemented by other researchers.

The results are trustworthy, because for multirate scenarios, the MOS index obtained is a value that is between the ones obtained in the single-rate scenarios.

The correct performance of the mechanism of the RDA was also verified, so that when the quality of the signal decreases, the control mechanism in the RDA switches to a lower coding rate, ensuring the best use of the transmission canal. When the parameters of the network drive to best conditions, the MOS index of the signal increases and the RDA decides to switch to a higher coding rate achieving a higher communication quality.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

1	Introdução	1
1.1	Contexto	1
1.2	Objetivo	3
1.3	Organização do Trabalho	3
2	Revisão Teórica	4
2.1	Codificadores de Voz	4
2.1.1	Técnicas de codificação	4
2.1.1.1	Codificadores da forma de onda	5
2.1.1.2	Codificadores paramétricos	5
2.1.1.3	Codificadores híbridos	5
2.1.2	Codificadores segundo as Recomendações ITU-T	5
2.1.2.1	Codificador segundo a Recomendação ITU-T G.711	6
2.1.2.2	Codificador segundo a Recomendação ITU-T G.726	7
2.1.2.3	Codificador segundo a Recomendação ITU-T G.728	7
2.1.2.4	Codificador segundo a Recomendação ITU-T G.729	8
2.1.2.5	Codificador segundo a Recomendação ITU-T G.723.1	8
2.1.3	Codificador <i>internet Low Bitrate Codec</i> (iLBC)	8
2.1.4	Codificador Speex	9

2.1.5	Parâmetros dos Codificadores de Voz	9
2.2	Estimadores de Qualidade de Voz	10
2.2.1	Classificação dos estimadores de qualidade de voz	10
2.2.1.1	Métodos não Intrusivos	10
2.2.1.2	Métodos Intrusivos	10
2.2.1.3	Recomendação ITU-T P.800	11
2.2.1.4	Recomendação ITU-T P.862	14
2.2.1.5	Recomendação ITU-T P.563	17
2.2.1.6	Recomendação ITU-T G.107	18
2.3	Descrição de uma rede VoIP	21
2.3.1	Arquitetura de uma rede VoIP	22
2.3.2	Informação de um pacote VoIP	22
2.3.3	Protocolos e Recomendações em uma rede VoIP	25
2.4	Principais fatores de degradação da qualidade em uma rede de dados	29
2.5	Taxa de Codificação e Taxa de Transmissão	30
2.5.1	Composição da taxa de transmissão de codificadores	30
2.5.2	Mecanismos para reduzir a taxa de transmissão	31
2.6	Algoritmo para Determinação da taxa de codificação (RDA)	32
2.6.1	Métodos existentes de RDA	33
2.6.1.1	RDA baseado na análise do sinal	33
2.6.1.2	RDA baseado em RTCP	35
2.6.2	Mecanismos de melhora da qualidade de uma rede IP	36
2.6.2.1	<i>Playout Buffer</i>	37
2.6.2.2	FEC - <i>forward error correction</i>	37
3	Cenários de Emulação e Resultados Experimentais	39
3.1	Ferramentas empregadas no cenário de emulação	39
3.2	Metodologia para predizer o comportamento de um codificador de VOZ	42

3.2.1	Procedimento de teste	43
3.2.2	Resultados dos testes da estimação da qualidade da voz . .	44
3.2.2.1	Cenário ideal	44
3.2.2.2	Cenário com atrasos fixos ponto a ponto	45
3.2.2.3	Cenário com probabilidade de perdas de pacotes .	48
3.3	Avaliação de um RDA baseado na qualidade do sinal	49
3.3.1	Cenário de emulação	50
3.3.2	Resultados obtidos na avaliação do RDA	54
3.3.3	Avaliação do RDA em uma rede com atrasos ponto a ponto e perda de pacotes	56
3.4	Implementação do laço de transmissão	58
3.4.1	Socket	59
3.4.1.1	Comunicação entre cliente e servidor	60
4	Conclusão	63
4.1	Principais contribuições	64
4.2	Sugestões para trabalhos futuros	65
	Referências	66
	Apêndice A – Códigos utilizados nesta dissertação	71
	Código adicional para P.563.c	71
	Código adicional para o gravador de som	72
	Código adicional do NISTNet	72
	Mecanismo de controle utilizando RDA	74
	client_socket_udp.c	74
	loop_time2.c	76
	server_socket_udp.c	78
	Variador de taxa no softphone	81

Lista de Figuras

2.1	Diferença entre métodos intrusivos e não intrusivo	11
2.2	Algoritmo utilizado pela PESQ	16
2.3	Relação entre o MOS e o parâmetro R	21
2.4	Elementos de uma Rede IP para comunicações de voz	22
2.5	Formato do pacote de voz transmitida	23
2.6	Formato do quadro Ethernet	24
2.7	Formato do quadro Frame Relay	24
2.8	Seqüência de mensagens de sinalização SIP	27
2.9	Protocolos da Recomendação H.323	28
3.1	Configuração do cenário básico de emulação	41
3.2	Endereços IP configurados na rede emulada	42
3.3	Cenário de rede utilizado para a determinação do índice MOS utilizando a ferramenta PESQ	43
3.4	Relação MOS-LQO vs. Atraso fim a fim para codificador G.723	45
3.5	Procedimento para determinação do fator de degradação do codificador	46
3.6	Determinação do índice I_e	47
3.7	Relação MOS vs. Atraso ponto a ponto em uma comunicação bidirecional	48
3.8	Cenário de emulação para avaliação do RDA	50
3.9	Cenário simplificado de emulação para avaliação do RDA	53
3.10	Valores MOS-LQO para diferentes períodos entre comutações da taxa de codificação	56
3.11	Cenário de emulação	57

3.12	Alocação do socket do RDA entre as camadas OSI	60
3.13	Mensagem enviada do cliente ao servidor por socket	61
3.14	Funcionamento do laço de transmissão	62
B.1	Campos do pacote SR	83

Lista de Tabelas

2.1	Resumo dos valores dos parâmetros: Taxa de codificação, tempo de antecipação dos codificadores de voz G.711, G.726, G.727, G.728, G.723.1, G.729 e iLBC	9
2.2	Pontuação para MOS Conversacional e para testes de audição . .	12
2.3	Pontuação MOS para testes de esforço de audição	13
2.4	Pontuação MOS para testes de preferência de audição	14
2.5	Cenários de teste significativos para o algoritmo PESQ	16
2.6	Cenários de teste impróprios para o algoritmo PESQ	17
2.7	Cenários de teste para os que PESQ não foi avaliado	17
2.8	Composição da Taxa de Transmissão para Codificadores G.711 e G.729	24
2.9	Taxa de transmissão nominal de codificadores	31
2.10	Taxa de transmissão nominal de codificadores com VAD e cRTP .	32
2.11	Classificação de um segmento de sinal	35
2.12	Decisão sobre taxa de codificação final	35
2.13	Exemplo da funcionalidade do FEC	38
3.1	Índices MOS-LQO médios para codificadores em um cenário ideal	45
3.2	Valores médios MOS-LQO e I_e para os codificadores G.723.1 e iLBC	47
3.3	Valores MOS-LQO para os codificadores G.723.1 e iLBC em um cenário com probabilidade de perda de pacotes	48
3.4	Exemplo de predição do comportamento de codificador em diferentes cenários de rede	49
3.5	Comportamento do RDA em um cenário com perda de pacotes - codificador G.726	52
3.6	Exemplo de relatório de qualidade da rede.	53

3.7	Valores MOS-LQO com taxas únicas para os codificadores G.726 e Speex	54
3.8	Valores MOS-LQO obtidos para cenários de 2 e 3 taxas do codificador G.726 (16, 24, 32 e 40 kbit/s)	55
3.9	Valores MOS-LQO obtidos para cenários de 2 e 3 taxas do codificador <i>Speex</i> (5, 8, 15 e 18,2 kbit/s)	55
3.10	Comportamento do RDA em um cenário com perda de pacotes e atraso para o codificador G.726.	57
3.11	Otimização da largura de banda com o emprego do RDA.	58

Lista de Abreviaturas

ACR *Absolute Category Rating*

ACELP *Algebraic Code Excited Linear Prediction*

CDMA-EVRC *Code Division Multiple Access- Enhanced Variable Rate Codec*

CELP *Code Excited Linear Prediction*

cRTP *Compressed Real Time Protocol*

CS-ACELP *Conjugate-Structure Algebraic-Code-Excited Linear-Prediction*

CNG *Comfort Noise Generation*

CoS *Class of Service*

DCR *Degradation Category Rating*

GSM-AMR *Global System for Mobile Communications - adaptive multi rate*

GSM-EFR *Global System for Mobile Communications - enhanced full rate*

GSM-FR *Global System for Mobile Communications - full rate*

GSM-HR *Global System for Mobile Communications - half rate*

HTML *Hypertext Markup Language*

IETF *Internet Engineering Task Force*

iLBC *internet Low Bitrate Codec*

IMBE *Improved Multiband Excitation*

IP *Internet Protocol*

ITU-T *International Telecommunication Union - Telecommunication Standardization Sector*

LAN *Local Area Network*

LD-CELP *Low-Delay Code Excited Linear Prediction*

LPAS *Linear Prediction Analysis-by-Synthesis*

MOS *Mean Opinion Score*

MOS_c *Mean Opinion Score of conversation*

MOS_{le} *Mean Opinion Score listening effort*

MOS_{lp} *Mean Opinion Score listening predilection*

MOS-LQO *Mean Opinion Score listening quality objective*

MOS-LQS *Mean Opinion Score listening quality subjective*

MP-MLQ *Multi-Pulse Maximum Likelihood Quantization*

NAK *Not Acknowledged*

NIST *National Institute of Standards and Technology*

OSI *Open Systems Interconnection*

PCM *Pulse Code Modulation*

PDT *Packet delay transfer*

PDV *Packet delay Variation*

PESQ *Perceptual Evaluation of Speech Quality*

PLC *Packet loss concealment*

PLR *Packet loss rate*

QoS *Quality of Service*

RDA *Rate Determination Algorithm*

RTP *Real Time Protocol*

RTCP *Real Time Control Protocol*

SIP *Session Initiation Protocol*

TCP *Transmission Control Protocol*

TDMA VSELP *Time Division Multiple Access - Vector Sum Excited Linear Prediction*

TDMA ACELP *Time Division Multiple Access -Algebraic Code Excited Linear Prediction*

SMTP *Simple Mail Transfer Protocol*

UDP *User Datagram Protocol*

VAD *Voice Activity Detection*

VoIP *Voice over Internet Protocol*

WAN *Wide Area Network*

1 Introdução

Neste capítulo daremos destaque para os tópicos que mostram a importância deste trabalho, juntamente com a organização estrutural deste.

1.1 Contexto

O serviço de tráfego de dados cresceu bastante nas últimas décadas e junto com seu crescimento, também vieram novas oportunidades de associação com outros tipos de tráfegos, voz e imagem, por exemplo.

Em uma rede de dados é possível fazer com que a voz digitalizada trafegue conjuntamente com os dados de outras aplicações na mesma rede. Portanto, desde que isso se tornou tecnicamente possível, o tráfego de voz começou a ser transportado pelas redes de dados (pacote) e não somente pelas redes comutadas por circuitos. Com a convergência entre dados e voz, as tecnologias de comunicação via *Internet Protocol* (IP) vêm se tornando cada vez mais viáveis. A partir de então o tráfego de voz transmitido por uma rede IP passou a ser denominado de *Voice over Internet Protocol* (VoIP).

Os operadores de telecomunicações têm integrado os serviços de voz na rede baseada na comutação de pacotes, obtendo os seguintes benefícios: unificação da tecnologia para a rede de transporte, unificação do modo de operação com o uso de um menor número de equipamentos a monitorar e otimização dos recursos de transmissão, acarretando uma redução de custos.

Alguns parâmetros devem ser analisados em uma rede de pacotes, a fim de manter um controle e saber como e o quanto podem afetar o fator qualidade de voz, percebida pelo ouvinte em uma comunicação telefônica. Tais parâmetros da rede são: perda de pacotes, atrasos fixos e variáveis, características do codificador e o eco.

A melhora da qualidade de transmissão de voz através de uma rede IP pode ser conseguida também, por soluções que abrangem os *Rate Determination Algorithm*

(RDA) baseados em diferentes tipos de informação: as características intrínsecas do sinal ou as características externas da rede. Os artigos [1], [2], [3] e [4] são baseados na análise do sinal de voz, através da detecção de atividade de voz (*Voice Activity Detection* (VAD)) e algoritmos que determinam o grau de sonoridade do sinal.

Este trabalho descreve um eficiente mecanismo de controle de qualidade baseado na estimação da qualidade da voz, avaliada em um ponto da comunicação. Os resultados mostram uma melhora na utilização do canal de transmissão empregando taxas de codificação menores quando a qualidade do sinal transmitido tem uma pontuação menor, devido a algumas perdas de pacotes ou atrasos na rede, permitindo que um número maior de usuários possa ter acesso à rede. Quando a qualidade do sinal alcança valores de qualidade comparáveis aos obtidos em um cenário de rede ideal, o RDA comuta à taxa de codificação mais alta garantindo a maior qualidade possível da comunicação VoIP.

Neste trabalho, é apresentado o cenário de emulação de um sistema de transmissão de voz sobre uma rede IP considerando diferentes elementos que intervêm neste tipo de rede, os quais serão descritos no decorrer deste trabalho, assim como seu comportamento e propostas de melhora na qualidade de uma comunicação VoIP.

Em termos globais este trabalho pode ser dividido em duas contribuições que estão relacionadas entre si de uma forma seqüencial:

- Uma metodologia para prever o comportamento do codificador, em estudo, em diferentes cenários de rede: de comunicações bidirecionais com probabilidades de perda de pacotes e atrasos. Ela permite para cada cenário de rede emulado, obter um índice de qualidade para o codificador, apresentando para isto, o procedimento completo para prever o comportamento do codificador, em termos da qualidade do sinal, em diferentes cenários de rede.

É importante mencionar que o ambiente experimental de testes utilizado neste trabalho pode ser reproduzido facilmente por ser composto de ferramentas *open sources*.

- Um mecanismo de controle, que permita fazer a variação da taxa de codificação em um codificador multitaxa, tendo como parâmetro de decisão a qualidade do sinal transmitido no ponto da recepção.

Em todos os casos, obtiveram-se resultados coerentes com o esperado, de

acordo com a teoria, e/ou com os trabalhos semelhantes.

1.2 Objetivo

Este trabalho tem por objetivo apresentar um mecanismo de controle em um sistema de comunicação de voz sobre IP baseado na qualidade do sinal de voz no ponto de recepção que otimiza os recursos da rede. Para este propósito, tem-se que ter um conhecimento dos parâmetros da rede (perda de pacotes e atrasos) e do codificador empregado na comunicação, que conjuntamente determinarão a qualidade do sinal de voz. Esta análise tem como objetivo adicional a apresentação de uma metodologia para prever o comportamento de um codificador em diferentes condições de rede.

1.3 Organização do Trabalho

Este capítulo mostra a parte introdutória, apresentando uma visão geral deste trabalho, a importância e o contexto em que ele está inserido, destacando o seu principal objetivo.

O Capítulo 2 apresenta uma Revisão Teórica, útil para a melhor compreensão do trabalho, que está dividida em: codificadores de voz, estimadores da qualidade de voz, estrutura de uma rede IP, principais fatores de degradação da qualidade, métodos existentes para elaboração do RDA e laço de transmissão.

O capítulo 3 apresenta o cenário de emulação deste trabalho, os testes efetuados e resultados obtidos a partir deste cenário, assim como uma explicação sobre os dois principais focos do trabalho: metodologia para prever o comportamento de um codificador em diferentes cenários de rede e o mecanismo de funcionamento do RDA.

O capítulo 4 apresenta as Conclusões relacionadas ao cenário, os testes efetuados e Trabalhos Futuros.

Por fim, o Apêndice A apresenta os códigos utilizados: P.563, gravador de som, NISTNet, mecanismo de controle com RDA, laço de transmissão e *softphone*. E o Apêndice B apresenta o detalhamento dos Pacotes SR e RR do *Real Time Control Protocol* (RTCP).

2 Revisão Teórica

Neste capítulo serão abordados os tipos de codificadores de voz, técnicas de codificação, os estimadores de qualidade, protocolos e recomendações em uma rede VoIP, os principais fatores de degradação de qualidade de voz em uma rede de dados, e métodos existentes na implementação de um RDA.

2.1 Codificadores de Voz

Nesta seção serão descritas as principais técnicas de codificação de voz, apresentando como estas causam influência principalmente na taxa de transmissão e nos atrasos, devido à complexidade dos algoritmos implementados. Considera-se a qualidade do sinal no receptor que se pode alcançar com cada uma destas técnicas.

A transmissão em sistemas telefônicos digitais, seja em uma rede fixa, celular, pela Internet ou em redes híbridas, tem muitos custos associados, onde um dos mais importantes é o meio de transmissão. Por isso, procura-se transmitir a maior quantidade possível de dados, usando uma boa técnica de compressão [5], sempre que não haja uma perda significativa na qualidade do sinal recebido.

A qualidade do sinal transmitido será aceitável, dependendo de muitos fatores da rede, principalmente da perda de pacotes e atrasos, e do tipo de serviço que se está atendendo como: voz ou dados.

2.1.1 Técnicas de codificação

Ao longo do tempo vêm-se implementando diversas técnicas de codificação [6] e [7] que têm como principal objetivo melhorar a relação entre a taxa de transmissão e a qualidade oferecida. Assim, os algoritmos de codificação de voz podem ser classificados de uma forma geral [8] ou seguindo uma perspectiva histórica em:

2.1.1.1 Codificadores da forma de onda

Estes codificadores são de baixa complexidade e de atraso menor, caracterizados por seguirem o sinal amostra-a-amostra, obtendo-se altas qualidades do sinal, mas utilizam uma largura de banda maior.

2.1.1.2 Codificadores paramétricos

Estes codificadores descrevem o sinal de fala de um modo paramétrico, conseguindo uma diminuição da taxa de transmissão, em geral às custas de um decréscimo da qualidade dos sinais sintetizados, do aumento do atraso e da complexidade computacional.

Comparando estas duas primeiras técnicas de codificação, tem-se que os codificadores da forma de onda transmitem as amostras do sinal, conseguindo uma qualidade muito boa, devido à ausência de um modelo matemático para a compressão do sinal, mas se têm taxas de transmissão muito altas. Em contrapartida, os codificadores paramétricos realizam uma modelagem bastante complexa do sinal de voz e transmitem só os parâmetros resultantes da modelagem empregada. Por tal motivo, conseguem taxas de transmissão menores, sendo que para estas taxas o índice de qualidade é menor em relação aos obtidos pelas altas taxas dos codificadores paramétricos.

2.1.1.3 Codificadores híbridos

Desenvolvidos na década de 1980, destacam-se dois tipos de técnicas: a codificação por análise mediante síntese baseada em predição linear *Linear Prediction Analysis-by-Synthesis* (LPAS) e a codificação senoidal. Esta última técnica empregada em codificadores que utilizam excitação por bandas de frequência, chamada *Improved Multiband Excitation* (IMBE) proposto em [9], [10] e utilizando no sistema INMARSAT [11]. Estes dois tipos de codificadores aproveitam as vantagens dos codificadores paramétricos e dos codificadores de forma de onda.

2.1.2 Codificadores segundo as Recomendações ITU-T

Abaixo, descrevem-se brevemente os codificadores de banda estreita mais importantes para este trabalho, ressaltando suas principais características segundo as normas publicadas pela *International Telecommunication Union - Telecommunication Standardization Sector* (ITU-T). Toda a informação descrita

nesta seção está baseada no site www.itu.int/rec que disponibiliza tanto a documentação como os softwares referentes às recomendações ITU-T.

2.1.2.1 Codificador segundo a Recomendação ITU-T G.711

Aprovação: 1972 (Genebra, modificada posteriormente).

Apêndice I: Algoritmo de baixa complexidade e alta qualidade para a dissimulação de perda de pacotes, segundo a Recomendação ITU-T G.711 - Setembro de 1999 [12].

Apêndice II: Definição da utilização do ruído de conforto, segundo a Recomendação ITU-T G.711 nos sistemas de comunicação multimídia por pacotes - Fevereiro de 2000.

Frequência de amostragem: O valor nominal recomendado é de 8000 amostras por segundo, com uma tolerância de ± 50 ppm.

Taxa de codificação: 64 kbit/s para transmissão do sinal digitalizado, já que são geradas 8.000 amostras de 8 bit/s.

Técnica de Codificação: *Pulse Code Modulation* (PCM).

Características de Codificação: Em um processo de amostragem, um sinal de voz contínuo no tempo é transformado em um sinal discreto no tempo.

O sinal de saída, ou seja, o sinal discreto no tempo é obtido por um processo de amostragem, tomando-se amostras do sinal em intervalos uniformes. No decodificador, para que o sinal original possa ser recuperado a partir do sinal amostrado, é preciso que a frequência de amostragem utilizada no codificador obedeça ao critério de *Nyquist*, isto é, seja maior ou igual ao dobro da maior frequência presente no sinal. Desta forma, para um sinal com faixa de 0 a f_{max} , a frequência de amostragem deve ser pelo menos:

$$f_a = 2 \times f_{max} \quad (2.1)$$

Com o advento das centrais com programa armazenado, as redes telefônicas tradicionais adotaram uma frequência de amostragem de 8.000 amostras do sinal por segundo. Em seguida, cada amostra é representada com oito bits, representando um nível de quantização dentre os 256 possíveis.

O processo de quantização é a atribuição de um valor discreto contável aos

níveis de amplitude do sinal contínuo amostrado. Esse processo gera um erro ou ruído de quantização, que consiste na diferença entre o sinal na entrada do quantizador e o sinal quantizado na saída.

Segundo a norma ITU-T G.711, se recomenda duas leis de quantização: A e μ .

2.1.2.2 Codificador segundo a Recomendação ITU-T G.726

Data de Aprovação: 14 de Dezembro de 1990 [13].

Taxa de codificação: 16 - 24 - 32 - 40 kbit/s.

Tipo de Codificação: ADPCM (Adaptive Differential Pulse Code Modulation).

Atraso: O atraso desta Recomendação é de 0,125 ms.

O algoritmo original ADPCM (Adaptive Diferencial PCM) que foi padronizado pela ITU-T em 1984 incluía unicamente a taxa de codificação de 32 kbit/s e tinha por objetivo codificar a voz na metade da taxa utilizada pelo codificador G.711, mantendo a mesma qualidade do sinal. Este algoritmo está descrito no livro vermelho (Red Book) como a recomendação G.721 [15]. Em 1988, no livro azul (Blue Book), foi publicada a recomendação G.723 [16], que além de incluir a taxa de 32 kbit/s, incluía as taxas de 40 e 24 kbit/s. Em 1990 as Recomendações G.721 e G.723 foram unidas e foi adicionada uma taxa menor igual a 16 kbit/s; esta nova recomendação foi denominada G.726 [10]. Estas quatro taxas são geralmente referidas pelo número de bits usados por amostra, que são: 2 bits (16 kbit/s), 3 bits (24 kbit/s), 4 bits (32 kbit/s) e 5 bits (40 kbit/s). Este codificador multitaxa será utilizado na avaliação do algoritmo de determinação de taxa de codificação proposto neste trabalho.

2.1.2.3 Codificador segundo a Recomendação ITU-T G.728

Data de Aprovação: 1 de Setembro de 1992 [14].

Taxa de codificação: 16 kbit/s.

Tipo de Codificação: *Low-Delay Code Excited Linear Prediction* (LD-CELP).

Atraso: Como o codificador trabalha com blocos de cinco amostras PCM e cada amostra acarreta um atraso de 0,125 ms, o atraso algorítmico dessa Recomendação é de 0,625 ms.

2.1.2.4 Codificador segundo a Recomendação ITU-T G.729

Data de Aprovação: 19 de Março de 1996 [15].

Taxa de codificação: 8 kbit/s.

Tipo de Codificação: *Conjugate-Structure Algebraic-Code-Excited Linear-Prediction* (CS-ACELP). Sua concepção foi de codificar um sinal de voz com alta qualidade em 8 kbit/s, para ser usado em primeiro lugar em aplicações de comunicação sem fio e em segundo lugar por redes cabeadas que necessitem de uma compressão da banda usada pelo sinal codificado, tais como circuitos transoceânicos.

Atraso: Esta recomendação codifica a fala e outros sinais de áudio em quadros de 10 ms. Adicionalmente, temos um tempo de antecipação (*look-ahead*) de 5 ms, resultando em 15 ms o tempo de atraso algorítmico.

2.1.2.5 Codificador segundo a Recomendação ITU-T G.723.1

Data de Aprovação: 29 de Maio de 2006 [16].

Taxa de codificação: 5,3 e 6,3 kbit/s.

Tipo de Codificação: Este codificador foi otimizado para representar a voz com qualidade aceitável nas taxas mencionadas, mediante uma complexidade aceitável. Codifica-se a voz ou outros sinais de áudio em quadros mediante a codificação preditiva linear de análise por síntese. O sinal de excitação do codificador de maior taxa (6,3 kbit/s) é a quantificação multipulsos de máxima verossimilhança *Multi-Pulse Maximum Likelihood Quantization* (MP-MLQ) e o codificador de menor taxa utiliza a predição linear excitada pelo dicionário de códigos algébricos (*Algebraic Code Excited Linear Prediction* (ACELP)).

Atraso: Este codificador trata o sinal de voz ou outros sinais de áudio em quadros de 30 ms. Adicionalmente, tem-se uma antecipação de 7,5 ms, o que apresenta como resultado um retardo algorítmico total de 37,5 ms.

2.1.3 Codificador *internet Low Bitrate Codec* (iLBC)

Data de Aprovação: Dezembro 2004.

Taxa de codificação: 13,33 kbit/s (30 ms) e 15,20 kbit/s (20 ms).

Tipo de Codificação: O algoritmo do codificador iLBC utiliza a codificação

predictiva linear, suporta quadros de 20 ms em uma taxa de 15,2 kbit/s e quadros de 30 ms em uma taxa de 13,33 kbit/s.

Foi aprovado pela *Internet Engineering Task Force* (IETF) na RFC 3951 [17]. O Codificador *internet Low Bitrate Codec* (iLBC) é utilizado nas comunicações de voz sobre internet.

Sua principal característica é ter uma melhor resposta em cenários de redes que apresentem perdas de pacotes. Este codificador é utilizado por algumas aplicações como Skype e Google Talk [18].

2.1.4 Codificador Speex

Speex é um codificador de voz open-source [19], sendo sua ultima versão publicada em Dezembro de 2007. Este codificador se diferencia dos demais codificadores (exemplo AMR [20]), por não ter sido projetado para comunicações móveis e sim para redes de pacotes e aplicações VoIP, suportando diferentes taxas de codificação no intervalo de 2,15 até 24,6 kbit/s. Speex é baseado em um algoritmo CELP (*Code-Excited Linear Prediction*) que codifica o sinal de voz em quadros de 20 ms, ou seja, 160 amostras a 8 kHz, permitindo mudança de taxa a cada 20 ms. As taxas de codificação suportadas por este codificador são: 5,15 kbit/s, 8 kbit/s, 11 kbit/s, 15 kbit/s, 18,2 kbit/s e 24,6 kbit/s. Este codificador multitaxa será utilizado na avaliação do algoritmo de determinação de taxa de codificação proposto em este trabalho.

2.1.5 Parâmetros dos Codificadores de Voz

Em [21] descrevem-se em detalhes as principais técnicas de codificação de voz.

Tabela 2.1: Resumo dos valores dos parâmetros: Taxa de codificação, tempo de antecipação dos codificadores de voz G.711, G.726, G.727, G.728, G723.1, G729 e iLBC

Codificador	Algoritmo	Taxa de Codificação (kbit/s)	Tamanho de quadro + antecipação (ms)
G.711	PCM	64	0,125 + 0
G.726	ADPCM	40,32,24,16	0,125 + 0
G.727	ADPCM	40,32,24,16	0,125 + 0
G.728	LD-CELP	16	0,625 + 0
G.723.1	ACELP	5,3	30 + 7,5
G.723.1	MP-MLQ	6,3	30 + 7,5
G.729	CS-ACELP	8, 6,4, 11,8	10 + 5
iLBC	FB-ACELP	15,2, 13,3	20 + 5, 30 + 10

A Tabela 2.1 apresenta um resumo dos parâmetros: taxa de codificação,

duração do quadro (ms) e tempo de antecipação de codificadores de voz.

2.2 Estimadores de Qualidade de Voz

Os estimadores de qualidade de voz têm como objetivo dar uma pontuação a uma determinada comunicação. Estes resultados são utilizados para realizar melhoras nas redes de transmissão e, em geral, em todos os equipamentos que intervêm neste processo.

Neste trabalho, as Recomendações ITU-T P.862 [22], ITU-T P.563 [23] e ITU-T G.107 [24] serão utilizadas na avaliação dos codificadores em diferentes cenários de rede. Em seguida se verá a classificação dos estimadores de qualidade de voz.

2.2.1 Classificação dos estimadores de qualidade de voz

2.2.1.1 Métodos não Intrusivos

Neste tipo de método não é necessária uma amostra do sinal original da comunicação, a avaliação só é determinada pelo sinal no ponto que é analisado.

Por sua vez, este método pode ser de dois tipos:

- Objetivo, caso se utilize uma ferramenta (software) para a análise e cálculo da pontuação da qualidade;
- Subjetivo, caso um ouvinte intervenha diretamente na pontuação dos resultados sobre a qualidade.

Exemplos destes métodos são as recomendações:

- Métodos Objetivos: ITU-T G.107 (E-Model), ITU-T P.563.
- Métodos Subjetivos: ITU-T P.800 [25].

2.2.1.2 Métodos Intrusivos

O Método intrusivo é aquele que necessita de uma amostra de fala no ponto da origem da comunicação para poder comparar com a amostra do ponto destino, dando um resultado de avaliação de qualidade. Exemplos deste tipo de método são:

- Métodos Objetivos: Rec. ITU-T P.861 [26] e Rec. ITU-T P.862.

A Figura 2.1 - adaptado de [25] - apresenta a diferença entre estes dois métodos.

Figura 2.1: Diferença entre métodos intrusivos e não intrusivo

Nos seguintes itens serão descritas as recomendações da ITU-T: P.800, P.862, P.563 e G.107, sendo que as 3 últimas são utilizadas nos testes experimentais deste trabalho.

2.2.1.3 Recomendação ITU-T P.800

Esta recomendação constitui um guia para efetuar provas subjetivas da qualidade da transmissão em laboratórios, tendo como finalidade indicar os métodos considerados adequados para determinação do grau de satisfação por parte dos usuários no funcionamento de conexões telefônicas.

Os métodos recomendados são:

a) Testes de opinião sobre uma conversação.

Os testes de conversação em laboratório têm como objetivo reproduzirem em um laboratório as condições reais do serviço telefônico. Para isto, é necessário selecionar adequadamente as condições da rede e os participantes, além de conduzir as provas de forma apropriada.

No Anexo A desta recomendação [25] são detalhadas as considerações que devem ser cumpridas na execução dos testes para obter resultados confiáveis.

Segundo este mesmo anexo, estes testes devem realizar-se em um lugar de dimensões não menores a 20 m^3 com um tempo de reverberação menor que 500 ms, com níveis de ruído que se aproximem aos característicos de uma biblioteca ou hospital.

Quanto à seleção dos participantes dos testes, exigem-se as seguintes condições:

- Não deve ter participado diretamente de trabalhos de avaliação da qualidade de transmissão dos circuitos telefônicos ou tarefas similares como a codificação de sinais vocais.
- Não deve ter participado de nenhum teste subjetivo nos últimos seis meses e em nenhum teste de conversação nos últimos doze meses.

Quanto às metodologias utilizadas na realização dos testes, a ITU-T recomenda:

- Escala de opinião sobre a conversação: podem utilizar 5 diferentes pontuações para avaliar as diversas categorias de qualidade. A pontuação de opinião da Tabela 2.2 é a mais utilizada. Nestes testes o ouvinte atribui o adjetivo da coluna da esquerda e o condutor de teste faz a equivalência individual com o número inteiro da coluna da direita. Este procedimento é válido para os demais tipos de testes que são expressos nas tabelas 2.2, 2.3 e 2.4.

Tabela 2.2: Pontuação para MOS Conversacional e para testes de audição

Qualidade do sinal	Pontuação
Excelente	5
Boa	4
Regular	3
Ruim	2
Péssima	1

A média aritmética de qualquer conjunto destas pontuações se denomina pontuação média de opinião sobre a conversação e é representada por *Mean Opinion Score of conversation* (MOSc).

- Escala de Dificuldade: Trata-se de uma resposta binária obtida de cada participante no final de cada conversação. A pergunta que se faz é: seu interlocutor ou você experimentou alguma dificuldade para falar ou escutar através da conexão? Resposta: Sim, Não.

Atribuindo os seguintes valores: Sim = 1 e Não = 0; A quantidade avaliada (porcentagem de respostas sim) é denominada de dificuldade

percentual e é expressa com o símbolo: %D.

b) Testes de Audição - determinação do índice por categorização absoluta (*Absolute Category Rating* (ACR)).

Como este tipo de teste não alcança o mesmo grau de realismo que os testes de conversação, as considerações são menos rigorosas. Entretanto, exige-se um controle rigoroso de certos parâmetros como: a gravação do sinal de voz fonte, uma adequada calibração do sistema emissor, e a sala de audição que deve ter as mesmas condições de uma sala de gravação. Em relação aos sinais fontes, é fundamental utilizar mais de uma voz masculina e mais de uma voz feminina para diminuir o risco de que os resultados dependam das peculiaridades das vozes escolhidas.

Com respeito aos participantes dos testes (ouvintes), estes são escolhidos em uma população que utiliza normalmente os serviços telefônicos, com as seguintes condições: não haver participado diretamente em trabalhos relacionados com a avaliação da qualidade da transmissão de serviços telefônicos ou tarefas afins, não ter participado de testes subjetivos pelo menos há seis meses e em testes de audição por um ano e não ter tido contato com as frases do teste.

As escalas de opinião recomendadas pela ITU - T para este tipo de avaliação são:

- Escala de qualidade de audição: a magnitude média destas pontuações é representada pelo símbolo *Mean Opinion Score* (MOS). As pontuações para cada categoria desta escala são apresentadas na Tabela 2.2.
- Escala de esforço de audição: a magnitude média destas pontuações é representada pelo símbolo *Mean Opinion Score listening effort* (MOSle). As pontuações para cada categoria desta escala são apresentadas na Tabela 2.3.

Tabela 2.3: Pontuação MOS para testes de esforço de audição

Esforço necessário para compreender o significado das frases	Pontuação
Audição perfeita, nenhum esforço	5
Certa atenção é necessária, nenhum esforço apreciável	4
Esforço moderado	3
Esforço considerável	2
Significado incompreensível, com maior esforço	1

- Escala de preferência de audibilidade: a magnitude média destas pontuações é representada pelo símbolo *Mean Opinion Score listening predilection* (MOSlp). As pontuações para cada categoria desta escala são apresentadas na Tabela 2.4.

Tabela 2.4: Pontuação MOS para testes de preferência de audição

Preferência de audibilidade	Pontuação
Muito maior que a preferida	5
Maior que a preferida	4
Preferida	3
Menor que a preferida	2
Muito menor que a preferida	1

No método ACR, a Recomendação ITU-T P.800 no Anexo D define o método *Degradation Category Rating* (DCR), que basicamente é uma avaliação comparativa entre uma locução original de alta qualidade e a amostra a ser avaliada. Este método é utilizado quando a degradação é pequena. A escala vai desde inaudível até muito ruim. Portanto, é útil para se otimizar o sistema, uma vez comprovado pelo ACR, que a condição de degradação esteja dentro dos limites aceitáveis de qualidade.

2.2.1.4 Recomendação ITU-T P.862

A Recomendação ITU-T P.862, preparada pelo Grupo de Estudo 12 (2001 2004) da ITU-T, foi aprovada em 23 de Fevereiro de 2001.

O método objetivo descrito nesta Recomendação é conhecido como: Avaliação perceptual da qualidade vocal - *Perceptual Evaluation of Speech Quality* (PESQ). Em termos gerais, esta recomendação descreve um método objetivo para predizer a qualidade subjetiva de codificadores.

É aplicável tanto para codificadores de voz e para medições terminal-a-terminal. Desde que este método é válido para medições terminal-a-terminal, a recomendação PESQ considera diversos parâmetros que são intrínsecos a uma transmissão em uma rede real como: filtragem e retardo variável, assim como distorções devidas a erros de canal e a codificadores de baixa taxa de transmissão binária.

Recomenda-se que o método PESQ seja utilizado para avaliação de qualidade de voz de codificadores de banda estreita (0,3 - 3,4 kHz). Porém, a emenda 2 da Recomendação ITU-T P.862, aprovada em 29 de novembro de 2005, permite avaliar a qualidade de voz de codificadores de banda larga.

Deve observar-se também que o algoritmo PESQ não estabelece uma avaliação exaustiva da qualidade da transmissão, pois só mede os efeitos de ruído e a distorção de voz unidirecional sobre a qualidade de voz. Os efeitos de perda de audibilidade, retardo, eco e outros fatores de degradação relacionados a uma comunicação bidirecional não são consideradas neste método. Assim, é possível obter resultados altos na pontuação da qualidade mesmo que a conexão total seja ruim.

a) Descrição do Algoritmo usado pela PESQ

A PESQ compara um sinal original com um sinal degradado, resultante da passagem do sinal através de um sistema de comunicação. A saída PESQ é uma predição da percepção da qualidade que seria sentida por um indivíduo em um teste subjetivo de audição. Esta estimativa de índice MOS é denominada como *Mean Opinion Score listening quality objective* (MOS-LQO).

Em uma primeira etapa, uma série de atrasos entre os sinais originais e degradados são computados, um para cada intervalo de tempo onde os atrasos são significativamente diferentes entre os dois sinais. Para cada intervalo, são calculados os pontos de início e fim. O algoritmo de alinhamento tem condições de tratar alterações de atraso tanto nos períodos de silêncio como nos de fala.

A Figura 2.2 mostra a idéia básica usada no PESQ. O modelo computacional compara a entrada e a saída do dispositivo em teste.

Baseado no conjunto de atrasos encontrados, o sinal original e degradado já alinhados são comparados utilizando-se um modelo perceptual, conforme ilustrado na Figura 2.2. A idéia desse processo é transformar ambos os sinais em uma forma de representação interna que é análoga à representação psicofísica do sinal no sistema auditivo humano, levando em consideração a frequência perceptual e a intensidade. Isto é alcançado em vários estágios: alinhamento temporal, alinhamento do nível de intensidade, mapeamento no tempo-frequência, transformação da escala de frequências e compressão da escala de intensidade.

b) Variáveis que influem nos testes de PESQ

As Tabelas 2.5, 2.6, 2.7 mostram de forma resumida os fatores que devem se considerar para obter resultados válidos nos testes. A Tabela 2.5 apresenta as relações dos fatores de prova, tecnologias de codificação e aplicações onde os testes são significativos.

Figura 2.2: Algoritmo utilizado pela PESQ

Tabela 2.5: Cenários de teste significativos para o algoritmo PESQ

Fatores de Testes
Níveis de entrada do sinal de voz em um codificador.
Erros no canal de transmissão.
Perda de pacotes e dissimulação da perda de pacotes com codificadores CELP.
Codificadores multitaxa.
Ruído ambiental no lado emissor.
Tecnologia de codificação
Codificadores de forma de onda, por exemplo: G.711; G.726; G.727.
CELP e codificadores híbridos com taxa maior que 4 kbit/s, por exemplo: G.728, G.729, G.723.1.
Outros codificadores: GSM-FR, GSM-HR, GSM-EFR, GSM-AMR, CDMA-EVRC, TDMA ACELP, TDMA-VMSELP.
Aplicações
Avaliação de codificadores.
Seleção de codificadores.
Provas em redes emuladas e redes protótipo.

Na Tabela 2.6 apresentam-se algumas variáveis onde o método PESQ tem predições não-confiáveis (impróprios).

A Tabela 2.7 apresenta fatores, tecnologias e aplicações para os quais o PESQ ainda não foi avaliado.

Tabela 2.6: Cenários de teste impróprios para o algoritmo PESQ

Predições que não possuem significância
Perda de sonoridade.
Atraso, efeito de atraso em provas conversacionais.
Eco percebido pelo falante.
No processo de codificação com seções contínuas de voz cortadas onde este período extraído representa mais do que 25% do total do período de atividade de voz.
Qualidade de comportamento de comunicações bidirecionais.

Tabela 2.7: Cenários de teste para os que PESQ não foi avaliado

Fatores de Testes
Substituição de sessões de fala por silêncio.
Saturação de amplitude do sinal vocal.
Efeitos de dependência com respeito ao locutor.
Falantes simultâneos.
Música como entrada do codificador.
Eco para o ouvinte e os efeitos dos compensadores de eco e ruído.
Tecnologia de codificação
CELP e Codificadores híbridos com taxas menores de 4 kbit/s.
MPEG4.
Aplicações
Provas de terminais acústicos mediante o simulador de cabeça e dorso.

2.2.1.5 Recomendação ITU-T P.563

O algoritmo P.563 é aplicável para a avaliação da qualidade de voz sem a necessidade de um sinal de referência. Por este motivo, é recomendado para a avaliação não intrusiva da qualidade vocal e para a supervisão de uma rede em funcionamento baseadas em um só extremo. Esta recomendação não se limita a medições extremo a extremo, pode ser também utilizada em um ponto qualquer da cadeia de transmissão. A pontuação, assim calculada, é comparável à qualidade percebida por um ouvinte humano que escuta o sinal no ponto de teste. Esta estimativa de índice MOS é denominada como MOS-LQO.

Esta recomendação deve ser utilizada para avaliar a qualidade de fala em aplicações de telefonia de 3,1 kHz de largura de faixa.

Deve-se ressaltar que o algoritmo P.563 não proporciona uma avaliação completa da qualidade de transmissão. Assim só se podem medir os efeitos da distorção unidirecional da voz e do ruído sobre a qualidade vocal, da mesma forma que se analisaria com um teste de audição, que avalia a qualidade numa escala ACR. Isto significa que os efeitos de perda de sonoridade, atrasos, eco do orador, e outros que afetem as interações bidirecionais não influenciam as pontuações do P.563.

Também é bom ressaltar que este algoritmo está projetado exclusivamente para a avaliação da voz humana, não podendo ser aplicado para: música ou em geral outros sinais de áudio não vocais.

O sinal de voz digitalizada deve satisfazer os seguintes requisitos:

- Frequência de amostragem: 8000 Hz;
- Resolução de amplitude PCM linear de 16 bits;
- Duração mínima de 3,0 s e máxima de 20 s;
- Relação mínima de atividade vocal 25 % e uma máxima de 75 %.

2.2.1.6 Recomendação ITU-T G.107

Esta Recomendação, mais conhecida como E - Model, propõe um modelo computacional que mede os efeitos dos parâmetros de uma rede de transporte e do sinal na qualidade de voz transmitida. O E-Model é definido pela seguinte equação:

$$R = R_o - I_s - I_d - I_e + A \quad (2.2)$$

Onde:

- R : fator de determinação do índice de transmissão, representa o parâmetro que está relacionado com o valor de qualidade e tem uma correspondência com a pontuação MOS ITU-T P.800;
- R_o : fator que expressa a relação sinal-ruído. Na prática este valor é obtido ao definir valores padrões para os diferentes parâmetros desta recomendação e expressa a qualidade máxima que se pode alcançar em um sistema de comunicação;
- I_s : fator de degradações simultâneas, representa as degradações que se produzem simultaneamente com o sinal de fala, como: os níveis dos sinais de fala muito forte, o efeito local não ótimo e o ruído de quantização;
- I_d : degradação da qualidade devida ao atraso;
- I_e : degradação da qualidade por efeito do equipamento (codificador);

- A : fator de melhora, não tem relação com os demais parâmetros da transmissão. É denominado como fator de vantagem devido à vantagem de acesso que alguns sistemas de comunicação proporcionam. Assim, este fator pode-se interpretar como um valor que reflete a tolerância que o usuário tem ao decremento da qualidade em compensação de uma vantagem de acesso como o que proporciona os sistemas moveis. A diferença dos outros parâmetros é um valor que se adiciona ao valor de R_o .

O fator A é considerado em alguns trabalhos como um fator que não deve ser considerado ou que em todo caso se deixa a critério do projetista de rede utilizá-lo ou não [27] e [28]. Neste trabalho o E-model será empregado para cenários com atrasos ponto a ponto, sendo o fator A desconsiderado já que a rede emulada onde são efetuados os testes é cabeada.

O valor padrão de R_o é 93,2 [24], que é obtido ao se colocar todas as entradas do modelo com seus valores padrão, por exemplo: o parâmetro de degradação de qualidade devido ao atraso (I_d) e o parâmetro correspondente à degradação do equipamento (I_e) para este cálculo não levam em conta a probabilidade de perda de pacotes, obtendo R_o com uma qualidade alta, próxima ao valor limite de 100.

Para o cenário de testes (emulação de rede de pacotes) realizado neste trabalho, explicado em detalhe na seção 3.2, com condições ideais (sem perdas e nem atrasos de pacotes) e utilizando o codificador G.711, tem-se um valor para R_o de 86,5.

O parâmetro I_s não será considerado nos cálculos, já que este descreve condições que são relacionadas com o sinal, não dependendo da rede de transporte.

O fator A , segundo [24] é nulo para redes cabeadas e é o que corresponde ao cenário de emulação deste trabalho. Os seguintes valores de A são descritos como:

- Sistema de comunicação convencional (cabeado), $A = 0$;
- Sistema de comunicações móveis em edifício (sem fio), $A = 5$;
- Sistema de comunicações móveis em veículos (sem fio), $A = 10$;
- Sistema de comunicações por satélites, $A = 20$.

O fator de atraso I_d é definido pela composição:

$$I_d = I_{dle} + I_{dte} + I_{dd} \quad (2.3)$$

As parcelas I_{dle} e I_{dte} correspondem aos atrasos devidos ao eco tanto no emissor como no receptor. Estes fatores não serão considerados neste trabalho já que para o cenário de testes se assumirá uma supressão perfeita do eco.

O I_{dd} representa a degradação causada pelos atrasos (T_a) produzidos na rede e no codificador, inclusive quando se assume uma supressão perfeita de eco. O atraso de rede será configurado segundo o tipo de teste no emulador da rede. O atraso ocasionado pelo codificador se define em [24] como:

$$2N + \text{Antecipação} \quad (2.4)$$

Onde:

N é o tamanho da duração do sinal de cada codificador. Valores de N e dos tempos de antecipação para diferentes codificadores podem ser encontrados na Tabela 2.1.

O I_{dd} é definido como:

Para $T_a \leq 100$ ms : $I_{dd} = I_d = 0$

Para $T_a > 100$ ms :

$$I_{dd} = I_d = 25 \left\{ (1 + d^6)^{\frac{1}{6}} - 3 \left(1 + \left[\frac{d}{3} \right]^6 \right)^{\frac{1}{6}} + 2 \right\} \quad (2.5)$$

Onde:

$$d = \frac{\log \left(\frac{T_a}{100 \text{ ms}} \right)}{\log 2} \quad (2.6)$$

Com estas considerações, pode-se calcular o parâmetro R em função dos parâmetros que correspondem ao valor R_o , ao atraso (I_d) e ao fator correspondente ao codificador (I_e) como:

$$R = R_o - I_d - I_e \quad (2.7)$$

O Fator R está relacionado com o índice M , segundo a seguinte equação:

$$R = 3,026 \times M^3 - 25,314 \times M^2 + 87,06 \times M - 57,336 \quad (2.8)$$

A Figura 2.3, extraída de [24], apresenta a relação entre o MOS e o parâmetro R .

Figura 2.3: Relação entre o MOS e o parâmetro R

2.3 Descrição de uma rede VoIP

A diferença entre uma rede comutada por circuitos e rede comutada por pacotes é que a comutada por circuitos é utilizada pela telefonia fixa tradicional, onde cada canal tem uma taxa de 64 kbit/s e só transmite uma conversação por canal; já as redes comutadas por pacotes podem misturar a transmissão de diversos serviços multimídia como: web, jogos, vídeos e áudio, além da transmissão de uma chamada de voz, já que esta é tratada como uma simples transmissão de pacotes, tendo grandes vantagens na utilização adequada da largura de banda e dos demais recursos da rede. Entretanto, deve-se considerar que o controle e gestão de uma chamada de voz sobre IP requerem mais recursos que uma rede de telefonia tradicional devido ao grande número de fatores que podem degradar a qualidade da transmissão e que devem ser controlados.

Estes fatores, assim como a descrição geral de uma rede IP, serão tratados neste capítulo.

2.3.1 Arquitetura de uma rede VoIP

Independente do protocolo, na recomendação de uma rede VoIP distinguem-se os seguintes elementos [29]:

- Interface do cliente: encarregada de originar ou receber as chamadas do cliente, podem ser equipamentos que integrem diretamente a rede VoIP, por exemplo, telefones que seguem a recomendação H.323 ou equipamentos que proporcionem uma interface para rede VoIP e uma ou mais interfaces tradicionais de voz como: POTS (*Plain Old Telephone System*) ou ISDN (*Integrated Services Digital Network*).
- O *Gateway* de Rede: Permite a comunicação entre a rede VoIP e as redes tradicionais de comutação de circuitos, é responsável pela digitalização, compressão, demodulação e funções de empacotamento do sinal de voz.
- O Agente de chamada: Realiza a mesma função que o nó de comutação de rede de voz tradicional (*switch*), sendo suas principais funções o encaminhamento das chamadas, a implementação e o gerenciamento de serviços suplementares.

Esta rede deve possuir um meio de transporte tanto para a sinalização das chamadas como para a voz digitalizada, devendo seguir condições de projeto específicas que permitam o transporte da voz com a qualidade adequada. A Figura 2.4 apresenta o esquema da arquitetura VoIP.

Figura 2.4: Elementos de uma Rede IP para comunicações de voz

2.3.2 Informação de um pacote VoIP

Como o objetivo deste trabalho é conhecer os fatores que influenciam a *Quality of Service* (QoS) em uma transmissão de voz sobre IP, é importante saber como a voz é transmitida na rede. Assim, as informações adicionais para uma adequada

transmissão de voz representam uma carga de bits adicional a ser transmitida, que tem influência primeiramente na taxa de transmissão e, dependendo do congestionamento da rede, na qualidade final do sinal de voz.

As tecnologias desenvolvidas para uma rede IP seguem um modelo de camadas hierárquico, como no modelo *Open Systems Interconnection* (OSI). Assim, cada concatenação de cabeçalho contém a informação correspondente a uma determinada camada [30].

A Figura 2.5 representa o formato dos pacotes de voz transmitidos em uma rede IP, mostrando o processo onde o sinal de voz é digitalizado e comprimido mediante um algoritmo de codificação, formando-se um pacote de N Bytes que é conhecido como carga útil. Este sinal de voz já codificado deve ser empacotado utilizando o protocolo *Real Time Protocol* (RTP). Este protocolo transporta os pacotes de voz codificados. Desde que os pacotes possam ser encaminhados de formas diferentes, suas chegadas no ponto destino não terão a mesma ordem. O RTP tem a função de reordenar os pacotes antes que a voz seja decodificada. Em conjunto com o protocolo RTP, utiliza-se o protocolo RTCP que é encarregado pelo controle da chamada, tendo também como função informar o estado dos parâmetros da rede aos pontos finais da comunicação. As informações do estado da rede que o RTCP contém serão tratadas no item 2.6.1.2.

Figura 2.5: Formato do pacote de voz transmitida

Nas aplicações interativas tem-se uma grande necessidade de decrementar o tempo da entrega da informação. Por tal motivo, utiliza-se o protocolo de transporte *User Datagram Protocol* (UDP) para transmissão de voz, através de RTP. Para o encaminhamento dos pacotes na rede, de modo que os roteadores encontrem o endereço do destino, utiliza-se o protocolo IP.

Com a adição de cabeçalhos (20 Bytes de IP, 8 Bytes de UDP e 12 Bytes de RTP) o pacote de voz codificado tem um cabeçalho adicional (*overhead*) de 40

bytes.

Cabe mencionar que existem técnicas de redução de largura de banda que serão tratadas posteriormente neste capítulo.

Os N bytes do quadro da voz digitalizada dependem do tipo de codificador que é utilizado na comunicação. Na Tabela 2.8, apresentam-se as taxas de codificação e as taxas de transmissão requeridas para cada codificador.

Tabela 2.8: Composição da Taxa de Transmissão para Codificadores G.711 e G.729

Codificador	Taxa Nominal	Duração Equivalente de voz (ms)	Tamanho do pacote (Bytes)	Taxa de Transmissão Requerida (kbit/s)
G.711	64 kbit/s	20	160	80
G.711	64 kbit/s	10	80	96
G.729	8 kbit/s	20	20	24
G.729	8 kbit/s	10	10	40

A Tabela 2.8 não inclui os bytes correspondentes aos cabeçalhos adicionais, que correspondem aos protocolos da camada de enlace. Esta quantidade de bytes referida vai depender do tipo de protocolo utilizado, por exemplo, em uma rede Ethernet tem o formato apresentado na Figura 2.6 [31].

ETHERNET							
	PREAMBULO ETHERNET	CABECALHO ETHERNET	CABECALHO IP	FRAME	CRC	GAP	
BYTES	8	14	40		4	12	78

Figura 2.6: Formato do quadro Ethernet

Porém, para uma rede Frame Relay, o formato é o apresentado na Figura 2.7 [31].

FRAME - RELAY							
	FLAG	ADDRESS	CABECALHO IP	FRAME	FCS	FLAG	
BYTES	1	4	40		2	1	48

Figura 2.7: Formato do quadro Frame Relay

Na Tabela 2.9, vide Seção 2.5.1, são apresentados exemplos de taxa de transmissão para diferentes codificadores de voz utilizando uma rede Ethernet e Frame Relay.

2.3.3 Protocolos e Recomendações em uma rede VoIP

As Organizações Internacionais de Normas têm definido um conjunto de novos mecanismos de sinalização e transporte, com o objetivo de transmitir a voz sobre uma rede de pacotes com a mesma ou similar qualidade que se tem em uma rede de telefonia fixa tradicional.

Os sofisticados mecanismos de sinalização desenvolvidos devem suportar todas as opções ou serviços de chamadas que já são atendidos, e os mecanismos de transporte devem enviar os pacotes com a maior rapidez e qualidade possível. Abaixo são citados os principais protocolos e recomendações.

a) Protocolo *Session Initiation Protocol* (SIP)

O SIP é definido pela IETF - RFC 2543 é um protocolo de sinalização que controla o início, as modificações e terminações de sessões multimídia interativas. Estas sessões podem ser de áudio ou vídeo entre uma ou mais partes, seções de conversas ou de jogos. Tem uma sintaxe baseada em protocolo de texto similar ao *Hypertext Markup Language* (HTML) ou ao *Simple Mail Transfer Protocol* (SMTP) [32]. SIP é um protocolo Cliente/Servidor com transporte sobre UDP ou *Transmission Control Protocol* (TCP), sendo o primeiro o mais comum nas implementações por sua simplicidade e maior velocidade.

As principais funcionalidades do SIP são:

- Localização de usuário: SIP proporciona a capacidade de localização do usuário final para o estabelecimento de uma sessão. Portanto, a mobilidade do usuário está suportada pelo SIP durante a sessão;
- Capacidade de serviços dos usuários: SIP habilita as capacidades de dados dos dispositivos que estabelecem a sessão;
- Disponibilidade do usuário;
- Início de Sessão;
- Manutenção da sessão: o SIP administra as modificações, transferências e terminação de uma sessão ativa.

Os elementos de uma rede SIP são:

- Agente Usuário: Existem dois tipos de agentes usuários, os quais sempre estão presentes e permitem a comunicação entre cliente e servidor:

- O Agente Usuário cliente (UAC): O UAC gera petições e recebe respostas.
- O Agente Usuário Servidor (UAS): O UAS responde às petições SIP.
- Servidores SIP: Existem três classes lógicas de servidores, um servidor pode ter uma ou mais destas classes. Estas classes são as seguintes:
 - Servidor de Registro (Registrar Server): realiza a operação de correspondência entre os endereços SIP e os endereços IP. Este servidor só aceita mensagens REGISTER, o que facilita a localização de usuários, pois o usuário, onde quer que ele se encontre, primeiro tem que se registrar neste servidor.
 - Servidor *Proxy* (*Proxy Server*): recebe requisições e as encaminha para outros servidores que contenham informações mais precisas sobre a localização do usuário chamado.
 - Servidor de Redirecionamento (*Redirect Server*): recebe requisições, mas no lugar de encaminhá-las para outros servidores, retorna o endereço do servidor para o qual a requisição deverá ser direcionada.

As mensagens SIP podem ser de dois tipos: requisições e respostas, segundo RFC3261.

Nas Requisições SIP, definem-se seis métodos básicos:

- INVITE: Permite convidar um usuário a participar de uma sessão ou modificar parâmetros de uma sessão já estabelecida.
- ACK: Confirma o estabelecimento da sessão.
- OPTION: Solicita informação de algum servidor em particular.
- BYE: Indica o término de uma sessão.
- CANCEL: Cancela uma petição pendente.
- REGISTER: Registra o agente usuário.

Existem também mensagens SIP que são respostas às requisições, sendo estas respostas de seis tipos. Cada uma destas respostas se diferencia pelo primeiro dígito de seu código estando os dois dígitos seguintes em formato decimal. Estas são:

- 1xx: Mensagens de inicialização da conexão;
- 2xx: Respostas de êxito;

- 3xx: Respostas de redirecionamento;
- 4xx: Respostas de falhas no seqüenciamento das mensagens;
- 5xx: Respostas de falhas de servidor;
- 6xx: Respostas de falhas globais.

Algumas destas mensagens apresentam-se no seguinte diagrama de fluxo de sinalização, Figura 2.8 - extraído de [29], em um cenário com Servidor *Proxy*.

Figura 2.8: Seqüência de mensagens de sinalização SIP

b) Recomendação H.323

O H.323 é um conjunto de especificações da ITU-T [33] relacionadas à transmissão de áudio, vídeo e dados através de uma rede IP. Esta recomendação contempla protocolos de sinalização, controle, transporte de multimídia, tanto para comunicações ponto a ponto como para multiponto [32].

A Figura 2.9 apresenta a pilha de protocolos [29].

Podemos destacar quatro componentes [34] de uma rede H. 323:

- **Terminais:** geralmente, as informações que circulam em um ambiente H.323 são originadas ou destinadas aos terminais. Um terminal H.323 pode ser um telefone IP ou uma máquina executando um aplicativo (*softphone*) com recursos multimídia.
- **Gateway:** tem como função a interoperabilidade entre redes. São necessários quando se quer estabelecer uma comunicação entre dois

Figura 2.9: Protocolos da Recomendação H.323

terminais que pertencem a redes distintas, este é um terminal pertencente a uma rede H.323 e aquele a rede SIP. Para que esta interoperabilidade seja possível, estes protocolos de sinalização proporcionam uma série de funções, estando entre as principais: conversão do formato de codificação de mídia, tradução dos procedimentos de estabelecimento e encerramento de chamadas. Também cumpre a função de interoperabilidade com uma rede telefônica convencional.

- *Gatekeeper* (GK): é uma entidade de gerência. Este permite o controle centralizado do sistema, dado que todo terminal deve estar registrado nele facilitando a implementação de políticas de acesso. Dentre suas principais funções destacam-se:
 - Tradução de endereço;
 - Roteamento de chamadas;
 - AAA (*Authentication, Authorization e Accounting*);
- Unidade de Controle Multiponto (MCU): é um componente que permite a comunicação multiponto (conferência) entre três ou mais terminais.

c) O Protocolo RTP

O RTP é o mecanismo padrão na implementação de transporte de mídia em telefonia de redes IP. É um mecanismo utilizado nos principais protocolos de sinalização, que proporciona transporte de ponto a ponto de dados em tempo real, tais como áudio e vídeo; e dados para aplicações que são em

tempo real. O RTP não garante QoS e não tem a função de reservar recursos ao longo do trajeto da conexão. Também requer um protocolo de sinalização para o início da conexão [35].

A funcionalidade de RTP apresenta melhoras com o RTCP, o qual proporciona um controle dos dados enviados e a qualidade de serviço. Estas Recomendações estão definidas na RFC-3550 desenvolvida pela IETF [36]. RTP e RTCP são usados geralmente em portas consecutivas de UDP.

No item 2.3.2 especifica-se a largura do cabeçalho RTP (12 Bytes) e como este protocolo cumpre a função de realinhamento de pacotes no ponto de destino.

No item 2.6.1.2 descreve-se que informação correspondente ao estado da rede é administrada pelo protocolo RTCP para ter um controle da qualidade em uma comunicação VoIP.

2.4 Principais fatores de degradação da qualidade em uma rede de dados

Podem-se dividir os fatores de degradação em:

- Taxa de perda de pacotes - *Packet loss rate* (PLR): está representada pela fração de pacotes que foram transmitidos pela rede, mas não alcançaram seu destino em um tempo determinado.

A localização dos pacotes perdidos dentro dos dados enviados influi na qualidade de voz [37].

- Atraso ponto a ponto - *Packet delay transfer* (PDT): é definido como o tempo que demora para uma mensagem chegar desde o ponto de origem até o ponto destino, este atraso ocorre devido a diversos fatores [32] como: o algoritmo de codificação, o meio de transmissão e o atraso de filas em roteadores. A Recomendação ITU-T G.114 [38] determina que para se ter uma qualidade aceitável, o atraso fim a fim não deve ser superior a 150 ms em uma direção. Atrasos entre 150 a 400 ms ainda podem ser aceitáveis, mas acima deste valor a qualidade é inaceitável.
- Variação de Atrasos (*Jitter*) - *Packet delay Variation* (PDV): é a variação de atrasos na chegada dos pacotes ao ponto destino. Em uma rede de transmissão de pacotes é necessário colocar um *jitter buffer* para cancelar a variação dos atrasos.

2.5 Taxa de Codificação e Taxa de Transmissão

Cada chamada VoIP requer uma quantidade mínima de banda para sua realização. Neste ponto se realiza um cálculo analítico da banda nominal necessária para cada codificador e em seguida apresentam-se os resultados destes. Adicionalmente se descrevem alguns mecanismos para reduzir a taxa de transmissão nominal.

2.5.1 Composição da taxa de transmissão de codificadores

Será determinada, analiticamente, a taxa de transmissão necessária para os seguintes codificadores: G.711, G.728, G.726, G.729, G.722, G.723.1, G.723.1.a, GSM (13 kbit/s), AMR (12,2 kbit/s), AMR (7,4 kbit/s), AMR (6,6 kbit/s), AMR (4,750 kbit/s) e para os protocolos de camada 2: Ethernet e Frame-Relay.

A largura de banda necessária para transportar a voz sobre uma rede IP depende de vários fatores. Entre eles podemos mencionar:

- Tipo de codificador de voz;
- Período de amostragem;
- Cabeçalho RTP/UDP/IP;
- Protocolo da camada 2.

A seguinte equação será usada para determinar a taxa de transmissão [35]:

$$T_t = [(T_p + C_{ip})/T_p] \times T_c \quad (2.9)$$

onde:

- T_p = Tamanho do pacote (Bytes);
- T_c = Taxa de Codificação (bit/s);
- T_t = Taxa de Transmissão (bit/s);
- C_{ip} = Cabeçalhos RTP/UDP/IP/Camada2 (Bytes).

O número de bits que correspondem ao overhead depende do protocolo de transporte. Assim para uma rede Ethernet ter-se-ia 78 bytes (considerando os 40 Bytes dos cabeçalhos RTP/IP/UDP) e para uma rede Frame Relay 48 Bytes.

Com estas considerações temos os seguintes resultados na Tabela 2.9, que mostram a largura de banda nominal por codificador.

Tabela 2.9: Taxa de transmissão nominal de codificadores

Codificador	Taxa de codificação (kbit/s)	Duração de quadro (ms)	Tamanho da carga útil (Bytes)	Número de quadros por pacote	Taxa de transmissão FR (bit/s)	Taxa de transmissão Eth (kbit/s)
G.711	64	20	160	1	83,200	95,200
G.728	16	10	20	1	54,400	78,400
G.726	32	20	80	1	51,200	63,200
GSM	13	20	33	1	32,200	44,200
G.729	8	10	10	2	27,200	39,200
G.723.1	6,4	30	24	1	19,200	27,200
G.723.1A	5,3	30	20	1	18,100	26,100
AMR	12,2	20	31	1	31,400	43,400
AMR	7,4	20	19	1	26,600	38,600
AMR	6,6	20	17	1	25,800	37,800
AMR	4,75	20	12	1	23,950	35,950

2.5.2 Mecanismos para reduzir a taxa de transmissão

Entre os mecanismos utilizados para reduzir a largura de banda necessária podemos citar:

- Número de quadros por pacote [39]: o codificador define o tamanho da amostra mais o número de amostras colocadas num pacote, afetando o número de pacotes enviados por segundo. Em geral, aumentando o tamanho do frame reduz-se a banda necessária.

Assim por exemplo, uma chamada com codificador G.723.1 (6,4 kbit/s) com tamanho de pacote de 24 Bytes (30 ms de amostragem) e com um só quadro por pacote tem uma taxa total de transmissão de: $[(24 + 78) / 24] \times 6400 = 27,2$ kbit/s. Com dois quadros por pacote ter-se-ia uma taxa de: $[(48 + 78) / 48] \times 6400 = 16,8$ kbit/s.

Para estes cálculos considerou-se o cabeçalho de uma rede Ethernet.

- Compressão de cabeçalho RTP/UDP/IP [40]: o cabeçalho IP de 40 bytes é um valor constante independente do codec utilizado. Usando *Compressed Real Time Protocol* (cRTP)(compressão de cabeçalho RTP/UDP/IP), esses 40 Bytes transformam-se em 2 ou 4 Bytes e, portanto, permite uma considerável redução na taxa de transmissão.
- Detecção da atividade de voz VAD (Voice Activity Detection): nas redes comutadas por circuitos, todas as chamadas de voz utilizam uma banda

fixa de 64 kbit/s, independente de quanto da comunicação é voz ou quanto é silêncio. Nas redes de pacotes toda a conversação e silêncio são empacotados. Alguns dos codificadores utilizados dispõem de recursos de supressão de silêncio, chamados de VAD, que suprimem a transmissão de dados durante os períodos de silêncio. Para evitar situações onde o silêncio pode ser considerado como um desligamento da chamada, conjugado com o VAD proporciona-se o chamado *Comfort Noise Generation* (CNG), que gera ruído de conforto de fundo, com a finalidade de manter ativa a comunicação estabelecida [41].

Tendo-se em consideração o cRTP e o VAD, obtiveram-se os resultados apresentados na Tabela 2.10 para uma rede Ethernet com cabeçalho de 42 bytes. Para o caso do VAD considera-se uma redução de 30% na taxa de transmissão.

Tabela 2.10: Taxa de transmissão nominal de codificadores com VAD e cRTP

Codificador	Taxa de codificação (kbit/s)	Duração de quadro (ms)	Tamanho da carga útil (Bytes)	Número de quadros por pacote	Taxa de transmissão Eth. com FR (bit/s)	Taxa de transmissão Eth. com cRTP e VAD (kbit/s)
G.711	64	20	160	1	80,80	56,56
G.728	16	10	20	1	49,60	34,72
G.726	32	20	80	1	48,80	34,72
GSM	13	20	33	1	29,80	20,86
G.729	8	10	10	2	24,80	17,36
G.723.1	6,4	30	24	1	17,60	12,32
G.723.1A	5,3	30	20	1	16,50	11,55
AMR	12,2	20	31	1	29,00	20,30
AMR	7,4	20	19	1	24,20	16,94
AMR	6,6	20	17	1	23,40	16,38
AMR	4,75	20	12	1	21,55	15,09

2.6 Algoritmo para Determinação da taxa de codificação (RDA)

A utilização de um RDA em um sistema VoIP tem como objetivo melhorar a qualidade das comunicações e otimizar o uso da rede, podendo adaptar-se a taxa de codificação de um codificador multitaxa segundo as condições da rede de uma forma dinâmica e sem a intervenção do usuário.

Nesta seção, são apresentados os principais métodos existentes de implementação de RDA na literatura.

Na seção 3.3, apresenta-se o RDA implementado neste trabalho, mostrando a metodologia, cenários de testes e resultados obtidos.

Adicionalmente, nesta seção apresentam-se alguns métodos existentes para melhorar a confiabilidade de uma comunicação e que podem ser utilizados conjuntamente (FEC, *Buffer Playout*) com o RDA proposto.

2.6.1 Métodos existentes de RDA

Existem algumas implementações baseadas em características do sinal [1], [2], [3] e [4] e outras baseadas nos parâmetros da rede [36], [42] e [43] tendo para cada um destes casos distintos fatores de decisão que comandam um RDA, com o propósito de modificar a taxa de codificação para uma mais adequada, e assim, conseguir uma melhor qualidade do sinal transmitido e uma melhora na otimização da rede.

A seguir, são apresentados os principais métodos.

2.6.1.1 RDA baseado na análise do sinal

Um dos principais objetivos no processamento digital de sinais é obter a máxima informação possível do sinal de voz. Assim, por exemplo, o propósito do processamento digital pode ser determinar se um segmento de uma onda de sinal corresponde ao sinal de voz ou não. Em [44] define-se que um segmento de sinal pode ser classificado como:

- Voz sonora (voz).
- Voz surda (silêncio).
- Silêncio (ruído de fundo).

Existem diferentes parâmetros intrínsecos ao sinal com os quais se pode determinar o tipo de sinal em estudo, que são:

- Energia de curto prazo.
- Taxa média de cruzamentos por zero em curto prazo.
- Função de auto correlação.

Ao observar um sinal de voz em função do tempo pode-se notar que a amplitude do sinal de voz varia substancialmente, em particular a amplitude dos segmentos do sinal surdo é geralmente muito mais baixa que a amplitude de um segmento de voz sonora. A energia de curto prazo de um sinal de voz proporciona uma conveniente representação das variações que ocorrem com a amplitude do sinal de voz, permitindo distinguir segmentos de voz sonora e voz surda, sendo possível determinar transições entre segmentos de voz sonora e segmentos de voz

surda. No caso de voz de alta qualidade (SNR alta) a energia de curto prazo pode ser utilizada para distinguir a voz do silêncio.

O modelo de geração de voz sugere que a energia de voz sonora é principalmente localizada abaixo dos 3 kHz e, para voz surda, a maior parte da energia é localizada em frequências mais altas. Assim, desde que altas frequências implicam uma alta taxa média de cruzamentos por zero e baixas frequências implicam uma baixa taxa média de cruzamentos por zero, pode-se observar que existe uma forte correlação entre a taxa média de cruzamento por zero e energia de curto prazo com a sonoridade. Uma razoável generalização é que se a taxa de cruzamentos por zero é alta o sinal de voz é surdo e se a taxa de cruzamentos por zero é baixa o sinal de voz é sonoro.

O problema de discriminar a voz do silêncio ou ruído de fundo não é trivial, salvo os casos onde a relação sinal-ruído seja alta e seja suficiente a medição de energia de cada segmento do sinal, onde a energia dos mais baixos níveis de sonidos de voz (fricativos) excede a energia do ruído de fundo.

A combinação dos parâmetros, energia e taxa de cruzamentos por zero, é utilizada para elaborar algoritmos que ajudem na determinação da localização do início e final do sinal de voz. Um exemplo deste tipo de algoritmo é apresentado em [45].

A função de autocorrelação é utilizada para estimar a periodicidade em diferentes tipos de sinais, incluindo o sinal de voz. A forma de onda dos segmentos de voz surda não apresenta periodicidade definida, isto é, os picos de sinal nestes segmentos de voz não têm período definido. Os segmentos de sinais de voz sonora, pelo contrário, apresentam periodicidade.

O VAD separa o sinal de voz (seja ele surdo ou sonoro) do silêncio (ruído de fundo). Uma regra básica para se implementar o VAD é que o sinal de voz deve ser considerado quase estacionário durante segmentos curtos de 10 ms a 20 ms.

Em um codificador de voz que não tenha VAD implementado, o mesmo número de bits é alocado para todos os frames de voz, sem considerar o fato de que o emissor da voz possa estar em silêncio, em aproximadamente 63% do tempo total de uma conversação bidirecional. Desta forma, não se tem um controle apropriado da utilização do canal de comunicação. Outro benefício da utilização do VAD é que em comunicações móveis se minimiza o consumo de energia dos terminais, sendo este fator muito importante a considerar, devido à maior duração das baterias e à possibilidade de diminuição do tamanho destes componentes [46].

Mudando-se a taxa de codificação em função das propriedades do sinal, é possível determinar uma taxa mais adequada, ou seja, uma maior atividade de fala requer uma taxa maior e em casos de silêncio pode ser usada uma taxa menor.

Estudos como [47], [48] e [49] apresentam um RDA que como parte do fator de decisão considera a discriminação do VAD e as diferentes técnicas de determinação de sonoridade do sinal de voz. Assim em [50] o algoritmo sugerido pode-se resumir da seguinte forma: são tomados 3 segmentos consecutivos do sinal para determinar se o seguinte segmento pode-se prever como segmento de voz ou silêncio, como é apresentado na Tabela 2.11. No exemplo, os 3 primeiros segmentos já foram determinados como voz (1) ou silêncio (0).

Tabela 2.11: Classificação de um segmento de sinal

Primeiro segmento	Segundo segmento	Terceiro segmento	Decisão de VAD	Saída VAD
0	0	0	silêncio	0
0	0	1	silêncio	0
0	1	0	voz	1
0	1	1	voz	1
1	0	0	silêncio	0
1	0	1	voz	1
1	1	0	voz	1
1	1	1	voz	1

Com a análise anterior pode-se determinar qual será a taxa de codificação final, como é apresentado na Tabela 2.12, sempre considerando-se que a maior atividade de fala possui uma maior taxa e, em casos de silêncio, há uma menor taxa.

Tabela 2.12: Decisão sobre taxa de codificação final

Taxa Preliminar	Decisão VAD	Taxa de codificação final
1	0	1/2
1	1	1
1/2	0	1/4
1/2	1	1/2
1/4	0	1/4
1/4	1	1/2

2.6.1.2 RDA baseado em RTCP

O RTCP tem por objetivo fornecer informações sobre a qualidade de serviço obtidas na distribuição de dados RTP, conseguindo isto pelas transmissões periódicas de pacotes de controle a todos os participantes da sessão RTP, utilizando

o mesmo mecanismo de distribuição do RTP (unicast ou multicast), e possuindo uma porta específica de controle de sessão. Suas funções são [51]:

- Fornecer informação sobre a qualidade de serviço obtida na distribuição de dados RTP. Exemplos de utilização são: controle na taxa de codificação (algoritmo de compactação dependendo da qualidade) e diagnóstico de problemas na rede.
- Enviar o nome canônico (CNAME) do transmissor dos dados, utilizado para que todos saibam quem originou a transmissão.
- Controle da periodicidade de envio dos pacotes RTCP, a fim de permitir a escalabilidade do sistema;
- Função opcional para permitir o transporte de informações mínimas de controle.

Os principais tipos de pacotes utilizados pelo RTCP são: SR (Sender Report), RR (*Receiver Report*), SDES (*Source Description*), BYE e APP (funções específicas da aplicação). No apêndice B são apresentados os campos dos pacotes SR e RR.

O protocolo RTCP gera pacotes de controle e isto poderia sobrecarregar a rede numa sessão com um grande número de participantes. Segundo a RFC 1889, o tráfego das mensagens do RTCP deveria estar em torno de 5% do tráfego total de determinada sessão RTP, correspondendo 25% deste tráfego (1,25% do total) aos transmissores (pacotes SR+RR+SDES) e os 75% restantes (3,75% do total) aos receptores (pacotes RR). Em [42], [43], [52] utilizam as informações do RTCP para controlar algum mecanismo que melhore a qualidade do sistema.

A geração de consideráveis pacotes RTCP é a grande desvantagem que este protocolo tem pelo uso do canal de transmissão.

Neste trabalho, propõe-se também usar a informação do receptor para enviar ao emissor o estado da qualidade do sinal neste ponto, mas neste caso, só é enviado o índice MOS que é uma informação muito menor que os pacotes gerados pelo RTCP, assegurando um melhor uso do canal.

2.6.2 Mecanismos de melhora da qualidade de uma rede IP

Abaixo, serão citados os estudos relacionados com a melhora da qualidade em uma rede IP, que podem ser associados com o RDA proposto.

2.6.2.1 *Playout Buffer*

O objetivo dos mecanismos de playout buffer (também conhecido como de-jitter, jitter buffer ou playout scheduler) é minimizar os efeitos da variação do atraso na qualidade do sinal de voz no receptor. Este mecanismo mantém as mensagens em um buffer, de forma que os pacotes não são apresentados imediatamente após a sua chegada e sim são mantidos até que seu playout deadline seja atingido. Ao armazenar os pacotes por um determinado tempo, introduz-se um certo atraso, mas a vantagem é permitir que pacotes que demoram mais tempo que o previsto ou que cheguem em diferentes ordens sejam utilizados, conseguindo que os pacotes possam ser apresentados a uma taxa constante, apesar de ter chegado ao receptor com taxas variáveis devido aos atrasos adicionados pela rede.

A utilização de um playout buffer adaptativo é recomendada no caso da internet porque o atraso de propagação não é conhecido e a distribuição do atraso fim a fim dos pacotes tampouco é conhecida [53] e pode mudar em um período de tempo relativamente curto. Alguns métodos que modificam o tamanho do playout buffer são apresentados em [54] e [53].

Quando o atraso fim a fim está maior ou mais próximo do tempo permitido para conversações bidirecionais, que é aproximadamente de 150 ms [38], recomenda-se a utilização de um buffer adaptativo mediante um algoritmo que tem como função manter o atraso adicionado próximo do mínimo possível.

2.6.2.2 *FEC - forward error correction*

A capacidade do receptor para detectar e corrigir erros é conhecida como correção de erros de repasse (*forward error correction* - FEC) [55]. As técnicas FEC são valiosas porque podem diminuir o número de retransmissões exigidas do transmissor. O mais importante é que elas permitem imediata correção de erros no receptor. Isso evita a espera pelo atraso de propagação da viagem de ida e volta que o remetente precisa aguardar para receber um pacote *Not Acknowledged* (NAK) e para que um pacote retransmitido se propague de volta ao receptor, uma vantagem potencialmente importante para aplicações de rede em tempo real [56].

A FEC trabalha construindo pacotes que incluem dois tipos de dados adicionais - detecção de erro de informação e dados redundantes que podem ser utilizados para preencher os espaços em branco quando os erros são encontrados. A quantidade dos bits adicionais necessários para o FEC depende do potencial

de ruído do meio de transmissão [57].

Com a finalidade de esclarecer a utilização do FEC, a Tabela 2.13 [58] apresenta um pré-determinado algoritmo, o mais simples possível e não efetivo em uma aplicação real, onde se incluem bits redundantes no lado do transmissor e mostra como o receptor deve interpretar esta informação. Assim, neste exemplo o receptor determinará que a informação enviada é 1 quando ao menos dois dos três bits são 1.

Tabela 2.13: Exemplo da funcionalidade do FEC

Bits Transmitidos	Tradução no Receptor
000	0
001	0
010	0
100	0
111	1
110	1
101	1
011	1

Em [59] apresenta-se um mecanismo que utiliza o FEC, controlando dinamicamente a quantidade de informação adicional que se envia, otimizando o uso do canal de transmissão.

Em [60] explica-se como o FEC é incorporado na tecnologia WiMAX para evitar um número excessivo de retransmissões quando o usuário está se movendo em veículos a altas velocidades.

3 Cenários de Emulação e Resultados Experimentais

Neste trabalho, implementou-se um ambiente de emulação de uma rede de transmissão de pacotes, utilizando-se diferentes ferramentas de monitoramento para a captura da informação que transita neste meio.

Todos os *softwares* empregados na montagem do cenário de testes são *open source*, sendo este cenário de fácil reprodução. Utilizando o ambiente de emulação do cenário implementado, com algumas variações que serão descritas neste capítulo, foi executada a parte experimental das duas principais contribuições deste trabalho que são:

- Predição do comportamento de um codificador de voz em diferentes cenários de redes;
- Implementação e avaliação do RDA proposto, tendo como fator de decisão a qualidade do sinal.

A seguir, são descritas as ferramentas empregadas no cenário básico de emulação e, posteriormente, serão descritos os testes realizados e os resultados obtidos para cada uma das duas principais contribuições.

3.1 Ferramentas empregadas no cenário de emulação

Abaixo são descritas as ferramentas utilizadas na implementação do cenário de emulação.

- Emulador de Rede NISTNet.

Este emulador de rede foi projetado pelo NIST (National Institute of Standard and Technology) [61] para a realização de testes de redes IP em diferentes condições, sendo possível emular vários tipos de cenários.

As funções deste software incluem:

- Limitação da taxa de transmissão;
- Implementação de atrasos de pacotes (fixos e variáveis);
- Re-ordenamento de pacotes, usualmente em simulações de *Wide Area Network* (WAN);
- Variação da taxa de perdas de pacotes;
- Duplicação de pacotes;
- Aplicação de classe de serviço (*Class of Service* (CoS)).

O NISTNet opera em um ambiente Linux, tanto no modo gráfico como por linha de comandos. Esta ferramenta é utilizada para emulação de redes Ethernet de 10 /100 Mbps ou também, no caso de se dispor de equipamentos com interfaces de alta taxa de transmissão, suporta 1000 Mbps, STM-1 ou STM-4.

Foram implementados alguns scripts para completar algumas funcionalidades que o software nativo (NISTNet) não possuía, como a possibilidade de se mudar a probabilidade de perda de pacotes durante uma mesma emulação.

- Servidor para áudio interativo: O software utilizado foi o softphone Myphone [62] que realiza comunicações diretas entre dois PCs sem servidores intermediários. É um software cliente VoIP de código aberto, desenvolvido pelo grupo OpenH323 da ITU-T. Este softphone permite que o usuário faça a escolha do codificador de voz que será utilizado na comunicação. Suporta a recomendação ITU-T H.323 para o estabelecimento, manutenção e finalização da chamada.

O Myphone também permite a transmissão de vídeo, mas para os testes deste trabalho, esta opção foi desativada. Com a utilização do analisador de pacotes de rede, Wireshark, observou-se que o Myphone utiliza o protocolo de transporte UDP.

- Analisador de Rede Wireshark.

O Wireshark 0.99.7 System [63] (nova versão do Ethereal) é um software freeware que suporta uma grande quantidade de protocolos, entre suas características podendo-se destacar:

- Os dados que trafegam na rede podem ser observados em tempo real ou podem ser capturados em arquivos para sua posterior análise;

- Podem-se ler arquivos de diferentes redes;
 - Possui opções de diversos filtros de informação, como por exemplo, de endereços IP. Devido à grande quantidade de informação de análise, facilita o monitoramento de determinadas mensagens;
 - Suporta os principais protocolos de rede como: Ethernet, PPP, IEEE 802.11 e Token-Ring.
- Software de gravação de som.

Neste caso, foi utilizada a versão freeware do software: VRS Recording System [64], que permite gravar o áudio emitido pela placa de som do computador, servindo para a gravação de comunicações telefônicas realizadas entre os computadores. O software possui múltiplos canais para gravação de diferentes fontes de áudio.

Na realização dos testes, para obter uma qualidade aceitável do arquivo gravado, isolou-se a saída de áudio para o alto-falante com um material isolante (porta P2 do computador).

A Figura 3.1 apresenta o esquema básico do cenário de testes deste trabalho com a utilização de três computadores (PC1, PC2 e PC3), onde PC1 e PC3 são os pontos finais da comunicação, o PC2 é um computador que emula a rede e onde se programam as perdas de pacotes e atrasos. Neste cenário foram utilizadas outras ferramentas como: estimadores de qualidade da voz, analisadores de rede, laço de transmissão e um mecanismo de RDA, que serão descritas nesta seção.

Figura 3.1: Configuração do cenário básico de emulação

A configuração lógica de endereçamento IP para este cenário é indicada pela Figura 3.2, onde o PC2, roteador da rede, interliga as redes do PC1 e PC3.

Este cenário de emulação de rede nos permite, principalmente, realizar testes considerando:

Figura 3.2: Endereços IP configurados na rede emulada

- Atraso ponto a ponto (ms);
- Probabilidade de perda de pacotes (%);
- Variabilidade do atraso;
- Características do codificador, como a variação da taxa de codificação;
- Estimadores de qualidade de voz;
- Avaliação do RDA em diferentes condições de rede;
- Implementação do laço de transmissão para enviar parâmetros que indiquem a qualidade da comunicação.

3.2 Metodologia para prever o comportamento de um codificador de voz

Sobre o cenário de rede, efetuaram-se testes de transmissão de voz sobre pacotes em comunicações entre computadores (sem servidores intermediários), seguindo os protocolos de sinalização e transporte de uma comunicação VoIP, implementados nos softphones empregados. Os Codificadores G.711, G.723.1 e iLBC foram testados nestas comunicações. Para a avaliação da qualidade de voz transmitida no cenário experimental utilizou-se a Recomendação ITU - T P.862 (PESQ). Os principais cenários desenvolvidos consideraram as transmissões sob condições ideais e com probabilidade de perdas de pacotes e atrasos ponto a ponto. No caso da avaliação do efeito que um atraso ponto a ponto tem na qualidade da voz transmitida, complementou-se a PESQ com o E-Model. Assim, para a determinação do fator de degradação do codificador utilizou-se o resultado, índice MOS estimado pela PESQ (MOS-LQO), como entrada no E-model.

A Figura 3.3 apresenta o cenário de emulação empregado nos testes realizados.

O diagrama da Figura 3.3 é composto por uma sub-rede (PC1) conectada a outra sub-rede (PC3), e estes interconectados por um roteador GNU/Linux (PC2). Foi utilizada uma rede cabeada (par trançado) para este cenário.

Figura 3.3: Cenário de rede utilizado para a determinação do índice MOS utilizando a ferramenta PESQ

Os softwares aplicativos utilizados nos testes foram:

- PC1, PC3: computadores clientes utilizando o softphone MyPhone 0.2b10, analisador de rede Wireshark e software para gravação de áudio .wav VRS Recording System.
- PC2: computador com função de roteador utilizando o emulador de redes Nistnet [30], com o intuito de simular perda de pacotes, atrasos, variações de atrasos e largura de banda e o software ITU-T P.862 utilizado para encontrar a estimativa do índice MOS-LQO de qualidade da voz.

O som transmitido foi gerado por um reproduutor de arquivos de áudio, que no caso deste trabalho utilizou-se o formato .wav, conectado à entrada do microfone do PC1 mediante um cabo de áudio. Este arquivo tem uma duração de 8 segundos, amostrado a 8 kHz com 16 bits/amostra.

3.2.1 Procedimento de teste

O Procedimento detalhado de teste para calcular o índice MOS utilizando a ferramenta PESQ é descrito nas seguintes linhas:

- Inicia-se uma comunicação entre o PC1 e o PC3 mediante os softphones que cada computador tem instalado e que permite escolher o codificador de voz utilizado para cada ligação. Esta chamada VoIP é uma chamada de computador a computador e a sinalização utilizada foi a recomendação H.323;

- Para cada cenário o emulador de rede é configurado com os parâmetros requeridos pelo teste a se executar;
- O reprodutor de áudio transmite o arquivo (arq_orig.wav) ao PC1, onde este som é gravado (arq_orig2). Como a chamada já está ativa, a voz é codificada e transmitida para o PC3 através do PC2;
- Enquanto os dados são transmitidos, o programa Wireshark, no PC1 e no PC3, salva toda a informação da rede como: as mensagens de sinalização para estabelecimento da conexão, manutenção e finalização das chamadas, as mensagens do protocolo RTP, o tamanho médio do pacote (Bytes), número médio de pacotes transferidos por segundo e largura de banda média;
- No PC3, o áudio recebido é gravado em um arquivo (arq_deg.wav). Este último arquivo e o arquivo arq_orig2 são comparados mediante o programa PESQ, que é executado no PC2, tendo como resultado uma pontuação MOS. Foram feitos dez testes para cada cenário, tomando-se o valor médio como resultado válido de cada cenário.

É importante mencionar que, antes de realizar cada teste, verificou-se que o programa PESQ estivesse configurado corretamente. Para isto, utilizaram-se todos os arquivos de teste que vêm com este software, obtendo-se resultados satisfatórios, ou seja, os resultados obtidos foram valores idênticos aos apresentados na mesma recomendação.

Da mesma forma para os cenários de perdas de pacotes e atrasos, configurados no Nistnet, fizeram-se testes para avaliar como estes atrasos e perdas eram realizados na rede, conseguindo-se resultados satisfatórios e que conferem um grau de confiabilidade ao sistema.

3.2.2 Resultados dos testes da estimação da qualidade da voz

Em seguida serão apresentados os resultados dos testes realizados nos diferentes cenários experimentais considerados.

3.2.2.1 Cenário ideal

Para um cenário ideal, sem perdas, sem atrasos e com largura de banda maior que a largura de banda nominal de cada codificador, conseguiram-se resultados

semelhantes aos relatados na literatura [65], [50] e [66].

Fizeram-se 15 testes para cada codificador utilizando-se a ferramenta PESQ obtendo-se em cada um destes testes uma estimativa MOS (MOS-LQO). A Tabela 3.1, apresenta as médias dos valores obtidos em cada teste.

Tabela 3.1: Índices MOS-LQO médios para codificadores em um cenário ideal

Codificador	Pontuação PESQ
G.711	4.09
G.723.1 (5.3)	3.52
iLBC (15.2)	3.79

3.2.2.2 Cenário com atrasos fixos ponto a ponto

Com a metodologia apresentada na seção 3.2.1, realizaram-se os testes com diferentes atrasos (50, 150, 300 e 500 ms) com os mesmos codificadores já utilizados, verificando-se que o índice MOS-LQO da PESQ não se modificou de uma maneira significativa. Isto era de se esperar, pois a PESQ foi desenvolvida para comunicações unidirecionais.

O gráfico da Figura 3.4 ilustra o atraso fim a fim e a pontuação MOS-LQO obtida pela PESQ para o codificador G.723.1.

Figura 3.4: Relação MOS-LQO vs. Atraso fim a fim para codificador G.723

A Figura 3.4 não representa a verdadeira qualidade do sinal de voz, como já foi mencionado anteriormente. Para uma melhor estimativa do atraso em uma comunicação interativa, utilizou-se o modelo matemático E-Model. A Recomendação do E-Model é utilizada neste trabalho somente nos cenários que apresentem

algum atraso ponto a ponto, considerando que ela tem recebido comentários desfavoráveis como em [67].

Para condições com atraso ponto-a-ponto não-nulo, fizeram-se provas experimentais com os codificadores de voz G.723.1 e iLBC, em um cenário de rede emulada e utilizando a ferramenta PESQ para estimar o índice MOS-LQO em diferentes cenários de probabilidade de perda de pacotes, incluindo o cenário sem perdas de pacotes. Com estes valores obtidos e empregando as equações do E-Model que relacionam o valor do índice MOS com o valor de determinação do índice de transmissão R , pode-se calcular o parâmetro I_e (Fator de degradação) do codificador em estudo.

Nestes testes, obtiveram-se resultados satisfatórios para o fator I_e do codificador G723.1, ou seja, o valor I_e obtido é muito próximo ao valor estabelecido na Rec. ITU-T G.113, oferecendo uma confiabilidade para o valor do I_e calculado do codificador iLBC seguindo a mesma metodologia.

Na Figura 3.5, apresenta-se um diagrama de blocos, representando o procedimento seguido para determinar o fator de degradação do codificador em estudo.

Figura 3.5: Procedimento para determinação do fator de degradação do codificador

Do diagrama de blocos da Figura 3.5, pode-se observar que o procedimento pode se dividir em dois pontos principais: obtenção do índice MOS-LQO utilizando a PESQ e obtenção do fator I_e utilizando o E-Model.

Seguindo a sequência para o cálculo de I_e , a Figura 3.6 apresenta o diagrama de blocos deste procedimento.

Figura 3.6: Determinação do índice I_e

Com a equação (2.8), apresentada no item 2.2.1.6, pode-se calcular o valor de R correspondente ao MOS-LQO calculado no item anterior. Com a equação (2.7) calcula-se diretamente o valor de I_e , considerando: $R_o = 86,5$ e $I_d = 0$.

Com estas considerações obtiveram-se os resultados apresentados na Tabela 3.2.

Da tabela 3.2, pode-se observar que o valor do I_e , obtido para o codificador iLBC, apresenta um melhor resultado, onde um menor valor de degradação (I_e) reflete em uma maior qualidade do sinal.

Tabela 3.2: Valores médios MOS-LQO e I_e para os codificadores G.723.1 e iLBC

codificador e taxa (kbit/s)	MOS-LQO	I_e
G.723.1 (5,3)	3,52	18
iLBC (15,2)	3,79	13

É importante ressaltar que o valor obtido para o $I_e = 18$ do codificador G.723.1 é muito próximo ao valor apresentado pela ITU-T no anexo I da Rec. G.113, que é igual a 19.

Para se obter resultados sensíveis ao atraso em uma comunicação bidirecional, utilizou-se a equação (2.2) do E-model, fazendo variar o parâmetro T_a e, por consequência, o valor do I_d utilizando as equações (2.5) e (2.6). Os valores de atrasos considerados para T_a foram de 0 a 500 ms com passo de 50 ms.

Da Figura 3.7, pode-se conferir que a qualidade de voz, expressa em MOS depois de converter o valor R , se conserva aceitável até 150 ms ou 200 ms, que é o atraso aceitável especificado pela Recomendação ITU-T G.114 [38].

Figura 3.7: Relação MOS vs. Atraso ponto a ponto em uma comunicação bidirecional

3.2.2.3 Cenário com probabilidade de perdas de pacotes

Para um cenário de probabilidade de perdas de pacotes e atrasos iguais a zero, e seguindo a metodologia explicada, mediu-se a qualidade de voz para os codificadores: G.723.1 e iLBC.

As probabilidades de perdas consideradas nos testes realizados foram de 5%, 10% e 20%.

Os resultados obtidos são apresentados na Tabela 3.3:

Tabela 3.3: Valores MOS-LQO para os codificadores G.723.1 e iLBC em um cenário com probabilidade de perda de pacotes

	Codificador G.723.1			Codificador iLBC		
	Probabilidade de perda de pacotes			Probabilidade de perda de pacotes		
	5%	10%	20%	5%	10%	20%
MOS	2,85	2,60	2,14	3,26	3,02	2,61
I_e	37	44	60	26	32	44

A utilidade de fazer uma predição do comportamento de um codificador é ajudar o projetista na escolha do codificador mais adequado para uma rede com características determinadas. A tabela 3.4 apresenta um exemplo de como pode se comparar o comportamento de codificadores para diferentes condições de rede.

Tabela 3.4: Exemplo de predição do comportamento de codificador em diferentes cenários de rede

Atraso	100 ms				200 ms				400 ms			
Perda de pacotes	1%	5%	10%	15%	1%	5%	10%	15%	1%	5%	10%	15%
Codificador A	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}
Codificador B	B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}	B_{11}	B_{12}
Codificador N	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}

Estes valores são calculados para qualquer codificador, incluindo os codificadores novos, desde que se tenha o código fonte, seguindo a metodologia descrita acima.

3.3 Avaliação de um RDA baseado na qualidade do sinal

O estudo da seção 3.2 é útil na etapa do planejamento de um projeto, já que a metodologia apresentada pode ser empregada como uma ferramenta de dimensionamento de rede ou de otimização do uso do canal de transmissão, desde que o projetista de rede possa escolher o codificador mais adequado para uma rede que tem um comportamento conhecido, ou seja, uma rede que tenha seus parâmetros da rede definidos com baixa tolerância. Mas em uma comunicação, os parâmetros da rede podem mudar aleatoriamente por alguns períodos de tempo, o que pode causar um grande impacto na qualidade da comunicação.

Esta seção aborda um sistema que é capaz de controlar a qualidade do sinal transmitido durante uma comunicação de voz em uma rede de pacotes, otimizando o uso do canal de transmissão, mediante o emprego de um algoritmo de determinação de taxa de codificação de um codificador multitaxa. Este algoritmo possui como fator de decisão a qualidade do sinal num ponto da comunicação. Este processo é totalmente automático e tem como finalidade ser imperceptível do ponto de vista dos participantes da comunicação.

A ITU-T na Recomendação G.191 [68] apresenta os principais blocos funcionais dos codificadores: G.711, G.722, G.726 e G.727, ajudando o entendimento do funcionamento geral dos algoritmos dos codificadores. Os principais blocos funcionais que são abordados nesta recomendação são: os filtros que são implementados nos codificadores, os algoritmos de codificação e decodificação em forma individual, algoritmos correspondentes a variadores de taxa para codificadores multitaxa, assim como ferramentas que modelam inserção de erros as

quais são úteis para avaliação do comportamento dos sistemas de transmissão digital sob condições de erro.

Nestes testes, utilizou-se como estimador da qualidade do sinal de voz a recomendação ITU-T P.563, tendo como resultado o índice MOS-LQO da qualidade do sinal, o qual é transmitido utilizando o laço de transmissão através da mesma rede. Este mecanismo de transmissão será explicado em detalhes no item 3.4, ao final deste capítulo, em forma isolada para um melhor entendimento do leitor.

3.3.1 Cenário de emulação

O cenário apresentado na Figura 3.8 mostra como foram realizados os primeiros testes para se determinar o correto funcionamento do RDA, baseado na qualidade da voz transmitida. Neste diagrama intervêm todos os processos que compõem o sistema de controle de qualidade.

Figura 3.8: Cenário de emulação para avaliação do RDA

No esquema da Figura 3.8 apresentam-se os detalhes do comutador de taxa do codificador multitaxa, abrindo os blocos apresentados como PC1 e PC3 no diagrama geral da Figura 3.1, onde o emulador de rede é representado como PC2.

O algoritmo simplificado do mecanismo RDA que tem como fator de decisão a qualidade do sinal, esquematizado na figura 3.8, tem as seguintes etapas:

- Inicialização e programação dos parâmetros de rede no NISNet;
- Estabelecimento da taxa inicial do codificador = 40 kbit/s;
- Reprodução do arquivo ArqT_1. Este processo é realizado manualmente.
- Codificação e transmissão do sinal pela rede (o canal de transmissão emulado) através do softphone empregado;

- Decodificação do sinal transmitido e armazenamento em um arquivo (ArqR_1), o software de gravação, salva arquivos a cada 8 segundos (tempo que pode ser configurável)
- Estimação da qualidade do sinal de voz recebido utilizando o P.563, resultando num índice MOS-LQO.
- Transmitir este índice MOS-LQO desde o receptor até a fonte utilizando o laço de transmissão;
- Determinação da nova taxa pelo RDA, tendo como fator de decisão o índice MOS-LQO recebido;
- Codificar o seguinte arquivo com a nova taxa determinada pelo RDA. Para isso utilizou-se um programa adicional, variador-taxa-softphone, apresentado no Apêndice A com a que se pode comutar a taxa de codificação no softphone.

Na realização destes testes foram utilizados arquivos de áudio de duração de 8 segundos que foram extraídos da Recomendação ITU-T P.862.

É importante mencionar que no código fonte do NISTNet se inseriu um código adicional para ter diferentes valores dos parâmetros da rede em uma mesma simulação, já que no software padrão só se podem atribuir valores fixos para toda uma emulação. Assim, por exemplo, com este código adicional pode-se programar que em uma emulação de 8 segundos a probabilidade de perda de pacotes varie a cada 2 segundos.

Também cabe destacar que o código P563.c foi modificado para reduzir a saída dele a só um valor numérico.

O código adicional inserido no NISTNet e a modificação do código do P563.c são detalhados no Apêndice A deste trabalho.

Para a determinação da nova taxa de codificação (I) em função do MOS (M), foram definidos valores limiares de MOS para cada faixa correspondente a uma taxa de codificação. Estes valores são baseados nos valores MOS obtidos para cada uma das taxas individuais, os quais são apresentados na Tabela 3.7. Pode-se notar que os valores limiares MOS são valores menores em relação aos valores MOS da tabela 3.7, já que estes últimos são obtidos em condições de rede ideal. No algoritmo, para o codificador G.726, utilizaram-se os seguintes valores limiares de MOS:

- $M \geq 3,7$ então $I = 40$ kbit/s;
- $M < 3,7$ e $M \geq 3,2$ então $I = 32$ kbit/s;
- $M < 3,2$ e $M \geq 3,0$ então $I = 24$ kbit/s;
- $M < 3,0$ então $I = 16$ kbit/s.

A fim de esclarecer este processo de controle automático de determinação da taxa de codificação dependente da qualidade do sinal de voz que por sua vez depende do estado da rede, na Tabela 3.5 são apresentados os resultados de um dos testes realizados com o canal de transmissão emulado configurado para mudar a probabilidade de perda de pacotes (PLR) durante uma mesma sessão.

Tabela 3.5: Comportamento do RDA em um cenário com perda de pacotes - codificador G.726

	Arq1	Arq2	Arq3	Arq4	Arq5
Taxa (kbit/s)	40	40	32	40	32
PLR	0	8	0	8	0
MOS-LQO	3,98	3,28	3,86	3,21	3,87

O valor inicial da taxa de codificação foi de 40 kbit/s, tendo um índice MOS-LQO de 3,98. Em seguida, iniciou-se um período com probabilidade de perda de pacotes igual a 8 %. Com isto o MOS-LQO decresce para 3,28, que leva o RDA a comutar a taxa para 32 kbit/s. Em piores condições de rede, o RDA determina uma taxa de codificação menor, otimizando o canal de transmissão. Assim, neste caso, mais usuários podem ingressar ao sistema de comunicação.

Em seguida, iniciou-se um período com a rede na configuração ideal atingindo um valor próximo (3,86) ao valor máximo de qualidade para a taxa de 32 kbit/s que é um índice MOS-LQO de 3,89. Com este valor de entrada no RDA, ele comutou para a taxa de 40 kbit/s. Neste caso pode-se verificar que, quando se tem uma melhora nas condições da rede, o RDA determina uma taxa de codificação maior, garantindo uma melhor qualidade da comunicação.

Pode-se observar que, quando a rede está em condições ideais e a taxa de codificação é de 32 kbit/s, o índice MOS-LQO não é sempre o mesmo. Isto atribuiu-se à existência de um atraso ocorrido pela execução do código adicional implementado neste trabalho, inserido no código base do NISTNet.

Da mesma forma, para o caso dos diferentes valores MOS-LQO obtidos para a taxa de 40 kbit/s e PLR de 8%, é verificado que a quantidade de pacotes

perdidos varia para cada simulação, apesar de utilizar o mesmo arquivo de áudio. Também deve-se considerar o estudo em [37] que considera que a qualidade do sinal transmitido varia de acordo com a importância que o pacote perdido tinha no segmento de fala transmitido.

Os valores MOS-LQO obtidos podem ser armazenados em um arquivo para uma posterior análise do desempenho da rede e geração de logs de qualidade do sistema. O código que realiza esta funcionalidade está descrito no Apêndice A.

A Tabela 3.6 apresenta um exemplo de relatório.

Tabela 3.6: Exemplo de relatório de qualidade da rede.

Data	Hora	MOS
05/02/2009	17:45:08	3,85
05/02/2009	17:45:16	3,33
05/02/2009	17:45:24	3,13
05/02/2009	17:45:32	3,41
05/02/2009	17:45:40	3,78
05/02/2009	17:45:48	3,32

Como a realização dos testes no cenário descrito acima é bastante complexa, por depender de processos manuais, e sincronização de diferentes programas por simplicidade, propõe-se o cenário apresentado na Figura 3.9 para avaliação do RDA em condições em que a qualidade do sinal é muito variável. O objetivo destes testes é a determinação da sensibilidade deste mecanismo de controle em intervalos pequenos de mudança de taxa de codificação.

Figura 3.9: Cenário simplificado de emulação para avaliação do RDA

Note-se que para este cenário o teste pode ser realizado empregando só um arquivo de áudio de entrada, sem a necessidade de programar o gravador de áudio e evitando todo processo manual no momento de tocar o áudio. Tampouco é preciso programar o emulador de rede NISTNet.

O arquivo de texto denominado lista_MOS contém diferentes valores de índice MOS onde cada um destes valores representa a qualidade do sinal no ponto de recepção e um determinado instante. Assim, estes valores MOS representam as degradações da qualidade devido às perdas no canal de transmissão. Pode-se observar que na Figura 3.9 já não aparece o emulador de rede, pois os efeitos de degradação causados pelo emulador estão representados no arquivo lista_MOS.

Os valores MOS do arquivo lista_MOS foram escolhidos considerando a faixa de decisão do RDA para uma taxa de codificação desejada.

3.3.2 Resultados obtidos na avaliação do RDA

Utilizando o cenário de teste da Figura 3.9 e considerando-se em todos os casos uma rede ideal, sem perdas de pacotes nem atrasos, foram configurados diferentes cenários de testes, a saber: de taxa única, taxa dupla e tri-taxa. Todos estes cenários foram avaliados para os codificadores multitaxa: ITU-T G.726 e Speex

- Resultados do RDA com taxa única.

Neste cenário não há a intervenção do RDA, ou seja, só é empregada uma única taxa de codificação.

A Tabela 3.7 apresenta os valores de índice MOS-LQO e desvio padrão para cada taxa de codificação.

Tabela 3.7: Valores MOS-LQO com taxas únicas para os codificadores G.726 e Speex

Speex		G.726	
Taxa (kbit/s)	MOS-LQO	Taxa (kbit/s)	MOS-LQO
24,6	3,95	40	3,98
18,2	3,91	32	3,91
15	3,81	24	3,72
11	3,71	16	3,19
8	3,45		
5,15	3,24		

- Resultados do emprego do RDA com duas e três taxas de codificação.

O sistema do RDA é programado segundo os valores MOS do arquivo lista_MOS para utilizar duas e três taxas de codificação.

A Tabela 3.8 e 3.9 apresenta os valores MOS-LQO obtidos para o caso em que se empregaram duas e três taxas para os codificadores G.726 e Speex.

Tabela 3.8: Valores MOS-LQO obtidos para cenários de 2 e 3 taxas do codificador G.726 (16, 24, 32 e 40 kbit/s)

Rate (kbit/s)			MOS-LQO	Desvio Padrão
Taxa1	Taxa2	Taxa3		
40	32	24	3,909	0,0032
40	32	16	3,362	0,0029
40	24	16	3,315	0,0037
32	24	16	3,299	0,0034
40	32	-	3,973	0,0022
40	24	-	3,948	0,0024
40	16	-	3,926	0,0029
32	24	-	3,822	0,0022
32	16	-	3,809	0,0027
24	16	-	3,497	0,0031

Tabela 3.9: Valores MOS-LQO obtidos para cenários de 2 e 3 taxas do codificador *Speex* (5, 8, 15 e 18,2 kbit/s)

Rate (kbit/s)			MOS-LQO	Desvio Padrão
Taxa1	Taxa2	Taxa3		
18,2	8	5,15	3,32	0,0039
18,2	8	-	3,719	0,0021
18,2	5,15	-	3,534	0,0029
8	5,15	-	3,29	0,0026

Nota-se, que quando são definidas diferentes taxas de codificação, o valor obtido é um valor que está dentro do intervalo dos valores MOS-LQO obtidos para cada taxa de codificação individual.

Assim, conforme a Tabela 3.7, para a taxa de 32 kbit/s temos um MOS de 3,91 e para a taxa de 24 kbit/s temos um MOS-LQO de 3,72. Quando estas duas taxas ocorrem conjuntamente o MOS-LQO determinado pelo RDA é 3,822 como pode-se observar na Tabela 3.8.

No caso de uma rede que tenha mudanças de seus parâmetros em períodos muito curtos, a qualidade da voz transmitida varia rapidamente, fazendo com que o RDA intervenha e modifique a taxa de codificação em várias oportunidades.

Como isto pode afetar a qualidade final da comunicação, para avaliar o impacto que este processo tem, fizeram-se testes para diferentes períodos de leitura do valor MOS, conseqüente à mudança de taxa controlada pelo arquivo lista_MOS da Figura 3.9. Assim, no arquivo lista_MOS colocaram-se valores MOS que exigiam uma mudança de taxa de codificação segundo o algoritmo do

RDA.

A Figura 3.10 apresenta valores MOS-LQO para mudanças de taxas de codificação em diferentes períodos de tempo, de 0,25 até 4 segundos. As taxas consideradas neste teste foram de 40 kbit/s e 32 kbit/s. O arquivo de áudio empregado neste teste tem uma duração de 8 segundos.

Figura 3.10: Valores MOS-LQO para diferentes períodos entre comutações da taxa de codificação

Nota-se que, se a variação de taxa é executada com períodos menores que 1 segundo, a qualidade do sinal de voz cai rapidamente. Considerando esta observação, todos os testes realizados nos cenários de duas e três taxas de codificação foram realizados com um período de comutação de taxa de 1,5 segundos, que pode ser efetuado executando a leitura dos valores MOS do arquivo `lista_MOS` neste período. Note-se que o período da leitura dos valores MOS não é necessariamente o período da comutação da taxa de codificação, pois poderia ocorrer que dois valores MOS lidos consecutivamente, a cada 1,5 segundos, do arquivo `lista_MOS` estivessem dentro do mesmo intervalo de valores que corresponde a uma determinada taxa de codificação, assim o RDA não mudaria de taxa.

3.3.3 Avaliação do RDA em uma rede com atrasos ponto a ponto e perda de pacotes

A Figura 3.11 apresenta o cenário de emulação empregado nos testes realizados onde foram configurados, no emulador de rede, diferentes atrasos ponto a ponto durante uma mesma simulação.

Para determinar como o atraso ponto a ponto afeta a qualidade do sinal transmitido, complementou-se o valor MOS-LQO obtido pelo P.563 com as equações do E-Model, seguindo os seguintes passos:

- Transformação do valor MOS-LQO num valor R mediante a equação 2.8,

Figura 3.11: Cenário de emulação

sendo que este valor de R ainda não considera a degradação da qualidade do sinal devida ao atraso.

- Os atrasos, configurados no emulador de rede (NISTNet), são transformados para o fator de degradação devido ao atraso I_d mediante as equações 2.5 e 2.6.
- Do valor R encontrado inicialmente se subtrai o fator de degradação devido ao atraso I_d encontrando-se R' que representa a qualidade do sinal recebido considerando as degradações devido aos atrasos de rede.

Para validar a performance do sistema de controle proposto neste trabalho, baseado no RDA, foram testados cenários que incluem atraso e perda de pacotes, tendo como resultados os valores apresentados na Tabela 3.10. Estes resultados correspondem ao cenário onde se empregou o codificador G.726.

Tabela 3.10: Comportamento do RDA em um cenário com perda de pacotes e atraso para o codificador G.726.

	Arq1	Arq2	Arq3	Arq4	Arq5	Arq6
Taxa(kbit/s)	40	40	32	24	32	40
PLR (%)	0	8	0	4	0	2
Atraso (ms)	150	50	300	0	100	200
MOS-LQO	3,93	3,33	3,15	3,42	3,88	3,41

Para simplificar o processamento no sistema de controle os limiares de decisão do RDA foram expressos em função de R . Assim para o codificador G.726, o novo RDA tem-se:

- $R \geq 71,51$ então $I = 40$ kbit/s;
- $R < 71,51$ e $R \geq 61,2$ então $I = 32$ kbit/s;
- $R < 61,2$ e $R \geq 57,72$ então $I = 24$ kbit/s;
- $R < 57,72$ então $I = 16$ kbit/s.

Com a utilização do RDA, pode-se liberar uma determinada quantidade de largura de banda, dependendo das taxas utilizadas (para o caso das taxas testadas com o codificador Speex varia de 3 a 13%), conseguindo que mais usuários possam utilizar o sistema de comunicação. A qualidade mínima de prestação de serviço pode ser controlada pelo administrador da rede através da configuração do RDA. Assim, por exemplo, se o serviço de qualidade que oferece o codificador Speex na taxa 5,15 kbit/s não atender requisitos mínimos, ela pode ser suprimida.

Tabela 3.11: Otimização da largura de banda com o emprego do RDA.

Taxa de Codificação (kbit/s)		Taxa de Tx (kbit/s)		Taxa de Tx média com RDA (kbit/s)	Largura de Banda Liberada (%)
Taxa1	Taxa2	Taxa1	Taxa2		
18,20	18,20	49,40	49,40	49,40	0
18,20	8,00	49,40	39,20	44,30	10
18,20	5,15	49,40	36,35	42,88	13
8,00	8,00	39,20	39,20	39,20	0
8,00	5,15	39,20	36,35	37,78	4
5,15	5,15	36,35	36,35	36,35	0

A Tabela 3.11 apresenta valores de porcentagem de largura de banda liberada com o emprego do RDA e o codificador Speex.

A taxa de transmissão é calculada pela equação 2.9.

3.4 Implementação do laço de transmissão

O laço de transmissão implementado neste trabalho trata-se de uma rotina executada a cada 1,5 segundos. A frequência com que a rotina é executada depende do cenário escolhido, podendo ser programada.

Essa rotina possui como objetivo enviar a informação correspondente à qualidade da voz, valor do MOS calculado, de uma forma síncrona de 1,5 em 1,5 segundos, desde o transmissor até o receptor. O índice de qualidade é enviado pelo mecanismo de socket [69], [70] através de uma comunicação UDP.

A seguir serão definidos, brevemente, alguns conceitos básicos de socket, assim como o seu funcionamento.

3.4.1 Socket

Um socket pode ser definido como uma comunicação bidirecional entre dois programas ou computadores para troca de informações.

No socket devem estar presentes as seguintes informações:

- Endereço IP do servidor;
- Porta onde se encontra o serviço solicitado;
- Endereço IP do cliente;
- Porta através da qual o cliente solicita o serviço.

Assim, pode-se dizer que um socket, em um computador, é definido como a combinação de um endereço IP, um protocolo, e o número da porta do protocolo.

Sockets não podem ser usados para acessar camadas de rede superiores ou inferiores; por exemplo, uma aplicação socket não tem conhecimento se está rolando em uma rede ethernet, token ring, 802.11b ou uma conexão dialup.

Na camada de transporte, o socket suporta 2 protocolos: TCP (Transmission Control Protocol) e UDP (User Datagram Protocol), podendo seu posicionamento na camada OSI ser observado conforme a Figura 3.12.

Quando uma aplicação socket é programada, pode-se escolher entre usar uma comunicação TCP ou UDP.

Em uma comunicação TCP, protocolo de fluxos, existe uma conexão aberta contínua entre o cliente e o servidor, sobre a qual bytes podem ser escritos e corrigidos com garantia de entrega enquanto a conexão existir.

Em uma comunicação UDP, protocolo de datagramas, não se requer que qualquer outra conexão entre cliente e servidor seja estabelecida, simplesmente se transmite uma mensagem entre endereços. Uma boa característica do UDP é que seus pacotes são auto-delimitadores, indicando cada datagrama exatamente onde é o seu início e fim. O UDP, contudo, não garante que seus pacotes cheguem em ordem. Os protocolos de alto nível construídos acima do UDP devem, se

Figura 3.12: Alocação do socket do RDA entre as camadas OSI

necessário, refinar as mensagens de *handshaking*¹ e *acknowledgements*².

No socket, atrás do protocolo, TCP ou UDP, existe um par de máquinas (cliente e servidor) que se comunica através de um endereço IP e uma porta.

Em vez de endereços IP, também podem ser utilizados nomes, mas deve-se ter o cuidado de se usar um servidor de nomes (DNS) para a tradução entre nomes e endereços.

Neste trabalho, utilizou-se o protocolo de transporte UDP pela sua simplicidade de implementação e rapidez na comunicação entre cliente e servidor.

3.4.1.1 Comunicação entre cliente e servidor

A comunicação entre computadores (cliente e servidor) ocorre conforme a Figura 3.13, onde o servidor recebe as conexões socket enviadas pelo cliente através de um determinado endereço IP e porta.

Resume-se o processo em:

¹processo pelo qual duas máquinas afirmam, uma a outra, que se reconhecem e estão prontas para iniciar a comunicação.

²mensagem em que o receptor diz, explicitamente, ao emissor que recebeu um pacote sem erros.

Figura 3.13: Mensagem enviada do cliente ao servidor por socket

- Máquina Cliente:
 - Cria um socket do tipo datagrama;
 - Envia mensagem ao destinatário (servidor), indicando o IP e porta do servidor;
 - O cliente pode receber mensagens enviadas pelo servidor;
 - No final da comunicação o cliente pode fechar o seu socket.
- Máquina Servidora:
 - Cria um socket e lhe atribui um endereço, conhecido pelos clientes;
 - Recebe as mensagens enviadas pelos clientes;
 - Envia mensagens (respostas) aos clientes;
 - Caso não seja mais necessária a troca de mensagens, fecha-se o socket.

A Figura 3.14 mostra a rotina (laço) de transmissão implementada neste trabalho, executa-se o P.563 para obter o MOS do lado da máquina cliente, o MOS é enviado para a máquina servidora por socket que, ao receber este valor MOS, seleciona a melhor taxa.

Foram implementadas neste trabalho duas redes (10.0.0.0 e 192.168.0.0) que se comunicam através de um roteador, em que a máquina servidora possui o IP 10.0.0.2 e a máquina cliente possui o IP 192.168.0.2. Definimos a porta de comunicação (7777) utilizada pelo socket com a que o servidor deve aceitar requisições em que a mensagem, valor MOS, é transmitida pelo cliente através do endereço IP (10.0.0.2).

O valor MOS enviado via `client_socket_UDP` (implementado na máquina cliente) para o `servidor_socket` (implementado na máquina servidora), Figura

Figura 3.14: Funcionamento do laço de transmissão

3.14, é armazenado em uma variável cujo valor determina o valor da taxa mais adequada (16, 24, 32 e 40 kbit/s):

- $M \geq 3,7$ então $I = 40$ kbit/s;
- $M < 3,7$ e $M \geq 3,2$ então $I = 32$ kbit/s;
- $M < 3,2$ e $M \geq 3,0$ então $I = 24$ kbit/s;
- $M < 3,0$ então $I = 16$ kbit/s.

Estes valores foram selecionados conforme os resultados obtidos para taxas de codificação individuais segundo a Tabela 3.7.

Caso as taxas não necessitem mais ser selecionadas a conexão socket é fechada.

4 Conclusão

Este trabalho foi focado em duas áreas principais, sendo as conclusões de cada uma destas áreas:

- Predição do comportamento de um codificador de voz em diferentes cenários de redes.

A metodologia apresentada na seção 3.2 pode ser de grande utilidade para o projetista de rede na avaliação e predição do comportamento do codificador em estudo, para diferentes cenários de rede, ou seja, com diferentes probabilidades de perda de pacotes e atrasos tanto em comunicações unidirecionais quanto em conversações bidirecionais. Isto é especialmente importante para codificadores novos ou que ainda não tenham sido avaliados formalmente pela ITU-T.

Pode ainda ser uma ferramenta de utilidade para a escolha do tipo de codificador mais adequado para uma rede com características conhecidas, pois, por exemplo, há codificadores que tem um melhor desempenho comparativo em cenários com perda de pacotes.

Os resultados obtidos foram bastante aceitáveis considerando que são valores muito próximos para os codificadores recomendados por esta entidade, dando um grau de confiabilidade ao método utilizado nestes testes e que podem ser facilmente reproduzidos, já que o cenário de teste foi baseado em *software open source*.

- Implementação e avaliação de um RDA, tendo como fator de decisão a qualidade do sinal.

Pode-se concluir que o sistema RDA apresentado neste trabalho otimiza a utilização do canal de transmissão, escolhendo a taxa de codificação mais adequada baseado na qualidade do sinal no ponto de recepção, podendo evitar uma sobrecarga na rede e garantindo a mais alta qualidade possível na comunicação quando as condições de rede o permitam.

O mecanismo de RDA apresentado é totalmente automático e autônomo, ou seja, o processo de controle não é percebido pelo usuário.

Em uma rede cujos parâmetros variam em períodos de tempo muito curtos a qualidade do sinal variará da mesma forma, fazendo com que o RDA comute a taxa de codificação repetidamente, acarretando uma degradação na qualidade do sinal. Realizaram-se testes com diversos valores de período de comutação obtendo-se como resultado que o índice MOS decresce abruptamente quando o período de comutação cai abaixo de 1 segundo.

Cabe ressaltar que os resultados obtidos nestes testes são confiáveis, pois quando são empregadas diferentes taxas de codificação, o valor MOS obtido é um valor que está dentro do intervalo dos valores MOS obtidos para cada taxa de codificação individual. Assim também as respostas que o RDA teve frente aos cenários de rede com probabilidade de perda de pacotes e atrasos foi satisfatória pois conseguiu manter um controle dinâmico otimizando a utilização do canal de transmissão.

4.1 Principais contribuições

Conforme mencionado na Seção 1.1, o objetivo principal deste trabalho foi apresentar um mecanismo de controle da qualidade em um sistema de comunicação de voz sobre IP baseado na qualidade do sinal transmitido. Para este propósito, tem-se que ter um conhecimento dos parâmetros da rede (perda de pacotes e atrasos) e do codificador empregado na comunicação, que conjuntamente determinam a qualidade do sinal de voz. Esta análise teve como objetivo adicional a apresentação de uma metodologia para predizer o comportamento de um codificador em diferentes condições de rede.

As principais contribuições são:

- O sistema RDA apresentado neste trabalho otimiza a utilização do canal de transmissão, escolhendo a taxa de codificação mais adequada para um cenário de rede baseado na qualidade do sinal no ponto de recepção, podendo evitar uma sobrecarga na rede e garantindo a mais alta qualidade possível na comunicação quando as condições de rede as permitam.
- Metodologia para a determinação do comportamento do codificador em estudo, para diferentes cenários de rede, ou seja, com diferentes probabilidades de perda de pacotes e atrasos tanto em comunicações unidirecionais quanto

em comunicações bidirecionais. Isto é especialmente importante para codificadores novos ou que ainda não tenham sido avaliados formalmente pela ITU-T.

4.2 Sugestões para trabalhos futuros

No transcurso deste trabalho, foram vislumbrados alguns temas relacionados que podem ser motivo de estudo, podendo ser desenvolvidos empregando o mesmo ambiente de testes.

1. O RDA apresentado é baseado na qualidade do sinal, que é um fator de decisão que pode se integrar com outros sistemas de controle como VAD, Buffer Dinâmico, análise do sinal, para compor um algoritmo RDA mais versátil.
2. No cenário de testes experimental apresentado para a avaliação do RDA devem-se efetuar mais testes com diferentes condições de redes, para medir como afeta cada variação destes parâmetros o desempenho do sistema do RDA.
3. Avaliação dos efeitos de interferência e desvanecimento do canal RF na qualidade de comunicações VoIP em uma rede WLAN.
4. Pode-se implementar um cenário sem fio, IEEE 802.16b/g e verificar como o *handoff* afeta a qualidade do sinal de voz transmitido.
5. Comparar os resultados das Recomendações ITU-T para a avaliação de Qualidade: P.862, P.563, G.107.
6. Pode-se implementar um só equipamento com uma interface amigável, que integre todo o cenário experimental correspondente à predição do comportamento do codificador em diferentes cenários de rede.
7. O cenário de emulação implementado neste trabalho pode ser empregado para avaliar codificadores de vídeo.

Referências

- [1] PAKSOY, E.; GERSHO, A. Multimode and variable-rate coding of speech, in *Speech Coding and Synthesis. IEEE International Conference on Computer Communications and Networks*, v. 7, n. 10, October 1995.
- [2] PAKSOY, E.; GERSHO, A. *Variable rate speech coding for cellular networks, in Speech and Audio Coding for Wireless and Network Applications*. [S.l.]: Kluwer Academic Publishers, 1993.
- [3] PAULUS, J. W. Variable Bitrate Wideband Speech Coding Using Perceptually Motivated Thresholds. *IEEE Workshop on Speech Coding for Telecommunications*, p. 35–36, September 1995.
- [4] HERMANSKY, H. Perceptual linear predictive (PLP) analysis of speech. *J. Acoust. Soc. Am.*, v. 87, p. 1738–1752, April 1990.
- [5] PAN, D. Y. *Digital Audio Compression*. May 1993. Disponível em: <<http://www.digital-audio.net/res/docs/pdf/Digital-Audio-Compression-01oct1993DTJA03P8.pdf>>.
- [6] MADHU, M.; MURTY, V. S.; KAMAKOTI, V. Dynamic coding technique for low-power data bus. *IEEE Computer Society Annual Symposium on VLSI*, p. 252–253, February 2003.
- [7] SHEN, G.; LIOU, M. L. A new padding technique for coding of arbitrarily-shaped image/video segments. *Proceedings of the IEEE International Conference on Computer Communications*, v. 2, p. 838–842, March 1999.
- [8] MARÍA, F. D. de. *Codificación de Voz*. June 2005. Disponível em: <www.tsc.uc3m.es/~fdiaz/docencia/IT_ATDS/Cod_Voz.pdf>.
- [9] GRIFFIN, D.; LIM, J. Multi-band Excitation Vocoder. *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. 36, n. 8, p. 1223–1235, August 1988.
- [10] HARDWICK, J.; LIM, J. The Application of the IMBE Speech Coder to Mobile Communications. *International Conference on Acoustics, Speech, and Signal processing*, p. 249–252, April 1991.
- [11] WONG, S.-W. An Evaluation of 6.4kbit/s Speech Codecs for INMARSAT-M System. *International Conference on Acoustics, Speech, and Signal processing*, v. 1, p. 629–632, April 1991.
- [12] ITU-T Rec. G.711. *General Aspects of Digital Transmission Systems Terminal Equipments - Pulse Code Modulation (PCM) of Voice Frequencies*. 1972. Disponível em: <www.itu.int/rec/T-REC-G.711/en>.

- [13] ITU-T Rec. G.726. *Terminal Equipments - 40, 32, 24, 16 kbps Adaptive Differential pulse code modulation (ADPCM)*. December 1990. Disponível em: <www.itu.int/rec/T-REC-G.726en>.
- [14] ITU-T Rec. G.728. *Terminal Equipments - Coding of Speech at 16 kbit/s Using Low-Delay Code Excited Linear Prediction*. September 1992. Disponível em: <www.itu.int/rec/T-REC-G.728/en>.
- [15] ITU-T Rec. G.729. *General Aspects of Digital Transmission Systems - Coding of Speech at 8 kbit/s Using Conjugate-Structure Algebraic-Code-Excited Linear-Prediction (CS-ACELP)*. March 1996. Disponível em: <www.itu.int/rec/T-REC-G.729/en>.
- [16] ITU-T Rec. G.723.1. *General Aspects of Digital Transmission Systems - Dual Rate Speech Coder for Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s*. May 2006. Disponível em: <www.itu.int/rec/T-REC-G.723.1/en>.
- [17] IETF RFC 3951. *Internet Low Bit Codec Rate*. December 2004. Disponível em: <www.ietf.org/rfc/rfc3951.txt>.
- [18] BASET, S.; SCHULZRINNE, H. G. An analysis of the Skype peer-to-peer internet telephony protocol. *IEEE International Conference on Computer Communications*, p. 1–11, April 2006.
- [19] VALIN, J. *The speex codec manual*. December 2007. Disponível em: <speex.org/docs/manual/speex-manual>.
- [20] 3GPP Rec. TS 26.071. *AMR speech Codec; General description*. July 1999. Disponível em: <www.3gpp.org/ftp/Specs/html-info/26-series.htm>.
- [21] RAMÍREZ, M. A.; MINAMI, M. *Technology and Standards for Low-Bit-Rate Vocoding Methods*. The Handbook of Computer Networks, vol. 2, p. 447–467, 2008.
- [22] ITU-T Rec. P.862. *Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs*. February 2001. Disponível em: <www.itu.int/rec/T-REC-P.862/en>.
- [23] ITU-T Rec. P.563. *Single-ended method for objective speech quality assessment in narrow-band telephony applications*. April 2004. Disponível em: <www.itu.int/rec/T-REC-P.563/en>.
- [24] ITU-T Rec. G.107. *The E-model: a computational model for use in transmission planning*. February 2003. Disponível em: <www.itu.int/rec/T-REC-G.107/en>.
- [25] ITU-T P. 800. *Methods for subjective determination of transmission quality*. August 1996. Disponível em: <www.itu.int/rec/T-REC-P.800/en>.
- [26] ITU-T Rec. P.861. *Objective quality measurement of telephone-band (300–3400 Hz) speech codecs*. February 1998. Disponível em: <www.itu.int/rec/T-REC-P.861/en>.

- [27] HUANG, S.-F.; CHANG, P.-C.; HSIAO-KUANG, E. Adaptive voice smoothing with optimal E-model method for VoIP services. *IEICE Transactions on Communications*, v. 6, p. 1862–1868, July 2006.
- [28] BANDUNG, Y. et al. Providing quality of service in rural next generation network (R-NGN). *International Conference on Information and Communication Technology and Systems*, v. 1, n. 2, p. 117–131, August 2008.
- [29] DOUSKALIS, B. *IP Telephony: The Integration of Robust VoIP Services*. [S.l.]: USA, Hewlet Packard Professional Books, 2003.
- [30] ROSHNI, W. C. W. et al. *IEEE 802.16 Broadband Wireless Access*. January 2006. Disponível em: <<http://ieee802.org/16>>.
- [31] ODOM, W. *CCENT/CCNA ICND1, Official Exam Certification Guide*. [S.l.]: Cisco Systems, 2007.
- [32] DAVIDSON, J. et al. *Voice Over IP Fundamentals*. [S.l.]: Cisco Systems, 2007.
- [33] ITU-T Rec. H.323. *Packet-based multimedia communications systems*. July 2006. Disponível em: <www.itu.int/rec/T-REC-H.323/en>.
- [34] BRANDL, M. et al. *IP Telephony*. 2007.
- [35] ODOM, W. *CCENT/CCNA ICND2, Official Exam Certification Guide*. [S.l.]: Cisco Systems, 2007.
- [36] IETF Rec. 3550. *A Transport Protocol for Real-Time Applications*. July 2003. Disponível em: <www.ietf.org/rfc/rfc3550.txt>.
- [37] SUN, L. F. et al. Impact of packet loss location on perceived speech quality. *Proceedings of 2nd IP-Telephony Workshop*, April 2001.
- [38] ITU-T Rec. G.114. *One way transmission time*. May 2003. Disponível em: <www.itu.int/rec/T-REC-G.114/en>.
- [39] MINOLI, D.; MINOLI, E. *Delivering voice over IP Networks*. [S.l.]: Wiley, 1999.
- [40] CASNER, S.; JACOBSON, V. *Compressing IP/UDP/RTP Headers for Low-Speed Serial Links*. 1999.
- [41] LIANG, J.; AHMAD, M. O. Implementation of a voice activity detection and comfort noise generation algorithm. *IEEE Symposium Circuit and System*, v. 1, p. 440–443, August 2005.
- [42] KOMISARCSUK, P.; KOUDRIN, A. Effect of rerouting on NGN VoIP quality. *IEEE, Australian Telecommunication Networks and Applications Conference*, p. 364–370, December 2007.
- [43] SFAIROPOULOU, A.; MACIAN, C.; BELLALTA, B. QoS Adaptation in SIP-based VoIP calls in Multi-rate IEEE 802.11 Environments. *IEEE - Wireless communications Systems*, p. 475–479, September 2006.
- [44] RABINER, L. R.; SCHAFER, R. W. *Digital Processing of Speech Signal*. [S.l.]: Prentice Hall, 1978.

- [45] RABINER, L. R.; SAMBUR, M. R. An Algorithm for Determining the End-points of Isolated Utterances. *Bell Sys, Tech. Journal*, v. 54, n. 2, p. 297–315.
- [46] HOLMA, H.; TOSKALA, A. *WCDMA for UMTS, Radio Access for Third Generation Mobile Communications*. [S.l.]: UK, John Wiley & Sons, 2004.
- [47] DELLER, J. R.; PROAKISAND, J.; HANSEN, F. Discrete-time processing of speech signals. *USA, IEEE Press*, 2000.
- [48] BERNARD, A. P. Channel adaptive joint source-channel coding of speech. *Conference on Signals, Systems and Computers*, 1998.
- [49] RAMLY, S. E.; TAWFIC, A.; MASOUD, M. A rate-determination algorithm for variable-rate speech coder. *IEEE - International Conference on Electrical, Electronic and Computer Engineering*, p. 697–701, September 2004.
- [50] NETWORKS, N. *Voice over packet, an assessment of voice performance on packet networks*. September 2001. Disponível em: <<http://www.nortel.com/products/library/collateral/74007.25-09-01.pdf>>.
- [51] IETF Rec.1889. *RTP: A Transport Protocol for Real-Time Applications*. June 2008. Disponível em: <www.ietf.org/rfc/rfc1889.txt>.
- [52] GALIOTOS, P.; DAGIUKLAS, T.; ARKADIANOS, D. QoS management for an enhanced VoIP platform using R-factor and network load estimation functionality. *5th IEEE International Conference on High Speed Networks and Multimedia Communications*, p. 305–314, July 2002.
- [53] MOON, S. B.; KUROSE, J. F.; TOWSLEY, D. F. Packet audio playout delay adjustment: performance bounds and algorithms. *Multimedia Systems*, v. 6, n. 1, February 1998.
- [54] NARBUTT, M.; MURPHY, L. VoIP Playout Buffer Adjustment Using Adaptive Estimation of Network Delays. *Proceedings of 18th International Teletraffic Congress*, p. 1171–1180.
- [55] KUROSE, J. F.; ROSS, K. W. *Redes de Computadores e a Internet - Uma nova abordagem*. São Paulo, Addison Wesley, vol. 1, p. 303, 2004.
- [56] RUBENSTEIN, D.; KUROSE, J.; TOWSLEY, D. Real-time reliable multicast using proactive forward error correction. *Proceedings of NOSSDAV*, July 1998.
- [57] WALTERS, E. G. *The essential guide to computing*. USA, Prentice Hall, vol. 1, 2001.
- [58] Wikipedia. *FEC - Forward Error Correction*. July 2009. Disponível em: <en.wikipedia.org/wiki/Forward_error_correction>.
- [59] PADHYE, C. et al. A new adaptive FEC loss control algorithm for voice over IP applications. *IEEE International Conference on Performance, Computing and Communication*, p. 307–313, 2000.
- [60] AHMAD, I.; HABIBI, D.; RAHMAN, Z. An Improved FEC Scheme for Mobile Wireless Communication at Vehicular Speeds. *Telecommunications Networks and Applications Conference, The Handbook of Computer Networks*, vol. 2, p. 312–316, December 2008.

- [61] LABORATORY, T. N. I. T. *NIST Net - A Linux-based Network Emulation Tool*. January 2008. Disponível em: <www.xantd.nist.gov/nistnet>.
- [62] MAGGUS. *MyPhone - Voice-Video IP telephony chat client*. January 2008. Disponível em: <myphone.sourceforge.net>.
- [63] LAMPING, U. *Wireshark - network protocol analyzer*. January 2008. Disponível em: <<http://www.wireshark.org>>.
- [64] NCH. *VRS Recording System*. January 2008. Disponível em: <www.nch.com.au/vrs/index.html>.
- [65] JIANG, W.; SCHULZRINNE, H. Comparisons of FEC and codec robustness on VoIP quality and bandwidth efficiency. *ICN*, August 2002.
- [66] GRANT, M.; TONISSEN, S. *Voice Quality Monitoring for VoIP Networks*.
- [67] HARDY, W. C. *VoIP Service Quality, Measuring and Evaluating Packet-Switched Voice*. [S.l.]: McGraw-Hill Networking, 2003.
- [68] ITU-T Rec. G.191. *Software tools for speech and audio coding standardization*. August 2005. Disponível em: <www.itu.int/rec/T-REC-G.191/en>.
- [69] KRISTIANSSEN, J.; PARNES, P. Application-layer mobility support for streaming real-time media. *Wireless Communications and Networking Conference*, v. 1, p. 268–273, March 2004.
- [70] BAIG, A. Protocols and sockets [internet programming]. *IEEE Potentials*, v. 18, p. 38–42, September 1999.

Apêndice A – Códigos utilizados nesta dissertação

Código adicional para P.563.c

```

1
2  /* Anula as saídas padrões do código ,
3  assim a saída é limitada a um valor ,
4  facilitando a leitura de só uma variável */
5
6
7  if (lWriteHeaderFlag)
8  {
9  # linha abaixo comentada para não imprimir a palavra Filename
10 //fprintf(hFile, "Filename");
11 # linha abaixo comentada para não imprimir a palavra MOS
12 //fprintf(hFile, "MOS\n");
13 }
14 //fprintf(hFile, "%s\t", pcProcessFilename);
15
16 fprintf(hFile, "%f\n", ptResults->fPredictedMos);
17
18 # linha abaixo comentada para não imprimir
19 informações adicionais sobre explicação de funcionamento do softw
20
21 //if (pcAdditionalInfo!=NULL) fprintf(hFile, pcAdditionalInfo);
22 //fprintf(hFile, "n");
23 if(strcmp(pcResultFilename, "stdout"))
24 {
25 fclose(hFile);
26 }
27 }

```

Código adicional para o gravador de som

```
1
2  /* Código que permite salvar cada 8
3  segundo um arquivo de áudio na saída da PC */
4
5  #!/bin/bash
6      n=1
7      cont=0
8
9  //processo que grava de 8 em 8 segundos a entrada
10 //do microfone para um arquivo
11 //pelo uso da ferramenta rec do pacote sox (softw de audio)
12
13      while ((cont==0)); do
14          file=gravacao_$(printf "%03d" $n).wav
15          rec $file -c 2 -t wav -r 44100 trim 0 8
16          let n+=1
17      done
18  }
19
20 //explicação dos parâmetros:
21 //file = nome do arquivo: gravacao_001.wav, gravacao_002.wav
22 // trim = grava 8 segundos de áudio
23 //Sample Size      : 16-bit
24 //Sample Encoding: signed (2's complement)
25 //Sample Rate      : 44100
```

Código adicional do NISTNet

```
1
2 #este é o IP fonte do tráfego
3 source_IP_addr="10.0.0.2"
4
5 #este é o IP destino do tráfego
6 dest_IP_addr="192.168.0.2"
7
8 #bw é a largura de banda e os valores
9 estão bytes/sec, porém não serão utilizados
```

```
10
11 #bw[1]=1250000
12 #bw[2]=12500
13
14 #os valores para perda de pacote e
15 estão em porcentagem (%)
16
17 // 0% de perda de pacotes
18 drop[1]=0
19 // 8% de perda de pacotes
20 drop[2]=8
21 drop[3]=0
22 drop[4]=8
23 drop[5]=0
24
25 #os valores de jitter estão em milliseconds ,
26 porém não serão utilizados
27
28 #jitter[1]=0
29 #jitter[2]=10
30
31 #echo ${temp[1]}
32 #primeiramente , acione o emulador NistNet
33 cnistnet -u
34
35 i=1
36 j=1
37
38 #Este while varia a perda de pacote %
39 while [ $j -lt 6 ]
40     do
41         cnistnet -a $source_IP_addr $dest_IP_addr --delay
42         0 --bandwidth 1250000 --drop ${drop[j]}
43
44         j=$((j+1))
45
46 #echo Pressione qualquer tecla para
47 continuar testando...
48 #read keypress
49
```

```
50 // após 2 segundos uma nova % de perda de pacotes
51 é aplicada na rede
52
53 sleep 1.92s
54 done
```

Mecanismo de controle utilizando RDA

client_socket_udp.c

```
1
2 /* Estabelece o socket no lado do cliente */
3 #include <stdio.h>                                \\
4
5 bibliotecas necessárias para o socket
6
7 #include <sys/socket.h>
8 #include <arpa/inet.h>
9 #include <stdlib.h>
10 #include <string.h>
11 #include <unistd.h>
12 #include <netinet/in.h>
13
14 #define BUFSIZE 255
15 void Die(char *mess) { perror(mess); exit(1); }
16 int main(int argc, char *argv[]) {
17     int sock;
18     struct sockaddr_in echoserver;
19     /*estrutura de criação de socket*/
20
21     struct sockaddr_in echoclient;
22     char buffer[BUFSIZE];                          /*guarda valor MOS*/
23     unsigned int echolen, clientlen;
24     int received = 0;
25
26     /* Cria o socket UDP */
27     if ((sock = socket(PF_INET, SOCK_DGRAM, IPPROTO_UDP)) < 0) {
28         Die("Falha ao criar o socket");
29     }
```

```
30
31  /* Construe no servidor o sockaddr_na estrutura */
32
33  /* Limpe struct */
34  memset(&echoserver, 0, sizeof(echoserver));
35  /* Internet/IP */
36  echoserver.sin_family = AF_INET;
37  /* endereço IP */
38  echoserver.sin_addr.s_addr = inet_addr(argv[1]);
39  /* porta do servidor */
40  echoserver.sin_port = htons(atoi(argv[3]));
41
42  /* Verifica tamanho de bytes do valor MOS */
43  echolen = strlen(argv[2]);
44  if (sendto(sock, argv[2], echolen, 0,
45            (struct sockaddr *) &echoserver,
46            sizeof(echoserver)) != echolen) {
47      Die("Os_números_de_bytes_enviados_se_diferem");
48  }
49
50  /* Imprime e Valida o valor MOS */
51  fprintf(stdout, "Recebido:");
52  clientlen = sizeof(echoclient);
53  if ((received = recvfrom(sock, buffer, BUFSIZE, 0,
54                          (struct sockaddr *) &echoclient,
55                          &clientlen)) != echolen) {
56      Die("Os_números_de_bytes_enviados_se_diferem");
57  }
58
59  /* Verifica se cliente e servidor
60   estão utilizando o mesmo socket */
61
62  if (echoserver.sin_addr.s_addr !=
63      echoclient.sin_addr.s_addr) {
64
65      Die("Recebeu_um_pacote_de_um_servidor_inesperado");
66  }
67
68  /* Verifica se o valor da string acabou */
69  buffer[received] = '0';
```

```
70
71     fprintf(stdout, buffer);
72     fprintf(stdout, "\n");
73     close(sock);
74     exit(0);
75 }
```

loop_time2.c

```
1
2  /* Chama o P.563 para achar o valor MOS, e envia este valor
3  mediante o laço de transmissão, inicializa a taxa
4  da primeira codificação */
5
6  #include<stdio.h>
7  #include<stdlib.h>
8  #include<string.h>
9  #include<time.h>
10 #include<signal.h>
11
12  int n=1000;
13
14  // Os testes finais foram realizados com n=1500
15
16  typedef void(*r)(int);
17
18  #função que de tempos em tempos executa o procedimento de cálculo do MOS
19
20  int __nsleep(const struct timespec *req, struct timespec *rem)
21
22  // função de dorme(sleep) durante um determinado
23  // tempo = n segundo
24  // (Nos testes utilizou-se n=1000 =1segundo e n=1.5 segundos)
25
26  {
27      struct timespec temp_rem;
28      if(nanosleep(req,rem)==-1)
29          __nsleep(rem,&temp_rem);
30      else
31          return 1;
```

```

32 }
33
34
35 int msleep(unsigned long milisec)
36 {
37     struct timespec req={0},rem={0};
38     time_t sec=(int)(milisec/1000);
39     milisec=milisec-(sec*1000);
40     req.tv_sec=sec;
41     req.tv_nsec=milisec*1000000L;
42     __nsleep(&req,&rem);
43     return 1;
44 }
45
46 int main(int argc, char *argv[])           // função principal
47 {
48     struct sigaction sa={0};
49     sa.sa_handler=&sigfunc;
50     sigaction(SIGINT, &sa, NULL);
51
52     int a=0;
53     float MOS = 0.0;
54     char MOS1[100];
55
56     MOS = atof(argv[1]);
57
58     FILE *fileMOS;
59     FILE *fileP563;
60     int taxa-inicial = 40;
61     r(taxa-inicial); // determina que a taxa inicial para 40 kbit/s
62
63     while (a==0)
64     {
65         \#executa o P563 para cálculo do MOS e
66         armazena no arquivo MOS_P563
67
68         system("./T-REC-563/Software/bin/p563_arquivo_saida
69 ~~~~~~>_/tmp/MOS_P563");
70
71         fileP563=fopen("/tmp/MOS_P563", "r+");

```

```

72      char MOSP563[8];
73      while( fgets (MOSP563,8 , fileP563 )!=NULL)
74      {
75          // imprime valor MOS na tela para debug
76          printf ("%s",MOSP563);
77
78      // guarda data em arquivo MOS.txt
79      system("date +%D-%T->>_MOS.txt");
80
81      // abre arquivo MOS.txt na variável file MOS
82      fileMOS=fopen("MOS.txt","a");
83      // salva valor MOS no arquivo de Gestão MOS.txt
84      fprintf(fileMOS,"%0.3f\n",MOS);
85      }
86      fclose (fileP563);
87      fclose (fileMOS);
88
89      #chama o cliente socket e envia MOS calculado por P563
90      char command[100] = "/home/demo/client_socket_udp_10.0.0.2";
91      sprintf(MOS1, "%0.3f",MOS);
92      printf("MOS1=%s",MOS1);
93      printf("command=%s",command);
94      strcat (command,MOS1);
95      strcat (command,"7778");
96      system(command);
97      msleep(n);      // dorme e reinicia o processo de
98      envio do MOS, para as condições
99
100     // da Tabela 3.5 foi configurado 8000
101     }
102     return 1;
103 }

```

server_socket_udp.c

```

1
2 /* Define o socket no servidor; executa á escolha de
3 Taxa - RDA - Segundo o valor MOS recebido;
4 traslada a taxa ao softphone */
5

```



```
6  /* Disponibilizado pelo usuário (./server_socket_udp 7778),
7  efetua a ação do RDA quando recebe o MOS, como um
8  servidor DHCP ou seja aguarda por requisições*/
9
10 #include <stdio.h>
11 #include <sys/socket.h>
12 #include <arpa/inet.h>
13 #include <stdlib.h>
14 #include <string.h>
15 #include <unistd.h>
16 #include <netinet/in.h>
17
18 #define BUFFSIZE 255
19 typedef void(*r)(int);
20 void Die(char *mess) { perror(mess); exit(1); }
21
22 int main(int argc, char *argv[]) {
23     int sock;
24     struct sockaddr_in echoserver;          // criação de estrutura socket
25     struct sockaddr_in echoclient;
26     char buffer[BUFFSIZE];
27     unsigned int echolen, clientlen, serverlen;
28     int received = 0;
29     int rate;                               # guarda valor da taxa
30     float buffer1;                          #guarda valor MOS
31     char rate1[10];
32
33     /* Cria um socket UDP */
34     if ((sock = socket(PF_INET, SOCK_DGRAM, IPPROTO_UDP)) < 0) {
35         Die("Falha_ao_criar_socket");
36     }
37     /* Constrói o sockaddr_ */
38     /* Limpe struct */
39     memset(&echoserver, 0, sizeof(echoserver));
40     /* Internet/IP */
41     echoserver.sin_family = AF_INET;
42     /* Qualquer endereço IP */
43     echoserver.sin_addr.s_addr = htonl(INADDR_ANY);
44     /* porta do servidor */
45     echoserver.sin_port = htons(atoi(argv[1]));
```

```
46
47  /* Bind socket */
48  serverlen = sizeof(echoserver);
49  if (bind(sock, (struct sockaddr *) &echoserver, serverlen) < 0) {
50      Die("Falha_bind_server_socket");
51  }
52
53  /* Execute (encontra o MOS ideal) até ser cancelado */
54  while (1) {
55      /* Recebe uma mensagem do cliente, o MOS no 'buffer' */
56      clientlen = sizeof(echoclient);
57      if ((received = recvfrom(sock, buffer, BUFSIZE, 0,
58                              (struct sockaddr *) &echoclient,
59                              &clientlen)) < 0) {
60          Die("Falha_ao_receber_mensagem");
61      }
62      else {
63          // buffer1 guarda valor MOS
64          buffer1 = atof (buffer);
65          printf("buffer1:_%f\n", buffer1);
66          //compara buffer1 (valor MOS) com rate (taxas)
67          if (buffer1 >= 3.7)   rate = 40;
68          if ((buffer1 <= 3.4) && (buffer1 >= 3.2))   rate = 32;
69          if ((buffer1 <= 3.2) && (buffer1 >= 3.0))   rate = 24;
70          if (buffer1 <= 3.0) rate = 16;
71          printf("Taxa:_%d\n", rate);      #imprime taxa(rate)
72
73          #seleciona a taxa correta conforme condição (if) escolhida
74
75          r(rate); // envia a taxa do myphone
76          printf("Message:_%s\n", buffer);
77      }
78  fprintf(stderr, "Cliente_conectado:
79  %s\n", inet_ntoa(echoclient.sin_addr));
80
81      /* Envia a mensagem de volta ao cliente -
82      só para reconhecimento (ack) */
83
84      if (sendto(sock, buffer, received, 0,
85                (struct sockaddr *) &echoclient,
```

```
86         sizeof(echoclient)) != received) {
87         Die("Mismatch_in_number_of_echo'd_bytes");
88     }
89 }
90 }
```

Variador de taxa no softphone

```
1
2  /* Código adicional no softphone para variar a taxa do codificador */
3  // variáveis globais
4  CString value;
5  int selection;
6
7  extern "C" void r(int rate);
8  // declara função r (int rate) usando função externa "C"
9
10 // Define a função que varia a taxa no Myphone
11 void r(int rate)
12 {
13     // ativa taxa = 40 kbit/s
14     if (rate == 40) {
15         m_CodecListCtrl.GetText(8,g726-40);
16         m_CodecList.Add(-1);
17     }
18     // ativa taxa = 32 kbit/s
19     if (rate == 32) {
20         m_CodecListCtrl.GetText(9,g726-32);
21         m_CodecList.Add(-1);
22     }
23     // ativa taxa = 24 kbit/s
24     if (rate == 24) {
25         m_CodecListCtrl.GetText(10,g726-24);
26         m_CodecList.Add(-1);
27     }
28     // ativa taxa = 16 kbit/s
29     if (rate == 16) {
30         m_CodecListCtrl.GetText(11,g726-16);
31         m_CodecList.Add(-1);
```

32 }

33 }

Apêndice B – Pacotes SR e RR do RTCP

a) SR (*Sender Report*)

Os pacotes SR (*Sender Report*) são enviados pelos transmissores, e contêm informações de *timestamp* NTP, *timestamp* RTP, número de pacotes enviados, número de bytes enviados, bem como informações da qualidade dos outros transmissores que chegam a eles. Com os timestamps (NTP e RTP), o receptor consegue sincronizar mais de uma sessão relacionada (como áudio e vídeo).

A Figura B.1 a seguir ilustra o formato do pacote SR [51].

Figura B.1: Campos do pacote SR

Os pacotes SR consistem de 3 seções: um cabeçalho, uma seção de informações do transmissor e uma seção de informações de outros transmissores. No final, é possível ainda a existência de uma quarta seção contendo extensões específicas de perfil. Os principais campos são explicados a seguir. Maiores informações podem ser encontradas em [51].

- Versão (V): 2 bits: identifica a versão do RTP, que é a mesma do RTCP;
- Padding (P): 1 bit: ver RFC 1889;
- Reception Report Count (RC): 5 bits: número de blocos de informações de outros transmissores;
- Packet Type (PT): 8 bits: valor 200, identifica pacote RTCP SR;
- Length: 16 bits: tamanho deste pacote em palavras de 32 bits menos uma palavra. Este tamanho inclui o cabeçalho;
- SSRC: 32 bits: identificador SSRC do transmissor deste pacote;
- NTP timestamp: 64 bits: indica o tempo NTP que este pacote foi enviado;
- RTP timestamp: 32 bits: corresponde ao mesmo momento do campo NTP, mas nas mesmas unidades, e offset dos pacotes de dados RTP. Com a correspondência do tempo entre NTP e RTP, é possível efetuar sincronização intermídia;
- Sender's packet count: 32 bits: o número total de pacotes de dados transmitidos desde o início da transmissão;
- Sender's octet count: 32 bits: número total de bytes de dados úteis (sem incluir cabeçalho) transmitidos desde o início da transmissão. Este campo pode ser usado para estimar a taxa de transmissão média dos dados úteis;
- SSRC_n (source identifier): 32 bits: indica o SSRC do transmissor para o qual este report está sendo gerado. O transmissor envia um bloco com estatísticas de cada transmissor que ele ouviu desde o último report;
- Fraction lost: 8 bits: fração de pacotes de dados perdidos desde o último report. Esta fração é igual a número de pacotes perdidos dividido pelo número de pacotes esperado;
- Cumulative number of packets lost: 24 bits: o número total de pacotes de dados RTP perdidos desde o início da recepção;
- Extended highest sequence number received: 32 bits: ver RFC 1889;
- Interarrival jitter: 32 bits: estimativa da variação do tempo de chegada dos pacotes RTP, medido em unidades do timestamp e expresso em um valor inteiro;

- Last SR timestamp (LSR) e outros campos: ver [51].

b) RR (Receiver Report)

Os pacotes RR (Receiver Report) são enviados pelos receptores das sessões RTP existentes. O formato do pacote RR é o mesmo do SR, exceto que o tipo de pacote é 201 em vez de 200 e a seção *sender info* é eliminada.