

UNIVERSIDADE ESTADUAL DE CAMPINAS
Faculdade de Engenharia Elétrica e de Computação

“Análise de Atraso dos Pacotes de Voz em Redes IP Utilizando a Técnica de Multiplexagem de Quadros de Voz”

Autor: Carlos Fielde de Campos
Orientador: Prof. Dr. Shusaburo Motoyama

Tese de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: **Telecomunicações e Telemática**

Banca Examinadora

Prof. Dr. Shusaburo Motoyama, Titular. DT/FEEC/Unicamp

Prof. Dr. Carlos Roberto dos Santos, Titular. INATEL

Prof. Dr. Yuzo Yano, Titular. DECOM/FEEC/Unicamp

Campinas, SP

Outubro/2008

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

C157a Campos, Carlos Fielde de
Análise de atraso dos pacotes de voz em redes IP
utilizando a técnica de multiplexagem de quadros de voz
/ Carlos Fielde de Campos. --Campinas, SP: [s.n.], 2008.

Orientador: Shusaburo Motoyama.
Dissertação de Mestrado - Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica e de
Computação.

1. Multiplexação. 2. Desempenho. 3. Sistemas de
telecomunicação. I. Motoyama, Shusaburo. II.
Universidade Estadual de Campinas. Faculdade de
Engenharia Elétrica e de Computação. III. Título.

Título em Inglês: Voice packet delay evaluation in an IP network using voice stream
multiplexing scheme

Palavras-chave em Inglês: Multiplexing, Modeling, Performance Evaluation

Área de concentração: Telecomunicações e Telemática

Titulação: Mestre em Engenharia Elétrica

Banca examinadora: Carlos Roberto dos Santos, Yuzo Yano

Data da defesa: 02/10/2008

Programa de Pós Graduação: Engenharia Elétrica

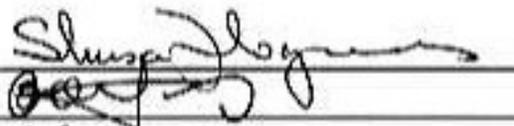
COMISSÃO JULGADORA - TESE DE MESTRADO

Candidato: Carlos Felde de Campos

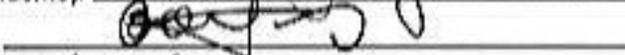
Data da Defesa: 2 de outubro de 2008

Título da Tese: "Análise de Atraso dos Pacotes de Voz em Redes IP Utilizando a Técnica de Multiplexagem de Quadros de Voz"

Prof. Dr. Shusaburo Motoyama (Presidente):



Prof. Dr. Carlos Roberto dos Santos:



Prof. Dr. Yuzo Iano:



Resumo

Neste estudo são avaliados o desempenho e o impacto do tráfego de voz em uma rede *IP*. Na rede *IP* o sinal de voz é transmitido através de pacotes que são identificados por um conjunto de cabeçalhos. Cada pacote de voz tem a carga útil, composta por um quadro (*frame*) de voz, muito menor que a carga útil de um pacote de dados. Desta forma, a rede *IP* pode tornar-se pouco eficiente para transportar sinal de voz. Além disso, o serviço de voz requer também interatividade o que implica em realocar, através de mecanismos apropriados, os recursos da rede *IP*. Uma das técnicas que pode ser usada é a aplicação de política de priorização do tráfego de voz que em contrapartida pode degradar o desempenho de tráfego com prioridade menor. A previsão de tráfego de voz crescente na rede *IP* pode torná-la bastante ineficiente, além de degradar o desempenho do tráfego de dados. Para minimizar este problema o esquema de multiplexagem dos quadros de voz pode ser usado. Um modelo matemático é proposto neste estudo para analisar o impacto do tráfego do pacote de voz em uma rede *IP*. O estudo é conduzido considerando-se sem e com a aplicação de multiplexagem de quadros de voz nos pacotes de voz e o parâmetro de desempenho é o atraso de cada pacote na rede *IP*. Em seguida, um outro modelo mais complexo, com fontes de voz *ON-OFF* e de dados do tipo *HTTP*, é simulado no ambiente do *MatLab/SimEvent* para o mesmo cenário estudado no modelo analítico. Os resultados obtidos mostraram que o comportamento da rede *IP* é similar tanto no modelo analítico como no modelo simulado e que a técnica de multiplexagem dos quadros de voz é um esquema eficiente para economizar a largura de banda de um enlace da rede *IP*.

Palavras-chave: VoIP, Multiplexagem, Modelagem, Análise de Desempenho.

Abstract

In this study, the performance and impact of voice traffic in an *IP* network are evaluated. In the *IP* network, the voice signal is transmitted through packets that are identified by a set of headers. Each voice packet has payload composed of one voice frame, which is smaller than the payload of data packet. In this way, the *IP* network may become ineffective to carry the voice signal. Furthermore, the voice service also requires interactivity which means the need for reallocation, through appropriate mechanisms, the resources of the *IP* network. One of the techniques may be used is through the adoption of voice traffic priority policy which on other hand it may degrade the performance of data traffic with lower priority. The expected fast growing of voice traffic in *IP* network can make it inefficient as well as to degrade the data traffic performance. To minimize this problem the voice stream multiplexing scheme can be used. A mathematical model is proposed in this study to examine the impact of voice packet traffic in an *IP* network. The study is carried out considering with and without the application of voice frame multiplexing and the performance parameter is the delay of each packet in the *IP* network. Then, a more complex model, with *ON-OFF* voice sources and data sources of *HTTP* type, is simulated in *MatLab/SimEvent* environment for the same scenario used in analytical model. The obtained results showed that the *IP* network behavior is similar in both analytical and simulated models and that voice frame multiplexing technique is an efficient scheme to save the bandwidth of a link.

Keywords: VoIP, Multiplexing, Modeling, Performance Evaluation.

Agradecimentos

Ao Prof. Shusaburo Motoyama, pela orientação neste trabalho, por sua disponibilidade e suporte ao longo de todo o estudo.

À minha família pelo apoio, em especial à minha esposa Kátia e aos meus filhos Isabela e Pietro.

Sumário

Lista de Figuras.....	vii
Lista de Tabelas	viii
Glossário	ix
Lista de Símbolos.....	xi
Trabalho Publicado Pelo Autor.....	xii
Capítulo 1	1
Introdução.....	1
Capítulo 2	5
Tecnologia VoIP	5
2.1 Introdução.....	5
2.2 Componentes do Sistema VoIP.....	5
2.3 Problemas no Sistema VoIP	8
2.3.1 Largura de Banda	8
2.3.2 Atraso (fim-a-fim) de pacotes de Voz	9
2.3.3 Atraso Dejitter Buffer	13
2.3.4 Perda de Pacotes de Voz.....	14
2.3.5 Arquitetura de VoIP	16
2.3.6 Protocolos para VoIP.....	18
2.3.6.1 Protocolos RTP e RTCP	18
2.3.6.2 Protocolo H.323	19
2.3.6.3 Protocolo SIP	21
2.4 Conclusão:	22
Capítulo 3	24
Modelagem Analítica do tráfego de Voz em redes IP.....	24
3.1 Introdução.....	24
3.2 Estudo do Atraso em redes IP – Modelagem Matemática.....	25
3.3 Análise de Resultados	30
3.4 Conclusão:	36
Capítulo 4	37
Simulação de Tráfego em redes IP	37
4.1 Introdução.....	37
4.2 Fonte de Voz – Modelo ON-OFF	38
4.2.1 Histogramas das Fontes de Voz – Modelo ON-OFF.....	40
4.3 Fonte de Dados – Modelo HTTP	43
4.3.1 Propriedades da Fonte de Dados – Modelo HTTP.....	43
4.3.2 Parametrização da Fonte de Dados – Modelo HTTP	45
4.3.3 Histogramas das Fontes de Dados – Modelo HTTP.....	47
4.3.4 Escalonadores com esquema FIFO (SimEvent).....	49
4.3.5 Escalonadores com esquema de Priorização (SimEvent)	50
4.4 Análise de Resultados (Matlab/SimEvent)	51
4.5 Conclusão:	58
Capítulo 5	60
Conclusões.....	60
Referências Bibliográficas.....	62
Apêndice A – Tempo de Espera (Simulação).....	64
Apêndice B – Modelos de filas no SimEvent	71
Apêndice C – Trabalho Publicado no International Workshop on Telecommunications – IWT2007	73

Lista de Figuras

Figura 1 – Técnica de multiplexagem na transmissão de pacotes de voz.	3
Figura 2.1 – Elementos usados para estabelecer uma ligação <i>VoIP</i>	6
Figura 2.2 – <i>VoIP Gateway</i> interconecta redes de telefonia (RTPC) e redes <i>IP</i>	7
Figura 2.3 – <i>VoIP Gatekeeper</i> interconecta duas redes <i>IP</i>	8
Figura 2.4 – Restrição de largura banda (recursos) no núcleo da rede <i>IP</i>	9
Figura 2.5 – Atraso fim-a-fim de cada pacote de Voz na rede <i>IP</i>	10
Figura 2.6 – <i>CODEC G.729</i> gera fluxo de voz (quadros de voz) para empacotamento <i>IP</i>	12
Figura 2.7 – Atraso de enfileiramento de cada pacote de voz na rede <i>IP</i>	13
Figura 2.8 – Tratamento dos pacotes de voz no terminal receptor (Dejitter Buffer)	14
Figura 2.9 – Perda de pacote no <i>buffer</i> (nó da rede)	14
Figura 2.10 – Arquitetura <i>PC-a-PC</i>	17
Figura 2.11 – Arquitetura com <i>Gateway</i>	17
Figura 2.12 – Ligação <i>VoIP</i> entre terminais <i>H.323</i>	19
Figura 2.13 – <i>Gatekeeper</i> e zona <i>H.323</i>	20
Figura 3.1 – Tempo de espera: os pacotes de voz e dados no caso da disciplina <i>FIFO</i>	33
Figura 3.2 – Tempo de espera: prioriza pacotes de voz (sem multiplexagem)	34
Figura 3.3 – Tempo de espera: prioriza pacotes (com multiplexagem)	35
Figura 4.1 – Nível de Conversação e transmissão de quadros de voz (<i>CODEC</i>)	38
Figura 4.2 – Modelo de Fonte de Voz <i>ON-OFF</i> (com 02 estados)	39
Figura 4.3 – Modelagem de Tráfego de Voz <i>ON-OFF</i>	40
Figura 4.4 – Histograma do Tamanho do Pacote de Voz <i>ON-OFF</i> (50 bytes)	41
Figura 4.5 – Histograma do Tamanho do Pacote de Voz <i>ON-OFF</i> (340 bytes)	41
Figura 4.6 – Histograma do intervalo entre pacotes <i>ON-OFF</i> (50 bytes)	42
Figura 4.7 – Histograma do intervalo entre pacotes <i>ON-OFF</i> (340 bytes)	42
Figura 4.8 – Ordem de chegada dos pacotes <i>ON-OFF</i> (50 bytes)	43
Figura 4.9 – Transferência de Pacotes durante uma seção <i>HTTP</i>	44
Figura 4.10 – Algoritmo do Tráfego de Voz <i>ON-OFF</i>	47
Figura 4.11 – Histograma do Tamanho do Pacote <i>HTTP</i>	48
Figura 4.12 – Ordem de chegada dos pacotes <i>HTTP</i>	48
Figura 4.13 – Histograma do intervalo entre pacotes <i>HTTP</i>	49
Figura 4.14 – Tempo de espera dos pacotes de voz e dados sem <i>MUX</i> e fila <i>FIFO</i>	56
Figura 4.15 – Tempo de espera dos pacotes de voz e dados com <i>MUX</i> e fila <i>FIFO</i>	56
Figura 4.16 – Tempo de espera com priorização (voz) e sem multiplexagem	57
Figura 4.17 – Tempo de espera com priorização (voz) e com multiplexagem	58
Figura A.1 – Distribuição do tempo de espera fila <i>FIFO</i> (Sem Mux e 69,4% de carga)	64
Figura A.2 – Distribuição do tempo de espera fila <i>FIFO</i> (Com Mux e 25,61% de carga)	65
Figura A.3 – Distribuição do tempo de espera fila <i>FIFO</i> (Sem Mux e 96,74% de carga)	65
Figura A.4 – Distribuição do tempo de espera fila <i>FIFO</i> (Com Mux e 60,84% de carga)	66
Figura A.5 – Distribuição do tempo de espera fila <i>ON-OFF</i> (Sem Mux e 69,4% de carga)	66
Figura A.6 – Distribuição do tempo de espera fila <i>HTTP</i> (Sem Mux e 69,4% de carga)	67
Figura A.7 – Distribuição do tempo de espera fila <i>ON-OFF</i> (Com Mux e 25,61% de carga)	68
Figura A.8 – Distribuição do tempo de espera fila <i>HTTP</i> (Com Mux e 25,61% de carga)	68
Figura A.9 – Distribuição do tempo de espera fila <i>ON-OFF</i> (Sem Mux e 96,74% de carga)	69
Figura A.10 – Distribuição do tempo de espera fila <i>HTTP</i> (Sem Mux e 96,74% de carga)	69
Figura A.11 – Distribuição do tempo de espera fila <i>ON-OFF</i> (Com Mux e 60,84% de carga)	70
Figura A.12 – Distribuição do tempo de espera fila <i>HTTP</i> (Com Mux e 60,84% de carga)	70

Lista de Tabelas

Tabela 2.1 – Algumas especificações de <i>CODEC</i> usados em <i>VoIP</i>	11
Tabela 3.1 – Parâmetros usados na análise numérica	31
Tabela 3.2 – Relações entre x e y considerando a utilização do enlace: sem <i>MUX</i> e com <i>MUX</i>	32
Tabela 3.3 – Relações entre x, o esquema de multiplexagem e a utilização do enlace.....	33
Tabela 3.4 – Tempo de espera: 30 canais de voz (ativos) e pacotes de dados na disciplina <i>FIFO</i> ...	34
Tabela 4.1 – Parâmetros dos modelos de <i>VoIP</i> por <i>CODEC</i>	40
Tabela 4.2 – Parâmetros para Modelagem de Tráfego <i>HTTP</i>	46
Tabela 4.3 – Parâmetros obtidos a partir de simulação de fonte voz (<i>ON-OFF</i>) e dados (<i>HTTP</i>) ...	53
Tabela 4.5 – Relações entre x, o esquema de multiplexagem e a utilização do enlace (simulação).	55
Tabela 4.6 – Tempo de espera: 30 canais de voz (ativos) e pacotes de dados na disciplina <i>FIFO</i> ...	57

Glossário

ADSL - Asymmetric Digital Subscriber Line
ASCII - American Standard Code for Information Interchange
ATA - Analog Telephone Adaptor
CODEC - Coder/Decoder
CRC - Cyclic Redundancy Check
DiffServ – Differentiated Service
DSP - Digital Signal Processing
FEC - Forwarding Equivalency Class
FIFO - First In First Out
HTML - Hyper Text Markup Language
HTTP - Hypertext Transfer Protocol
IETF - Internet Engineering Task Force
IETF - Internet Engineering Task Force
IntServ - Integrated Service
IP - Internet Protocol
IP - Internet Protocol
ITU-T - International Telecommunication Union (Telecommunication Standardization Sector)
LAN – Local Area Network
LER - Label Edge Router
LSP - Label Switched Path
MCU - Multi Control Unit
MIPS – Millions of Instructions Per Seconds
MPLS - Multi Protocol Label Switching
MTU - Maximum Transfer Unit
MUX - Multiplexer
NGN - Next Generation Network
OSI - Open Systems Interconnection
PDF - Probability Distribution Function
QoS - Quality of Service
RAM - Random Access Memory

RAS - Remote Access Service
RFC - Request for Comments
RSVP - Resource ReSerVation Protocol
RTCP - Real-Time Transport Control Protocol
RTP - Real Time Protocol
RTP - Real Time Protocol
SIP - Session Initiation Protocol
SLA - Service Level Agreement
TCP - Transmission Control Protocol
TL – Tempo de Leitura
TTL - Time-to-Live
UDP - User Data Protocol
UDP - User Datagram Protocol
URL - Uniform Resource Locator
VoIP - Voice over IP
WEB - World Wide Web

Lista de Símbolos

λ_v - Taxa de chegada dos pacotes de voz

λ_D - Taxa de chegada dos pacotes de dados

ρ_D - Carga de tráfego dos pacotes de dados

ρ_v - Carga de tráfego dos pacotes de voz

C - Capacidade do enlace de saída (*bps*)

$E\{T_0\}$ - Tempo médio para completar a transmissão do pacote atual quando um outro pacote chega

$E\{W\}$ - Tempo médio de espera dos pacotes na fila

$E\{R_v\}^2$ - Quadrado do Primeiro momento da distribuição do comprimento dos pacotes de voz

$E\{R_D\}^2$ - Quadrado do Primeiro momento da distribuição do comprimento dos pacotes de dados

$E\{R_v^2\}$ - Segundo momento da distribuição do comprimento dos pacotes de voz

$E\{R_D^2\}$ - Segundo momento da distribuição do comprimento dos pacotes de dados

σ_v^2 - Variância da distribuição dos pacotes de voz

σ_D^2 - Variância da distribuição dos pacotes de dados

$\frac{1}{\lambda_v}$ - Intervalo entre os pacotes de voz (em segundos)

$\frac{1}{\lambda_D}$ - Intervalo entre os pacotes de dados (em segundos)

$\frac{1}{\mu_v}$ - Tamanho médio dos pacotes de voz (em bytes)

$\frac{1}{\mu_D}$ - Tamanho médio dos pacotes de dados (em bytes)

Trabalho Publicado Pelo Autor

1. C. F. Campos, S. Motoyama. “Voice Packet Delay Evaluation in an IP Network Using Voice Stream Multiplexing Scheme” – *International Workshop on Telecommunications – IWT2007* – Santa Rita do Sapucaí – MG. Páginas 6-11. 23-27 de Fevereiro de 2007

Capítulo 1

Introdução

As redes de telecomunicações, baseadas no protocolo IP, tiveram expressivo crescimento principalmente a partir da expansão da *Intenet*. A consequência é que as redes IP têm sido utilizadas como plataforma para integrar os diferentes tipos de tráfego. As aplicações de voz, vídeo e dados geram tráfego que demandam diferentes qualidades de serviços (*QoS – Quality of Service*). As redes IP (IPv4), por exemplo, que efetuam conexão IP na Internet, não estão preparadas para suportar *QoS*. Estas redes utilizam a tecnologia do melhor esforço (*Best-effort Service ou Lack of QoS*) em que não há classificação e priorização de tráfego além de qualquer garantia de alocação de recursos.

Nas redes *IP* o encaminhamento de pacotes é baseado na técnica de armazenamento e encaminhamento (*Store and Forward*). A vantagem desta técnica é otimizar a utilização dos recursos de banda disponíveis através da multiplexagem estatística. Os pacotes que chegam primeiro, em qualquer ponto (nó) da rede, podem ser armazenados (enfileirados) e saem primeiro através do esquema *FIFO (First In First Out)* sem qualquer garantia. Isto implica na inserção de atraso nos pacotes enfileirados, e dependendo do tamanho da fila pode haver descarte de tráfego, indistintamente. Este cenário caracteriza a ausência de aplicação de *QoS* na rede. Portanto se faz necessário, nas redes *IP*, o provimento de diferenciação de tráfego conforme os requisitos de cada aplicação. As aplicações de voz, por exemplo, demandam tempo real e são muito sensíveis ao atraso [5]. O IETF (*Internet Engineering Task Force*) [1] especificou dois modelos de classes de serviço para o provimento de *QoS* em redes IP.

Um dos modelos é o *IntServ (Integrated Service)* também denominado de *HardQoS*. Para estabelecer na rede IP caminho com reserva de recursos, largura de banda e tempo de conexão, é utilizado o protocolo de sinalização fim-a-fim *RSVP (Resource ReSerVation Protocol)* [2]. Este protocolo interage, ao longo do caminho, com todos os roteadores associados sobre as condições de uso da rede. Aloca largura de banda enquanto durar a conexão. Efetua, através de algoritmo de gerenciamento de *buffers*, a criação de filas específicas e garante, complementarmente, o provimento de priorização. O caminho resultante, roteador a roteador, implica em um serviço orientado à conexão no ambiente da Internet com aplicação de *QoS*. Uma das restrições para a utilização deste modelo está relacionada à escalabilidade da rede. Para que o transporte de pacotes ocorra, cada roteador relacionado deve classificar o fluxo gerado por aplicação alocando-os em filas específicas. Em situações de aumento de tráfego, como o controle ocorre em cada roteador, aumenta significativamente o tempo de processamento na rede.

Outro modelo apresentado pelo *IETF* para que a rede *IP* possa prover *QoS* é baseado na arquitetura *DiffServ* [3]. Este modelo é mais interessante, pois demanda menos capacidade de processamento dos roteadores, e desta forma permite um nível maior de escalabilidade.

Na solução *DiffServ* os pacotes são classificados em classes de fluxo. Cada classe de fluxo pode ser tratada de uma mesma forma em toda a rede. A aplicação fim-a-fim de serviços diferenciados pode ser estabelecida desde que cada domínio *IP* suporte *QoS* (domínio *QoS*). O encaminhamento de pacotes através de domínios *QoS* adjacentes requer o estabelecimento de *SLA* (Acordo de Nível de Serviço) entre os mesmos. O *SLA* determina as classes de serviços suportadas e a banda alocada entre domínios.

Neste modelo foram inseridas entidades para gerenciar e controlar a largura de banda, as políticas e prioridades dentro e entre domínios. Nos roteadores de borda (*Edge Routers*) os fluxos provenientes de aplicações afins são agregados em classes de tráfego pré-estabelecidas diferente do modelo *IntServ* em que o fluxo de tráfego é por aplicação. Desta forma, é possível diminuir o tempo de processamento nos roteadores da rede e suportar escalabilidade.

Outra alternativa para aplicação de *QoS* em redes *IP* é baseada na plataforma *MPLS* (*Multi Protocol Label Switching*) [4]. O *MPLS* é uma técnica que integra o processo de troca de rótulos com os protocolos de nível de rede (por exemplo, o *IP*). O rótulo é um identificador de comprimento curto usado para identificar uma *FEC* (*Forwarding Equivalency Class*). *FEC* é um agrupamento de pacotes que possuem os mesmos requisitos de tráfego na rede. O pacote *IP* ao ingressar na rede *MPLS* é mapeado em rótulos no *LER* (*Label Edge Router*) que determinam caminhos nesta plataforma. Uma vez definido um caminho *MPLS*, os pacotes *IP* rotulados são comutados nó a nó e conforme atravessam a rede por este conjunto de nós estabelecem um *Label Switched Path* (*LSP*). O *LSP* garante o mesmo tratamento do tráfego até o seu destino.

Através da *FEC* é possível priorizar classes de tráfego que demandam reserva de recursos na rede. A implementação de *QoS* no ambiente *MPLS* ocorre a partir da capacidade de escolha de rotas que atendam aos requisitos de tráfego das aplicações afins (fluxos agregados) tal que o caminho na rede forneça o nível de serviço desejado. Para a obtenção de diferentes níveis de serviço é necessário o uso de política de policiamento de tráfegos que entram e que saem em toda a rede *MPLS*. Por exemplo, as aplicações que necessitam banda mínima e baixa latência utilizam caminhos *MPLS* nó a nó com reserva de banda e filas específicas com nível de prioridade elevado. Os nós de borda da rede *MPLS* (*LER*) encapsulam os pacotes *IP* ingressantes em rótulos apropriados e os nós do núcleo da rede – *Label Switching Router* (*LSR*) – efetuam o encaminhamento do tráfego *IP* já rotulado e garantem a alocação de recursos em todo caminho *MPLS*. Os mesmos nós de borda (*LER*) retiram os rótulos dos pacotes egressos e encaminham somente o pacote *IP* para o destino.

As conversações telefônicas através de redes *IP* são aplicações que estão crescendo de maneira bastante significativa, mas como são sensíveis à atraso excessivo, devem ser tratadas como tráfego de tempo real. Portanto, o tráfego de voz demanda uma plataforma *IP* que suporte *QoS*. A plataforma *IP* pode ser qualquer uma das plataformas discutidas anteriormente que suportam *QoS*.

Em qualquer caso, os pacotes de voz podem ser classificados em classes de fluxo e tratados com alta prioridade em cada nó e assim por todo caminho estabelecido na rede *IP*.

Quando o tráfego de voz é transportado em uma plataforma *IP*, uma outra questão importante que deve ser considerada é o tamanho dos quadros (*frames*) de voz. Para que ocorra a transmissão de voz em redes *IP* é necessário que o fluxo de voz analógico (fonte) seja digitalizado e comprimido em fluxo de voz digital através de um codificador. Na recepção o processo inverso é aplicado tal que o fluxo de voz digital é decodificado em fluxo de voz analógico novamente. A aplicação destes processos de codificação e decodificação é denominada *CODEC*. Para que os fluxos digitais que deixam o *CODEC* sejam transferidos até a rede *IP* é necessário que o fluxo de voz seja convertido em *frames*. O tamanho de cada *frame* está entre 10 e 30 *bytes* (valores típicos). A cada unidade de *frame* é inserido um conjunto de cabeçalhos da arquitetura *IP* até o nível de rede. O cabeçalho resultante da concatenação de cabeçalhos tem 40 *bytes* e pode ser formado a partir dos protocolos *RTP* (*Real Time Protocol*), *UDP* (*User Data Protocol*) e *IP* (*Internet Protocol*). O tamanho de cada *frame* de voz depende do tipo de *CODEC*. Por exemplo, o *CODEC* G.729 transmite *frame* de voz a uma taxa constante de 8 *kbps* com intervalo entre *frames* de 10 ms. Neste contexto o comprimento de *frames* de 10 ms pode acomodar 10 *bytes*, 20 ms pode acomodar 20 *bytes* e assim por diante. Quanto maior o tamanho do *frame* de voz, mais bits podem ser acomodados em cada *frame*. A contrapartida é o aumento do atraso de *frame* no processo de digitalização aplicado pelo *CODEC*, pois a taxa de saída (8 *kbps*) é mantida fixa. Outro fator importante na transferência dos *frames* de voz é o atraso de propagação através da rede *IP*. A conjunção do atraso de *frames* (tempo de processamento do *CODEC*) e atraso da rede *IP* podem afetar a interatividade da conversação telefônica. Portanto, o comprimento dos *frames* de voz deve ser mantido pequeno o que torna muito ineficiente a transmissão de voz na rede *IP*.

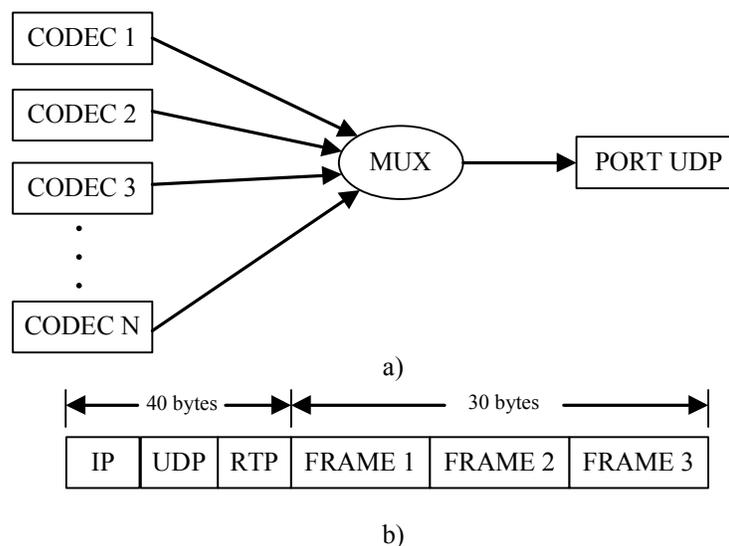


Figura 1 – Técnica de multiplexagem na transmissão de pacotes de voz.

Um exemplo típico é o encapsulamento de *frame* de voz de 10 *bytes* (G.729) com o cabeçalho resultante da plataforma *IP* (*RTP*, *UDP* e *IP*) de 40 *bytes* que resulta numa eficiência de apenas

20%. Para *frames* maiores há incremento no atraso de *frames*, e a eficiência torna-se melhor, porém, é uma abordagem ainda muito ineficiente. Em [6] uma técnica de multiplexagem é proposta para melhorar a eficiência de transmissão dos pacotes de voz sem o respectivo aumento no atraso de *frames*. Na técnica de multiplexagem proposta em [6], os *frames* que deixam os *CODECs* de diferentes fontes de voz são multiplexados antes que o cabeçalho resultante dos protocolos *RTP*, *UDP* e *IP* seja inserido como mostrado na Figura 1.

Como pode ser observado no exemplo da Figura 1b, três *frames* distintos de voz com 10 *bytes* cada são multiplexados resultando num pacote *IP* de 70 *bytes* sendo 30 *bytes* carga útil (*payload*), e eficiência de 42,85% invés de 20% quando os mesmos *frames* são transmitidos separadamente. Para relacionar atraso de pacote e utilização de rede foi suposto em [9] um *link* de 512 *kbps* com tráfego de voz crescente proporcional ao número de conexões simultâneas. O comparativo de desempenho do atraso resultante é referente à aplicação ou não do esquema de multiplexagem de *frames*. O estudo [9] através de simulação não prevê aplicação de *QoS* na rede através de priorização de tráfego.

Neste trabalho é estudado o impacto dos pacotes de voz com e sem a aplicação de multiplexagem de *frames* no desempenho da rede *IP*. O estudo é realizado através de análise de modelos matemáticos (resultados analíticos) e também através de simulação com o uso de fontes de tráfego desenvolvidas no ambiente do *Simulink* e do *Matlab*. O parâmetro de análise de restrição de *QoS* é o atraso médio. Para o caso de *Best Effort* (ausência de *QoS*), cenário típico das redes *IP*, o servidor (nó da rede) trata todos os pacotes sem aplicar qualquer diferenciação, os pacotes de voz e dados aguardam pelo atendimento numa única fila *FIFO*. Em outro cenário, ocorre a aplicação de *QoS* em que os pacotes de voz são armazenados em filas específicas e o servidor trata primeiro a fila de voz e em seguida a fila de dados. O tráfego de voz gerado é parametrizado em 30 canais ativos e o tráfego de dados é incrementado gradativamente até a capacidade máxima de atendimento do servidor. No modelo matemático o tráfego de voz e dados é tratado como fonte poissoniana e na simulação o tráfego de voz reproduz as fontes “*ON-OFF*” e o tráfego de dados, as fontes *HTTP*. É estabelecido um comparativo entre os resultados obtidos com e sem aplicação de *QoS*, referentes ao atraso médio, no modelo teórico e no simulado. O objetivo é demonstrar, através de uma carga de tráfego de voz constante, as vantagens do uso do esquema de multiplexagem dos *frames* de voz.

O complemento deste trabalho está disposto em mais quatro capítulos: No capítulo 2 são apresentados os conceitos de *VoIP* e estudos comparativos sobre eficiência de transporte de tráfego de voz em rede *IP*. A modelagem teórica de tráfego e a análise do impacto do tráfego de voz com e sem o esquema de multiplexagem de *frames* na rede *IP* são efetuadas no capítulo 3. No capítulo 4 são utilizadas fontes de voz (modelo *ON-OFF*) e dados (modelo *HTTP*) no ambiente de simulação do *Matlab* e *SimuLink* para gerar tráfegos compatíveis com a realidade e é analisado o esquema de multiplexagem dos *frames* de voz. Finalmente, as principais conclusões e sugestões para trabalhos futuros são apresentadas no capítulo 5.

Capítulo 2

Tecnologia *VoIP*

2.1 Introdução

Uma ligação telefônica convencional usa a rede de telefonia pública comutada (RTPC) para a transmissão de voz. A tecnologia empregada é baseada na técnica de comutação por circuito. Neste caso são alocados recursos dedicados na rede de telefonia para garantir uma qualidade mínima na conversação telefônica. Por outro lado uma ligação telefônica que usa a tecnologia *VoIP* (Voz sobre *IP*) transmite os sinais de voz através da rede *IP* (ex. *Internet*). A rede *IP* encaminha o tráfego de voz baseada na técnica de comutação de pacotes, efetua compartilhamento de recursos e não garante qualidade mínima na conversação telefônica. A rede *IP* quando comparada à RTPC provê melhor capacidade de alocação e utilização de recursos, além de redução dos custos telefônicos, porém não garante qualidade de serviço. Para suprir esta deficiência existem modelos de classes de serviços (ex. *DiffServ*) que podem ser aplicados na rede *IP*.

Neste capítulo são apresentados os principais componentes e problemas no sistema *VoIP* para a transmissão dos sinais de voz. Em seguida são introduzidas as arquiteturas de *VoIP* empregadas no estabelecimento de ligações telefônicas. Para completar são descritos os protocolos utilizados na implementação de *VoIP* e ao final são apresentadas as conclusões.

2.2 Componentes do Sistema *VoIP*

Para que uma ligação telefônica *VoIP* ocorra é necessário, além da rede *IP*, o uso de componentes *VoIP* para tratar e transmitir o sinal de voz. Estes componentes permitem que o serviço de *VoIP* realize as mesmas funcionalidades providas em uma ligação telefônica convencional. Um sistema de *VoIP* típico necessita de 03 componentes básicos conforme mostrado na Fig. 2.1:

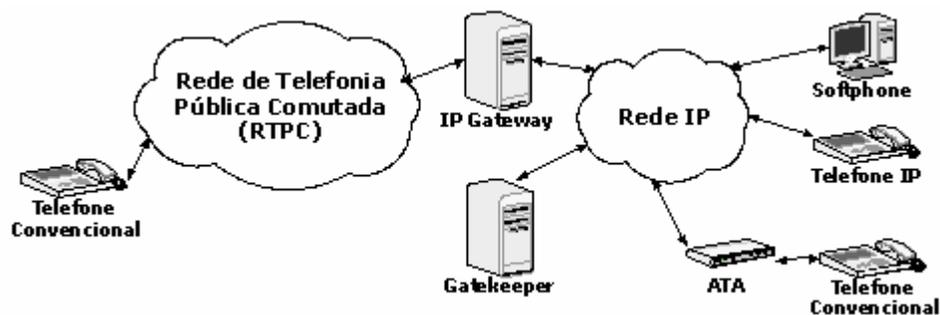


Figura 2.1 – Elementos usados para estabelecer uma ligação *VoIP*.

- Elemento Terminal ou dispositivo de comunicação (ex. telefone *IP* e *Softphone*);
- *Gatekeeper* (guardião) que realiza a tradução de endereços *IP* e o controle de acesso à *LAN* por terminais e roteadores;
- *Gateway* efetua a interconexão entre a rede *IP* e a rede de telefonia convencional;

Uma conexão *VoIP* fim-a-fim, ou uma chamada telefônica *VoIP*, inicia no terminal (origem) e é completada no terminal (destino) . O terminal, por exemplo, telefone *IP* é um dispositivo que na origem digitaliza o sinal de voz e no destino transforma o sinal digital em analógico. O telefone *IP* é um dispositivo dedicado composto por *hardware* e *software* específicos que permitem o encaminhamento e recebimento de ligações telefônicas através de redes *IP*. Outra opção de terminal *VoIP* é o *Softphone* que possibilita a utilização de *PC* em ligações *VoIP*. Neste caso o *PC* é provido de aplicações multimídia (*Softwares* de *VoIP*) que permitem o estabelecimento de uma ligação *VoIP*. É possível também o uso de telefone analógico o qual necessita de uma interface (*ATA*) para conectar-se ao sistema *VoIP*. Neste caso envia a voz analógica para um dispositivo *ATA*. Este dispositivo digitaliza, empacota e encaminha o sinal de voz para o sistema *VoIP*. O *ATA* permite a integração entre a tecnologia legada (sistema analógico) e a plataforma *VoIP*. Uma vantagem do sistema *VoIP* é que diferentes tipos de terminais podem originar e receber ligações *VoIP* e também ligações convencionais entre si (ex. Telefone *IP* chama terminal *Softphone*).

O terminal *VoIP* (origem) que encaminha uma ligação para a rede de telefonia pública comutada (terminal analógico) necessita do *VoIP Gateway*. Este componente provê a interconexão entre o sistema *VoIP* e a plataforma RTPC. O *VoIP Gateway* recebe o pacote de voz (sistema *VoIP*), extrai informações sobre o terminal de destino na RTPC e consulta tabela de roteamento pré-definida. Em seguida efetua a troca de sinalização e uso de protocolos conforme demandam a plataforma RTPC e o sistema *VoIP*. O *VoIP Gateway* mapea os números de telefones da RTPC com os terminais *VoIP* (endereço *IP*) o que possibilita a interconexão entre a rede *VoIP* e a plataforma RTPC. Numa ligação originada do terminal *VoIP* para um telefone convencional o *VoIP Gateway*, no lado da RTPC, atua como um switch de telefonia e estabelece caminhos dedicados na RTPC conforme indicado na Fig. 2.2. No sentido contrário, no lado da rede *IP*, o *VoIP Gateway* gera pacotes *IP*. Após o completamento da ligação ocorre a transmissão unidirecional do sinal de voz.

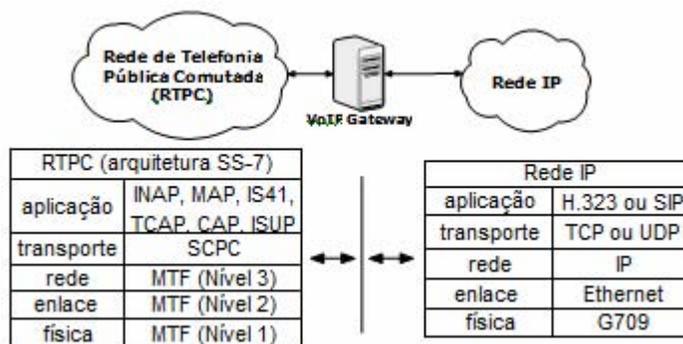


Figura 2.2 – VoIP Gateway interconecta redes de telefonia (RTPC) e redes IP

Outra possibilidade de uso do VoIP Gateway é para ligações entre terminais VoIP. A tabela do VoIP Gateway relaciona os endereços IP (origem e destino), mapea e estabelece uma ligação VoIP exclusivamente na rede IP. Adicionalmente, o VoIP Gateway também integra sistemas VoIP baseados ou em protocolos H.323 ou SIP e habilita a compatibilidade entre diferentes sistemas e dispositivos VoIP. O VoIP Gateway é implementado em switches, roteadores ou em servidores dedicados.

O VoIP Gateway (origem) recebe uma chamada e se o terminal de destino não está mapeado em seu domínio (área de cobertura) procura, através do Gatekeeper, outro VoIP Gateway (destino) que suporte o terminal chamado. O VoIP Gateway primeiro consulta o Gatekeeper que possui mapeado em tabela própria o VoIP Gateway (destino). Após consulta à tabela o Gatekeeper verifica a disponibilidade de recursos na rede IP, gerencia e limita a quantidade de ligações VoIP ativas e, se não houver restrições, efetua o mapeamento entre os Gateways (origem e destino). Em seguida o Gatekeeper sai da conexão e o VoIP Gateway (destino) encaminha a ligação ao terminal de destino (conexão fim-a-fim). Após o completamento da chamada VoIP somente os terminais (origem e destino) e os VoIP Gateways (origem e destino) estão em uso. Ao término da ligação os Gateways (origem e destino) interagem com o Gatekeeper e solicitam a liberação dos recursos alocados para o estabelecimento da chamada. A conexão VoIP fim-a-fim é desfeita e o Gatekeeper atualiza a alocação de recursos do sistema VoIP.

O uso dos componentes básicos do sistema VoIP possibilita ligações telefônicas na plataforma de rede IP. A conexão VoIP fim-a-fim é estabelecida porém a qualidade da ligação depende do desempenho da rede IP. Esta situação difere do sistema de telefonia convencional em que uma vez estabelecida uma conexão telefônica há garantia de qualidade mínima da ligação fim-a-fim. O estudo da rede IP é importante para apontar os problemas no sistema VoIP e ações necessárias para garantir qualidade mínima nas ligações VoIP.

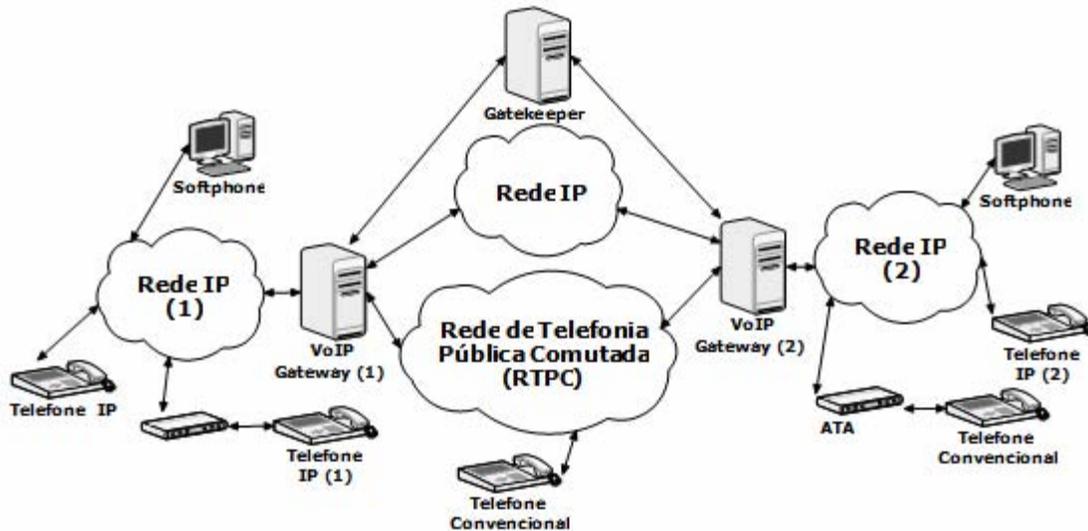


Figura 2.3 – VoIP Gatekeeper interconecta duas redes IP

2.3 Problemas no Sistema VoIP

O sistema VoIP usa a rede IP para encaminhar ligações VoIP. A condição de uso da rede IP determina se é ou não possível o estabelecimento de nova chamada telefônica. Em caso de restrição de recursos na rede IP (ex. congestionamento) o sistema VoIP apresenta limitações na manutenção e também no estabelecimento de uma ligação VoIP. Uma chamada que é originada no terminal VoIP (origem) até o terminal VoIP (destino) só é encaminhada se a rede IP prover condição mínima de uso. O desempenho da rede IP tem impacto direto no desempenho do sistema VoIP. Os quatro principais fatores, na rede IP, que influenciam a qualidade do sinal de voz recebido no terminal (destino) são: largura de banda, atraso (fim-a-fim) de pacotes de voz, *jitter* e perda de pacotes de voz.

2.3.1 Largura de Banda

O sinal de voz analógico é digitalizado e comprimido através do uso de técnicas de codificação. Cada técnica empregada (ex. CODEC G.723.1) requer, para a transmissão do sinal de voz digital, largura de banda mínima. A rede IP, baseada na técnica de armazenamento e encaminhamento de pacotes, não prevê garantia de alocação de recursos. A largura de banda disponível na rede IP é função da carga de tráfego (evento não determinístico) resultante na rede. Um segmento de rede IP (nó da rede) congestionado não garante largura de banda e ocorre restrição na capacidade de transmissão do sinal de voz. O pacote de voz é armazenado no nó da rede IP (ex: *buffer* do roteador) e só é enviado quando há banda disponível. Quanto menor a banda disponível mais tempo o pacote de voz permanece no *buffer* que é definido como tempo de espera no *buffer*. Se o *buffer* do roteador está com a capacidade excedida o pacote de voz é descartado. A condição de uso da rede IP (ex. carga de tráfego) influi no atraso do pacote de voz. A Fig. 2.4 apresenta uma condição de uso de um nó no núcleo da rede IP que tem capacidade de transmitir até 10Mbits/s.

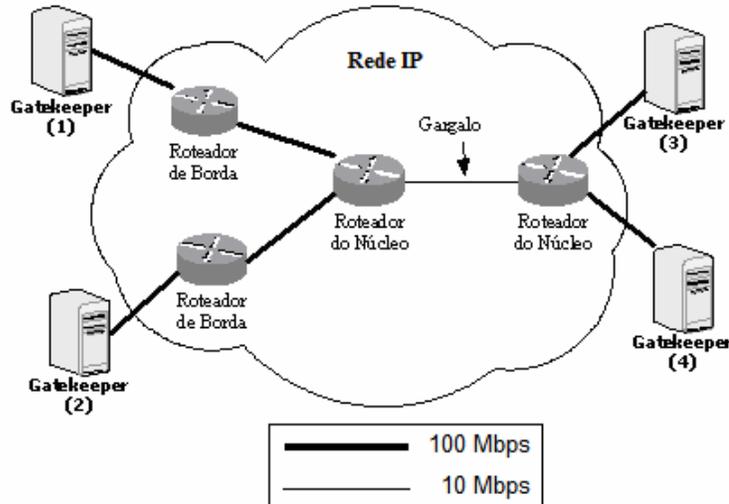


Figura 2.4 – Restrição de largura banda (recursos) no núcleo da rede *IP*

Os nós de borda da rede multiplexam (agregam) tráfego de voz de até 100Mbits/s. Neste caso o núcleo da rede *IP*, que suporta até 10Mbps, tem menos capacidade que o tráfego resultante dos roteadores de borda e pode haver inserção de atraso (enfileiramento) na transmissão dos pacotes de voz o que degrada o sinal de voz (qualidade de áudio) no terminal de destino. O sistema *VoIP* é vulnerável à dinâmica de uso da rede *IP* (carga de tráfego). A implementação de diferenciação de tráfego em redes *IP* (ex. *DiffServ*) possibilita a alocação de largura de banda mínima na rede conforme os requisitos do sistema (aplicação) *VoIP*.

2.3.2 Atraso (fim-a-fim) de pacotes de Voz

Uma chamada *VoIP* inicia no terminal (origem) que codifica o fluxo de voz analógico (fonte) em digital e gera periodicamente pacotes de voz. Nesta fase o intervalo entre pacotes é constante. Cada pacote transita através dos componentes de *VoIP* e da rede *IP*, para transmitir o sinal de voz, até o terminal (destino). O atraso fim-a-fim ou latência é o tempo gasto para transmitir o fluxo de voz desde o terminal de origem até o terminal de destino. Em cada fase de encaminhamento de uma chamada *VoIP* há inserção de atraso. Uma chamada iniciada no terminal de origem sofre o atraso de codificação que compreende o processo de digitalização e compressão além do atraso de empacotamento do sinal de voz.

Em seguida, o pacote de voz trafega na da rede *IP*. Cada pacote de voz é transportado, na rede *IP*, através de um caminho (rota). O caminho estabelecido na rede é composto de vários nós (roteadores). O pacote de voz em cada nó da rede *IP* pode ser armazenado (enfileirado) e em seguida é repassado. O tempo de enfileiramento do pacote de voz em cada nó da rede é aleatório e depende da carga de tráfego (rede *IP*). Quanto maior a carga de tráfego na rede *IP* maior é o tempo de permanência do pacote de voz no *buffer* de cada nó (roteador). Como o uso da rede *IP* é baseado na técnica de multiplexagem estatística a previsão da carga de tráfego (rajadas) nos nós da rede

torna-se pouco provável (evento não determinístico). Com isto cada pacote de voz egresso da rede *IP* apresenta um valor aleatório do tempo de enfileiramento (atraso) na rede. O resultado é que o intervalo (tempo) entre os sucessivos pacotes de voz que chegam ao terminal de destino não é constante. A variação do intervalo (tempo) entre pacotes de voz provocada pela rede *IP* é o *jitter* (variação de atraso). Uma técnica usada para reduzir o *jitter* é através da inserção de *buffer* na recepção. O pacote permanece enfileirado neste *buffer* até que o intervalo entre pacotes saíntes fique constante. O *buffer* (recepção) para tratar o *jitter* também adiciona atraso no pacote de voz (atraso *dejitter buffer*).

No terminal de destino o sinal de voz sofre o atraso de decodificação e atraso de remontagem (desempacotamento) ambos referentes à conversão do sinal de voz digital em analógico. O atraso de codificação e decodificação do sinal de voz é constante (determinístico) e depende do *CODEC* em uso. O atraso do pacote de voz na rede *IP* e o atraso no *buffer* (amortecedor de *jitter*) são aleatórios (não determinísticos) e dependem da carga de tráfego na rede. O *budget* de atraso (atraso total) no sistema *VoIP* está representado na Fig. 2.5.

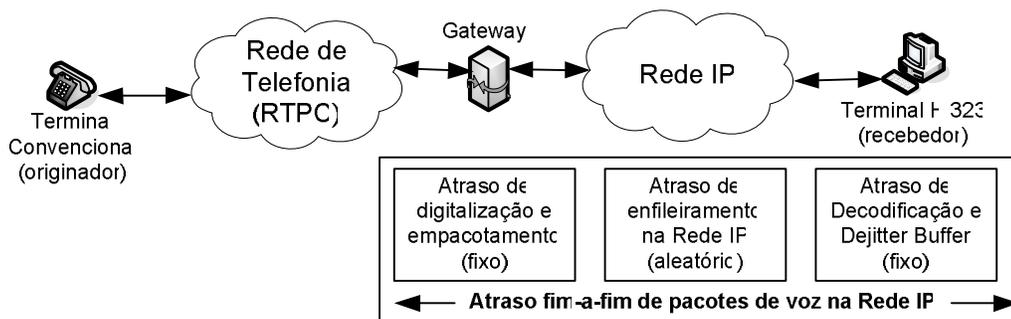


Figura 2.5 – Atraso fim-a-fim de cada pacote de Voz na rede *IP*.

A soma de todos os atrasos, em cada fase de encaminhamento de uma chamada *VoIP*, é o atraso fim-a-fim. O atraso fim-a-fim tem impacto na qualidade do sinal de voz e a rede *IP* é o maior ofensor pois gera eventos não determinísticos (aleatórios).

A recomendação *ITU-T G.114* [7] estabelece limites no tempo de transmissão de pacotes em redes *IP* (atraso fim-a-fim) para garantir, por exemplo, a interatividade nas conexões *VoIP* e permitir que os mecanismos de controle atuem adequadamente. Os valores recomendados são:

- 0 a 150ms: aceitável para a maioria das aplicações;
- 150 à 400ms: aceitável para conexões internacionais;
- 400ms: não aceitável para objetivos gerais de planejamento de rede. Em situações excepcionais este limite é excedido (ex. meio de transmissão satélite resulta 600ms);

A rede *IP* transporta os pacotes de voz, no entanto adiciona atraso de enfileiramento nos nós e *jitter* de atraso. Numa ligação *VoIP* (aplicação fim-a-fim) a rede *IP* insere perturbação no sinal de voz. O *ITU-T* determina os limites máximos do atraso fim-a-fim (perturbação) na rede e o perfil de aplicação suportado. A interatividade de uma ligação *VoIP* ocorre se o atraso fim-a-fim atinge até

200ms. Para garantir o funcionamento de uma ligação *VoIP* é preciso determinar o impacto de cada fase na inserção de atraso no pacote de voz. Estas fases estão descritas a seguir:

Meio de transmissão: insere **atraso de propagação** no pacote de voz. É o tempo que o sinal leva para se propagar através do meio de transmissão tais como fibra óptica, cabo coaxial, par metálico, espaço livre, etc. O **atraso de propagação** é constante (fixo) e é função do tipo do meio de transmissão empregado e também da distância que o sinal de voz percorre neste meio. Num enlace de fibra óptica o atraso de propagação é 50,0 ns/Km.

Codificação/Decodificação: no *CODEC* o sinal de voz analógico (fonte) é digitalizado e no destino é convertido em analógico novamente. Nesta fase é inserido o **atraso de codificação e decodificação** respectivamente. O **atraso de codificação/decodificação** é constante (fixo) e depende do **atraso de processamento** e do **atraso de lookahead**. O tamanho do quadro (*frame* de voz) e o tipo de algoritmo de codificação/decodificação usados impactam diretamente no **atraso de processamento**. É aproveitada também a correlação entre os quadros de voz sucessivos. Neste caso há o **atraso de lookahead** que é o tempo necessário para tratar o próximo quadro de voz depois de comparado com o quadro atual (em processamento). A tabela 2.1 relaciona 03 tipos de codificadores (*CODEC*) usados na conversão de frame de voz analógico em frame de voz digital com o **atraso de codificação/decodificação**.

Tabela 2.1 – Algumas especificações de *CODEC* usados em *VoIP*

<i>CODEC</i>	<i>G.723.1</i>	<i>G.729</i>	<i>G.729 A</i>
Taxa de <i>bits</i>	5,3 / 6,4 <i>kbps</i>	8 <i>kbps</i>	8 <i>kbps</i>
Tamanho de quadro	30 ms	10 ms	10 ms
Atraso de processamento	30 ms	10 ms	10 ms
<i>Lookahead delay</i>	7,5 ms	5 ms	5 ms
Comprimento do quadro	20 / 24 <i>bytes</i>	10 <i>bytes</i>	10 <i>bytes</i>
<i>DSP MIPS</i>	16	20	10,5
<i>RAM</i>	2200	3000	2000

O conjunto de especificações descrito na tabela 2.1 provê as condições operacionais do *CODEC*. Estes parâmetros permitem definir, além dos atrasos já descritos, a capacidade mínima de

hardware para suportar o *CODEC* e as características de geração (saída) do fluxo de frames de voz digital. Além dos parâmetros já descritos seguem:

1. O comprimento do quadro corresponde ao tamanho do quadro em *bytes* (sem o cabeçalho), ou seja, carga útil;
2. A taxa *DSP MIPS* define a capacidade (velocidade) mínima do processador para realizar a implementação do processador *DSP (Digital Signal Processing)* e do algoritmo de codificação. Quanto maior a complexidade deste algoritmo maior é o tempo de processamento.
3. O campo *RAM* especifica o tamanho mínimo de memória em termos de palavras de *16bits*;

Tempo de empacotamento: também definido como **atraso de empacotamento** é o tempo necessário para o *CODEC* gerar um número suficiente de fluxo de voz (em *bytes*) tal que o quadro de voz resultante preencha o *payload* (carga útil) de um pacote *IP*. Na Fig. 2.6, por exemplo, o *CODEC G.729* deve gerar um quadro de voz com *10 bytes* a cada *10ms*. Durante o empacotamento pode haver quadro de voz menor que a capacidade máxima do *payload* (carga útil) de *10 bytes*. Neste caso é necessário limitar o **atraso de empacotamento**. Uma alternativa é enviar os pacotes *IP* de voz parcialmente carregados. Esta técnica implica em reduzir a eficiência do sistema uma vez que o *payload* é completado com *bits* de enchimento (dados não úteis).

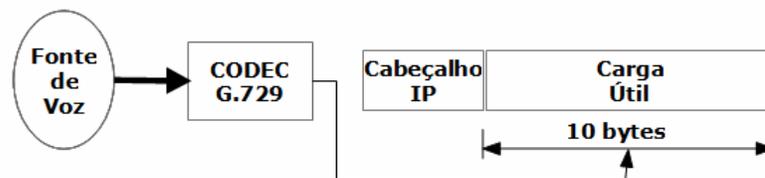


Figura 2.6 – *CODEC G.729* gera fluxo de voz (quadros de voz) para empacotamento *IP*.

Tempo de enfileiramento na rede *IP*: o pacote de voz na rede *IP* sofre **atraso de enfileiramento** nos roteadores (ou comutadores de pacotes) conforme indicado na Fig. 2.7. Este atraso, apresentado em [10], é não determinístico (variável) e depende da condição de uso de cada roteador. Cada pacote que chega permanece no *buffer* de entrada (fila) se o roteador está em uso. Caso contrário o roteador trata imediatamente o pacote. O tempo de permanência do pacote de voz no *buffer* de entrada é não determinístico. Em seguida o roteador efetua a comutação espacial e temporal entre a entrada e a saída (roteamento) para encaminhar o pacote. Esta atividade, também definida como tempo de chaveamento, compreende o tempo necessário para transferir o pacote do *buffer* de entrada para o *buffer* de saída. Outro fator que influi no tempo de enfileiramento é a capacidade do enlace de saída (*bits* por segundo). A conjunção da taxa de chegada de pacotes (entrada), tamanho do pacote de voz e a capacidade do enlace (saída) refletem na carga de tráfego resultante neste nó da rede (ou fator de utilização do enlace). Quanto maior a capacidade do enlace, mantida a taxa de

chegada de pacotes de voz, menor é o tempo de enfileiramento. O comportamento da rede *IP* resulta no pacote de voz um atraso total não determinístico (aleatório) e o atraso de enfileiramento é o principal responsável por esta característica. Em situações de congestionamento (fator de utilização elevado) o tempo de enfileiramento torna a rede *IP* inviável para suportar o serviço de voz.

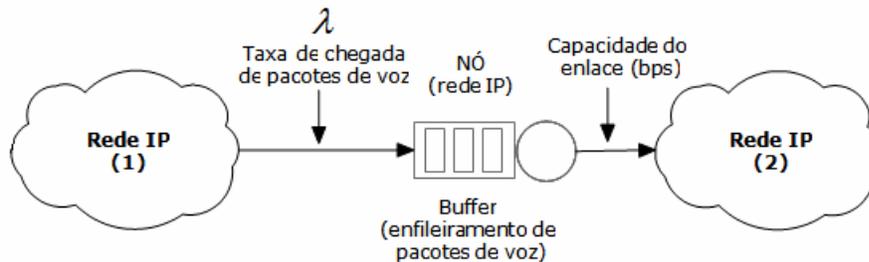


Figura 2.7 – Atraso de enfileiramento de cada pacote de voz na rede *IP*.

2.3.3 Atraso *Dejitter Buffer*

Tempo de *Dejitter Buffer*: no terminal de origem o intervalo entre os pacotes de voz é constante. Em seguida os pacotes são transportados pela rede *IP* que multiplexa pacotes de tamanhos diversos (ex. voz e dados). A rede, através do atraso de enfileiramento, provoca perturbação no intervalo entre pacotes de voz subjacentes. No terminal receptor (pacotes egressos da rede) o intervalo entre os pacotes de voz que chegam não é constante (*jitter*). Outro fator que contribui para o *jitter* é que a rede *IP* transporta tanto pacote de dados (ex. 1400 bytes) quanto pacote de voz (ex. 50 bytes) o que potencializa o atraso de enfileiramento. Pacote maior (ex. dados) demanda maior tempo para encaminhamento nos nós da rede e enquanto o pacote menor (ex. voz) é penalizado e permanece no *buffer* (nó da rede) por mais tempo até atendimento. Para tornar o intervalo entre pacotes de voz constante é necessário o uso de *buffer* no terminal receptor. Os pacotes são armazenados no *buffer* (recepção) e em seguida enviados, em intervalos regulares, ao decodificador. O tempo que o pacote permanece no *buffer* é o **atraso de *dejitter buffer***. Se o *jitter* provocado pela rede for elevado o ***dejitter buffer*** insere atraso adicional significativo (inviável) para compensar esta perturbação. É necessário estabelecer um limiar (valor máximo) de **atraso do *dejitter buffer*** para que o atraso total da rede suporte o serviço de voz. A Fig. 2.8 representa a inserção do atraso do *Dejitter Buffer* para eliminação do *jitter* inserido pela rede *IP* nos pacotes de voz egressos.

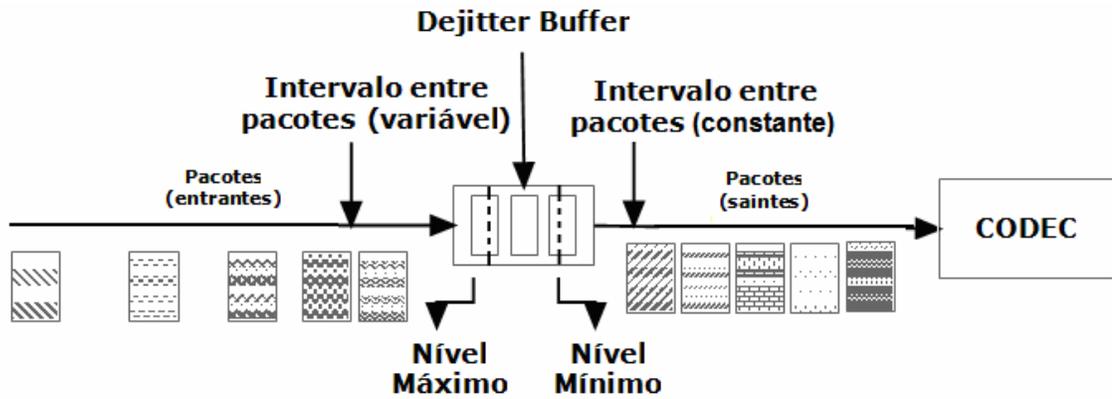


Figura 2.8 – Tratamento dos pacotes de voz no terminal receptor (Dejitter Buffer)

2.3.4 Perda de Pacotes de Voz

A rede *IP* permite otimizar a utilização de recursos através da agregação (multiplexagem) estatística dos pacotes *IP*. Nesta técnica de multiplexagem a rede transmite o sinal de voz, mas não garante a entrega dos pacotes de voz no terminal receptor. Para compensar este comportamento da rede é indicado o uso de protocolos de transporte confiáveis (ex. *TCP*). Quando há perda de pacotes na rede *IP* o protocolo *TCP* efetua a retransmissão dos pacotes perdidos. O serviço de *VoIP* é uma aplicação que requer interatividade em tempo real por isso o *TCP* não deve ser usado. Desta forma a perda de pacotes de voz é um evento intrínseco da rede *IP* (inevitável) e tem impacto significativo na qualidade do sinal de voz na recepção. A métrica utilizada para aferir a perda de pacotes (em %) é o percentual dos pacotes de voz transmitidos que foram perdidos (descartados) na rede *IP*. A Fig. 2.9 representa a perda e pacotes no nó da rede *IP*. Na Fig. 2.9 a perda de pacotes resultante é $\frac{1}{9} \times 100\% \cdot ou \cdot 11,11\%$.

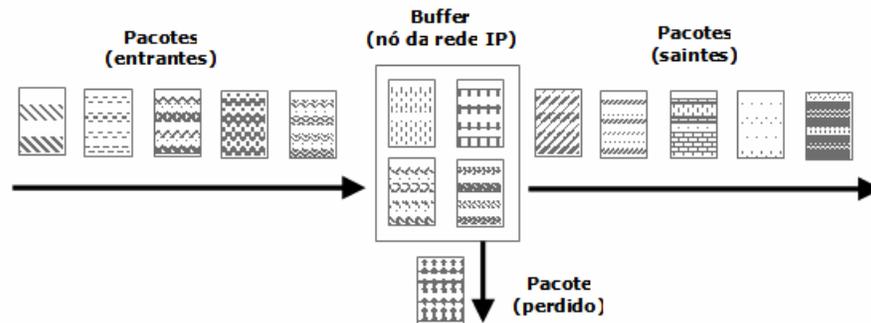


Figura 2.9 – Perda de pacote no *buffer* (nó da rede)

A melhor abordagem para tratar perda de pacotes é entender os principais eventos causadores que seguem:

- 1. Degradação no meio de transmissão:** é comum haver erro (falha) de configuração de *clock* na rede (meio físico) e ocorrer *slipping* (escorregamento na sincronização). Esta perda de sincronismo resulta em perda de pacotes na rede sempre que o sistema busca a resincronização. Outra situação comum é ocasionada por *fading* (desvanecimento) ou também por interferência em sistema de rádio enlace. Estas perturbações ocasionam taxa de erro no meio de transmissão o que resulta em pacotes de voz corrompidos na recepção que são descartados (perdidos). Qualquer meio de transmissão está sujeito à degradação que depende de suas respectivas características físicas.
- 2. Fator de utilização (roteadores):** valores típicos da carga de tráfego resultante em cada enlace de saída de um roteador são de até 70% (fator de utilização). Acima deste limiar há incremento significativo na ocupação dos *buffers*. Quando o fator de utilização atinge 100% ou mais da capacidade do enlace de saída (congestionamento) ocorre *overflow do buffer* do roteador. Neste caso há perda de pacotes de voz.
- 3. Inserção de Jitter (perturbação da rede IP):** a condição de uso da rede *IP* tem impacto direto no nível de *jitter* dos pacotes de voz *IP* egressos da rede. O nível de *jitter* não deve ultrapassar a capacidade máxima de compensação pré-estabelecida do *buffer* na recepção. Em caso de *jitter* excessivo ocorre *overflow* no *buffer* da recepção e conseqüente perda de pacotes de voz.
- 4. Rajadas de Tráfego (ausência de formatação):** o tráfego de pacotes na rede *IP* ocorre por rajadas (*burst*). Na transmissão de pacotes de voz as rajadas de tráfego (*bytes* por segundo) na rede *IP* podem exceder a capacidade do enlace na recepção. O tráfego excedente é descartado. É importante implementar nos elementos terminais da rede a função de *traffic shapping* (modelador de tráfego). Esta facilidade efetua o controle das rajadas de tráfego na transmissão que sejam compatíveis com a capacidade do enlace na recepção.
- 5. Atraso excessivo:** caso o parâmetro *TTL (Time-To-Live)* que é o tempo máximo pré-estabelecido de permanência do pacote de voz na rede *IP* seja excedido, este pacote é descartado. Uma das causas para este evento é o tempo excessivo de enfileiramento em cada nó da rede. Este parâmetro indica também condição de congestionamento em um ou mais nós da rede *IP*.

Em *VoIP* a perda de pacotes na rede *IP* resulta em degradação do sinal de voz. No terminal receptor o pacote de voz é convertido em sinal analógico. Na fase de remontagem dos pacotes de voz ocorrem “picotes” (falsos silêncios) na conversação telefônica causados pela ausência de informação (ou perda de *frames* de voz). Numa conexão *VoIP* é tolerável a perda na faixa 3% a 5% dos pacotes de voz transmitidos na rede *IP*. O *CODEC* do receptor mascara (compensa) a perda de informação (*frames* de voz) e aplica esquemas de regeneração. Neste caso o *CODEC* receptor pode ser provido de algoritmo que preenche com amostras de voz (quadros de voz) já recebidas quando há perda de pacote. O algoritmo substitui a amostra de voz perdida pela amostra anterior. A

recomendação de perda máxima de pacotes *IP* (voz) para manter a inteligibilidade tolerável numa conversação de telefonia *VoIP* é de até 10%. O esquema de compensação do *CODEC* receptor só é eficaz quando há perda individual de pacote. Este esquema de regeneração do sinal de voz não suporta a perda em rajadas dos pacotes de voz. Para tratar perdas de pacotes em rajadas duas funcionalidades importantes podem ser implementadas no receptor: ***Packet Loss Concealment*** e ***Playout Scheduling***.

- ***Packet Loss Concealment***: é a recuperação (regeneração) da informação perdida na rede *IP* (pacotes de voz) com base na redundância do sinal de áudio recebido (ou pacotes subjacentes). A estratégia é camuflar a informação perdida com algo que seja similar ao sinal original (fonte). Uma técnica para tratar perda de pacotes de voz em rajadas é estimar o sinal de voz perdido em cada pacote. Neste caso são geradas “falas” sintéticas para recuperar a perda de informação.
- ***Playout Scheduling***: esta técnica requer o uso de *playout buffer* (ou *de jitter buffer*) no receptor que absorve o *jitter* antes da saída de áudio. Um algoritmo típico de *playout buffer* monitora o tempo de chegada de cada pacote a partir do *time stamp* (extraído do cabeçalho do *RTP*) trata os pacotes recebidos. Compara o tempo de chegada com o relógio local (receptor), mede o atraso de cada pacote e calcula o intervalo entre pacotes.

O uso combinado das duas funcionalidades permite tratar *talkspurts* individuais.

2.3.5 Arquitetura de *VoIP*

A definição da arquitetura *VoIP* é referente aos equipamentos (*PC* ou *Gateway*) utilizados para estabelecer uma ligação *VoIP*. Neste caso a ligação pode ser estabelecida diretamente entre dois terminais *PC* sem o uso de *Gateway* e/ou *Gatekeeper*. A ligação provida através de *Gateway* permite maior flexibilidade na conexão entre os equipamentos terminais. Cada tipo de equipamento terminal é conectado à rede *IP* por condições específicas. Por exemplo, um telefone convencional necessita de um *ATA* (adaptador) para conectar-se à rede *IP* e é suportado por *Gateway*. A arquitetura utilizada depende também da implementação de *VoIP* empregada se baseada por exemplo no protocolo *SIP* ou no *H.323*. A definição e composição de cada arquitetura para implementação de *VoIP* está descrita a seguir.

- **Arquitetura *PC-a-PC***: nesta arquitetura são necessários dois computadores suportados por recursos de multimídia e que estejam conectados à rede *IP*. Estes computadores podem se conectar diretamente à uma *LAN* (rede local), ou através de acesso discado (ligação telefônica via *RTPC*), ou acesso *ADSL* (*Asymmetric Digital Subscriber Line*) ou ainda através de Cable Modem (TV a cabo) à um provedor de serviços *Internet* para transmitir sinais de voz. Os computadores efetuam a codificação (amostragem, compressão e empacotamento) do sinal de voz analógico (fonte) e estabelecem a chamada de voz especificando o endereço *IP* de destino. Na Fig. 2.10 os dois terminais (*PC*) efetuam ligação telefônica (*VoIP*) somente no ambiente da rede *IP*. É apresentado o caso de uma

ligação direta *PC-a-PC* (conexões 1 e 2) ou através de consulta ao *Gatekeeper* (conexões 1,3 e 2 respectivamente).

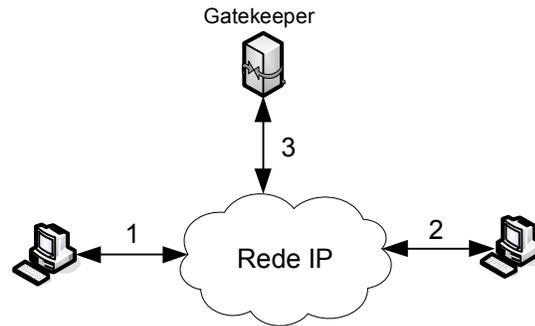


Figura 2.10 – Arquitetura *PC-a-PC*

- **Arquitetura com gateway:** nesta arquitetura, representada na Fig. 2.11, os telefones convencionais analógicos efetuam e recebem ligações telefônicas sobre a rede *IP*. O terminal telefônico chamador estabelece uma conexão telefônica convencional, através da RTPC, com o *gateway* mais próximo de sua central telefônica (*gateway* chamador). O *gateway* chamador solicita do terminal chamador o número de telefone de destino (terminal telefônico receptor). O *gateway* chamador, após encontrar o *gateway* mais próximo possível da central telefônica do terminal telefônico receptor (*gateway* receptor), estabelece uma seção para transmissão de sinais de voz. O *gateway* receptor efetua uma ligação telefônica convencional até o terminal telefônico receptor e após seu completamento (atendimento), a comunicação fim-a-fim tem início, o sinal de voz é então transmitido entre os *gateways* (originador e receptor) através de pacotes *IP*. O processo de codificação/empacotamento e decodificação/desempacotamento é realizado respectivamente nos *gateways* originador e/ou receptor.

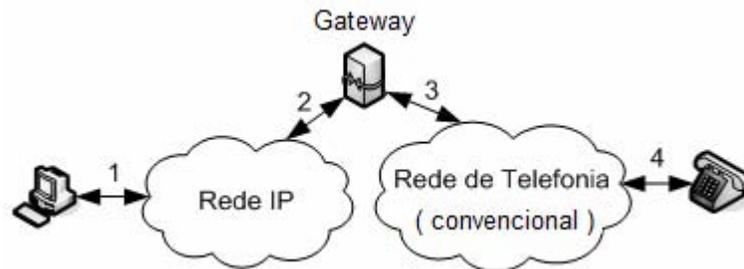


Figura 2.11 – Arquitetura com *Gateway*

- **Arquitetura híbrida:** a possibilidade de uso combinado das arquiteturas anteriores para a realização de *VoIP* caracterizam os esquemas híbridos e são desejáveis. A utilização híbrida ocorre se um telefone analógico origina (ou recebe) uma ligação para um terminal

PC (telefone *IP*). Neste caso deve haver uma entidade, em muitos casos é o próprio *gateway*, que efetue a tradução ou mapeamento de endereços *IP* em números telefônicos (plano de numeração convencional). As possibilidades de interconexão, formadas por caminhos unidirecionais, para o provimento de *VoIP* a partir das arquiteturas descritas são: *PC-a-PC*, *gateway-a-gateway*, *PC-a-gateway* (híbrida) e *gateway-a-PC* (híbrida).

2.3.6 Protocolos para *VoIP*

O estabelecimento de uma ligação telefônica através da rede *IP* requer o uso de protocolos *VoIP*. Estes protocolos permitem a transmissão do sinal de voz do terminal originador até terminal receptor. Os principais protocolos *VoIP* recomendados para uso na rede *IP* são *RTP*, *H.323* e *SIP*.

2.3.6.1 Protocolos *RTP* e *RTCP*

O protocolo *RTP* (*Real-time Transport Protocol*) provê um formato de pacote padrão para a transmissão de voz em redes *IP*. Foi desenvolvido pelo grupo de trabalho do IETF para transporte de áudio (voz) e vídeo e sua última versão está descrita na *RFC 3550*. Os serviços providos pelo *RTP* são:

- Identificação do tipo de carga útil: identifica os pacotes de voz;
- Numeração seqüencial dos pacotes de voz: é utilizada para reordenação de pacotes (no ponto receptor);
- Sincronização temporal (Time stamping): permite sincronização e cálculo de *jitter*;
- Monitoramento da entrega dos pacotes;

O *RTP*, em conjunto com os protocolos de transporte *TCP* ou *UDP*, provê funcionalidades da camada de transporte (modelo *OSI*). O *TCP* recupera pacotes perdidos, elimina pacotes duplicados, trata dados corrompidos e restaura uma ligação em caso de falhas na rede. Por outro lado o protocolo *UDP*, usado em aplicações que demandam interatividade, não garante a entrega dos pacotes e nem efetua retransmissões. A premissa do *UDP* é que a aplicação controle os pacotes de voz e efetue as correções necessárias. A escolha do protocolo de transporte é função dos requisitos de cada aplicação. Uma ligação *VoIP* demanda interatividade e é mais sensível ao atraso do que à perda de pacotes na rede *IP*. Desta forma o uso do protocolo *UDP* em conjunto com o *RTP* atende as premissas da aplicação *VoIP*. O *RTP* trata o sinal proveniente dos *CODEC*, efetua a fragmentação do fluxo de voz (quadros de voz), insere em cada fragmento a seqüência (numeração) e o de tempo máximo para entrega do pacote no terminal receptor. Para isto insere no campo *TTL* (Time-to-Live) o tempo máximo (pré-estabelecido) de permanência do pacote de voz na rede *IP*. A aplicação de *VoIP* suporta atraso de até 200ms e o uso do campo *TTL* permite descartar o pacote com tempo de propagação superior a este limite.

O protocolo *RTCP* (*Real-Time Transport Control Protocol*) efetua função complementar de controle e não transporta sinal de voz. Este protocolo transmite apenas pacotes de controle aos componentes de *VoIP* usados na ligação telefônica. O protocolo *RTCP* provê informações da qualidade de serviço que o *RTP* estabelece numa ligação *VoIP*. O *RTCP* coleta, a partir das

condições de uso da rede *IP*, as estatísticas da conexão *VoIP* tais como *bytes* enviados, pacotes enviados, pacotes perdidos, *jitter*, realimentação e atraso fim-a-fim e repassa estas informações para os terminais envolvidos na ligação. As informações de desempenho da aplicação *VoIP* na rede *IP* fornecidas pelo RTPC é um indicador do nível de serviço provido pela própria rede *IP* para a transmissão do sinal de voz.

2.3.6.2 Protocolo *H.323*

O *H.323* é um padrão que especifica os componentes, protocolos e procedimentos para o provimento de serviços de comunicação multimídia, por exemplo, *VoIP*, que demandam interatividade. A Fig. 2.12 representa uma ligação *VoIP* estabelecida por terminais *H.323* conectados à rede *IP* e que utilizam a plataforma *H.323* para efetuar uma chamada *VoIP*.

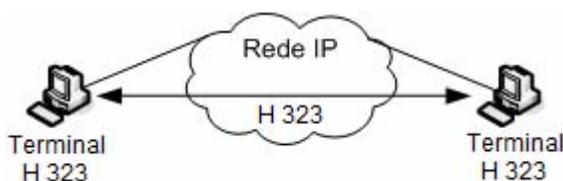


Figura 2.12 – Ligação *VoIP* entre terminais *H.323*

O padrão *H.323* especifica os tipos de componentes que interligados à rede *IP* suportam serviços de voz fim-a-fim ou ponto-multiponto. Os elementos são:

- 1- Terminais (*H.323*);
- 2- Gateways (*H.323*);
- 3- Gatekeepers (*H.323*);
- 4- Unidade de Controle Multiponto ou *MCU* (*H.323*);

A funcionalidade de cada componente do padrão *H.323* já foi apresentada no item 2.2 à exceção da *MCU H.323*. A *MCU* possibilita a interligação (conferência) de 03 ou mais terminais. Todos os terminais que participam de uma ligação, neste caso conferência, estabelecem uma conexão lógica com a *MCU*. Neste cenário a *MCU* gerencia os recursos para o estabelecimento da conferência, negocia entre os terminais *H.323* o tipo de *CODEC* a ser usado e permite controlar o envio de fluxo de quadros (sinais de voz). O *Gatekeeper*, *Gateway* e a *MCU* são componentes que efetuam funcionalidades distintas porém no padrão *H.323* podem ser implementados num mesmo dispositivo físico.

Uma zona *H.323* é o conjunto de todos os terminais, *Gateways* e *MCU's* que são gerenciados por um único *Gatekeeper* conforme ilustra a Fig. 2.13. Neste caso os terminais 1 e 2, os *Gateways* 1, 2 e 3 pertencem à mesma zona 1 que é gerenciada pelo *Gatekeeper* 1.

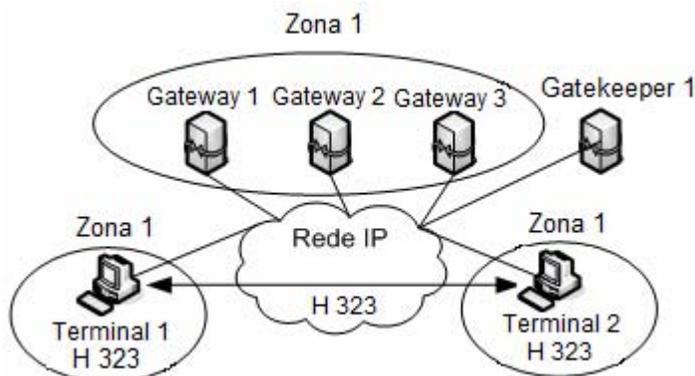


Figura 2.13 – *Gatekeeper* e zona *H.323*

Para o estabelecimento de uma ligação *VoIP* fim-a-fim é necessário codificar a voz, configurar uma chamada, efetuar troca de sinalização e transportar os quadros de voz (sinal de voz). Estas funcionalidades são providas por um conjunto de protocolos. Por esta razão o padrão *H.323* é considerado uma arquitetura de telefonia na rede *IP*. O *Gateway* interconecta a rede *IP* à RTPC (rede de telefonia convencional). Faz uso e tradução dos protocolos da plataforma *H.323* na rede *IP* e dos protocolos RTPC no lado da telefonia convencional para o estabelecimento e desconexão de uma ligação telefônica. Uma ligação direta entre 02 terminais *H.323* (ex. Personal Computer) não requer necessariamente o uso do *Gateway* e *Gatekeeper*. Para tornar obrigatório o uso destes componentes, em qualquer ligação, é necessário implementar regras específicas na zona *H.323*

Uma chamada *VoIP* é digitalizada pelo *CODEC* (ex. *G.723.1*). O sistema *H.323* suporta qualquer padrão de *CODEC*. O protocolo de controle de sinalização *H.245* é usado, numa conexão *VoIP*, para trocar mensagens entre os terminais originador e receptor. Os terminais *H.323* interagem através de mensagens específicas e negociam a capacidade de troca de pacotes de voz (taxa de transferência), abertura e fechamento de canais lógicos para transportar fluxos de sinais de voz e geram mensagens de controle de fluxo. O protocolo *RTP* (*Real-Time Transport Protocol*) provê o serviço de entrega fim-a-fim do sinal de voz e garante interatividade. O *RTP* fornece identificação do tipo de carga útil, efetua numeração seqüencial, insere marcação temporal e monitora a entrega dos datagramas de voz. O protocolo *RTCP*, em complemento ao *RTP*, informa ao terminal originador a qualidade de transmissão que a rede *IP* provê para o transporte dos pacotes de voz.

No plano de controle é necessário também o protocolo *ITU Q.931* que estabelece e encerra conexões, fornece tons de discagem, gera sons de chamada e outras funcionalidades da telefonia convencional. A comunicação com o *Gatekeeper H.323* é estabelecida através do protocolo *H.225*. Este protocolo gerencia o canal lógico estabelecido entre o terminal (ex. *PC*) e o *Gatekeeper* e é denominado canal *RAS* (*Registration/Admission/Status*). Através do canal *RAS* o *Gatekeeper H.323* estabelece e desfaz conexões entre os terminais que pertencem à sua zona (domínio) e solicitam e

disponibilizem largura de banda, forneçam atualizações da condição de uso (*status*) de seus respectivos domínios (zona) dentre outras funcionalidades. O protocolo *RTP* efetua a transmissão propriamente dita do sinal de voz (carga útil). O controle sobre este protocolo é através do protocolo *RTCP*.

Como exemplo, seja uma ligação *VoIP* na arquitetura *H.323* que inicia em um terminal *PC H.323* e estabelece uma conexão em um terminal telefônico convencional (RTPC). O *PC* transmite por difusão (*broadcasting*) pacotes UDP e define o *Gatekeeper* da zona. Em seguida o *Gatekeeper* ecoa e o *PC* (terminal *H.323*) identifica o endereço *IP* deste *Gatekeeper*. O *PC* se registra e envia uma mensagem RAS (*H.225*) pela conexão UDP. O *Gatekeeper H.323* efetua o registro e admissão do *PC*. Através do canal RAS (*H.225*) o *PC* solicita largura de banda. O *Gatekeeper* verifica condições de uso da rede *IP* e se houver disponibilidade de recursos provê a largura de banda solicitada pelo *PC*. Em seguida o *PC* cria uma conexão TCP com o *Gatekeeper* para configurar a chamada telefônica. Após alocação de banda o *PC* envia uma mensagem *SET UP (Q.931)* pela conexão TCP. Esta mensagem define o terminal receptor, neste caso o número do telefone convencional (destino).

Se o terminal receptor for *PC* é definido o número *IP* deste terminal. O *Gatekeeper* responde com a mensagem *CALL PROCEEDING (Q.931)* e confirma a solicitação. Neste momento o *Gatekeeper H.323* encaminha mensagem de *SETUP* para o *Gateway H.323* da zona. O *Gateway H.323* inicia uma ligação convencional e usa os protocolos da telefonia convencional (plataforma RTPC) e no lado da rede *IP* envia mensagem de *ALERT (Q.931)* e informa o início da chamada ao *PC* originador. O terminal receptor (telefone convencional) atende a chamada, o *Gateway* envia ao *PC* originador mensagem de *CONNECT (Q.931)* e estabelece a conexão. O *Gatekeeper H.323* deixa de interagir e permanece na conexão somente o *Gateway H.323*. O protocolo *H.245* define os parâmetros da chamada unidirecional entre os terminais (originador e receptor) e, por exemplo, atribui para cada terminal um mesmo tipo de *CODEC* e outros requisitos necessários.

Após a definição de parâmetros da conexão o protocolo *RTP* inicia a transferência do fluxo de voz. Em paralelo o protocolo *RTCP* gerencia a transmissão do sinal de voz (fluxo de quadros) e controla o tráfego resultante para evitar congestionamento na plataforma *H.323*. São criados canais lógicos que são utilizados no estabelecimento de uma conexão *VoIP*. No encerramento da chamada *VoIP* um dos terminais (ou o *PC* ou o telefone convencional) finaliza a ligação e o *Gateway* trata as informações (tanto da plataforma RTPC quanto da plataforma *H.323*) e neste caso o protocolo *ITU Q.931*(canal de sinalização) desfaz a conexão. Na rede *IP* o *PC* interage através do canal RAS (*H.225*) com o *Gatekeeper H.323* para liberar a largura de banda concedida (ou recursos da rede *IP*).

2.3.6.3 Protocolo SIP

O protocolo *SIP (Session Initiation Protocol)* definido na *RFC 3261* é resultado do esforço do *IETF* em simplificar a transmissão de voz sobre a rede *IP*. Enquanto a arquitetura *H.323* é um conjunto de protocolos agregados a solução *SIP* representa um único protocolo. No protocolo *SIP* o número do telefone pode ser uma *URL* (ou página da *WEB*). Desta forma através de um *link*, pode

ter início uma conexão *VoIP*. É possível estabelecer sessões entre duas (chamada entre A e B) ou mais partes (conferência à três) para o provimento de chamadas telefônicas. O protocolo *SIP* configura, gerencia e encerra as sessões. Os protocolos *RTP* e *RTCP* são usados na transmissão do sinal de voz. Os protocolos *TCP* e *UDP* são usados pelo *SIP* de acordo com a necessidade da aplicação e condições de uso da rede *IP*. A estrutura do protocolo *SIP* é baseada em texto e utiliza o *HTTP*. Uma mensagem *SIP* contém texto em *ASCII* composto por Métodos (1ª linha) e Cabeçalhos para a Passagem de Parâmetros (demais linhas). Os métodos são descritos a seguir conforme o uso no estabelecimento de uma conexão fim-a-fim do tipo *VoIP*.

Antes do início de qualquer sessão *SIP* o método *OPTIONS* é usado para definir se o terminal originador e receptor têm aplicativos (recursos) que suportem uma sessão *VoIP* além de verificar se há ou não sessão *SIP* em uso.

Para estabelecer uma sessão, o terminal originador cria uma conexão *TCP* ou *UDP* com o terminal receptor e envia a mensagem *INVITE*. A mensagem contém as linhas (com cabeçalhos) que definem os recursos do originador, tipos de mídia e formatos. O terminal receptor aceita a chamada e informa seus recursos, tipos de mídia e formatos. A conexão *SIP* é realizada em três etapas (métodos) consecutivas e acontecem na seguinte ordem:

- 1- *INVITE*: iniciada pelo originador, solicita o início de uma sessão *SIP*;
- 2- *OK*: terminal receptor ecoa ao terminal originador;
- 3- *ACK*: O terminal chamador envia, por último, a confirmação do estabelecimento de uma sessão *SIP* (conexão *VoIP*).

Para terminar a sessão um dos terminais envia a mensagem do método *BYE*. A sessão é encerrada quando o outro terminal envia a mensagem de *ACK*. O protocolo *SIP* também provê a interconexão entre a rede *IP* e a rede *RTPC*. Para isto é necessário um *Gateway SIP* que execute estas funcionalidades. Assim como na arquitetura *H.323* o *Gateway SIP* também usa protocolos *SIP* na rede *IP* e protocolos de telefonia convencional (plataforma *RTPC*) para estabelecer ligações telefônicas híbridas.

2.4 Conclusão:

Neste capítulo foram apresentados os principais componentes *VoIP* para estabelecimento de uma ligação telefônica. O uso destes componentes possibilita a transmissão de voz através da rede *IP*. O sinal de voz no receptor sofre distorção devido às condições de tráfego da rede. Neste caso a rede *IP* insere, por exemplo, atraso e *jitter* nos pacotes de voz. Uma ligação *VoIP* requer interatividade em tempo real e o atraso na rede *IP* implica em perda de qualidade na conexão.

A perda de pacotes em rajadas na rede *IP* é outro fator que degrada significativamente a recepção do sinal de voz. No receptor há mecanismos apropriados para tratar estas perturbações que são intrínsecas da rede e estão presentes no sinal de voz (recepção). Um exemplo é a proposta de inserir no receptor um sistema adaptativo (dinâmico) de reprodução e compensação das perdas de tráfego de voz por rajadas na rede *IP*. Em testes esta proposta proveu resultados objetivos e satisfatórios sobre desempenho do serviço de *VoIP*. Outro ponto importante é a arquitetura usada

em *VoIP* que relaciona os componentes e protocolos *VoIP* necessários para realizar uma conexão telefônica. Em alguns casos o *gateway* e *gatekeeper* são dispensáveis. O protocolo *RTP* está presente tanto na arquitetura *H.323* quanto na arquitetura *SIP*. Este protocolo (*RTP*) em conjunto com o protocolo *RTCP* transmite o fluxo de voz e permite a avaliação no receptor do impacto da rede *IP* no sinal de voz e implementa mecanismos de compensação. Uma ligação *VoIP* pode tanto utilizar a arquitetura *H.323* (conjunto de protocolos) quanto ser suportada pelo protocolo *SIP*. A opção pelo protocolo *SIP* se deve à simplicidade de sua aplicação.

Capítulo 3

Modelagem Analítica do tráfego de Voz em redes *IP*

3.1 Introdução

Em muitos casos o tráfego em redes *IP* é modelado como auto-similar. Nesta linha de estudo o tráfego é tratado como rajadas com diversas durações em diferentes escalas de tempo. A distribuição do tráfego *IP*, no modelo auto-similar, permanece a mesma e independe da escala de tempo usada. Esta abordagem de modelamento matemático e análise de desempenho é estudada em [7], e quando aplicada em redes *IP*, torna-se muito complexa.

No entanto, em recentes estudos experimentais, é verificado em [8] e [9] que quando ocorre alta agregação de tráfego, por exemplo, um nó da rede *IP* o tráfego *IP* de entrada pode ser modelado como uma distribuição de Poisson. Com esta abordagem a distribuição do serviço independe do tráfego de entrada se considerado uma escala de tempo pequena. Portanto, baseado nas considerações de [8] e [9], a teoria clássica de filas será usada para análise de desempenho de tráfego de voz e dados em redes *IP*. O principal motivador é o crescente tráfego de voz na rede *IP* (*VoIP*) que demanda priorização e tem impacto no desempenho desta rede. O parâmetro de análise é o tempo de espera no “buffer” de cada nó da rede.

A rede *NGN* (*Next Generation Network*), que implementa *VoIP*, é utilizada pelas operadoras de telecomunicações para compor, de forma híbrida, a plataforma que suporta o serviço de telefonia convencional. O objetivo, no emprego desta tecnologia, é otimizar, através da multiplexagem estatística da rede *IP*, o uso de recursos para transmitir voz. Estas operadoras, por exemplo, fornecem serviços de telefonia para *call centers* que geram tráfego de voz em rajadas a partir de discadores automáticos (*call centers* ativos). O perfil de tráfego resultante, neste caso, não condiz com o modelo tradicional de uso da rede de telefonia. Esta é uma situação crítica de uso da rede de telefonia em conjunto com a rede *IP*. Este é um exemplo de uso da rede *IP* e de *VoIP* que demanda tratativas apropriadas.

Neste capítulo são apresentados os modelos matemáticos e o estudo do atraso dos pacotes *IP* (voz e dados). Os 02 tipos de tráfego considerados são o tráfego de tempo real (voz) e tráfego elástico (dados). O cenário de análise é comparar, através de parametrização específica, o atraso resultante para o tráfego de voz e dados considerando-se o ambiente de melhor esforço (sem

priorização) e quando há aplicação de *QoS* (com priorização) na rede *IP*. Em seguida é feita a análise dos resultados, através das equações associadas, sobre o impacto do tráfego de voz na rede *IP* e ao final são apresentadas as conclusões.

3.2 Estudo do Atraso em redes *IP* – Modelagem Matemática

O estudo de desempenho da rede *IP* proposto verifica o impacto da crescente migração do tráfego de voz da rede de telefonia convencional para a plataforma *IP*. A *NGN* (Next Generation Network) é um exemplo desta tendência uma vez que usa a rede *IP* e efetua, através de *gateways* específicos, interconexão com a rede de telefonia convencional (RTPC). A rede de telefonia *NGN* suporta *VoIP* e utiliza a rede *IP* como plataforma para transmitir o sinal de voz. Os pacotes de voz provenientes da *NGN* e os pacotes de dados são encaminhados de nó em nó na rede *IP*. Cada nó da rede *IP* é composto em [10] do conjunto servidor e buffer que enfileiram cada pacote para posterior encaminhamento. O parâmetro em análise é o tempo médio que cada pacote permanece no buffer em cada servidor (nó) da rede *IP*.

Para análise de desempenho na rede *IP* é necessário caracterizar as fontes de tráfego (voz e dados) que geram os pacotes na rede. Tanto para as fontes de voz quanto para as fontes de dados são consideradas, em cada caso, respectivamente a taxa de chegada (distribuição Poissoniana) e a taxa de partida dos pacotes que ingressam em cada nó da rede *IP*.

No equacionamento a taxa de chegada de voz e a taxa de chegada de dados são denotadas respectivamente por λ_v e λ_D . Estas taxas são medidas em pacotes por segundos. O tamanho do pacote de voz é denotado por $1/\mu_v$ e o tamanho do pacote dados é denotado por $1/\mu_D$ e, neste caso, ambos são medidos em bytes. As distribuições de chegada tanto dos pacotes de voz quanto de dados são de Poisson. Complementarmente é necessário definir o tamanho médio dos pacotes de voz e dados. A distribuição do tamanho dos pacotes de voz é fixo (constante), uma vez que os quadros de voz têm tamanho fixo. Já o tamanho médio dos pacotes de dados é uma distribuição exponencial negativa.

É estabelecido que x e y são as proporções que representam, respectivamente, os percentuais do tráfego de voz e o comprimento médio dos pacotes de voz com relação ao tráfego de dados e o comprimento médio dos pacotes de dados. A partir destas considerações as seguintes equações podem ser escritas:

$$\lambda_v = x \cdot \lambda_D \quad \mu_v = y \cdot \mu_D \quad (1)$$

Desde que o tamanho médio dos pacotes de voz, em geral, é menor do que o tamanho médio dos pacotes de dados, resulta da Eq.1 que $y > 1$. Por outro lado, a taxa de chegada dos pacotes de voz é, em geral, menor do que a taxa de chegada dos pacotes de dados. No entanto, num futuro próximo, com o crescimento do tráfego *VoIP* a taxa de chegada dos pacotes de voz pode tornar-se

tão intensa quanto a taxa de chegada dos pacotes de dados e neste cenário a variável x ficará maior do que 1. Além disto, a carga de tráfego resultante pode ser expressa por:

$$\rho_D = \frac{\lambda_D}{C \cdot \mu_D} \quad \rho_V = \frac{\lambda_V}{C \cdot \mu_V} \quad (2)$$

Onde ρ_D e ρ_V são as cargas de tráfego para dados e voz, respectivamente, e C é a capacidade do enlace de saída em bits por segundos (*bps*). A Eq.1 pode ser inserida na Eq.2, e desta forma a Eq.3 é obtida. A carga de tráfego de voz é estabelecida como uma fração da carga de tráfego de dados.

$$\rho_V = \frac{x \cdot \lambda_D}{C \cdot y \cdot \mu_D} \quad (3)$$

A equação 3 expressa também a dependência da carga de tráfego de voz em relação à carga de tráfego de dados. Esta equação permite a parametrização entre o tráfego de dados e o tráfego de voz. A seguir a equação 4 relaciona a carga de tráfego total aplicada num enlace de capacidade C (*bps*). Na rede *IP*, atualmente, a carga de tráfego de voz é bem menor que a carga de tráfego de dados, isto é, $\rho_V \ll \rho_D$. No entanto esta situação é transitória, pois há fortes perspectivas de migração das chamadas telefônicas convencionais para serem suportadas pela Internet. Este serviço, conhecido como *VoIP*, é atrativo principalmente pela redução no custo e otimização de recursos das redes de telecomunicações.

$$\rho_{TOTAL} = \rho_V + \rho_D \quad (4)$$

Inserindo-se as Eqs. 2 e 3 na Eq.4 a carga total de tráfego pode ser re-escrita como a apresentada na Eq.5.

$$\rho_{TOTAL} = \frac{x \cdot \lambda_D}{C \cdot y \cdot \mu_D} + \frac{\lambda_D}{C \cdot \mu_D} \quad (5)$$

A equação 5, se o termo $\frac{\lambda_D}{C \cdot \mu_D}$ é deixado em evidência, pode ser também expressa como:

$$\rho_{TOTAL} = \rho_D \cdot \left(\frac{x}{y} + 1 \right) \quad (6)$$

A equação 6 representa a carga de tráfego total usando apenas o parâmetro carga de tráfego de dados e as variáveis x e y .

Um nó da rede *IP* pode ser modelado como um elemento de rede composto por um “buffer” simples e apenas um enlace de saída de capacidade C (*bps*). Considerando-se multi-tipos de chegadas de tráfego e disciplina *FIFO* (*Firt In First Out*), a expressão do tempo médio de espera $E\{W\}$ expressa em [11] é dada por:

$$E\{W\} = \frac{E\{T_0\}}{1 - \rho_{TOTAL}} \quad (7)$$

Onde $E\{T_0\}$, na Equação 7, é o tempo de espera residual de serviço, assumindo-se uma estratégia sem preempção. Implica que o pacote que chega, de voz ou de dados, deve esperar o servidor tratar o pacote que está no sistema. Neste caso o servidor não interrompe (sem preempção) o atendimento de um dado pacote até sua efetiva transmissão na rede *IP*. O tempo médio de espera no *buffer*, conforme Eq.3, é diretamente proporcional ao tempo de espera residual de serviço.

$E\{T_0\}$ em [11] é representado por:

$$E\{T_0\} = \frac{1}{2} \cdot \lambda_v \cdot E\{R_v^2\} + \frac{1}{2} \cdot \lambda_D \cdot E\{R_D^2\} \quad (8)$$

Os termos $E\{R_v^2\}$ e $E\{R_D^2\}$ representam o segundo momento das distribuições do comprimento dos pacotes de voz e dados, respectivamente.

Assumindo que a distribuição do tamanho dos pacotes de voz tem tamanho fixo e a distribuição do tamanho dos pacotes de dados obedece a uma distribuição exponencial negativa resulta que o quadrado do primeiro momento (média) da distribuição de voz e o quadrado do primeiro momento (média) da distribuição de dados podem ser escritos, respectivamente, como:

$$E\{R_v\}^2 = \frac{1}{(C \cdot y \cdot \mu_D)^2} \quad (9)$$

e

$$E\{R_D\}^2 = \frac{1}{(C \cdot \mu_D)^2} \quad (10)$$

A variância da distribuição dos pacotes de voz e dos pacotes de dados relaciona os momentos de primeira e segunda ordem de cada distribuição, e é expressa, respectivamente, por:

$$\sigma_v^2 = E\{R_v^2\} - E\{R_v\}^2 \quad \text{e} \quad \sigma_D^2 = E\{R_D^2\} - E\{R_D\}^2 \quad (11)$$

Na distribuição dos pacotes de voz (fixa) e dos pacotes de dados (exponencial negativa) a variância resultante em cada caso é dada por:

$$\sigma_v^2 = 0 \quad \text{e} \quad \sigma_D^2 = \frac{1}{(C \cdot \mu_D)^2} \quad (12)$$

Inserindo as Equações 9, 10 e 12 na Equação 11, e após algumas manipulações algébricas, é possível expressar o segundo momento das distribuições do comprimento dos pacotes de voz e dados, respectivamente, como:

$$E\{R_V^2\} = \frac{1}{(C \cdot y \cdot \mu_D)^2} \quad \text{e} \quad E\{R_D^2\} = \frac{2}{(C \cdot \mu_D)^2} \quad (13)$$

A expressão do tempo de espera de serviço $E\{T_0\}$ pode ser escrita de forma parametrizada na Equação 14 se os resultados obtidos nas Equações 1 e 13 são inseridos na Equação 8. A equação de $E\{T_0\}$ resultante é:

$$E\{T_0\} = \frac{1}{2} \cdot \left(\frac{x \cdot \lambda_D}{(C \cdot y \cdot \mu_D)^2} + \frac{2 \cdot \lambda_D}{(C \cdot \mu_D)^2} \right) \quad (14)$$

Na Equação 14 o termo $\left(\frac{\lambda_D}{C \cdot \mu_D} \right) \cdot \frac{1}{C \cdot \mu_D}$ é deixado em evidência. Este termo também pode ser re-escrito, a partir da Equação 2, como $\frac{\rho_D}{C \cdot \mu_D}$. O tempo de espera de serviço $E\{T_0\}$, conforme Equação 14 e após arranjo algébrico adequado, é representado de forma reduzida como:

$$E\{T_0\} = \frac{\rho_D}{C \cdot \mu_D} \cdot \left(\frac{1}{2} \cdot \frac{x}{y^2} + 1 \right) \quad (15)$$

Inserindo-se as Equações 6 e 15 na Equação 7 e após algumas manipulações matemáticas a seguinte expressão pode ser escrita para a expressão do tempo médio de espera $E\{W\}$, que é o mesmo dado o esquema *FIFO*, para os pacotes de voz e pacotes de dados.

$$E\{W\} = \frac{\frac{\rho_D}{C \cdot \mu_D} \cdot \left(\frac{1}{2} \cdot \frac{x}{y^2} + 1 \right)}{1 - \rho_D \cdot \left(\frac{x}{y} + 1 \right)} \quad (16)$$

A Equação 16 representa o tempo médio de espera dos pacotes (voz e dados) sem a aplicação de priorização na rede *IP*. Esta permite o estudo dos efeitos das variáveis x e y no tempo de espera.

Se os pacotes de voz tiverem prioridade sobre os pacotes de dados, a expressão para filas com prioridade e sem preempção estudada em [11] pode ser usada. Neste caso as expressões para tempo médio de espera para tráfego de voz, $E\{W_V\}$, e o tempo médio de espera para o tráfego de dados, $E\{W_D\}$, são dados por:

$$E\{W_V\} = \frac{E\{T_0\}}{1 - \rho_V} \quad (17)$$

e

$$E\{W_D\} = \frac{E\{T_0\}}{(1 - \rho_v) \cdot (1 - \rho_v - \rho_D)} \quad (18)$$

O tempo de espera residual de serviço, $E\{T_0\}$, é a mesma expressão definida no esquema *FIFO*. Inserindo-se as Equações 2, 3 e 15 nas Equações 17 e 18 podem ser obtidas as expressões do tempo médio de espera do tráfego de voz e do tempo médio de espera do tráfego de dados.

$$E\{W_v\} = \frac{\frac{\rho_D}{C \cdot \mu_D} \cdot \left(\frac{1}{2} \cdot \frac{x}{y^2} + 1 \right)}{\left(1 - \frac{x}{y} \cdot \rho_D \right)} \quad (19)$$

e

$$E\{W_D\} = \frac{\frac{\rho_D}{C \cdot \mu_D} \cdot \left(\frac{1}{2} \cdot \frac{x}{y^2} + 1 \right)}{\left(1 - \frac{x}{y} \cdot \rho_D \right) \cdot \left(1 - \frac{x}{y} \cdot \rho_D - \rho_D \right)} \quad (20)$$

A Equação 20, após manuseio algébrico, pode ser expressa como segue:

$$E\{W_D\} = \frac{\frac{\rho_D}{C \cdot \mu_D} \cdot \left(\frac{1}{2} \cdot \frac{x}{y^2} + 1 \right)}{1 - \rho_D \cdot \left(2 \cdot \frac{x}{y} + 1 \right) + \rho_D^2 \cdot \left(\frac{x^2}{y^2} + \frac{x}{y} \right)} \quad (21)$$

A partir das expressões de tempo médio de espera na fila obtidas nas Equações 16, 19 e 21 é possível estudar o desempenho da rede quando os pacotes de voz são multiplexados no esquema *FIFO* e são encaminhados sem prioridade e quando para transmissão dos pacotes de voz é estabelecido (aplicado) prioridade na rede. Estas equações permitem analisar e comparar o impacto da carga de tráfego de voz na rede *IP*. A aplicação de voz necessita um tratamento especial para ter bom desempenho na rede de comutação de pacotes (ex. Internet). A aplicação de priorização na rede pode resolver este requisito do tráfego de voz. No entanto o tráfego de menor prioridade terá degradação de desempenho. Para minimizar o impacto devido à carga de tráfego de voz na rede *IP* é necessário reduzir o uso ineficiente de banda (recursos da rede) pelos pacotes de voz. Uma

abordagem para resolver este problema é o uso da técnica de multiplexagem de quadros de voz que torna possível aumentar o tamanho da carga útil (sinal de voz) de um pacote *IP*.

3.3 Análise de Resultados

Atualmente na rede *IP* tal como a Internet é implementada a política do melhor esforço para tratar os pacotes, o que significa que não há garantia de entrega do pacote ou controle de latência. No modelamento matemático proposto este cenário pode ser estudado em parte pelo uso do esquema *FIFO*.

Quando uma plataforma de rede *IP* suporta *QoS* (Qualidade de Serviço), assim como o caso *DiffServ*, é possível prover priorização de tráfego. Para implementar esta funcionalidade é necessário criar diferentes tipos de filas (classes de serviço) e estabelecer prioridade para cada tipo de tráfego. No nó *IP* o servidor trata primeiro o tráfego da fila de mais alta prioridade e em seguida trata, de forma gradativa, o tráfego de cada fila com prioridade mais baixa. Neste estudo são consideradas duas diferentes classes de serviço: tráfego de voz, que tem prioridade mais alta, e tráfego de dados que tem prioridade mais baixa.

Os modelos matemáticos apresentados nas Equações 19 e 21 são usados para analisar o desempenho destes casos.

Agora alguns exemplos numéricos são dados para examinar o desempenho da rede *IP*. Os seguintes valores numéricos são assumidos quando 30 fontes de voz estão ativas com e sem multiplexagem dos quadros de voz. Cada fonte de voz equivale, por exemplo, a um canal telefônico convencional. Este número de fontes de voz (canais de voz) é equivalente à uma unidade do padrão E1, que é o padrão Europeu para transmissão digital. Um tributário E1 transporta sinais em $2Mbps$, ou mais precisamente em $2048kbps$. Cada E1 é composto de 32 canais de voz de $64kbps$. No tributário E1 dois canais são reservados para sinalização e controle.

Os pacotes de voz seguem o padrão – G.729 – com 01 quadro de voz de 10 bytes e ciclo de transmissão igual à 10ms. Na rede *IP* cada pacote de voz deve incluir 40 bytes de cabeçalho (resultante) que corresponde à concatenação dos cabeçalhos dos protocolos *RTP*, *UDP* e *IP*.

Sem *MUX*, o pacote *IP* tem 50 bytes, e é a sobreposição do conjunto de cabeçalhos dos protocolos *RTP*, *UDP* e *IP* (40bytes) e um quadro de voz de 10 bytes. É um pacote *IP* que transmite o sinal de uma única fonte de voz.

Com *MUX*, o pacote *IP* tem 340 bytes. Neste caso cada pacote *IP* transporta um quadro de voz composto de 30 diferentes fontes de voz com 10 bytes cada. A carga útil, ou quadro de voz resultante, é de 30×10 bytes (ou 300 bytes). Ao pacote *IP* é inserido também o cabeçalho resultante de 40 bytes.

Cada fonte de voz é tratada por *CODEC* específico. Um conjunto de *CODEC*'s digitaliza e transmite os 30 canais de voz sem *MUX*. O ciclo de transmissão equivalente das 30 fontes de voz, sem *MUX*, torna-se $\frac{10}{30}ms$. Esta carga de tráfego é mantida constante e representa neste caso

tráfego de voz intenso. O tamanho médio dos pacotes de dados é 1000 bytes e a fonte de dados gera tráfego que varia de zero *bps* à 2048 *kbps*.

A partir das considerações anteriores, a utilização de banda (recursos da rede) pelo tráfego de voz sem *MUX* é 1,2*Mbps* e com *MUX* é 272*kbps*. O primeiro caso (sem *MUX*) representa 58,59% de ocupação da capacidade do enlace (2048*kbps*) enquanto no segundo caso (com *MUX*) representa 13,28%.

Quatro diferentes cenários são considerados. No primeiro cenário os tráfegos de voz e dados são tratados no esquema *FIFO* (Melhor Esforço) e sem aplicar a multiplexagem nos quadros de voz. No segundo cenário o mesmo esquema de encaminhamento (*FIFO*) do primeiro cenário é usado, mas agora há a aplicação de multiplexagem. O terceiro cenário aplica priorização e trata o os pacotes de voz com a mais alta prioridade, mas sem multiplexar de quadros de voz. Já o quarto cenário repete o esquema de priorização do terceiro cenário e considera a multiplexagem dos quadros de voz. Para analisar estes quatro cenários os parâmetros apresentados na tabela 3.1 são usados.

Tabela 3.1 – Parâmetros usados na análise numérica

Esquema de Multiplexagem	$\frac{1}{\lambda_V}$	$\frac{1}{\mu_V}$	$\frac{1}{\lambda_D}$	$\frac{1}{\mu_D}$
Sem <i>MUX</i>	$\frac{10}{30} ms$	50 bytes	Variável	1000 bytes
Com <i>MUX</i>	10ms	340 bytes	Variável	1000 bytes

A Figura 3.2 mostra o tempo de espera para o esquema *FIFO* (primeiro e segundo cenários) e reflete a expressão obtida na Equação 16. O aumento gradual da carga de tráfego de dados ocorre, uma vez que o tamanho médio do pacote de dados (1000bytes) é mantido constante, desde que o ciclo de transmissão dos pacotes de dados diminua gradualmente. A relação entre x e λ_D , e os valores de y usando ou não o esquema de multiplexagem estão mostrados na tabela 3.2.

Como pode ser observado na tabela 3.2, os valores que x e y assumem na situação “Com *MUX*” são menores do que na situação “Sem *MUX*”. Esta é uma consequência da adoção de carga constante para o tráfego de voz e diferentes tamanhos de pacotes de voz quando a técnica de multiplexagem é ou não usada.

Além disto, quando $\rho_V > \rho_D$ pode ser encontrado, aplicando estas condições nas Equações 1 e 2, resulta que $x > y$, e quando $\rho_V < \rho_D$ decorre que $x < y$. Estes resultados se aplicam tanto para o caso do “Sem *MUX*” como para o caso do “Com *MUX*”.

Tabela 3.2 – Relações entre x e y considerando a utilização do enlace: sem *MUX* e com *MUX*

Esquema de Multiplexagem	x	y	ρ_D	ρ_V	ρ_{TOTAL}
Sem <i>MUX</i>	234,38	20	0,0500	0,5859	0,6359
	102,71	20	0,1141	0,5859	0,7000
	28,30	20	0,4141	0,5859	1,0000
Com <i>MUX</i>	7,81	3	0,0500	0,1328	0,1828
	3,42	3	0,1141	0,1328	0,2469
	0,94	3	0,4141	0,1328	0,5469

Na figura 3.1 pode ser notado que o tempo médio de espera dos pacotes para o caso multiplexado diminuiu. A utilização do enlace nesta figura é relacionada apenas à carga de tráfego de dados uma vez que a carga de tráfego de voz é mantida constante. A carga de tráfego (constante) de voz é 0,5859 E (*Erlang*) sem multiplexagem e com o esquema de multiplexagem a carga de tráfego é 0,1328 E.

A melhoria de desempenho fica maior para carga de tráfego mais alta, normalmente a utilização do enlace atinge o limiar de uso em 70% da capacidade máxima do enlace (carga máxima recomendada para uma operação estável da rede). Por outro lado, para carga de tráfego mais baixa (por exemplo, menor que 50% da capacidade do enlace) a aplicação da técnica de multiplexagem é desnecessária, pois os tempos médios de espera para ambos os casos são quase os mesmos. Considerando-se o limiar de uso do enlace em 70%, a carga de tráfego de dados pode atingir, nesta situação, até 0,1151 E e 0,562 E sem e com aplicação do esquema de multiplexagem, respectivamente.

Se este limiar (70%) é excedido a carga de tráfego pode atingir até 100% da utilização do enlace que significa uma operação altamente saturada. Neste caso o limiar de uso saturado do enlace é 100% da capacidade, a carga de tráfego de dados pode atingir, nesta situação, até 0,4141 E e 0,8672 E sem e com aplicação do esquema de multiplexagem, respectivamente. A tabela 3.3 mostra a relação entre os valores de x, a ocupação do enlace (carga de tráfego) e os esquemas de multiplexagem em uso para as situações pré-limites (70% de utilização do *link*) e os casos de operações saturadas (100% de utilização do *link*).

Pode ser concluído da tabela 3.3 que se a mesma quantidade de informação de voz e dados é transferida através de um *link* de capacidade de 2Mbps, por exemplo, um *link* E1, e menos banda é necessária no caso de uso do esquema de multiplexagem de voz.

Como conclusão geral pode ser observado que usando *MUX* o sistema economiza banda para o tráfego de voz e para o tráfego de dados, isto é fica mais eficiente.

Tabela 3.3 – Relações entre x , o esquema de multiplexagem e a utilização do enlace

Esquema de Multiplexagem	Carga de Tráfego	X
Sem <i>MUX</i>	70,0% da capacidade do enlace	102,71
Com <i>MUX</i>	24,7% da capacidade do enlace	0,45
Sem <i>MUX</i>	100% da capacidade do enlace	28,3
Com <i>MUX</i>	54,7% da capacidade do enlace	4,5

No caso do esquema com *MUX* a carga de tráfego para atingir a região de limiar em que a capacidade do enlace começa a saturar é maior do que sem *MUX*. Isto implica que mais informação, neste caso uma quantidade maior de pacotes de dados, pode ser transmitida pelo mesmo enlace se aplicado o esquema de multiplexagem nos quadros de voz.

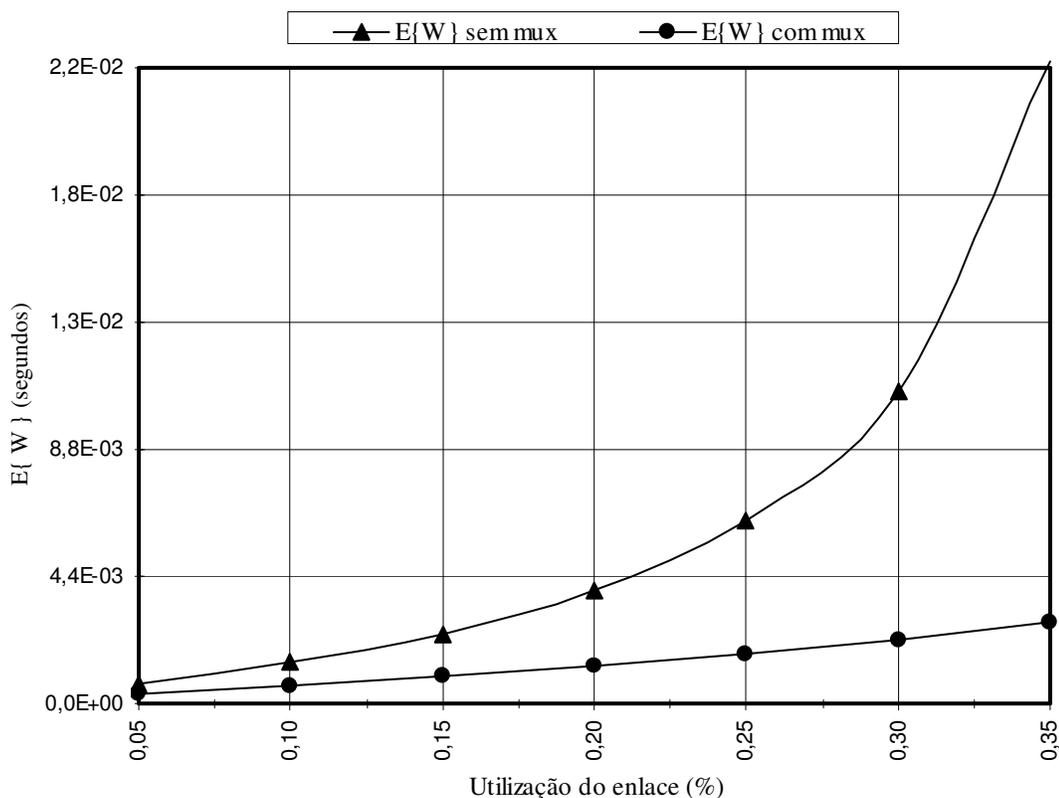


Figura 3.1 – Tempo de espera: os pacotes de voz e dados no caso da disciplina *FIFO*

Na tabela 3.4 o tempo de espera para as duas situações é mostrado. Na primeira situação sem multiplexagem a carga de tráfego para a região de limiar é 0,7 E e na segunda situação com multiplexagem a carga é 0,2469 E. A mesma quantidade de informação é transmitida em ambas situações. O tempo médio de espera, para cada caso, é calculado usando-se a Equação 16. Como pode ser observado, a partir da tabela 3.4, para transportar a mesma quantidade de informação o uso da técnica de multiplexagem pode trazer melhoria (redução) no tempo de espera.

Tabela 3.4 – Tempo de espera: 30 canais de voz (ativos) e pacotes de dados na disciplina *FIFO*

Esquema de Multiplexagem	$E\{W\}$
Sem <i>MUX</i>	1,676ms
Com <i>MUX</i>	0,708ms

Na Figura 3.2 está mostrado o terceiro cenário com prioridade para os pacotes de voz e sem multiplexagem.

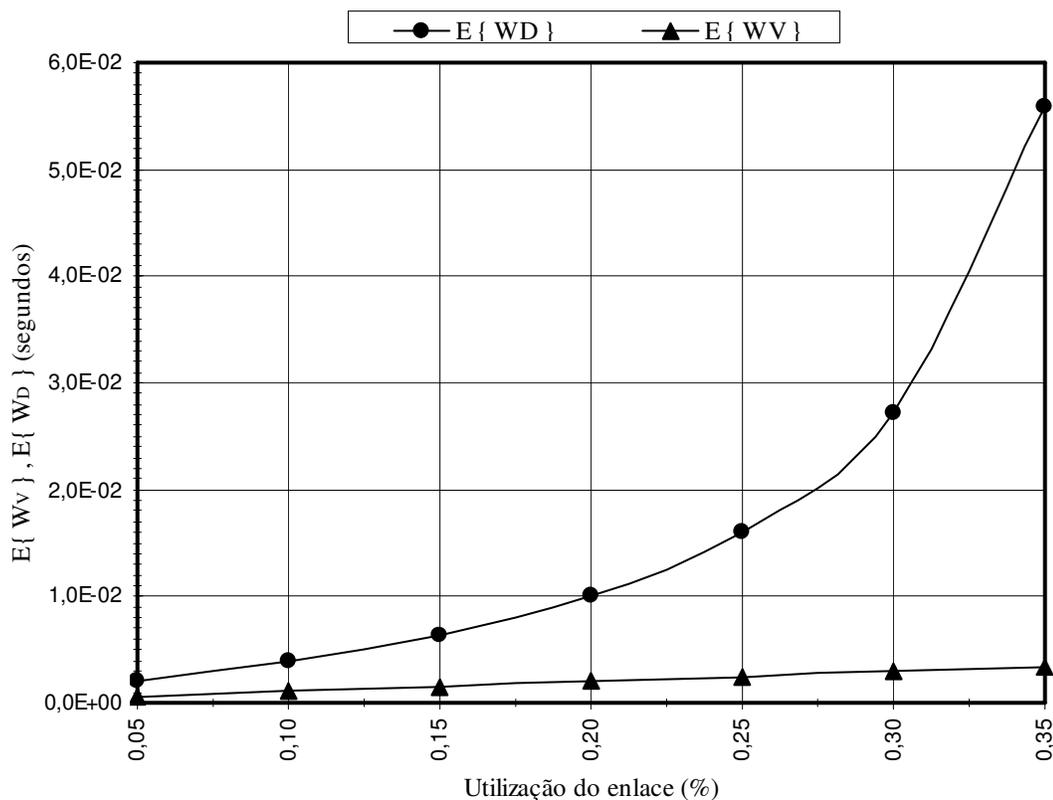


Figura 3.2 – Tempo de espera: prioriza pacotes de voz (sem multiplexagem)

Na Figura 3.3 é possível notar o impacto da prioridade dos pacotes de voz em ambos os tempos médios de espera de voz e dados. O desempenho de voz é significativamente melhor do que o desempenho de dados. Comparando as Figuras 3.1 e 3.2, fica claro que, os pacotes de dados são penalizados para um melhor desempenho dos pacotes de voz. Para compensar esta característica é necessário usar eficientemente a estrutura de transporte *IP*, que pode ser obtida pela técnica de multiplexação de quadros de voz.

A Figura 3.3 mostra o caso para a disciplina de prioridade e com multiplexagem (quarto cenário). Neste caso ambos os tráfegos de voz e dados têm considerável melhora de desempenho, isto é o tempo médio de espera é menor para transportar a mesma quantidade de informação. Comparando os resultados obtidos na Figura 3.3 com os resultados obtidos na Figura 3.2 os resultados são muito interessantes. Não somente o tráfego de voz tem melhoria de desempenho na rede *IP*, mas os pacotes de dados também. Estes resultados são interessantes, pois provam que o esquema de multiplexagem de voz é muito importante para melhorar o desempenho de uma rede *IP*.

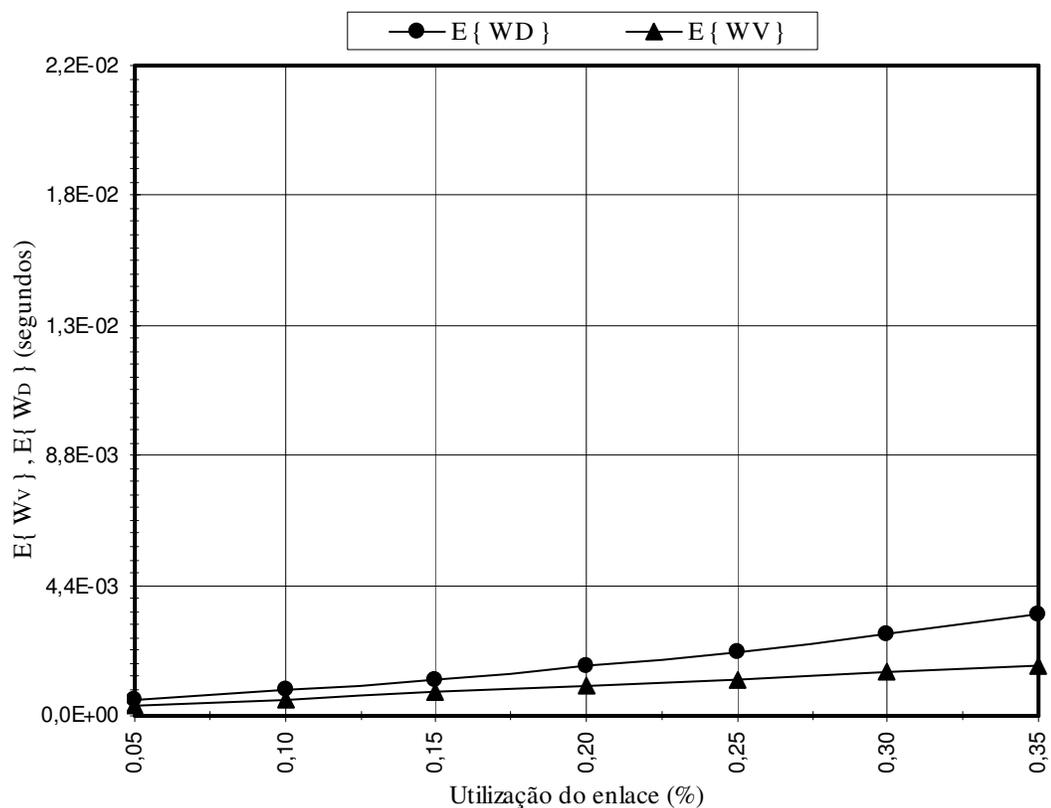


Figura 3.3 – Tempo de espera: prioriza pacotes (com multiplexagem)

3.4 Conclusão:

Neste capítulo, a partir de modelagem específica, o desempenho dos pacotes de voz na rede *IP* é avaliado considerando-se a aplicação do esquema de multiplexagem dos quadros de voz e efetuado a comparação para o caso de não aplicação de multiplexagem. Além da questão da aplicação ou não de multiplexagem, a avaliação também considerou a aplicação de prioridade nos pacotes de voz. Os pacotes de voz ficaram com nível de prioridade mais alto e os pacotes de dados com nível de prioridade mais baixo. Para avaliação todos os casos dos modelos matemáticos foram desenvolvidos.

Os resultados mostram que o uso da técnica de multiplexagem dos quadros de voz é muito importante para melhorar, através do uso eficiente de recursos (banda), o desempenho de uma rede *IP*. Como para a rede *IP* é esperado, num futuro próximo, tornar-se a plataforma de transporte para todo o tipo de tráfego telefônico, a técnica de multiplexagem deve ser usada para garantir utilização eficiente de banda (recursos) disponível.

Capítulo 4

Simulação de Tráfego em redes *IP*

4.1 Introdução

No capítulo anterior, foi feito o estudo do impacto do tráfego de voz na rede *IP* considerando um modelo de tráfego baseado em fontes poissonianas. Os modelos matemáticos desenvolvidos a partir dessas fontes permitiram um estudo aproximado do comportamento do tráfego de voz com multiplexação e, também, sem multiplexação. Para um estudo que considere modelos mais próximos dos tráfegos reais, a modelagem analítica torna-se muito complexa e de difícil equacionamento. Desta forma, nesta dissertação foi feito um estudo mais realístico do impacto de tráfego de voz na rede *IP* através da simulação em computador. Nesta simulação, foi desenvolvida uma plataforma de rede *IP* no ambiente de *Matlab e Simulink*. Os modelos de tráfegos reais considerados para fontes de voz e para fontes de dados são do tipo *ON-OFF* e *HTTP* respectivamente. Cada fonte de voz *ON-OFF* e cada fonte de dados *HTTP* possuem características específicas definidas em modelagem estatística que são reproduzidas a partir de simulação. Para a simulação do tráfego real é utilizada a ferramenta de software do *Matlab* e a implementação dos escalonadores e extratores de estatística ocorre através do *SimEvent*.

É possível realizar, através da simulação das fontes de tráfego real, cenários que reproduzem a carga de tráfego analisadas no capítulo 3 (resultados analíticos), aplicar esquemas de escalonamento *FIFO*, esquemas de escalonamento por prioridade e comparar os tempos médios de espera em cada cenário (analítico e simulado). Para tratar o impacto do uso da técnica de multiplexagem, sugerida neste estudo, varia-se durante a simulação de fonte de tráfego real o tamanho médio do pacote *IP* (carga útil do pacote *ON-OFF*) e o ciclo de transmissão entre pacotes (pacotes *HTTP*).

O cenário de análise é comparar, a partir dos resultados numéricos extraídos da simulação, o atraso médio inserido nos pacotes de voz e o atraso médio inserido nos pacote de dados considerando-se o ambiente de melhor esforço (sem priorização) e quando há a aplicação de *QoS* (com priorização) na rede *IP* com os resultados analíticos (vide equações) obtidos no capítulo 3. Em seguida é feita a análise e a validação dos resultados, através da compilação dos dados obtidos na simulação, sobre o impacto do tráfego real de voz na rede *IP* e ao final são apresentadas as conclusões.

4.2 Fonte de Voz – Modelo *ON-OFF*

Uma ligação telefônica através do serviço de *VoIP* pode ser considerada, assim como uma ligação telefônica convencional, um canal de comunicação composto por dois interlocutores, um chamador e o outro receptor. E quando um fala o outro escuta. Além disso, quando o interlocutor fala, freqüentemente faz pequenas pausas entre as frases ditas. A Fig. 4.1 relaciona o nível de conversação e a transmissão dos quadros de voz.

A fonte de voz, a partir deste comportamento, pode ser modelada como uma fonte do tipo *ON-OFF*. O padrão de tráfego de voz *ON-OFF*, conforme estudados em [13], é caracterizado por um período ativo (fala ou *talk spurt*) seguido por um período inativo (silêncio). Durante o período ativo, a fonte envia pacotes em intervalos regulares de comprimento T (tempo de empacotamento). E durante o período inativo, a fonte não envia pacotes.

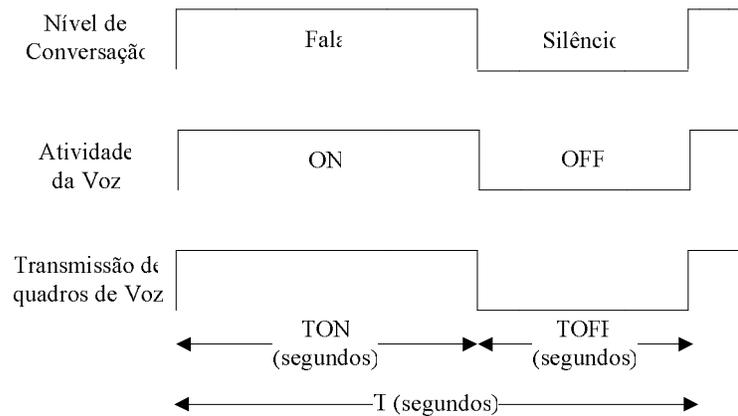


Figura 4.1 – Nível de Conversação e transmissão de quadros de voz (*CODEC*)

A duração dos períodos ativos (T_{ON}) e dos períodos inativos (T_{OFF}) segue uma distribuição exponencial negativa e é representada através dos parâmetros α (taxa de voz ativa) e β (taxa de voz inativa) sendo que estas independem, entre si. A fonte de tráfego de voz mostrada na Fig. 4.2 é definida como modelo *ON-OFF* Markoviano, em diagrama de estado, e é composta por três parâmetros de distribuição:

1. A duração média do período ativo (*ON*) é $T_{ON} = \frac{1}{\alpha}$. O parâmetro α segue uma distribuição exponencial negativa no período ativo;
2. A duração média do período inativo (*OFF*) é $T_{OFF} = \frac{1}{\beta}$. O parâmetro β segue uma distribuição exponencial negativa no período inativo;

3. A taxa de chegada dos pacotes de voz é constante e durante o período ativo é expressa por

$$\lambda = \frac{1}{T}$$

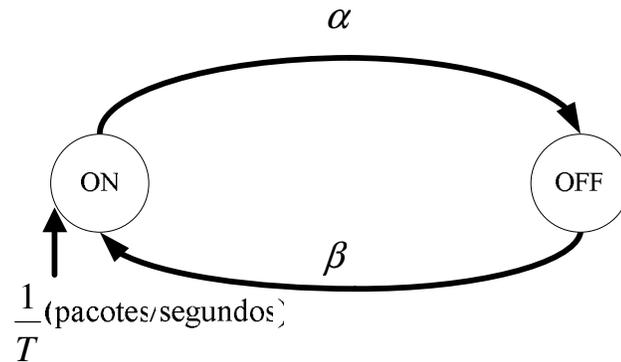


Figura 4.2 – Modelo de Fonte de Voz ON-OFF (com 02 estados)

A taxa média de chegada dos pacotes voz (T_{ON}), medida em pacotes por segundo, depende da taxa de voz ativa (α), da taxa de voz inativa (β) e da taxa de chegada dos pacotes de voz que resulta em:

$$\lambda_{MÉDIO} = \frac{T_{ON}}{(T_{ON} + T_{OFF}) \cdot T} \quad \text{ou} \quad \lambda_{MÉDIO} = \frac{\beta}{(\beta + \alpha)} \cdot \lambda \quad (22)$$

O tempo médio de duração de uma ligação telefônica é de 3 minutos e segue uma distribuição exponencial negativa. Em cada pacote de voz gerado, além do cabeçalho *RTP*, é inserido o cabeçalho do protocolo *UDP*. O tamanho médio de cada pacote *IP* e o ciclo de transmissão entre pacotes dependem do *CODEC* usado para efetuar o encapsulamento e a transmissão dos quadros de voz. A tabela 4.1 relaciona os principais *CODEC*'s em uso e mostra os valores típicos (parâmetros) usados na geração dos modelos de fontes de tráfego de *VoIP*. A duração do período ativo (T_{ON}) e do período inativo (T_{OFF}) é obtida a partir de medidas extraídas em [14].

Com base nas características apresentadas da fonte de voz é descrito a seguir o algoritmo, descrito na Fig. 4.3, que permite criar no *MATLAB* a fonte de voz real (*ON-OFF*) considerando-se o *CODEC G729*. A tabela 4.1 contém os parâmetros para ligações telefônicas que duram em média três minutos. No caso da simulação os valores usados como parâmetros para gerar tráfego de voz foram diferentes dos sugeridos na tabela 4.1, pois o modelo analítico utilizou o tamanho médio de quadros de voz com dez *bytes* e ciclo de transmissão de dez mili segundos para simular fontes de voz *ON-OFF* sem aplicar a técnica de multiplexagem sugerida. Para gerar tráfego de voz considerando-se o uso da técnica de multiplexagem o tamanho médio do quadro de voz foi trezentos *bytes* e ciclo de transmissão de um trinta avos de dez mili segundos.

Tabela 4.1 – Parâmetros dos modelos de VoIP por CODEC

CODEC	Tamanho dos pacotes (Bytes)	Ciclo dm Transmissão (ms)	T_{ON} (segundos)	T_{OFF} (segundos)
G711	136	12	0,352	0,65
G726	104	16	0,352	0,65
G729	70	30	0,352	0,65

A implementação do código de fonte de voz no *MATLAB* e o uso dos parâmetros sugeridos no modelo analítico permitem reproduzir a estimativa do tráfego real de voz.

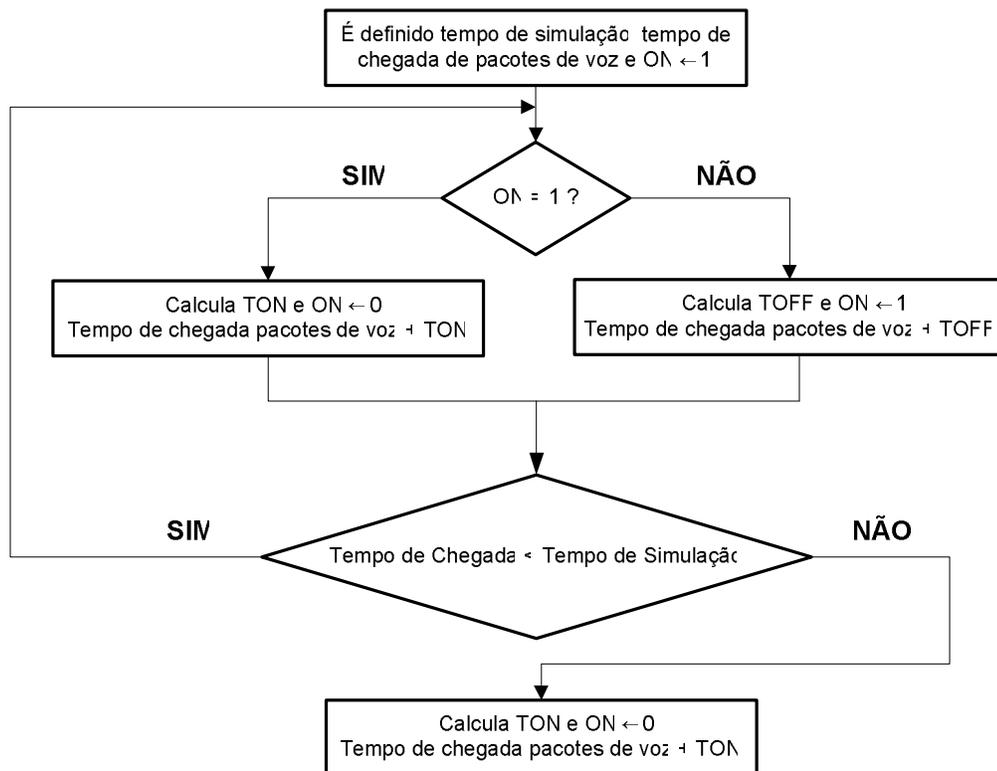


Figura 4.3 – Modelagem de Tráfego de Voz ON-OFF

4.2.1 Histogramas das Fontes de Voz – Modelo ON-OFF

Os histogramas da simulação da fonte de voz (modelo *ON-OFF*) apresentados nas Fig. 4.4 e Fig. 4.5 exibem a distribuição estatística da modelagem do tráfego de voz *ON-OFF* no ambiente de simulação do *Matlab* para pacotes de 50 bytes e 340 bytes respectivamente.

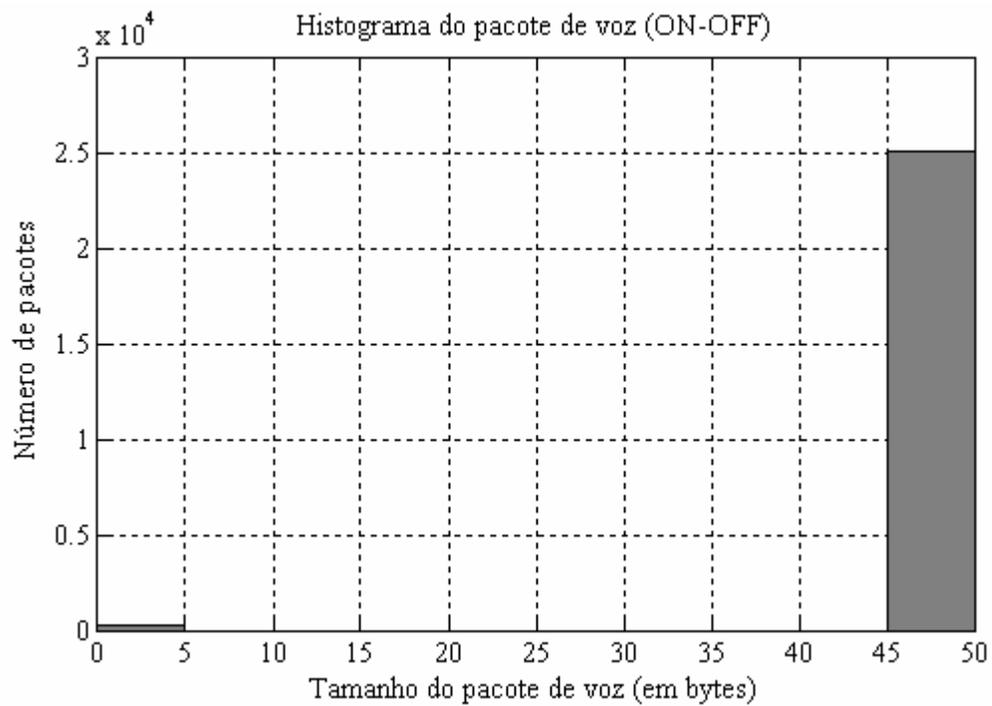


Figura 4.4 – Histograma do Tamanho do Pacote de Voz *ON-OFF* (50 bytes)

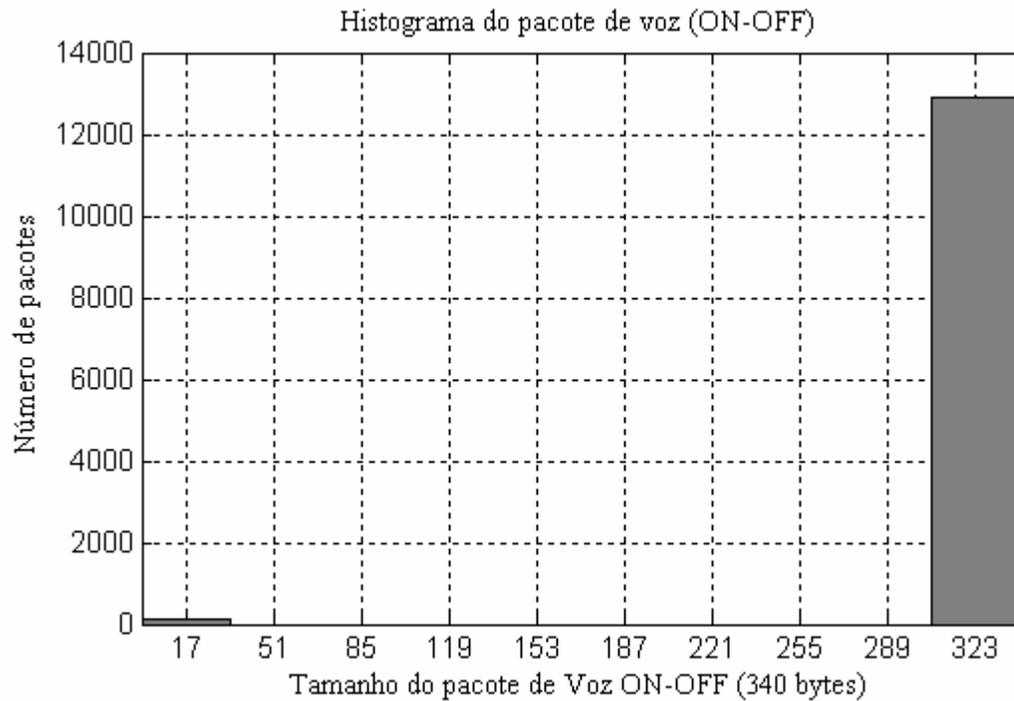


Figura 4.5 – Histograma do Tamanho do Pacote de Voz *ON-OFF* (340 bytes)

Na simulação são gerados quadros de voz com 10 bytes (sem esquema de multiplexagem) ou de 300 bytes (com esquema de multiplexagem) que são encapsulados pelo conjunto de protocolos RTP (12 B), UDP (8 B) e IP (20 B). Resulta que a distribuição dos pacotes de voz, associados ao conjunto de cabeçalhos IP, têm tamanho de 50 bytes ou 340 bytes respectivamente. Em ambos os casos existem pacotes residuais com tamanho de até 5 bytes e 34 bytes respectivamente, que correspondem à 10% do tamanho dos pacotes de voz e são resultantes do processo de simulação.

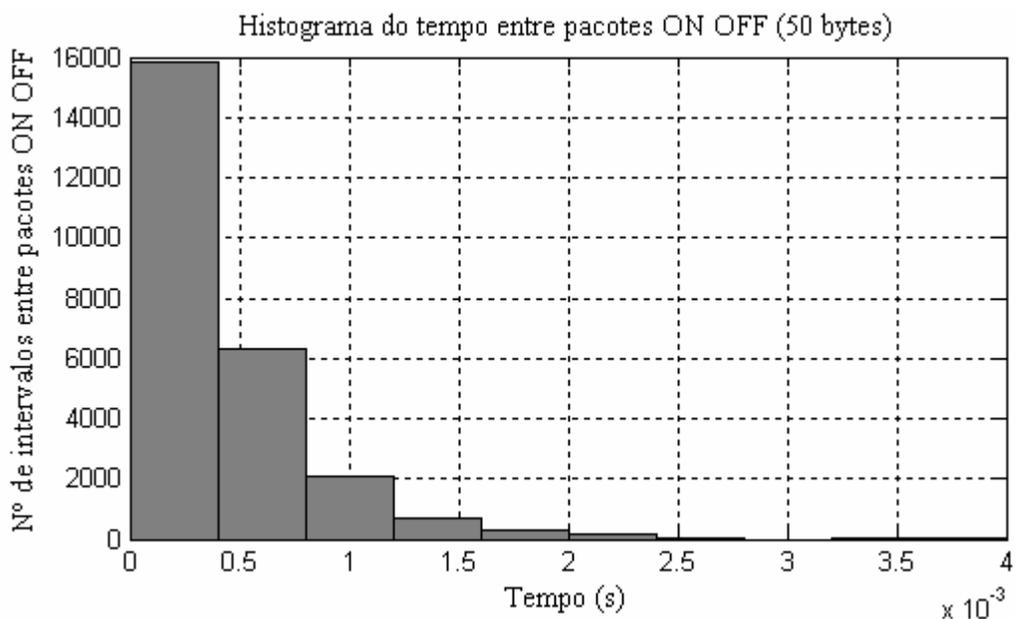


Figura 4.6 – Histograma do intervalo entre pacotes *ON-OFF* (50 bytes)

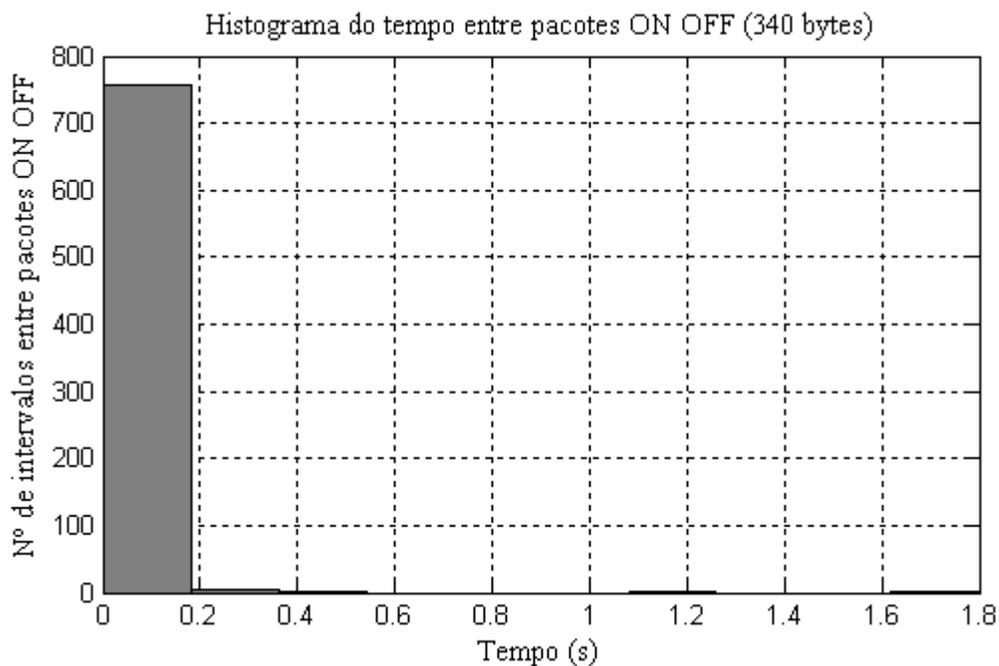


Figura 4.7 – Histograma do intervalo entre pacotes *ON-OFF* (340 bytes)

O tráfego voz é composto de períodos ativos (fala) e inativos (silêncio). A Fig. 4.8 apresenta a ordem de chegadas dos pacotes do tráfego de voz *ON-OFF*. Nesta figura, conforme indica a distribuição temporal dos pacotes de voz *ON-OFF*, ocorre a alternância de intervalos de tempo que contêm rajadas de pacotes (fala) com intervalos de tempo que contêm silêncio (ausência de fala).

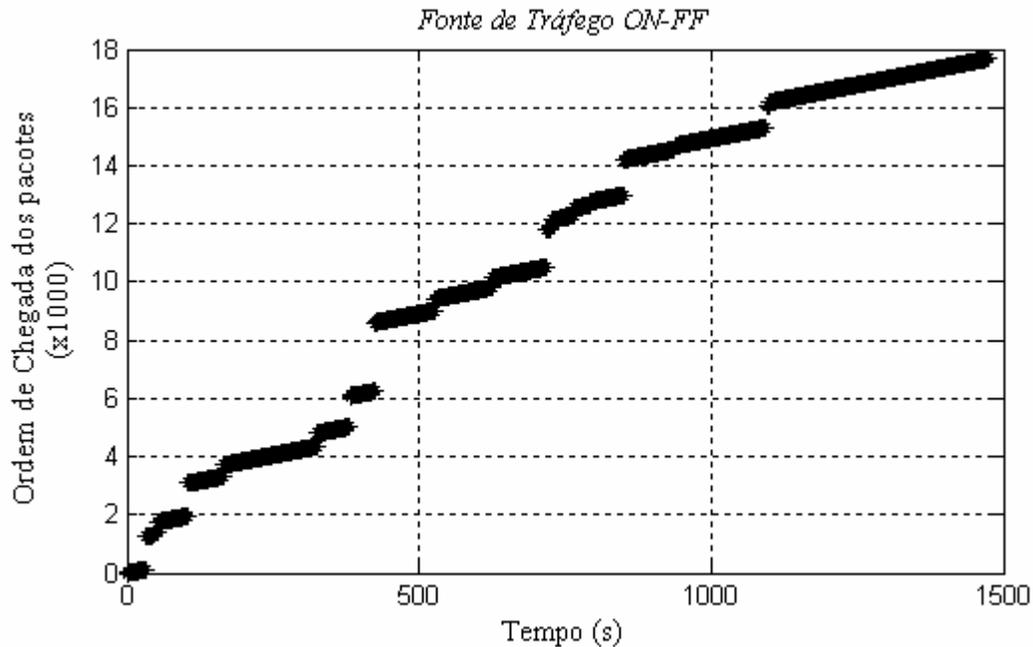


Figura 4.8 – Ordem de chegada dos pacotes *ON-OFF* (50 bytes)

4.3 Fonte de Dados – Modelo *HTTP*

4.3.1 Propriedades da Fonte de Dados – Modelo *HTTP*

O tráfego *HTTP* é definido e modelado por seção de tráfego (navegação) *web*. Na Fig. 4.9 a seção *web* é dividida em períodos ativos (*ON*) e períodos (*OFF*) que representam respectivamente períodos de *downloads* de páginas *web* e períodos de tempo de leitura desta página. O período de tempo de leitura é o tempo que o usuário interpreta uma página *web*. Uma seção *web*, além de conter uma quantidade aleatória de páginas *web*, pode ser composta de um conjunto de períodos de *downloads* e um conjunto de períodos de leitura de páginas *web*. Esta decomposição em períodos ativos (*ON*) e períodos inativos (*OFF*) são resultantes da interação homem máquina. O período de *download* representa a solicitação de informação pelo usuário e neste instante ocorre transferência de pacotes a partir do servidor de tráfego *HTTP*. O período de tempo de leitura é referente ao tempo gasto pelo usuário para interpretar a página *web* e neste período não há requisição de tráfego. Além

disso, o tráfego *web* é autosimilar, isto é, em diferentes escalas de tempo apresenta a mesma distribuição estatística de tráfego.

Uma seção de navegação na *web* ocorre a partir do acesso a uma página *web*. Os componentes de uma página *web* são um objeto principal e um conjunto de objetos embutidos.

O navegador, antes de carregar uma página *web*, busca o objeto principal da página *HTML* (*Hyper Text Markup Language*) através da requisição *HTTP GET*. Em seguida, após o recebimento do objeto principal, o navegador define o layout desta página para então inserir os objetos embutidos tais como botões e figuras. O tempo usado para definir o layout da página *web* é denominado como tempo de *parsing*. Para completar a página *web* o navegador efetua o *download* (carregamento) dos objetos embutidos e em seguida o usuário inicia a leitura desta página. O período de busca e o período de *download* do objeto principal e do conjunto de objetos embutidos compreendem o período ativo (*ON*) enquanto o período de leitura das páginas *HTML* compreende o período inativo (*OFF*). A Tabela 4.10 apresenta os componentes, objeto principal e conjunto de objetos embutidos, de uma página *web* durante a transferência de pacotes numa seção *HTTP*.

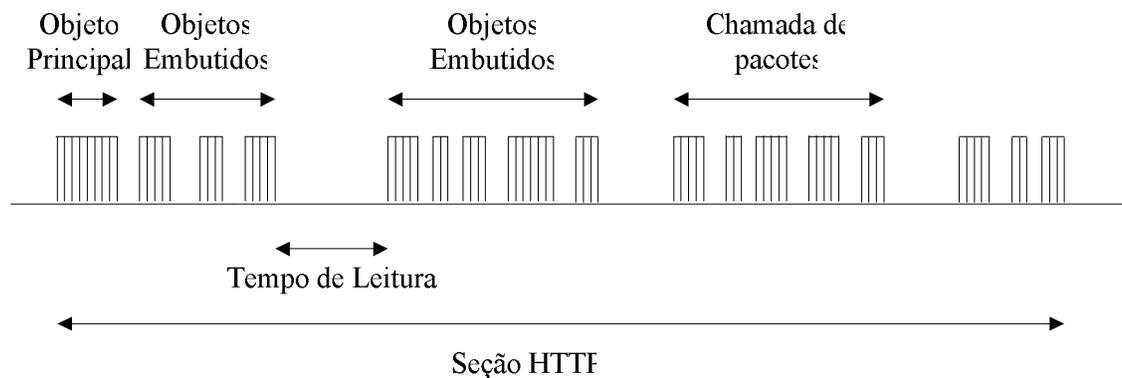


Figura 4.9 – Transferência de Pacotes durante uma seção HTTP

A descrição da distribuição estatística do tráfego *HTTP* também depende do esquema de *download* usado para transferir os segmentos de dados e neste cada pacote *HTTP* é encapsulado pelo protocolo *TCP*. O uso deste protocolo estabelece o tamanho da *MTU* (fragmentação de pacotes) e o controle de fluxo por variação do tamanho de janela (*windowing*) para a transferência de pacotes *HTTP*. Assim a distribuição do tráfego *web* depende da capacidade do servidor de tráfego *HTTP*, da condição de uso dos navegadores de páginas *HTML* e da versão do protocolo *HTTP* empregada. As duas versões do protocolo *HTTP* em uso atualmente, na maioria dos servidores e navegadores, são *HTTP/1.0* e *HTTP/1.1*. O tráfego *web* resultante em cada versão *HTTP* difere na distribuição do tamanho e fluxo de pacotes. Além da transferência de cada objeto principal e do conjunto de objetos embutidos ocorre através de uma ou mais conexões *TCP*.

Na versão do protocolo *HTTP/1.0* é usada uma conexão *TCP* para efetuar o *download* de cada objeto principal ou uma ou mais conexões *TCP* para efetuar o *download* de cada objeto do conjunto de objetos embutidos. O protocolo *HTTP/1.0* efetua a transferência de pacotes específicos, originados do objeto principal ou do conjunto de objetos embutidos, através da técnica de *burst mode transfer* (modo de transferência em rajadas). A distribuição de tráfego em rajadas é resultante do número máximo, até 04 conexões *TCP* simultâneas, ativas numa mesma seção de navegação *web*.

Em contrapartida, na versão do protocolo *HTTP/1.1*, a transferência de pacotes ocorre através da técnica *persistent mode transfer* (modo de transferência persistente). Neste caso são estabelecidas conexões *TCP* persistentes (únicas) e assim tanto os objetos principais quanto o conjunto de objetos embutidos são transferidos serialmente através do estabelecimento de uma única conexão *TCP*.

4.3.2 Parametrização da Fonte de Dados – Modelo *HTTP*

O estudo e análise das características do tráfego *HTTP* ocorre a partir do uso de parâmetros referentes à distribuição estatística da dinâmica de tráfego *web* gerado. O tráfego *HTTP* cria diferentes tipos de fluxos de pacotes, isto é, diferentes tamanhos de pacotes e intervalos entre pacotes na mesma seção de navegação *web*. O tráfego *web* é composto por um conjunto de diferentes distribuições estatísticas e parâmetros específicos. O uso das distribuições estatísticas e a aplicação dos parâmetros do modelo de tráfego *HTTP* permitem o desenvolvimento de algoritmos específicos e também a simulação do tráfego *web*. A reprodução da distribuição estatística do tráfego *HTTP* torna-se possível no software *MATLAB/SimEvent*. Os parâmetros e padrões de distribuição estatística usados estão apresentados na tabela 4.2.

Uma página *web* é composta por um único objeto principal e por um conjunto de objetos embutidos. O tamanho do objeto principal obedece a uma distribuição lognormal com tamanho mínimo de 100 *bytes*, tamanho máximo de 2 *megabytes* e média de 10710 *bytes* com os parâmetros variância (σ) igual à 1,37 e a taxa de partida (μ) igual à 8,35. No caso dos objetos embutidos, que também obedecem a uma distribuição lognormal, o tamanho médio de cada objeto é 7758 *bytes*, o tamanho mínimo é 50 *bytes* e o tamanho máximo de 2 *megabytes* com os parâmetros variância igual à 2,36 e a taxa de partida igual à 6,17. O tamanho (em *bytes*) de cada objeto embutido, do ponto de vista estatístico, é menor que o objeto principal.

O tempo de leitura (TL) ou o tempo de lacuna entre sessão *web* é o intervalo de tempo entre duas requisições consecutivas de página *web* e neste caso este tempo segue uma distribuição exponencial com média de 30 segundos.

Em seguida é descrito o tempo de *parsing* que é o tempo necessário para completar o layout da página *web*. O tempo de *parsing* é medido após a busca do objeto principal e segue uma distribuição exponencial com média 0,13 segundos. Os pacotes *HTTP* (objeto principal e conjunto de objetos embutidos) são gerados conforme os parâmetros da tabela 4.2. O tamanho de cada objeto

principal, o número de objetos embutidos e o tamanho de cada objeto embutido seguem o padrão de distribuição apresentado na tabela 4.2.

Tabela 4.2 – Parâmetros para Modelagem de Tráfego *HTTP*

Componente	Distribuição	Parâmetros			PDF
		Média	Mínimo	Máximo	
Tamanho do Objeto Principal	Lognormal Truncada	10710 bytes	100 bytes	2 Mbytes	$f_x = \frac{1}{\sqrt{2\pi\sigma x}} \exp\left[-\frac{(\ln x - \mu)^2}{2\sigma^2}\right], x \geq 0$ $\sigma = 1.37, \mu = 8.37$
Quantidade de Objetos Embutidos	Pareto Truncada	5.64	-	53	$f_x = \left(\frac{a_k^\alpha}{x^{\alpha+1}}\right), k \leq x < m$ $f_x = \left(\frac{k}{m}\right)^\alpha, x = m$ $\alpha = 1.1, k = 2, m = 55$
Tamanho dos Objetivos Embutidos	Lognormal Truncada	7758 bytes	50 bytes	2 Mbytes	$f_x = \frac{1}{\sqrt{2\pi\sigma x}} \exp\left[-\frac{(\ln x - \mu)^2}{2\sigma^2}\right], x \geq 0$ $\sigma = 2.36, \mu = 6.17$
Tempo de Parsing	Exponencial	0.13 seg	-	-	$f_x = \lambda \cdot e^{-\lambda \cdot x}, x \geq 0$ $\lambda = 0.033$
Tempo de Leitura	Exponencial	30 seg	-	-	$f_x = \lambda \cdot e^{-\lambda \cdot x}, x \geq 0$ $\lambda = 0.033$

Cada objeto gerado é fragmentado em pacotes, encapsulado através do protocolo *TCP* e em seguida é inserido o cabeçalho *IP*. A distribuição do tamanho dos pacotes *IP* que transmitem o tráfego *web*, a partir de observações da rede *IP*, resulta em pacotes de 576 bytes e 1500 bytes. A distribuição de carga de tráfego *web*, baseados no tamanho dos pacotes *IP* (*MTU*), é composta por 24% dos pacotes com 576 bytes e 76% dos pacotes com 1500 bytes outra possibilidade é considerar que 50% dos pacotes possuem 576 bytes e a outra metade possui 1500 bytes. A Fig. 4.10 apresenta o algoritmo para geração e modelagem do tráfego *web*.

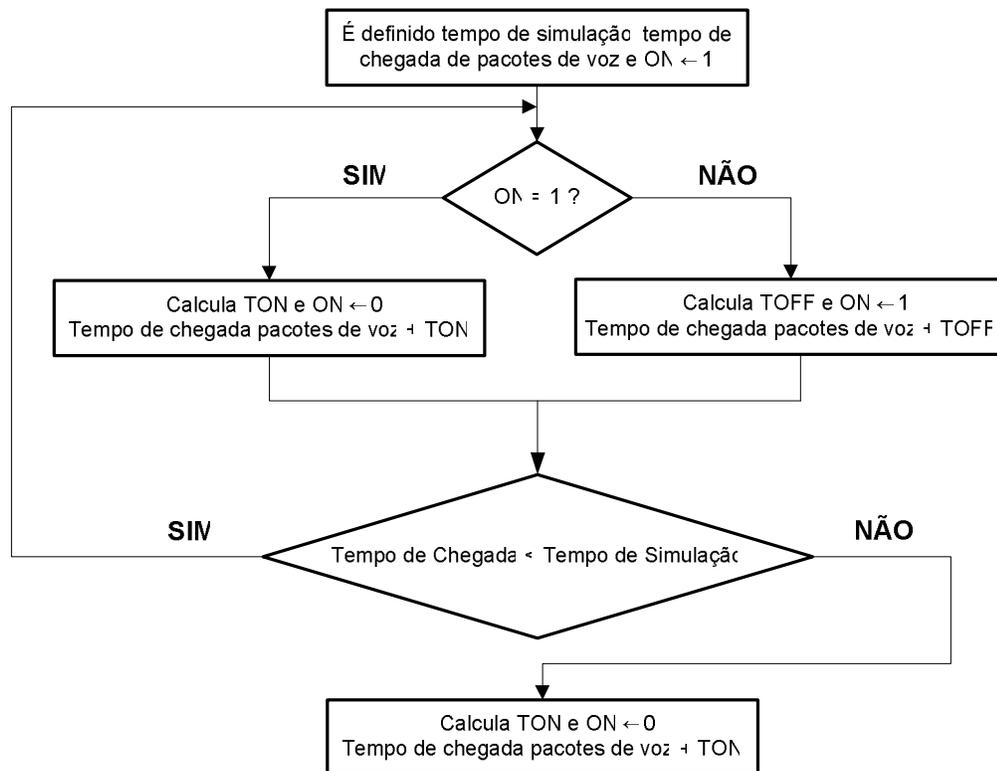


Figura 4.10 – Algoritmo do Tráfego de Voz *ON-OFF*

A tabela 4.2 também apresenta as funções densidades de probabilidades (*PDF*) usadas para a obtenção da modelagem de tráfego *HTTP* além dos parâmetros já compilados. Importante esclarecer que para a geração de números aleatórios, que obedecem as distribuições lognormal, *pareto*, exponencial e geométrica, foi usada neste estudo o método da transformada inversa. Com este procedimento as distribuições mencionadas podem ser invertidas de forma analítica exceto a distribuição lognormal. Esta abordagem matemática foi utilizada no modelo de tráfego *web* estudado. No ambiente de simulação do *Matlab/Simulink* a geração dos números aleatórios que obedecem as distribuições compiladas anteriormente, conforme parâmetros da tabela 4.2, já estavam disponíveis no recurso de ferramentas *toolbox statistic (Matlab/Simulink)* inclusive a distribuição de *pareto*.

4.3.3 Histogramas das Fontes de Dados – Modelo *HTTP*

O histograma da simulação da fonte de dados (modelo *HTTP*) apresentado na Fig. 4.11 provê a distribuição estatística do tráfego *web* no ambiente de simulação do *Matlab*. Os pacotes *HTTP* já com 40 *bytes* de cabeçalho têm tamanho de 576 *bytes* e 1500 *bytes*, ou seja, possuem 536 *bytes* e 1460 *bytes* de carga útil respectivamente. Há também pacotes com tamanhos de 40 *bytes*, 80 *bytes* e 160 *bytes* que são resultantes do esquema de controle do protocolo *TCP* (pacotes de controle e pacotes de sincronismo).

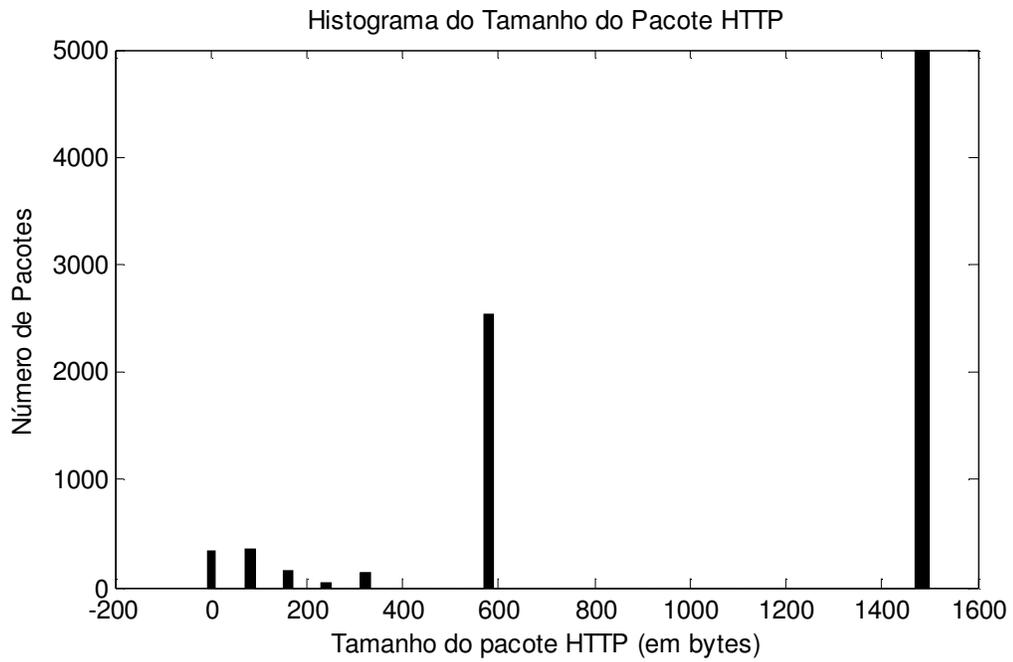


Figura 4.11 – Histograma do Tamanho do Pacote *HTTP*

O tráfego *web* é composto de períodos ativos (*download*) e inativos (tempo de leitura). A Fig. 4.12 apresenta a ordem de chegadas dos pacotes do tráfego *HTTP*. Nesta figura, conforme indica a distribuição temporal dos pacotes *HTTP*, ocorre alternância de intervalos de tempo que contêm rajadas de pacotes (*download*) com intervalos de tempo que contêm silêncio (tempo de leitura).

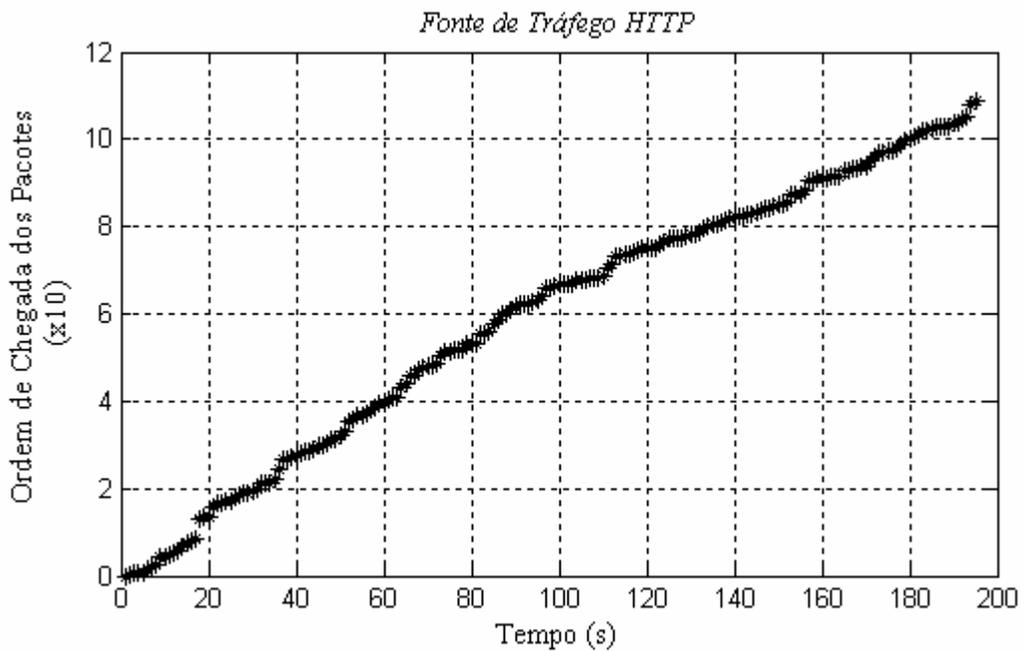


Figura 4.12 – Ordem de chegada dos pacotes *HTTP*

Os resultados do tráfego *web* obtidos da simulação, e apresentados nas figuras 4.11 e 4.12, quando interpretados em conjunto permite a visualização da dinâmica de tráfego gerado. O esforço é reproduzir o conjunto de distribuições estatísticas sugeridas na tabela 4.2, pois a resolução algébrica torna-se muito complexa.

O histograma da distribuição do intervalo (em tempo) entre chegadas de pacotes do tráfego *HTTP* é mostrado na Fig. 4.13.

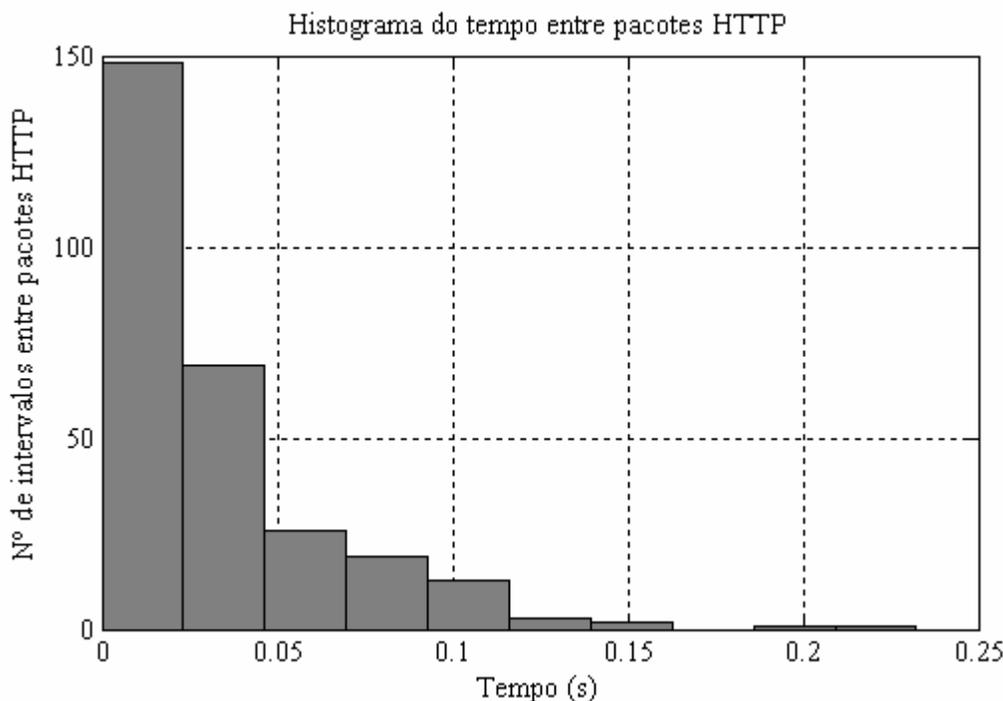


Figura 4.13 – Histograma do intervalo entre pacotes *HTTP*

Conforme pode ser observado, há uma grande concentração de pacotes em torno de 50ms que são resultantes do tráfego do período ativo (*download*) enquanto os poucos pacotes provenientes dos períodos inativos estão distribuídos e são desprezíveis.

4.3.4 Escalonadores com esquema *FIFO* (*SimEvent*)

O escalonador *FIFO* (*First In Firsts Out*) cria uma única fila (de tamanho definido) e mantém no *buffer*, por um determinado intervalo de tempo, cada pacote que chega. A disciplina desta fila define que o primeiro pacote que chega é o primeiro pacote transmitido. A média do tempo que cada pacote, tanto de voz (*ON-OFF*) quanto de dados (*HTTP*), fica na fila é o tempo médio espera.

A ferramenta de simulação *SimEvent* (*Matlab*), conforme apêndice B, possui modelo específico (bloco) que cria uma fila com o esquema *FIFO* e trata os eventos discretos (pacotes) gerados em cada uma das fontes de tráfego que utiliza o modelo *ON-OFF* e o modelo *HTTP*. A fig. B1 apresenta os blocos usados para implementar este esquema de fila em que cada evento (pacote)

contém o tempo de chegada de cada pacote e o tamanho de cada pacote (*bytes*). Neste caso, é necessário converter durante a simulação de cada fonte (voz e dados) o tamanho do pacote (*bytes*) para tempo (segundos). O tamanho do pacote (em segundos) é obtido a partir do produto do tamanho de cada pacote (em *bytes*) por 8 ($1\text{ byte} = 8\text{ bits}$) e em seguida divide-se este resultado pela capacidade do enlace (*bps*). A capacidade do *buffer* determina a quantidade máxima de pacotes que pode permanecer na fila simultaneamente.

No caso desta simulação é assumido que a capacidade do *buffer* é infinita. Além disso, para inserir a capacidade do enlace de 2 *Mbps* foi usado o modelo de servidor (*single server*), conectado diretamente à saída do escalonador *FIFO*. O tempo gasto pelo servidor para tratar cada pacote (evento) egresso da fila *FIFO* é o tempo de serviço. Na simulação o tempo de serviço (duração do serviço) foi extraído diretamente de cada evento o que equivale ao tamanho de cada pacote tratado (em segundos).

No apêndice A estão representados os histogramas normalizados do tempo de espera na fila (*SimEvent*) tratadas pelo escalonador *FIFO* com servidor simples e sem preempção usado na simulação. As situações representadas nestas figuras refletem o índice de utilização do enlace de 70% (condição pré-limite) e 100% (saturação), respectivamente.

4.3.5 Escalonadores com esquema de Priorização (*SimEvent*)

O escalonador que aplica a técnica de priorização de fila (*Priority Queue*) cria, neste experimento e está apresentado no apêndice B na fig. B2, duas filas: uma para armazenar os pacotes de voz e outra fila para armazenar os pacotes de dados. Cada pacote, no processo de geração de eventos (*Matlab*), é identificado (marcado) como ou de dados ou de voz. Em seguida, a partir do tempo de chegada, os pacotes (eventos) são sequenciados em ordem cronológica crescente. O tempo entre pacotes é obtido através da diferença do tempo de chegada entre pacotes adjacentes. Ou seja, é o tempo de chegada do pacote atual menos o tempo de chegada do pacote antecessor. Desta forma cada evento gerado possui tempo de chegada, tamanho do pacote e intervalo entre eventos. Com esta abordagem é possível inserir a estatística do intervalo entre eventos no *SimEvent*. Na entrada do escalonador com esquema de prioridade, cada pacote que chega, é alocado ou na fila de voz (*ON-OFF*) ou na fila de dados (*HTTP*). O atendimento de cada fila (voz ou dados) é pelo esquema armazena e envia. A disciplina de priorização estabelece que o escalonador trata primeiro todo pacote de voz que está na fila de voz e em seguida trata os pacotes de dados armazenados na fila de dados. Neste caso, o tempo médio de espera que os pacotes de voz permanecem na fila de voz é representado por $E\{W_{VOZ}\}$ e o tempo médio de espera na fila de dados por $E\{W_{DADOS}\}$ que são valores distintos entre si.

A ferramenta *SimEvent* (*Matlab*) cria eventos discretos de simulação e possui modelo específico (bloco) que define o esquema de priorização com duas filas (dados e voz). A fig. B2 apresenta os blocos usados para implementar este esquema de fila com prioridade que trata os eventos discretos (pacotes) gerados por cada uma das fontes de tráfego que seguem, respectivamente, o modelo de distribuição estatística do tráfego de voz *ON-OFF* (pacotes de voz) e

do tráfego *HTTP* (pacotes de dados). Cada evento contém o tempo de chegada de um pacote, o tamanho do pacote (*bytes*) e a marcação se pacote de voz ou se pacote de dados. O mesmo procedimento já descrito de conversão do tamanho dos pacotes de *bytes* para segundos deve ser efetuado. A capacidade de cada *buffer*, fila de voz ou fila de dados, determina a quantidade máxima de pacotes que podem permanecer em cada fila simultaneamente.

No caso desta simulação é assumido que a capacidade do *buffer* de voz e do *buffer* de dados é infinita. Além disso, para inserir a capacidade do enlace de 2 *Mbps* foi usado o modelo de servidor (*single server*), conectado diretamente à saída do escalonador com priorização. O tempo gasto pelo servidor para tratar cada pacote (evento) egresso ou da fila de voz ou da fila de dados é o tempo de serviço. Na simulação o tempo de serviço (duração do serviço) foi extraído diretamente de cada evento o que equivale ao tamanho de cada pacote tratado (em segundos).

No apêndice A estão representados os histogramas normalizados do tempo de espera em cada fila (*SimEvent*) tratadas pelo escalonador de prioridade com servidor simples e sem preempção usado na simulação. As situações representadas nestas figuras refletem o índice de utilização do enlace de 70% (condição pré-limite) e 100% (saturação), respectivamente.

4.4 Análise de Resultados (*Matlab/SimEvent*)

Para obter os resultados foram utilizados dois ambientes de simulação. O modelo de filas, os escalonadores com esquema *FIFO* e com Prioridade, o servidor simples e extratores de estatística estão no ambiente de simulação do *SimEvent*. Os resultados preliminares das fontes de dados e de voz, obtidos a partir do *Matlab*, são inseridos no ambiente de simulação do *SimEvent* que usa informações dos pacotes (eventos). É usado o tempo entre pacotes gerados e o tamanho de cada pacote (em segundos). Em seguida a extração da distribuição estatística da taxa de chegada e da taxa de partida, para conferência, ocorre a partir do *SimEvent*. É possível extrair a distribuição estatística somente dos pacotes de voz (*ON-OFF*), somente dos pacotes de dados (*HTTP*) e também da agregação destes pacotes que representa o tráfego híbrido (voz e dados).

Cada evento (pacote), conforme o cenário de análise, permanece na fila e é tratado com o escalonador *FIFO* ou permanece na fila e é tratado com o escalonador de prioridade. Após tempo de espera na fila o pacote é transferido conforme recursos do servidor (*single server*). O servidor representa a capacidade do enlace uma vez que processa e encaminha cada pacote que chega conforme política do escalonador. O tamanho de cada pacote (em segundos) é usado pelo servidor, neste caso é o tempo de serviço, que só trata outro pacote após completar o processamento e envio do pacote atual. O servidor efetua o encaminhamento de pacotes sem aplicar técnica de preempção. A taxa de partida obedece à distribuição estatística do tamanho dos pacotes (em segundos) que são tratados pelo servidor.

A política do melhor esforço para tratar cada pacote é obtida, no *SimEvent*, pelo bloco que aplica a técnica de escalonamento *FIFO*. Assim o tráfego de voz e dados é processado pelo conjunto de blocos do *SimEvent* que simulam uma fila *FIFO* e um *Single Server*.

Por outro lado quando uma plataforma de rede *IP* suporta *QoS* (Qualidade de Serviço) é necessário aplicar priorização no tratamento das filas. No ambiente do *SimEvent* a fila com prioridade é atendida conforme esquema usado pelo escalonador de prioridade. Foi configurado que o pacote de voz tem prioridade alta e o pacote de dados prioridade baixa.

A distribuição estatística do tamanho (em tempo) dos pacotes de voz *ON-OFF* e dados *HTTP* em conjunto com o ciclo de transmissão destes pacotes permitem a reprodução do tráfego real. Para examinar o desempenho da rede *IP* foram consideradas 30 fontes de voz ativas.

No *MATLAB* foi gerada rotina que chama recursivamente 30 vezes o código de cada fonte de voz sem a aplicação de multiplexagem dos quadros de voz. Com isto é possível simular as 30 fontes de voz ativas que compõem um tributário E1. Na simulação cada fonte de voz gera pacotes *IP* com 49,62 bytes (9,62B de carga útil e 40 B do conjunto de cabeçalhos *RTP,UDP* e *IP*) e ciclo de transmissão de 10ms conforme padrão *G.729*. É um pacote *IP* que transmite o sinal de uma única fonte de voz.

Para simular o esquema de multiplexagem dos quadros de voz no mesmo pacote *IP* (cenário Com *MUX*) foram alterados o parâmetro tamanho do pacote e o parâmetro de ciclo de transmissão. O mesmo código do programa fonte de voz *ON-OFF* foi usado. Neste caso cada pacote *IP* tem 300 bytes de carga útil que é resultante da multiplexagem dos 30 diferentes quadros de voz (10 x 30B) acrescido de 40 bytes do conjunto de protocolos *RTP, UDP* e *IP*. Assim a parametrização da fonte de voz (no *MATLAB*) tem o tamanho resultante de cada pacote *IP* (fonte de voz) com 340 bytes e o ciclo de transmissão equivalente é $\frac{10}{30} ms$.

A carga de tráfego de voz é mantida constante, isto é, 30 canais de voz ativos sem e com esquema de multiplexagem. O tamanho médio dos pacotes de dados *HTTP* é 1000 bytes e a fonte de dados *HTTP* gera tráfego que varia de 0 bps (zero) à 2048 kbps. Cada fonte *HTTP* gera pacotes de 1000 bytes (em média) e ciclo de transmissão 50ms conforme conjunto de distribuição estatística. Neste caso é necessário alterar o parâmetro ciclo de transmissão dos pacotes *HTTP* conforme carga de tráfego sugerida. Para tanto é gerada rotina que chama recursivamente o programa que simula a fonte de dados *HTTP*. Com isto é preservada a distribuição estatística de tráfego da fonte dos pacotes *HTTP* e simulado o incremento gradativo da carga de tráfego de dados. Cada fonte *HTTP* gera, em média, 0,07 E que é o passo mínimo para incremento de tráfego *HTTP*.

A partir das considerações anteriores, e do uso de parâmetros específicos em cada fonte de tráfego na simulação, a utilização de banda (recursos da rede) pelo tráfego de voz sem *MUX* é 1,2Mbps e com *MUX* é 272kbps da mesma forma que no modelo analítico. O primeiro caso (sem *MUX*) representa 58,59% de ocupação da capacidade do enlace (2048kbps) e no segundo caso (com *MUX*) representa 13,28%.

Novamente quatro diferentes cenários são considerados. Desta vez no ambiente de simulação do *Matlab/SimEvent* que reproduz os cenários sugeridos no modelo analítico. No ambiente do *MatLab* são gerados eventos assíncronos e no ambiente do *SimEvent* estas entidades são tratadas. No primeiro cenário os tráfegos de voz e dados (eventos discretos) são gerados no *MatLab*

conforme algoritmos das fontes *ON-OFF* (voz) e *HTTP* (dados) e em seguida são tratados no esquema *FIFO* (Melhor Esforço) implementado pela ferramenta *SimEvent* através da agregação de modelos de fila e servidor. Neste cenário não há aplicação de multiplexagem nos quadros de voz. No segundo cenário o mesmo esquema de encaminhamento (*FIFO*) é usado no *SimEvent*, e com a aplicação de multiplexagem. A obtenção da multiplexagem ou não é obtida no estágio de simulação do *MatLab*. O terceiro cenário, que prevê priorização, usa o esquema de fila com escalonador de prioridade no ambiente do *SimEvent* e trata os pacotes de voz (eventos) com a mais alta prioridade, e sem multiplexagem de quadros de voz. Este cenário é composto do conjunto fila (com priorização) e servidor simples. No quarto cenário o mesmo esquema de priorização do terceiro cenário é usado e considera, no ambiente de simulação do *MatLab*, a multiplexagem dos quadros de voz. Para analisar estes quatro cenários os parâmetros apresentados na tabela 4.3 são extraídos a partir da simulação dos modelos de fonte *ON-OFF* (voz) e fonte *HTTP* (dados). Estes resultados são próximos dos valores teóricos (analíticos), conforme tabela 3.1, e expressam a característica de tráfego real a partir de modelos pré-definidos.

Tabela 4.3 – Parâmetros obtidos a partir de simulação de fonte voz (*ON-OFF*) e dados (*HTTP*)

Esquema de Multiplexagem	$\frac{1}{\lambda_v}$	$\frac{1}{\mu_v}$	$\frac{1}{\lambda_D}$	$\frac{1}{\mu_D}$
Sem <i>MUX</i>	0,432 ms	49,48 bytes	Variável	1078,42 bytes
Com <i>MUX</i>	8,60 ms	338,76 bytes	Variável	1078,42 bytes

As Fig. 4.14 e Fig. 4.15 mostram respectivamente o tempo de espera para o esquema *FIFO* (primeiro e segundo cenários) obtido no ambiente de simulação do *MATLAB/SimEvent*. O aumento gradual da carga de tráfego de dados, uma vez que o tamanho médio do pacote de dados (1000bytes) é mantido constante, ocorre desde que o ciclo de transmissão dos pacotes de dados diminua gradualmente.

Na simulação cada fonte *HTTP* (dados) tem, em média, intervalo entre pacotes de 49,40ms. O uso de chamadas recursivas do modelo de fonte *HTTP* permite reduzir o ciclo de transmissão dos pacotes de dados. A relação entre x e λ_D , e os valores de y (extraídos da simulação) usando ou não o esquema de multiplexagem estão mostrados na tabela 4.3.

Para extração dos valores da tabela 4.4 são reutilizados os eventos gerados a partir do *MatLab* para os casos de aplicação (Com *MUX*) ou não (Sem *MUX*) da técnica de multiplexagem. Estes eventos são tratados individualmente no ambiente de *SimEvent*. A extração das estatísticas de tráfego, especificamente carga de tráfego, no *SimEvent* possibilita a compilação dos dados apresentados na tabela 4.4. Estes resultados são próximos dos valores teóricos (analíticos) obtidos na tabela 3.2

A taxa de chegada dos pacotes de voz é menor quando a técnica de multiplexagem dos quadros de voz é usada e o tamanho médio dos pacotes de voz, nesta situação, fica maior. Ainda neste caso (aplicação de multiplexagem) as características do tráfego de voz implicam em diminuição do valor de x e de y .

Tabela 4.4 – Relações entre x e y e o uso do enlace: sem MUX e com MUX (simulação)

Esquema de Multiplexagem	x	y	ρ_D	ρ_V	ρ_{TOTAL}
Sem MUX	234,38	20	0,0860	0,5090	0,5950
	102,71	20	0,1657	0,5300	0,6957
	28,30	20	0,4513	0,5161	0,9674
Com MUX	7,81	3	0,0734	0,1206	0,1940
	3,42	3	0,1375	0,1206	0,2561
	0,94	3	0,4285	0,1166	0,6084

Podem ser verificados, a partir da tabela 4.4, que quando $\rho_V > \rho_D$ implica que $x > y$, e quando $\rho_V < \rho_D$ resulta que $x < y$. Estes resultados (extraídos de simulação) comprovam os valores analíticos já descritos na tabela 3.2 que relacionam os percentuais do tráfego de voz e dados (variável x) e comprimento médio dos pacotes de voz e dados (variável y) e a carga de tráfego de voz, de dados e totais.

Na figura 4.15, desta vez no ambiente de simulação, pode ser notado que o tempo médio de espera dos pacotes para o caso multiplexado também diminuiu. A utilização do enlace nesta figura é relacionada apenas à carga de tráfego de dados uma vez que a carga de tráfego de voz é mantida constante. A carga de tráfego de voz (deve ser constante) na simulação variou entre 0,5090 e 0,5300 E (*Erlang*) no esquema sem multiplexagem e com o esquema de multiplexagem a carga de tráfego assumiu valores entre 0,1166 e 0,1206 E (*Erlang*). Embora os eventos sejam os mesmos para a carga de tráfego de voz (constante) no ambiente de simulação do *SimEvent* cada incremento de tráfego de dados reflete em pequenas alterações no tráfego individual (voz e dados) e total.

A melhoria de desempenho também é verificada no ambiente de simulação e fica maior para carga de tráfego maior, e conforme recomendação de operação estável da rede é possível comprovar que a utilização do enlace atinge o limiar de uso em 70% de sua capacidade máxima. Em contrapartida, para carga de tráfego menor o uso da técnica de multiplexagem torna-se desnecessária, pois os tempos médios de espera para ambos os casos são quase os mesmos. Considerando-se o limiar de uso do enlace em 70% (pré-saturação), a carga de tráfego de dados

pode atingir, nesta situação, até 0,1151 E e 0,5672 E sem e com aplicação do esquema de multiplexagem, respectivamente.

Pode ser verificado, via simulação, que quando a carga de tráfego total atinge 100% da utilização do enlace, o que significa uma operação altamente saturada, a carga de tráfego de dados pode atingir o complemento da carga de tráfego de voz que é 0,4141 E sem a aplicação do esquema de multiplexagem e 0,8672 E com a aplicação do esquema de multiplexagem. Desta forma há mais recursos disponíveis para o tráfego de dados.

Tabela 4.5 – Relações entre x , o esquema de multiplexagem e a utilização do enlace (simulação)

Esquema de Multiplexagem	Carga de Tráfego	X
Sem <i>MUX</i>	70,0% da capacidade do enlace	102,71
Com <i>MUX</i>	24,7% da capacidade do enlace	0,45
Sem <i>MUX</i>	100% da capacidade do enlace	28,3
Com <i>MUX</i>	54,7% da capacidade do enlace	4,5

A tabela 4.5 mostra a relação entre os valores de x , a ocupação do enlace (carga de tráfego) e os esquemas de multiplexagem em uso para as situações pré-limites (70% de utilização do enlace) e os casos de operações saturadas (100% de utilização do enlace).

Pode ser concluído e comprovado da tabela 4.5, cujos dados foram compilados a partir de simulação, que para transmitir a mesma quantidade de informação de voz e dados através de um *link* de capacidade de 2Mbps (por exemplo, um *link* E1) menos banda é necessária no caso do esquema de multiplexagem de voz.

Através do ambiente de simulação do *MatLab/SimEvent*, é possível reproduzir a condição de uso estabelecida no modelo analítico o que permite comprovar que um sistema que usa *MUX* economiza recursos (banda) para o tráfego de voz e para o tráfego de dados, isto é, fica mais eficiente.

A partir dos resultados obtidos na modelagem analítica é possível verificar que a capacidade do enlace começa a saturar antes, para uma carga de tráfego de voz constante, se não há uso do esquema de *MUX*. O mesmo cenário é reproduzido no ambiente de simulação e resultados semelhantes são obtidos. O uso da técnica de multiplexagem dos quadros de voz implica, conforme pode ser verificado, que mais eventos de voz e dados (pacotes gerados pelo *MatLab*) são tratados no *SimEvent*.

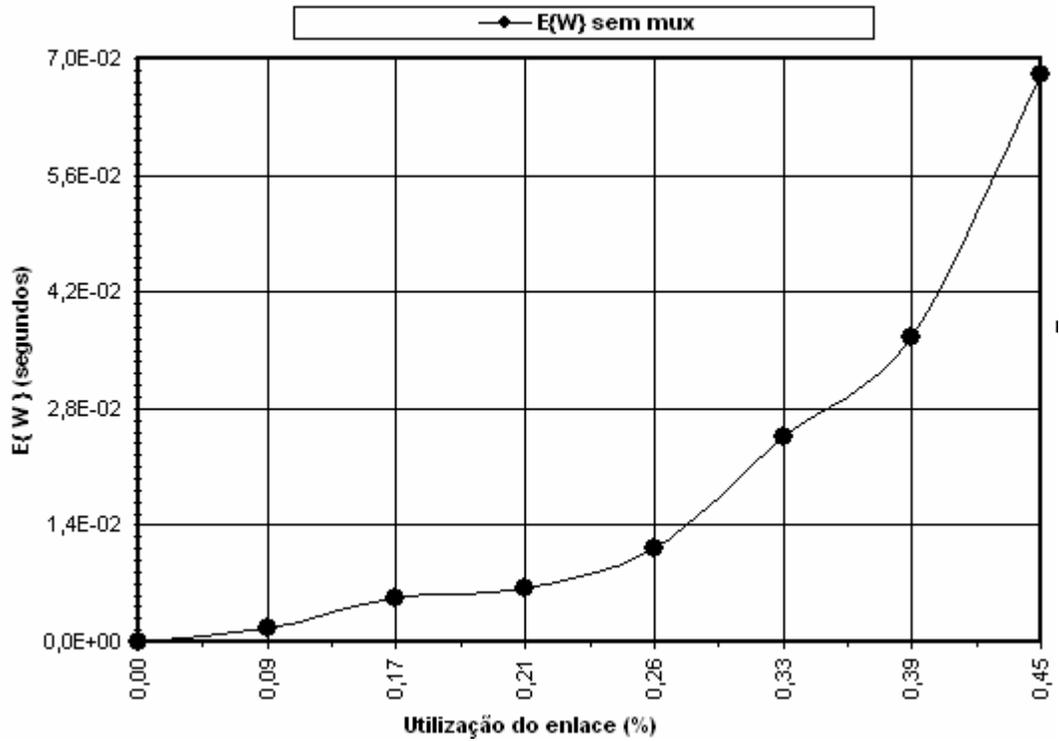


Figura 4.14 – Tempo de espera dos pacotes de voz e dados sem *MUX* e fila *FIFO*

Os resultados de ambos os cenários estão compilados na tabela 4.6 em que o tempo de espera para as duas situações é mostrado.

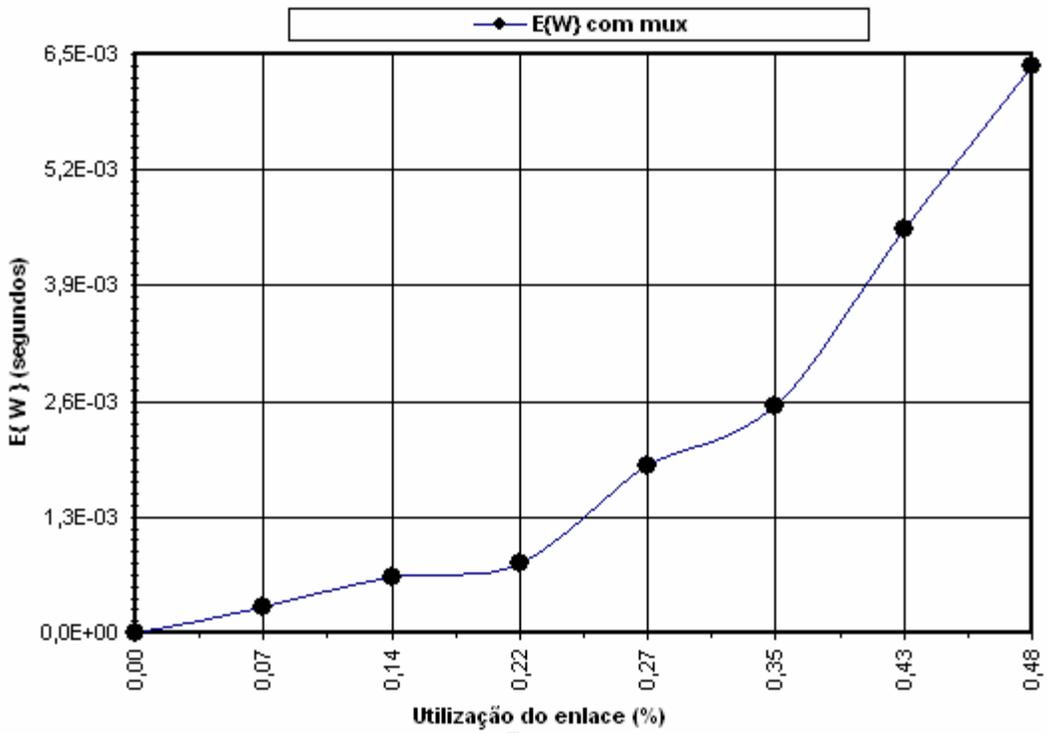


Figura 4.15 – Tempo de espera dos pacotes de voz e dados com *MUX* e fila *FIFO*

Na primeira situação sem multiplexagem a carga de tráfego para a região de limiar é 0,7 E e na segunda situação com multiplexagem a carga é 0,2469 E. E novamente pode ser observado que a mesma quantidade de informação é transmitida em ambas situações.

O tempo médio de espera, para cada caso, é calculado usando-se a Equação 16. Como pode ser observado, a partir da tabela 4.6, para transportar a mesma quantidade de informação o uso da técnica de multiplexagem pode trazer melhoria (redução) no tempo de espera. A Fig. 4.17 mostra o caso para a disciplina de prioridade e com multiplexagem (quarto cenário). Neste caso ambos os tráfegos de voz e dados têm considerável melhora de desempenho, isto é o tempo médio de espera é menor para transportar a mesma quantidade de informação. Comparando os resultados obtidos na Fig. 4.16 com os resultados obtidos na Fig. 4.17 os resultados são muito interessantes.

Tabela 4.6 – Tempo de espera: 30 canais de voz (ativos) e pacotes de dados na disciplina FIFO

Esquema de Multiplexagem	$E\{W\}$
Sem <i>MUX</i>	1,676ms
Com <i>MUX</i>	0,708ms

Na Fig. 4.16 está mostrado o terceiro cenário com prioridade para os pacotes de voz e sem multiplexagem. Nesta figura é possível notar o impacto da prioridade dos pacotes de voz em ambos os tempos médios de espera de voz e dados.

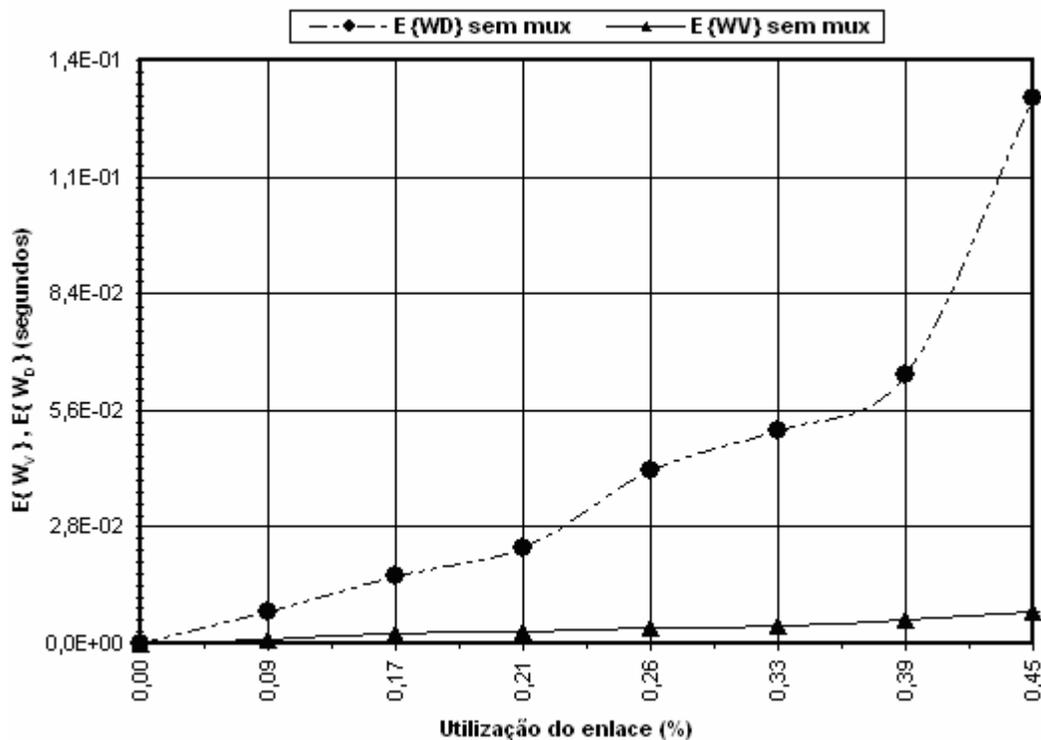


Figura 4.16 – Tempo de espera com priorização (voz) e sem multiplexagem

O desempenho de voz é significativamente melhor do que o desempenho de dados. Comparando as Fig. 4.16 e Fig. 4.17, fica claro que os pacotes de dados são penalizados para um melhor desempenho dos pacotes de voz.

Para compensar esta característica é necessário usar eficientemente a estrutura de transporte *IP*, que pode ser obtida pelo uso da técnica de multiplexagem de quadros de voz.

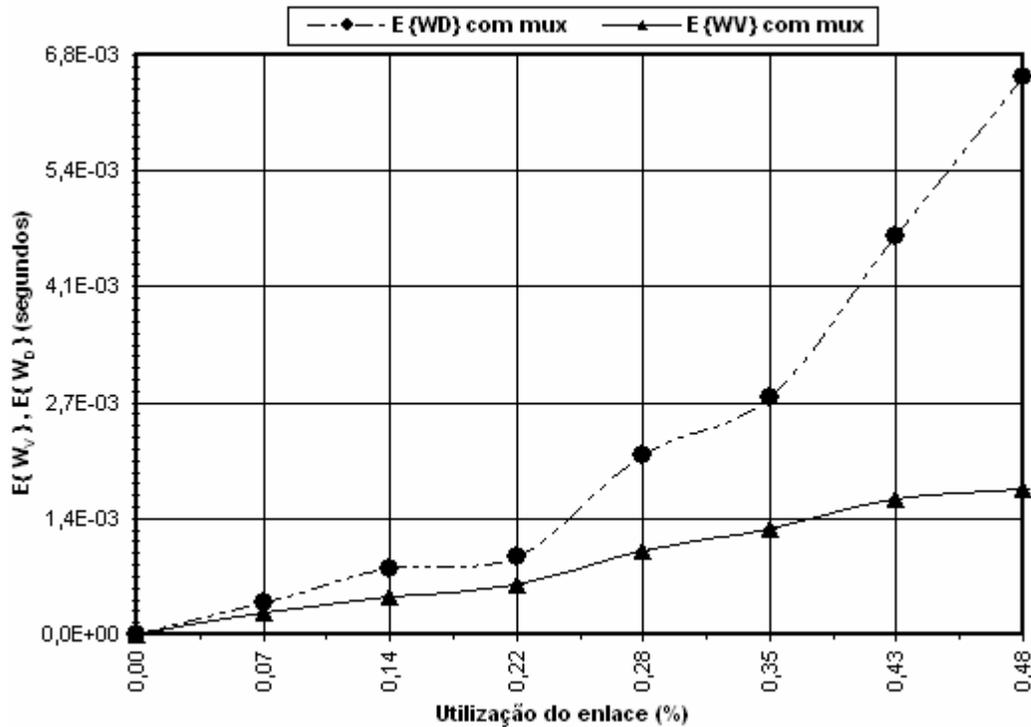


Figura 4.17 – Tempo de espera com(priorização (voz) e com multiplexagem

Não somente o tráfego de voz tem melhoria de desempenho na rede *IP*, mas os pacotes de dados também. Estes resultados são interessantes, pois comprovam a partir de simulação que o esquema de multiplexagem de voz é muito importante para melhorar o desempenho de uma rede *IP*.

4.5 Conclusão:

Neste capítulo os mesmos cenários de análise usados no capítulo 3 são empregados. O ambiente de simulação de eventos do *MatLab/SimEvent* permite analisar, a partir das fontes de tráfego real, o desempenho dos pacotes de voz na rede *IP*. Neste caso a avaliação considera a aplicação do esquema de multiplexagem dos quadros de voz e efetua a comparação para o caso de não aplicação de multiplexagem. Além da questão da aplicação ou não de multiplexagem, a avaliação também considerou a aplicação de prioridade nos pacotes de voz. Os pacotes de voz ficaram com nível de prioridade mais alto e os pacotes de dados com nível de prioridade mais baixo. Para avaliação foram usadas fontes *ON-OFF* (voz) e fontes *HTTP* (dados) implementadas no ambiente de

simulação do *MatLab/SimEvent* e em seguida os dados destas fontes foram utilizados em modelos (blocos) que geram eventos.

Os resultados extraídos da simulação também comprovam que a aplicação da técnica de multiplexagem dos quadros de voz é muito importante para melhorar, através do uso eficiente de recursos, o desempenho de uma rede IP. Como para a rede IP é esperado, num futuro próximo, tornar-se a plataforma de transporte para todo o tipo de tráfego telefônico, a técnica de multiplexagem deve ser usada para garantir utilização eficiente de banda (recursos) disponível.

Capítulo 5

Conclusões

Nesta dissertação, foi feita a análise de atraso dos pacotes de voz em uma rede IP utilizando a técnica de multiplexagem de quadro de voz. Na análise, foram utilizados modelos analíticos e de simulação que levou em consideração as fontes mais complexas de voz e de dados. Tanto no modelo analítico como na simulação foram adotadas as políticas de escalonamento *FIFO* e de prioridade. Importante ressaltar que linhas de pesquisas, através de simuladores específicos, que usam o tráfego *HTTP* (dados) e tráfego *ON-FF* (voz) em estudo de filas, por exemplo, análise do tempo médio de espera, são incipientes e raras.

Foram considerados dois cenários de estudo: um sem aplicação do esquema de multiplexagem e o outro cenário com o esquema de multiplexagem dos quadros de voz. Em ambos os casos, estabeleceu-se que o tráfego de voz fosse constante, resultante de 30 canais de voz ativos no sistema. A carga de tráfego de dados foi variada para verificar o impacto tanto no atraso do tráfego de voz como no atraso do próprio tráfego de dados.

O estudo levou em conta um *link* de 2048 kbps e, as equações, no modelo analítico, foram parametrizadas tendo como variável independente a carga de tráfego de dados. Para aumentar a carga de tráfego de dados, foi mantido o tamanho de cada pacote de dados em 1000 bytes, e variado o ciclo de transmissão dos pacotes de dados. O recurso, a banda disponível, para dados foi o complemento da capacidade máxima do sistema relativo ao uso de recursos por voz. Isto é, o tráfego de dados teve disponível no caso de não uso de multiplexagem de até 41,41% da capacidade do sistema. E com a aplicação de multiplexagem pôde usar até 86,72% da capacidade do sistema.

Para o cenário sem a multiplexagem de quadro de voz a carga máxima de tráfego gerada pelos pacotes de dados foi de 41,41% da capacidade do enlace. Esta mesma carga de tráfego quando aplicada no cenário de multiplexagem de voz tem como carga de tráfego resultante (voz e dados) de 54,69%. Ou seja, o uso de recurso da rede IP para transmitir a mesma quantidade de informação diminuiu de 100% da capacidade para 54,69%.

Em ambos os cenários de política de escalonamento de tráfego, isto é, no escalonador *FIFO* (*Best Effort*), o método tradicional de encaminhamento de pacotes IP ou com o uso do escalonamento com prioridade, o tempo médio de espera teve diminuição se usado o esquema de multiplexagem de quadros.

No escalonamento com prioridade, que reproduz o ambiente de rede *IP* com *QoS*, o tempo médio de espera dos pacotes de dados ficou muito maior que o tempo médio de espera dos pacotes de voz sem a aplicação de multiplexagem e a carga de tráfego resultante superior à 70% da capacidade do enlace. A aplicação de multiplexagem nos quadros de voz proporcionou um tempo médio de espera dos pacotes de dados menor, próximo aos valores do tempo médio de espera dos pacotes de voz. Ou seja, a aplicação de prioridade, necessária para atender os requisitos de interatividade da aplicação de voz penalizou os tráfegos de dados. O uso eficiente de recursos da rede *IP* (esquema de multiplexagem de quadros de voz) minimizou o impacto resultante nos pacotes de dados. O tempo médio de espera de dados diminuiu tanto no esquema *FIFO* e principalmente no ambiente que aplica *QoS*.

Resultados semelhantes foram obtidos a partir do ambiente de simulação do *MatLab/SimEvent* em que as fontes de voz foram representadas por modelos *ON-OFF* e as fontes de dados por modelos *HTTP*. A estimativa de tráfego real, em ambos os casos, reproduziu a composição de distribuições estatísticas relacionadas. Neste ambiente as mesmas conclusões do modelo analítico são extraídas.

No ambiente de simulação foram extraídos também os histogramas referentes à distribuição estatística do tempo de espera por pacote (*ON-OFF* e *HTTP*) em cada cenário de análise proposto. Com isto é possível comparar o valor médio do tempo de espera com a distribuição estatística relacionada.

Como proposta de trabalhos futuros, pode-se analisar o impacto no tempo médio de espera, no ambiente de simulação do *MatLab/SimEvent*, utilizando diferentes fontes de tráfego de dados (tais como, *FTP*, vídeo, etc). E, também, efetuar a agregação desses tráfegos para compor uma única fonte de dados e analisar o desempenho de cada aplicação sem e com algum critério de priorização.

Referências Bibliográficas

1. Braden, B.; Clark, D.; Shenker, S.: “*An Architecture for Integrated Services*”; RFC-1633; Network Working Group; Junho 1994.
2. Mankin, A.; Baker, F.; Braden, B.; Bradner, S.; O`Dell, M.; Romanow, A.; Weinrib, A.; Zhang, L.: “*Resource ReSerVation Protocol (RSVP) Version 1 Applicability Statement Some Guidelines on Deployment*”; RFC-2208; Network Working Group; Setembro 1997.
3. Blake, S.; Black, D.; Carson, M.; Davies, E.; Wang, Z.; Weiss, W.: “*An Architecture for Differentiated Services*”; RFC-2475; Network Working Group; Dezembro 1998.
4. Rosen, E.; Viswanathan, A.; Callon, R.: “*Multiprotocol Label Switching Architecture*”; RFC-3031; Network Working Group; Janeiro 2001.
5. Douskalis, B.: “*IP Telephony, the Integration of Robust VoIP Services*”; Prentice Hall PTR, 2000.
6. Yamada, H.; Higuchi, N.: “*Voice Quality Evaluation of IP-Based Voice Stream Multiplexing Schemes*”; KDDI R&D Laboratories Inc.; Japan, 2001.
7. Park, K.; Willinger, W.; Editors: “*Self-Similar Network Traffic and Performance Evaluation*”; John Wiley & Sons, USA, 2000.
8. Karagiannis, T.; Molle, M.; Faloutsos M.; Broido, A.: “*A Nonstationary Poisson View of Internet Traffic*”; Proceeding of IEEE INFOCOM, 2004.
9. J. Cao, W. S. Cleveland, D. Lin, D. X. Sun, *Internet Traffic Tend Toward Poisson and Independent as the Load Increases*.
10. T. G. Robertazzi, *Computer Networks and Systems – Queueing Theory and Performance Evaluation*, second edition, Springer-Verlag, 1994.
11. S. Motoyama, *Fila de multi-usuários sem e com prioridade*. Apostila do curso IE670, aula 11, DT-Unicamp, 2005.
12. Sze, H.P., Liew, S.C., Lee, J.Y.B. and Yip, D.C.S, “*A multiplexing scheme for H.323 voice-over-IP applications*”, *IEEE J. Select. Areas Commun*, Vol. 20, pp. 1360-1368, Sept. 2002.
13. Paul T. Brady; *A model for generating on-off speech patterns in two way conversation*, *Bell System Technical Journal*; 48(9):2445-2472; September 1969.
14. Deng S. “*Traffic Characteristics of Packet Voice*” *IEEE Journal*, pages 1369-1374, (1995)
15. M. Evans, N. Hastings, B. Peacock, *Statistical Distributions*, second edition, 1993.
16. M. E. Crovella, A. Bestavros, *Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes*.

17. Choi, H. e Limb, J. (1999) "A Behavioral Model of Web Traffic", *Seventh International Conference on Network Protocols, ICNP '99*, Novembro 1999, p. 327 – 334.
18. R. Stewart, J. Schormans, "Accurate Dimensioning of Service Level Agreements in Voice-over-IP Networks", *Journal of the Institution of British Telecommunications Engineers*, Vol 2, part 3, July-September 2001;
19. Hoshi, Tohru, "Voice Stream Multiplexing Between IP Telephony Gateways," *IEICE Trans*, Apr. 1999.
20. Das Gupta, Jishu and Howard, Srecko and Howard, Angela, "Traffic behaviour of VoIP in a simulated access network. *International Transactions on Engineering, Computing and Technology*, 18 . pp. 189-194. ISSN 1305-5313, 2006.
21. Klepec, B., e Kos, A. "Performance of VoIP applications in a simple differentiated services network architecture". *EUROCON*, pp. 214--217, ACM Press, 2001.
22. B. Ahlgren, A. Andersson, O. Hagsand, I. Marsh, "Dimensioning Links for IP Telephony", T2000-09, *Swedish Institute of Computer Science (SICS)*, 2000
23. T. Karagiannis, M. Molle, M. Faloutsos, and A. Broido, "A Nonstationary Poisson View of Internet Traffic," in *INFOCOM 2004*. March 2004, *INFOCOM 2004*.
24. W. Wang, S. C. Liew, Q. X. Pang, and V. O. Li, "A Multiplex-Multicast Scheme that Improves System Capacity of Voice-over-IP on Wireless LAN by 100%," in *Proceedings of the Ninth IEEE Symposium on Computers and Communications (ISCC)*, 2004.

Apêndice A – Tempo de Espera (Simulação)

Neste apêndice são apresentados os histogramas normalizados da distribuição do tempo de espera em cada fila. No capítulo 04 foi usado o tempo médio de espera em cada fila, com a política de escalonamento FIFO e fila com prioridade, extraído de cada cenário (simulação). Neste caso é possível verificar a distribuição estatística dos tempos de espera, conforme carga de tráfego total relacionada, e prover informações complementares dos valores médios obtidos em cada situação.

A.1 Análise de Resultados (Matlab/SimEvent)

A figura A.1 representa o tempo de espera dos pacotes de voz e dados no cenário do esquema *FIFO* (Melhor Esforço) com uso da ferramenta *SimEvent* e sem aplicar a multiplexagem nos quadros de voz.

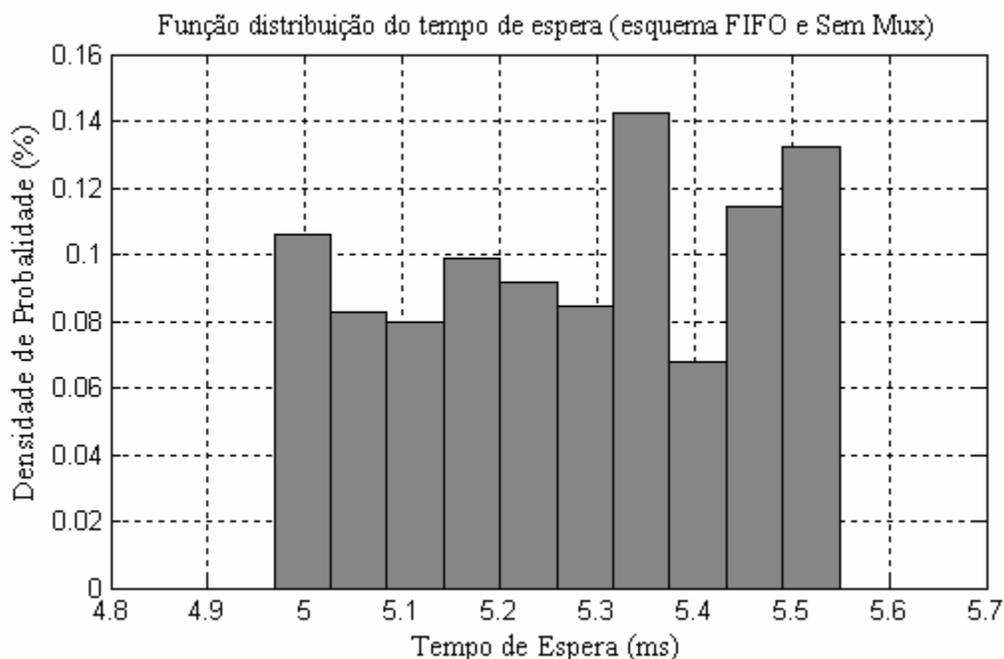


Figura A.1 – Distribuição do tempo de espera fila *FIFO* (Sem Mux e 69,4% de carga)

Nesta figura é exibido o histograma normalizado da distribuição do tempo de espera na fila (SimEvent) tratada pelo escalonador *FIFO* com servidor simples e sem preempção usado na simulação. Neste caso o índice de utilização dos pacotes *ON-OFF* é 53% enquanto o índice de utilização dos pacotes *HTTP* é 16,57%.

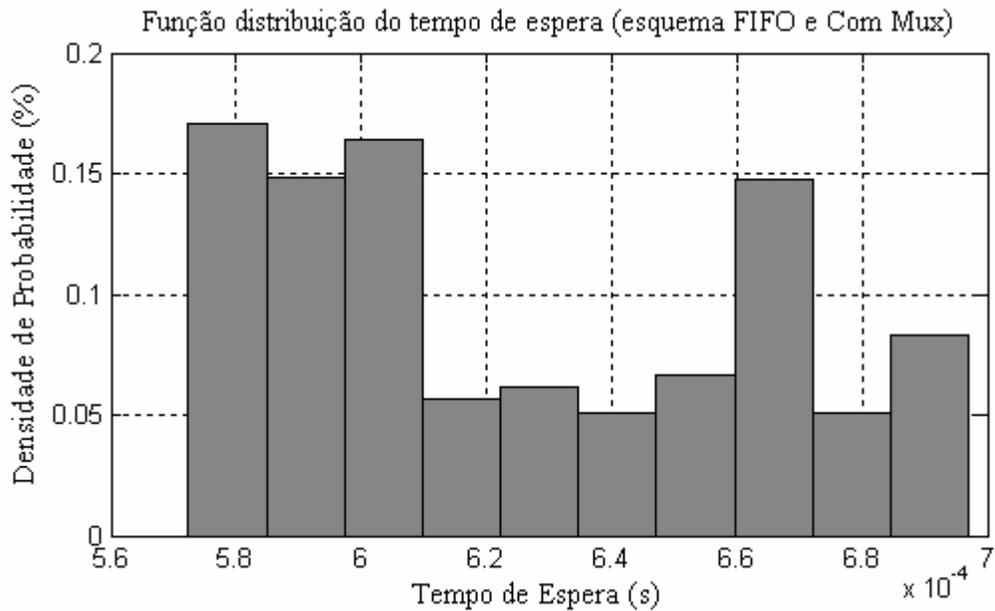


Figura A.2 – Distribuição do tempo de espera fila *FIFO* (Com Mux e 25,61% de carga)

A figura A.2 representa o tempo de espera dos pacotes no cenário do esquema *FIFO* (Melhor Esforço) da ferramenta *SimEvent* e com aplicação multiplexagem nos quadros de voz. É exibido o histograma normalizado da distribuição do tempo de espera na fila (*SimEvent*) tratada pelo escalonador *FIFO* com servidor simples e sem preempção usado na simulação. Neste caso o índice de utilização dos pacotes *ON-OFF* é 12,06% enquanto o índice de utilização dos pacotes *HTTP* é 13,55%.

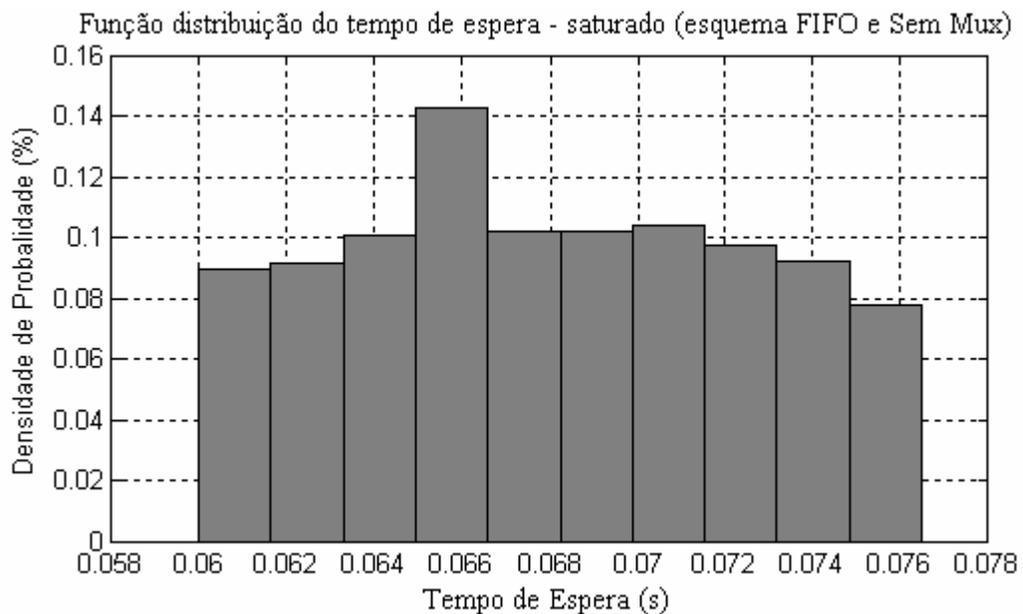


Figura A.3 – Distribuição do tempo de espera fila *FIFO* (Sem Mux e 96,74% de carga)

A figura A.3 representa o tempo de espera dos pacotes no cenário do esquema *FIFO* (Melhor Esforço) sem a aplicação de multiplexagem nos quadros de voz. O índice de utilização dos pacotes *ON-OFF* é 51,61% enquanto o índice de utilização dos pacotes *HTTP* é 45,13%.

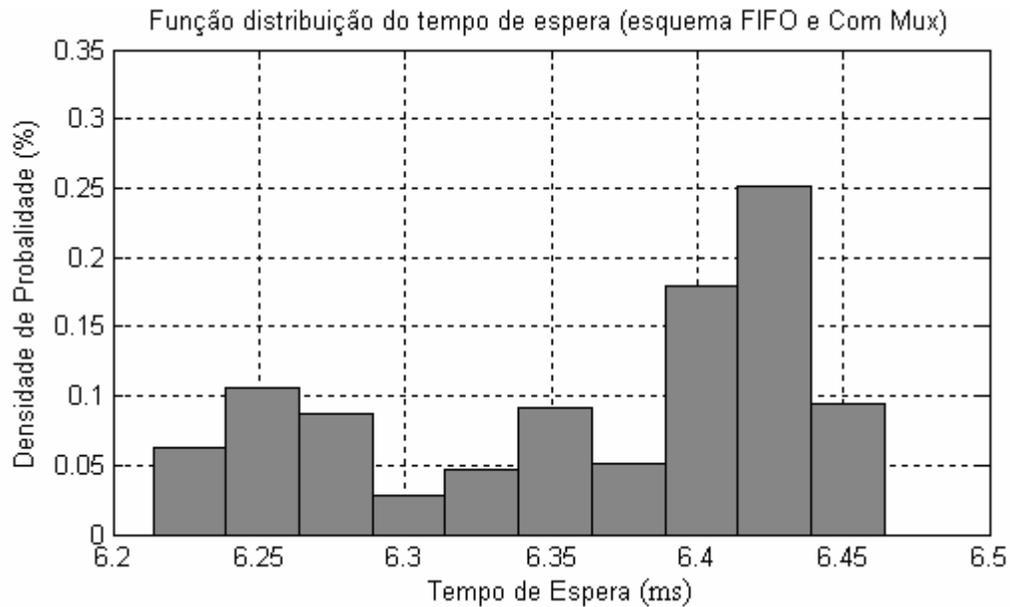


Figura A.4 – Distribuição do tempo de espera fila *FIFO* (Com Mux e 60,84% de carga)

A figura A.4 representa o tempo de espera dos pacotes no cenário do esquema *FIFO* (Melhor Esforço) com a aplicação de multiplexagem nos quadros de voz. O índice de utilização dos pacotes *ON-OFF* é 12,46% enquanto o índice de utilização dos pacotes *HTTP* é 48,38%.

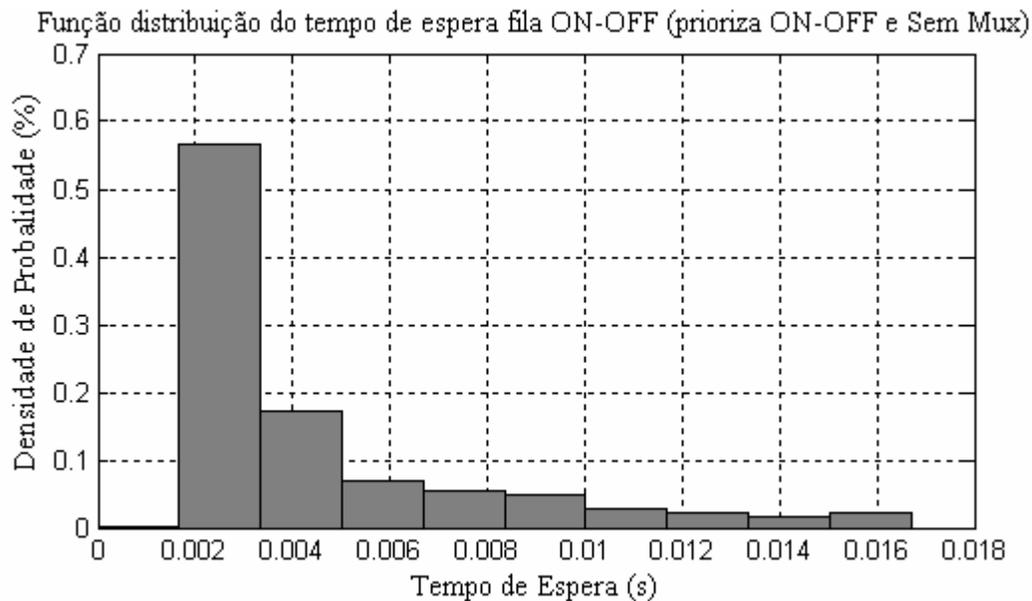


Figura A.5 – Distribuição do tempo de espera fila *ON-OFF* (Sem Mux e 69,4% de carga)

As figuras a seguir representam o tempo de espera dos pacotes ON-OFF e dos pacotes HTTP no cenário de fila com prioridade (aplicação de QoS) e uso da ferramenta SimEvent. Neste cenário é extraído de cada fila, ou ON-OFF (prioridade alta) ou HTTP (prioridade baixa), o tempo de espera de cada pacote em cada fila. Em cada caso é considerado o esquema sem aplicação de multiplexagem nos quadros de voz. É também considerado o cenário com aplicação de multiplexagem. A figura A.5 exibe o histograma normalizado da distribuição do tempo de espera na fila ON-OFF sem aplicação de multiplexagem. O índice de utilização dos pacotes ON-OFF é 53% enquanto o índice de utilização dos pacotes HTTP é 16,57%.

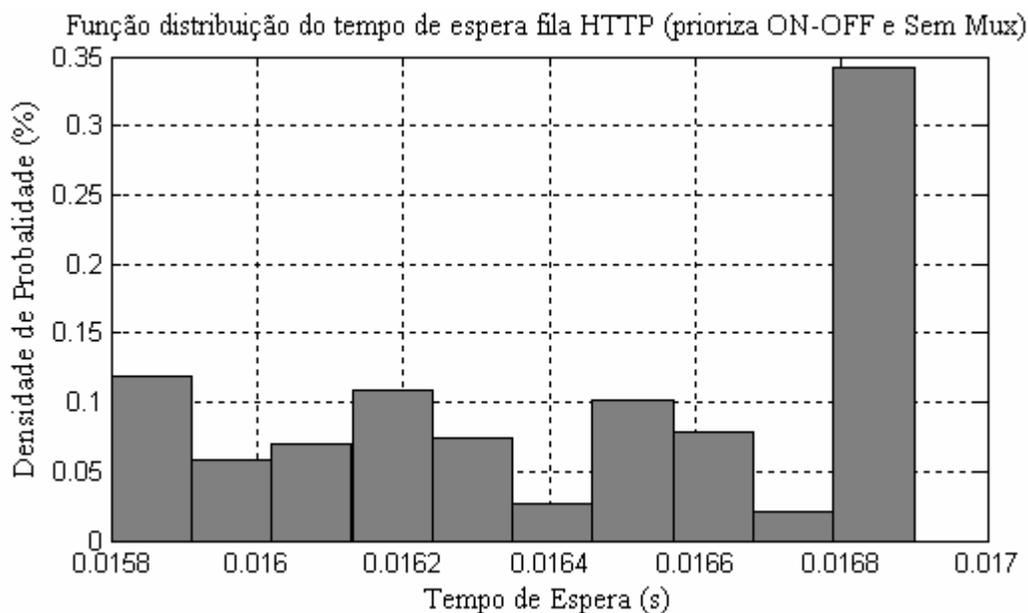


Figura A.6 – Distribuição do tempo de espera fila HTTP (Sem Mux e 69,4% de carga)

A figura A.6 exibe, para a mesma carga de tráfego considerada na figura A.5, o histograma normalizado da distribuição do tempo de espera na fila HTTP sem aplicação de multiplexagem.

As figuras A.7 e A.8 exibem o histograma normalizado da distribuição do tempo de espera na fila ON-OFF e na fila HTTP com aplicação do esquema de multiplexagem dos quadros de voz.

O índice de utilização dos pacotes ON-OFF é 12,06% enquanto o índice de utilização dos pacotes HTTP é 13,55%. A carga de tráfego resultante ficou menor. Esta redução se deve ao uso mais eficiente, pelos pacotes ON-OFF, de recursos da rede IP

O tempo médio de espera, quando o sistema de 2048 kbps apresenta saturação (limite da capacidade), apresenta incremento significativo. Principalmente se aplicado política de priorização.

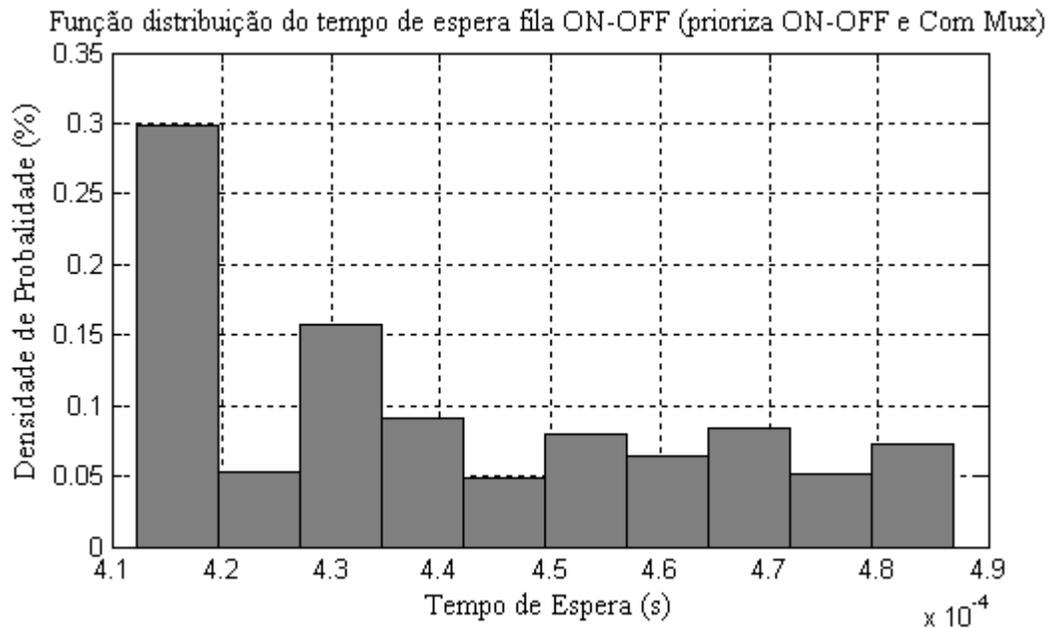


Figura A.7 – Distribuição do tempo de espera fila *ON-OFF* (Com Mux e 25,61% de carga)

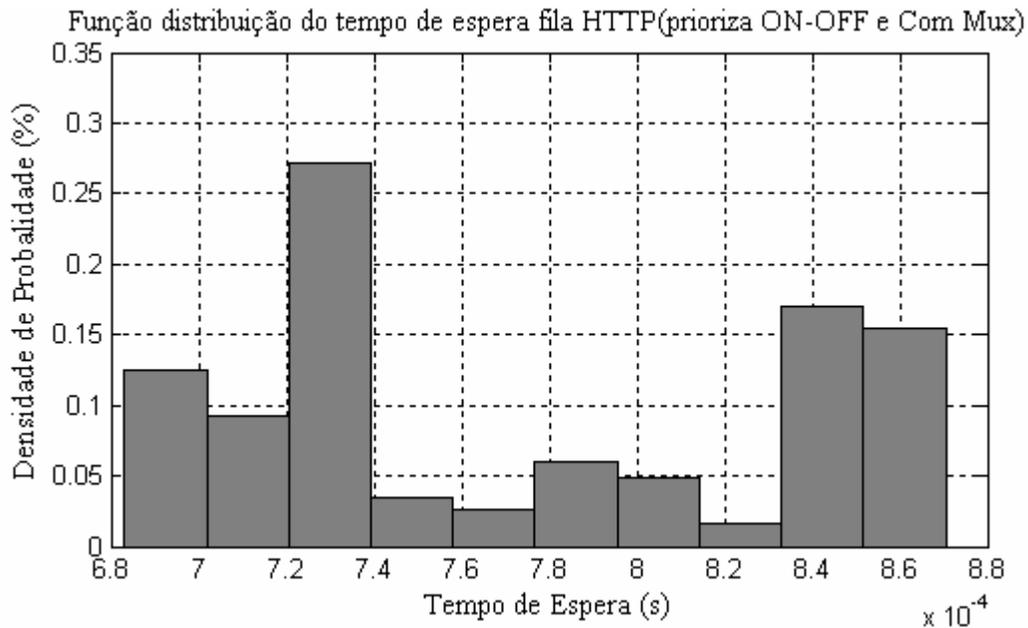


Figura A.8 – Distribuição do tempo de espera fila *HTTP* (Com Mux e 25,61% de carga)

As figuras A.9 e A.10 representam o tempo de espera na fila *ON-OFF* e *HTTP* respectivamente. O índice de utilização dos pacotes *ON-OFF* é 51,61% enquanto o índice de utilização dos pacotes *HTTP* é 45,13%. A não aplicação de multiplexagem dos quadros de voz resulta em penalidade para o tráfego de pacotes *HTTP*, isto é, aumento significativo do tempo de espera.

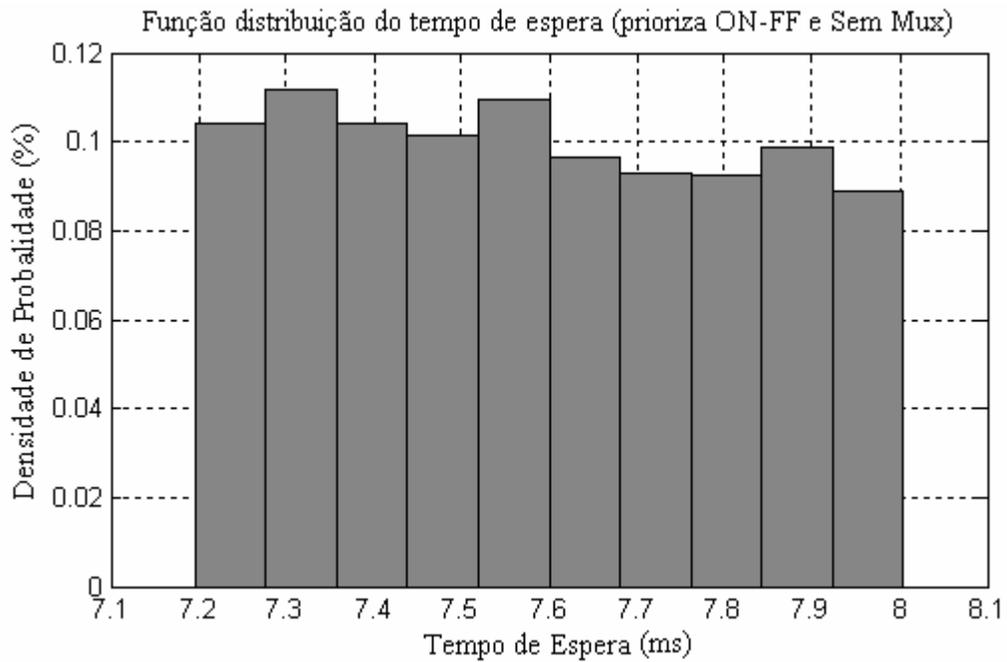


Figura A.9 – Distribuição do tempo de espera fila *ON-OFF* (Sem Mux e 96,74% de carga)

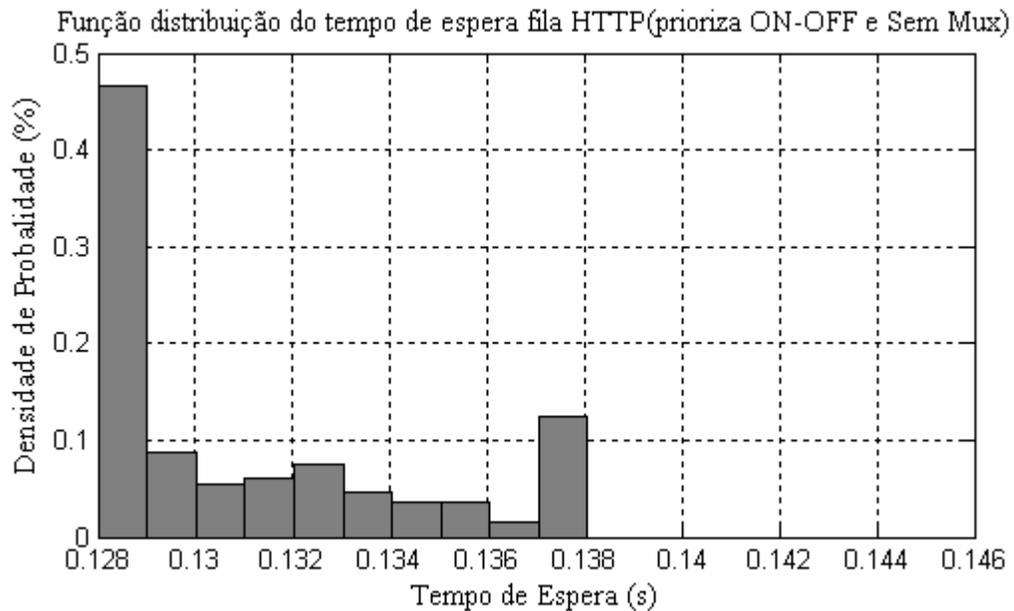


Figura A.10 – Distribuição do tempo de espera fila *HTTP* (Sem Mux e 96,74% de carga)

As figuras A.11 e A.12 representam o tempo de espera na fila *ON-OFF* e *HTTP* respectivamente. O índice de utilização dos pacotes *ON-OFF* é 12,06% enquanto o índice de utilização dos pacotes *HTTP* é 48,38%. A não aplicação de multiplexagem dos quadros de voz resulta em penalidade para o tráfego de pacotes *HTTP*, isto é, aumento significativo do tempo de espera.

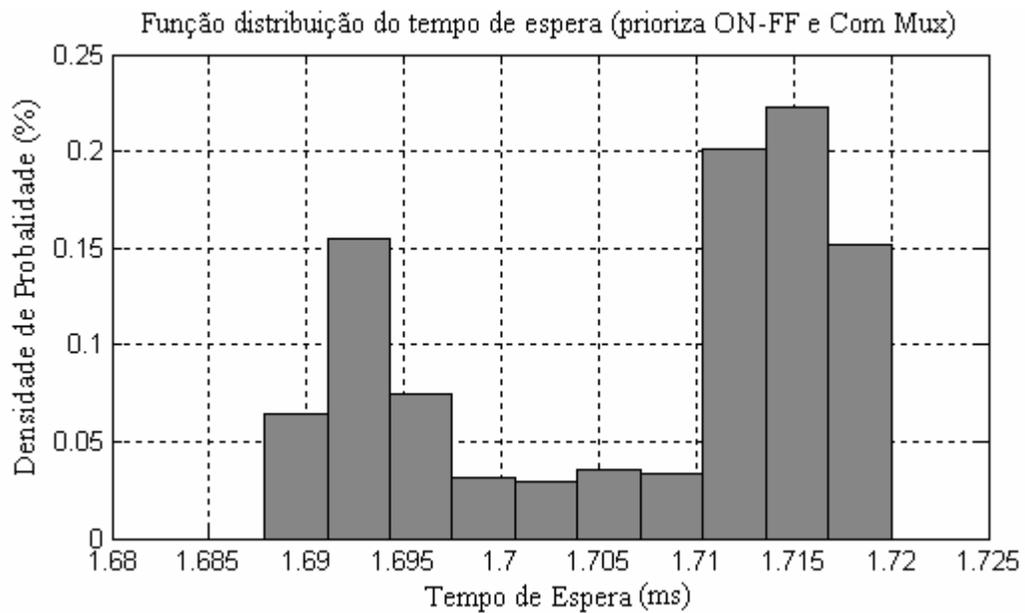


Figura A.11 – Distribuição do tempo de espera fila *ON-OFF* (Com Mux e 60,84% de carga)

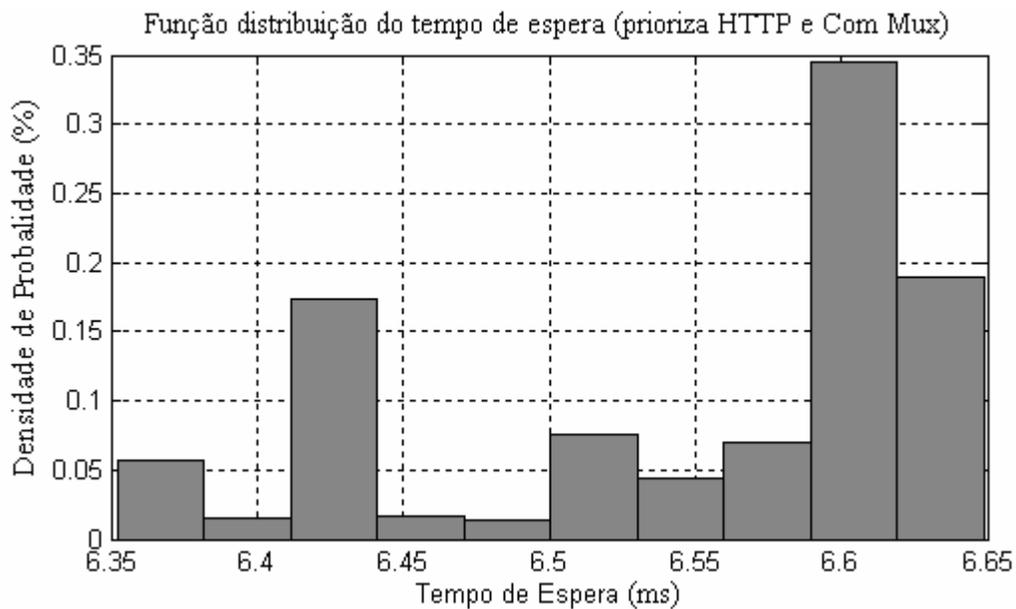


Figura A.12 – Distribuição do tempo de espera fila *HTTP* (Com Mux e 60,84% de carga)

A partir do uso da distribuição do tempo de espera é possível obter o tempo médio de espera dos pacotes *ON-OFF* e *HTTP* em cada fila. Os histogramas normalizados expressam a distribuição estatística do tempo de espera dos pacotes em cada fila tratada.

Apêndice B – Modelos de filas no *SimEvent*

Neste apêndice são apresentados os diagramas em blocos que provêm as funcionalidades do *SimEvent* para tratar eventos discretos através do esquema *FIFO* e também com escalonador que aplica priorização. Na figura B1 os eventos gerados no *MatLab* (tamanho do pacote e tempo de chegada) são usados através da interface, por exemplo, *from_workspace_tamanho_voz* e *from_workspace_tempo_voz*.

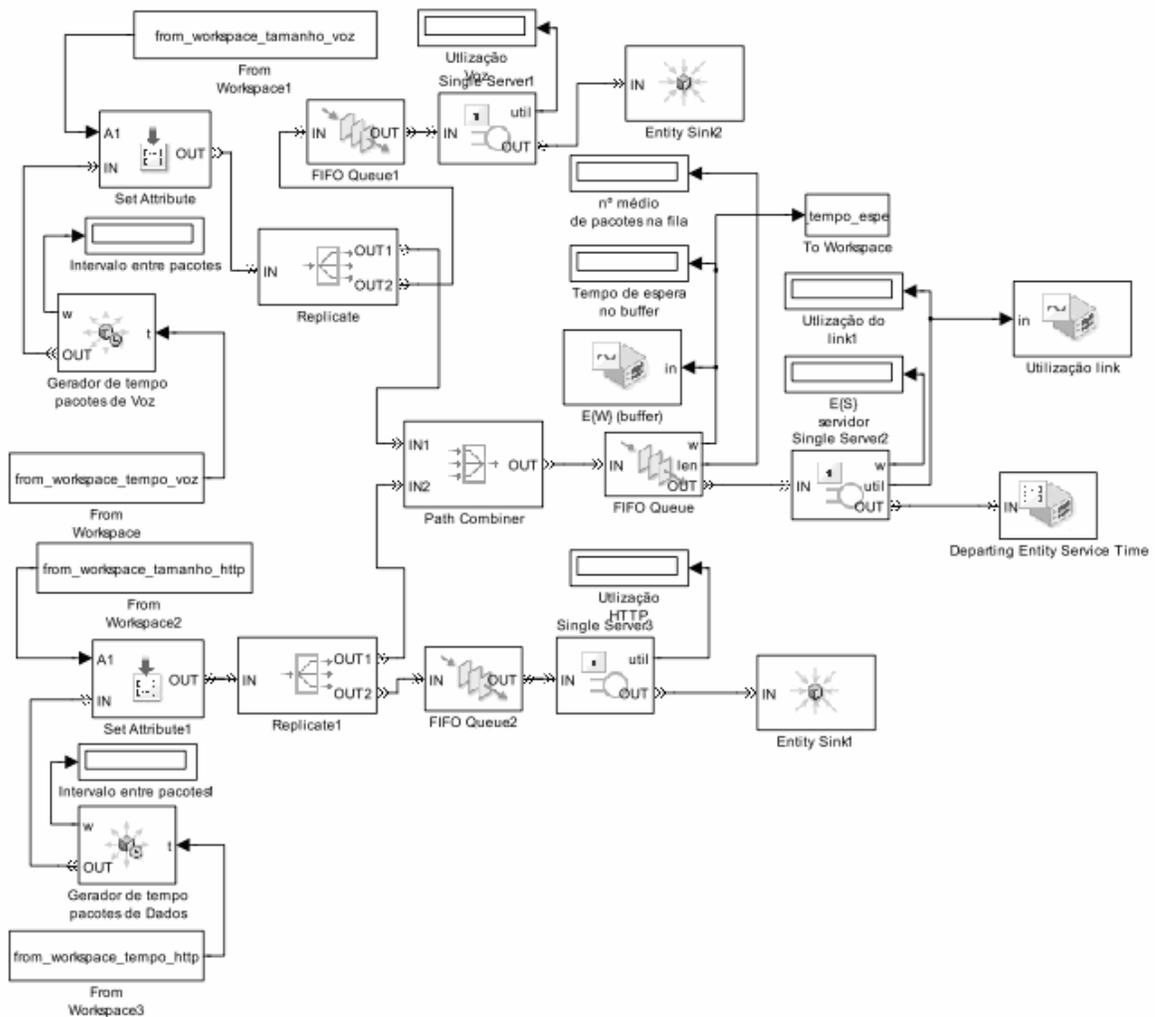


Figura B.1 – Diagrama em blocos – esquema de fila FIFO

Com este procedimento é possível reproduzir no *SimEvent* a estatística gerada no *MatLab*. No bloco *Set Attribute* são gerados eventos (pacotes) que possuem os tamanhos, já definidos no *MatLab*, e em seguida cada evento é encaminhado ao conjunto Escalonador FIFO ou Escalonador por Priorização seguido de servidor. Para medir o tráfego de dados ou de voz, de forma isolada, os eventos são replicados e em cada conjunto de escalonador e servidor é extraída a informação de

utilização parcial do enlace. Para medir o tráfego total os eventos são combinados (justapostos) e encaminhados ao conjunto de escalonador e servidor do sistema. No escalonador do sistema é possível extrair o tempo médio de espera na fila. No caso do escalonador *FIFO*, conforme fig.B1, o tempo médio de espera é o mesmo para voz e dados. Já no caso de escalonador com priorização, mostrado na figura B2, os tempos médios de espera de voz e dados são obtidos através da inserção de blocos *Start Timer* que permitem a marcação dos eventos de voz e dados antes da chegada na fila. Após permanência na fila, já na saída deste sistema, é extraído o tempo médio de permanência em cada fila específica de cada evento de dados ou de voz.

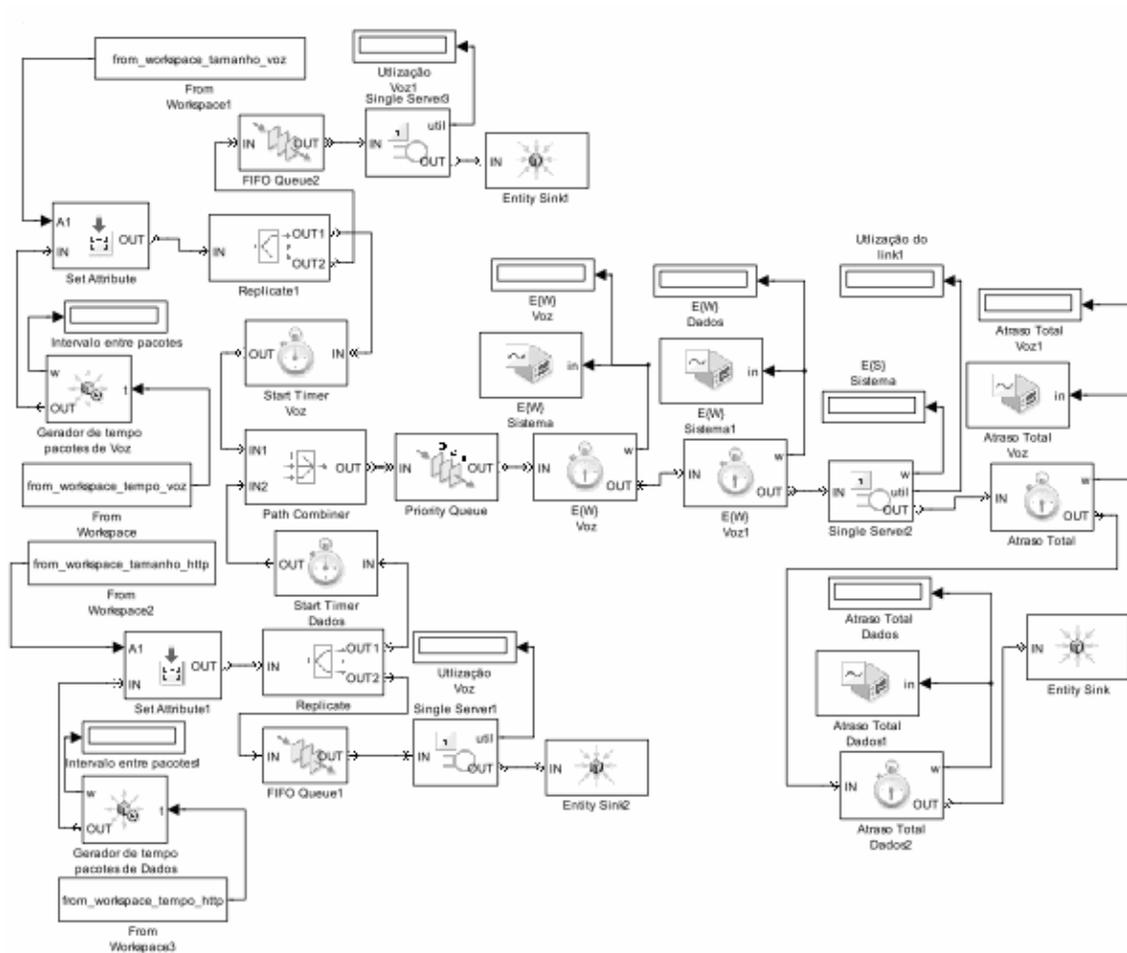


Figura B.2 – Diagrama em blocos – esquema de fila com prioridade

Apêndice C – Trabalho Publicado no *International Workshop on Telecommunications – IWT2007*

“Voice Packet Delay Evaluation in an IP Network Using Voice Stream Multiplexing Scheme”

Carlos Fielde de Campos

Empresa Brasileira de Telecomunicações – EMBRATEL ,
R. Bernardino de Campos, 799 2º andar,
Zip Code 13010-151,
Campinas-SP, Brazil.
Email: fielde@embratel.com.br

Shusaburo Motoyama

Department of Telematics, State University of Campinas,
PO box 6101, Zip Code 13083-970,
Campinas-SP, Brazil.
Email: motoyama@dt.fee.unicamp.br

Abstract- In this paper the performance of voice packet in an IP network is evaluated. Since voice packets have larger overhead when compared to data packets the intense increasing in voice traffic as is expected in near future may become IP network inefficient. To overcome or minimize this problem the voice stream multiplexing scheme can be used. A mathematical model is proposed in this paper to study the impact of voice packet traffic in an IP network. The study is carried out considering voice packets without and with multiplexing and performance parameter is the packet delay. The study showed that voice packet multiplexing is an efficient scheme to save the bandwidth of a link.

Index Terms— VoIP, Multiplexing, Modeling, Performance Evaluation.

I. INTRODUCTION

IP network has been used as a platform to integrate all kind of traffics. However, for integration of traffics such as voice, video and data which each one demands a different Quality of Service (QoS), the current IP network, for instance the Internet, is not prepared to supply such a distinct quality for each traffic. One of approaches to solve this problem presented by IETF is DiffServ [1]. In the DiffServ solution the packets are classified into flow classes. Each class of flow can be treated in a same manner in the whole network. For example, since the telephonic conversations have more stringent time delay [2] and must be treated as an almost real time traffic, voice packets can be classified in a flow and treated with high priority in each node in the entire network.

Another important issue to be addressed is the length of voice packet. For the transmission of voice in an IP network the analog voice stream is compressed and transformed in a digital voice stream by using a coder and in the reception, the digital voice stream is decoded in analog voice stream again. These codification and decodification processes are usually known as CODEC. Before the digital stream is transmitted into the IP network the voice stream is packed in frames and in each frame an overhead composed

by RTP (Real Time Protocol), UDP (User Data Protocol) and IP (Internet Protocol) is added, corresponding total overhead of 40 bytes. The length of the frame depends on the type of CODEC. For example, for the G.729 CODEC the frame length of 10 ms can accommodate 10 bytes, 20 ms can accommodate 20 bytes and so on. As the length of frame becomes larger, more bits can be accommodated in each frame, however, this frame delay cannot be large because this delay added to network delay can affect the naturalness of voice conversation. Thus, the frame length must be kept small which becomes the voice transmission in the IP network very inefficient. For instance, if the frame length is 10 bytes and since the overhead is 40 bytes, the efficiency is only 20%. For larger frames the efficiency will be better but still very inefficient. In [3] a multiplexing technique is proposed to improve the voice packet transmission efficiency. In the multiplexing technique proposed in [3], the voice frames of different conversations are multiplexed before the RTP, UDP and IP overhead is added as is shown in Fig. 1.

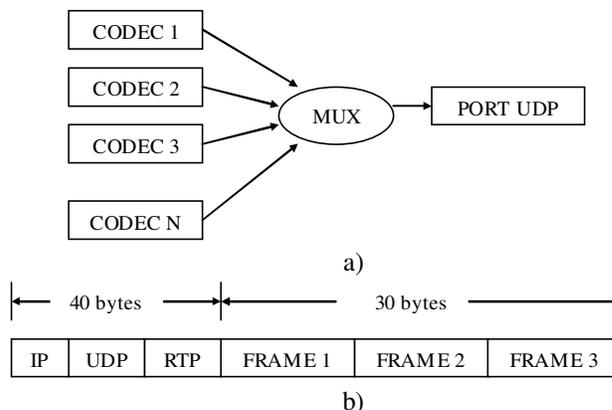


Figure 1. A multiplexing technique to improve the voice packet transmission.

As can be observed in the example of Fig. 1b three frames of 10 bytes are multiplexed resulting in an efficiency of 42,85 % instead of 20 % when frames are transmitted separately. The main study carried out in [3] was to evaluate how the loss of multiplexed packet can affect the voice quality. The whole study was carried out by experimental means. The study in [3] does not address the impact of multiplexed packet delay through the network. In this paper, it is studied the impact of the voice packet with and without multiplexing in the network performance. The study in this paper is carried out by mathematical analysis.

The paper is organized as follows. In section II the mathematical model for packet delay performance study is proposed. The main results of performance study are shown in section III. Finally, in section IV main conclusions are presented.

II. THE MATHEMATICAL MODEL FOR PACKET DELAY PERFORMANCE STUDY

In many cases the traffic in IP network is modeled as self-similar, leading to a very difficult traffic modeling and network performance analysis [4]. However, in the recent experimental measurement [5], in the high aggregate traffic, it was found that the input traffic can be modeled as Poisson and the service distribution is independent of input traffic in a small scale of time. Thus, based on the findings of [5], the classic queuing theory will be used for the network performance analysis.

For the performance analyses following assumptions are made. The voice arrival rate and data arrival rate are denoted λ_V and λ_D , respectively. The voice packet length is denoted $1/\mu_V$ and the data packet length as $1/\mu_D$, both measured in bytes. The arrival distributions of both voice and data packets are Poisson, and the voice packet length distribution is fixed and data packet length is negative exponential distribution.

Let x and y be the proportions representing the percentages of voice traffic and voice length in relation to the data traffic and data length, respectively. The following equations can be written.

$$\lambda_V = x \cdot \lambda_D \quad \mu_V = y \cdot \mu_D \quad (1)$$

Since the voice packet length, in general, is smaller than data packet length, it results from Eq.1 that $y > 1$. On the other hand, the voice arrival rate is, in general, smaller than data arrival rate, but in near future may become as high as data rate thus the variable x can be greater than 1. Furthermore, it can be written.

$$\rho_D = \frac{\lambda_D}{C \cdot \mu_D} \quad \text{and} \quad \rho_V = \frac{\lambda_V}{C \cdot \mu_V} \quad (2)$$

Where ρ_D and ρ_V are traffic loads for data and voice, respectively, and C is the output link capacity. The Eq.1 can be inserted into Eq.2, and Eq. 3 is obtained.

$$\rho_V = \frac{x \cdot \lambda_D}{C \cdot y \cdot \mu_D} \quad (3)$$

Equation 3 expresses the dependence of voice traffic load in relation to data traffic load. Eq.4 gives the total traffic load applied in a link. In today IP Network, it has the situation that $\rho_V \ll \rho_D$, however, this situation is transitory because there is a strong perspective of migration of conventional telephone call to Internet. This service known as VoIP is attractive mainly because the reduction in the cost.

$$\rho_{TOTAL} = \rho_V + \rho_D \quad (4)$$

By inserting Eqs.2 and 3 into Eq.4 the total load can be written as presented in Eq. 5.

$$\rho_{TOTAL} = \frac{x \cdot \lambda_D}{C \cdot y \cdot \mu_D} + \frac{\lambda_D}{C \mu_D} \quad (5)$$

Equation 5 represents the total traffic load using only data traffic parameters and the x and y variables.

A network node can be modeled as a simple buffer and only one output link of capacity C . Considering a multi-types of traffic arrivals and FIFO discipline, the expression for average waiting time $E\{W\}$ is given by:

$$E\{W\} = \frac{E\{T_0\}}{1 - \rho_{TOTAL}} \quad (6)$$

Where $E\{T_0\}$ is the residual waiting time of service, assuming no preemption strategy. $E\{T_0\}$ is given by:

$$E\{T_0\} = \frac{1}{2} \cdot \lambda_V \cdot E\{R_V^2\} + \frac{1}{2} \cdot \lambda_D \cdot E\{R_D^2\} \quad (7)$$

The terms $E\{R_V^2\}$ and $E\{R_D^2\}$ represent the second moment of voice and data length distributions, respectively.

Assuming that voice packet length distribution has fixed length and data length has negative exponential distribution it can be written:

$$E\{R_V\}^2 = \frac{1}{(C \cdot y \cdot \mu_D)^2} \quad (8)$$

and

$$E\{R_D\}^2 = \frac{1}{(C \cdot \mu_D)^2} \quad (9)$$

Thus, the residual waiting time of service is given by:

$$E\{T_0\} = \frac{1}{2} \cdot \left(\frac{x \cdot \lambda_D}{(C \cdot y \cdot \mu_D)^2} + \frac{2 \cdot \lambda_D}{(C \cdot \mu_D)^2} \right) \quad (10)$$

Using Eq.1 and Eq.2, $E\{T_0\}$ can be written as:

$$E\{T_0\} = \frac{\rho_D}{C \cdot \mu_D} \cdot \left(\frac{1}{2} \cdot \frac{x}{y^2} + 1 \right) \quad (11)$$

By inserting Eqs.5 and 11 into Eq.6 and after some mathematical manipulations the following expression can be written for average waiting time $E\{W\}$ for voice and data packets:

$$E\{W\} = \frac{\frac{\rho_D}{C \cdot \mu_D} \cdot \left[\frac{1}{2} \cdot \frac{x}{y^2} + 1 \right]}{1 - \rho_D \cdot \left(\frac{x}{y} + 1 \right)} \quad (12)$$

Equation 12 represents the average waiting time of packets without priority. This equation allows the study of effects of x and y variables in the waiting time.

If the voice packets have priority over data packets, the expression for priority queue without preemption can be used. In this case the expressions for average waiting time for voice traffic, $E\{W_V\}$, and average waiting time for data traffic, $E\{W_D\}$, are given by:

$$E\{W_V\} = \frac{E\{T_0\}}{(1 - \rho_V)} \quad (13)$$

and

$$E\{W_D\} = \frac{E\{T_0\}}{(1 - \rho_V) \cdot (1 - \rho_V - \rho_D)} \quad (14)$$

$E\{T_0\}$ is the same expression defined in the FIFO scheme. By inserting Eqs. 2 and 11 into Eqs. 13 and 14 it can be obtained the expressions for voice and data waiting times.

$$E\{W_V\} = \frac{\frac{\rho_D}{C \cdot \mu_D} \cdot \left[\frac{1}{2} \cdot \frac{x}{y^2} + 1 \right]}{\left(1 - \frac{x}{y} \cdot \rho_D \right)} \quad (15)$$

$$E\{W_D\} = \frac{\frac{\rho_D}{\mu_D} \cdot \left[\frac{1}{2} \cdot \frac{x}{y^2} + 1 \right]}{1 - \rho_D \cdot \left(2 \cdot \frac{x}{y} + 1 \right) + \rho_D^2 \left(\frac{x^2}{y^2} + \frac{x}{y} \right)} \quad (16)$$

By using Eqs. 12, 15 and 16 it is possible to study the network performance when the voice packets are multiplexed or not for the FIFO situation or in the case that have priority. These equations permit to analyze and compare the impact of voice traffic load in an IP network. The voice application needs a special treatment to have good performance in the packet switching network. The priority scheduling can solve this requirement. However, low priority traffic will have performance degradation. To minimize the impact due to voice traffic load in an IP network it is necessary to reduce the bandwidth utilization inefficiency by voice packets. One of approaches to solve this problem is the voice frame multiplexing technique.

III. ANALYSIS OF RESULTS

In the today IP network such as Internet is implemented the best effort policy to deal with the packets, which means that there is no guarantee of packet delivery or latency control. In the proposed mathematical modeling this situation can be studied in part by using FIFO scheme.

When a network platform supports QoS as in DiffServ case, it is possible to provide priority scheduling. To implement this functionality is necessary to create different kinds of queuing (class of service) and establishing priority for each traffic. At node the server treats the traffic from high to low priority queuing. In this paper two different classes are considered: voice traffic, which has highest priority, and data traffic in second priority.

The mathematical models presented in Eqs. 15 and 16 are used to analyze the performance in these cases.

Now some numerical examples are given to examine the IP network performance. The following numerical values are assumed when 30 voice sources are actives with and without multiplexing. This number of sources is equivalent to one E1 standard, which is the European standard for digital transmission. E1 carries signals at 2 Mbps (32 channels at 64Kbps, with 2 channels reserved for signaling and controlling).

The voice packets follow the standard - G.729 with voice frame of 10 bytes (transmission cycle equal to 10 ms). In IP Network each voice packet must include 40 bytes of header corresponding to overhead of RTP, UDP and IP protocol. Without MUX, the voice IP packet is 50 bytes (voice frame is 10 bytes) and with MUX results 340 bytes (30 different voice frames of 10 bytes). Without MUX and 30 voice channels from CODEC it results equivalent

transmission cycle of voice IP packet with $\frac{10}{30}$ ms. This

voice traffic load is kept constant representing in this case intense voice traffic situation. The average length of data packets is 1000 bytes and the data sources generate traffic varying from 0 bps up to 2048 kbps. The link capacity is 2048 kbps.

With above consideration the bandwidth utilization by voice traffic without MUX is 1,2Mbps and with MUX is 272kbps which represent 58,59% and 13,28%, respectively of link occupancy

Four different scenarios are considered. In the first scenario the voice and data traffics are served in a FIFO scheme and there is no voice frame multiplexing. In second scenario the same scheduling of first scenario is used but now with multiplexing. The third scenario deals with priority scheduling considering highest priority for voice packet but without multiplexing, and the fourth scenario is the same as third but with multiplexing. To analyze these four scenarios the parameters shown in Table 1 are used.

Table 1 – Parameters used in the numerical analysis.

Multiplexing scheme	$\frac{1}{\lambda_V}$	$\frac{1}{\mu_V}$	$\frac{1}{\lambda_D}$	$\frac{1}{\mu_D}$
Without	$\frac{10}{30} ms$	50 bytes	Variable	1000 bytes
With	10ms	340 bytes	Variable	1000 bytes

Fig. 2 shows the waiting times for FIFO case (first and second scenarios) and using Eq. 12. To increase gradually data traffic load it is initially necessary to decrease transmission cycle for data packets. The relationship between x and y and values of ρ_D and ρ_V using or not the multiplexing scheme are shown in Table 2.

Table 2 – x and y proportions considering link utilization without and with MUX.

Multiplexing scheme	x	y	ρ_D	ρ_V	ρ_{TOTAL}
Without	234,38	20	0,05	0,5859	0,6359
	102,71	20	0,1141	0,5859	0,7000
	28,30	20	0,4141	0,5859	1,0000
With	7,81	3	0,05	0,1328	0,1828
	3,42	3	0,1141	0,1328	0,2469
	0,94	3	0,4141	0,1328	0,5469

As it can be observed in Table 2, x and y values with MUX are smaller than without MUX. This is consequence of adoption of constant load for voice traffic and different voice packet sizes when the multiplexing technique is used or not. Furthermore, when $\rho_V > \rho_D$ it can be found $x > y$, and when $\rho_V < \rho_D$ it has $x < y$ for without and with MUX cases.

In Fig. 2 it can be noticed that the average waiting time of packets for multiplexed case decreased. The link utilization in figure is related to data traffic load only once voice traffic load is constant. The voice traffic load is 0.5859

E (Erlang) without multiplexing and with multiplexing scheme is 0.1328 E. The improvement is better for high load, normally when total link utilization reaches a threshold (a maximum practical load for a stable network operation) of 70% of link capacity, but for low load the multiplexing effort is unnecessary because the average waiting times for both cases are almost same. Using above threshold the data traffic load can reach 0,1151 E and 0,5672 E without and with multiplexing scheme, respectively. If this threshold is exceeded the traffic load can be 100% of link utilization which means a highly saturated operation. For 100% of occupation the data loads are 0,4141 E and 0,8672 E for without and with multiplexing cases, respectively.

Table 3 – Relationship among x , multiplexing scheme and link occupancy

Multiplexing scheme	Traffic load	x
Without	70,0% link capacity	102,71
With	24,7% link capacity	0,45
Without	100% link capacity	28,3
With	54,7% link capacity	4,5

Table 3 shows the relationship among x , link occupancy and multiplexing schemes for limiting and saturated operation cases.

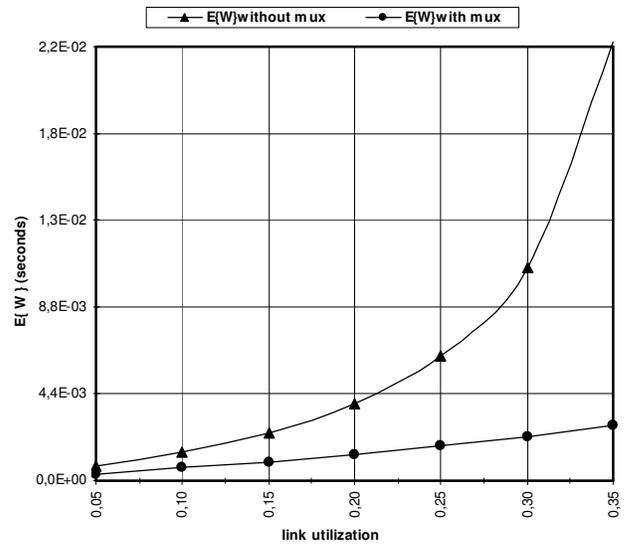


Figure 2. Waiting times for voice and data packets in the case of FIFO discipline.

It can be concluded by Table 3 that if the same amount of data and voice information are transferred by 2Mbps link capacity less bandwidth is necessary in the voice multiplexing scheme case.

As a general conclusion it can be observed that using MUX the system saves bandwidth for voice traffic and data traffic has more bandwidth available.

In the MUX case the load for the threshold region where data link capacity starts to saturate is higher than without MUX. In Table 4 the waiting times for two situations are shown. In first situation without multiplexing the load for threshold region is 0,7 E and in the second situation with multiplexing the load is 0,2469 E. The waiting times are calculated using Eq. 12. As can be seen by Table 4 for the transport of same amount of information the multiplexing technique can bring improvement in waiting time.

Table 4 - Waiting time for 30 voice channels and data packets in the case of FIFO discipline

Multiplexing scheme	E{W}
Without	1,676ms
With	0,708ms

In Fig. 3 the third scenario with priority for voice packets and without multiplexing is shown.

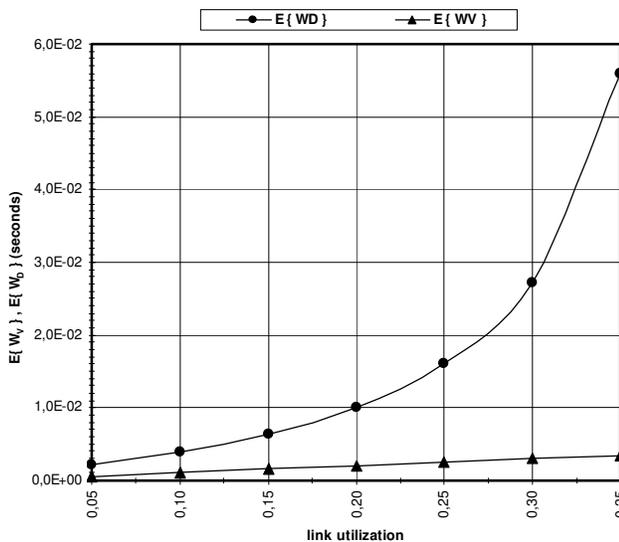


Figure 3 - Waiting times for voice and data packets in the case of priority discipline and without multiplexing.

In Fig. 3 it is possible to notice the voice priority impact on both data and voice average waiting times. The voice performance is significantly better than data performance. Comparing Figs. 2 and 3, it is clear that the data packets are penalized for better performance of voice packets. To compensate this characteristic is necessary to use efficiently IP transport structure, which can be obtained by frame multiplexing technique.

Fig. 4 shows the case for priority discipline and with multiplexing (fourth scenario). In this case both data and voice traffics have considerable performance improvement. Comparing to Fig.2 the results are very interesting. Not only voice packet has improvement but data

packet has too. These results are interesting because they prove that the voice multiplexing scheme is very important for performance improvement in an IP network.

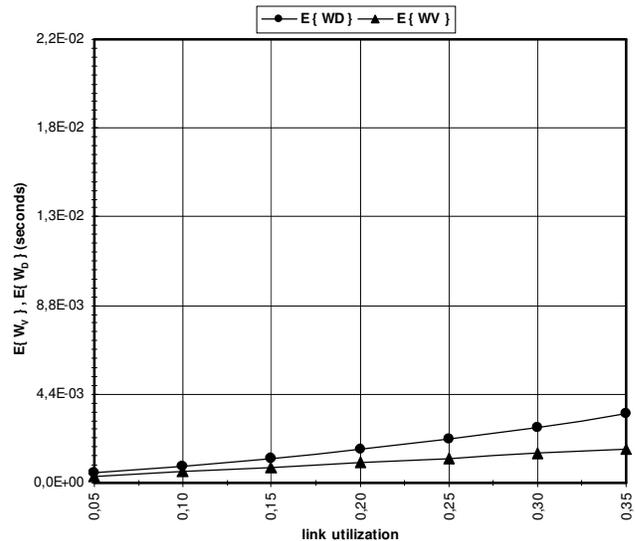


Figure 4 Waiting times for voice and data packets in the case of priority discipline and with multiplexing

IV. CONCLUSIONS

In this paper the performance of voice packet in an IP network was evaluated considering the voice stream multiplexing scheme and compared to non multiplexing case. The evaluation was also carried out considering the voice traffic with highest priority and data traffic in second priority. For evaluation of all cases the mathematical models were developed.

The results showed that voice multiplexing technique is very important for performance improvement in an IP network. As the IP network is expected, in near future, to become the platform for all telephonic traffic, the voice multiplexing technique must be used for efficient utilization of available bandwidth.

REFERENCES:

- [1] Blake, S.; Black, D.; Carson, M.; Davies, E.; Wang, Z.; Weiss, W.: "An Architecture for Differentiated Services"; RFC-2475; Network Working Group; Dezembro 1998.
- [2] Douskalis, B.: "IP Telephony, the Integration of Robust VoIP Services"; Prentice Hall PTR, 2000.
- [3] Yamada, H.; Higuchi, N.: "Voice Quality Evaluation of IP-Based Voice Stream Multiplexing Schemes"; KDDI R&D Laboratories Inc.; Japan, 2001.
- [4] Park, K.; Willinger, W.; Editors: "Self-Similar Network Traffic and Performance Evaluation"; John Wiley & Sons, USA, 2000.
- [5] Karagiannis, T.; Molle, M.; Faloutsos M.; Broido, A.: "A Nonstationary Poisson View of Internet Traffic"; Proceeding of IEEE INFOCOM, 2004.