

T.D./621-3/A72

UNIVERSIDAD NACIONAL DEL CALLAO

ESCUELA DE POSGRADO

**SECCIÓN DE POSGRADO DE LA FACULTAD DE
INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**



**“UN ENFOQUE OBJETIVO DE QOS, FIN-A-FIN
PARA APLICACIONES DE VOZ (VOIP)
UTILIZANDO EL PROTOCOLO *REAL-TIME*
“RTP/RTCP”**

**TESIS PARA OPTAR EL GRADO ACADÉMICO DE:
DOCTOR EN INGENIERÍA ELÉCTRICA**

MG. MANUEL ALEJANDRO ARIAS BARANDIARAN

Callao, 2016
PERU

HOJA DE REFERENCIA DEL JURADO

MIEMBROS DEL JURADO

DOCTOR	: JUAN HERBER GRADOS GAMARRA	PRESIDENTE
DOCTOR	: FERNANDO JOSÉ OYANGUREN RAMÍREZ	SECRETARIO
DOCTOR	: CÉSAR LORENZO TORRES SIME	MIEMBRO
DOCTOR	: NOÉ MANUEL JESÚS CHÁVEZ TEMOCHE	MIEMBRO
DOCTOR	: MARCELO NEMESIO DAMAS NIÑO	ASESOR

Nº DE LIBRO : 01

FOLIO : 44

FECHA DE APROBACIÓN : julio1, 2016

RESOLUCIÓN DIRECTORAL : 041-2016-DUPG-FIEE

DEDICATORIA

A mis padres, Manuel Ramos Arias y
María Laura Barandiaran, *In Memoriam*,
por su constante presencia espiritual,
quienes por su repentina partida, no
pudieron ver mi objetivo alcanzado.

AGRADECIMIENTO

Agradezco a Dios ante todo; a las personas que han marcado mi vida de uno u otro modo, sería largo enumerar pero aquí una pequeña muestra de ellas.

Primero a mis padres quienes me dieron la vida, educación, apoyo y consejos, a mis hermanas por su apoyo emocional en las etapas de mi vida, en particular a mi hermana Dora por su apoyo constante y comprensión.

Agradezco a los Drs. Ings. Herbert Grados Gamarra, Marcelo Nemesio Damas Niño, por su apoyo constante en la materialización de la presente tesis.

En general mi agradecimiento profundo a todas las personas que tuvieron contacto conmigo y me apoyaron en todo momento para hoy culminar y ver materializada finalmente esta tesis.

INDICE GENERAL

INTRODUCCION

I. PLANTEAMIENTO DE LA INVESTIGACION	20
1.1 Identificación del Problema.....	26
1.2. Formulación de Problemas.....	29
1.3. Objetivos de la Investigación.....	31
1.4. Justificación de la Investigación.....	32

II MARCO TEORICO

2.1. Antecedentes de la Investigación.....	34
2.2. Marco Teórico.....	34
2.3 Definición de Términos Básicos.....	41
2.4. Abreviaturas Utilizadas.....	45

III VARIABLES E HIPOTESIS

3.1. Definición de la Variables en la Investigación.....	49
3.2. Operacionalización de las Variables.....	49
3.3. Hipótesis general y específica.....	50

IV METODOLOGIA

4.1. Tipo de Investigación.....	51
4.2. Diseño de la Investigación.....	51
4.2.1 Introducción.....	52
4.2.2 Motivación.....	52
4.2.3 Contribuciones.....	57
4.2.4. Estructuración de la Tesis.....	60

4.2.5. Consideraciones a respecto de QoS.....	62
4.2.6. QoS y aplicaciones.....	64
4.2.7 Tipos de QoS y sus parámetros/métricas.....	67
4.2.8 Métricas de QoS y sus límites.....	71
4.2.9. Throughput.....	71
4.2.10. Atraso (Delay).....	71
4.2.11. Composición del atraso fin a fin.....	72
4.2.12. Variación del atraso (jitter).....	74
4.2.13. Perdida de paquetes (Packet Loss).....	75
4.2.14. Eco y su efecto en la calidad de VoIP.....	76
4.2.15. Organizaciones internacionales y sus consideraciones sobre calidad de servicio.....	78
4.2.16. Parámetros Internacionales de QoS de la ITU-T, TIPHON ...	85
4.2.17. Caracterización y modelaje de los parámetros/métricas del servicio de voz.....	85
4.2.18. Los límites de los parámetros QoS.....	87
4.2.19. Cuantificación lineal de QoS.....	88
4.2.20. Cuantificación lineal de los codecs de este estudio.....	89
4.2.21. Calculo de las métricas de QoS.....	91
4.2.22. Ejemplo de cálculo de QoS.....	93
4.2.23. QoS de 3 métricas (IPDT; IPDV, IPLR).....	93
4.2.24. El protocolo RTP.....	95
4.2.25 El protocolo RTCP.....	96

4.2.26. Estadísticas del protocolo RTP/RTCP.....	96
4.2.27 Las RFCs del RTP/RTCP y sus principales especificaciones	98
4.2.28. La RFC 4885 (AVPF) y sus principales características	100
4.2.29. El algoritmo de generación del "Early Packet"	101
4.2.30. Calculo del "Maximun Dithering Interval" (T-dither-max)	102
4.2.31 Reglas para envío y recepción de paquetes RTCP (RFC .1889/3550).....	103
4.2.32. Calculo del intervalo de transmisión del RTCP (RFC1889/3550)	104
4.2.33. Inicialización de los participantes de la sesión	106
4.2.34. Recibiendo un paquete RTP o RTCP que no es de tipo "BYE"	107
4.2.35. Recibiendo un paquete RTCP de tipo "BYE".....	108
4.2.36. Time-out de SSRC	108
4.2.37. Finalización del timer de transmisión.....	109
4.2.38. Transmitiendo un paquete del tipo "Bye"	109
4.2.39. Actualización de We_sent	111
4.2.40. Asignación de Bandas en el SDES	112
4.2.41. Informes SR y RR.....	113
4.2.42. Ejemplo de Aplicación del cálculo de ancho de Banda	114
4.2.43. Formato de los paquetes RTCP	115

4.2.44. Calculo de las métricas de QoS (del RTCP)	117
4.2.45. Ejemplo de calculo del Atraso fin a fin (RTT)	118
4.2.46. Calculo del Jitter (IPDV)	120
4.2.47. Calculo de perdida de paquetes (IPLR).....	122
4.2.48. Algoritmo de monitoreo y control de QoS.....	123
4.2.49. Sistema de QoS	124
4.2.50. Módulos del sistema de QoS.....	126
4.2.51. Codecs utilizados en este estudio	130
4.2.52. Ejemplo de cálculo de la banda necesaria y mínima para el codec G.711	132
4.2.53. Algoritmo propuesto	133
4.2.54. Experiencias efectuadas	139
4.2.55. QoS por medio de simulación	139
4.2.56. Topología para Simulación.....	140
4.2.57. Descripción general de las simulaciones.....	140
4.2.58. Experiencias "reales"	141
4.2.59. Validación de la banda efectiva de la red local.....	142
4.2.60. Validación de las velocidades del enlace WAN.....	143
4.2.61. Validacion de las simulaciones.....	144
4.2.62. Flujo único en el enlace	145
4.2.63. Con flujo interferente TCP	145
4.2.64. Con flujo interferente UDP	145
4.2.65. Validación del algoritmo propuesto.....	145

4.2.66. Herramientas de prueba	146
4.3. Metodica de cada momento de la Investigaci3n.....	147
4.4. Poblaci3n y muestra	147
4.5. T3cnicas e instrumentos de recolecci3n de datos.....	147
4.6. Plan de an3lisis estadisticos de datos	148

V RESULTADOS

5.1. Resultados de las Simulaciones de voz y trafico interferente	
UDP y TCP	149
5.1.1. Flujo 3nico de voz en un enlace de 128K. (G.711)	149
5.1.2. Dos Flujos de voz (voz+tcp) en un enlace de 128K	151
5.1.3. Dos Flujos de voz (voz+udp) en un enlace de 128K	151
5.1.4. Resultados de las Simulaciones Incluyendo QoS.	157
5.1.5. Enlace de 128Kbps	157
5.1.6. Enlace de 256 Kbps.	158
5.1.7. Enlace de 512 Kbps	159
5.1.8. Enlace de 1000 Kbps.	160
5.1.9. Enlace de 1300 Kbps.	161
5.1.10. Enlace 1544 Kbps	162
5.1.11. Resumen de las metricas obtenidas por simulaci3n.....	163
5.2. Resultados de las experiencias de validaci3n	164
5.2.1. Validaci3n de la simulaci3n con flujos Interferentes	
TCP y UDP.	164
5.2.2. Flujo unico en el Enlace	164

5.2.3. Con Flujo interferente TCP	164
5.2.4. Con Flujo interferente UDP	165
5.2.5. El Procedimiento de las pruebas.....	165
5.2.6. Validación del algoritmo de restauración propuesto	174
5.2.6.1. Comportamiento de los codecs en las situaciones de congestionamiento de la red.....	174
5.2.6.2. Comportamiento de los codecs del SJphone	175
5.2.6.3. Evaluación del SJphone com Ethereal.....	179
5.2.7. Comportamiento del algoritmo de QoS	182
5.2.7.1. Validación del algoritmo	182
5.2.7.2. Ejemplo de prueba del algoritmo	184
VI DISCUSION DE LOS RESULTADOS	
6.1. Contrastación de la Hipótesis con los resultados	185
6.2. Contrastación de resultados con otros trabajos anteriores.....	189
VII CONCLUSIONES	192
VIII RECOMENDACIONES	203
IX REFERENCIAS BIBLIOGRAFICAS	205
ANEXOS	
1.- Matriz de Consistencia	220
2.- Ejemplo básico de QoS.....	221

ÍNDICE DE FIGURAS

4.1.	Esquema general de la Tesis.....	59
4.2.	Efecto de la pérdida de paquetes	76
4.3.	Calidad en función del eco.....	78
4.4.	Cabezales de los principales protocolos usados para VoIP.....	84
4.5.	Modelaje de la QoS.....	86
4.6.	Efecto simultaneo.....	86
4.7.	Los límites de las métricas de QoS.....	87
4.8.	Cuantificación lineal de la QoS	88
4.9.	Función "rampa".....	92
4.10.	Dependencia de la QoS del IPDT y TRU.....	92
4.11.	QoS en función del IPDT e IPDV	94
4.12.	QoS por la fórmula 4.2.....	94
4.13.	Calculo del IPDT	120
4.14.	Los módulos del sistema de QoS	126
4.15.	El módulo 1	127
4.16.	El módulo 2	128
4.17.	El módulo 3	129
4.18.	El módulo completo.....	131
4.19.	Topología para simulación	140
4.20.	Topología general para pruebas	142
4.21.	Topología para validación de banda configurada	143
4.22.	Topología para tráfico interferente	145
5.13.	Topologia para validación de las métricas para simulación	166
5.15.	Comunicación con codec GSM.....	175

5.16. Comunicación con Codec G711 law A.....	176
5.17. Comunicación con Codec G711 Law u.....	177
5.18. Informaciones del paquete 3505.....	177
5.19. Continuación de IP y UDP anterior	178
5.20. ETHEREAL con RTP	178
5.21. Paquete UDP visto con ETHEREAL	179
5.22. Detalles adicionales del paquete 3505.....	179
5.23. Frame 3505 e IP	180
5.24. Paquete 3505 con RTP.....	180
5.25. Paquete Siphone.....	181
5.26. Comunicación G711 law Ay RTP	181
6.1. Testbed de trabajos anteriores	190

ÍNDICE DE TABLAS

4.1.	Ejemplo de un SLS	68
4.2.	Clases de QoS del ITU-T	69
4.3.	Clases de QoS del TIPHON.....	70
4.4.	Categoría de servicio del foro ATM.....	70
4.5.	Características de los codecs para VoIP	80
4.6.	Medidas de "performance" de la ITU-T	81
4.7.	Ejemplos de requisitos de QoS.....	82
4.8.	Métricas de este trabajo.....	82
4.9.	Métricas del TIPHON	85
4.10.	ITU-T recomendación G-114	85
4.11.	Límites de los parámetros-métricas de QoS	88
4.12.	Cuantificación lineal códec G711	89
4.13.	Cuantificación lineal códec G729.....	90
4.14.	Cuantificación lineal códec GSM	90
4.15.	Cuantificación lineal códec G723.....	91
4.16.	Formato del paquete RTP.....	116
4.17.	Formato del reporte de Tx.....	116
4.18.	Formato del reporte de Rx	117
4.19.	Formato de DES	117
4.20.	Secuencia de Fallback	132
4.21.	Banda configurada y medida	144
5.1.	Flujo único de voz en enlace de 128Kbps.-Simulado.....	157

5.2.	TCP+voz en enlace de 128 Kbps-S.....	157
5.3.	UDP+voz en enlace de 128Kbps-S.....	158
5.4.	Flujo único de voz en enlace de 256 Kbps-S.....	158
5.5.	TCP+voz en enlace de 256 Kbps-S.....	158
5.6.	UDP+voz en enlace de 256 Kbps-S.....	159
5.7.	Flujo unico de voz en enlace de 512kbps-S.....	159
5.8	TCP+voz en enlace de 512 Kbps-S.....	159
5.9	UDP+voz en enlace de 512 Kbps-S.....	160
5.10.	Flujo único de voz en enlace de 1000 Kbps-S.....	160
5.11.	TCP+voz en enlace de 1000 Kbps-S.....	160
5.12.	UDP+voz en enlace de 1000 Kbps-S.....	161
5.13.	Flujo único de voz en enlace de 1300 Kbps-S.....	161
5.14.	TCP+voz en enlace de 1300 Kbps-S.....	161
5.15.	UDP+voz en enlace de 1300 Kbps-S.....	162
5.16.	Flujo único de voz en enlace de 1544 Kbps-S.....	162
5.17.	TCP+voz en enlace de 1544 Kbps-S.....	162
5.18.	UDP+voz en enlace de 1544 Kbps-S.....	166
5.19.	Flujo único de voz en enlace de 128 Kbps-Real.....	167
5.20.	TCP+voz en enlace de 128Kbps-R.....	167
5.21.	UDP+voz en enlace de 128 Kbps-R.....	168
5.22.	Flujo único de voz en enlace de 256 Kbps –R.....	168
5.23.	TCP+voz en enlace de 256 Kbps-R.....	168
5.24.	UDP+voz en enlace de 256 Kbps-R.....	168

5.25. Flujo único de voz en enlace de 512 Kbps-R.....	169
5.26. TCP+voz en enlace de 512 Kbps-R.....	169
5.27. UDP+voz en enlace de 512 Kbps-R	169
5.28. Flujo único de voz en enlace de 1000 Kbps –R	170
5.29. TCP+voz en enlace de 1000 Kbps –R	170
5.30. UDP+voz en enlace de 1000 Kbps-R	170
5.31. Flujo único de voz en enlace de 1300 Kbps-R.....	171
5.32. TCP+voz en enlace de 1300 Kbps-R.....	171
5.33. UDP+voz en enlace de 1300 Kbps –R.....	171
5.34. Flujo único de voz en enlace de 1544 Kbps-R.....	172
5.35. TCP+voz en enlace de 1544 Kbps-R.....	172
5.36. UDP+voz en enlace de 1544 Kbps-R	172
5.37. Comunicación con códec GSM.....	176
5.38. Resumen de los codecs.....	181
6.3. Características de los codecs utilizados.....	187
7.1. Ejercicio extra: resultado de "simulación".....	197
7.2. Ejercicio extra resultado de prueba "real".	200

S = SIMULADO

R = REAL

ÍNDICE DE GRÁFICOS

5.1.	Métricas QoS de voz (G711) a 128 Kbps.	150
5.2.	Métricas QoS de voz (G711) +TCP a 128 Kbps	151
5.3.	Métricas QoS de voz (G711)+UDP a 128Kbps	152
5.4.	Métricas QoS de voz (G729) a 128 Kbps	153
5.5.	Métricas QoS de voz (G729)+TCP a 128 kbps.....	153
5.6.	Métricas QoS de voz (G729)+UDP a 128 kbps	154
5.7.	Métricas QoS de voz (GSM) a 128 kbps.....	154
5.8.	Métricas QoS de Voz (GSM)+TCP a 128 kbps.....	155
5.9.	Métricas QoS de Voz (GSM)+UDP a 128 kbps	155
5.10.	Métricas QoS de Voz (G723.1A) a 128 kbps	156
5.11.	Metricas QoS de Voz (G7231.A)+TCP a 128 kbps.....	156
5.12.	Comportamiento del IPDT (simulado) para los 4 codecs.....	163
5.13.	Comportamiento del IPDT "Real" para los 4 codecs.....	173
6.1.	IPDT E2E de Voz para los 4 codecs.....	186
6.2.	IPDT E2E de voz+FTP hasta 2 Mbps	187
6.3.	IPDV E2E (voz+UDP) en función de la banda	188
6.4.	IPDV de 2 flujos (voz+FTP) en función del enlace.....	188
6.5.	IPDV de 2 flujos (voz+ UDP) en función del enlace.....	189
	Grafico A: IPDT para los 3 mecanismos de filas	222
	Grafico B: Detalle del IPDT para PQ y WFQ	222
	Grafico C: IPDV para los 3 mecanismos de filas	223
	Grafico D: Detalle de IPDV para PQ y WFQ.....	223

Grafico E: IPLR para los 3 mecanismos de filas.....	224
Grafico F: Detalle del IPLR para PQ y WFQ.....	224

RESUMEN

El *Real-Time Transport Protocol (RTP)* fue originalmente concebido para funcionar bien con grupos *multicast* en escalas muy grandes, sin embargo, hoy día, más y más aplicativos lo usan para pequeños grupos *multicast*, como videoconferencia y mismo aplicaciones *unicast* como *streaming de multimedia*. Su protocolo compañero, el *Real-Time Control Protocol (RTCP)*, es usado para monitorear los datos de *media* y proveer *feedback* sobre la calidad de la recepción. En el ámbito de los grandes grupos *multicast*, en lo relacionado a cuando enviar *feedback*, las reglas eran muy restrictivas, intentando evitar un “*explosion feedback*” causador de congestión en las redes. *Feedback* mas frecuente sería necesario para aumentar la calidad de recepción, más el **RTP** restringía mucho el uso de *feedback RTCP*. En Julio de 2006 el IETF, decidió crear un nuevo perfil **RTP extendido**”, redefiniendo algunas reglas de *timing* del **RTCP**, pero manteniendo la mayoría de los algoritmos que probaron funcionar bien. La Norma es la **RFC 4585 – Extended RTP Profile for Real-Time Transport Control Protocol (RTCP) - based feedback (RTP/AVPF)**. Las nuevas reglas tratan básicamente de dos aspectos: 1) el intervalo mínimo de 5 seg.es prácticamente abolido y 2) los receptores obtienen una vez durante el intervalo de **RTCP** la posibilidad de enviar paquetes **RTCP**, no de acuerdo con el intervalo calculado, más casi que inmediatamente. En esta pesquisa en lugar de definir más un protocolo propietario, se propone el protocolo **RTCP** de acuerdo a definición del IETF. La aplicación de la fuente

transmisora usa los datos de los informes de estadísticas recibidos para el cálculo de las métricas de **QoS**. Basado en esas métricas se determina el estado de congestión de la red y ajusta la banda de acuerdo con las velocidades, intentando mantener la **QoS** dentro de límites aceptables. Justamente en esta línea, esta investigación también propone un algoritmo que ajusta los requerimientos de banda de las aplicaciones de acuerdo con el estado de carga de la red, partiendo de parámetros puramente objetivos.

RESUMO

O Real-time Transport Protocol (RTP) foi originalmente concebido para funcionar bem com grupos multicast em escalas muito grandes, no entanto hoje mais e mais aplicativos usam-no para pequenos grupos multicast, como Videoconferencia ou mesmo aplicacoes unicast como streaming de multimedia. Seu protocolo companheiro, o Real-Time Control Protocol (RTCP) é usado para monitorar os dados de midia e fornecer feedback sobre a qualidade da recepcao. No âmbito de los grandes grupos multicast, no relacionado a quando enviar feedback, as regras eran muito restritivas tentando evitar uma "Explosion feedback" causadora de congestionamiento nas redes. Mais feedback seria necessario para aumentar a qualidade de recepcao, mais o RTP restringia muito o uso de feedback RTCP. Em julho de 2006, o IETF, decidiu criar um novo perfil "RTP estendido" redefinindo algumas regras de timing do RTCP, mas mantendo a maioria dos algoritmos que provaram funcionar bem. A norma e a RFC 4585- "Extended RTP Profile for Real Time Transport Protocol (RTCP) – based Feedback (RTP/AVPF)": As novas regras tratam basicamente de dois aspectos: 1) O intervalo mínimo de 5 seg .e prácticamente abolido e 2) os receptores obtén uma vez durante o intervalo de RTCP, a possibilidade de enviar pacotes RTCP, nao de acordo con o intervalo calculado, mais quase que imediatamente. Nesta pesquisa, usase o protocolo RTCP de acordo a nova definição do IETF.

A aplicacao da fonte transmisora usa os dados dos relatorios de estatisticas recibidos, para o calculo das métricas de QoS .dentro de límites aceitaves. Justamente nesta linha, esta pesquisa propoe um algoritmo que ajusta os requerimentos de banda das aplicacoes de acordó com o estado de carga da rede, partindo de parámetros puramente objetivos.

ABSTRACT

The Real-Time Transport Protocol (**RTP**) was originally designed to work well for multicast groups in very large scales, however today, more and more applications uses it for small multicast groups like videoconference or even unicast applications like streaming multimedia. Its companion protocol, called Real-Time Control Protocol (**RTCP**), is used to monitor the media data and provide feedback of the reception quality. Considering the scope of large multicast groups in mind, the rules when to send feedback were much restricted to avoid feedback explosion originating congestion in the networks. As said before, more feedback would be needed to increase the reception quality, but the initial **RTP** restricted the use of **RTCP** feedback very much. In July of 2006 it was decided by the **IETF**, to create a new "extended **RTP**" profile, which redefines some of the **RTCP** timing rules, but keeps most of the algorithms for **RTP** and **RTCP**, which have proven to work well. The standard is the **RFC 4585 - Extended RTP Profile for Real-Time Transport Control Protocol (RTCP) - Based Feedback (RTP/AVPF)**. The timing rules of this new **RFC** consists basically of two components: 1) the minimum interval of 5 sec. is abolished and 2) receivers get once during **RTCP** interval the chance to send and **RTCP** packet "early", not according to the calculated interval, but almost immediately. Instead of defining one more proprietary protocol, is used the **RTCP** as defined by the **IETF**. The sending application uses receiver reports to compute the main parameters of the metrics of **QoS** and based on these metrics the network congestion

state is determined and the bandwidth adjusted. In resume, this document propose a control scheme that adjusts the bandwidth requeriments of the end applications according to the network load using a new algorithm for monitoring, maintain and recover the **QoS** in voice applications starting from objective parameters.

CAPÍTULO I

1. PLANTEAMIENTO DE LA INVESTIGACIÓN

El mayor problema de la utilización del protocolo UDP/IP como mecanismo de transporte para aplicaciones en tiempo real (real-time applications) es la falta de garantía de entrega de los paquetes a sus destinos, además que los paquetes dañados o retrasados con respecto al momento de su reproducción también son descartados. La alternativa de utilizar TCP/IP no es muy eficaz, debido a su gran "overhead" de encabezado ya que las múltiples solicitudes de retransmisión la convierten en no apta para aplicaciones en tiempo real debido a la generación de tráfico, a menudo inútil, si se considera que los paquetes retrasados pueden llegar demasiado tarde para su correcta reproducción. Con el objetivo de hacer frente a estas limitaciones, proporcionar una herramienta útil para aplicaciones multimedia y múltiples tentativas de solución, fue diseñado por el IETF (Internet Engineering Task Force), órgano de Ingeniería de la INTERNET, el protocolo "RTP/RTCP".

Aunque previsto para uso exclusivo con UDP/IP, también se utiliza con otros protocolos, como TCP/IP y AAL5/IP¹. El RTP, como es definido por el IETF, consta de dos partes, el RTP propiamente dicho, utilizado para la transmisión de datos, y su correspondiente protocolo de control RTCP, para envío de estadísticas periódicas de los receptores para sus fuentes (Tx) de

envío de estadísticas periódicas de los receptores para sus fuentes (Tx) de datos. Entre otros recursos, el RTP proporciona: 1) Identificación del tipo de información; 2) Identificación de la fuente; 3) Identificación de la secuencia de los paquetes; 4) Timestamping.

El protocolo RTCP, proporciona informaciones periódicas sobre la calidad de los datos transmitidos/recibidos y estadísticas de los participantes de las sesiones. El lado "receptor" de la aplicación que está siendo utilizada, envía periódicamente informes de recepción (RR) para su fuente de datos, estos informes incluyen datos de los parámetros que tienen que ver con las métricas de QoS, entre otros: pérdida de paquetes, retraso, variación de retraso y jitter. De forma general, el RTCP permite implementar cuatro funciones básicas:

- Mejora de la calidad de la comunicación, utilizando las estadísticas generadas periódicamente.
- Mantener un registro de todos los participantes de la sesión a través del campo CNAME (Canonical Name).
- En el caso de conferencia todos los participantes envían paquetes RTCP a todos los demás participantes.
- Transporta alguna información de control de la sesión.

Cada paquete RTCP comienza con un cabezal fijo de 8 bytes seguido de elementos estructurados que pueden ser de longitud variable, de modo que varios paquetes RTCP pueden ser concatenados sin la necesidad de un separador, formando un paquete compuesto que luego se encapsula en un

paquete UDP/IP.

Las RFCs del RTP y sus principales especificaciones

El IETF es el organismo que regula y norma el área de la INTERNET. Sus normas o recomendaciones equivalentes a las normas de la UIT para telecomunicaciones son llamadas de RFCs (Request for Comments). Estas RFCs aplicables al RTP/RTCP y consultadas para el tema de esta investigación fueron:

- RFC 1889, Enero de 1996, "RTP: A transport protocol for real-time applications"
- RFC 1890, Enero de 1996 "RTP: Profile for audio and videoconferences with minimal control"
- RFC 3550, Julio de 2003, "RTP: A transport protocol for real-time applications".
- RFC 3551, Julio de 2003, "RTP: Profile for audio and videoconferences with minimal control"
- RFC 3611, noviembre de 2003, "RTP: Control protocol extended reports (RTCP-XR)".
- RFC 4585, Julio de 2006, "Extended RTP profile for RTCP- based feedback (RTP/AVPF [Audio-Visual Profile with Feedback])".
- RFC 4586, Julio de 2006, "Real-time transport control protocol (RTCP) based feedback: results of the timing rule simulations".

Antes de la creación de la RFC 4585, las principales normas del RTP restringían el envío de información estadística, como se indica a

continuación:

- Los mensajes de las estaciones receptoras se enviaban en paquetes RTCP tipo híbridos que incluían al menos un SR (Sender Report) o RR y un mensaje de SDES (Source Description).
- Estos paquetes RTCP eran enviados en intervalos de tiempo (T-rr) calculados en función del tamaño medio del paquete, número de transmisores y receptores en el grupo y del ancho de banda de la sesión (un porcentaje del ancho de banda de la sesión es asignada para los paquetes RTCP). Esta banda es dividida entre todos los miembros de la sesión y los transmisores pueden conseguir más banda que los receptores.
- El intervalo mínimo entre los 2 paquetes RTCP consecutivos de la misma fuente era de 5 segundos.

El objetivo de estas reglas era evitar una “explosión de realimentación” de estadísticas y escalar a grupos multicast muy grandes. Sin embargo, se observa que principalmente la regla del intervalo mínimo (5 seg.) puede impedir a los receptores de enviar realimentación en tiempo hábil, es decir enviar fuera del instante en que era realmente necesario tomar una actitud correctiva (era necesario esperar 5 segundos).

La RFC 4585, de julio de 2006, introduce cambios en el perfil RTP para audio y vídeo. Los principales fueron los siguientes:

- Fue introducido el concepto de “**early RTCP message**” que, actuando en conjunto con algoritmos adecuados, propicia realimentación de

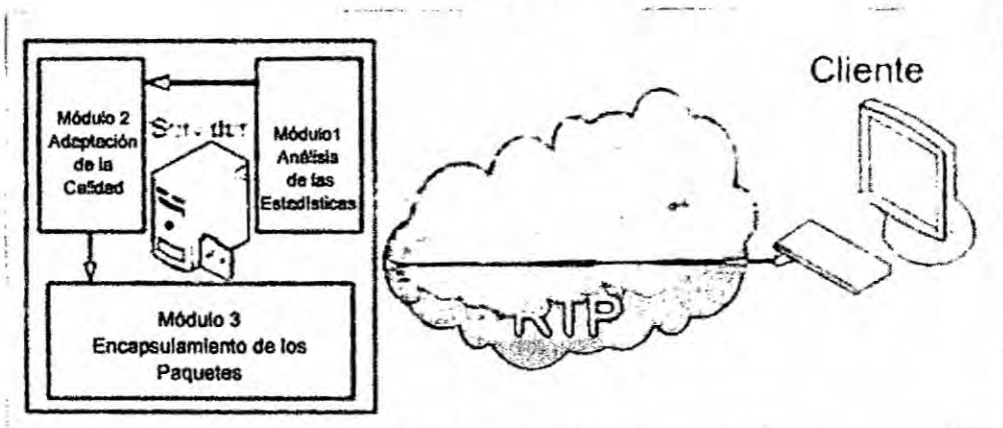
estadísticas con bajos retrasos.

- Introducción de pequeños mensajes de realimentación que serían transmitidos en los "payloads" del RTCP.

En realidad, esta RFC visa resolver las restricciones principalmente de "timing" introducidas en las RFCs 3550/3551, pero manteniendo la mayoría de los algoritmos del RTP y RTCP, originales, que habían probado trabajar adecuadamente en la Internet. Los detalles pueden ser encontrados en la propia RFC 4585.

FIGURA N°1

ESQUEMA GENERAL DE LA TESIS



FUENTE PROPIA DEL AUTOR

La INTERNET (Red mundial de computadores) inició sus actividades utilizando básicamente aplicaciones TCP altamente confiables y hoy conocidas como: Aplicaciones Standard que la convirtieron rápidamente en un grande suceso. Estas aplicaciones usan TCP como protocolo de transporte, no tienen problemas de timing o de ancho de banda ni tampoco exigencias de atraso, perdida o secuencia de paquetes. Si un paquete se

pierde o atrasa, esto puede ser resuelto, por ejemplo, aplicando un mecanismo de retransmisión de paquetes.

La llegada de las tecnologías de banda ancha llevó a aumentos drámaticos, la capacidad de las redes conmutadas de paquetes desde algunos megabits por segundo a miles de megabits por segundo. Esta capacidad de comunicación de datos era necesaria para las nuevas aplicaciones tales como videoconferencia, voz y telefonía vía Internet conocidas genéricamente como Aplicaciones Multimedia.

Esas aplicaciones imponen requisitos muy estrictos a los parámetros que tienen que ver con las métricas de Calidad de Servicio (QoS) como atraso, variación, de atraso, pérdida de paquetes, ancho de banda, etc.

Algunas requieren rígidos valores de atraso fin a fin y otras requieren un mínimo de ancho de banda, velocidad etc. Algunas requieren baja pérdida de paquetes, otras pueden requerir alto "throughput". Visando la satisfacción de esos requerimientos los Investigadores mejoran la Internet actual principalmente en su sistema de QoS y un grande número de protocolos tratan los problemas con él relacionados; sin embargo, muchos de estos problemas son complicados y difíciles de implementar, algunos son de grande costo de recursos y otros no atienden a los objetivos de garantía de QoS. Da la impresión que la Internet llegó a un impase.

Así, puede observarse que la relevancia de este problema es grande lo suficiente que merece una solución urgente e inmediata y en la cual este trabajo pretende colocar "su granito de arena".

***Capa de Adaptación ATM 5 (AAL5)**, es una capa de adaptación ATM utilizada para enviar paquetes de longitud variable de hasta 65.535 octetos de tamaño a través de un modo de transferencia asíncrono de la red (ATM).

A diferencia de la mayoría de las tramas de la red, que la información de control es la cabecera, lugares AAL5 información de control en un octeto remolque al final del paquete. El remolque AAL5 contiene un largo de longitud de 16 bits, un campo de 32 bits de comprobación de redundancia cíclica (CRC) y dos campos de 8 bits etiquetados *UU* y *CPI* que están actualmente sin usar.

Cada paquete AAL5 se divide en un número entero de células ATM y vuelto a montar en un paquete antes de la entrega a la recepción de acogida.

Este proceso se conoce como segmentación y re ensamblado (ver abajo).

La última celda contiene relleno para asegurar que todo el paquete es un múltiplo de 48 octetos de largo. La célula final contiene hasta 40 octetos de datos, seguido de bytes de relleno y el "trailer" 8 octetos.

En otras palabras, AAL5 coloca el remolque en los últimos 8 octetos del final de las células donde se puede encontrar sin saber la longitud del paquete; el final de células se identifica mediante un bit en la cabecera ATM, y el trailer está siempre en los últimos 8 octetos de esa célula.

1.2¹ Identificación del problema

El inicio de la INTERNET se hizo con aplicaciones TCP, muy diferente de las actuales aplicaciones UDP que prácticamente acapararon el contexto

de las comunicaciones con computadores y que se las conoce hoy, genéricamente como "Aplicaciones Multimedia" o también "aplicaciones que utilizan video y audio". Recurrente de esta situación, estas aplicaciones altamente sensibles a "timing" originaron la búsqueda de soluciones para los problemas resultantes de su uso, identificando como efectos negativos en las redes de conmutación de paquetes y circuitos, el atraso de las señales transportadas, la variación del atraso, el ancho de banda, la pérdida de paquetes, eco, etc, estos problemas, que en las redes de conmutación de circuitos ya han sido cuantificados y siendo este tipo de red, la referencia de calidad para los señales de audio y voz, la cual se pretende alcanzar también en la conmutación de paquetes, es imprescindible y urgente la identificación plena de estas características. La busca de valores de referencia para esos parámetros/métricas llevó inclusive a la revisión de los métodos de medida de la calidad de voz utilizada. Se ha mejorado mucho, pero aún no existe algo definitivo. Debido entre otros, a los costos de la forma (subjetiva) más utilizada actualmente, y al carácter subjetivo y dependiente de cada persona "evaluadora", otras formas están siendo evaluadas, principalmente modelos "OBJETIVOS". En este camino se creó inclusive, por el IETF, el protocolo, Real-Time Transport Protocol (RTP) que fue originalmente concebido para funcionar bien con grupos multicast en escalas muy grandes, sin embargo, hoy en día, más y más aplicativos lo usan para pequeños grupos multicast, como videoconferencia y mismo aplicaciones unicast como streaming de

multimedia. Su protocolo compañero, el Real-Time Control Protocol (RTCP), es usado para monitorear los datos de la aplicación y proveer feedback sobre la calidad de la recepción. En el ámbito de los grandes grupos multicast, en lo relacionado a “cuando” enviar feedback, las reglas eran muy restrictivas, intentando evitar la generación de más una anomalía, la conocida “explosion feedback” causante de congestionamiento, en las redes. Es necesario aumentar la frecuencia del feedback de la recepción, para evitar periodos largos de silencio sobre la calidad en curso en las redes. En Julio del 2006 el IETF, creó un nuevo perfil el RTP extendido”, redefiniendo algunas reglas de timing del RTCP, pero manteniendo la mayoría de los algoritmos que probaran funcionar bien hasta ese momento. La norma es la RFC 4585 – Extended RTP Profile for Real-Time Transport Control Protocol (RTCP) – base feedback (RTP/AVPF). Las nuevas tratan básicamente de dos aspectos:

1) el intervalo mínimo de 5 segundos es abolido. 2) los receptores obtienen una única vez durante el intervalo de RTCP la posibilidad de enviar paquetes RTCP no de acuerdo con el intervalo calculado, mas casi inmediatamente.

Como ejemplo puede observarse en la tabla abajo la pérdida en paquetes y bps en 5 seg. (1% de pérdida).

Velocidad del enlace bps.	Paquetes en 5 seg. paquetes/seg.	Bits perdidos en 5 seg.	Paquetes perdidos en 5 seg.
82.800 Kbps.	G711=50 paquetes/seg	2680 bits	5 paquetes
	IPLR(%)=1/50 (100)= 2 %		

En este trabajo de investigación, en lugar de usar mas un protocolo propietario, se usa el protocolo RTCP de acuerdo a la definicion del IETF, pero incluyendo los items de interes de su evolucion y mejoras.

La aplicación de la fuente transmisora usa datos de los informes de estadísticas recibidas para el cálculo de las métricas de QoS. Basado en estas métricas se determina el estado de congestión de la red y ajusta la banda de acuerdo con la velocidades en curso, intentando mantener la QoS dentro de los límites aceptables. Justamente en esta línea, se propone un algoritmo que ajusta los requerimientos de banda de las aplicaciones de acuerdo con el estado de carga de la red.

1.3 Formulación de problemas

En este ítem, se pretende de forma general, atender a los procedimientos recomendados en la mayoría de documentos sobre tesis de grado que fueron revisados. Específicamente se pretende responder a la pregunta: **¿Qué se investigará?**, a través de los ítems 1) la formulación del problema, 2) definición, 3) Delimitación, 4) Objetivos y 5) otros.

Con referencia al título del tema, podemos decir que:

“Un enfoque objetivo de QoS”, se refiere a la metodología de evaluación de la calidad de las aplicaciones a las que se destina, el caso “Objetivo”, significa diferente de la forma tradicional “Subjetiva” que la evaluación de QoS a ser seguida será basada en parámetros puramente de redes, a diferencia de la forma “subjetiva” tradicional; **fin a fin**, quiere decir que las mediciones serán efectuadas en las puntas de los

equipamientos terminales. **“Para aplicaciones de voz”** indica que este trabajo se destina a las aplicaciones de “voz” solamente.

“Utilizando el protocolo de tiempo real RTP/RTCP”, indica que se seguirá el procedimiento de operación de la norma del IETF de ese protocolo, para analizar el comportamiento de sus métricas, (a través de uso de las estadísticas generadas por el protocolo RTP/RTCP desde un único punto) y utilizará métricas puramente “objetivas”. Estos parámetros son: 1)retraso fin a fin, 2) jitter, 3)perdida de paquetes y 4) ancho de banda que influyen en mayor o menor grado a las aplicaciones que trafican en las redes de paquetes, particularmente a las que tienen que ver con audio y video llamadas comúnmente de Aplicaciones Multimedia. Disponer de monitoreo y control de esos parámetros es lo que se conoce como QoS (Calidad de Servicio).

Resumiendo, la propuesta aquí es un procedimiento para implementar QoS, utilizando el protocolo RTCP incluyendo adicionalmente la respuesta a las siguientes preguntas: ¿no se vuelve más vulnerable al sistema?, ¿Cómo funciona la información cuando se usa el protocolo RTCP?, ¿Cómo se implementa?, ¿Cuáles son los efectos del RTCP sobre los parámetros de QoS?, incluyendo pruebas de evaluación de su efectividad. El grande atractivo es la disponibilidad en una única fuente de las métricas de QoS y consecuentemente, un costo menor.

1.3. Objetivos de la Investigación

Existen varios objetivos, entre ellos: el de ofrecer un mecanismo simple y fácil de implementar que sustituya al método "Subjetivo" de evaluación de la calidad de la voz en las redes de comunicación por paquetes. Al mismo tiempo, ayudar al entendimiento de las causas y efectos limitantes y restrictivos en las redes tipo IP o lo mínimo colaborar en la determinación de una metodología de trabajo en esta área.

1.3.1 Objetivo general

Proponer una metodología para monitoreo y control de calidad de las Aplicaciones Multimedia iniciando con el servicio de voz digitalizado, verificando su comportamiento dentro de los valores límites de QoS utilizados por las principales administraciones de Telecomunicaciones mundiales. Si se conseguir resultados positivos., esta aplicación, podría extenderse también a otras Aplicaciones Multimedia.

1.3.2 Objetivos específicos

- Experimentar los efectos de los flujos TCP con el tráfico simultáneo de UDP de voz y viceversa, su efecto en el ancho de banda y el comportamiento de las métricas de QoS en enlaces Wan de baja y media velocidad (acceso).
- También esta experiencia pretende cuantificar los canales de voz por tipo de codec para un determinado ancho de banda vía simulación.
- Contrastar los valores del ítem anterior con experiencias similares con ruteadores reales.

- Proponer un método alternativo de implementación de QoS de forma objetiva para los circuitos de acceso.
- Adquirir conocimientos detallados de las redes IP, y de las Aplicaciones Multimedia verificando la ganancia tecnológica y de costo al usar las métricas desde un punto de fuente única y analizar el grado de aproximación de los valores obtenidos cuando comparados con el método actual subjetivo de la formula empírica de cálculo de la QoS indicada en el Cap. IV y de las referencias máximas de las métricas.

1.4 Justificación de la investigación

El estudio se viabiliza y justifica por si solo en la medida que un mejor entendimiento de los conceptos de las redes IP pueden representar enormes beneficios y facilidades en la implementación de la calidad de servicio necesaria en las redes de conmutación de paquetes y en la Internet en particular; además de las economías resultantes de su implementación.

- Intentando responder a las preguntas clásicas de este ítem, ¿Para qué se investiga?, ¿Qué importancia tiene esta investigación?, etc, a la primera pregunta, se puede responder, que si consideramos que hoy día el referencial de calidad de la voz es el ofrecido por las redes de conmutación de circuitos, y que este tipo de redes están prácticamente acabando, se hace muy necesario, mecanismos que mantengan calidad y sustentabilidad en las comunicaciones no solamente cuando los sistemas están libres u ociosos sino también en las situaciones de congestionamiento. Al mismo tiempo su importancia es vital en la transmisión a largas distancias

incluyendo servicios integrados y compartidos por ejemplo en las redes convergentes (voz, video, datos), además de la economía de esta integración.

- Esta investigación se justifica también en la medida que su divulgación permitirá expandir en nuestro medio, el conocimiento y la problemática de la *Internet* actual e sus desafíos en la gestión de redes.

CAPÍTULO II

2. MARCO TEÓRICO

2.1 Antecedentes de la investigación

Grande cantidad de investigaciones están en curso, procurando hacer las redes de conmutación de paquetes, más confiables principalmente para comunicación con aplicaciones Multimedia. Como las aplicaciones de voz (VoIP) son altamente sensibles al retraso, una buena ingeniería de red, fin-a-fin, es necesaria para tornar la VoIP un suceso. Hasta hace poco tiempo el principal requerimiento de las aplicaciones tradicionales tales como FTP y Telnet era la correcta entrega de los paquetes. Debido a que el protocolo TCP fue proyectado como confiable, retransmite los paquetes que no alcanzaron su destino o llegaron muy tarde, garantizando así la entrega de los paquetes.

Las técnicas convencionales para reducir la pérdida de paquetes como retransmisión y FEC (*Forward Error Correction*) podrían ser usadas. Sin embargo, la retransmisión consume tiempo lo que las hace inadecuadas para aplicaciones *real time* y FEC está proyectada para mejorar los efectos de la pérdida de paquetes introduciendo datos redundantes. Otras técnicas como enrutamiento en diversidad y *adaptive rate*, tampoco son suficiente eficientes para este tipo de aplicaciones.

2.2. Marco teórico

La transmisión de voz sobre redes IP (VoIP) tiene un rol cada vez más importante y un uso cada vez más difundido. Los usuarios esperan ver

satisfechas sus expectativas de calidad del servicio con independencia de la tecnología utilizada. En este sentido, la "calidad de experiencia" QoE mide, "cuan bueno" un servicio de red satisface las expectativas y necesidades vistas por el usuario. Por otro lado, la "calidad de servicio" QoS se refiere a la medida del rendimiento de la red desde el punto de vista técnico, y a la posibilidad de gestionarla para cumplir con las prestaciones necesarias para las aplicaciones.

La VoIP enfrenta problemáticas propias de las redes de datos, que se manifiestan como degradaciones en la calidad del servicio percibida por los usuarios. Estas degradaciones pueden deberse por ejemplo a retardos, *jitter* y pérdida de paquetes, entre otros factores. Para que la tecnología de VoIP pueda ser utilizada en forma masiva y comercial, es esencial garantizar una calidad de voz aceptable. Para ello se han desarrollado métodos para evaluarla. Estos métodos se dividen en dos grupos: **subjetivos** y **objetivos**. Los métodos **subjetivos**, se basan en conocer directamente la opinión de los usuarios. Típicamente resultan en un promedio de opiniones (por ejemplo, en un valor de MOS – Mean Opinión Score - con valores de 1 a 5). Los métodos **objetivos** miden propiedades físicas de una red para prever o estimar el rendimiento percibido por los usuarios. A su vez, estos últimos se subdividen en **Intrusivos** (se inyecta una señal de voz conocida en el canal y se estudia su degradación a la salida, (por ejemplo PESQ) y **No Intrusivos** (monitorean ciertos parámetros en un punto de la red y en base a estos permite establecer en

tiempo real la calidad que percibiría un usuario).

Entre otros motivos, por el costo y la dependencia de los aspectos subjetivos de las personas en medidas del tipo subjetivo, (método más usado) actualmente, se está buscando métodos confiables del tipo objetivo que atiendan plenamente para este fin y dejar de depender, entre otras cosas, de la subjetividad de cada persona.

Algunas consideraciones sobre redes

A seguir algunas consideraciones sobre redes de paquetes y los protocolos de transporte TCP y UDP. Por ejemplo el control de flujo no es adecuado para media continua pues datos perdidos pueden ser tolerados pero retransmisiones o atrasos extras no son tolerados, y cuando un segmento no llega a su destino, el control de congestión asume "congestión" en lugar de errores. Protocolos nuevos tales como RTP [7] superan la mayoría de las limitaciones arriba mencionadas mediante el uso de APPLICATION LEVEL FRAMING (ALF), la integración de su procesamiento con el procesamiento propio de la aplicación (Integrated layer processing- ILP) [5] y el uso de perfiles para los algoritmos de control que son más adecuados para los requerimientos de las aplicaciones. Además RTP tiene un protocolo de control asociado, el RTCP, para evaluar las condiciones de la red, pero no impone ningún algoritmo predefinido para trabajar en el control de los datos; en este documento, no es tan importante destacar el caso concreto de nuestra elección del RTP pero si considerar esas características abstractas de los nuevos protocolos. Una forma natural de

usar redes ATM con un servicio "best-effort" es escoger la clase de servicio ABR (Available Bit Rate).

Sin embargo hay algunas razones porque esto no es adecuado para multimedia: El ABR fue proyectado para atender a aplicaciones de datos que tienen condiciones de controlar ancho de banda de reserva. Ellos pueden adaptarse a las condiciones variantes de la red pero es importante que la pérdida de células sea tan baja cuanto posible (Aplicaciones de datos son bastante sensibles a pérdida de células).

Esto no es un requisito para las aplicaciones multimedia. ABR tiene su propio ciclo de control (para prevenir pérdidas) el cual es basado en conceptos abstractos de multimedia, tales como células (el mismo problema indicado arriba para los protocolos de transport tradicionales).

El camino más sensato es entonces usar la clase de servicio Variable Bit Rate (VBR), y usar prioridades para trabajar con valores sobre contrato. La aplicación fija un límite inferior de ancho de banda para la conexión, bajo del cual no sirve para mantener la interacción. Después intenta elevar la calidad usando CLP=1 células, tanto cuanto la red pueda manejar.

Se asume que los mecanismos UPC/NPC usa "Cell tagging" hasta que el valor del Peak cell rate (PCR) es alcanzado y descarta células arriba de esta tasa. Por consiguiente una conexión de video, por ejemplo debería fijar el PCR de acuerdo a la longitud del frame, pero puede fijar el Cell rate sostenible (SCR) a un valor menor que el valor real que el pretende usar (Apenas arriba del mínimo nivel de QoS, por ejemplo).

Trabajos relacionados:

Solamente para dar una idea del cuadro de actividades en esta área de Calidad de Servicios, a seguir, incluimos la relación de algunos trabajos completos y otros en proceso. En [105] el autor proporciona un survey sobre sistemas multimedia distribuidos. Se unifica una propuesta [99] para manipular un programa de niveles de QoS basado en consulta de calidad por ejemplo para hacer que el usuario escoja la calidad del servicio sin importarle los niveles de los parámetros de ajuste dinámico del QoS" descrito en [91], tiene fuertes similitudes con este sistema aquí presentado pero ninguna arquitectura de gerenciamiento de QoS es propuesta. (El algoritmo es parte de la aplicación). La aplicación transmisora usa los paquetes del reporte RTCP para calcular las pérdidas de paquetes. Basada en esta medida y medido el ancho de banda disponible es determinado y el QoS es ajustado para ese valor.

El algoritmo es igual a uno propuesto en [103] aunque el nuestro no solo calcula la pérdida de paquetes sino también el atraso de los paquetes con el objetivo de medir el ancho de banda disponible. Esto significa que no siempre consideramos las condiciones de la red como [91] lo hace sino también las condiciones de carga de las maquinas finales [93] propone una architecture abstracta par la manipulación de QoS, llamada arquitectura de QoS (QoS-A). [94] propone una capa que es parte de su arquitectura, llamada Multimedia enhanced transport system (METS), para gerenciar QoS para la aplicación.

Este enfoque es similar al nuestro aunque nuestra arquitectura es más compleja y puede manipular completamente datos multimedia para la aplicación y no apenas ajustar QoS [94]. También propone un algoritmo similar al nuestro pero que no usa RTP/RTCP: Un esquema específico para controlar congestión de red basado en bit and packet rate scaling, esto es, QoS degradation, en un sistema sin reserva de ancho de banda es tratado en [104]. La llave del problema tratado es encontrar valores sustentables aceptables para el QoS durante la congestión. El algoritmo es similar al estudiado por nosotros aunque la situación de los hosts terminales no es considerado. Tampoco es propuesta una arquitectura.

En [101] una arquitectura flexible para QoS es propuesta. El sistema toma en cuenta, conceptos de la aplicación, red y sistema operacional y un Broker de QoS, calcula su mejor balance para proporcionar el QoS deseado. Esta es principalmente una solución de configuración que un control de sistema on-going. Y requiere mucho del ambiente (real time OS, scheduler policies, etc).

Existe también un mapeamiento de los parámetros de QoS de la aplicación a los de la red, pero tiene pequeño cambio de significado entre los 2.

El asunto de QoS también ha sido un tópico de estandarización. La ISO Standard [99], define un framework; describe como la QoS puede ser caracterizada y sus restricciones especificadas. Un juego de mecanismos para gerenciamiento de QoS son identificados. Una lista de parámetros de QoS es también dado. Nuestro algoritmo puede ser pensado como una

implementación del mecanismo genérico que ellos proponen para control de QoS. Ninguna arquitectura es propuesta, al control de QoS de este artículo, es receptor- dependiente QoS para multicast. El problema tratado en [97] indica como liberar diferentes niveles de QoS a receptores diferentes usando multicast. Nuestro algoritmo no puede ser usado en este caso porque es el transmisor que ajusta la velocidad de envío para todo el sistema.

Para multicast el QoS tendría que ser ajustado por el receptor en la peor situación (por ejemplo conectado al tx por el enlace de menor ancho de banda de la red no es aceptable. [96] propone una solución basada en filtros que ajustan el QoS de la media que está siendo enviada a cada receptor o grupo de receptores participantes. Otro aspecto relacionado con media específica, es el desafío de maximizar la QoS de un stream MPEG. Las propuestas presentadas en [98] y [102] hacen esa tarea programando los frames MPEG de acuerdo con su relevancia para el receptor (alta prioridad a frames I y baja prioridad para los tipos B).

El usuario tiene una baja percepción cuando un frame de baja prioridad es descartado por eso son descartados primero, cuando alguna acción debe ser tomada. Nosotros no consideramos eso, mismo para MPEG, pero algunas ideas de esos documentos pueden ser extendidas a nuestros algoritmos para esos formatos de media.

2.3 Definición de Términos

AS: Autonomus System, áreas que en su conjunto modelan una red y

dentro de las cuales las rutas son determinadas por el ruteo intradominio.

ASN1: Abstract Syntax Notation 1, lenguaje de definición de objetos estándar usados por SNMP.

BGP: Border Gateway Protocol. Protocolo de ruteo intradominio.

CBR: Constraint Based Routing, ruteo que intenta encontrar un camino que optimice cierta métrica escalar y al mismo tiempo no viole un conjunto de restricciones.

CLNS: OSI Vonnexion Less Network Srvce, protocolo bajo el cual opera SNMPv1.

CR-LDP: Constraint Route LDP, protocolo de distribución de etiquetas del tipo enrutamiento explicito que ofrece características de Ingeniería de Trafico.

CSPF: Constraint Shorset Path First, algoritmo para la computación de caminos que toma en cuenta al mismo tiempo un conjunto de restricciones.

DDP: Apple Talk Datagram-Delivery Protocol, protocolo bajo el cual opera SNMP V1.

DLCI: Data Link Connection Identificator, ejemplo de etiquetas o encabezado que pueden utilizarse como etiqueta de MPLS.

FEC: Forwarding Equivalence Class, representación de un conjunto de paquetes que comparten los mismos requerimientos para su transporte en MPLS.

IETF: Internet Engineering Task Force, grupo de trabajo dedicado en su mayoría al control del tráfico en lo que la Ingeniería de Tráfico se refiere.

IPX: Novel Internet Packet Exchange, protocolo bajo el cual opera SNMPv1.

IS-IS: Intermediate System to Intermediate system, protocolo de ruteo intradominio.

ISP: Internet Service Provider, proveedor de acceso a Internet.

LDP: Label Distribution Protocol, protocolo responsable de que el LSP sea establecido para que sea funcional mediante el intercambio de etiquetas entre los nodos de la red.

LER: Label Edge Router, router encargado de la distribución de etiquetas.

LIB: Label Information Base, tabla de conectividad con la cual es examinada y comparada la etiqueta MPLS al llegar del LER al LSR, determinando la acción a ser seguida.

LSP: Label Switched Paths, ruta que sigue un paquete entre dos nodos de la red MPLS.

LSR: Label Switch router, router encargado de dirigir el tráfico dentro de la red MPLS.

MIB: Management Information Base, colección de información organizada jerárquicamente donde los objetos son accedidos usando SNMP y la cual reside en el elemento de red.

MIRA: Minimum Interface Routing Algorithm, algoritmo de ruteo de caminos que intenta minimizar la "interferencia" que provoca el establecimiento

de un nuevo camino a potenciales nuevos caminos que son desconocidos.

MMF: Max-Min Fairness, principio de asignación usado para formular el esquema de utilización de recursos en donde se intenta asignar la mayor cantidad de recursos a cada demanda, al mismo tiempo que se intenta mantenerlos lo más similares posible.

MNF: Maximum Network Flow, máximo ancho de banda que puede traficar la red entre determinado par de nodos ya sea por un único camino o varios.

MPLS: Multi Protocol Label Switching, tecnología de ruteo y reenvío de paquetes en redes IP que se basa en la asignación e intercambio de etiquetas, que permiten el establecimiento de caminos a través de la red.

Network Delay: Es el tiempo en el cual, el nodo transmisor

NRT-TE: Networking Traffic Engineering, nombre del software diseñado en este proyecto que hace alusión a la Ingeniería de Tráfico en redes; tema principal de este trabajo.

NMS: Network Management System, estación administradora o, lo que es similar elemento de red que contiene un agente SNMP y pertenece a la red administrada.

QoS: Quality of Service, distintos niveles de servicio que son ofrecidos por la red al cliente.

RSVP-TE: Reservation Protocol with Traffic Engineering, protocolo de

enrutamiento explícito. En este caso en particular es una extensión de la versión original RSVP que incorpora el respaldo para MPLS.

SDP: Shortest Distance Path, ruteo basado en preservar los recursos de la red por medio de la selección de los caminos más cortos.

SLA: Service Level Agreement, acuerdo sobre el nivel de servicio con el cliente donde se especifican parámetros como performance, confiabilidad y seguridad.

SMI: Structure of Management Información, partes del ASN 1 que usa

SNMP. Es lo que en realidad describe la estructura de datos SNMP.

SNMP: Simple Network Management Protocol, protocolo de la capa de aplicación que facilita el intercambio de información de gestión entre los elementos de la red y es parte del stack de protocolos TCP/IP.

SWP: Shortest Widest Path, ruteo que se basa en la búsqueda del camino con el ancho de banda más grande y, en caso de haber múltiples caminos, se queda con el que tiene la mínima cantidad de saltos.

TE: Traffic Engineering, disciplina que procura la optimización de la performance de las redes operativas.

TED: Traffic Engineering Especialized Data Base, base de datos contenida en cada ruteador, la cual mantiene atributos de los enlaces de la red e información de la topología.

UDP: User Datagram Protocol, protocolo de transporte que provee servicios de datagramas por encima de IP.

VPI/VCI: Virtual Circuit Identifier used in ATM, etiqueta o encabezado

que puede utilizarse como etiqueta de MPLS en redes ATM.

WSP: Widest Shortest Path, ruteo que se basa en la búsqueda de caminos con el mínimo número de saltos y, si encuentra múltiples caminos, se queda con el que tiene ancho de banda mayor.

2.4 Abreviaturas utilizadas

ACELP - Algebraic Code Excited Linear Prediction

ADSL - Asymmetric Digital Subscriber Line

APP - Application Defined Packet

ATM - Asynchronous Transfer Mode

ATT - American Telephone and Telegraph

AVPF - Audio-Visual Profile with Feedback

BW - Bandwidth

CNAME - Canonical Name

CPU - Central Processing Unit

CSRC - Computer Security Resource Center

DiffServ - Differentiated Service

DLSR - Delay since Last Sender Report

DSP - Digital Signal Processor

EMAIL - Electronic Mail

ERL - Echo Return Loss

ETSI - European Telecommunications Standards Institute

GMPLS - Generalized Multi-Protocol Label Switching

GPL - General Public License

GSM - Global System for Mobile

HTTP - Hypertext Transfer Protocol

IETF - Internet Engineering Task Force

ILBC - Internet Low Bit Rate Codec

IntServ - Integrated Service

IP - Internet Protocol

IPDT - Internet Packet Delay Transfer

IPDV - Internet Packet Delay Variation

IPLR - Internet Packet Loss Ratio

ISDN - Integrated Services Digital Network

ISP - Internet Service Provider

ITU-T - International Telecommunications Union - Telecommunications

LSR - Last Sender Report

Mbone - Multicast Backbone

MOS - Mean Opinion Score

MPLS - Multi Protocol Label Switching

MTU - Maximum Transmission Unit

PLC - Packet Loss Concealment

PPP - Point-to-Point Protocol

QoS - Quality of Service

RFC - Request for Comments

RR - Receiver Report

RTCP - Real-Time Transport Control Protocol

RTP - Real-Time Transport Protocol

RTT - Round Trip Time

SDES - Source Description

SLA - Service Level Agreement

SLS - Service Level Specification

SR - Sender Report

SSRC - Synchronization Source

TCP - Transmission Control Protocol

TIPHON - Telecommunications and Internet Protocol Harmonization Over
Network.

UDP - User Datagram Protocol

UTC - Universal Time Coordinated

VAD - Voice Activity Detection

VoD - Video on Demand

VoIP - Voice over Internet Protocol

VQA - Voice Quality Assessment

WAN - Wide Area Network

WWW - World Wide Web.

CAPÍTULO III

3. VARIABLES E HIPÓTESIS

Se asumió como hipótesis de este trabajo:

La calidad del servicio de voz (telefonía), depende exclusivamente de los parámetros físicos que influyen sus características presentes en la red física en la cual trafican. La magnitud de la interferencia o degradación depende de la magnitud de estos factores los cuales son indicados a seguir y fueron retirados de los documentos publicados por la UIT (Unión Internacional de Telecomunicaciones) organismo especializado de las Naciones Unidas en el campo de las telecomunicaciones que estudia los aspectos técnicos, de explotación y tarifarios y publica recomendaciones sobre los mismos, con miras a la normalización de las Telecomunicaciones en el plano mundial.

La Asamblea Mundial de Normalización de las Telecomunicaciones (AMNT), que se celebra cada cuatro años, establece los temas que han de estudiar las Comisiones de Estudio del UIT-T, que a su vez producen recomendaciones sobre dichos temas. La aprobación de recomendaciones por los miembros de la UIT-T es el objeto del procedimiento establecido en la Resolución 1 de la AMNT.

En ciertos sectores de la tecnología de la información que corresponden a la esfera de competencia del UIT-T, se preparan las normas necesarias en colaboración con la ISO y la CEI (Comisión electrónica Internacional). La norma vigente es la Rec. UIT-T G.1020 (07/2006).

3.1 Definición de las variables en la investigación

Para los fines de este trabajo hemos considerado los parámetros a seguir como los influyentes en la calidad del servicio de voz: retardo, variación de retardo, pérdida de paquetes y ancho de banda.

A.- Variables independientes

- **IPDT = Atraso:** El intervalo de tiempo entre la salida del paquete de la fuente y la llegada al destino.
- **IPDV = "Jitter":** Variación del atraso: cada paquete demora un cierto tiempo variable en llegar a su destino.
- **IPLR = Pérdida de paquetes.** El porcentaje de unidades de datos que no llegaron a su destino en un intervalo dado.
- **Ancho de Banda. (banda):** El ancho de banda necesario para el canal de voz en función del Codec utilizado.

B. Variables dependientes

- **QoS = Calidad de Servicio**

3.2 Operacionalización de variables

a.- Calidad de servicio normal

- **QoS= f (IPDT; IPDV; IPLR, Banda):**
- **Donde: $IPDT \leq 150ms$**
- **$IPDV \leq 50 ms$**
- **$IPLR \leq 1\%$**
- **Ancho de Banda = Función de Codec**

b.- Calidad de servicio durante la congestión:

- **QoS_c = f (IPDT_c, IPDV_c, IPLR_c, BANDA):**
- **Dónde:** IPDT_c > 150ms, IPDV_c > 50ms, IPLR_c >1%, BANDA = f(CODEC)

Operacionalización de Variables:

- **QoS = f (IPDT; IPDV; IPLR, BANDA)**
- **QoS_{servicio-voz} = [α.f(IPDT)+β.g(IPDV)+γ.t(THRU)+δ.w(IPLR)]f(IPDT).g**
IPDV).t(ITHRU).w (IPLR)]
- Donde: $0 \leq \{\alpha, \beta, \gamma, \delta\} \leq 1$, y
- $(\alpha+\beta+\gamma+\delta) = 1$

3.3. Hipótesis

3.3.1 Hipótesis general

Manteniendo los valores de los parámetros de red:

- IPDT = Atraso ≤ de X ms.
- IPDV = Variación del atraso ≤ Y ms.
- IPLR = Pérdida de paquetes ≤ Z %
- BANDA = (depende del Codec utilizado)

Se garantiza calidad de los servicios Multimedia. X, Y, Z y BANDA a ser determinados.

3.3.2 Hipótesis específica

Manteniendo los parámetros de QoS por debajo de los valores indicados, la QoS de voz será garantizada.

IPDT < 150 ms, IPDV ≤ 50 ms, IPLR ≤ 1%, BANDA = dependiente del códec de voz.

CAPÍTULO IV

4. METODOLOGÍA

4.1. Tipo de investigación

Documental / Experimental vía simulación y "real".

De una manera general, las actividades fueron: Identificación del problema, verificación de la documentación existente, selección de los instrumentos de recolección de datos, definición del procedimiento y método de obtención de los datos, resultados e informe final.

Los datos resultantes fueron obtenidos por simulación con el software OPNET MODELER, versión académica. (Esta herramienta permite modelar, el tráfico través de un potente y amigable Interface gráfico), contrastando con resultados obtenidos en pruebas con equipos (ruteadores) reales.

4.2 Diseño de la investigación

La estrategia seguida fue la siguiente:

- **Verificación** del grado de influencia de flujos UDP-UDP Y TCP - UDP, traficando en el mismo enlace con un flujo de voz, G711, G 729, GSM y G723.1A, simulando situaciones de congestionamiento y calculando los valores de QoS correspondientes.
- **Repetición** de las experiencia indicada arriba, pero esta vez con equipos (routers) "reales".
- **Contrastación** de los valores obtenidos en las 2 experiencias, simulación y real.

- **Ejemplo** de implementación de QoS para tráfico sensible a pérdida de paquetes.

4.2.1 Introducción

En este Capítulo presentamos aspectos relativos a la motivación, objetivos y principales contribuciones de esta tesis.

4.2.2 Motivación:

El control de la Calidad de Servicio (**QoS**) es una de las tecnologías clave para las redes de la próxima generación (**NGN**). Actualmente, sólo la calidad de la red "troncal" está totalmente gestionada, pero la **QoS** fin-a-fin, para los usuarios finales necesita ser también garantizada.

La calidad de una red pública conmutada de telefonía (**PSTN**) es totalmente garantizada, y la calidad del terminal telefónico es bastante estable. De esta forma, el gerenciamiento de la calidad fin-a-fin de los servicios de telefonía fue conseguido. De otro lado, ninguna de las redes **IP** o servicios **VoIP** tiene su calidad garantizada. Existen numerosos tipos de terminales de **VoIP** con diferencias en sus implementaciones que afectan fuertemente la calidad de voz de los usuarios finales, indicando la necesidad urgente de tecnologías de gerenciamiento de calidad embarcadas en los terminales **VoIP** para conseguir el equivalente **QoS**, fin-a-fin, que el conseguido por la **PSTN**. Estas tecnologías son recomendadas en la **RFC 4585 "Extended RTP Profile for RTCP"** del **IETF** y en la recomendación **P.564** del **ITU-T**, como la estructura indicada para el gerenciamiento fin-a-fin de la calidad, pero la forma de su uso no es indicada. En este trabajo se está proponiendo un

método de gerenciamiento fin-a-fin de la calidad de las señales de voz usando "*Extended RTP Profile for RTCP*".

Las normas E.800 de la ITU-T (*International Telecommunications Union – Telecommunications*) [1] y ETSI (*European Telecommunications Standards Institute*) [2] definen **QoS** de la siguiente manera: "es el efecto global de *performance* de un servicio el que determina el grado de satisfacción del usuario". El IETF (*Internet Engineering Task Force*) lo considera como "la habilidad de segmentar tráfico o diferenciar entre tipos de tráfico" indicando que la red debe tratar algunos flujos de forma diferente de otros. Así, **QoS** engloba la categorización del servicio y la *performance* total de la red para cada categoría. Observe que el objetivo principal debe ser la satisfacción de los usuarios (calidad percibida) de forma subjetiva. En el caso de los servicios de telecomunicaciones, una buena parte de la satisfacción del usuario guarda relación directa con el funcionamiento de la red, razón por la cual la mayoría de las arquitecturas estandarizadas por el IETF están centralizadas en la red, como es el caso de la RFC (*Request for Comments*) 2386 del IETF [3], que define **QoS** como: "el conjunto de requisitos del servicio que la red debe atender para el transporte de un flujo". Esta definición está centrada en lo que [4] denomina "calidad de funcionamiento de la red", y en ella se encajan las diversas propuestas de soluciones para las redes IP como **IntServ** (*Integrated Service*), **DiffServ** (*Differentiated Service*), **MPLS** (*Multi Protocol Label Switching*) o **GMPLS** (*Generalized Multi Protocol Label Switching*), propuestas por el IETF, y las

recomendaciones Y-1540 y Y-1541, propuestas por el ITU-T en 2003. Este cuadro indica la necesidad urgente de mecanismos que proporcionen realimentación adecuada sobre la **QoS** que está siendo ofrecida por la red. En el caso de los servicios de telefonía fija y móvil, los organismos de estandarización han conseguido establecer una correspondencia entre calidad percibida y calidad de funcionamiento de la red. Como ejemplo, en el servicio de voz el modelo E, de la recomendación G.107 [5] permite evaluar como la calidad de la conversación telefónica es afectada por los parámetros de los sistemas de transmisión y, como consecuencia, permite garantizar el grado de satisfacción de los usuarios limitando las características de transmisión fin-a-fin. Sin embargo, en el caso de los servicios de acceso a *Internet* existen aún grandes dificultades para establecer la relación entre la calidad percibida y la calidad de funcionamiento de la red, debido principalmente al carácter flexible y multiservicio de la *Internet*, que posibilita el uso de múltiples aplicaciones simples y complejas, pasando por navegación *web* o correo electrónico, hasta las actuales complejas aplicaciones multimedia interactivas.[6] [7]. La calidad del servicio de acceso a *Internet* dependerá, por tanto, de la calidad percibida en cada una de las diferentes aplicaciones en ejecución, pero para conseguir esto, faltan todavía investigaciones conclusivas sobre el nivel de satisfacción de los usuarios en cada una de las aplicaciones individuales que son utilizadas en la *Internet* y sobre todo falta identificar como las diferentes percepciones individuales inciden en la calidad del

servicio de acceso. El funcionamiento de las diversas aplicaciones es influenciado de forma diferente por variables como la capacidad de transferencia o ancho de banda en la comunicación, el retraso que sufren los paquetes, la variación del retraso y la pérdida de paquetes. Es posible prever que los efectos de esos parámetros pueden afectar de manera sustancial la calidad de funcionamiento de las aplicaciones, sea directamente (en VoIP puede causar "cortes" en las conversaciones telefónicas), o indirectamente (efecto negativo en la capacidad de transferencia). Peor aún si se considera que la *Internet* está interconectada a redes de diferentes tecnologías, operadas por diferentes proveedores y con un único servicio, el *best effort* como base, el cual atiende muy bien a las aplicaciones tradicionales como correo electrónico, transferencia de archivos, telnet, HTTP (*Hypertext Transfer Protocol*), etc., la mayoría de ellas basadas en el protocolo TCP (*Transmission Control Protocol*), pero que no garantiza cualquier control de estos parámetros.

Intentando mejorar este cuadro, y visando proporcionar fiabilidad y eficiencia en la *Internet*, un gran número de propuestas vienen siendo presentadas a los organismos de estandarización. La forma de implementación de todas ellas se resume a dos metodologías: 1) ofrecer servicios con calidad garantizada para flujos individuales; 2) ofrecer servicios con calidad garantizada para flujos agrupados en clases.

Las aplicaciones *real-time* son las que más fuertemente dependen de los parámetros de la red y el grado de tolerancia o sensibilidad a cada uno de

esos parámetros varía de una aplicación para otra. Las llamadas aplicaciones "críticas", tales como telemedicina o negocios *on-line*, requieren alta fiabilidad de estos parámetros, bien como garantía de entrega de los paquetes en su destino. Otras, como las llamadas "aplicaciones emergentes" del tipo "*home networking*" (aparatos domésticos inteligentes, bien como la gran mayoría de las aplicaciones multimedia), toleran algunos de esos parámetros, pero requieren, también, diferentes niveles de calidad de servicio en la red.

Cuanto a la forma de evaluación de la calidad, en la mayoría de las aplicaciones multimedia es hecha, actualmente, de forma "subjetiva", que significa que lo que interesa es la percepción de la calidad de la comunicación por parte de los usuarios y, como, esta forma de evaluación toma demasiado tiempo y es muy cara, están siendo investigados métodos de evaluación subjetiva a partir de las características físicas de los terminales y redes o de las señales de voz que ellos generan. Tales métodos caracterizan la llamada "evaluación objetiva de la calidad".

Las metodologías para evaluación objetiva de la calidad pueden ser categorizadas en 3 grupos:

- Las que exploran los parámetros de calidad de la red y de los terminales que son llamadas de "*opinion models*".
- Las que requieren señales de voz como entradas y producen estimativas de la calidad de la escucha, conocidas como "*speech-layer objective models*".

- Las que exploran las características de los paquetes IP y estiman la calidad a partir de la escucha, conocidas como “*packet-layer objective models*” (también conocidos como “paramétricos”).

Los detalles de cada metodología no son parte del objetivo de este trabajo, pero hace parte del modelo “*packet-layer*”, tanto en la forma subjetiva cuanto en la objetiva.

Para finalizar este párrafo, se puede decir que, entre otros, por los motivos citados anteriormente, la calidad de servicio en redes IP es un punto muy activo de investigación incluyendo este trabajo de tesis.

4.2.3 Contribuciones

El objetivo general de este trabajo es proponer un método para seguimiento, evaluación y restauración de la calidad de servicio (**QoS**) para aplicaciones Multimedia sobre IP (**VoIP**), a partir de las características más relevantes de las redes y que influyen en la calidad de los servicios de voz y video partiendo de las siguientes premisas/referencias e incluyendo las actividades siguientes:

- a. Como datos de entrada para evaluación: las estadísticas generadas por el protocolo **RTCP** (*Real-Time Transport Control Protocol*) de acuerdo con la recomendación **RFC 4585** “*Extended RTP Profile for RTCP*” de Julio de 2006.
- b. Como límites de las métricas de **QoS**, los valores “objetivos” máximos siguientes (compatibles con las recomendaciones **ITU-T-Y-1540/1541**):
 - **IPDT** (*Internet Packet Delay Transfer*) (retraso) ≤ 150 ms.

- **IPDV** (*Internet Packet Delay Variation*) (variación del retraso) ≤ 50 ms.
 - **IPLR** (*Internet Packet Loss Ratio*) (tasa máxima de pérdida de paquetes) $\leq 1\%$.
- c. Como medio de restauración: un algoritmo a ser implementado en la capa de aplicación que cambia el *codec* por uno de menor banda cuando los límites son excedidos.
- d. Codecs utilizados: G.711, G.729, GSM (Global System for Mobile) y G.723.1 (ACELP [Algebraic Code Excited Linear Prediction]).
- e. Utilización de la Ecuación empírica 4.1, desarrollada para el cálculo de la Calidad de Servicio (**QoS**) indicada a continuación:

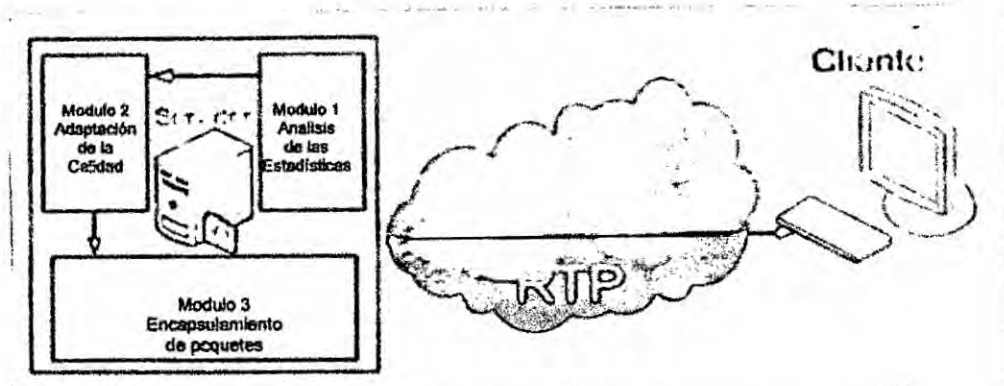
$$QoS = [\alpha \times f(IPDT) + \beta \times g(IPDV) + \delta \times w(IPLR) + \gamma \times t(THRU)] \times f(IPDT) \times g(IPDV) \times w(IPLR) \times t(THRU)$$

en la cual: $0 \leq (\alpha, \beta, \delta, \gamma) \leq 1$ y $\alpha + \beta + \delta + \gamma = 1$

- f. Modelamiento de los parámetros de QoS
- g. Validación del algoritmo en un ambiente de simulación y real.
- h. El esquema general de la propuesta es presentado en la Fig. 4.1.

FIGURA. 4.1

ESQUEMA GENERAL DE LA TESIS



Las posibles contribuciones visualizadas de esta investigación son:

- La implementación de un algoritmo de seguimiento, evaluación y restauración de **QoS** para telefonía VoIP a partir de parámetros puramente objetivos obtenidos a partir de las estadísticas fornecidas por el protocolo **RTCP** de acuerdo con la especificación de la **RFC 4585** del IETF.
- Aplicación práctica de la Ecuación 4.1 para el cálculo de la **QoS**.

Las estadísticas del protocolo **RTP** (*Real-Time Transport Protocol*)/**RTCP** son generadas automáticamente de tiempo en tiempo incluyendo los parámetros relacionados con el cálculo de las métricas de **QoS**. Restricciones cuanto a los instantes de *timing*, no permitieron hasta hoy una utilización en gran escala, de estas estadísticas, pero la introducción, en 2006, de la **RFC 4585**, tiende a cambiar esta situación debido a retirada parcial de las restricciones iniciales y al hecho de la tendencia creciente por parte de los organismos internacionales para utilización preferencial de la

evaluación objetiva de **QoS** de los *packet-layer objective models*. Los límites máximos de las métricas están indicados en la tabla 4.11. Los valores considerados ya fueran ampliamente utilizados y probados en la red pública de telefonía. La escala de calidad aceptable tiene valores de **QoS** de 0 a 1 en incrementos de 0,1 siendo "1" el valor de la calidad estándar y "0" el valor mínimo permitido. La validación fue hecha vía simulación y con equipos reales (ruteadores).

4.2.4 Estructuración de la tesis

El trabajo de tesis propiamente dicho se inicia en el ítem 4.2.1 del Capítulo IV con el título de Introducción. El trabajo completo comprende los capítulos del I a X.

A seguir una breve descripción del contenido de cada capítulo.

En la "Introducción", se hace una breve descripción de la motivación, objetivos y contribuciones visualizadas además de la estructuración completa del trabajo. En el capítulo I, se trata el Planteamiento inicial de la investigación, en el capítulo II se trata del Marco Teórico de la investigación considerando entre otros los antecedentes principales en la área de QoS y algunas abreviaciones y términos utilizados. En el capítulo III están incluidas las variables e Hipótesis hechas para este trabajo. En el capítulo IV entre otros conceptos (calidad de servicio) se presentan algunos conceptos fundamentales sobre QoS analizando entre otros los parámetros que tienen que ver con las métricas, sus valores, límites y los tipos utilizados por los principales organismos internacionales de

estandarización, se abordan las estadísticas generadas por el protocolo **RTCP** y se explica la terminología utilizada en las diversas **RFCs** aplicables al asunto. Es detallado el algoritmo de monitoreo y control de **QoS** propuesto para reducir los efectos de congestionamiento en las redes **IP** y describe cada uno de los módulos que lo componen.

Se detalla las experiencias efectuadas con flujos individuales de voz viajando solos y cuando comparten enlaces con flujos interferentes **TCP** y **UDP**. Trata también de la validación de la propuesta presentada.

En el Capítulo V "Resultados de las experiencias", se presentan los resultados de las experiencias efectuadas; En el Capítulo VI "Discusión de los resultados", se discuten y comentan los resultados obtenidos en el ítem anterior.

En el Capítulo VII se presentan las conclusiones tanto de las simulaciones como de las validaciones efectuadas.

En el capítulo VIII "Recomendaciones y Trabajos futuros" se proponen algunos puntos que se juzgan adecuados para seguir en esta línea de investigación.

En la capítulo IX son incluidas las "Referencias bibliográficas", donde se relacionan los principales datos, nombres y direcciones de los principales artículos y documentos revisados para este trabajo. Finalmente en el capítulo X, "Anexos", son incluidos: 1) La Matriz de consistencia, 2) Ejemplo básico de implementación de QoS.

4.2.5 Consideraciones respecto a QoS

La *Internet* viene creciendo rápidamente con respecto al número de usuarios y el aumento del tráfico a sido reconocido como una infraestructura muy importante para información en uso particular social y de negocios. Aunque el inicial y principal objetivo de la *Internet* há sido su conectividad y capacidad de transmisión, hace ya algún tiempo, la atención está siendo dada también, a la calidad de los servicios.

El tráfico transmitido por la *Internet* es generado por una amplia variedad de aplicaciones, las cuales tienen diferentes características y diferentes requisitos de calidad. Siendo así Calidad de Servicio y mediciones son cruciales en el control y gerenciamiento de redes. En los recientes años, diferentes enfoques han sido usados para abordar tales cuestiones. Podemos citar el enfoque 1) de *middleware* que encapsula el comportamiento de **QoS** como componentes de *software*, 2) Sistema de control de autogerenciamiento para datos de aplicaciones *streaming* 3) modelos abstractos para mecanismos de recursos llamados de **VRT** (*Virtual Resource Tree*) que enfatiza la escalabilidad como un rol importante en escenarios de aplicaciones multimedia, imponiendo gerenciamiento distribuido de recursos heterogeneos y el soporte para adaptación de los diferentes algoritmos de gerenciamiento subadjacentes. De una forma general esta discusión mostró 2 corrientes de pensamiento: Los que proponen que se implementen recursos de *software* y *hardware* para **QoS**

y los que proponen simplemente que se adicione mas banda cuando los enlaces se tornen críticos.

Los defensores de la corriente que propone implementación de mecanismos para **QoS** argumentan a los defensores de aumento de banda pura y simplemente, que el desarrollo de nuevas aplicaciones multimedia avanzadas y el uso creciente de *Internet* en nuestro día a día se consumirán a corto plazo, la banda añadida, retornándose así a la situación anterior de necesidad de más ancho de banda en un muy corto plazo. Asimismo, el número de conexiones con tráfico de otras aplicaciones tales como telnet, **WWW** (*World Wide Web*), etc., puede hacer que los paquetes *real-time* sufran variaciones y aumentos en los valores de sus métricas, especialmente en ausencia de cualquier mecanismo para establecer prioridades o equivalente, por lo tanto, una de las ventajas del uso de mecanismos de **QoS** puede ser las prioridades y la protección a flujos de tráfico seleccionado.

Últimamente la discusión de estos puntos se está dando en un enfoque más equilibrado que mezcla "ancho de banda" y la instalación de características de **QoS** específicas para aplicaciones pre seleccionadas.

Las aplicaciones multimedia utilizan **UDP** (*User Datagram Protocol*) y son particularmente sensibles a retraso fin-a-fin y la variación de retraso, aunque muchas toleren una buena cantidad de pérdida de paquetes. Además, **TCP**, en el cual la mayoría de las aplicaciones convencionales se basan, fue diseñado para dar la máxima seguridad para la entrega fiable

de datos, pero sin ningún tipo de control en caso de retraso, *jitter* o cualquier otro parámetro de **QoS**.

4.2.6 QoS y aplicaciones

A continuación un resumen de las principales aplicaciones de acuerdo con sus características básicas y requisitos de **QoS**:

a. Aplicaciones interactivas: Normalmente usadas para la relación entre personas o entre personas y máquinas. Implica en una secuencia de operaciones y transferencia de información entre los puntos finales de las aplicaciones. Estos tipos de aplicaciones pueden depender de uno o más parámetros de **QoS** al mismo tiempo, como **BW** (*Bandwidth*), retraso (*delay*), variación de retraso (*jitter*) y pérdida de paquetes (*Packet Loss*). En el caso de **VoIP**, las llamadas telefónicas pueden requerir un retraso fin-a-fin máximo de 300 ms, aunque la degradación se nota después de 150 ms. *Jitter* es aceptable hasta 50 ms, por encima del cual es muy difícil adaptarse a sus efectos sobre la conversación.

b. Aplicaciones no interactivas: No interactúan con los criterios de valoración. Por ejemplo, se puede citar los datos de copia de seguridad y transferencia de archivos.

c. Aplicaciones elásticas: Pueden trabajar en una multitud de condiciones de la red sin una degradación significativa del rendimiento. Por lo general, el servicio *best effort* abastece a este tipo de aplicaciones que pueden ser "*hard*" y "*soft*". El tipo "*hard*" no puede funcionar si sus requisitos de **QoS** no se cumplen plenamente todo el tiempo, y los "*soft*", como "multimedia",

toleran cierta degradación de los parámetros para un corto período de tiempo con calidad reducida.

d. Aplicaciones inelásticas: O *real-time* como se les conoce también, imponen controles estrictos *delay/jitter* y *bandwidth*, que son muy sensibles, pero pueden tolerar una cierta pérdida de paquetes a costa de implementación de las funciones adicionales, por ejemplo, el tamaño de los *buffers*.

e. Aplicaciones tolerantes: no deben confundirse con elásticas. Mientras que las elásticas no imponen ninguna exigencia, las tolerantes imponen restricciones de **QoS**, pero no muy rígidas, ya que permiten a las aplicaciones ejecutarse aún si los parámetros de **QoS** no son óptimos. Aplicaciones tolerantes son generalmente inelásticas con respecto a sus requisitos de **QoS**, pero si se cambian estos parámetros, pueden no ejecutarse correctamente. Un ejemplo de este tipo de aplicación es **VoIP**.

f. Aplicaciones intolerantes: Son aquellas en las que se especifican los valores fijados por su **QoS** y no ejecutan si estos valores no se cumplen, Un ejemplo de aplicación intolerante es **VoD** (*Video on Demand*).

g. Aplicaciones adaptativas: Tienden a mantener la calidad percibida dentro de un nivel aceptable, incluso bajo condiciones críticas de la red. Esto puede hacerse mediante la reducción de la tasa de transmisión, o el descenso de la resolución de la transmisión (para vídeo), o incluso usando técnicas especiales de compresión y corrección de errores. Estas aplicaciones pueden usar un "*buffer*" extra para compensar los

cambios transitorios en la red y permitir una reducción de "clase" de rendimiento. La mayoría de las aplicaciones de *streaming* de audio y vídeo en la *Internet* son de tipo adaptativa.

h. Aplicaciones no adaptativas: No pueden permitirse el lujo de hacer frente a oscilaciones transitorias de la red, ya que puede afectar directamente la calidad percibida por los usuarios finales. La adaptabilidad y la tolerancia se consideran dos características diferentes para *real-time*. En la mayoría de los casos, las aplicaciones adaptativas son tolerantes y las no adaptativas son intolerantes, pero pueden encontrarse otras combinaciones.

i. Aplicaciones de vídeo y audio real-time y streaming:

Las aplicaciones multimedia de audio y vídeo pueden ser de radio/TV *broadcasting*, telefonía IP y videoconferencia y las aplicaciones *streaming* no *real-time*, tales como VoD. Las aplicaciones *real-time* tienen requisitos de QoS más exigentes y suelen utilizar técnicas de adaptación para hacer frente a las oscilaciones transitorias de la red. Por otra parte, las aplicaciones de *streaming* de vídeo y de audio pueden retrasar el comienzo de *playback* en el plazo máximo de la red para lograr igualar *jitter*, pero esto puede causar *buffer overflow* y la pérdida de datos.

j. Aplicaciones multimedia, datos y computación a gran escala:

No todas las aplicaciones que dependen de QoS implican la transmisión de vídeo y audio. Hay muchas otras aplicaciones que requieren versiones fiables y precisas o muestras de datos periódicos, por ejemplo, las

aplicaciones de red de ordenadores, las transacciones de comercio electrónico, control remoto de los sistemas distribuidos y herramientas de colaboración, que también requieren grandes volúmenes de datos (*terabytes*) que son transmitidos y procesados entre puntos *alejados geográficamente y dentro de un cierto intervalo de tiempo*.

4.2.7 Tipos de QoS y sus parámetros/métricas

Se identifica tres tipos de QoS: Intrínseca, percibida y evaluada.

Intrínseca: Es suministrada directamente por la propia red y se puede describir en términos de parámetros objetivos, tales como retraso, pérdida de paquetes, *jitter*, etc.

Percibida: Es la calidad "sentida" por los usuarios y depende en gran medida de rendimiento de la red, pero se mide por la "opinión media" de los usuarios, normalmente a través del método de MOS (*Mean Opinion Score*), cuyos valores significan: 1 = pobre, 2 = deficiente, 3 = regular, 4 = buena y 5 = excelente, siendo el MOS la media de todos los valores individuales.

Evaluada: Se define como la QoS tolerada por un usuario que está utilizando un servicio particular. Depende entre otras cosas, del mecanismo de precios, el nivel de apoyo de su ISP (*Internet Service Provider*) y otros aspectos comerciales y de *marketing*. En este tipo, una posible reducción en la calidad sin duda va ser tolerada por el usuario si el servicio no tiene costo, pero seguramente habrá quejas si el usuario está pagando por este. En resumen, se puede suponer que la **QoS** intrínseca representa lo que podría hacer atractivo un servicio al cliente en particular. La calidad

percibida es lo que hará que el servicio sea aceptable cuando el cliente lo use. El servicio evaluado es el que determina si el cliente mantendrá el servicio o lo devolverá a la mayor brevedad.

En la actualidad, la prestación de calidad de servicio que se ofrece como intrínseca se llama **SLS** (*Service Level Specification*), que es un conjunto de parámetros y sus valores que definen conjuntamente el servicio ofrecido a un determinado tipo de tráfico. **SLS** es el detalle técnico en el **SLA** (*Service Level Agreement*), que representa el acuerdo negociado entre el cliente y el proveedor de *Internet* y que incluye características de los niveles de servicio y sus indicadores asociados.

La tabla 4.1 muestra un ejemplo típico de un **SLS**

TABLA 4.1
EJEMPLO DE UN SLS

Service Level Specification	Range
Connection type	Constante Bit Rate (CBR)
Scope	End-to-end
Connection identification	Sequence of identifiers
Traffic description and conformance testing	Peak rate = 64 Kbps/bucket size for peak rate = 512 bytes/maximym burst size not applicable
Performance guarantees	Packet loss rate = 1% packet transfer delay = 250 ms/packet delay jitter = 30 ms
Multi level precedence and preemption (MLPP)	Priority

Fuente: ITU-T

Puesto que hay muchos tipos de aplicaciones, por lo general cada una requiere un grado diferente de servicio. Varias organizaciones internacionales de normalización han estado tratando de definir las

categorías de servicios (también llamadas "clases de **QoS**") implementadas en la capa de aplicación. Entre ellas podemos mencionar: la recomendación Y-1541 del **ITU-T** que presenta una definición de los tipos de **QoS** para el mundo **IP**, que se resumen en la tabla 4.2. El **ETSI** en el proyecto **TIPHON** (*Telecommunications and Internet Protocol Harmonization Over Network*), propone una definición alternativa de **QoS**, que se resumen en la tabla 4.3.

TABLA 4.2
CLASES DE QOS DEL ITU-T

QoS class	Characteristics
0	Real-time, jitter sensitive, highly interactive
1	Real-time, jitter sensitive, interactive
2	Transaction data, highly interactive
3	Transaction data, interactive
4	Low loss only (short transaction, bulk data, video streaming)
5	Traditional applications of default IP networks

Fuente: ITU-T

TABLA 4.3
CLASES DE QOS DEL TIPHON

QoS class	Components	QoS characteristics
Real-time conversational (telephone, videoconference, videophone and	Speech, audio, video, multimedia	Delay and delay variation sensitive limited tolerance to loss and errors, constant and variable bit rate
Real-time streaming (audio and video broadcast, surveillance, graphics)	Audio, video, multimedia	Tolerant to delay, delay variation sensitive, limited tolerance to loss and errors, variable bit rate
Near real-time interactive (web browsing)	Data	Delay sensitive, tolerant to delay variation, error sensitive, variable
Non-real-time background (e-mail and file transfer)	Data	Not delay and delay variation sensitive, error sensitive, best

FUENTE: TIPHON

Derivado de las categorías enumeradas en el servicio ISDN (*Integrated Services Digital Network*), el Foro ATM (*Asynchronous Transfer Mode*) define cinco categorías de servicios que figuran en la tabla 4.4.

TABLA 4.4
CATEGORÍAS DE SERVICIO DEL FORO ATM

ATM service categories	Representative applications	QoS characteristics
CBR (Constant Bit Rate)	Circuit emulation	Low cell delay variation, low loss
rt-VRB (real-time Variable Bite Rate)	Video on demand	Moderate cell delay variation, low loss
nrt-VRB (non-real-time Variable Bite Rate)	Packet traffic	Moderate loss
ABR (avariable Bit Rate)	Adaptable rate sources	Low loss
UBR (Unspecified Bit Rate)	Best-effort traffic	No requeriments

FUENTE: FORO ATM

4.2.8 Métricas de QoS y sus límites

La especificación de los parámetros de **QoS** por lo general depende del contexto de las aplicaciones involucradas, pero los parámetros que se enumeran a continuación se consideran básicos y se discuten brevemente en lo que se refiere a su efecto sobre la comunicación.

4.2.9 Throughput

Es el número efectivo de unidades de datos transportados por unidad de tiempo (*bits/segundo*). En el caso de la voz, la banda necesaria depende del *codec* usado, de la compresión del encabezado **RTP**, **VAD** (*Voice Activity Detection*), etc.

4.2.10 Atraso (delay)

Es el intervalo de tiempo entre la salida del paquete de la fuente y la llegada a su destino. Es comúnmente conocido como *end-to-end delay*. Sus efectos sobre el rendimiento de la transmisión generalmente están separados en dos áreas: 1) el eco 2) el problema de romper el flujo normal de la conversación. Los debates sobre **VoIP** generalmente tratan de esta cuestión sobre los efectos del atraso en cuanto a su posible interferencia con la cadencia normal de las conversaciones de voz.

La recomendación G.114 del **ITU**, trata de esto y especifica que el atraso "one way" se puede acumular hasta 150 ms, sin efecto sobre la conversación. Para verificar esta recomendación, la **ATT** (*American Telephone and Telegraph*) realizó en 2004 en su laboratorio **VQA** (*Voice Quality Assessment*) nuevos estudios de atraso que se resumen a

continuación. Se hicieron llamadas telefónicas a 24 miembros del grupo del laboratorio de apoyo de **MOS, ATT, VQA** y mantenidas conversaciones de un promedio de cinco minutos de duración. El atraso "one way" de estas conexiones se fijó en 200 ms. Las conversaciones fueron concluidas preguntando a los participantes si sentían alguna diferencia de esta conversación en relación a sus conversaciones normales. Nadie se quejó o dijo algo relacionado con el atraso o la superposición de las comunicaciones. La mayoría indicó que se trataba de una conversación normal, lo que llevó a la conclusión de que hasta 200 mseg. no se observó ninguna alteración. El resultado práctico de la prueba fue la relajación por parte de la **ATT**, en el rango de atraso tolerable para las comunicaciones **VoIP**, de 150 mseg. para 200 mseg.

4.2.11 Composición del atraso fin-a-fin (E2E)

El atraso fin-a-fin puede ser de dos tipos:

- Los de tipo fijo que se suman directamente durante la conexión
- Los variables que se originan principalmente por filas en los *buffers* de salida en el puerto serial conectado a la **WAN**.

Son identificados los siguientes componentes del retraso **E2E**

- Atraso de procesamiento (*coder delay*)

Tiempo que el **DSP** (*Digital Signal Processor*) toma para comprimir un bloque de muestras **PCM**. Este atraso varía con el tipo de codificador y la velocidad del procesador.

- **Atraso de packetización**

Es el tiempo necesario para completar un paquete de datos de voz codificado/comprimido. También conocido como atraso accumulation, debido a las muestras de voz que se acumulan en buffer antes de ser liberado. Normalmente se considera un máximo de 30 ms por ese concepto.

- **Atraso de serialización**

Es el atraso fijo necesario para sincronizar una trama de datos o voz en la interfaz de red, está directamente relacionado con la velocidad del reloj del circuito.

- **Atraso de enfilamiento**

Una vez que (el paquete de voz con carga útil) se construye, tiene un encabezado añadido y el marco se pone en la cola para su transmisión por la red.

- **Atraso de Conmutación**

Las redes públicas de datos, *Frame Relay* o **ATM**, o redes que interconectan terminales, son las fuentes de mayores atrasos para las conexiones de voz y son difíciles de cuantificar.

En general, los componentes fijos son ocasionados por la propagación del atraso en los enlaces internos de la red, y los variables de los atrasos generados por las colas en los conmutadores intermedios, de entrada y salida.

4.2.12 Variación de atraso (JITTER)

Por lo general, se llama de "Jitter", a la variación del atraso. En el RTP se mide acumulativamente utilizando la ecuación siguiente:

$$J(i) = (15/16)J(i-1) + (1/16)D \quad 4.2$$

Dónde:

- J (i) = último valor reportado;
- J (i - 1) = valor anterior;
- D = diferencia de tiempo de los paquetes en el receptor con respecto al lado del transmisor. Se calcula como sigue:

$$D = (R_j - R_i) - (S_j - S_i) \quad 4.3$$

R = tiempo de llegada del paquete

S = RTP *timestamp* del paquete.

Normalmente se les llama de "interarrival time" de los paquetes RR (Receiver) del RTCP.

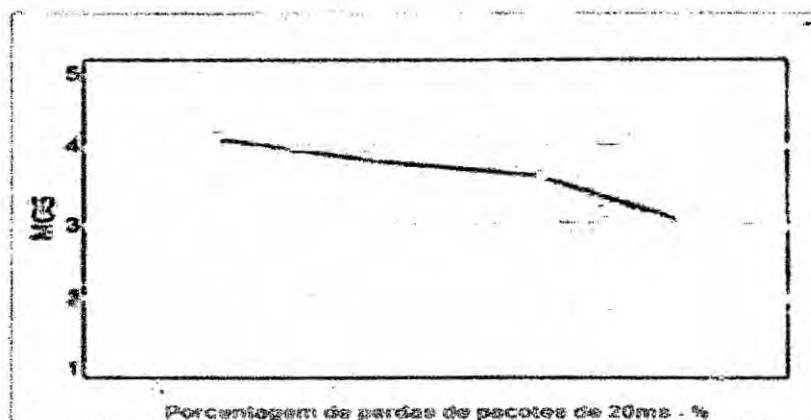
El procesamiento de paquetes de voz dentro de los *gateways* y su transmisión entre ellos no es una invariante en el tiempo. Más bien, hay una variación de retraso asociado al proceso en su conjunto, y para absorber este cambio, por lo general el *gateway* de salida "guarda" paquetes entrantes tratando de reducir la eliminación de los paquetes cuya hora de llegada es ya fuera de este rango, de manera que el "stream" de audio original de tasa constante es recuperado y entregado a su destino. ATT, menciona que en la experiencia práctica, en las redes VoIP particulares, propias y de terceros, así como en las redes de acceso de banda ancha y

ADSL (*Asymmetric Digital Subscriber Line*), el *jitter* normalmente presenta valores pequeños y valores superiores a 10 ms de retraso en relación a su tiempo previsto de llegada son raros.

4.2.13 Pérdida de paquetes (*Packet Loss*)

Es el porcentaje de unidades de datos que no han llegado a su destino en un intervalo de tiempo especificado. Por lo general se representa como una probabilidad de pérdida. La retransmisión de paquetes perdidos no cambia el valor de la probabilidad de pérdida de red, pero es un método eficaz para determinadas aplicaciones. Por encima de un límite determinado, la pérdida de paquetes en una red **VoIP** introduce distorsión de audio que causa una disminución en la calidad de la voz cuando la tasa de pérdida de paquetes aumenta. En cualquier vinculación especial en este sentido genérico puede ser modelado por: 1) la distribución de paquetes perdidos, 2) el algoritmo de **PLC** (*Packet Loss Concealment*) en uso. Muchos artículos sobre el tema, en particular los más antiguos, indican que la pérdida de paquetes no se convierte en un problema importante hasta llegar a una tasa del 5%. Un escenario más realista se presenta en la Fig. 4.3, donde fueron graficados algunos **MOS** medios obtenidos en el laboratorio de **ATT**. Los datos fueron probados en una variedad de algoritmos **PLC**, con G.711 (64 Kbps) y 20 ms para el volumen de carga útil.

FIGURA. 4.2
EFFECTO EN PÉRDIDA DE PAQUETES



FUENTE: ITU-T

4.2.14 Eco y su efecto en la calidad de VoIP

Las instalaciones iniciales de los enlaces VoIP en la red básica, mostraron una anomalía en el comienzo de una conexión, por lo general después de la primera palabra o dos palabras, llamada "eco inicial", cuyo origen se puede explicar brevemente de la siguiente manera:

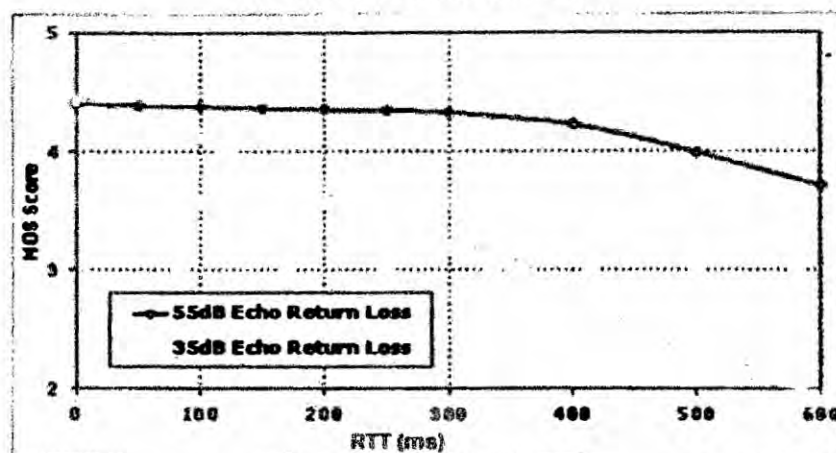
El procesador del filtro de adaptación del "supresor de eco" necesita un poco de tiempo para procesar los datos mientras está convergiendo, y por lo tanto una cierta cantidad de energía de eco residual "se escapa" del supresor de eco y se refleja de vuelta al usuario (que habla). La duración del eco inicial es una función compleja del ERL (*Echo Return Loss*) de la conexión, del proyecto del filtro adaptativo y del límite del supresor de eco (que bloquea la energía residual prácticamente "abriendo" el camino de regreso). La sensibilidad del usuario a este eco inicial aumenta directamente con el retraso de la conexión. El aumento de un enlace VoIP

no hace que el supresor de eco de la red trabaje más lento, generalmente existe antes y después de la introducción del enlace **VoIP**, lo que pasa es que sin el retraso adicional de la conexión de **VoIP**, el eco inicial está por debajo del límite de la percepción del usuario y con este retraso adicional el eco comienza a ser "sentido" (oído). Este problema está relacionado a los usuarios cuyas líneas fueron indebidamente equilibradas en la red y, por lo tanto, caracterizadas por **ERL** bajos (en el rango de 8 a 12 **dB**).

Como ya se mencionó, el eco se expresa en términos de **ERL**, que es la relación entre la señal original y el nivel de eco expresado en decibeles (**dB**). Una relación alta significa un eco bajo, ejemplo, 55 **dB**, mientras que un eco de 15 **dB**, representa un alto nivel de eco. El tipo más común de eco es conocido como "*talker echo*" (eco del usuario que habla) y representa la proporción de la voz de la persona que habla que se refleja de nuevo para sí mismo. La Fig. 4.3, muestra la relación entre *delay* y la calidad de la conversación para dos condiciones: 1) para **ERL** de 55 **dB**; 2) para un nivel moderado de eco (35 **dB** de **ERL**).

Si el **RTT** (*Round Trip Time*) es muy bajo, digamos 30 **ms**, el usuario "que habla" no puede distinguir entre el eco y el tono deliberadamente introducido.

FIGURA. 4.3
CALIDAD EN FUNCIÓN DEL ECO



FUENTE: ITU-T

Si el atraso es un poco más alto, digamos de 50 ms, el usuario que está hablando no puede escuchar la copia atrasada de su discurso como una copia distinta, pero impacta la calidad de la voz, dando como resultado una calidad de sonido generalmente descrita como "hollow" (agujero), "cave-like" (cavernoso), "tunnel-like" (túnel) o similar. A medida que aumenta el atraso, el eco se hace más evidente, y el efecto combinado de la intensidad del sonido (*loudness*) del eco y su retraso causa considerables molestias.

4.2.15 Organizaciones internacionales y sus consideraciones sobre calidad de servicio.

La evaluación de la calidad de la comunicación implica la asociación de "necesidades objetivas de calidad de servicio" con los distintos tipos de tráfico. En relación con el entorno IP, las métricas objetivas más utilizadas son los siguientes (usando la terminología de la ITU-T):

- **IPDT:** *Internet Packet Delay Transfer*
- **IPDV:** *Internet Packet Delay Variation (jitter)*

- **IPLR:** *Internet Packet Loss Ratio*
- **IPER:** *Internet Packet Error Rate*

Los parámetros antes mencionados direccionan colectivamente los aspectos más importantes de la calidad de los usuarios de las redes IP. El IPDT se refiere al tiempo medio que requiere una red para transferir paquetes del origen al destino. Su limitación es de vital importancia para las implementaciones con éxito de la VoIP, videoconferencias y aplicaciones de datos en tiempo real. El IPDV caracteriza el *jitter* y debe ser controlado para evitar *underflow* u *overflow* en los *routers IP* o *buffers* de los terminales. El IPLR debe limitarse para garantizar la inteligibilidad de la voz.

$$\text{IPLR} = \frac{\text{paquetes perdidos}}{\text{paquetes enviados}}$$

Es importante señalar que la ITU-T 1540/1541 no hace referencia al parámetro de "rendimiento" o "ancho de banda". Lo mismo ocurre con la mayoría de los trabajos y publicaciones en este ámbito. Se utilizan sólo los parámetros de retraso, *jitter* y pérdida de paquetes como métricas para QoS. Aunque la pérdida de paquetes también es usada como alerta de inicio de una situación de congestión hay que tener en cuenta que debe ser ofrecida la misma banda calculada por *codec*, cantidad de paquetes, etc. La tabla 4.5 muestra un resumen de las principales características de los *codecs* utilizados para VoIP.

TABLA 4.5

CARACTERÍSTICAS PRINCIPALES DE LOS CODECS PARA VOIP

	Codec	Codec Bite Rate (Kb)	Sample Period (msg) (Packets/seg)	Payload o Frame Sizes (bytes)	Frames por Paquete (Ethernet)	Overhead FR/ETH (bytes)	Banda Frame Relay (Kb)	Banda Ethernet (Kb)
1	G.711 (PCM)	64.000	20 (50)	160	1	47/58	82800	87.200
2	G.728 (LD-ELP)	16.000	10 (100)	20	1	47/58	26800	62.400
3	G.726 (ADPCM)	32.000	20 (50)	80	1	47/58	50.800	55.200
4	GSM	13.000	20 (50)	33	1	47/58	31800	36200
5	G.729 (CS-CELP)	8.000	10 (100)	10	1	47/58	45600	54.400
6	G.723-1 (MP-MLQ)	6.400	30 (33)	24	1	47/58	19.210	22202
7	G.723-1 A (ACELP)	5.300	30 (33)	20	1	47/58	18.190	21182
8	AMR (ACELP)	12.200	20 (50)	31	1	47/58	31.000	35400
9	AMR (ACELP)	7.400	20 (50)	19	1	47/58	26.200	30600
10	AMR WB/G.722-2 (ACELP)	6.600	20 (50)	17	1	47/58	26.000	29800
11	AMR (ACELP)	4.750	20 (50)	12	1	47/58	25400	27950

FUENTE: PROPIA DEL AUTOR

Nota: El simulador Opnet MODELER version académica usa overhead de 47 bytes en el protocolo PPP/Frame Relay. (No incluye RTP de 12 bytes), y 58 bytes para el protocolo Ethernet.

Ejemplo de calculo de ancho de banda (Para codec G723.1 A con protocolo relay).

Banda: $5300 \times 0.03 = 159/8 = 19.875 = 20$ bytes (Payload en bytes).

$20 + 47 = 67 \times 8 = 536 \times 34 = 18224$ bps (Payload + cabezal).

Como ya se mencionó, es necesario usar los valores adecuados para cada aplicación específica. Por lo tanto, debe tener una serie de requisitos de QoS por tipo de tráfico. Los valores posibles máximos se muestran en la tabla de métricas de rendimiento, fin-a-fin en la tabla 4.6 que se tomaron de la recomendación ITU-T-Y-1541.

TABLA 4.6
MEDIDAS DE "PERFORMANCE" DE LA ITU-T

QoS CLASS	CHARACTERISTICS	IPDT	IPDV	IPLR	IPER
0	Real-time, jitter sensitive, highly interactive	100 ms	50 ms	1×10^{-3}	1×10^{-4}
1	Real-time, jitter sensitive, interactive	400 ms	50 ms	1×10^{-3}	1×10^{-4}
2	Transaction data, interactive	100 ms	*U	1×10^{-3}	1×10^{-4}
3	Transaction data, interactive	400 ms	*U	1×10^{-3}	1×10^{-4}
4	Low loss only (Short transactions, buçk data, video streaming)	1 s	*U	1×10^{-3}	1×10^{-4}
5	Traditional applications of default IP networks	*U	*U	*U	*U

* U: sin especificar.

Fuente: ITU-T

Los valores indicados en la tabla se aplican a las redes públicas IP y se limitan a los valores promedios referenciales y cada empresa o ISP puede proporcionar sus propios valores de rendimiento fin-a-fin. La clase 0, por ejemplo, significa que para una aplicación genérica, en tiempo real, sensible al *jitter* y altamente interactiva, los siguientes límites deberían garantizarse: retraso de transferencia fin-a-fin promedio por debajo de 0,1 segundos, *jitter* promedio por debajo de 0,05 segundos, *packet loss rate* por debajo de 10^{-3} y *packet error rate* debajo de 10^{-4} . La recomendación no se refiere a aplicaciones específicas, establece apenas los límites para

tipos de tráfico. En la tabla 4.7 están indicados valores de referencia para algunas aplicaciones.

TABLA 4.7
EJEMPLOS DE REQUISITOS DE QOS DE ALGUNAS APLICACIONES

Application	IPDT	IPDV	IPLR
Data acquisition from sensors	100 ms	50 ms	0
Radar traces	20 msg	1-3 ms	10^{-3}
Weapon control	20 msg	10 ms	0
Sensor control	20 msg	10 ms	0
Voice	250 msg	30 ms	10^{-2}
Video streaming	5-10 msg	U	2×10^{-2}

FUENTE: ITU-T

En la tabla 4.8 están indicados los valores para voz que se utilizarán en este trabajo. Se definieron considerando los valores de las tablas anteriores y también los valores utilizados por las administraciones de telefonía convencional para conmutación de circuitos y serán las referencias (límites máximos) para todo este trabajo.

TABLA 4.8
MÉTRICAS DE ESTE TRABAJO

Aplicación	IPDT (msg)	IPDV (msg)	IPLR (%)
Voz sobre IP	≤ 150	< 50	$\leq 1\%$

FUENTE: PROPIA DEL AUTOR

Comentarios:

Hasta poco tiempo atrás la evaluación de calidad de la voz era hecha exclusivamente de forma subjetiva, a través del llamado "QoS percibido" (P QoS) medido a través de la opinión media de los usuarios, usando el llamado MOS (mean opinión score) asignando un valor a la aplicación en evaluación, con los siguientes valores de MOS: 1=malo, 2=pobre, 3=médico, 4= bueno y 5=excelente. Aunque el objetivo principal sea la calidad percibida por el usuario, la calidad de servicio depende fuertemente de la performance de la red.

Esta situación ha cambiado un poco y actualmente, la mayoría de las implementaciones de QoS son ofrecidas en términos de QoS intrínseco (parámetros objetivos) usando el llamado "Service Level Specification" (SLS), que son los parámetros con sus valores que en conjunto definen el servicio ofrecido al tráfico. SLS es la parte técnica separada del "acuerdo negociado entre el cliente y el proveedor del servicio, en lo relacionado a las características de los niveles de servicio y su conjunto de métricas asociadas" [ITU-t-Y-1241], el cual es comúnmente conocido como Service Level Agreement (SLA). En resumen podemos decir que el SLS especifica el servicio desde el punto de vista técnico y el SLA incluye también aspectos no técnicos como precios y recursos de los dispositivos de la red.

FIGURA. 4.4

CABEZALES DE LOS PRINCIPALES PROTOCOLOS USADOS EN VOIP
MESSAGE

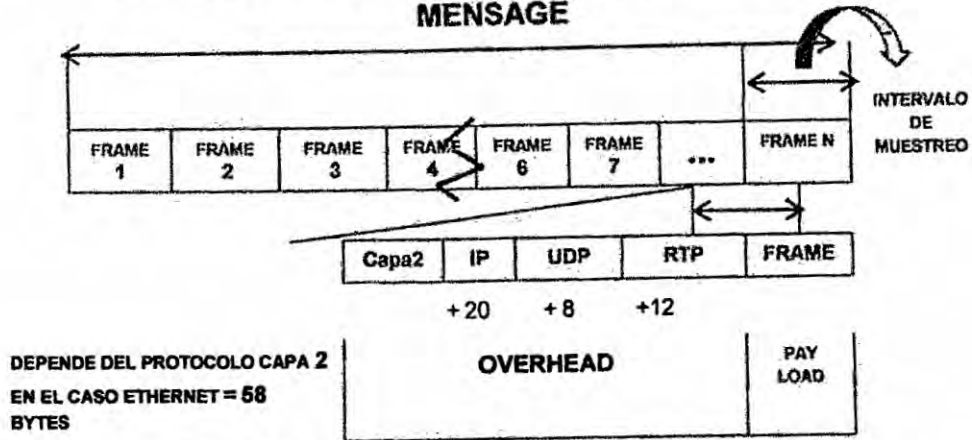


FIGURA 1- PAQUETE ETHERNET COMPLETO

8	14	40	4	12	(BYTES)
PREAMBULO ETHERNET	CABEZAL ETHERNET	CABEZAL IP	FRAME	CRC	GAP

FIGURA 2- DETALLAMIENTO DEL PAQUETE ETHERNET

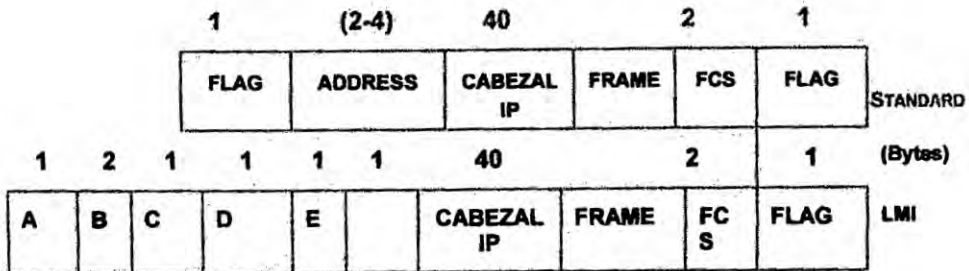


FIGURA 3- PAQUETE FRAME - RELAY

A= FLAG/B=LMI DLCI/C= UNNUMBERED INFORMATION INDICATOR/D= PROTOCOL DISCRIMINATOR/E= CALL REFERENCE / F = MESSAGE TYPE

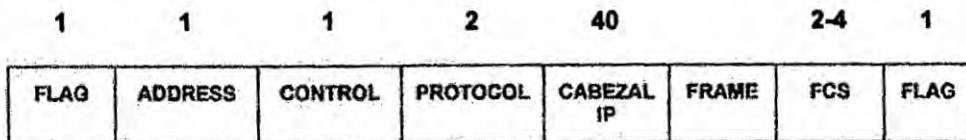


FIGURA 4- PAQUETE PPP

4.2.16 Los Parámetros internacionales de QoS de la ITU-T y TIPHON
 Las tablas, 4.9 y 4.10 muestran los valores de los parámetros de QoS y sus límites utilizados por el TIPHON, (*Telecommunications and Internet Protocol Harmonization over Network*), Órgano Europeo y el ITU-T, respectivamente.

TABLA 4.9
METRICAS TIPHON

Categoría de degradación	Pérdida de paquetes	Pico de jitter
Muy buenos	0%	0 ms
Bueno	3%	75 ms
Medio	15%	125 ms
Malo	25%	225 ms

Fuente: TIPHON

TABLA 4.10
ITU-T- RECOMENDACIÓN G:114

Atraso	Valor (mseg)
0 a 150 ms	aceptable para la mayoría de las aplicaciones del usuario
150 a 400 ms	aceptable desde que se conozca la influencia de ese atraso en la calidad de las aplicaciones del usuario
Más de 400 ms	inaceptable para efectos de planificación de la red

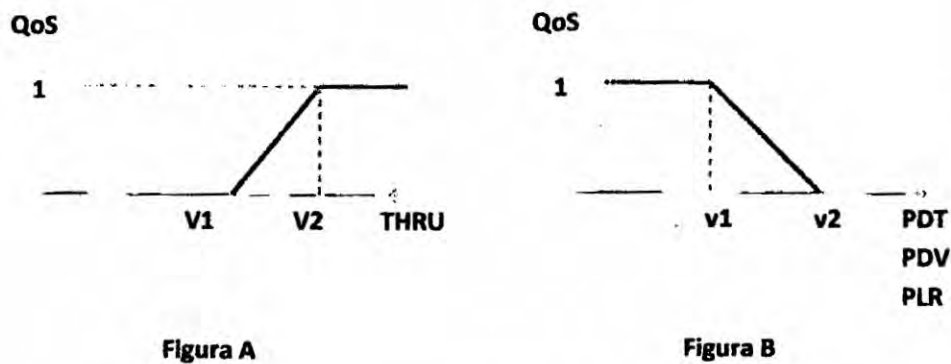
FUENTE: ITU-TN

4.2.17 Caracterización y modelaje de los parámetros /métricas del servicio de voz

Conociendo el comportamiento de los parámetros de las métricas de QoS podemos caracterizarlos por ejemplo, de acuerdo a lo indicado en la fig. 4.5; abajo, recordando que la QoS de los servicios de voz dependen de

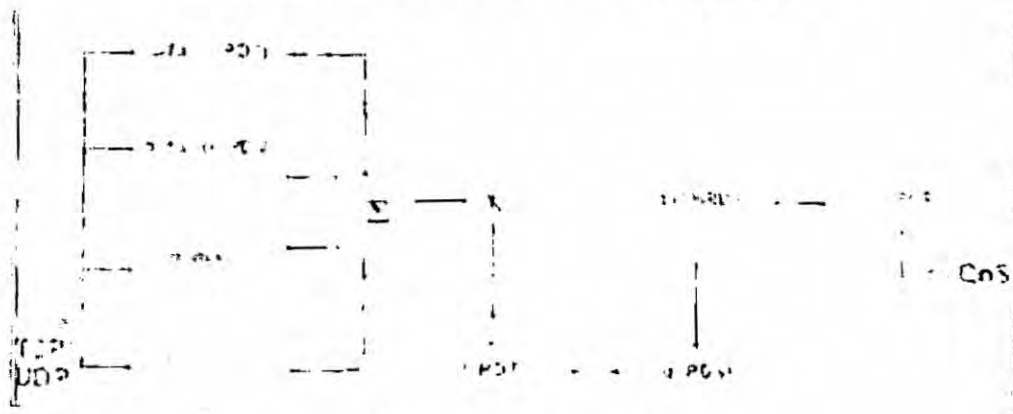
la THRU, IPDV, IDPV, e IPLR y aumentan con el aumento de THRU (fig A), y disminuyen con el aumento del IPDT, IPDV e IPLR (fig, B) y todas la afectan simultáneamente. Esta última afirmación puede ser representada por el grafico indicado en la fig. 4.6.

FIGURA.4.5
MODELAJE DE LA QoS



FUENTE: PROPIA DEL AUTOR
FIGURA.4.6

EFFECTO SIMULTANEO DE LAS MÉTRICAS DE QoS



FUENTE: PROPIA DEL AUTOR

De acuerdo con la fig. 4.6 [1], la expresión para la QoS de voz, puede ser escrita como:

Por consiguiente la fórmula para el cálculo de este modelo puede ser escrita como indicada en [1] y la expresión para la QoS de voz puede ser considerada:

$$QoS_{servicio-voz} = [\alpha \times f(PDT) + \beta \times g(PDV) + \gamma \times t(THRU) + \delta \times w(PLR)] \quad \text{Ecuación (4.1)}$$

$$\text{Donde: } 0 \leq \{\alpha, \beta, \gamma, \delta\} \leq 1, \gamma \quad \text{Ecuación (4.2)}$$

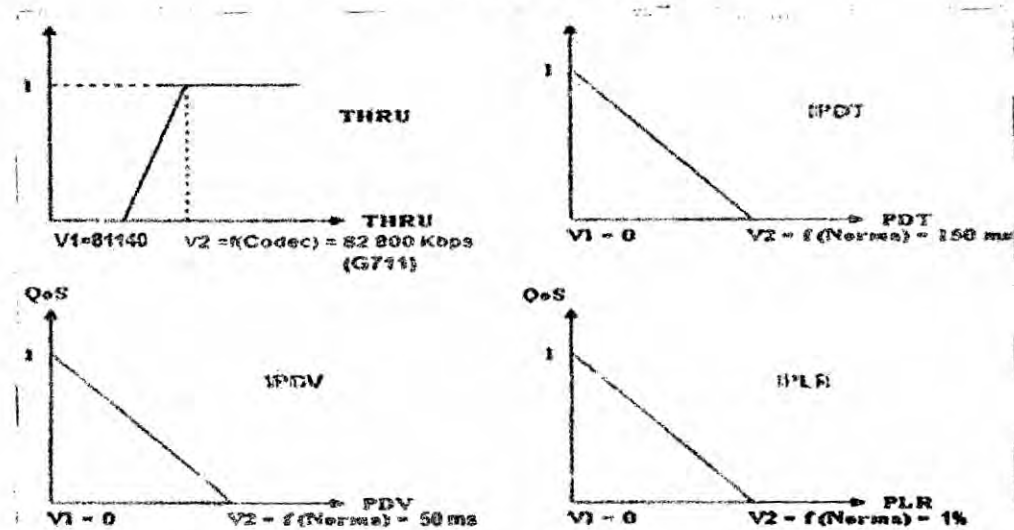
$$\alpha + \beta + \gamma + \delta = 1 \quad \text{Ecuación (4.3)}$$

Las funciones $f(\cdot)$, $g(\cdot)$, $w(\cdot)$ y $t(\cdot)$, son del tipo representado en la Fig.4.5 y deben ser determinadas para cada uno de los servicios. *Alpha, beta, gamma* y *delta* son parámetros de ponderación.

4.2.18- Los límites de los parámetros de QoS

FIGURA. 4.7

LOS LÍMITES DE LAS MÉTRICAS DE QoS



FUENTE: PROPIA DEL AUTOR

TABLA 4.11
LÍMITES DE LOS PARÁMETROS/METRICAS DE QoS

Codec	Banda V1 bps	Banda V2 bps	PDT V1 ms	PDT V2 ms	PDV V1 ms	PDV V2 ms	PLR V1 %	PLR V2 %
G.711	*81140	82800	0	150	0	50	0	1
G.729	*45140	45600	0	150	0	50	0	1
GSM	*31160	31800	0	150	0	50	0	1
G.723.1	*17650	18190	0	150	0	50	0	1

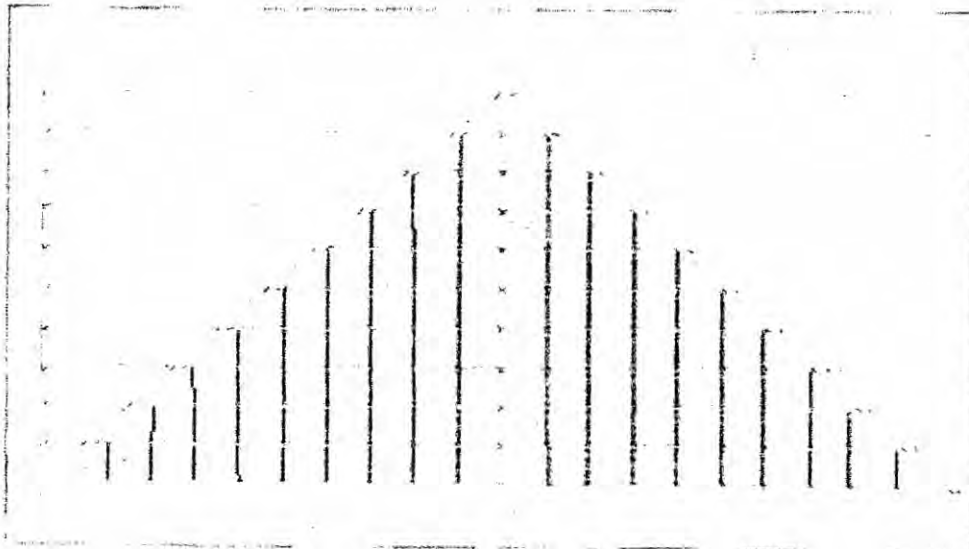
* Determinada en función de la IPLR

FUENTE: PROPIA DEL AUTOR

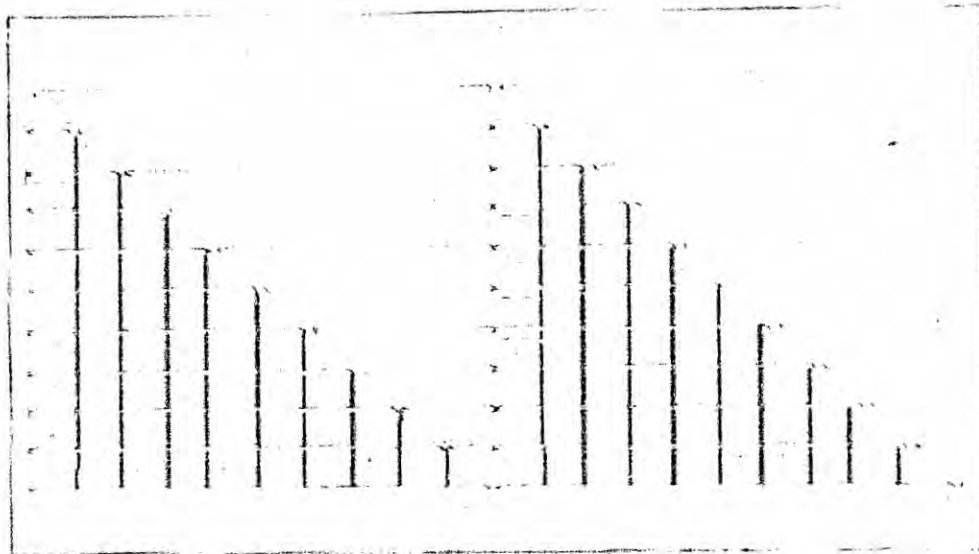
4.2.19 Cuantificación lineal de la QoS

La Fig. 4.8 abajo muestra el procedimiento de aproximación lineal de la participación de los parámetros THRU, IPDT, IPDV y IPLR en la QoS. Se asume la variación de QoS en incrementos iguales a 0.1.

FIGURA. 4.8
CUANTIFICACION LINEAL DE LA QoS



FUENTE: PROPIA DEL AUTOR



FUENTE: PROPIA DEL AUTOR

4.2.20 Cuantificación lineal de los codecs de este estudio

A seguir los resultados de la cuantificación aproximada para cada uno de los 4 Codecs analizados (para protocolo frame relay).

TABLA 4.12
CUANTIFICACIÓN LINEAL PARA EL CODEC G.711

QoS	THRU-bps	IPDT-mseg	IPDV-mseg	IPLR 3 x 10 ⁻²	IPLR Paquetes
1	82800	0.0	0.0	0.0	0.0
0.9	82634	15	.5	0.003	1
0.8	82468	30	10	0.006	1
0.7	82302	45	15	0.009	1
0.6	82136	60	20	0.012	1
0.5	81970	75	25	0.015	1
0.4	81804	90	30	0.018	1
0.3	81638	105	35	0.021	1
0.2	81472	120	40	0.024	1
0.1	81306	135	45	0.027	1
0.0	81140	150	50	0.030	1

FUENTE: PROPIA DEL AUTOR

TABLA 4.13
CUANTIFICACION LINEAL PARA EL CODEC G.729
(100 PAQUETES POR SEGUNDO)

Qos	THRU-bps	IPDT-mseg	IPDV-mseg	IPLR 3x10-2	IPLR Paquetes
1	45600	0	0	0.00	0
0.9	45554	15	5	0.003	1
0.8	46508	30	10	0.006	1
0.7	45462	45	15	0.009	1
0.6	45416	60	20	0.012	1
0.5	45370	75	25	0.015	1
0.4	45324	90	30	0.018	1
0.3	45278	105	35	0.021	1
0.2	45232	120	40	0.024	1
0.1	45186	135	45	0.027	1
0.0	45140	150	50	0.03	1

FUENTE: CODEC G.729

TABLA 4.14
CUANTIFICACION LINEAL DE LA QoS PARA CODEC GSM
(50 PAQUETES POR SEGUNDO)

Qos	THRU-bps		IPDT mseg	IPDV mseg	IPLR Paquetes
1	31800		0	0	0
0.9	31736		15	5	1
0.8	31672		30	10	1
0.7	31608		45	15	1
0.6	31544		60	20	1
0.5	31480		75	25	1
0.4	31416		90	30	1
0.3	31352		105	35	1
0.2	31288		120	40	1
0.1	31224		135	45	1
0.0	31160		150	50	1

FUENTE: CODEC G.729

TABLA 4.15
CUANTIFICACION LINEAL DE LA QoS PARA CODEC G.723.1 ACELP
(34 PAQUETES POR SEGUNDO)

Qos	THRU-bps	IPDT-mseg	IPDV-mseg	IPLR Pacotes
1	18190	0	0	0
0.9	18136	15	5	1
0.8	18082	30	10	1
0.7	18028	45	15	1
0.6	17974	60	20	1
0.5	17920	75	25	1
0.4	17866	90	30	1
0.3	17812	105	35	1
0.2	17758	120	40	1
0.1	17704	135	45	1
0.0	17650	150	50	1

FUENTE: CODEC G.723.1 ACELP

4.2.21 Cálculo de las métricas de QoS

Aplicando la formula (4.1) a los valores del Codec G.711, obtendremos:

$$QoS_{servicio-voz} = [\alpha \cdot f(IPDT) + \beta \cdot g(IPDV) + \gamma \cdot t(THRU) + \delta \cdot w(IPLR)] \cdot f(IPDT) \cdot g(IPDV) \cdot t(THRU) \cdot w(IPLR) \quad \text{Ecuación (4.1)}$$

$$\text{Donde: } 0 \leq \{\alpha, \beta, \gamma, \delta\} \leq 1, \text{ y} \quad \text{Ecuación (4.2)}$$

$$\alpha + \beta + \gamma + \delta = 1 \quad \text{Ecuación (4.3)}$$

Aproximando las métricas de QoS a la funcion "Rampa" tenemos:

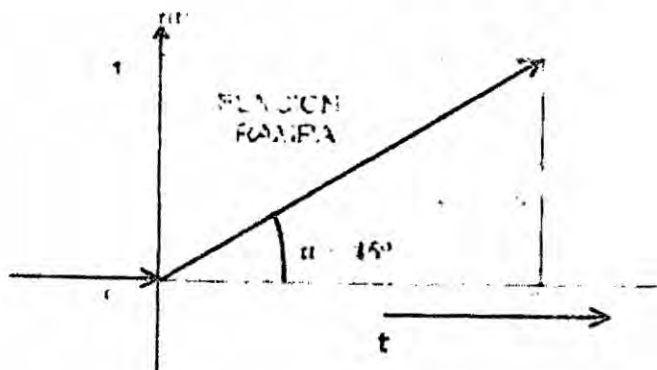
Si una función rampa continua en el tiempo es denotada por $r(t)$ y es expresada matemáticamente como:

$$r(t) = t, \text{ si } t \geq 0,0$$

$$= 0, \text{ caso contrario (esto es si } t < 0)$$

La función rampa, $r(t)$ establece que la señal iniciara en el tiempo cero e instantáneamente tomara una inclinación dependiendo de las características del tiempo, para el lado positivo o negativo de un eje de coordenadas. Así la función rampa $r(t)$ abajo indicada es un tipo de función elemental la cual existe solamente para el lado negativo.

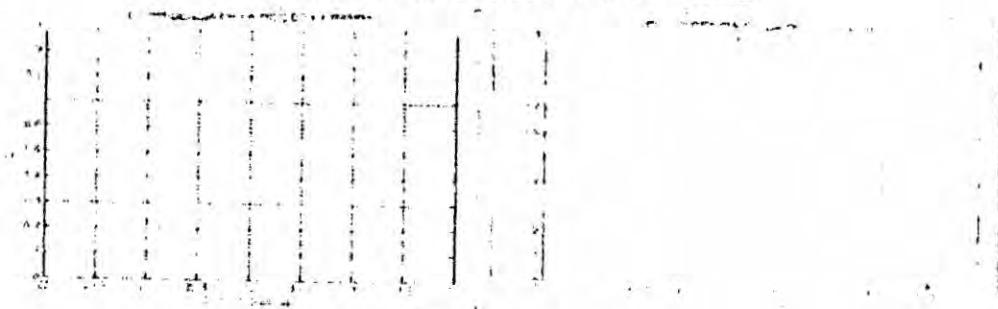
FIGURA.4.9
FUNCION RAMPA



Fuente: Propia del Autor

Utilizando la función rampa de la fig.4.9 y también las fig.4.10 a, y b, tomadas de [8] podremos calcular las funciones $f(\text{IPDT})$; $g(\text{IPDV})$; $t(\text{THRU})$; y $w(\text{IPLR})$.

FIGURA 4.10
DEPENDENCIA DE LA QOS DEL IPDT Y THRU



FUENTE: PROPIA DEL AUTOR

4.2.22 Ejemplo de Cálculo de QoS

- Para un IPDT de 90 y para un IPDV de 30 codec G.711
- Para $\alpha = 0.25$, $\beta = 0.25$, $\gamma = 0.25$ y $\delta = 0.25$
- Aplicando la ecuación 4.1 y función rampa a las funciones $f(\cdot)$:
 - $f(\text{IPDT}) = f(V2\text{IPDT} - \text{IPDT}/V2\text{IPDT} - V1\text{IPDT}) = 150 - 90/150 = 0.4$ mseg.
 - $g(\text{IPDV}) = (V2\text{IPDV} - \text{IPDV})/V2\text{IPDV} - V1\text{IPDV} = 50 - 30/50 = 0.4$
 - $t(\text{THRU}) = (V2\text{THRU} - \text{THRU})/V2\text{THRU} - V1\text{THRU} = 82800 - 81804/82800 - 81140 = 0.6$
- $W(\text{IPLR}) = (V2 \text{ IPLR} - \text{IPLR}/V2 \text{ IPLR} - V1\text{IPLR}) = (\text{NOTA})$

NOTA: De acuerdo con la tabla 3.6, la ITU-T recomienda 1×10^{-3} paquetes/seg., o 1×10^{-4} bits por seg.

Somos del parecer de usar esta métrica para indicar alerta de una anomalía en curso tipo inicio de congestión.

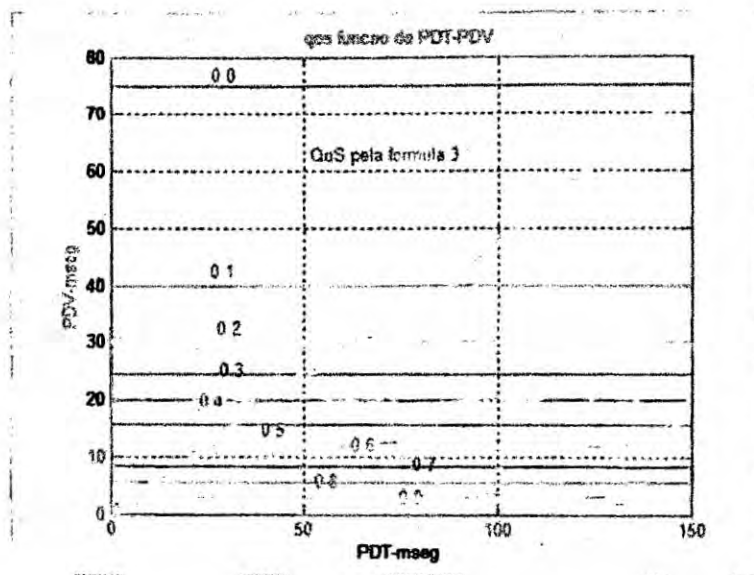
Por tanto $QoS = (0.4 + 0.4 + 0.6) \times 0.25 = 0.35$

4.2.23 QoS de tres métricas

Como mencionado arriba, es nuestra opinión, usar la métrica IPLR como alerta de una situación que pueda resolver la situación cuando conmutar para un códec de menor banda. De esta forma se recomienda dejar la fórmula con 3 métricas hasta efectuar pruebas de sistema que confirmen la validez de esta decisión. Después de algunas pruebas analizando el comportamiento sería decidido si vale la pena continuar como una alerta del sistema de una situación de congestión en curso.

FIGURA 4.11

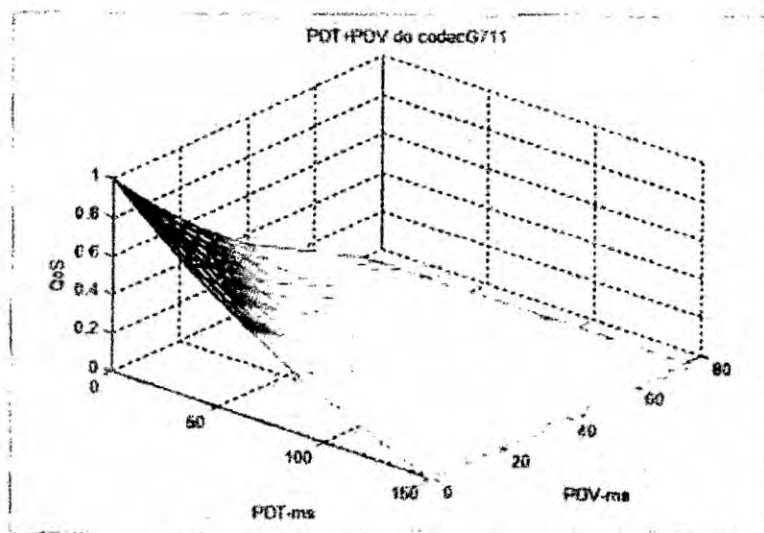
QoS FUNCIÓN DEL IPDT E IPDV



FUENTE: PROPIA DEL AUTOR

FIGURA. 4.12

QoS POR LA FÓRMULA 4.2 (IPDT + IPDV)



FUENTE: PROPIA DEL AUTOR

4.2.24 Protocolo RTP

El **TCP** es actualmente, el protocolo de transporte estándar utilizado en Internet (aproximadamente 60%). Sin embargo, varios hechos lo hacen inadecuado para su uso con tráfico real-time. Entre ellos:

- **TCP** es un protocolo punto a punto sin un apoyo directo para la transmisión multicast.
- Utiliza un mecanismo de retransmisión interno inútil para aplicaciones en real-time.
- No transporta ningún tipo de información de "timing" (requerido por la mayoría de las aplicaciones real-time).

El otro protocolo de transporte **UDP**, tampoco incluye ninguna información de "timing". Por estos motivos, entre otros, un nuevo protocolo de transporte, llamado **RTP**, fue especificado por el **IETF** y rápidamente adoptado como protocolo estándar para el transporte de sesiones multimedia por la **ITU-T**. El **RTP** es considerado como un protocolo de transporte real-time fin-a-fin, recordando solamente que llamarlo así (como un protocolo de transporte) puede llevar a confusión, ya que se utiliza normalmente con **UDP** o **TCP**, que son, de hecho, los protocolos de transporte. Como actúa muy próximo de la aplicación transportada, es visto más como un marco que las aplicaciones pueden utilizar para implementar un nuevo protocolo (individual). El **RTP** no garantiza la entrega sincronizada ni ordenada de paquetes y deja la responsabilidad de recuperación de los segmentos perdidos y el ordenamiento de los paquetes por cuenta de la

capa de aplicación. Funciona junto con el **RTCP**, su protocolo de control y retroalimentación de la calidad de la comunicación.

4.2.25 Protocolo RTCP

Utilizando el **RTCP**, el lado "receptor" de la aplicación envía "informes de recepción" (**RR**) para su fuente de datos. Estos informes incluyen informaciones de los parámetros que permiten calcular las métricas de **QoS** específicamente la pérdida de paquetes, variación de retraso (*jitter*) y retraso (*delay*). En general, el **RTCP** realiza cuatro funciones básicas:

- Proporciona mejora de la calidad de la comunicación;
- Mantiene un registro de todos los participantes de la reunión a través del campo **CNAME** (*Canonical Name*).
- En el caso de la conferencia, todos los participantes envían paquetes **RTCP** a todos los demás participantes.
- Transporta alguna información de control de la sesión.

Cada paquete **RTCP** comienza con un encabezado fijo de 8 *bytes* seguido de elementos estructurados que pueden ser de longitud variable, de modo que varios paquetes **RTCP** pueden ser concatenados sin la necesidad de un separador, formando un paquete compuesto que luego se encapsula en un paquete **UDP/IP**.

4.2.26 Estadísticas del protocolo RTP/RTCP

El **RTP** consiste de 2 protocolos: **RTP** para aplicaciones transmitiendo datos real time y **RTCP** para monitoramiento de **QoS** y transmitir informaciones de los participantes durante una sesión.

RTP es normalmente integrado dentro de la aplicación, sus paquetes son normalmente encapsulados dentro de paquetes UDP a los cuales proporciona checksum y multiplexación. Los paquetes RTP consisten normalmente de un header y datos de payload los cuales pueden ser frames de video o varias muestras de audio; algunos campos en el header pueden ser:

- **Payload type:** Identifica el formato del payload RTP, ejemplo H.261 para video.
- **Sequence number:** El cual es incrementado por uno cada vez que un paquete es enviado. Normalmente usado para detectar pérdida de paquetes o paquetes fuera de secuencia.
- **Timestamp:** Es el instante en que el paquete de datos es generado. Usado en sincronismo y cálculo de jitter.

RTCP es usado para transmisión periódica de paquetes de control a todos los participantes de la sesión. Una idea de su funcionamiento es indicada a seguir:

- **Monitoramiento de QoS:** Los informes de RTCP incluyen ciertas estadísticas de control, las cuales son enviadas como realimentación de la calidad de los datos distribuidos. Un tipo de informe de realimentación es llamado "receiver report" (RR), este es transmitido del receptor de vuelta al transmisor. El contiene estadísticas de performance sobre número de paquetes perdidos, secuencia de paquetes máxima recibida, jitter y otras medidas de atraso para calcular el RTT.

- **Identificación:** RTCP transporta información de identificación acerca de los participantes de la sesión en otro tipo de informe llamado "Source description items" (SDES)

4.2.27 RFCs del RTP/RTCP y sus principales especificaciones

Los estándares del IETF participantes en el tema de esta investigación fueron:

- **RFC 1889**, Enero de 1996, "*RTP: A transport protocol for real-time applications*".
- **RFC 1890**, Enero de 1996, "*RTP: Profile for audio and videoconferences with minimal control*".
- **RFC 3550**, Julio de 2003, "*RTP: A transport protocol for real-time applications*".
- **RFC 3551**, Julio de 2003, "*RTP: Profile for audio and videoconference with minimal control*".
- **RFC 3611**, noviembre de 2003, "*RTP: Control protocol extended reports (RTCP-XR)*".
- **RFC 4585**, Julio de 2006, "*Extended RTP profile for RTCP - based feedback (RTP/AVPF [Audio-Visual Profile with Feedback])*".
- **RFC 4586**, Julio de 2006, "*Real-time transport control protocol (RTCP) based feedback: result of the timing rule simulation*".

Antes de la creación de la **RFC 4585**, las principales normas de la RTP (**RFC 1889** y **3550**) restringían el envío de información estadística, como se muestra a continuación:

- a) Los mensajes se envían en paquetes **RTCP** compuestos que incluyen al menos un **SR** (*Sender Report*) o **RR** y un mensaje de **SDES** (*Source Description*)
- b) Estos paquetes **RTCP** son enviados en intervalos de tiempo (**T-rr**) calculados en función del tamaño medio del paquete, número de transmisores y receptores en el grupo y del ancho de banda de la sesión (un porcentaje del ancho de banda de la sesión es asignada para los paquetes **RTCP**). Esta banda es dividida entre todos los miembros de la sesión y los transmisores pueden conseguir más banda que los receptores.
- c) El intervalo mínimo entre 2 paquetes **RTCP** de la misma fuente era de 5 segundos.
- **RFC 3550**, Julio de 2003, "**RTP: A transport protocol for real-time applications**".
 - **RFC 3551**, Julio de 2003, "**RTP: Profile for audio and videoconference with minimal control**".
 - **RFC 3611**, noviembre de 2003, "**RTP: Control protocol extended reports (RTCP-XR)**".
 - **RFC 4585**, Julio de 2006, "**Extended RTP profile for RTCP - based feedback (RTP/AVPF [Audio-Visual Profile with Feedback])**".
 - **RFC 4586**, Julio de 2006, "**Real-time transport control protocol (RTCP) based feedback: result of the timing rule simulation**".

El objetivo de estas reglas fue prevenir una explosión de realimentación de estadísticas y escalar grupos *multicast* muy grandes. Todavía, se observa que principalmente la regla del intervalo mínimo puede impedir a los receptores de enviar realimentación en tiempo hábil, es decir, enviar fuera del instante en que es realmente necesario (es preciso esperar 5 segundos).

La RFC más reciente, la 4585, de julio de 2006, introduce cambios en el perfil RTP para audio y vídeo. Los principales cambios son:

- Fue introducido el concepto de “*early RTCP* mensaje” que, actuando en conjunto con algoritmos adecuados, propicia realimentación con bajos retrasos.
- Introducción de pequeños mensajes de realimentación que serán transmitidas en los “*payloads*” del RTCP.

En la verdad, esta RFC visa resolver las restricciones principalmente de “*timing*” especificadas en la RFC 3550/3551, pero manteniendo la mayoría de los algoritmos del RTP y RTCP originales, que probaron trabajar adecuadamente en la *Internet*. Los detalles pueden ser encontrados en la propia RFC 4585.

4.2.28 RFC 4585 (AVPF) y principales características

Esta RFC prácticamente corta las limitaciones existentes hasta antes de su creación. Los cambios introducidos en el nuevo perfil del RTP-AVPF pueden ser resumidos básicamente en dos:

- El intervalo mínimo de 5 segundos es abolido.

- Los receptores pueden, durante el intervalo, enviar una única vez un paquete **RTCP** llamado "*early packet*", que no necesita esperar que el "intervalo calculado" transcurra para enviar estadísticas, si no que puede transmitir prácticamente de inmediato. Es importante notar que el intervalo de envío de las estadísticas **RTCP** se mantiene calculado de acuerdo con su forma original detallada en la **RFC 1889**.

Durante la operación normal, todos los miembros del grupo, calculan el intervalo **T-r** de acuerdo con las reglas especificadas del **RTP** original, sin que exista el intervalo mínimo de 5 segundos. Esto genera condiciones para que los algoritmos de control de congestión trabajen adecuadamente, mismo cuando se trata de pequeños grupos o *unicast*. Las estadísticas son muy frecuentes y no es necesario esperar por los 5 segundos inicialmente definidos. Esto significa, por ejemplo, que si un receptor detecta la presencia de pérdida de paquetes, puede enviar muy rápido cuanto posible, paquetes de "realimentación" por medio de un "*early packet*" sin tener que esperar el tiempo regularmente programado.

4.2.29 El algoritmo de generación del "*Early Packet*"

De modo general, el algoritmo para generación del "*early packet*" funciona de la siguiente forma: Si el receptor necesita enviar la "realimentación" en el tiempo **T-0** primero verifica si el paquete **RTCP** anterior ya fue enviado como "*early packet*" o si otro "*early packet*" ya está programado para transmisión. En caso afirmativo, no puede enviar otro paquete de este tipo, pero puede adjuntar la "realimentación" en el próximo paquete **RTCP**

normalmente programado. Si aún no fue enviado ningún RR, programa un "early packet", calculando el llamado "maximum dithering interval".

4.2.30 Cálculo del "Maximum Dithering Interval" (T-dither-max)

- Si es una sesión *unicast* (tamaño del grupo = 2), **T-dither-max** = 0. Esto debido a que no tiene sentido un receptor *unicast* esperar por otros receptores si ellos tienen la misma información para enviar.
- Si el receptor tiene un valor de **RTT** (conocido) hasta el transmisor de la media, entonces el **T-dithermax** = $K \times T\text{-rtt}/2 \times \text{Members}$ (con $K = 1$, **T-rtt** indicando el *round trip time*).
- Si el receptor no tiene un valor de **RTT** (conocido) hasta el transmisor, entonces:

$$T - dither - max = lXT - rr, \text{ en que } l = 0,5 \text{ y } T - rr \quad 3.4$$

indica el intervalo RTCP

Si el próximo paquete **RTCP** regularmente programado está dentro del intervalo $T-0$ a $(T-0 + T\text{dither-max})$, no puede ser programado un "early packet", pero las estadísticas son adjuntadas en el próximo paquete **RTCP**. Otro "dithering interval", **T-dither** es escogido al azar del intervalo $[0; T\text{-dither-max}]$. El "early packet" es programado para $T-e = T-0 + T\text{-dither}$.

De modo general, los "early packets" también son paquetes **RTCP** compuestos, pero pueden no incluir mensajes **SDES**, por tanto, consistirían sólo en mensajes **SR** o **RR** y el mensaje de "realimentación".

Después del envío del "early packet", un flag "allow_early" es hecho igual a "FALSE", para prevenir que la misma fuente envíe dos "early packets"

consecutivos. Después que el próximo paquete **RTCP** normalmente programado es enviado, el parámetro anterior podría ser hecho igual a "**TRUE**" nuevamente. El próximo paquete **RTCP** es programado para $T_n = T_p + 2 \times T\text{-rr}$, en que T_n = tiempo del próximo paquete **RTCP**, T_p es el tiempo del último paquete programado regularmente, y $T\text{-rr}$ el intervalo **RTCP**.

4.2.31 Reglas para envío y recepción de paquetes RTCP (RFC 1889/3550)

En este ítem es explicado el procedimiento para enviar y lo que hacer cuando se recibe un paquete **RTCP**.

Una implementación *unicast* deberá mantener la "aleatoriedad" del intervalo de transmisión para evitar sincronización indeseada de instancias múltiples operando en el mismo ambiente, pero pueden omitirse los algoritmos de "*timer reconsideration*" y "*reverse reconsideration*" indicados a seguir.

Para ejecutar estas reglas, los participantes de la sesión deben tener disponibles, entre otros, los siguientes datos:

- **T_p** = El momento (*timestamp*), cuando el último paquete **RTCP** fue transmitido
- **T_c** = El tiempo (*time*), en el instante actual.
- **T_n** = El tiempo (*time*), programado de transmisión del próximo paquete **RTCP**.
- **$Pmembers$** = El número estimado de miembros de la sesión recalculado en el momento T_n .

- **Members** = La estimativa más actual del número de miembros de la sesión.
- **Senders** = Estimativa más actual del número de transmisores en la sesión.
- **RTCP_BW** = la banda RTCP estimada. Es decir, la banda total que será usada para los paquetes RTCP por todos los miembros de esta sesión, en octetos por segundo (*bytes*). Deberá ser especificada como una fracción del parámetro "banda de la sesión".
- **We_sent** = *flag* que es "TRUE" si la aplicación envió datos desde que el segundo informe anterior fue transmitido.
- **AVG-RTCP-size** = el tamaño medio del paquete RTCP compuesto, en octetos, por todos los paquetes RTCP enviados y recibidos por este participante. El tamaño incluye encabezados de la capa baja de los protocolos de transporte y de red (UDP y IP).
- **Initial** = *flag* que es "TRUE" si la aplicación aún no envió ningún paquete RTCP.

La mayoría de estos parámetros utiliza el "intervalo calculado" T para la transmisión de paquetes y será detallado en el ítem siguiente.

4.2.32 Cálculo del intervalo de transmisión del RTCP (RFC 1889/3550)

Para mantener escalabilidad, el intervalo medio entre paquetes de un participante de la sesión deberá escalar con el tamaño del grupo. Este

intervalo es llamado de "intervalo calculado" T , y es determinado de la siguiente forma:

$$T (Tr) = \frac{n(C)}{0,25(banda RTCP)} \quad 3.5$$
$$T (rx) = \frac{n(C)}{0.75(banda RTCP)}$$

Dónde:

$T (Tr)$: T de los transmisores.

$T (rx)$: T de los receptores.

- Si el número de transmisores es menor o igual que 25% de los miembros del grupo, el intervalo va depender si el participante es transmisor o no. Si es transmisor, la constante " C " es considerada como el tamaño medio del paquete **RTCP (AVG-RTCP-size)** dividido por 25% del ancho de banda **RTCP (RTCP_BW)**, y la constante " n " representa el número de transmisores. Si el participante es receptor, la constante " C " es considerada como el tamaño medio del paquete **RTCP** dividido por 75% del ancho de banda del **RTCP**. La constante " n " es considerada como el número de receptores (miembros - transmisores). Si el número de transmisores es mucho más grande de que 25%, transmisores y receptores son considerados conjuntamente.
- La constante " C " es calculada como el tamaño medio del paquete **RTCP** dividido por la banda total y " n " es el número total de miembros. Y, de acuerdo con el indicado en el ítem 1, un perfil **RTP** puede especificar que una banda **RTCP** puede ser explícitamente definida por 2 parámetros separados llamados " S " y " R ", tanto para aquellos participantes que son

transmisores cuanto para los que no lo son. En este caso, la fracción de 25% se convierte en $S/(S + R)$ y la fracción de 75% se convierte en $R/(S + R)$. Mire que si $R = 0$, el porcentaje de transmisores nunca será más grande que $S/(S + R)$ y la implementación debe evitar la división por cero;

- Si el participante aún no envió un paquete **RTCP** (la variable *Initial* es "TRUE") la constante **Tmin** es configurada en 2,5 segundos; de otra manera, será configurada en 5 segundos.

- El intervalo determinístico calculado **Td** es configurado para $Td = \max(Tmin, n \times C)$.

- El intervalo calculado, **T**, es fijado en un número uniformemente distribuido, entre 0,5 y 1,5 veces el intervalo calculado determinísticamente **Td**.

- El valor resultante de **T** es dividido por $(e - 1,5) = 1,21828$ para compensar la convergencia del algoritmo "timer reconsideration" para un valor de la banda **RTCP** abajo del valor medio.

Este procedimiento resulta en un intervalo al azar, pero que, en la media, destina en el mínimo 25% de la banda **RTCP** para los transmisores y el restante para los receptores. Si los transmisores constituyen más de un cuarto de los miembros, este procedimiento divide la banda igualmente entre todos los participantes.

4.2.33 Inicialización de los participantes de la sesión

Al unirse a sesión, el participante inicializa sus parámetros de la siguiente forma: **Tp** = 0, **Tc** = 0, *Senders* = 0, *Pmembers* = 1, *Members* = 1, *We_sent*

= "FALSE", **RTCP_BW** = a fracción especificada de la banda de la sesión, *Initial* = "TRUE", **AVG-RTCP-size** = al tamaño probable del primero paquete **RTCP** que la aplicación construirá después.

Con estos datos es calculado el intervalo T, después de esto, el primero paquete es programado para el tiempo $T_n = T$. Esto significa que un *timer* de transmisión es fijado y expira en el tiempo T (mire que la aplicación puede usar el foco que desear para implementar este *timer*).

El participante agrega su propio **SSRC** (*Synchronization Source*) en el cuadro de miembros.

4.2.34 Recibiendo un paquete RTP o RTCP que no es del tipo "BYE"

Cuando un paquete **RTP** o **RTCP** es recibido de un participante cuyo **SSRC**, no está registrado en el cuadro de miembros, su **SSRC** es agregado al cuadro y el valor de los miembros es actualizado después del participante haber sido validado. El mismo procedimiento es seguido para cada **CSRC** (*Computer Security Resource Center*) cuando un paquete RTP es validado. Cuando un paquete RTP es recibido de un participante cuyo **SSRC** no está en el cuadro de transmisores, el **SSRC** es agregado al cuadro y el valor de los transmisores es actualizado.

Para cada paquete **RTCP** compuesto recibido, el valor del **AVG-RTCP-size** es actualizado de la siguiente forma:

$$AVG - RTCP - size (i) = \left(\frac{1}{16}\right) x packet - size + \left(\frac{15}{16}\right) x AVG - RTCP - size (i - 1) \quad 3.6$$

Dónde: *Packet-size* = tamaño del paquete **RTCP** recientemente recibido.

4.2.35 Recibiendo un paquete RTCP del tipo "BYE"

Si fuera recibido un paquete RTCP del tipo "BYE", el SSRC es verificado en el cuadro de miembros. Si está presente, la entrada es removida del cuadro y el valor de los miembros es actualizado. Después, el SSRC es verificado en el cuadro de transmisores. Si está presente, la entrada es removida del cuadro y el valor de los transmisores es actualizado.

4.2.36 Time Out de un SSRC

A cada cierto tiempo, los participantes deben verificar si algún de los otros participantes temporizó "time out". Para hacer esto, el participante calcula el intervalo determinístico T_d , (sin el factor al azar) para un receptor que está con $We_sent = "FALSE"$. Cualquier otro miembro de la sesión que no tenga enviado un paquete RTP o RTCP desde el tiempo $T_c - MT_d$ (M es el multiplicador *time out*, con *default* igual a 5 es considerado "timed out"), esto significa que su SSRC es eliminado de la lista de miembros y el cuadro *Members* es actualizado. Una verificación similar es efectuada en la lista de transmisores.

Cualquier miembro de la lista de transmisores que no tenga enviado un paquete RTP desde el tiempo $T_c - 2T$ (dentro de los 2 últimos intervalos de informes RTCP) es eliminado de la lista de transmisores y el cuadro es actualizado.

Si cualquier miembro es considerado "timed out", el algoritmo de reconsideración reversa, descrito arriba, debería ser ejecutado.

El participante debe efectuar esta verificación lo mínimo una vez por intervalo de transmisión RTCP.

4.2.37 Finalización del *timer* de transmisión

Cuando expira el *timer* del paquete de transmisión, el participante efectúa las siguientes operaciones:

- El intervalo de transmisión T es calculado de acuerdo con el que fue descrito en la sección 5.3.1 de la RFC 3550, incluido el factor al azar.
- Si $T_p + T \leq T_c$, es transmitido un paquete RTCP. $T_p = T_c$, y otro valor para T es calculado de acuerdo con el ítem arriba, $T_n = T_c + T$.

El *timer* de transmisión expira en el tiempo T_n . Si $T_p + T$ es más grande que T_c , $T_n = T_p + T$ y ningún paquete RTCP es transmitido.

- $Pmembers$ es configurado igual a $Members$.

Si un paquete RTCP es transmitido, el valor *Initial* = "FALSE". Consecuentemente, el valor de **AVGR-TCP-size** es actualizado de la siguiente forma:

$$\text{AVG - RTCP - size (i)} = \left(\frac{1}{16}\right) \times \text{packet - size} + \left(\frac{15}{16}\right) + \text{AVG - RTCP - size (i - 1)} \quad 3.8$$

Donde: *packet-size* es el tamaño del paquete RTCP recientemente transmitido.

4.2.38 Transmitiendo un paquete de tipo "BYE"

Cuando un participante desea terminar una sesión, transmite un paquete *BYE* para informar a los otros participantes de este hecho con el objetivo de evitar una avalancha de paquetes *BYE* cuando muchos participantes

desean dejar el sistema, un participante debe ejecutar el siguiente algoritmo se desea dejar el sistema cuando el número de miembros fuera más grande que 50.

Cuando el participante decide dejar el sistema, $T_p = T_c$, el tiempo en ese instante, *Members* y *Pmembers* son inicializados para 1, "*Initial*" es configurado en *We_sent* = "FALSE", *senders* es configurado para 0 y *AVG-RTCP-size* = tamaño del paquete compuesto *BYE*. El intervalo calculado *T* es calculado. El paquete *BYE* es programado para el tiempo $T_n = T_c + T$. Cada vez que un paquete *BYE* de otro participante es recibido, el campo *Members* es incrementado de un, sin importar si el participante existe en el cuadro de miembros o no, y cuando el muestreo de *SSRC* está siendo utilizado, independiente del *BYE SSRC* ser incluido en la muestra o no. *Members* no es incrementado cuando otros paquetes *RTCP* o *RTP* son recibidos, pero apenas cuando son paquetes *BYE*. Del mismo modo, *AVG-RTCP-size* es actualizado solamente cuando son recibidos paquetes *BYE*. *Senders* no es actualizado cuando llegan paquetes *RTP*, permanece en 0. A partir de este instante, la transmisión del paquete *BYE* sigue las reglas para transmitir un paquete *RTCP* regular, como indicado arriba. Este procedimiento permite que los paquetes *BYE* sean transmitidos inmediatamente, controlando su uso total de banda. En el peor de los casos, esto provocaría que paquetes de control *RTCP* usasen el doble de la banda normal (10%), 5% para paquetes *RTCP BYE* y 5% para paquetes de otro tipo.

Un participante que no desee esperar por el procedimiento indicado para la transmisión del paquete *BYE* puede dejar el grupo sin la transmisión de ningún paquete *BYE*. Probablemente, este participante será *timed out*, por los otros miembros del grupo.

Si el tamaño del grupo de miembros es estimado en menos de 50 cuando el participante decide salir, el participante puede enviar un paquete *BYE* inmediatamente o, si desear, puede ejecutar el algoritmo indicado en el ítem anterior. En cualquier de los casos, el participante que nunca envió un paquete RTP o RTCP no debe enviar un paquete *BYE* cuando fuera dejar el grupo.

4.2.39 Actualización de *We_sent*

La variable *We_sent* contiene el valor "*TRUE*", si el participante envió un paquete RTP recientemente, o un valor "*FALSE*", en el caso contrario. Si el participante envía un paquete RTP cuando *We_sent* está configurado como "*FALSE*", el se incluye sólo en el cuadro de transmisores y configura *We_sent* para "*TRUE*". El algoritmo de "reconsideración reversa" debería ser ejecutado para reducir el retraso antes de enviar un paquete SR. Cada vez que otro paquete RTP es enviado, el tiempo de transmisión de ese paquete es mantenido en el cuadro y el algoritmo normal de *time out* del transmisor es aplicado.

Si un paquete RTP no fue transmitido desde el tiempo $T_c - 2T$, el participante se elimina sólo del cuadro, disminuí el valor del mínimo de transmisores y configura *We_sent* para "*FALSE*".

4.2.40 Asignación de banda en el SDES

Esta especificación define varios ítems SDES adicionales en el ítem *CNAME*, tales como *NAME* (nombre personal) y *EMAIL* (dirección de correo electrónico). Ella también proporciona un medio de definir nuevos tipos de paquetes RTCP para aplicaciones específicas. Todavía, será preciso tener mucho cuidado en la hora de asignar control de banda para estas aplicaciones, ya que puede reducir la tasa en que los informes de recepción y *CNAME* son enviados, perjudicando, así, la performance del protocolo.

Se recomienda que no más de 20% de la banda RTCP asignada para un único participante sea usada para transportar la información adicional. Por consiguiente, no se pretende que todos los ítems SDES sean incluidos en cada aplicación. Para aquellos que son incluidos debería ser asignada una fracción de la banda, de acuerdo con su utilidad y, en vez de estimar esas fracciones dinámicamente, es recomendable que los porcentajes sean estimados estáticamente dentro de los intervalos de los informes, basados en el tamaño típico de un ítem.

Por ejemplo, una aplicación puede ser designada para enviar solamente *CNAME*, *NAME* y *EMAIL* y nada más. Al *NAME* podría ser dada mayor prioridad que en el *EMAIL*, porque el *NAME* podría ser enseñado continuamente en la *interface* de la aplicación del usuario, mientras que el *EMAIL* podría ser enseñado sólo cuando solicitado. Para cada intervalo

RTCP, un paquete RR y un paquete SDES con el ítem *CNAME* sería enviado.

4.2.41 Informes SR y RR

Los receptores RTP proporcionan realimentación de la calidad de la recepción utilizando informes de paquetes RTCP que pueden ser de dos formas, dependiendo si el receptor también es un transmisor o si es solamente receptor. La única diferencia entre las formas del informe de transmisor (SR) y el informe de recepción (RR), además del código del tipo, es que el informe de transmisor incluye una sesión de información del transmisor de 20 bytes, a ser usada por los transmisores activos. El SR es generado si un *síte* envía cualquier paquete de datos durante el intervalo de creación del último informe o informe previo, caso contrario, un informe RR es creado.

Ambas las formas, SR y RR, incluyen cero o más bloques en el informe de recepción, un para cada una de las fuentes de sincronismo de las cuales este receptor recibió paquetes de datos RTP desde el último informe.

Cada bloque del informe de recepción proporciona estadísticas sobre los datos recibidos de la fuente indicada en ese mismo bloque. Como 31 bloques de informe RR caben en un paquete SR o RR, los paquetes RR adicionales deberán ser apilados después de los paquetes enviados SR o RR, segundo las necesidades.

Los paquetes RR adicionales deberían ser apilados después del SR o RR inicial, de acuerdo con la necesidad de contener los informes de recepción

para todas las fuentes presentes durante el intervalo desde el último informe.

Si existen muchas fuentes para completar todos los informes de paquetes RR en un paquete RTCP compuesto sin exceder al MTU (*Maximum Transmission Unit*) del enlace de la red, entonces solamente el subconjunto que cabe en una MTU deberá ser incluido en cada intervalo. Los subconjuntos deberán ser seleccionados de la forma *round-robin* por medio de intervalos múltiples de forma que todas las fuentes sean informadas.

4.2.42 Ejemplo de aplicación del cálculo de banda del RTCP

El RTCP intenta limitar su tráfico a 5% de la largura de banda de la sesión. Por ejemplo, suponga que una fuente está se comunicando con otra a una velocidad de 95200 bps (G.711). Entonces, el RTCP tiende a limitar su tráfico a $0,05 \times 95200 = 4760$ bps. De este valor, 75% (3570 bps) son asignados para los receptores y el restante es asignado para el transmisor. Los 3570 bps son igualmente distribuidos entre los receptores. Si fueran 2 receptores, cada un recibirá $3570 \text{ bps} / 2 = 1785$ bps. El transmisor utiliza 1190 bps para transmitir tráfico RTCP. De esta forma, los participantes determinan el período de transmisión de paquetes RTCP calculando el paquete RTCP medio, durante la sesión entera, y dividiendo por su tasa asignada.

En resumen:

- El período para transmisión de paquetes RTCP por un transmisor es:

$$T1 = \frac{\text{número de transmisores}}{25 \times 0,05 \text{ banda da sección}} (AVG - RTCP - \text{packet size}) \quad 4.9$$

- El período para transmisión de paquetes RTCP por un receptor es:

$$T2 = \frac{\text{número de receptores}}{75 \times 0,05 \text{ banda da sección}} (AVG - RTCP - \text{packet size}) \quad 4.10$$

- Para **AVG-RTCP-packet size = 238 bytes**:

$$T1 = \frac{1 \times 288 \text{ bytes} \times 8 \text{ bits/bytes}}{0,25 \times 0,05 \times 95200} = 1,6 \text{ segundos(transmisores)} \quad 4.11$$

$$T2 = \frac{1 \times 288 \text{ bytes} \times 8 \text{ bits/bytes}}{0,75 \times 0,05 \times 95200} = 0,42 \text{ segundos(receptores)} \quad 4.12$$

4.2.43 Formato de los paquetes RTCP

En general, el RTP y el RTCP tienen formato parecido. En el RTCP los formatos varían, entre otras cosas, en función del objetivo de cada uno de ellos; específicamente con relación al tipo de "payload", están definidos 5 tipos: SR = 200, RR = 201, SDES = 202, BYE = 203 y APP (*Application Defined Packet*) = 204. Los formatos de mayor interés de este tema son básicamente 2, el SR y el RR y permiten, que cuando son implementados, proporcionan las siguientes estadísticas:

- Atraso del paquete fin-a-fin.
- Número de paquetes recibidos.
- Tasa de paquetes perdidos.
- Número de paquetes esperados.
- Fracción de paquetes perdidos.
- Tasa de pérdida de paquetes por segundo.
- Validación estadística de las estimativas de pérdida de paquetes.
- *Throughput* aparente disponible para los receptores.

- *Interarrival jitter.*

En las tablas 4.11, 4.12, 4.13 y 4.14 están detallados los formatos de los paquetes RTP, SR, RR y SDES respectivamente.

TABLA 4.16

FORMATO DEL PAQUETE RTP

V	P	X	CC	M	PT	Sequence number
Timestamp						
Synchronization source (SSRC) identifier						
Contributing source (SSRC 1) identifier						
...						
Contributing source (SSRC n) identifier						
PAYLOAD						

FUENTE: PROTOCOLO RTP

TABLA 4.17

FORMATO DEL REPORTE DE TRANSMISIÓN

V	P	X	RC	PT - 200	Length
SSRC of the sender					
NTP timestamp (MSB)					
NTP timestamp (LSB)					
RTP timestamp					
Sender's packet count					
Sender's octet count					
First reception report block (SSRC 1)					
...					
Lost reception report block (SSRC n)					

FUENTE: PROTOCOLO RTP

TABLA 4.18
FORMATO DEL REPORTE DE RECEPCIÓN

V	P	RC	PT - 201	Length
SSRC of the sender				
SSRC of the first source				
Fract. Lost			Cum. nº of packet lost	
Ext. Highest sequence number received				
Interarrival jitter estimate				
Lost sender report timestamp (LSR)				
Delay since lost sender report (DLSR)				
...				
Lost reception report block				

FUENTE: PROTOCOLO RTP

TABLA 4.19
FORMATO DE DESCRIPCIÓN DE LA FUENTE

V	P	SC	PT - 202	Length
SSRC/CSRC of the sender				
Type			Length	Text
Fract. Lost			Cum. nº of packet lost	
Text continued				
...				
Lost chunk				

FUENTE: PROTOCOLO RTP

4.2.44 Cálculo de las métricas de QoS del RTCP

- **Cálculo del atraso fin-a-fin (RTT)**

De acuerdo con la RFC 3550 (Sección 6.4.1) el RTT es calculado por

$$\text{la Ecuación: } D = A - \text{LSR} - \text{DLSR} \quad 4.13$$

- **D = RTT**
- **A = instante de tiempo en que el RR es recibido por el sender**

- **LSR** (*Last Sender Report*) = last timestamp received from this source, tomado del RR
- **DLSR** (*Delay since Last Sender Report*) = delay since the receiving the LSR from this source, tomado del RR

4.2.45 Ejemplo de cálculo de atraso fin a fin (RTT)

Datos colectados con *Wireshark*:

TIMESTAMP

[10 NOV 1995 11:33:25.125 UTC]

> *SENDER REPORT*

> **MSW**: 0xb44db705 = 3024992005 **seg**

> **LSW**: 0x20000000 = **LSW/16** = 1.25 **seg**

> *Sender's packet count* = 2

> *Sender's octet count* = 320

> *RECEIVER REPORT*

[10 NOV 1995 11:33:36.5 UTC]

> *Cummulative number of packets lost* = 0

> *Extended highest sequence number received* = 0

> *Interarrival jitter* = 0

> *Last SR timestamp* (**LSR** = 46853.125 **seg.**)

> *Delay since last SR timestamp* (**DLSR**) = 5.250s

Utilizando la Ecuación:

$$D = A - LSR - DLSR$$

4.14

- **NTP timestamp del SR (10 NOV 1995 11:33:36.125) = 3024992005.125 seg**
- **LSR = 46853.125 seg**
- **DLSR = 5.250 seg**
- **A = 46864.500 seg;**
- **RTT = 46864.500 - 46853.125 - 5.250 = 6.125 seg**
- **Retraso = 6.125 seg.**

Alternativamente:

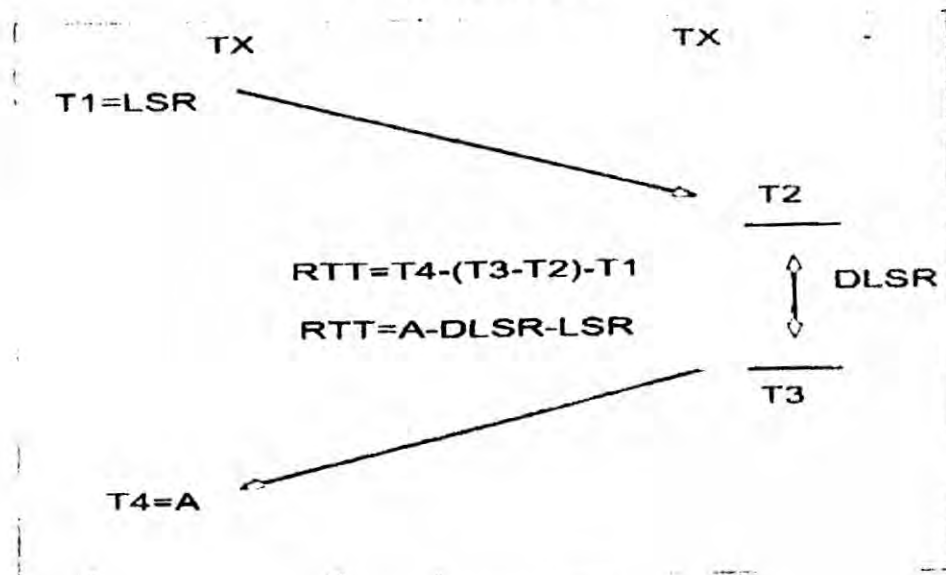
- **RTT = Timestamp del RR (UTC [Universal Time Coordinated])
Timestamp del último SR (UTC) – DLSR**
- **RTT = 11:33:36.500 - 11:33:25.125 – DLSR**
- **RTT = 11.375 - 5.25 = 6.125 seg. Packets lost = 3**
- **Packets lost = 3**
- **$IPLR = \frac{3}{50} = x 100 = 6$**

4.15

De forma equivalente:

Lá Fig. 4.5 resume gráficamente los parámetros envueltos en los cálculos arriba indicados.

FIGURA 4.13
CÁLCULO DEL IPDT



FUENTE: PROTOCOLO RTP

4.2.46 Cálculo del *Jitter* (IPDV)

La definición de "*jitter*" es bastante compleja. Algunos investigadores definen el *jitter* como la diferencia entre el máximo y el mínimo valor durante la transmisión de paquetes dentro de un cierto tiempo. Otros, como la máxima diferencia entre el retraso de la transmisión de dos paquetes consecutivos durante un período de tiempo. Aquí se utilizará la definición de la recomendación RFC 1889 (el desvío medio de la diferencia *D* del espaciamiento de 2 paquetes consecutivos en el lado receptor comparado al lado transmisor). El *jitter* ocurre cuando paquetes consecutivos encuentran diferentes retrasos en las filas de los dispositivos de la red. Los retrasos dependen del modelo de servicio de cada fila y del "tráfico interferente" en el enlace de transmisión.

Otra fuente de *jitter* puede ser el servidor, durante la codificación de los datos *real-time*. Pasado un tiempo significativo entre la codificación del paquete y el envío del paquete para la red, debido, por ejemplo, el caso de la **CPU** (*Central Processing Unit*) del servidor estar ocupada, estos cálculos del *jitter* pueden no ser válidos. El *jitter* generado por el servidor puede llevar a estimativas erróneas de las condiciones de la red. Testes de laboratorio indican que el *jitter* sólo no es una herramienta muy confiable para evaluar las condiciones de la red. Para obtener una estimativa confiable de las condiciones de la red, se recomienda que sea utilizado en conjunto con otros parámetros como, por ejemplo, "*packet loss*".

La estimación del *jitter* en el **RTCP** es hecha observando el tiempo de tránsito entre 2 paquetes consecutivos.

Si en los paquetes *i* y *j* son colocados los *timestamps* ***S_i*** y ***S_j***, cuando son enviados y son recibidos en los tiempos ***R_i*** y ***R_j*** respectivamente, entonces la Ecuación del cálculo $D = (R_j - R_i) - (S_j - S_i)$ es la diferencia de tiempos de tránsito en unidades de *timestamps*. Los *timestamps* de envío ***S_i*** y ***S_j*** son aproximados del *timestamp* del encabezado del paquete **RTCP** y ***R_i*** y ***R_j*** son obtenidos imprimiendo el tiempo local inmediatamente después de la recepción del paquete. Los receptores monitorean continuamente la variación de los valores de ***D*** y ponen a disposición en el campo "*interarrival time*" de los paquetes **RR** del **RTCP**. Dado que este parámetro es medido en unidades de *timestamp*, para convertirlo a unidades de tiempo es

necesario dividir este valor por el valor de la tasa de muestreo del *codec* utilizado.

Parámetro a ser considerado en esta investigación: *interarrival time*/tasa de muestreo del *codec*.

4.2.47 Cálculo de pérdida de paquetes (IPLR)

Un paquete es considerado perdido si:

- Jamás llegó a su destino;
- Si el paquete llegó dañado
- Si el paquete llegó fuera del tiempo previsto.

La métrica es calculada comparando los datos recibidos en los paquetes RR, en el campo: *Fraction lost* que es resultado de la comparación siguiente:

$$IPLR = \text{fraction lost (\%)} = \frac{\text{number of packets lost}}{\text{number of packets expected}} \times 100 \quad 4.16$$

Donde:

- *Number of packets lost* = *number of packets expected* - *number of packets received*.
- *Number of packets expected* = **EHSNR** - *initial sequence number*.
- **EHSNR** = *extended highest sequence number received*.
- **EHSNR** = *number of sequence number cycles* x 216 + *lost sequence number received*.

Parámetro a ser considerado en esta investigación:

$$IPLR (\%) = \text{fraction lost} \times 100 \quad 4.17$$

Ejemplo:

- *Sample period: 20 ms = 0,020 x 10⁻³ seg;*
- *Sample (packets) = 1000 ms*
- *20 ms = 50 packets/sec*
- *Packets lost = 3*

$$IPLR = \frac{3}{50} \times 100 = 6\% \quad 4.18$$

4.2.48 Algoritmo de monitoreo y control de QoS

En esta Sección se detalla el procedimiento y construcción del algoritmo propuesto.

En cada instante que el servidor recibe informes **SR** y/o **RR**, la red será analizada a través de los campos que guardan relación con el cálculo de las métricas **IPDT**, **IPDV** e **IPLR**. A partir de estas métricas, serán calculados los valores de **QoS** de la red y, posteriormente, comparados con los límites pre-definidos para determinar el estado de la red y, a continuación, adoptar las acciones para la restauración correspondiente. La **QoS** será calculada con la Ecuación 4.1. Sabiendo también que las pérdidas de los paquetes son originadas por dos motivos principales: 1) *overflow* de *buffer* en el enlace congestionado; 2) debido a *bits* errados, y que en la mayoría de las redes la probabilidad de existencia de *bits* errados es muy pequeña, suponemos que la pérdida de paquetes ocurre principalmente por congestión, en esas circunstancias, una reducción rápida de la banda utilizada podría reducir la pérdida de grande cantidad de paquetes, por tanto, la degradación de la calidad de la comunicación.

4.2.49 Sistema de QoS

El proceso total de monitoreo y control, es visualizado de la siguiente forma: los datos recibidos en los informes del **RTCP** serán utilizados como punto de partida para implementar un sistema de *software* que será llamado "Sistema de **QoS**, cuya función básica es la implementación de un mecanismo "tipo" control de flujo que actuará sobre la tasa de transmisión del "sender", ajustándola de acuerdo a los valores calculados de **QoS**. La nueva tasa de transmisión será implementada vía cambio de *codec* para un otro de menor banda. Alternativamente podrá ser detallada la implementación cambiando el tamaño del "payload" para reducir la banda. Los *codecs* de audio y vídeo soportan bien este tipo de ajuste, pues ellos usan *frame sizes* de longitud fija, de forma que la tasa de transmisión puede ser alterada fácilmente por medio del cambio del *codec*, "on the fly". Se estima que después de 2 **RR** recibidos con **QoS** válida (dentro de los límites), la operación regresará a condiciones normales de operación (regreso hasta el *codec* "default"), en la secuencia inversa de *fallback*. En resumen, el algoritmo de **QoS** a ser implementado funcionará de la forma indicada a continuación:

- Los receptores envían sus informes para el servidor de origen (fuente) en la forma y frecuencia normal de operación.
- El servidor (fuente) recibe esos informes y analiza las estadísticas en los campos mencionados a continuación y calcula las métricas **IPDT**, **IPDV**

e IPLR, después calcula la QoS de la red y, a continuación, los compara con los valores límites pre-definidos.

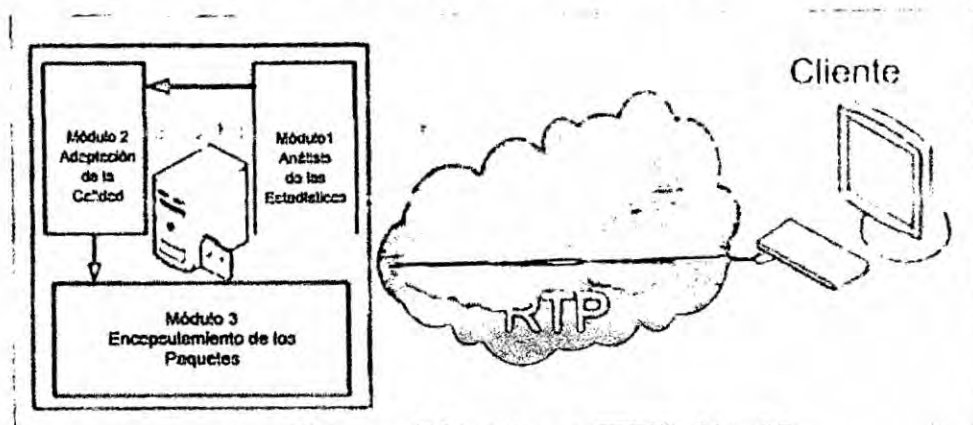
- Para el cálculo de la métrica IPDT serán utilizados los campos del RR, LSR (*timestamp* del último *sender report* recibido), DLSR (*delay* desde el último *sender report* recibido) y el parámetro "A" derivado del *timestamp* (NTP) del RR en el instante en que es recibido en el servidor y utilizando la Ecuación $RTT = A - DLSR - LSR$ calculase el RTT y, a continuación, calculase el $IPDT = (RTT/2)$ (retraso en un sentido).
- Para el cálculo de la métrica IPDV utilizase el valor directo del campo "*interarrival time*" dividido por la tasa de muestreo del *codec* utilizado.
- Para el cálculo de la métrica IPLR será utilizado el campo del RR "*cummulative number of packets lost*" y "*extended highest sequence number received*", de acuerdo con la Ecuación $IPLR (\%) = \frac{\text{cummulative number of packets lost}}{\text{extended highest sequence number received}} \times (100)$.
- Son considerados 3 posibles estados de la red: 1) Normal cuando los parámetros de QoS están bajo de los límites; 2) cargada cuando está próxima de los límites; 3) congestionada cuando está por encima de los límites.
- De acuerdo con la situación de la red, la aplicación multimedia podrá tener su banda aumentada (si regresara de una situación de congestión), mantenida (normal) o disminuida (si esta congestionada).

A continuación, son detalladas las características de cada uno de los 3 módulos componentes del Sistema de QoS para el modo de transmisión *unicast* (caso de la aplicación VoIP). La forma *multicast* podrá ser objeto de investigaciones futuras.

4.2.50 Módulos del Sistema de QoS

La propuesta considera la implementación de un mecanismo de transmisión adaptativo basado en el modelo cliente-servidor con todas las actividades efectuadas solamente en el servidor y compuesto de 3 módulos: 1) Módulo de análisis de las estadísticas; 2) módulo de adaptación de la calidad; 3) módulo de encapsulamiento y transmisión de los paquetes en la nueva velocidad (Fig. 4.14).

Figura. 4.14
MÓDULOS DEL SISTEMA DE QoS



FUENTE: PROPIA DEL AUTOR

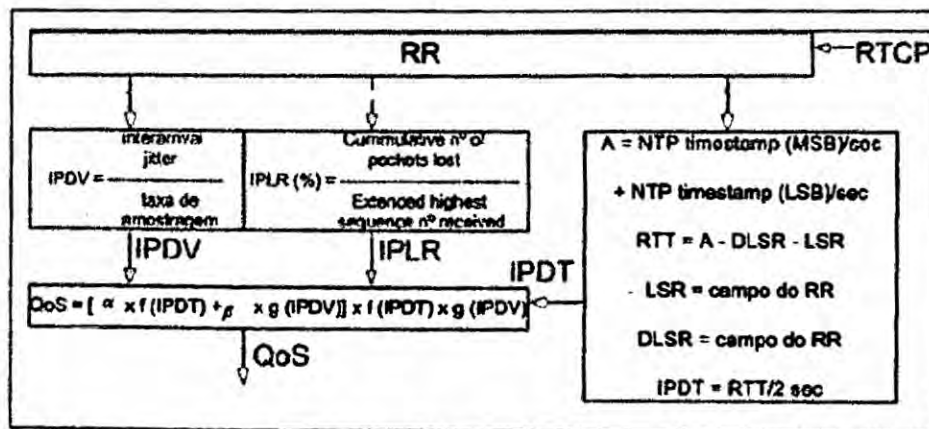
- **Módulo 1: Análisis de las estadísticas**

Su función es el monitoreo continuo de la red analizando los campos de los informes RR indicados en la Fig. 4.7 y envueltos en el cálculo de las

métricas de IPDT, IPDV o IPLR. Después de calcular las métricas indicadas, será usada la Ecuación 4.1 para determinar los valores de QoS que serán pasados para el módulo de adaptación de la calidad para determinar las condiciones de la red.

FIGURA. 4.15

MÓDULO 1



FUENTE: PROPIA DEL AUTOR

* En el cálculo de QoS normalmente son considerados sólo las métricas más significativas IPDT e IPDV. El IPLR cuando aumenta es considerado como inicio de una situación de congestión. (Indicación de inicio de congestión).

• **Módulo 2: Adaptación de la calidad**

Es responsable por la identificación de las condiciones de la red a partir de la calidad de la voz en ese instante.

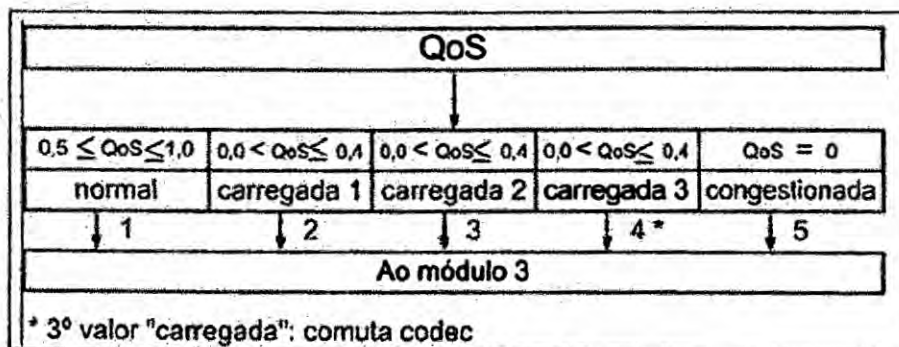
Este módulo está indicado en la Fig. 4.8 y recibe los valores de QoS del módulo 1 determinando el estado de la red dentro de una de las 3

alternativas siguientes: 1) Normal; 2) cargada; 3) congestionada.

Estos estados de la red son pasados para el módulo 3.

FIGURA. 4.16

MÓDULO 2



FUENTE: PROPIA DEL AUTOR

Si la opción "cargada" es aplicable, espera por un tercer valor si este fuera de la misma calidad (cargada 3) conmutará para un *codec* de menor banda. Caso contrario, mantendrá su condición en curso. Esto visa reducir efectos transitorios u oscilaciones en la red.

- **Módulo 3: Encapsulamiento de paquetes**

En función del estado de la red, este módulo indicado en la Fig. 4.17 actúa de la siguiente manera: 1) Mantiene el *codec* actual; 2) aumenta para un *codec* de mayor banda; 3) o disminuye (conmuta para un *codec* de menor banda).

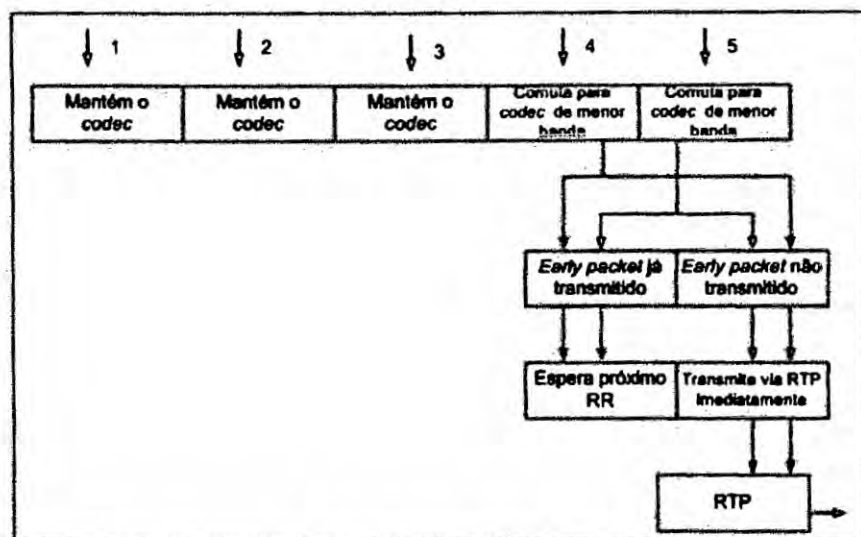
De forma general el módulo funciona de la siguiente manera:

- Del módulo 2, recibe el valor del campo 1 (normal) significando que debe mantener el *codec* actual.

- De los campos 2 y 3 puede recibir los valores 1, 2 y 3 significando que debe mantener el *codec* en curso.
- Recibiendo un valor 4 (congestionada 3) deberá conmutar para un *codec* de menor banda, después de la confirmación del *early packet*. Recibiendo un valor mejor que los anteriores, deberá mantener el *codec* en curso.
- Recibiendo el valor 5 deberá efectuar la conmutación para un *codec* de menor banda inmediatamente (después de la confirmación del *early packet*).

Nota: La conmutación para un *codec* de menor banda implica en la verificación previa del módulo de la transmisión, si ya efectuada o no la transmisión de un *early-packet*. Si ya fue transmitido deberá esperar por el siguiente RR, caso contrario, será enviado inmediatamente.

FIGURA. 4.17
MÓDULO 3



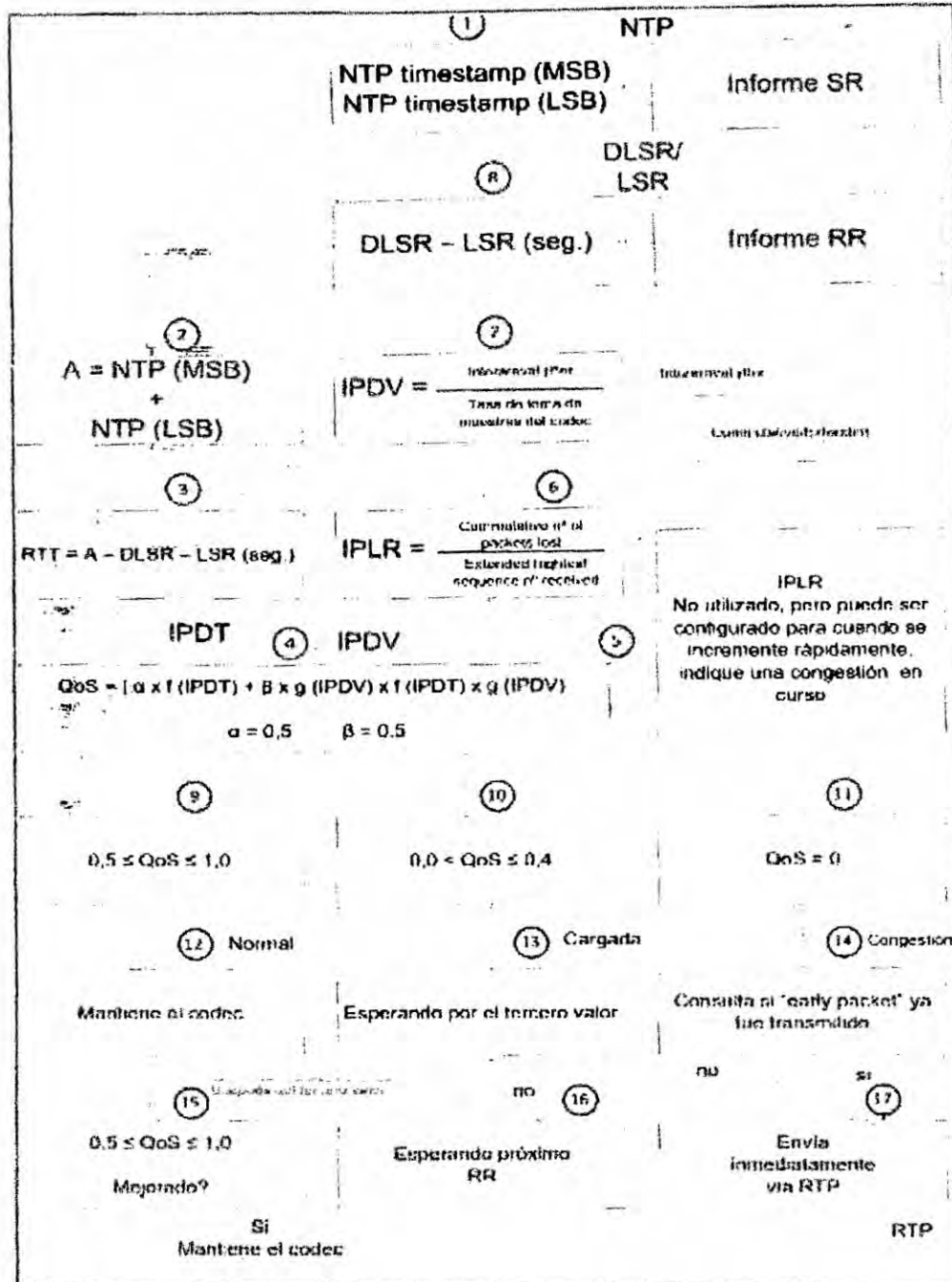
FUENTE: PROPIA DEL AUTOR

4.2.51 Codecs utilizados en este estudio

En esta investigación están siendo utilizados los siguientes *codecs*: 1) G.711; 2) G.729; 3) **GSM**; 4) G.723.1 A (**ACELP**). Esto representa una banda (utilizando protocolo de camada 2 **PPP** [*Point-to-Point Protocol*] o *frame relay*) de 81144 - 82800 **bps** para G.711, 45098 - 45600 **bps** para G.729, 31164 - 31800 **bps** para **GSM** y 17600 - 18190 **bps** para G.723.1 A. Supongamos que inicialmente (*default*) será utilizado el *codec* G.711 (*toll quality*). La degradación de la calidad, medida por la **QoS** llevará hasta el "*fallback*" para G.729, después para **GSM** y, finalmente, para G.723.1 A, en esta secuencia.

Es bueno mencionar que la relevancia de esta investigación se basa en el procedimiento propuesto y no necesariamente en los *codecs* aquí utilizados, una vez que otros *codecs* podrán ser escogidos. Se espera que después de transcurridos 2 informes consecutivos con métricas dentro de los límites, será iniciado el regreso para el *codec* "*default*" de la comunicación.

FIGURA 4.18
MÓDULO DE QoS COMPLETO



FUENTE: PROPIA DEL AUTOR

La tabla 4.15 indica, de arriba hacia abajo, la secuencia de *fallback* prevista.

Tabla 4.20
SECUENCIA DE FALLBACK

Codec		Banda (bps)	Posición
1	G.711	82800	Default
2	G.729	45600	↙
3	GSM	31800	↙
4	G.723.1	18190	↙

FUENTE: PROPIA DEL AUTOR

4.2.52 Ejemplo de cálculo de la banda necesaria y mínima para el codec G.711

La banda necesaria con todos los encabezados para el protocolo *frame relay* es calculada de la siguiente manera:

- *Sample period* = 20 ms.
- *Packets/sec* = 50.
- *Payload* = 160 bytes.
- *Overhead* para protocolo *frame relay* = 47 bytes.
- Banda nominal necesaria = 82800 bps.
- Banda mínima para un IPLR de 3% = $0,03 \times 50 \text{ packets/sec} = 1,5 \text{ packets}$ (aproximado a 2 packets).

Por tanto 2 paquetes representan:

$$2 \times (160 + 47) = 2 \times 207 = 414 \text{ bytes} \times 8 = 3312 \text{ bps} \quad (4.19)$$

Banda mínima necesaria: $82800 - 3312 = 82488 \text{ bps}$.

4.2.53 Algoritmo propuesto

El pseudo code del algoritmo.

```
#include <iostream.h>
#include <stdio.h>
#include <iomanip.h>
#include <string.h> void main ()
{
int i, x = 0, valor, y = 0, z = 1, cn = 0, cc = 0, cg = 0, p = 0, r = 3, sw = 0,
inicia = 0;
double qs, coded [] = {82800,45600,31800,18190};
double msb, lsb, a, dlsr, lsr, rtt, ipdt, ipdv, interarrival, _pdt, gipdv;
char qualidade [50];
strcpy (qualidade, "");
bool seguir = true; char resp;
while (seguir)
{
cout<<"ingressar MSB"
cin>>msb
cout<<"ingressar LSB"
cin>>lsb
cout<<"ingressar DLSR"
cin>>dlsr
cout<<"ingressar LSR"
```

```

cin>_>lsr
cout<_<"ingressar a"
cin>_>a;
cout<_<"ingressar interarrival";
cin>_>interarrival;
rtt = a-dlsr-lsr;
ipdt = rtt/2;
printf ("O valor de a %10.10f ",a);
cout<_<endl;
printf ("valor de rtt %10.10f ",rtt);
cout<_<endl;
printf ("valor de ipdt %10.10f ",ipdt);
cout<_<endl;
ipdv = interarrival/8000;
_pdt = (150-ipdt)/150;
gipdv = (50-ipdv)/50;
qs = (0,5 * _pdt + 0,5 * gipdv) * (_pdt * gipdv);
printf ("O valor de qs %10.10f ", qs);
cout<_<endl;
// Processo
if (qs> = 0,5 && qs< = 1);
{
Strcpy (qualidade, "normal:");

```

```

Valor = coded [0]; i = 0; cn = 0; sw = 0;
}
if (qs>0 && qs<= 0,4);
{
Strcpy (qualidade, "carregada:");
l = 2; cc = 1;
if (inicia==1)
{ cn = 0; sw = 0;}
}
if (qs==0)
{
Strcpy (qualidade, "congestionada:");
l = 3; cg = 1; sw = 0;
}
if (sw==0) // sw=0, baixando
{
if (i==0) // normal
{
if (cc==1)
valor = coded [x];
}
if (i==2) // carregada anterior foi normal
{

```

```

if (cn==0)
{
Valor = coded [x];
Cn =1; inicia = 0; r = 3; p = 0;
}
else
{
Valor = codec [x];
if (p==1)
{
X = x + 1;
Valor = coded [x];
P =-1;
}
P = p + 1;
If (x>= 3)
{ valor = coded [3]; sw = 1; x = 3; cg = 1;}
}
}
if (i==3) // congestionada e anterior foi normal ou carregada
{
if (cn==0 || cc==1)
{

```

```
x++;  
Valor = coded [x];  
if (x >= 3)  
{Valor = codec [3]; sw = 1; x = 3;}  
}  
}  
}  
else // retrocede  
{  
If (cg==1)  
{  
Valor = coded [3];  
Cg = 0;  
}  
else  
{  
Valor = coded [r];  
if (z==2)  
{  
If (r==0)  
{valor = coded [0];}  
else  
{
```

```

R = r - 1;
Valor = coded [r];
Z = 0;}
}
z++;
if (r==0)
{inicia =1; x = r; sw = 0;}
}
X = r;
}
cout<<_< qualidade;
cout<<_<valor;
cout<<_<"\n";
cout<<_<"Deseja seguir s/n:";
cin>>_>resp;
if (resp == 's')
seguir = true;
else
seguir = false;
}
cout<<_<endl;
}

```

4.2.54 Experiencias efectuadas

4.2.55 QoS por medio de simulación

En este ítem se realizó un estudio del comportamiento de la **QoS** por medio de simulación, utilizando la herramienta de *software* **OPNET MODELER** versión académica 17.5A y la topología indicada en la Fig. 4.11. **OPNET MODELER** es un simulador de red de la Empresa Americana **RIVERBED Technologies** que posee muchos recursos para simulación de redes. Esta versión es ofrecida libremente para fines académicos.

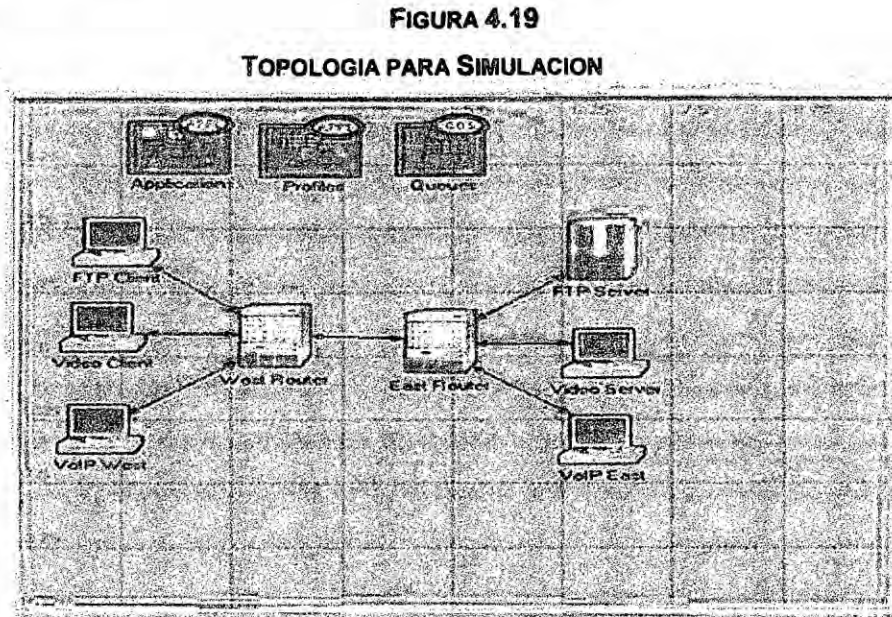
Esta experiencia de simulación tiene por objetivo verificar la influencia del tráfico de datos TCP y (**UDP**) sobre los parámetros de **QoS** (**THRU**, **IPLR**, **IPDT** e **IPDV**) de un tráfico **VoIP**. Para esto, las características de las aplicaciones utilizadas fueron las siguientes:

- Aplicación **VoIP**: voz operando con los siguientes *codecs*: a) G.711, 20 ms de muestreo, 1 *frame* por paquete; b) G.729, 10 ms, 1 *frame* por paquete; c) **GSM**, 20 ms, 1 *frame* por paquete; d) G.723.1 A, 30 ms, 1 *frame* por paquete.
- Aplicación **FTP**: carga pesada (tamaño del archivo de 1.000.000 *bytes*).
- Aplicación de vídeo: (videoconferencia) de baja resolución (10 *frames* por cuadro).

Las simulaciones fueron hechas sobre un enlace WAN, protocolo PPP con tasa de transmisión variable entre 128 Kbps y 2 Mbps.

4.2.56 Topología para la simulación

La Fig. 4.19, muestra los equipos y la red de pruebas utilizada para las simulaciones.



FUENTE: PROPIA DEL AUTOR

4.2.57 Descripción general de las simulaciones

Las simulaciones fueron realizadas en 3 escenarios diferentes:

- Tráfico VoIP único: el tráfico VoIP fue simulado sólo, sin la presencia de ningún otro tráfico interferente en el enlace WAN (entre los routers West y East), para los 4 codecs utilizados y diferentes tasas de transmisión del enlace. Esas simulaciones corresponden a las mejores condiciones posibles de obtención de QoS (tráfico único);
- Tráfico VoIP en la presencia de un tráfico interferente TCP: El tráfico VoIP comparte el enlace con un tráfico interferente FTP (TCP), para las diferentes combinaciones de codecs y tasas de transmisión.

Los parámetros de **QoS** fueran determinados y la degradación de **QoS** constatada por medio de la Ecuación 4.1.

- Tráfico **VoIP** en la presencia de un tráfico interferente **UDP**: El tráfico **VoIP** comparte el enlace con un tráfico interferente de vídeo (**UDP**), para las diferentes combinaciones de *codecs* y tasas de transmisión. Los parámetros de **QoS** fueran determinados y la degradación de **QoS** constatada por medio de la Ecuación 4.1.

Los resultados de las simulaciones fueran resumidos por medio de tablas y gráficos proporcionando una visión bastante detallada de la degradación de **QoS**.

Este modelo empírico de la **QoS** auxiliará en el ajuste de los parámetros del algoritmo de restauración de la misma.

4.2.58 Experiencias reales

Las experiencias fueron efectuadas en dos grandes actividades:

A. Verificación del ambiente de pruebas en el cual fueran efectuadas las actividades de pruebas experimentales, y es compuesta de tres sub actividades:

- Validación de la red de pruebas (banda efectiva de la red local (**LAN**) utilizada).
- Validación de las velocidades configuradas en la simulación del enlace **WAN**.
- Validación del efecto del tráfico interferente **TCP** y **UDP** efectuado por simulación.

- Validación del algoritmo propuesto.
- La topología general de pruebas está indicada en la Fig. 4.20.

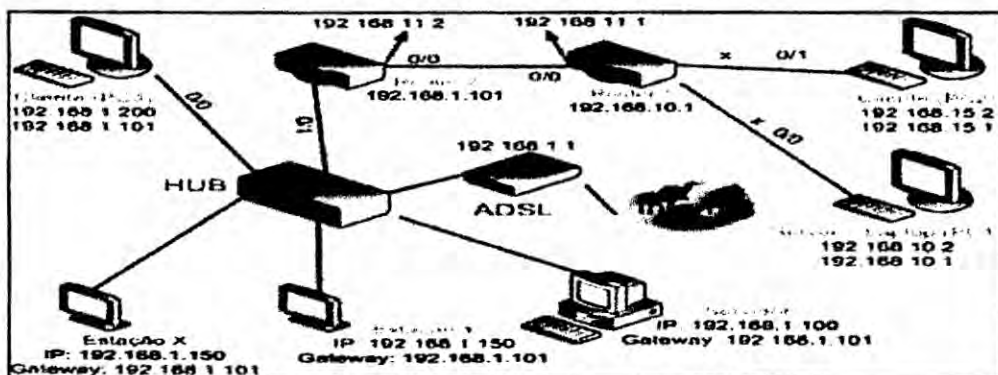
4.2.59 Validación de la banda efectiva de la red local

La primera actividad realizada fue la verificación del ambiente en el cual será efectuada la validación de la propuesta, es decir, la verificación de la confiabilidad del ambiente sobre el cual serán efectuados las pruebas de validación, principalmente para evitar que los resultados obtenidos sean distorsionados (por ejemplo, si la red (LAN) funciona nominalmente a 10 Mbps se verifica si el *throughput* es efectivamente de 10 Mbps o próximo). Fue utilizada la red local indicada en la Fig. 4.20 (topología general de pruebas). El *software* IPERF con el protocolo TCP mide la banda disponible en la red entre el servidor (192.168.1.100) y las estaciones x o y los resultados están indicados en la tabla 4.16

Los comandos del IPERF para esta experiencia son:

- Servidor: Iperf -s -i 1
- Cliente: Iperf -c (IP do servidor) -i 1

FIGURA. 4.20
TOPOLOGÍA GENERAL PARA PRUEBAS



FUENTE: PROPIA DEL AUTOR

La tabla 4.16 muestra los resultados obtenidos y la Fig. 4.12 la topología y los equipos utilizados.

TABLA 4.16
BANDA EFECTIVA DE LA RED LOCAL

Ventana tx – TCP (Kbps)	ID	Intervalo (seg.)	Transferencia (MB)	Banda (Mbps)
3	[1956]	0,0 – 10,0	9,1	7,3
6	[1956]	0,0 – 10,0	9,1	7,3
8	[1956]	0,0 – 10,0	9,1	7,3
16	[1956]	0,0 – 10,0	9,1	7,3
32	[1956]	0,0 – 10,0	9,1	6,6 *

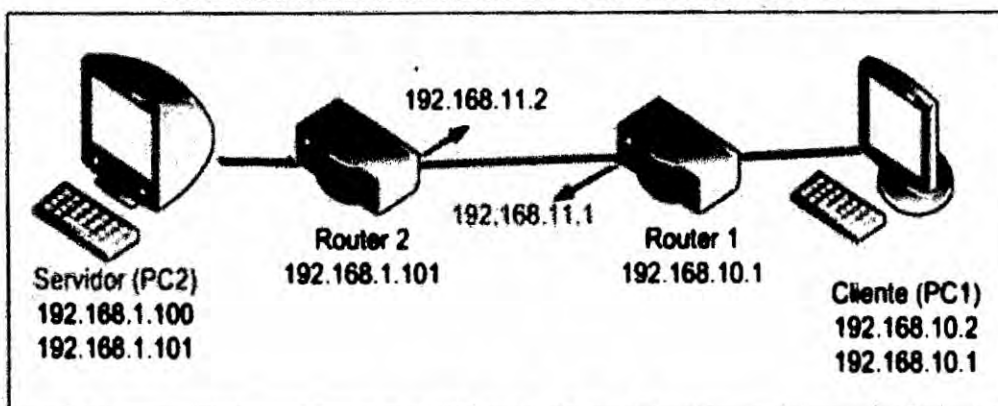
FUENTE: PROPIA DEL AUTOR

4.2.60 Validación de las velocidades del enlace WAN

En esta experiencia fueran comparadas las velocidades configuradas en los 2 *routers* y las velocidades medidas (con **IPERF**) entre los *routers* indicados en la Fig. 4.21. Las velocidades configuradas fueran: 128; 256; 512 Kbps; 1; 1,3 y DS1. Los resultados están indicados en la tabla 4.16.

FIGURA. 4.21

TOPOLOGÍA PARA VALIDACIÓN DE LA BANDA CONFIGURADA



FUENTE: PROPIA DEL AUTOR

TABLA 4. 21
BANDAS CONFIGURADAS Y MEDIDAS

Banda configurada (bps)	ID	Intervalo (seg.)	Transferencia (KB)	Banda medida (Kbps)
2000000	[1956]	0,0 – 10,0	2400	1910
1300000	[1956]	0,0 – 10,1	1600	1200
1000000	[1956]	0,0 – 10,1	1200	967
512000	[1956]	0,0 – 10,2	600	483
256000	[1956]	0,0 – 10,3	312	242
128000	[1956]	0,0 – 10,6	160	121
64000	[1956]	0,0 – 10,8	88	59,9

FUENTE: PROPIA DEL AUTOR

Las velocidades disponibles en los *routers* 2611 son:

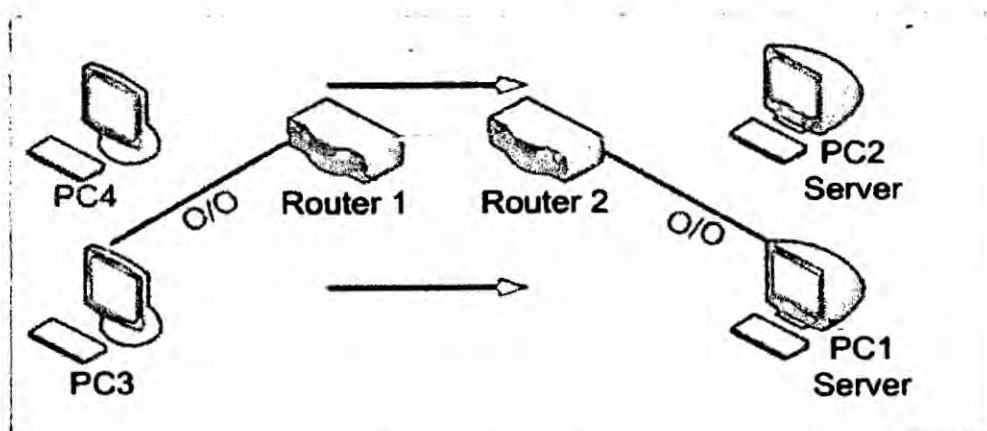
1200/2400/4800/9600/14400/19200/28800/32000/38400/56000/57600/64000/72000/1152000/125000/128000/148000/192000/250000/256000/384000/500000/512000/768000/8000000/1000000/1300000/2000000 Kbps.

4.2.61. Validación de las simulaciones

En esta experiencia son validadas las simulaciones con tráfico interferente cuando se encaminan dos flujos simultáneos (voz y TCP) y (voz y UDP), en las velocidades ya indicadas y para cada uno de los *codecs* estudiados. La topología de testes está indicada en la Fig. 4.22 separada de la Fig. 4.20 para fines de una mejor visualización.

FIGURA. 4.22

TOPOLOGÍA PARA VALIDACIÓN DEL TRÁFICO INTERFERENTE



FUENTE: PROPIA DEL AUTOR

4.2.62 Flujo único en el enlace

Las tablas 4.20, 4.23, 4.26, 4.29, 4.32, 4.35 muestran los resultados de las pruebas efectuadas con equipos "reales" para el caso de un flujo único de voz.

4.2.63 Con flujo interferente TCP

Las tablas 4.21, 4.24, 4.27, 4.30, 4.33, 4.36 muestran los resultados para el caso de un flujo de voz simultáneo con tráfico interferente TCP.

4.2.64 Con flujo interferente UDP

Las tablas 4.22, 4.25, 4.28, 4.31, 4.34, 4.37 muestran los resultados de las pruebas efectuados con equipos "reales" para el caso de un flujo de voz.

4.2.65 Validación del algoritmo propuesto

Este fue validado de forma manual, de acuerdo con lo indicado en los ítems 5.2.6 y 5.2.7.

4.2.66 Herramientas de pruebas

A seguir una breve descripción del *software* y *hardware* utilizado en las pruebas.

Wireshark

Es un *software* analizador de protocolos que soporta más de 700 tipos de protocolos. Es mantenido bajo licencia **GPL**: captura datos directamente de la red o de archivos previamente grabados. Trabaja en modo promiscuo o no promiscuo.

MODELER RIVERBED Versión Académica

El *software* **Opnet MODELER** es un simulador de red creado por el MIT y comercializado por la empresa americana RIVERBED *Technologies Inc.*, que posee grandes recursos para simulación de redes de Comunicaciones. Usamos la versión académica 17.5 A, que es distribuida libremente para fines académicos.

- **Iperf**

Iperf es una herramienta (*software*) para medida de la banda máxima **TCP/UDP** y permite sintonizar varios parámetros y características **UDP**. Reporta medidas de banda, variación de retraso y pérdida de datagramas. Es mantenido por la Universidad de *Illinois* bajo **GPL** (*General Public License*).

- **SJPhone**

Es un *software* aplicativo que implementa **VoIP** en las estaciones de teste. Es fabricado por la empresa **SJLABS** de *Ohio, USA*. Incluye los siguientes

codecs: **G.711, G.729, GSM e ILBC**. Es un *software* propietario y precisa ser instalado en todas las estaciones que usarán **VoIP**.

- **Hardware**

Las microcomputadoras usadas en la validación tienen las siguientes características: compatibles **PC**, con memoria **Ram** de **768 Mb**, sistema operativo *Windows XP sp2*, **CPU AMD Athlon** de **1.25 Ghz**. Los *routers* marca *Cisco 2611*, con dos puertos *Ethernet* de **10 Mbps ETH0 y ETH1**, puerto serial **WAN** de hasta **2 Mbps** con protocolo **PPP**. La *laptop* es un *Pentium III, Compac* modelo *Armada V300*, con **OS Windows XP packet 2**, **256 Mbps de Ram**, **clock** de **399 Mhz** y procesador *Intel Celeron*.

4.3 Metodica de cada momento de la investigación

- La metódica de esta investigación ha sido la siguiente:
 - Observación y entendimiento del problema y sus parámetros
 - Revisión de los documentos sobre el asunto.
 - Identificación y selección de los parámetros que serán analizados
 - Identificación de la solución a ser propuesta. Ventajas y desventajas
- Implementación y presentación de la investigación.

4.4 Población y muestra

No aplicable.

4.5 Técnicas e instrumentos de recolección de datos

La recolección de datos fue hecha de "*papers*", periódicos, revistas especializadas etc., En general, fue efectuada una análisis documental. Los

instrumentos de recolección de datos fueron microcomputadores y material de informática vía simulación y mediciones de pruebas físicas.

4.6 Plan de análisis estadístico de datos

Los datos obtenidos fueron clasificados por tipo de prueba, velocidad, tipo de *codec*, métrica y servicio.

CAPÍTULO V

5. RESULTADOS

5.1 Resultados de las simulaciones de voz y tráfico TCP y UDP.

5.1.1 Flujo único de voz en el enlace WAN de 128k- codec G.711

En los gráficos y tablas siguientes son presentados los valores y gráficos de las métricas de **QoS**, **IPDT**, **IPDV** e **IPLR**, resultantes de la simulación de tráfico "único" de voz y su comportamiento cuando el enlace WAN es usado en conjunto con tráfico interferente **TCP (FTP)** o **UDP (vídeo)**, para cada uno de los *codecs* de voz utilizados en esta investigación.

Se utilizó la topología de la Fig. 4.20 con el enlace **WAN** configurado de 128 Kbps hasta (1.5) Mbps. Los gráficos resultantes reflejan el comportamiento de las métricas hasta esta velocidad, aunque son presentados como ejemplo solo en la velocidad de 128 Kbps. En las tablas los datos están completos hasta 1.5 Mbps.

Los gráficos 5.1 a 5.11 como mencionado arriba muestran las características de las métricas de **QoS** de voz cuando los tres tipos de tráfico se encuentran compartiendo el enlace **WAN** (dos a dos, voz/TCP y voz/UDP), en la faja de 128 Kbps a **DS1 (1.544 Mbps)**. Estos gráficos, inician mostrando sus valores cuando trafican solas en el enlace **WAN**, sin interferencia ninguna y como es esperado, es la mejor situación para comunicaciones, ella representa nuestra referencia inicial para este estudio visto que en esta situación el tráfico de voz es el único, utilizando todo el ancho de banda disponible y después es mostrado cuando es transmitido

en conjunto con tráfico interferente **TCP** y cuando es transmitido en conjunto con tráfico **UDP**.

En cuanto a los títulos de los gráficos, el título "Métricas de **QoS** para Voz (**GSM**)" por ejemplo, sin ninguna extensión adicional indica que este es la referencia antes mencionada. Cuando es del tipo Voz (**GSM**) + **UDP**, significa que será evaluado el efecto del tráfico interferente **UDP** sobre los señales de voz con *codec* **GSM**. De la misma forma para el tráfico interferente **TCP** es mostrado en el grafico 5.2.

GRÁFICO 5.1
MÉTRICAS DE QoS DE VOZ (G.711) EN ENLACE DE 128 KBPS



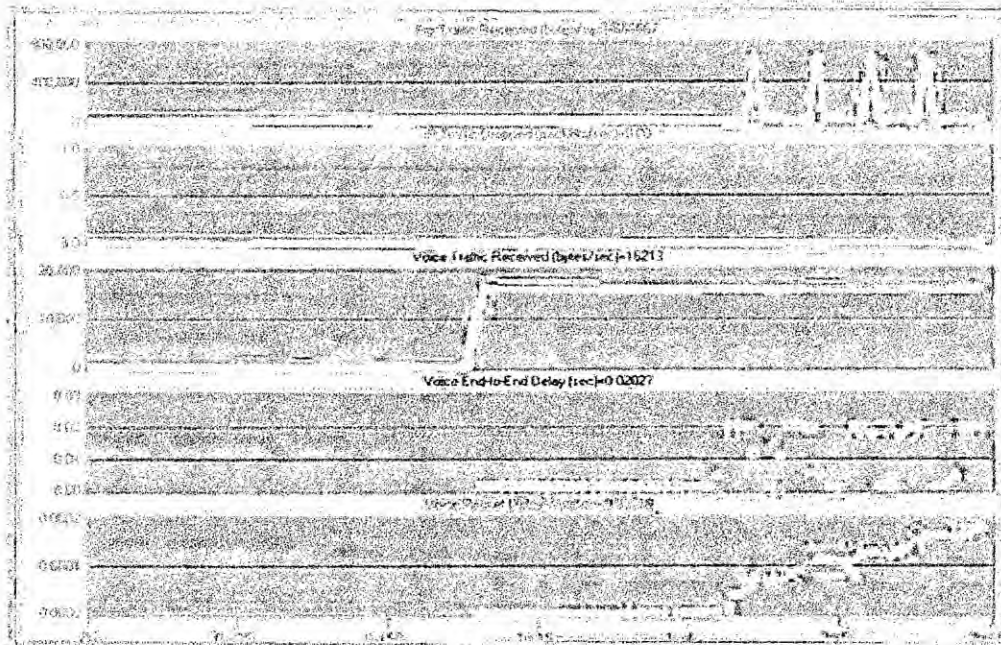
FUENTE: SIMULADOR

Observe el grafico 5.1 que el **IPDV** e **IPLR** son ambos iguales e/o próximos a cero como era de esperar, indicando que no existen pérdidas de paquetes y el atraso **E2E** (fin a fin) tampoco existe o su valor es insignificante.

5.1.2 Dos flujos (voz+tcp) en un enlace de 128 kbps

GRÁFICO 5.2

MÉTRICAS DE QOS PARA VOZ (G.711+TCP) EN ENLACE DE 128 KBPS



FUENTE: SIMULADOR

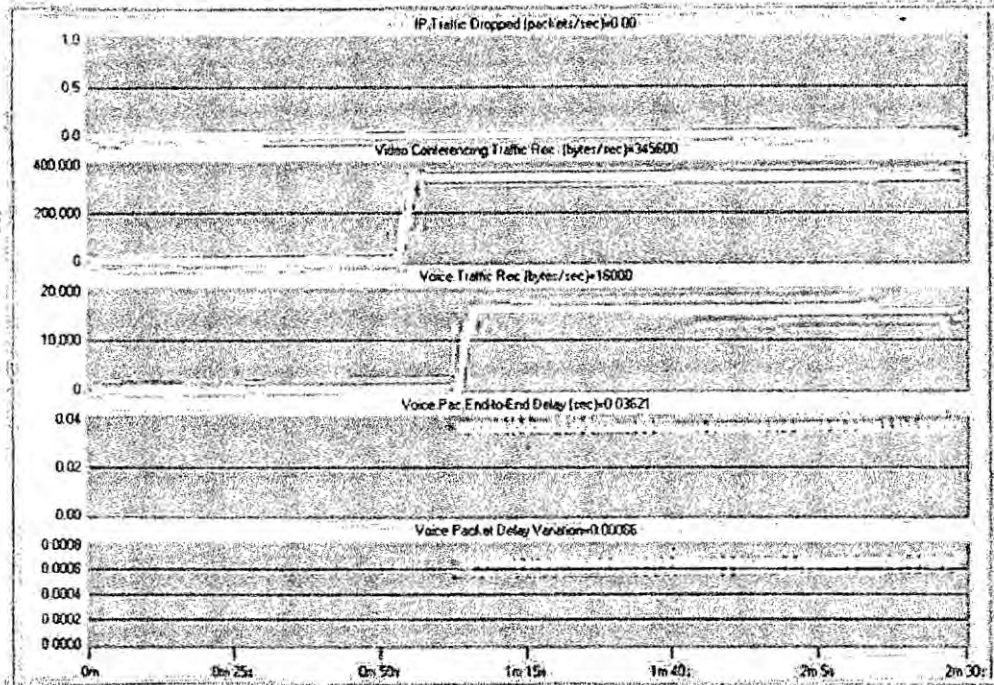
Los resultados muestran que: 1) No presenta pérdida de paquetes; 2) los valores del IPDT no exceden los límites establecidos; 3) el tráfico simultáneo ocupa toda la banda disponible en el canal WAN.

5.1.3 Dos flujos (Voz + UDP) en el enlace de 128 Kbps-codec G.711

El gráfico. 5.3 muestra los resultados cuando en el enlace de 128 Kbps se encamina vídeo + voz, simultáneamente (tráfico interferente: videoconferencia por ejemplo). Los gráficos restantes muestran el tráfico de veoz enviado de los 4 codecs en paquetes/seg.

GRÁFICO 5.3

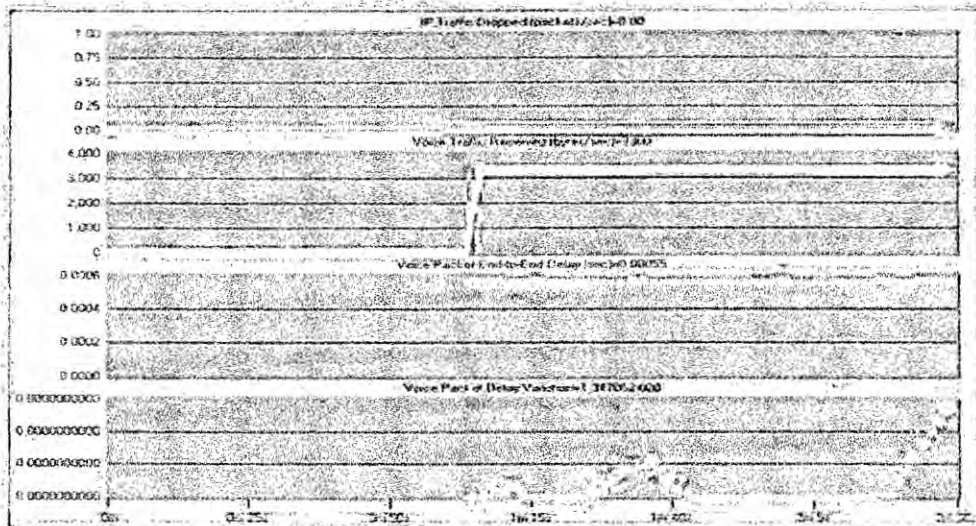
Métricas de QoS para Voz (G.711) + UDP EN UN ENLACE DE 128K



FUENTE: SIMULADOR

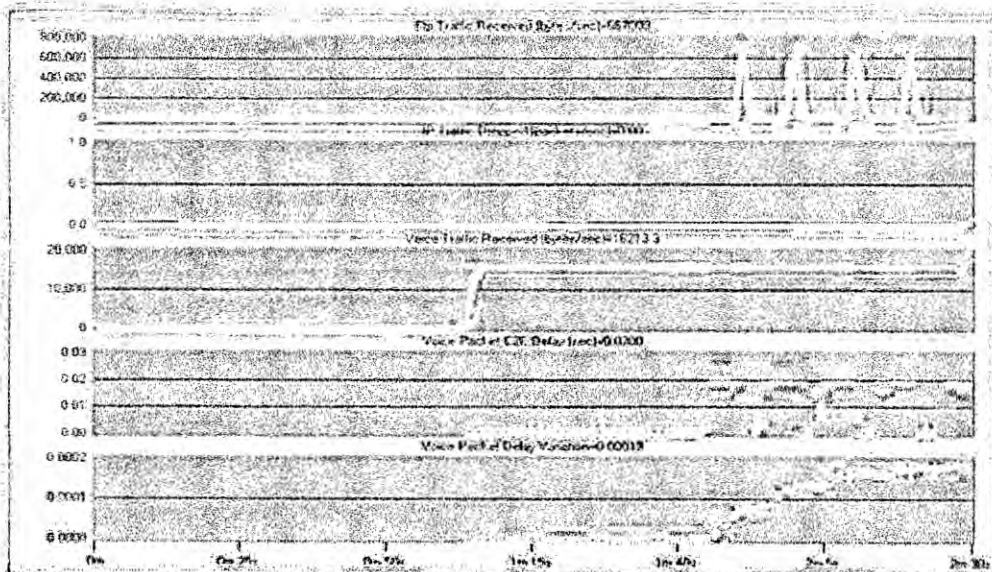
Los resultados de los graficos 5.1 a 5.3 muestran que para una banda de 128 Kbps, en la simulación efectuada y con las topologías indicadas, tiene las siguientes características. 1) No se registran pérdida de paquetes; 2) el retraso de voz se incrementa según indicado en la tabla 5.3; 3) el ancho de banda del enlace WAN ocupa toda la banda disponible cuando se comparte Voz (G.711) más tráfico interferente. Lo mismo es mostrado en los graficos 5.4 a 5.11 para los codecs G.729, GSM, G.723.1 A más el tráfico interferente.

GRÁFICO 5.4
MÉTRICAS DE QOS PARA VOZ (G 729) EN ENLACE DE 128 KBPS



FUENTE: SIMULADOR

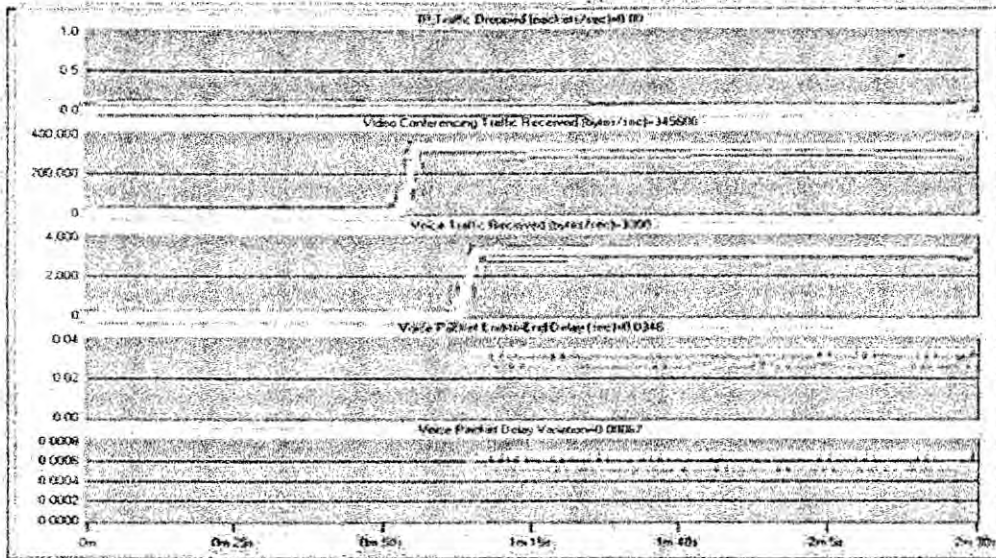
GRÁFICO 5.5
MÉTRICAS DE QOS PARA VOZ (G 729) + TCP EN ENLACE DE 128 KBPS.



FUENTE: SIMULADOR

GRÁFICO 5.6

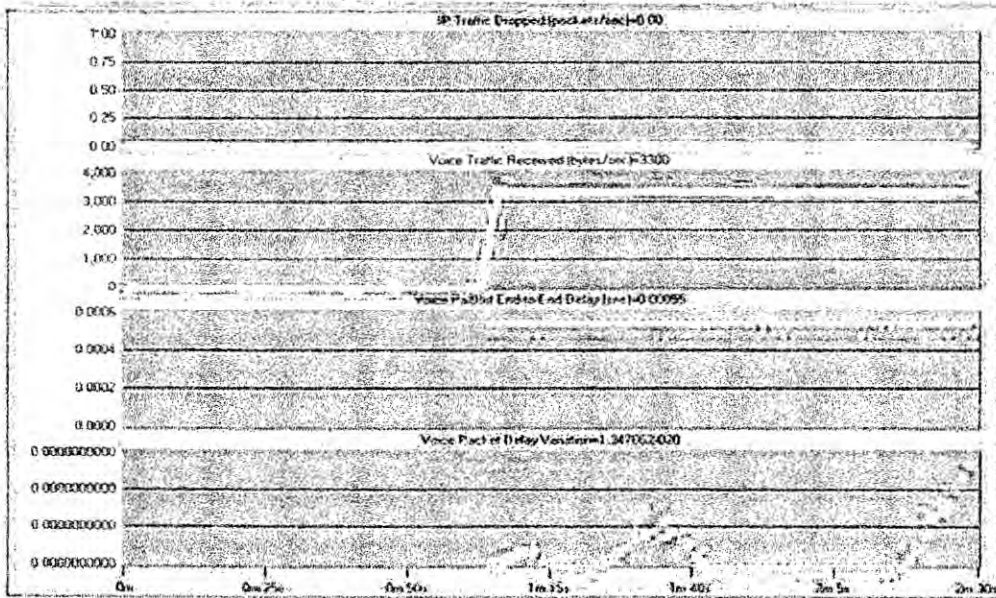
MÉTRICAS DE QOS PARA VOZ (G 729) + UDP EN ENLACE DE 128 KBPS



FUENTE: SIMULADOR

GRÁFICO 5.7

MÉTRICAS DE QOS PARA VOZ (GSM) EN ENLACE DE 128 KBPS



FUENTE: SIMULADOR

GRÁFICO 5.8

MÉTRICAS DE QOS PARA VOZ (GSM+TCP) + ENLACE DE 128 KBPS



FUENTE: SIMULADOR

GRÁFICO 5.9

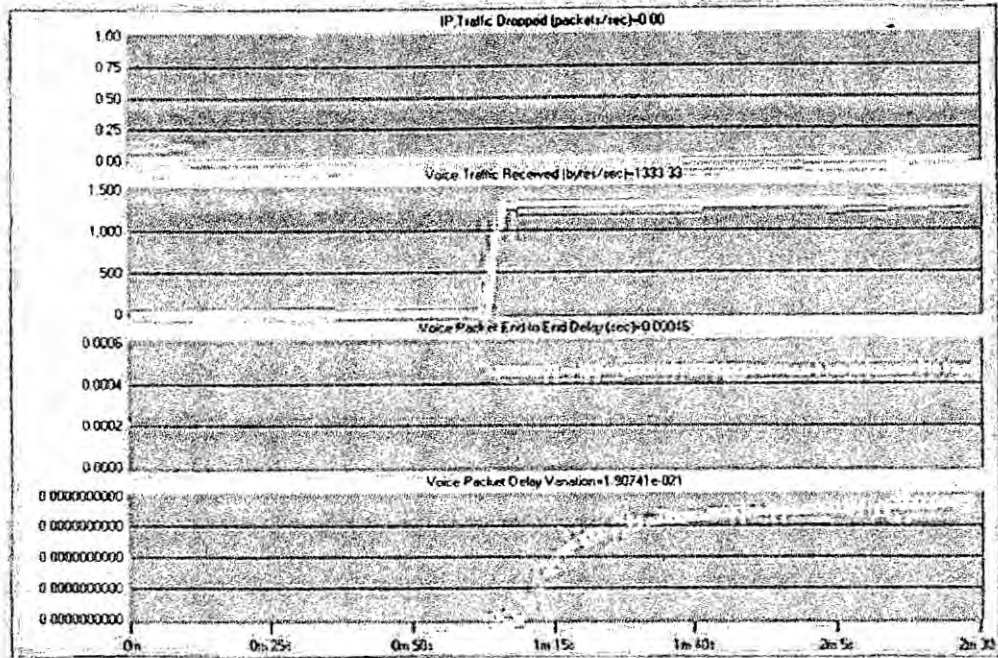
MÉTRICAS DE QOS DE VOZ (GSM) + UDP EN ENLACE DE 128 KBPS



FUENTE: SIMULADOR

GRÁFICO 5.10

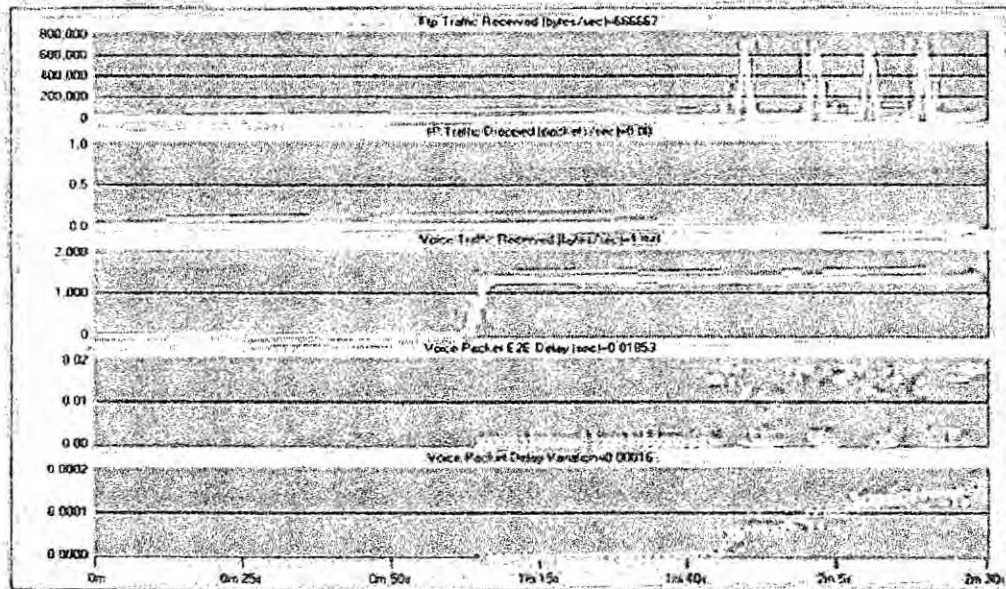
MÉTRICAS DE QOS PARA VOZ (G.723.1 A) EN ENLACE DE 128 KBPS



FUENTE: SIMULADOR

GRÁFICO 5.11

MÉTRICAS DE QOS PARA VOZ (G.723.1 A)+TCP EN ENLACE DE 128 KBPS



FUENTE: SIMULADOR

5.1.4 Resultados de la simulación incluyendo QoS

Las tablas 5.1 a 5.18 muestran en el campo "QoS" los valores de QoS calculados por la Ecuación 4.1 a partir de los parámetros IPDT y IPDV de esa misma línea. Se muestran los valores de QoS calculados en las velocidades de 128 Kbps a 1.5 Mbps. Los parámetros de ponderación utilizados fueron: $\alpha = 0,5$ y $\beta = 0,5$. El campo "Accion" indica si el QoS resultante hará conmutar el códec (Conmuta) o mantendrá el que está siendo usado (mantiene).

5.1.5 Enlace de 128 Kbps

TABLA 5.1

FLUJO ÚNICO DE VOZ EN ENLACE DE 128 KBPS

*Codec	Banda (bps)	IPDT (seg)	IPDV (seg)	IPLR (%)	QoS	Acción
G711	82800	0.0121	0.00	0.00	0.9193	Mantiene
G729	45600	0.0025	0.00	0.00	0.9833	Mantiene
GSM	31800	0.0040	0.00	0.00	0.9733	Mantiene
G723.1A	18190	0.0032	0.00	0.00	0.9786	Mantiene

FUENTE: SIMULADOR

TABLA 5.2

TCP + VOZ EN ENLACE DE 128 KBPS

Codec	Banda (bps)	IPDT seg.	IPDV seg.	IPLR* %	QoS	Acción
G711	82800	1.4177	0.5780	0.00	Neg	Conmuta
G729	45600	1.0693	0.3274	0.00	Neg	Conmuta
GSM	31800	0.0693	0.3274	0.00	Neg	Conmuta
723.1A	18190	0.9314	0.2338	0.00	Neg.	Conmuta

FUENTE: SIMULADOR

TABLA 5.3
UDP + VOZ EN ENLACE DE 128 KBPS

Codec	Banda (bps)	IPDT seg.	IPDV seg.	IPLR %	QoS	Acción
G711	82800	31.6115	96.4832	331.333	Neg	Conmuta
G729	45600	23.8992	54.8107	423.333	Neg	Conmuta
GSM	31800	1.0693	0.3274	0.00	Neg	Conmuta
G723.1A	18190	0.9314	0.2338	0.00	Neg	Conmuta

FUENTE: SIMULADOR

5.1.6 Enlace de 256 Kbps

TABLA 5.4
FLUJO ÚNICO DE VOZ EN ENLACE DE 256 KBPS

Codec	Banda (bps)	IPDT seg.	IPDV seg.	IPLR %	QoS	Acción
G711	82800	0.0063	0.0000	0.00	0.1080	Mantiene
G729	45600	0.0013	0.0013	0.00	0.1413	Mantiene
GSM	31800	0.0040	0.0000	0.00	0.1233	Mantiene
G723.1A	18190	0.0017	0.0000	0.00	0.1387	Mantiene

FUENTE: SIMULADOR

TABLA 5.5
TCP + VOZ EN ENLACE DE 256 KBPS

Codec	Banda (bps)	IPDT seg.	IPDV seg.	IPLR %	QoS	Acción
G711	82800	0.6012	0.0406	0.00	Neg	Conmuta
G729	45600	0.4653	0.0709	0.00	Neg	Conmuta
GSM	31800	0.4653	0.0709	0.00	Neg	Conmuta
G723.1A	18190	0.4316	0.0118	0.00	Neg	Conmuta

FUENTE: SIMULADOR

TABLA 5.6
UDP+VOZ EN ENLACE DE 256 KBPS

Codec	Banda (bps)	IPDT seg.	IPDV seg.	IPLR %	QoS	Acción
G711	82800	15.6901	25.1247	301.333	Neg	Conmuta
G729	45600	11.8181	14.1134	383.333	Neg	Conmuta
GSM	31800	11.8181	14.1134	383.333	Neg	Conmuta
G723.1A	18190	16.6187	27.8066	270.667	Neg	Conmuta.

FUENTE: SIMULADOR

5.1.7 Enlace de 512 Kbps

TABLA 5.7
FLUJO ÚNICO DE VOZ EN ENLACE DE 512 KBPS

Codec	Banda (bps)	IPDT seg.	IPDV seg.	IPLR %	QoS	Acción
G711	82800	0.0033	0.00	0.00	0.9780	Mantiene
G729	45600	0.0008	0.00	0.00	1.000	Mantiene
G711	31800	0.0008	0.00	0.00	1.000	Mantiene
G723.1A	18190	0.0009	0.00	0.00	1.000	Mantiene

FUENTE: SIMULADOR

TABLA 5.8
TCP + VOZ EN ENLACE DE 512 KBPS

Codec	Banda (bps)	IPDT seg	IPDV seg	IPLR %	QoS	Acción
G711	82800	0.2284	0.0047	0.00	0,4000	Mantiene
G729	45600	0.1424	0.0048	0.00	0.1047	Mantiene
GSM	31800	0.1424	0.0048	0.00	0.1047	Mantiene
G723.1A	18190	0.0695	0.0022	0.00	0.7467	Mantiene

FUENTE: SIMULADOR

TABLA 5.9

UDP + VOZ EN ENLACE DE 512 KBPS

Codec	Banda (bps)	IPDT seg	IPDV seg	IPLR mseg	QoS	Acción
G711	82800	7.7013	5.999	237.333	Neg.	Conmuta
G729	45600	5.8033	3.3497	299.333	Neg.	Conmuta
GSM	31800	5.8033	3.3497	299.333	Neg.	Conmuta
G23.1A	18190	8.2459	6.9656	210.667	Neg.	Conmuta

FUENTE: SIMULADOR

5.1.8 Enlace de 1000 Kbps

TABLA 5.10

FLUJO ÚNICO DE VOZ EN ENLACE DE 1000 KBPS

Codec	Banda (bps)	IPDT seg	IPDV seg	IPLR mseg	QoS	Acción
G711	82800	0.0019	0.00	0.00	0,9873	Mantiene
G729	45600	0.0005	0.00	0.00	0.9966	Mantiene
GSM	31800	0.0005	0.00	0.00	0.996	Mantiene
G723.1A	18190	0.0006	0.00	0.00	0.9966	Mantiene

FUENTE: SIMULADOR

TABLA 5.11

TCP + VOZ EN ENLACE DE 1000 KBPS

Codec	Banda (bps)	IPDT seg	IPDV-seg	IPLR mseg	QoS	Acción
G711	82800	0.03180	0.00077	0.00	0.7880	Mantiene
G729	45600	0.0292	0.0007	0.00	0.8066	Mantiene
GSM	31800	0.0291	0.0007	0.00	0.8066	Mantiene
G723.1A	18190	0.0285	0.0007	0.00	0.8066	Mantiene

FUENTE: SIMULADOR

Tabla 5.12
UDP + VOZ EN ENLACE DE 1000 KBPS

Codec	Banda (bps)	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	3.9221	1.5950	113.333	Neg	Conmuta
G729	45600	2.8896	0.8626	129.333	Neg	Conmuta
GSM	31800	2.8896	0.8626	129.333	Neg	Conmuta
G723.1A	18190	4.1801	1.8189	94.667	Neg	Conmuta

FUENTE: SIMULADOR

5.1.9 Enlace de 1300 Kbps

Tabla 5.13
FLUJO ÚNICO DE VOZ EN ENLACE DE 1300 KBPS

Codec	Banda (bps)	IPDT seg	IPDV Seg	IPLR	QoS	Acción
G711	82800	0.0016	0.00	0.00	0.9893	Mantiene
G729	45600	0.0004	0.00	0.00	0.9973	Mantiene
GSM	31800	0.004	0.00	0.00	0.9973	Mantiene
G723.1A	18190	0.0005	0.00	0.00	0.9967	Mantiene

FUENTE: SIMULADOR

Tabla 5.14
5.14: TCP + VOZ EN ENLACE DE 1300 KBPS

Codec	Banda (bps)	IPDT mseg	IPDV mseg	IPLR mseg	QoS	Acción
G711	82800	0.02565	0.00033	0.00	0.8299	Mantiene
G729	45600	0.02233	0.00022	0.00	0.8513	Mantiene
GSM	31800	0.0223	0.0002	0.00	0.8513	Mantiene
G723.1A	18190	0.0223	0.0002	0.00	0.8513	Mantiene

FUENTE: SIMULADOR

TABLA 5.15
UDP + VOZ EN ENLACE DE 1300 KBPS

Codec	Banda (bps)	IPDT seg	IPDV Seg	IPLR	QoS	Acción
G711	82800	3.1845	1.069	0.00	Neg	Conmuta
G729	45600	2.4232	0.6038	0.00	Neg	Conmuta
GSM	31800	2.4232	0.6038	0.00	Neg	Conmuta
G723.1A	18190	0.0220	0.0002	0.00	1.846	Mantiene

FUENTE: SIMULADOR

5.1.10 Enlace de 1544 Kbps

TABLA 5.16
FLUJO ÚNICO DE VOZ EN ENLACE DE 1544 KBPS

Codec	Banda (bps)	IPDT seg	IPDV Seg	IPLR	QoS	Acción
G711	82800	0.0012	0.00	0.00	0.9920	Mantiene
G729	45600	0.0003	0.00	0.00	0.9980	Mantiene
GSM	31800	0.0003	0.00	0.00	0.9980	Mantiene
G723.1A	18190	0.0004	0	0.00	0.9980	Mantiene

FUENTE: SIMULADOR

TABLA 5.17
TCP + VOZ EN ENLACE 1544 KBPS

Codec	Banda (bps)	IPDT seg	IPDV seg	IPLR seg	QoS	Acción
G711	82800	0.01432	0.00013	0.00	0.9046	Mantiene
G729	45600	0.0135	0.0001	0.00	0.910	Mantiene
GSM	31800	0.0135	0.0001	0.00	0.910	Mantiene
G23.1A	18190	0.0136	0.0001	0.00	0.910	Mantiene

FUENTE: SIMULADOR

TABLA 5.18

UDP + VOZ EN ENLACE DS1

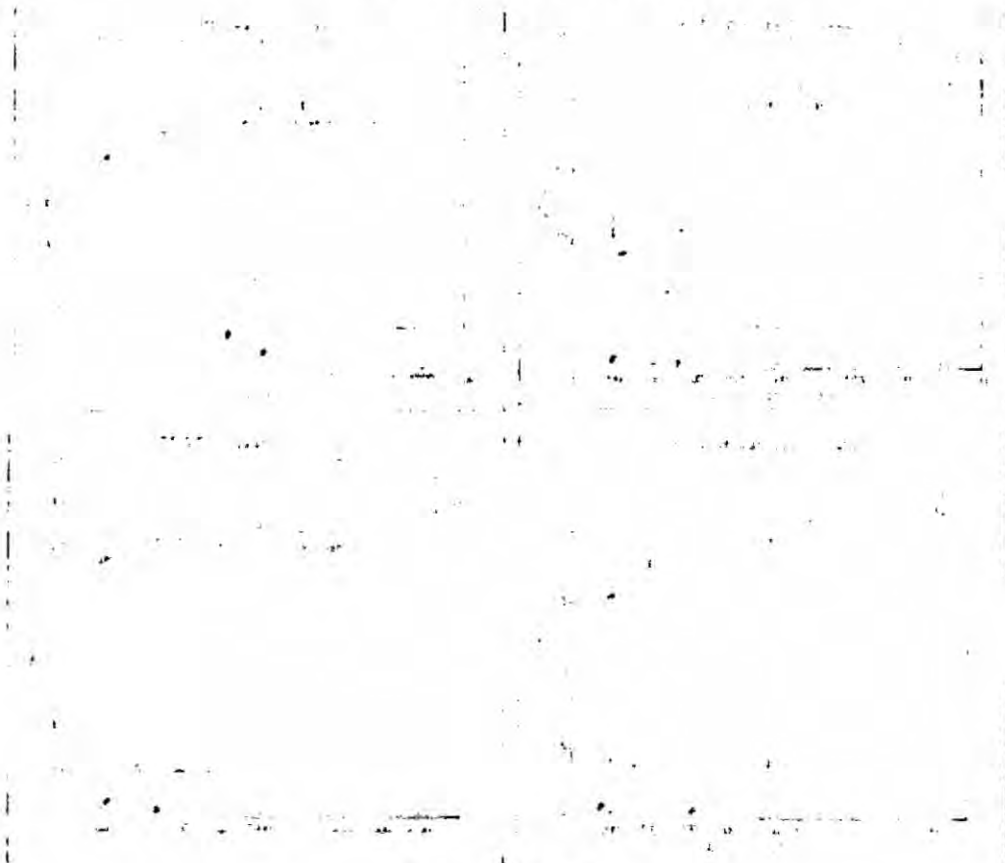
Codec	Banda (bps)	IPDT seg	IPDV Seg	IPLR	QoS	Acción
G711	82800	0.01926	0.00036	0.00	0.8730	Mantiene
G729	45600	0.0203	0.0003	0.00	0.8660	Mantiene
GSM	31800	0.02031	0.0003	0.00	0.8660	Mantiene
G723.1A	18190	0.0189	0.0003	0.00	0.8860	Mantiene

FUENTE: SIMULADOR

5.1.11 Resumen de las métricas obtenidas por simulación

GRÁFICO 5.12

COMPORTAMIENTO DEL IPDT PARA LOS 4 CODECS



FUENTE: SIMULADOR

5.2 Resultados de las experiencias de validación

5.2.1 Validación de los flujos interferentes TCP y UDP

En esta experiencia fueron validadas las simulaciones con tráfico interferente cuando se encaminan dos flujos simultáneos (voz y TCP) y (voz y UDP), en las velocidades ya indicadas y para cada uno de los *codecs* estudiados, con equipos routers "reales". La topología de testes está indicada en la Fig. 5.13 separada de la Fig.4.22 para fines de una mejor visualización.

La experiencia empezó transmitiendo sólo el tráfico de voz en el enlace configurado, con el objetivo de obtener el referencial para el comportamiento del flujo único de voz y con tráfico TCP y UDP.

5.2.2 Flujo único en el enlace

Las tabla 5.20, 5.23, 5.26, 5.29, 5.32 y 5.35 muestran los resultados de los testes efectuados con equipos "reales" para el caso de un flujo único de voz.

5.2.3 Con flujo interferente TCP

Las tablas 5.21, 5.24, 5.27, 5.30, 5.33 y 5.36 muestran los resultados para el caso de un flujo de voz simultáneo con tráfico interferente TCP.

El terminal PC1 se comunica con el PC3 con tráfico de voz (G711) y simultáneamente el PC2 se comunica con el PC4, transportando el tráfico interferente TCP y UDP uno de cada vez. El tráfico TCP fue generado a través del *software FileZilla (FTP)*, transmitiendo un archivo de datos entre PC2 y PC4.

Las características de esta FTP es: tamaño del archivo= 1000000 bytes, que en la velocidad de 128 Kbps representa un tiempo de transmisión de 62.5 seg.

5.2.4 Con flujo interferente UDP

Las tablas 5.22, 5.25, 5.28, 5.31, 5.34 y 5.37 muestran los resultados de los testes efectuados con equipos "reales" para el caso de un flujo de voz simultáneo con tráfico interferente **UDP**. Este tráfico fue generado por el software **IPERF**.

5.2.5 Procedimiento de las pruebas

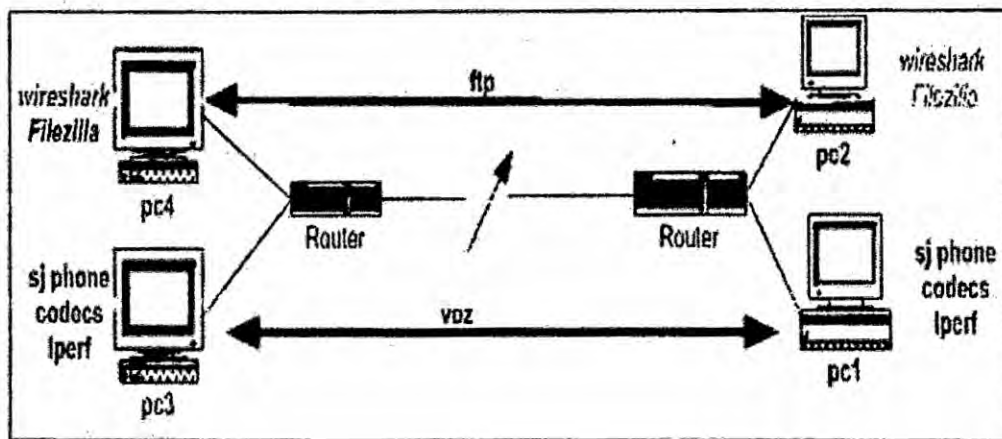
Esta experiencia fue efectuada sobre la red mostrada en la Fig. 5.13, en la cual han sido configuradas las velocidades desde 128 **Kbps** hasta 1.5 **Mbps**, fueron contados los paquetes recibidos y transmitidos con software *Wireshark (Ethereal)*, registrando los valores de los parámetros relacionados con las métricas de **QoS**.

Inicialmente el PC3 y el PC1 se comunican entre si utilizando una banda que soporta la banda del códec G.711 generada por el **IPERF**. Manteniendo esta comunicación en curso, fue efectuada una transferencia de archivos entre las *workstations* PC2 y PC4 (**TCP**) con cargas equivalentes a 1.5 **MB** (62,5 segundos en el enlace de 128000 **bps**) usando los mismos valores de la simulación. En la *workstation* PC4 fue instalado el analizador de protocolo *Wireshark* para la captura de paquetes **UDP** y **TCP** cuando hay tráfico sólo de voz y mide los nuevos valores cuando se adiciona tráfico interferente **TCP** o **UDP**; para **UDP** fue generada por el **IPERF** una banda

igual a la configurada para simular una situación de congestión. Han sido evaluados los efectos de estos flujos interferentes sobre cada uno de los codecs estudiados y los resultados están registrados en las tablas 5.20 a 5.36. Las pruebas tuvieron las características parecidas a esos simulados. El valor de IPDT fue obtenido a través del comando *ping*, durante la transferencia de archivo en TCP y la generación de tráfico en UDP.

FIGURA 5.13

TOPOLOGÍA PARA VALIDACIÓN DE LAS MÉTRICAS OBTENIDAS POR SIMULACIÓN



FUENTE: PROPIA DEL AUTOR

TABLA 5.18
UDP+VOZ EN ENLACE DS1

Codec	Banda (bps)	IPDT(seg)	IPDV(seg)	IPLR	QoS	Acción
G711	83200	0.0155	0.0011	0.00	1,8746	Mantiene
G729	46400	0.0110	0.0012	0.00	1,9026	Mantiene
GSM	32400	0.0095	0.0020	0.00	1,8966	Mantiene
G723.1 ^a	18133	0.0080	0.0023	0.00	1,9006	Mantiene

FUENTE: SIMULADOR

TABLA 5.19
FLUJO ÚNICO DE VOZ EN ENLACE DE 128 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0155	0.0011	0.00	0.9370	Mantiene
G729	45600	0.0110	0.0012	0.00	0.9510	Mantiene
GSM	31800	0.0095	0.0020	0.00	0.9480	Mantiene
G723.1A	18190	0.0080	0.0023	0.00	0.8930	Mantiene

FUENTE: SIMULADOR

TABLA 5.20
TCP + VOZ EN ENLACE DE 128 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0690	0.4700	0.00	0,3000	Mantiene
G729	45600	0.0830	0.0044	0.00	0.4700	Mantiene
GSM	31800	0.0660	0.0474	0.00	0,3100	Mantiene
G723.1A	18190	0.0750	0.0295	0.00	0,4600	Mantiene

FUENTE: SIMULADOR

TABLA 5.21
UDP + VOZ EN ENLACE DE 128 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0072	0.0038	0.00	0.9380	Mantiene
G729	45600	0.0072	0.0038	0.00	0.9380	Mantiene
GSM	31800	0.0072	0.0038	0.00	0.9380	Mantiene
G723.1A	18190	0.0072	0.0038	0.00	0.9380	Mantiene

FUENTE: SIMULADOR

TABLA 5.22
FLUJO ÚNICO DE VOZ EN ENLACE DE 256 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0060	0.0029	0.00	0.9510	Mantiene
G729	45600	0.0035	0.0010	0.00	0.9780	Mantiene
GSM	31800	0.0030	0.0010	0.00	0.9800	Mantiene
G723.1A	18190	0.0030	0.0010	0.00	0.9800	Mantiene

FUENTE: SIMULADOR

TABLA 5.23
TCP + VOZ EN ENLACE DE 256 KBPS

Codec	Banda bps	IPDT seg	IPDV Seg	IPLR	QoS	Acción
G711	82800	0.0041	0.0027	0.00	0.9590	Mantiene
G729	45600	0.0037	0.0041	0.00	0.9460	Mantiene
GSM	31800	0.0035	0.0051	0.00	0.9370	Mantiene
G723.1A	18190	0.0035	0.0091	0.00	0.8970	Mantiene

FUENTE: SIMULADOR

TABLA 5.24
UDP + VOZ EN ENLACE DE 256 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.037	0.0067	0,00	0.8095	Mantiene
G729	45600	0.037	0.0067	0,00	0.8095	Mantiene
GSM	31800	0.037	0.0067	0,00	0.8095	Mantiene
G723.2A	18190	0.037	0.0067	0,00	0.8095	Mantiene

FUENTE: SIMULADOR

TABLA 5.25
FLUJO ÚNICO DE VOZ EN ENLACE DE 512 KBPS

Codec	Banda bps	IPDT mseg	IPDV mseg	IPLR mseg	QoS	Acción
G711	82800	0.0015	0.4910	0,00	-7,8300	Conmuta
G729	45600	0.0020	0.4020	0,00	-6,053	Conmuta
GSM	31800	0.0010	0.0010	0,00	0.9900	Mantiene
G723.1A	18190	0.0015	0.0060	0,00	0.9350	Mantiene

FUENTE: SIMULADOR

TABLA 5.26
TCP + VOZ EN ENLACE DE 512 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0140	0.0015	0,00	0.9383	Mantiene
G729	45600	0.0165	0.0031	0,00	0.9156	Mantiene
GSM	31800	0.0150	0.0043	0,00	0.9070	Mantiene
G723.1A	18190	0.0125	0.0053	0,00	0.9053	Mantiene

FUENTE: SIMULADOR

TABLA 5.27
UDP + VOZ EN ENLACE DE 512 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0190	0.0066	0,00	0.8706	Mantiene
G729	45600	0.0190	0.0066	0,00	0.8706	Mantiene
GSM	31800	0.0190	0.0066	0,00	0.8706	Mantiene
G723.1A	18190	0.0190	0.0066	0,00	0.8706	Mantiene

FUENTE: SIMULADOR

TABLA 5.28
FLUJO ÚNICO DE VOZ EN ENLACE DE 1000 KBPS

Codec	Banda bps	IPDT mseg	IPDV mseg	IPLR mseg	QoS	Acción
G711	82800	0.0005	0.0010	0,00	1	Mantiene
G729	45600	0.0005	0.1980	0,00	¿	Mantiene
GSM	31800	0.0005	0.0005	0,00	1	Mantiene
G723.1A	18190	0.0005	0.0012	0,00	1	Mantiene

FUENTE: SIMULADOR

TABLA 5.29
TCP + VOZ EN ENLACE DE 1000 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0045	0.0060	0,00	0.9250	Mantiene
G729	45600	0.0065	0,0060	0,00	0.9183	Mantiene
GSM	31800	0.0060	0,0060	0,00	0.9200	Mantiene
G723.1A	18190	0.0030	0,0060	0,00	0.9300	Mantiene

FUENTE: SIMULADOR

TABLA 5.30
UDP + VOZ EN ENLACE DE 1000 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0105	0.0048	0,00	0.4866	Mantiene
G729	45600	0.0105	0.0048	0,00	0.4866	Mantiene
GSM	31800	0.0105	0.0048	0,00	0.4866	Mantiene
G723.1A	18190	0.0105	0.0048	0,00	0.4866	Mantiene

FUENTE: SIMULADOR

TABLA 5.31
FLUJO ÚNICO DE VOZ EN ENLACE DE 1300 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0050	0.0014	0,00	0.4833	Mantiene
G729	45600	0.0050	0.0023	0,00	0.4833	Mantiene
GSM	31800	0.0050	0.0027	0,00	0.4833	Mantiene
G723.1A	18190	0.0050	0.0018	0,00	0.4833	Mantiene

FUENTE: SIMULADOR

TABLA 5.32
TCP + VOZ EN ENLACE DE 1300 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0040	0.0052	0,00	0.8145	Mantiene
G729	45600	0.0035	0.0046	0,00	0.9423	Mantiene
GSM	31800	0.0015	0.0020	0,00	0.9300	Mantiene
G723.1A	18190	0.0015	0.0016	0,00	0.9300	Mantiene

FUENTE: SIMULADOR

TABLA 5.33
UDP + VOZ EN ENLACE DE 1300 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0050	0.0003	0,00	0.9200	Mantiene
G729	45600	0.0050	0.0012	0,00	0.9200	Mantiene
GSM	31800	0.0050	0.0049	0,00	0.9200	Mantiene
G723.1A	18190	0.0050	0.0012	0,00	0.9200	Mantiene

FUENTE: SIMULADOR

TABLA 5.34
FLUJO ÚNICO DE VOZ EN ENLACE DE 1.544 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	Acción
G711	82800	0.0025	0.0010	0,00	0.9532	Mantiene
G729	45600	0.0020	0.0075	0,00	0.9532	Mantiene
GSM	31800	0.0020	0.0015	0,00	0.9532	Mantiene
G723.1A	18190	0.0020	0.0001	0,00	0.9532	Mantiene

FUENTE: SIMULADOR

TABLA 5.35
TCP + VOZ EN ENLACE DE 1544 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	acción
G711	82800	0.0025	0.0010	0,00	0.9533	Mantiene
G729	45600	0.0020	0.7590	0,00	0.9533	Mantiene
GSM	31800	0.0020	0.0015	0,00	0.9533	Mantiene
G723.1A	18190	0.0020	0.0010	0,00	0.9533	Mantiene

FUENTE: SIMULADOR

TABLA 5.36
UDP + VOZ EN ENLACE DE 1544 KBPS

Codec	Banda bps	IPDT seg	IPDV seg	IPLR	QoS	acción
G711	82800	0.0055	0.0085	0,00	0.8966	Mantiene
G729	45600	0.0055	0.0085	0,00	0.8966	Mantiene
GSM	31800	0.0055	0.0085	0,00	0.8966	Mantiene
G723.1A	18190	0.0055	0.0085	0,00	0.8966	Mantiene

FUENTE: SIMULADOR

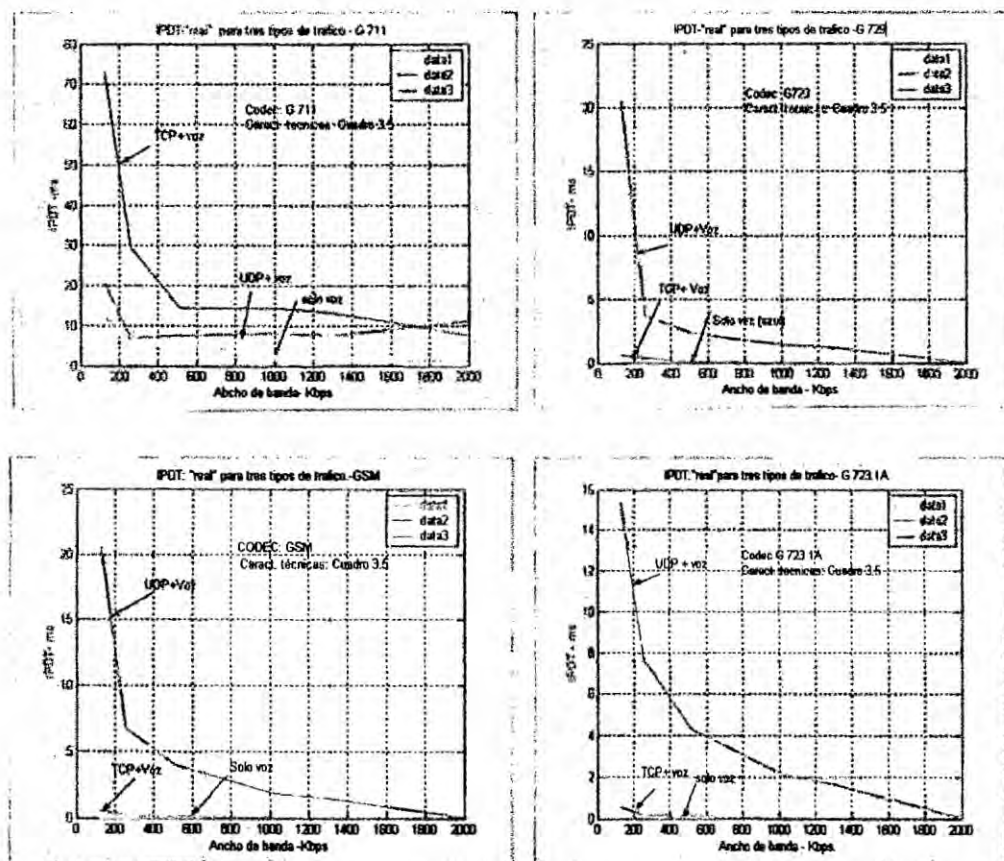
Resumen de la experiencia de validación.

La Fig 5.14 muestra gráficos del IPDT de voz resultado de los datos obtenidos con equipos y comunicaciones. "reales".

Puede observarse que el comportamiento de la métrica IPDT en los cuatro gráficos, simulados y reales, son bastante semejantes (Fig.5.1). Lo mismo sucede con el IPDV y IPLR. Por tanto para los fines de este trabajo podemos considerarlo, lo mínimo coherentes con la propuesta.

GRAFICO 5.13

COMPORTAMIENTO DEL IPDT "REAL" PARA LOS 4 CODECS



FUENTE: SIMULADOR

5.2.6 Validación del algoritmo de restauración propuesto

La validación del algoritmo propuesto fue efectuada en 2 actividades indicadas a seguir: 1) Análisis del comportamiento de los *codecs* en situaciones de congestión de la red (para diversos anchos de banda) y 2) el funcionamiento del algoritmo

5.2.6.1 Comportamiento de los *codecs* en las situaciones de congestionamiento de la red.

El procedimiento general puede resumirse a lo siguiente: 1) fueron utilizadas Cuatro estaciones de trabajo de red LAN. Dos estaciones para la transferencia de archivo TCP y UDP simultánea con la voz y las otras dos para el canal de audio/voz. El enlace WAN foi efectuado configurando las velocidades de 128K hasta DS1 en los mismos routers, en sus puertas seriales. La conexión de las puertas seriales fue hecha con cable serial cross, Cualquiera de los lados terminales (pc4 o pc3) o (pc2, pc1) puede iniciar una comunicación telefónica con el otro lado utilizando o software VoIP SJPhone. Manteniendo esta comunicación activa, se procede con los microcomputadores restantes a hacer una transferencia de archivos de tamaño igual al de la simulación, uno de cada lado (1 MB), utilizando el software Filezilla Este procedimiento fue repetido 5 veces. Las herramientas de *software* siguientes fueron también utilizadas: el analizador de protocolo *Wireshark*, para la captura y contaje de los paquetes y medidas de retraso y *jitter*. El *software* IPERF para medida de banda y generación de tráfico UDP. El *software* FileZilla, tanto como cliente

como servidor de FTP, para la generación de tráfico interferente TCP. El *software ping plotter* básicamente para la medida del RTT y retraso fin-a-fin. Finalmente el *software VoIP SJPHONE* para pruebas de los codecs: **GSM**, G.711 **LAW** μ , G.711 **LAW** A. Los resultados están incluidos en las tablas 5.38 y figuras 5.8 a 5.13 y el resumen en la tabla 5.39. Estas tablas y figuras son productos de las experiencias exclusivas con el *Wireshark* y **SJPHONE** comunicándose en una red local *Ethernet* de 10 **Mbits** con cada uno de los *codecs* citados en el modo punto a punto.

5.2.6.2 Resultados del comportamiento de los *codecs* del SJPHONE

A seguir están indicados los resultados de comunicaciones **VoIP** entre dos computadores, punto a punto y la captura de algunos paquetes con *Wireshark*.

FIGURA.5.15
COMUNICACIÓN CON CODEC GSM, 10 MBPS.

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)
192.168.1.39	49158	192.168.1.100	49182	0x4AAS8B3	GSM 06.10	804	0 (0.0%)	49.57
192.168.1.100	49182	192.168.1.39	49158	0x311C127	GSM 06.10	803	0 (0.0%)	43.35

Select a forward stream with left mouse button
Select a reverse stream with SHIFT + left mouse button

FUENTE: SIMULADOR

TABLA 5.37
COMUNICACIÓN CON CODEC GSM, 10 MBPS.

Dirección IP de origen	Puerto
Puerto de origen	49158
Dirección IP de destino	192.168.1.100
Puerto de destino	49182
Fuente de sincronismo	0x4AAS863
Tipo de paquete	GSM 96.10
Paquetes enviados	80
Paquetes perdidos	0 0%
Máximo delta (mseg)	49.57
Máximo IPDV (mseg)	20
IPDV MEDIO	19,63

FUENTE: SIMULADOR

La tabla 5.38 arriba resume los resultados obtenidos (de máximo IPDT, máximo IPDV e IPDV medio) para una comunicación punto a punto con *codec* GSM.

FIGURA 5.16 - COMUNICACIÓN CON CODEC G. 711 LAW A

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter
192.168.1.39	49172	0x1A6803ED	ITU-T G.711 PCMA	3948	0 (0.0%)	43.06	10000000000.00	1000000000
192.168.1.100	49172	0x3E1A480	ITU-T G.711 PCMA	3924	0 (0.0%)	30.36	10000000000.00	1000000000

Select a forward stream with left mouse button
 Select a reverse stream with SHIFT + left mouse button

Unselect Find Reverse Save As Mark Packets Prepare Filter Copy Analyze Close

FUENTE: ANALIZADOR DE PROTOCOLO

FIGURA. 5.17 - COMUNICACIÓN CON CODEC G.711 LAW μ

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Dest port	SSRC	Payload	Packets	Lost	Max Delay (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
49160	0x1381368	ITU-T G.711 PCMU	2626	0 (0.0%)	39.90	10000000000.00	10000000000.00	X
49160	0x16499308	ITU-T G.711 PCMU	2626	0 (0.0%)	40.87	10000000000.00	10000000000.00	X

Select a forward stream with left mouse button
Select a reverse stream with SHIFT + left mouse button

Unselect Find Reverse Save As Mark Packets Prepare Filter Copy Analyze Close

FUENTE: ANALIZADOR DE PROTOCOLO

FIGURA 5.18 - INFORMACIONES DEL PAQUETE 3505 – G711 PCMA

File Edit View Go Capture Analyze Statistics Help

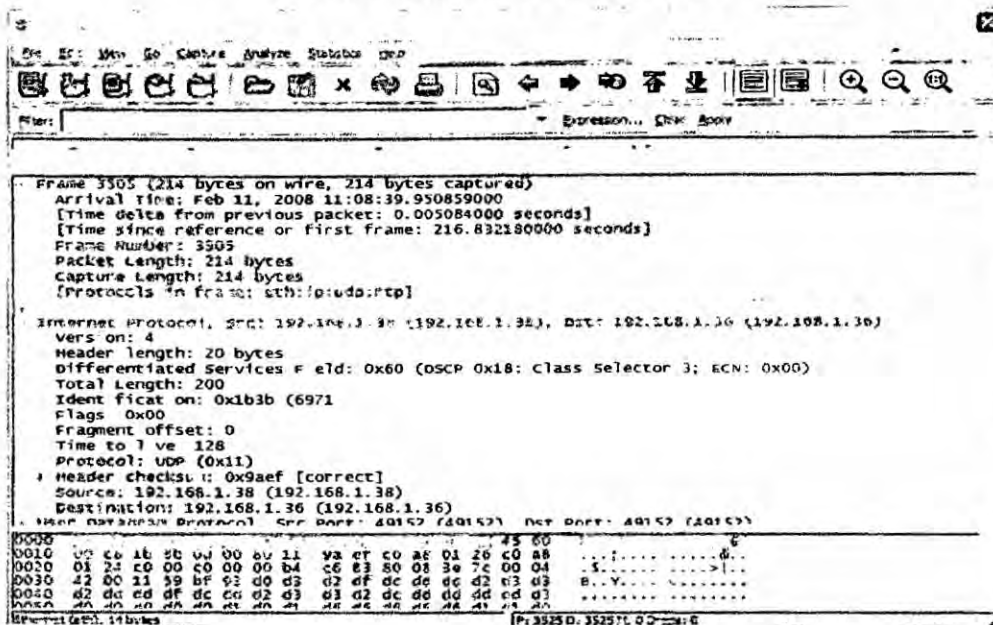
Filter: Expression... Gear Copy

Time	Source	Destination	Protocol	Info
0.0000	00 08 34 39 35 8c 00 50	b6 00 95 d6 06 00 45 80	..735..PE
0.0010	00 c8 1b 3b 00 00 80 11	9a ef c0 a8 01 26 c0 a8&&
0.0020	01 71 20 00 00 00 00 00	26 87 60 00 20 70 00 01	E	E

File: C:\net\cap161 790 KB 00:01:37 | P: 3525 D: 3525 M: 0 Drops: 0

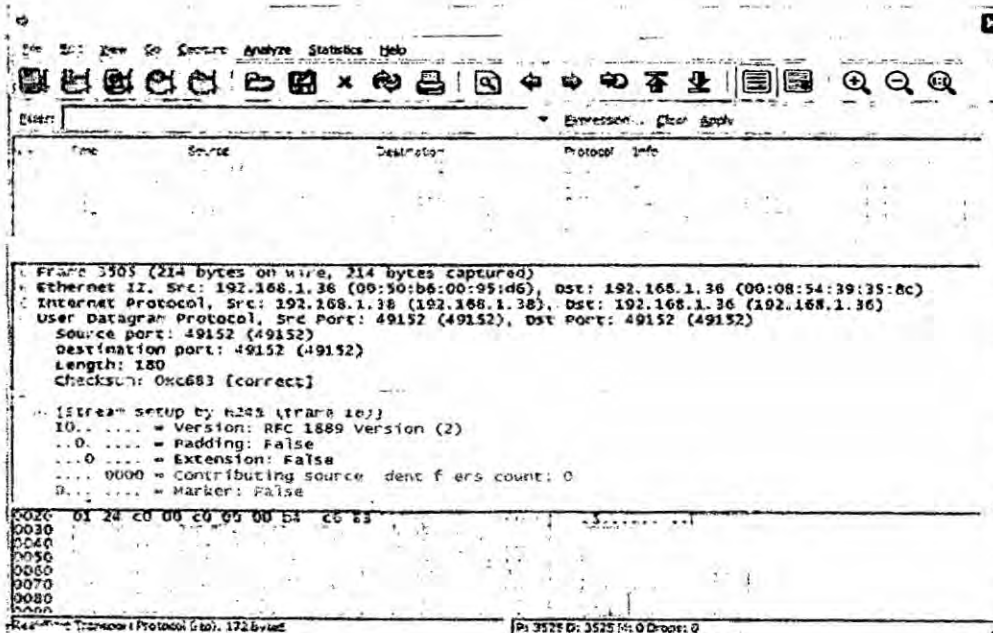
FUENTE: ANALIZADOR DE PROTOCOLO

FIGURA - 5.19
CONTINUACIÓN DE LA PAG. ANTERIOR



FUENTE: ANALIZADOR

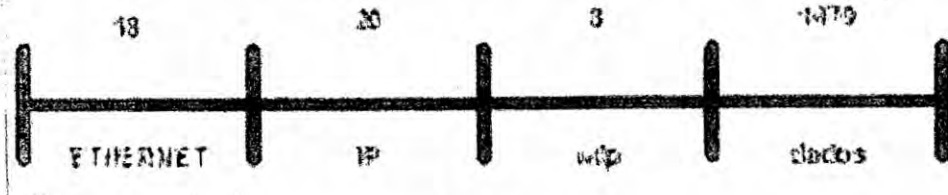
FIGURA 5.20- ETHERREAL CON RTP (RFC 1889)



FUENTE: ANALIZADOR

FIGURA 5.21

PAQUETE UDP VISTO CON ETHEREAL



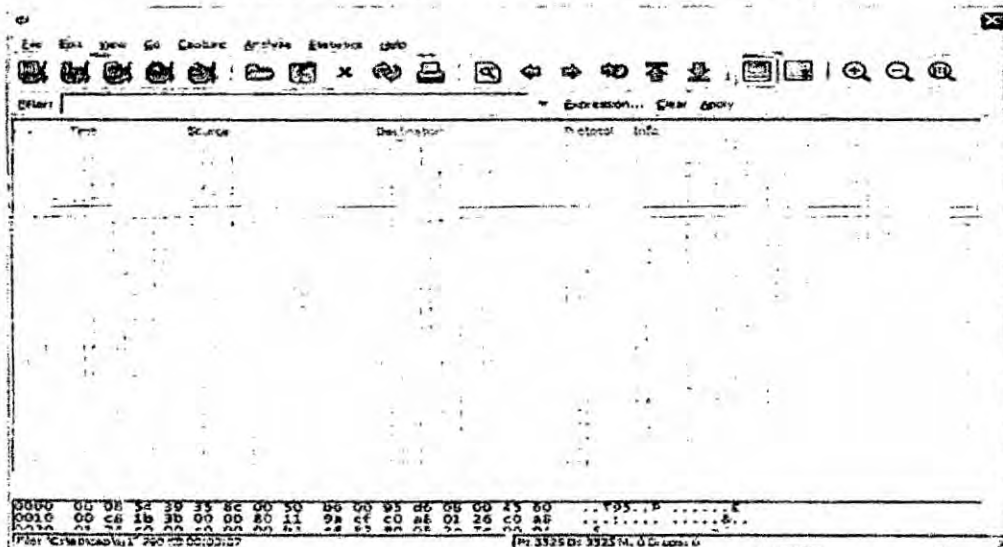
FUENTE: ANALIZADOR

5.2.6.3 Evaluación del SJPHONE con ETHEREAL

La figura 5.22 abajo muestra el resultado de la activación de SJPHONE tanto en la *Laptop* como en la *PC3*, visto por el *Ethereal*. Observamos que ambos terminales negocian en el inicio el *codec* que va ser usado en la conexión, en función de la banda disponible. Como la conexión es *ethernet* 10 Mbps, en este caso, intentarían conectarse con el *codec* de mas alta banda (y calidad), esto es G.711, se este *codec* estuviera implementado.

FIGURA 5.22

DETALLES ADICIONALES DEL PAQUETE 3505 (G711 PCMA)



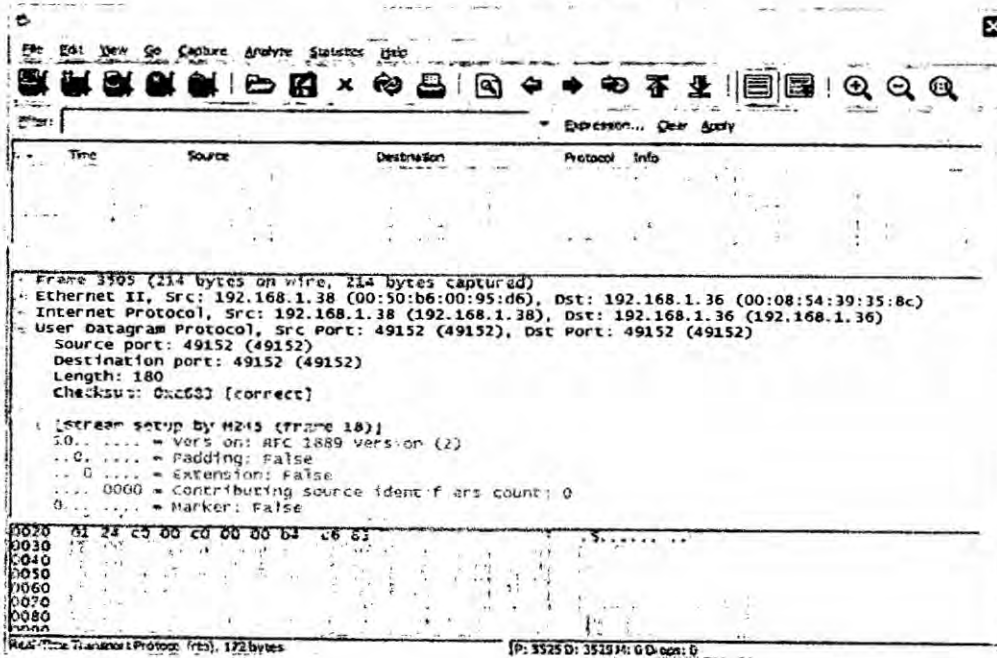
FUENTE: ANALIZADOR

FIGURA 5.23
FRAME 3505 (ETH: IP:UDP:RTP)



FUENTE: ANALIZADOR

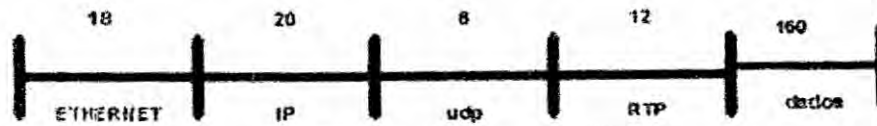
FIGURA 5.24
FRAME 3505 CON RTP



FUENTE: ANALIZADOR

Los gráficos arriba muestran el paquete **SJPHONE** con la siguiente estructura:

FIGURA 5.25
PAQUETE SJPHONE



FUENTE: ANALIZADOR

FIGURA 5.26
COMUNICACION CON CODEC G.711 LAW A Y RTP

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter
192.168.1.39	49172	0x1A6803ED	ITU-T G.711 PCMA	3948	0 (0.0%)	43.06	1000000000.00	100000000.00
192.168.1.100	49172	0x3E1AABD	ITU-T G.711 PCMA	3924	0 (0.0%)	30.36	1000000000.00	100000000.00

Select a forward stream with left mouse button
Select a reverse stream with SHIFT + left mouse button

Unselect Find Reverse Save As Mark Packets Prepare Filter Copy Analyze Close

TABLA 5.38
RESUMEN DE LOS CODECS

Codec	Total de Paquetes	Paquetes perdidos	IPDT Max. (ms)	IPDV Max. (ms)	IPDV Medio (ms)
GSM	804	0.00	33.65	0.00	0.00
	803	0.00	46.79	0.00	0.00
G.711 A	3928	0.00	43.06	0.00	0.00
	3924	0.00	30.36	0.00	0.00
G.711 μ	2626	0.00	39.90	0.00	0.00
	2626	0.00	40.87	0.00	0.00

FUENTE: ANALIZADOR

La tabla 5.39 resume las características **IPDT** e **IPDV** de los *codecs* disponibles en el **SJPHONE**

5.2.7 Comportamiento del algoritmo de QoS

Esta prueba es también llamada de "Validación Manual", visto que la experiencia no es efectuada dinámicamente con los flujos de **RTP/RTCP** en curso, si no es evaluada en función de valores digitados por el operador cuando se ejecuta el algoritmo propuesto en un microcomputador que va solicitando los valores de los parámetros de entrada que en la realidad serán automáticos.

5.2.7.1 Validación del algoritmo

La validación, fue efectuada ejecutando el *software* en un microcomputador, y digitando los valores de los parámetros solicitados.

Los parámetros de la validación manual del algoritmo.

- **TIMESTAMP** do **SR**
- Entre el valor del **NTP timestamp (MSB)** = 0XB44DB705 (hexadecimal o decimal); 3024992005
- Entre el valor del **NTP timestamp (LSB)** = 0X20000000/16 = 0.125 (hexadecimal o decimal);
- **Timestamp** del **SR** = 3024992005.125;
- Entre **LSR** = 46853.125 seg;
- Entre **DLSR** = 5.250 seg;
- **A** = b710: 8000 = 46864.500;
- **RTT** = **A**- **LSR** – **DLSR**;

- $RTT = 46864.500 - 46858.375 = 6.125$ seg;
- $IPDT = 6.125$ seg;
- Entre el valor del *Interarrival jitter* = 0.000000;
- Tasa de muestreo del *codec* = 8000;
- $IPDV = 0.00$;
- *Enter cumulative number of packet lost* = 10;
- *Enter highest sequence number received* = 61209;
- $IPLR = 0.016$ %;
- $QoS = 0,5 * f (IPDT) + 0,5 * g (IPDV)$;
- Donde: $f (IPDT) = (V2pdt - IPDT)/(V2pdt - V1pdt)$;
- $G (IPDV) = (V2pdv - IPDV)/(V2pdv - V1pdv)$;
- $V2pdt = 150$ ms, $V1pdt = 0$, $V2pdv = 50$ ms, $V1pdv = 0$.
 $\alpha = 0,5$; $\beta = 0,5$
- Si $0,5 \leq QoS \leq 1.0$, red = normal (mantiene el *códec*).
 Si $0,0 \leq QoS \leq 0.4$, red = cargada, espera hasta el tercero valor consecutivo para decidir.
 Si $0,0 \leq QoS \leq 0.4$, red = cargada, espera por más un valor.
 Si $0,0 \leq QoS \leq 0.4$, red = cargada, conmuta para bajo.
 Si tercer valor $0,5 \leq QoS \leq$ red = normal, mantiene el *codec*.
- Si $QoS = 0,0$, red = congestionada, consulta.
- Consulta: ¿Ya fue transmitido "*early packet*"?
 Respuesta = Ya (Espera próximo **RR**).

Respuesta = No (Envía inmediatamente vía **RTP** para cambiar de *codec*).

5.2.7.2 Ejemplo de prueba del algoritmo

En este ejemplo ejecutase el programa del algoritmo que se inicia "solicitando" ingresar **MSB**. El operador deberá digitar el valor solicitado del **MSB** correspondiente en hexadecimal o decimal y así sucesivamente hasta solicitar ingresar *interarrival*. El ítem 2 parámetros de salida (en negrito) representa el resultado de las operaciones del algoritmo.

a. Parámetros de entrada manual

- Ingresar **MSB**: 3024992005;
- Ingresar **LSB**: 0.125;
- Ingresar **DLSR**: 5.250;
- Ingresar **LSR**: 46853.125;
- Ingresar **A**: 46864.500;
- Ingresar **Interarrival**: 0.00.

b. Parámetros de salida

- valor de **A**: 46864.5000000000;
- El valor de **RTT**: 6.1250000000;
- El valor de **IPDT**: 3.0625000000;
- El valor de **QoS**: 0.9695834201;
- **Normal**: 82800 bps.

Para estos parámetros de entrada, se mantendrá el *codec* G.711, dado el resultado de la **QoS** = 0.9695.

CAPÍTULO VI

6. DISCUSION DE LOS RESULTADOS

Sin la pretensión de haber agotado el asunto, en este capítulo son comentados y analizados los resultados obtenidos, tanto en la simulación como en las experiencias reales en lo relacionado con el comportamiento de las métricas: **IPDT**; **IPDV** y **QoS** en las varias topologías realizadas. Se debe recordar solamente que todos estos datos están referenciados al servicio ofrecido actualmente por la Internet que es conocido como "best effort" y que utiliza un mecanismo de colas de paquetes conocido como FIFO (First In, First Out), y hasta aquí, en este trabajo, no se ha considerado ningún mecanismo de QoS para mejorar este servicio básico.

6.1 Contrastación de la hipótesis con los resultados

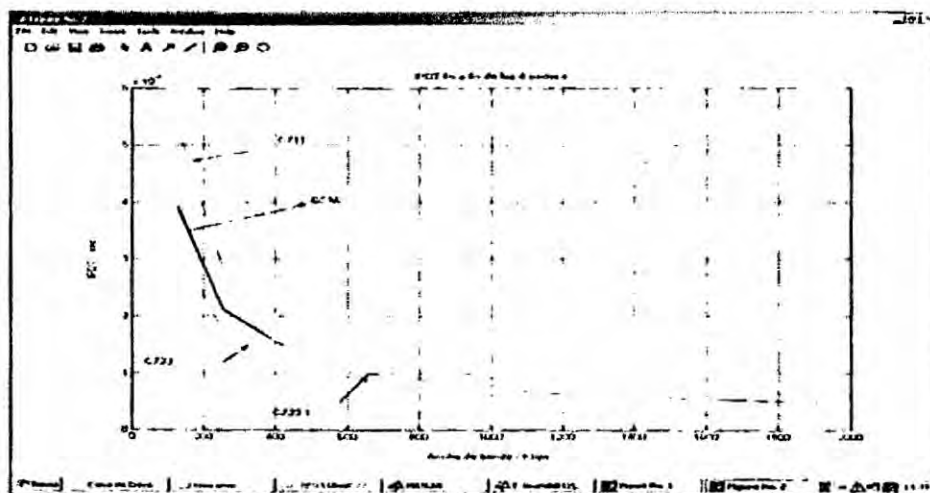
Uno de los parámetros que causa mayor desconforto al escuchar una conversación es sin duda el atraso del audio al ser recibido por el receptor, después de que el emisor ha iniciado su habla. Se observó que para un IPDT de 200 mseg., el atraso inicial en recibir el audio es percibido casi nulo y la recepción no presenta interrupciones. Cuando se aumenta a 300 mseg se percibe el atraso inicial a partir de 280 ms. Para un IPDT de 400 mseg. la espera inicial aumenta a 600 ms, comenzando a producir desconforto. Finalmente para un IPDT de 2000 mseg el atraso inicial se vuelve muy incómodo, mismo que el audio se vuelve continuo después del atraso. Las pruebas con IPDT fueron hechas con valores muy superiores a los practicados normalmente en telefonía, pero fueron hechas considerando

que puede ocurrir en grandes congestionamientos, en desastres o conmoción social.

En el gráfico 6.1 el códec G729 presenta el valor más bajo de IPDT de aproximadamente 1.8 mseg (muy bueno) en relación al límite superior de 150 mseg. Para una banda un poco mayor que 256 Kbps, tornándolo muy interesante para estas aplicaciones de voz.

GRAFICO 6.1

IPDT DE VOZ FIN-A-FIN DE LOS 4 CODECS EN FUNCIÓN DE LA BANDA

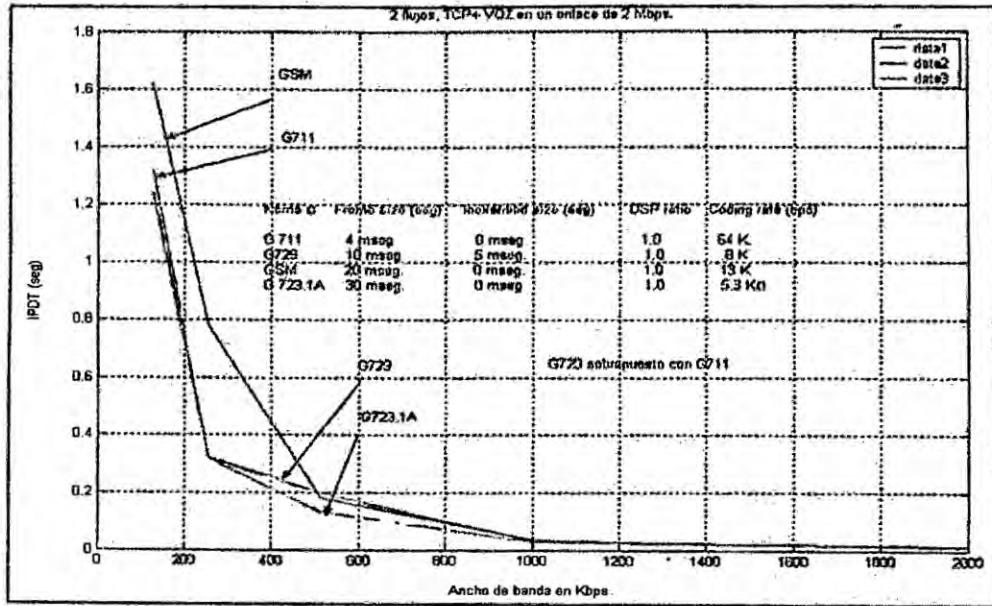


FUENTE: SIMULADOR

El gráfico 6.2 muestra el comportamiento del IPDT de los 4 codecs para el caso de voz y FTP simultáneo en función del ancho de banda del enlace; note que para hasta 256 Kbps de banda el IPDT del canal de voz es aproximadamente 400 ms, muchas veces superior al anterior valor de 1.8 ms. y por tanto, fuera de los límites especificados.

GRÁFICO. 6.2

IPDT PARA LOS 4 CODECS DE VOZ + FTP EN UN ENLACE DE HASTA 2 MBPS



FUENTE: SIMULADOR

TABLA 6.3

CARACTERÍSTICAS DE LOS CODECS UTILIZADOS

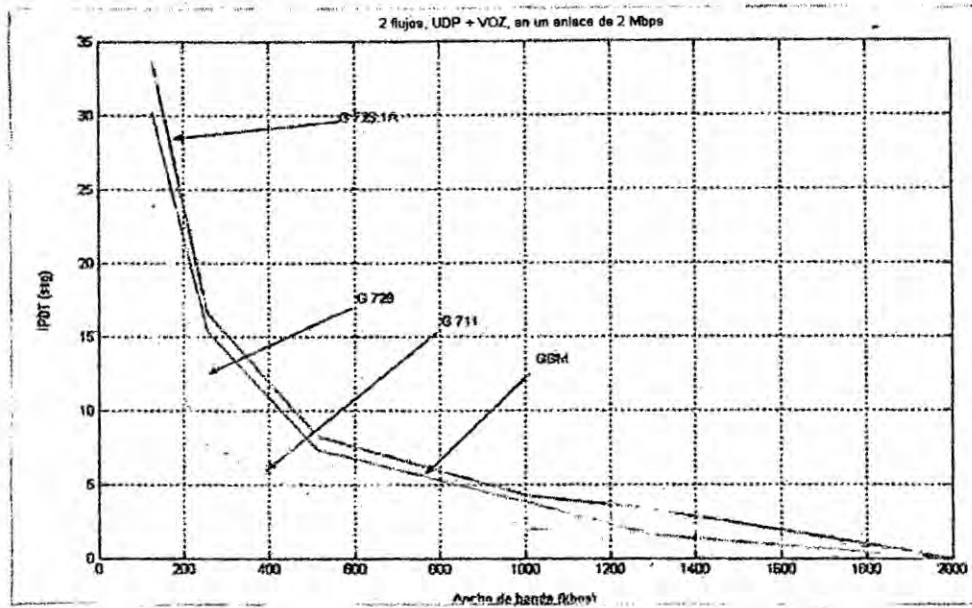
Codec	Frame size (seg)	Lookahead Size (seg)	DSP ratio	Coding rate (bps)
G711	0,004	0,00	1.0	64K
G729	0,010	0.005	1.0	8K
GSM	0,020	0,00.	1.0	13 K
G 723.1*	0.030	0.00	1,0	5.3 K

FUENTE: SIMULADOR

El grafico 6.3 muestra el comportamiento del IPDT cuando opera simultáneamente un flujo de vídeo y uno de voz en el enlace de hasta 1.5 Mbps; note que el valor para 256 Kbps es de aprox. 8 seg., muy superior al anterior de 400 miliseg.

GRÁFICO. 6.3

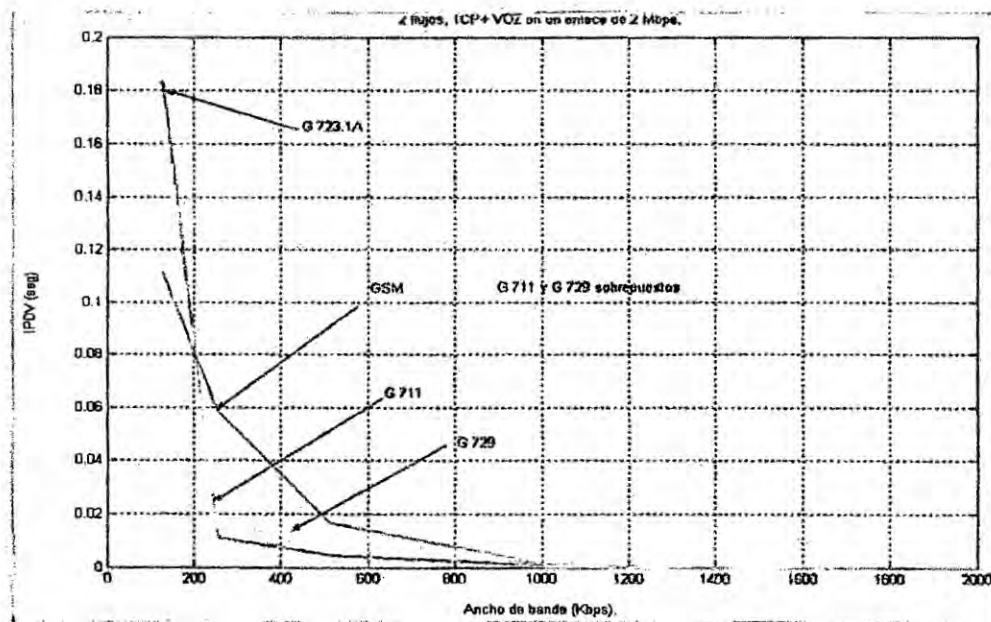
IPDT PARA VOZ + VIDEO EN FUNCIÓN DE LA BANDA



FUENTE: SIMULADOR

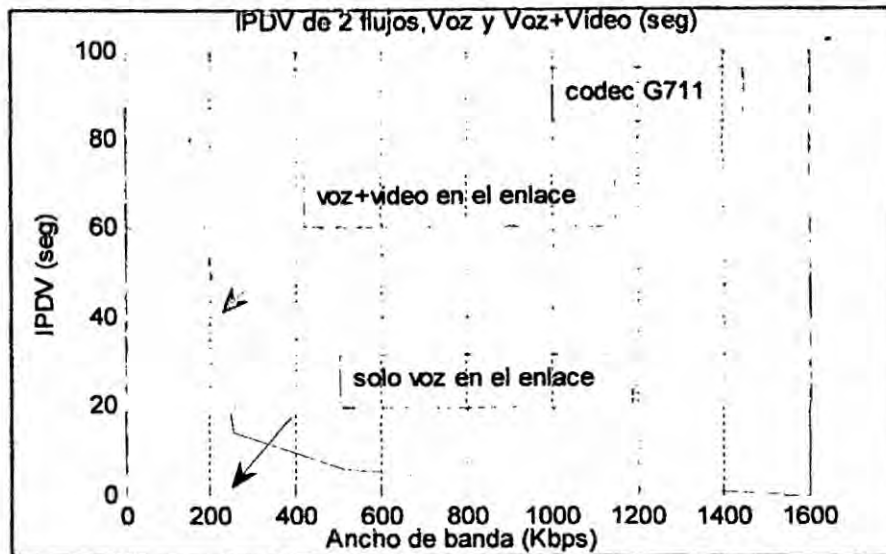
GRÁFICO. 6.4

IPDV DE 2 FLUJOS (VOZ + FTP) EN FUNCIÓN DEL ENLACE



FUENTE: SIMULADOR

GRÁFICO. 6.5
IPDV DE 2 FLUJOS (VOZ + UDP) EN FUNCIÓN DEL ENLACE



FUENTE: SIMULADOR

Debe observarse que la solución más simple es el aumento del ancho de banda, para esto es necesario analizar en conjunto con el costo adicional o programar los usos de los canales, o también como veremos en el ejemplo anexo, mudar el tipo de servicio "best effort" por otro servicio con protocolos mas elaborados y eficientes.

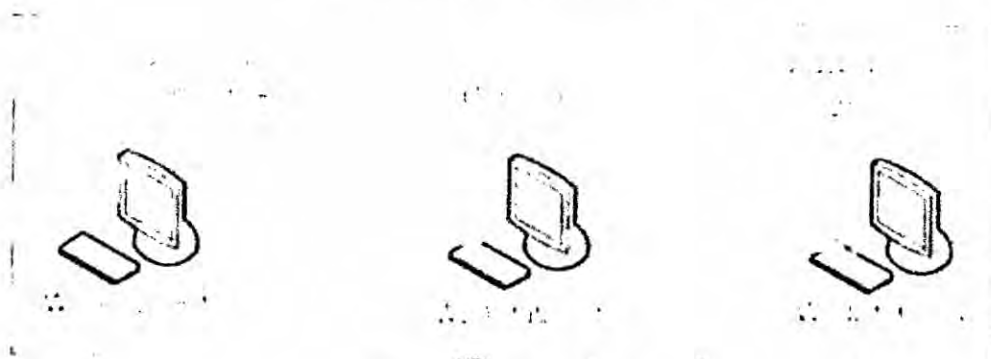
Los datos hasta aqui presentados confirman nuestra hipótesis en lo relacionado a IPDT:

6.2 Contratación de los resultados con trabajos anteriores.

Varios estudios anteriores han abordado este tema con enfoques diferentes cada uno de ellos, tanto a nivel interno de la Escuela Politécnica, como externo a ella. El mas parecido con el nuestro es el indicado a continuación pero que no llega a ninguna conclusión concreta y los valores

experimentados están muy fuera de la realidad como por ejemplo experiencias con valores de atraso de la orden de 2000 ms, si sabemos que la norma limita em 150 a 200 ms. En el ámbito interno se utiliza el modelo de caracterización de los parámetros de **QoS** y la fórmula de cálculo de la propia **QoS**. Estos estudios usaran un *testbed* cuya topología se muestra en la Fig.6.1

FIGURA. 6.1
TESTBED DE TRABAJOS ANTERIORES



FUENTE: PROPIA DEL AUTOR

En esta topología no son empleados *routers*, sino **PCs** emulando *routers* con el software emulador *NistNet*, un *software* que permite a un ordenador con sistema operativo *Linux* ser configurado como un *router* y emular muchas de las condiciones de congestión de la red, mediante el control de varios parámetros, como banda disponible, *delay*, *jitter* y pérdida de paquetes. Externamente, podrá ser citado el software *Mview* de la Universidad de Columbia, Nueva York, una herramienta que le permite ver el funcionamiento de la red **Mbone** (*Multicast Backbone*) a través del uso de diversas herramientas de red independientes que recogen informaciones topológicas y estadísticas de rendimiento.

Los detalles se pueden encontrar en el documento "**RTP/RTCP** extensión de **Mview-Mbone** herramienta de seguimiento" o correo electrónico **mk@comet.columbia.edu**, dirigido a Michael E. Kounaves. *Mview* se instala bajo *Solaris 2.5.1* y requiere la versión 4.1 o superior de **Tcl/Tk**.

CAPÍTULO VII

7. CONCLUSIONES

Actualmente la prueba de calidad de la voz es resultado de una percepción subjetiva y de un proceso de juzgamiento durante el cual, un “escucha”, compara el evento perceptual, con una referencia interna que es juzgada a ser de buena calidad.

Resumiendo, este proceso incluye:

El promedio de la puntuación dadas por los “escuchas”, el cual es conocido como **MOS** (*Mean Opinión Score*), o como sugerido por la recomendación P.800.1 del ITU-T, **MOS-LQS** (*Listening Quality Subjective*). Las pruebas subjetivas, sin embargo son costosas y consumen mucho tiempo, tornándolas inadecuadas para aplicaciones *on-the-fly* (sobre la marcha) real time.

En las mediciones objetivas de calidad de la voz, el panel de “escuchas” es remplazado por un algoritmo computacional facilitando las mediciones automáticas de aplicaciones *real-time*. Para comunicaciones **VoIP**, los métodos objetivos son clasificados en 2 tipos: 1) basados en señales y 2) basados en parámetros. Los métodos basados en señales usan características perceptuales extraídas de las señales del habla para estimar la calidad. Los métodos basados en parámetros usan los parámetros de conexión **VoIP**, tales como codecs, pérdida de paquetes, *jitter*, *delay*, etc., para computar el deterioro de los factores que son usados para determinar la calidad del habla. Este segundo método, es justamente el caso de este

trabajo de investigación (objetivo paramétrico) con los parámetros de calidad obtenidos vía el protocolo **RTP/RTCP** que es presentado en una metodología simple para monitoreo y restauración de la calidad de servicio sobre aplicaciones de voz. La validación visa atender a esta etapa del trabajo realizando el teste del algoritmo de forma separada del protocolo **RTP**, pero verificando todas y cada una de las actividades que serían efectuadas con el protocolo mencionado.

.Se visualiza una mejoría en el modelo de servicio tradicional de las redes **IP**, conocido como "*best effort*", que sería conseguido con su implementación. Algunos resultados complementares fueron obtenidos en el transcurso de este trabajo y aunque no hacen parte del objetivo fueron brevemente comentados. Las tablas 5.20 a 5.37 muestran los valores de **IPDV** e **IPDT** y consecuentemente **QoS**, obtenidos con equipos "reales" (*routers Cisco*) que se muestran un poco más altos que los valores simulados para la misma situación (único en el enlace, por ejemplo). Similarmente se observa que para la misma banda el tráfico **UDP** presenta valores mayores de **IPDV** que el tráfico interferente **TCP**, sin embargo, esto es entendible si consideramos la naturaleza y funcionamiento de los protocolos de transporte. El **UDP** es más crítico y rígido que el **TCP**. El tema mantiene su relevancia mismo cuando comparado con otras propuestas como las resumidas a seguir: Analizando el enfoque de esos trabajos, efectuados por otros investigadores y para uso con aplicaciones en tiempo real (en la *Internet*), podremos observar lo siguiente. La primera experiencia

implementada en el **IVS** (*Institut National de Recherche en Informatique et en Automatique*) **INRIA** (*Videoconference System*), ejecutando sobre el protocolo **RTP/RTCP** y basado en el valor medio de la tasa de pérdida de paquetes de los receptores. Este algoritmo presenta dos inconvenientes, 1) evalúa la red solamente por la tasa de pérdida de paquetes y 2) aumenta su velocidad mediante un incremento "X" sin considerar la banda disponible. Cuando este aumento es inadecuado, la tasa de envío puede oscilar ampliamente ocasionando una grande pérdida de paquetes a otras aplicaciones que comparten el mismo enlace. Otro algoritmo, intenta ajustar la velocidad de acuerdo con la variación del **RTT** medido antes del aumento de la pérdida de paquetes. Aumenta o reduce la velocidad en valores pequeños dependiendo si el **RTT** está abajo o por encima de un límite previamente definido. Aumento de la velocidad es lo mismo que aumento de ancho de banda y cae en uno de los casos perjudiciales anteriores pues significa un aumento de banda que puede no estar disponible. Un tercer algoritmo, conocido como **RAP** (*Rate Adaptation Protocol*), usa un aumento aditivo y una disminución multiplicativa. Durante la congestión, inmediatamente baja la velocidad para la mitad y la va aumentando aditivamente uno por uno hasta comenzar a perder paquetes donde para. Procede como en el caso de **TCP**: Su limitación en aplicaciones de tiempo real se encuentra en el requerimiento de la confirmación de cada paquete transmitido. Debido a que su velocidad de transmisión depende de la pérdida de paquetes, esta puede oscilar de forma parecida al **TCP** y una

grande oscilación puede ocasionar un grande *jitter* perjudicando así la calidad de la voz y vídeo. Es importante mencionar que además del algoritmo, las actividades de simulación y análisis de tráfico interferente **TCP y UDP**; este trabajo, visa mostrar y ayudar a aclarar las dudas sobre las penalidades ocasionadas por situaciones de congestión con tráfico igual o diferente del tráfico principal de voz (**UDP**). Esta forma facilita la visualización del mecanismo propuesto, simple y de bajo costo. Sin duda alguna, si fuese posible disponer de más recursos reales y facilidad de acceso a las redes comerciales o a laboratorios especializados, completaríamos de otra forma las conclusiones aquí presentadas.

Sin la pretensión de haber completado el asunto, ni considerarlo con el rigor científico necesario correspondiente, se puede observar en los resultados mostrados en las tablas 5.1 a 5.18 en lo que dice respecto a las métricas de QoS, **IPDV** e **IPDT** y en las figuras 5.1 a 5.11, lo siguiente:

- Un flujo de voz, codificado por cualquier uno de los *codecs* G.711, G.729, **GSM** o G.723.1 A, cuando es transmitido solo en un enlace de 128 Kbps, no sufre ninguna degradación significativa (como era de esperar).
- Cuando un flujo **TCP** entra para competir con el flujo de voz en el mismo enlace se nota que si la banda "residual" es insuficiente para el tráfico **TCP**, fuerza la conmutación del flujo de voz para el *codec* siguiente en orden decreciente y así sucesivamente hasta el *codec* de menor banda, esperando que en al final la situación de congestión finalice, para iniciar el retorno hasta el *codec default*. Los valores obtenidos indican que la

banda residual ($128000 - 82800 = 45200$ bps) es insuficiente para el tráfico FTP analizado, pero se transmite al costo del aumento de las métricas de QoS, *delay* y *jitter*.

- Una situación similar a la anterior fuerza la conmutación cuando un segundo flujo, en este caso UDP de vídeo, compite con el flujo de voz, pero de una forma más rígida que en el caso del flujo TCP del FTP.
- Considerando que solamente era previsto la propuesta del algoritmo y otros datos surgieron durante la preparación de este documento decidimos incluir los casos más significativos. Queda en fase de evaluación el caso de la determinación del punto en el cual el tráfico adicional deja de interferir en el flujo de voz, es decir, cual es el ancho de banda mínimo residual que permite al tráfico interferente TCP y UDP viajar simultáneamente sin provocar aumento de los parámetros de QoS del flujo de voz, para los codecs utilizados. Quedara para el proximo trabajo en esta área.
- El resultado de la validación de la simulación muestra adicionalmente, lo siguiente: La experiencia efectuada con equipos reales, muestran un resultado mas estable que el simulado. El simulado conmuta para protección 7 veces (33% de continuidad, y el real ninguna vez (100% de continuidad).

Lo máximo que se ha llegado en la parte real es a la situación de "red cargada". Es probable que exista algún parámetro siendo influenciado físicamente o configurado no correctamente en la red de pruebas utilizada.

De cualquier forma será investigado en la primera oportunidad disponible. Como conclusión final podemos afirmar que el procedimiento propuesto se muestra adecuado a los propósitos de esta tesis y se recomienda continuar investigando en esta línea con otros trabajos nuevos o complementarios.

TABLA 7.1
EJERCICIO EXTRA: RESULTADO DE DE SIMULACIÓN (1)

Codec	IPDT	QoS1	IPDV	QoS2	QoS1 + 2 TOTAL	Acción
128K solo voz						
G711	0,0121	0,9193	0,0000	0,0000	0,9193	Mantiene
G729	0,0025	0,9833	0,0000	0,0000	0,9833	Miene
gsm	0,0040	0,9733	0,0000	0,0000	0,9733	Mantiene
g723.1A	0,0032	0,9787	0,0000	0,0000	0,9787	Mantiene
128K Voz + TCP						
4.2	1,4177	-8,4513	0,5780	-10,5600	-19,0113	Conmuta
	1,0692	-6,1280	0,3274	-5,5480	-11,6760	Conmuta
	0,0693	0,5380	0,3274	-5,5480	-5,0100	Conmuta
	0,9314	-5,2093	0,2338	-3,6760	-8,8853	Conmuta
128K-Voz+UDP						
4.3	31,6115	-209,7433	96,4832	1928,6640	-2138,4073	Conmuta
	23,8992	-158,3280	54,8107	1095,2140	-1253,5420	Conmuta
	1,0693	-6,1287	0,3274	-5,5480	-11,6767	Conmuta
	0,9314	-5,2093	0,2338	-3,6760	-8,8853	Conmuta
256K-solo voz						
4.4	0,0063	0,9580	0,0000	0,0000	0,9580	Mantiene
	0,0013	0,9913	0,0000	0,0000	0,9913	Mantiene
	0,0040	0,9733	0,0000	0,0000	0,9733	Mantiene
	0,0017	0,9887	0,0000	0,0000	0,9887	Mantiene
256k-Voz+TCP						
4.5	0,6012	-3,0080	0,0041	0,9188	-2,0892	Conmuta
	0,4653	-2,1020	0,0709	-0,4180	-2,5200	Conmuta
	0,4653	-2,1020	0,0709	-0,4180	-2,5200	Conmuta

	0,4316	-1,8773	0,0118	0,7640	-1,1133	Conmuta
256k- Voz + UDP						
4.6	15,6901	-103,6007	25,1247	-501,4940	-605,0947	Conmuta
	11,8181	-77,7873	14,1134	-281,2680	-359,0553	Conmuta
	11,8181	-77,7873	14,1134	-281,2680	-359,0553	Conmuta
	16,6187	-109,7913	27,8066	-555,1320	-664,9233	Conmuta

EJERCICIO EXTRA: RESULTADO DE SIMULACIÓN (2)

512K-Solo voz						
4.7	0,0033	0,9780	0,0000	0,0000	0,9780	Mantiene
	0,0008	0,9947	0,0000	0,0000	0,9947	Mantiene
	0,0008	0,9947	0,0000	0,0000	0,9947	Mantiene
	0,0009	0,9940	0,0000	0,0000	0,9940	Mantiene
512K- Voz + TCP						
4.8	0,2284	-0,5227	0,0047	0,9060	0,3833	Mantiene
	0,1424	0,0507	0,0048	0,9040	0,9547	Mantiene
	0,1424	0,0507	0,0048	0,9040	0,9547	Mantiene
	0,0695	0,5367	0,0022	0,9560	1,4927	Mantiene
512K-Voz+ UDP						
4.9	7,7013	-50,3420	5,9990	-118,9800	-169,3220	Conmuta
	5,8033	-37,6887	3,3497	-65,9940	-103,6827	Conmuta
	5,8033	-37,6887	3,3497	-65,9940	-103,6827	Conmuta
	8,2459	-53,9727	6,9656	-138,3120	-192,2847	Conmuta
1000K-Solo voz						
4.10	0,0019	0,9873	0,0000	0,0000	0,9873	Mantiene
	0,0005	0,9967	0,0000	0,0000	0,9967	Mantiene
	0,0005	0,9967	0,0000	0,0000	0,9967	Mantiene
	0,0006	0,9960	0,0000	0,0000	0,9960	Mantiene
1000K-voz+TCP						
4.11	0,0318	0,7880	0,0000	0,0000	0,7880	Mantiene
	0,0292	0,8053	0,0000	0,0000	0,8053	Mantiene
	0,0291	0,8060	0,0000	0,0000	0,8060	Mantiene
	0,0285	0,8100	0,0000	0,0000	0,8100	Mantiene
1000K- Voz + UDP						
4.12	3,9221	-25,1473	1,5950	-30,9000	-56,0473	Conmuta
	2,8896	-18,2640	0,8626	-16,2520	-34,5160	Conmuta

	2,8896	-18,2640	0,8626	-16,2520	-34,5160	Conmuta
	4,1801	-26,8673	1,8189	-35,3780	-62,2453	Conmuta
1300K-Solo voz						
4.13	0,0016	0,9893	0,0000	0,0000	0,9893	Mantiene
	0,0004	0,9973	0,0000	0,0000	0,9973	Mantiene
	0,0004	0,9973	0,0000	0,0000	0,9973	Mantiene
	0,0005	0,9967	0,0000	0,0000	0,9967	Mantiene

EJERCICIO EXTRA: RESULTADO DE DE SIMULACIÓN (3)

1300K- voz + TCP						
4.14	0,0256	0,8293	0,0000	0,0000	0,8293	Mantiene
	0,0223	0,8513	0,0000	0,0000	0,8513	Mantiene
	0,0223	0,8513	0,0000	0,0000	0,8513	Mantiene
	0,0223	0,8513	0,0000	0,0000	0,8513	Mantiene
1000K- Voz+UDP						
4.15	3,1845	-20,2300	1,0690	-20,3800	-40,6100	Conmuta
	2,4232	-15,1547	0,6038	-11,0760	-26,2307	Conmuta
	2,4232	-15,1547	0,6038	-11,0760	-26,2307	Conmuta
	0,0220	0,8533	0,0020	0,9600	1,8133	Mantiene
DS1- Solo voz						
4.16	0,0012	0,9920	0,0000	0,0000	0,9920	Mantiene
	0,0003	0,9980	0,0000	0,0000	0,9980	Mantiene
	0,0003	0,9980	0,0000	0,0000	0,9980	Mantiene
	0,0004	0,9973	0,0000	0,0000	0,9973	Mantiene
DS1-Voz + TCP						
4.17	0,0143	0,9047	0,0001	0,9974	1,9021	Mantiene
	0,0135	0,9100	0,0001	0,9980	1,9080	Mantiene
	0,0135	0,9100	0,0001	0,9980	1,9080	Mantiene
	0,0136	0,9093	0,0001	0,9980	1,9073	Mantiene
DS1- Voz + UDP:						
4.18	0,0192	0,8720	0,0030	0,9400	1,8120	Mantiene
	0,0020	0,9867	0,0003	0,9940	1,9807	Mantiene
	0,0203	0,8647	0,0003	0,9940	1,8587	Mantiene
	0,0189	0,8740	0,0003	0,9940	1,8680	Mantiene

FUENTE: SIMULADOR

TABLA 7.2

EJERCICIO EXTRA: RESULTADO DE DE PRUEBA "REAL" (1)

CODEC	IPDT	QOS1	IPDV	QOS2	TOTAL	ACCION
VELOCIDAD/FLUJO/TABLA²						
128KBPS/VOZ/ 5.20						
G711	0.0155	0.4483	0.0011	0.4890	0.9373	mantiene
G729	0.0110	0.4633	0.0012	0.4880	0.9433	mantiene
GSM	0.0095	0.4683	0.0020	0.4800	0.9433	mantiene
G723.1A	0.0080	0.4733	0.0023	0.4770	0.9503	mantiene
128K/ Voz+TCP/ 5.21						
G711	0.0690	0.2700	0.0004	0.5000	0.7700	mantiene
G729	0.0830	0.2233	0.0044	0.4560	0.6793	mantiene
GSM	0.0660	0.2800	0.0074	0.4260	0.7060	mantiene
G723.1A	0.0750	0.2500	0.0295	0.2050	0.4550	mantiene
128K/ Voz+UDP/ 5.22						
G711	0.0072	0.4760	0.0038	0.4620	0.9320	mantiene
G729	0.0072	0.4760	0.0038	0.4620	0.9320	mantiene
GSM	0.0072	0.4760	0.0038	0.4620	0.9320	mantiene
G723.1A	0.0072	0.4760	0.0038	0.4620	0.9320	mantiene
256/Voz/ 5.23						
G711	0.0060	0.4800	0.0029	0.4710	0.9510	mantiene
G729	0.0035	0.4883	0.0010	0.4900	0.9783	mantiene
GSM	0.0030	0.4900	0.0010	0.4900	0.9800	mantiene
G723.1A	0.0030	0.4900	0.0010	0.4900	0.9800	mantiene
256/Voz+TCP/ 5.24						
G711	0.0041	0.4863	0.0027	0.4730	0.9593	mantiene
G729	0.0037	0.4876	0.0041	0.4590	0.9466	mantiene
GSM	0.0035	0.4883	0.0051	0.4490	0.9373	mantiene
G723.1A	0.0035	0.4883	0.0091	0.4090	0.8973	mantiene
256/Voz+UDP/ 5.25						
G711	0.0370	0.3766	0.0067	0.4330	0.8096	mantiene
G729	0.0370	0.3766	0.0067	0.4330	0.8096	mantiene
GSM	0.0370	0.3766	0.0067	0.4330	0.8096	mantiene
G723.1A	0.00370	0.3766	0.0067	0.4330	0.8096	mantiene

FUENTE: RUTEADORES

² En la segunda línea de esta tabla: velocidad/flujo/tabla indican la velocidad del enlace, el flujo de Voz (Voz, Voz+FTP O Voz+UDP) y tabla indican el N° de la tabla del capítulo V que origino estos datos. En este caso primer calculo de la tabla %0 del capítulo V.

EJERCICIO EXTRA: RESULTADO DE PRUEBA REAL (2)

512/Voz/ 5.26						
G711	0.0015	0.4966	0.4910		* -7.8300	cambia
G729	0.0020	0.4933	0.4020	0.4800	*- 6.053	cambia
GSM	0.0010	0.4966	0.0010	0.4900	0.9866	mantiene
G723.1A	0.0015	0.4966	0.0060	0.4400	0.9366	mantiene
512/Voz+TCP/ 5.27						
G711	0.0140	0.4533	0.0015	0.4850	0.9383	mantiene
G729	0.0165	0.4450	0.0031	0.4690	0.9140	mantiene
GSM	0.0150	0.4500	0.0043	0.4570	0.9070	mantiene
G723.1A	0.0125	0.4583	0.0053	0.4470	0.9053	mantiene
512/Voz+UDP/ 5.28						
G711	0.0190	0.4366	0.0066	0.4340	0.8706	Mantiene
G729	0.0190	0.4366	0.0066	0.4340	0.8706	mantiene
GSM	0.0190	0.4366	0.0066	0.4340	0.8706	mantiene
G723.1A	0.0190	0.4366	0.0066	0.4340	0.8706	mantiene
1000 k/Voz/ 5.29						
G711	0.0005	0.4983	0.0010	0.4900	0.9883	Mantiene
G729	0.0005	0.4983	0.0080	0.4200	0.9183	Mantiene
GSM	0.0005	0.4983	0.0005	0.4983	0.9966	Mantiene
G723.1A	0.0005	0.4983	0.0012	0.4880	0.9863	Mantiene
1000 k/Voz+TCP/ 5.30						
G711	0.0045	0.4850	0.0060	0.4400	0.9250	Mantiene
G729	0.0060	0.4800	0.0060	0.4400	0.9200	Mantiene
GSM	0.0060	0.4800	0.0060	0.4400	0.9200	Mantiene
G723.1A	0.0030	0.4900	0.0060	0.4400	0.9300	Mantiene
1000 k/Voz+UDP/ 5.31						
G711	0.0105	0.4650	0.0048	0.4520	0.9170	Mantiene
G729	0.0105	0.4650	0.0048	0.4520	0.9170	Mantiene
GSM	0.0105	0.4650	0.0048	0.4520	0.9170	Mantiene
G723.1A	0.0105	0.4650	0.0048	0.4520	0.9170	Mantiene
1300 k/Voz/ 5.32						
G711	0.0050	0.4833	0.0014	0.4860	0.9693	Mantiene
G729	0.0050	0.4833	0.0023	0.4770	0.9600	Mantiene
GSM	0.0050	0.4833	0.0027	0.4730	0.9563	Mantiene
G723.1A	0.0050	0.4833	0.0018	0.4820	0.9653	Mantiene

FUENTE: PROPIA DEL AUTOR

EJERCICIO EXTRA: RESULTADO DE PRUEBA REAL (3)

1300 k/Voz+TCP/ 5.33						
G711	0.0040	0.4866	0.0052	0.4480	0.9346	Mantiene
G729	0.0035	0.4883	0.0046	0.4540	0.9423	Mantiene
GSM	0.0015	0.4950	0.0020	0.4800	0.9750	Mantiene
G723.1A	0.0015	0.4950	0.0016	0.4840	0.9790	Mantiene
1300 k/Voz+UDP/ 5.34						
G711	0.0050	0.4833	0.0003	0.5000	0.9833	Mantiene
G729	0.0050	0.4833	0.0012	0.4880	0.9713	Mantiene
GSM	0.0050	0.4833	0.0049	0.4500	0.9333	Mantiene
G723.1*	0.0050	0.4833	0.0012	0.4880	0.9713	Mantiene
1544k/Voz/ 5.35						
G711	0.0025	0.4916	0.0010	0.4900	0.9816	Mantiene
G729	0.0020	0.4933	0.0075	0.4250	0.9183	Mantiene
GSM	0.0020	0.4933	0.0015	0.4850	0.9783	Mantiene
G723.1A	0.0020	0.4933	0.0001	0.5000	0.9933	Mantiene
1544k/Voz+TCP / 5.36						
G711	0.0025	0.4916	0.0010	0.4900	0.9816	Mantiene
G729	0.0020	0.4933	0.0390	0.1100	0.6033	Mantiene
GSM	0.0020	0.4933	0.0015	0.4850	0.9783	Mantiene
G723.1A	0.0020	0.4933	0.0010	0.4900	0.9833	Mantiene
1544k/Voz+UDP / 5.37						
G711	0.0055	0.4816	0.0085	0.4150	0.8960	Mantiene
G729	0.0055	0.4816	0.0085	0.4150	0.8960	Mantiene
GSM	0.0055	0.4816	0.0085	0.4150	0.8960	Mantiene
G723.1A	0.0055	0.4816	0.0085	0.4150	0.8960	Mantiene

FUENTE: PROPIA DEL AUTOR

Los ejercicios de las tablas 7.1, 7.2 son anexados para fin de ilustración de los resultados y cálculos de QoS.

CAPÍTULO VIII

8. RECOMENDACIONES

Las redes de comutación de paquetes actualmente ofrecen servicio *best effort* en una base de primer paquete que llega primero a ser atendido (FIFO) y de esta forma el atraso puede ser altamente variable. Esto puede no afectar al tráfico de datos, debido a que estos son transmitidos en ráfagas e insensibles al atraso, y cualquier pérdida de paquetes puede ser fácilmente retransmitida sin cualquier restricción de límite de tiempo. Sin embargo estas características no son aceptables por el tráfico "real time," Siendo la voz y video dos ejemplos de este tipo de tráfico. Otro aspecto es que la evaluación de la calidad de la voz es hecha principalmente de forma subjetiva en la cual se pide a un grupo de personas para escuchar algunos trechos de conversaciones y después expresar su opinión sobre la calidad. El otro método emergente, es el conocido como método "objetivo" que en su modalidad paramétrica analiza las métricas antes mencionadas, (atraso, variación de atraso, perdida de paquetes, eco, etc), Siendo este el cuadro actual de las telecomunicaciones, y VoIP considerado el sustituto de la telefonía tradicional, es necesario una actualización en los nuevos conceptos y técnicas de telecomunicaciones. Anteriormente, la red telefónica era el grande exponente de esta área y hoy en día son las redes de paquetes IP, redes *frame relay*, redes ópticas, etc. y principalmente la *Internet*.

De aquí nuestra recomendación a quienes participan en esta área sea industrial o académica, a actualizar sus conceptos y particularmente a los de la área académica, a participar más activamente con trabajos de investigación, publicación de trabajos de graduación/Post graduación o tesis de forma general. Como mencionado anteriormente, no consideramos el caso como concluido, pero si consideramos los resultados obtenidos altamente promisorios y recomendamos continuar en esta línea de investigación con trabajos complementares.

La continuidad de trabajos en esta área es facilitada por ser justamente un área con mucho dinamismo visto la gran cantidad de tesis presentadas y las mudanzas tecnológicas ocurriendo cada vez en menor espacio de tiempo. Específicamente, por ejemplo, este trabajo enfoca un tema de **QoS** usando el protocolo **frame relay** como protocolo de enlace WAN. Un proyecto alternativo de pruebas podría incluir SDLC o *Ethernet*, como protocolo WAN etc, o utilizar una topología diferente .en lugar del PPP. El campo es muy extenso.

CAPÍTULO IX

9. REFERENCIAS BIBLIOGRÁFICAS

- [1] ITU-T E-800. **"Terms and Definitions Related to Quality of service and network performance including dependability"**. Agosto de 1994.
- [2] ETSI TIPHON TS101 329-5 Annex E, **"Method for determining an equipment impairment factor using passive monitoring"**. Noviembre de 2000.
- [3] RFC 2386. A **"Framework for QoS-based Routing in the Internet."** Agosto de 1998.
- [4] ITU-T I. 350. **"Aspectos Gerais da qualidade de serviço e da qualidade de funcionamento nas redes digitais, incluindo as redes digitais de serviços integrados"**. Marzo de 1993.
- [5] ITU-T G.107. **"O modelo E, um modelo informático para utilização em planejamento da transmissão"**. Marzo de 2003.
- [6] BELLIDO, L.; FERNANDEZ, D.; PASTOR, E. **"Architectural Issues for Multimedia Cooperative Systems. Proceedings of the 3 International Workshop on Protocols for Multimedia Systems (PROMS)"**. Págs. 33-47. Octubre de 1996.
- [7] RFC 1889. **"RTP: "A transport protocol for real-time applications"**. Enero 1996.

- [8] BARRA, C. **"Caracterização experimental, por simulação e modelagem da Qualidade de Serviço obtida na transmissão de áudio e vídeo em tempo real."** Tese de Doutorado, EPUSP (2005).
- [9] TAKAHASHI, A. et al. **"Objective assessment Methodology for Estimating Conversational Quality in VoIP, IEEE transactions on audio, speech and language processing"**. Novembro de 2006.
- [10] CLARK, D.; SHENKER, S. AND ZHANG, L. **"Supporting real-time applications in an integrated services packet network Architecture and mechanism"**. ACM SIGCOMM'92". Agosto de 1992.
- [11] MARCHESE, M. **"QoS over heterogeneous networks"**. John Wiley & sons Ltd. 2007.
- [12] ITU-T. Y 1541, **"Recommendation Network Performance Objectives for IP-based Services"**, Febrero de 2003.
- [13] JAMES, J.H. et al. A **"Voice Transmission Performance Progress Report. IEEE Communication Magazine"**. Julio de 2004.
- [14] Tommi K. Protocol Overview: RTP and RTCP. Nokia, **"Telecommunications, paper"**. Junio 2003.
tommi.koistineu@ntc.nokia.com.
- [15] OPNET IT GURU, versão acadêmica 9.1 A. **"www.opnet.com"**.
- [16] lperf. LANR/DAST. **"http://dast.nlanr.net/"**.
- [17] SJPhone. **"www.sjlabs.com"**.

- [18] Ethereal – **“Network Protocol Analyzer”**. Disponível em: <http://www.ethereal.com> acessado em 22/03/2003.
- [19] ETSI TIPHON TS101 329-5 Annex E, **“Method for determining an equipment impairment factor using passive monitoring”**.
Noviembre de 2000.
- [20] ITU-T Recommendation Y.2011, **“General principles and general reference model for next generation networks”**. Octubre de 2004.
- [21] FERNANDEZ, D.; PASTOR, E.; BELLIDO, L. **“Experiences Integrating Cooperative Multimedia Applications into WWW”**. 2 CYTED-RITOS International Workshop on Groupware (CRIWG86), Puerto Varas, Chile. Octubre de 1996.
- [22] NIST **Network Simulator Versão 2.0. 12. 2003**. Disponível em <http://snad.ncsl.nist.gov/itg/nistnet/> acessado em 21/07/2005.
- [23] ITU-T Recommendation Y.2001, **“General overview of NGN.”**
Diciembre de 2004.
- [24] DUAN, Z.; ZHANG, Z. L.; HOU, Y. T. **“Service overlay networks: SLAs, QoS, and bandwidth provisioning. IEEE/ACM Transactions on Networking (TON)”**, volume 33 Issue 6. Diciembre de 2003.
- [25] HUSTON, G. RFC 2990. **“Next Steps for the IP QoS Architecture. Source Internet RFCs archive. 2000”**.
- [26] APOSTOLOPOULOS, G.; KAMA, S.; WILLIAMS, D. et al. RFC 2676. **“QoS Routing Mechanisms and OSPF Extensions”**. Agosto de 1999.

- [27] SNIR, Y.; RAMBERG, Y.; STRASSNER, J. et al. **"RFC 3644. Policy Quality of Service (QoS) Information Model"**. Noviembre de 2003.
- [28] RFC 3583. **"Requirements of a Quality of Service (QoS) Solution for Mobile IP"**. Septiembre de 2003.
- [29] MOORE, B.; DURHAM, D.; STRASSNER, J. et al. **"RFC 3670. Information Model for Describing Network Device QoS Datapath Mechanisms"**. Janeiro de 2004.
- [30] JIANG, S. Reviewed articles: **"Granular differentiated queueing services for QoS: structure and cost model. ACM SIGCOMM Computer Communication Review"**, volume 35 Issue 2. Abril de 2005.
- [31] CHANG, W. S.; SIMON, R. **"Performance Analysis for Multi-Service Networks with Congestion-Based Pricing for QoS Traffic. Proceedings of the 38th annual Symposium on Simulation"**. Abril de 2005.
- [32] YANG, S. J.; CHOU, H. C. **"Adaptive QoS parameters approach to modeling Internet performance. International Journal of Network Management"**, volume 13 Issue 1. Janeiro de 2003.
- [33] LINDEMANN, C.; LOHMANN, M.; THÜMLER, A. **"A unified approach for improving QoS and provider revenue in 3G mobile networks. Mobile Networks and Applications"**, volume 8 Issue 3. Junho de 2003.

- [34] TESANOVIC, A.; AMIRIJOO, M.; BJÖRK, M. et al. "Empowering configurable QoS management in real-time systems. Proceedings of the 4th international conference on Aspect-oriented software development". Março de 2005. Disponível em: <http://doi.acm.org/10.1145/1052902>.
- [35] LI, Z.; MOHAPATRA, P. "QoS-aware multicasting in diffserv domains. ACM SIGCOMM Computer Communication Review", volume 34 Issue 5. Outubro de 2004. <http://doi.acm.org/10.1145/1039111.1039112>.
- [36] ITO, Y.; TASAKA, S.; FUKUTA, Y. "Residential user characterisation: Psychometric analysis on the buffering control on user-level QoS in an interactive audio-visual application. Proceeding of the 2004 ACM workshop on Next-generation residential broadband challenges". Outubro de 2004.
- [37] VENKATASUBRAMANIAN, N.; TALCOTT, C.; AGHA, G. A. "A formal model for reasoning about adaptive QoS-enabled middleware. ACM Transactions on Software Engineering and Methodology (TOSEM)", volume 13 Issue 1. Janeiro de 2004.
- [38] THIO, N.; KARUNASEKERA, S. "Automatic Measurement of a QoS Metric for Web Service Recommendation. IEEE Computer Society. Source ASWEC archive. Proceedings of the 2005 Australian conference on Software Engineering table of contents", pages: 202 - 211. 2005. Disponível em: <http://portal.acm.org/citation.cfm?id=1050726.1050798>.

- [39] YAMAGUCHI, H.; FAKIH, K. E.; KIROMORI, A. et al. **A “formal approach to design optimized multimedia service overlay. Proceedings of the international workshop on Network and operating systems support for digital audio and video Stevenson”**, pages: 57 - 62. 2005. Disponível em: <http://doi.acm.org/10.1145/1065983.1065998>.
- [40] WIEMAN, A.; BALTER, M. H. **“Bandwidth sharing & scheduling: Classifying scheduling policies with respect to higher moments of conditional response time. Proceeding of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems”**. Junho de 2005.
- [41] WÜST, C. C.; STEFFENS, L.; VERHAEGH, W. F. et al. **“QoS Control Strategies for High-Quality Video Processing. Real-Time Systems”**, volume 30 Issue 1-2. Maio de 2005. Disponível em: 26.
- [42] ZIVIANI, A.; WOLFINGER, B. E.; REZENDE, J. F. et al. **“Adaption of QoS Schemes for MPEG Streams. Multimedia Tools and Applications”**. Volume 26 Issue 1. Maio de 2005.
- [43] MOBIOT, F.; SANZO, B.; GIRARD, A. **“Design of Reliable IP/GMPLS Networks: Na Integrated Approach. Journal of Network and Systems Management”**, volume 13 Issue 1 pages: 77-97. Março de 2005.
- [44] CHEN, I. R.; LI, S. T.; YEN, I. L. **“Adaptive QoS Control Base don Benefit Optimization for Video Servers Providing Differentiated**

- Services. Multimedia Tools and Applications**", volume 25 Issue 2
pages: 167-185. Fevereiro de 2005.
- [45] MATSUKAWA, T.; SATAKE, T.; YAMADA, J. **"Metrics for Detection of QoS Degradation over Wireless LAN. Proceedings of the 2005 Symposium on Applications and the Internet (SAINT'2005)"**, volume 00. Janeiro de 2005.
- [46] BOURAS, C.; GKAMAS, A.; PRIMPAS, D. et al. **"Performance Evaluation of the Impact of QoS Mechanisms in an IPv6 Network for IPv6-Capable Real-Time Applications"**. Journal of Network and Systems Management, volume 12 Issue 4. Dezembro de 2004.
- [47] Computational Management Science. Springer - Verlag GmbH, Issue: volume 1, number 1, pages: 1-14. Dezembro de 2003.
- [48] LEEL, I. H.; KIML, S. J. A **"QoS Improvement Scheme for Real-Time Traffic Using IPv6 Flow Labels. Lecture Notes in Computer Science"**, volume 3043/2004, pages: 278-285. Abril de 2004.
- [49] LIU, Q.; ZHOU, S.; GIANNAKIS, G. B. Cross-Layer **"Modeling of Adaptive Wireless Links for QoS Support in Multimedia Networks"**. **"Proceedings of the First International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QSHINE'04)"**, volume 00. Outubro de 2004.
- [50] LIU, D.; HU, X. S.; LEMMON, M. D. et al. **"Firm Real-Time System Scheduling Based on a Novel QoS Constraint. Proceedings of the**

24th IEEE International Real-Time Systems Symposium table of contents", page: 386. 2003.

- [51] SALAMATIAN, K.; FDIDA, S. **Measurements and Testbeds: A Framework for Interpreting Measurement over Internet. Proceedings of the ACM SIGCOMM workshop on Models, methods and tools for reproducible network research**". Agosto de 2003.
- [52] REISSLEIN, M.; RAJAGOPAL, S. A. **"Frame Work for Guaranteeing Statistical QoS. IEEE ACM Transaction on Networking"**, volume 10 N° 1. Fevereiro de 2002.
- [53] XIAO, L.; LUI, K. S.; WANG, J. et al. **"QoS Extension to BGP. Proceedings of the 10th IEEE International Conference on Network Protocols Full text available"**. Publisher. Novembro de 2002.
- [54] ITU-T P.800. **"Methods for subjective determination of transmission quality"**. Former Rec. P.80.
- [55] ITU-T P.830. **"Subjective performance assessment of telephone-band and wideband digital codecs"**.
- [56] ITU-T P.861. [Withdrawn] **"Objective quality measurement of telephone-band (300-3400 Hz) speech codecs P.861 was recognized as having certain limitations in specific areas of application. It was replaced by P.862, which contains an improved objective speech quality assessment algorithm. P.861 included one diskette containing test vectors for implementation verification of the PSQM algorithm.**

- [57] TAKAHASHI, A. et al. **"Perceptual QoS assessment technologied for VoIP. NTT labs, IEEE Communications Magazine"**. Julho de 2004.
- [58] FLOYD, S.; PAXSON, V. **"Difficulties in Simulating the Internet. IEEE ACM Transaction on Networking"**, volume 9 N° 4, pages: 392 - 402. Agosto de 2001.
- [59] SCHULTZRINNE, H. et al. RFC 3550. **RTP - A "Transport Protocol for Real Time Applications"**. Julho de 2003.
- [60] SCHULTZRINNE, H. et al. RFC 1890. **"RTP Profile for Audio and Video Conferences with Minimal Control"**. Janeiro de 1996.
- [61] SCHULTZRINNE, H. et al. RFC 3551. **"RTP Profile for Audio and Video Conferences with Minimal Control"**. Julho de 2003.
- [62] FRIEDMAN, T. et al. RFC 3611. **"RTP Control Protocol Extended Reports (RTCP XR)"**. Novembro de 2003.
- [63] WENGER, J. et al. **"RFC 4585. Extended RTP Profile for RTCP-based Feedback (RTP/AVPF)"**. Julho de 2006.
- [64] ITU-T. Y 1540, **"Recommendation IP Packet transfer and Availability Performance Parameters"**, Novembro de 2002.
- [65] HUA, S.; QU, G. A **"New Quality of Service Metric for Hard/Soft Real-Time Applications. Proceedings of the International Conference on Information Technology Computers and Communications"**. Abril de 2003.

- [66] RAYMOND, R. et al, "RTP/RTCP Extension of Mview - Mbone Monitoring Tool", 2002, www.comet.columbia.edu.
- [67] SCHULTZRINNE, H. et al. RFC 1889. "RTP - A transport Protocol for Real Time Applications. IETF". Janeiro de 1996.
- [68] LEE, W.; SRIVASTAVA, J.; SABATA, B. "QoS - Aware admission control and dynamic resource provisioning framework ubiquitous multimedia computing environments. Source the journal of supercomputing", of supercomputing archive volume 32, Issue 1, pages: 25 - 50. Janeiro de 2005.
- [69] ARMITAGE, G. J. "Revisiting IP QoS: why do we care, what have we learned? ACM SIGCOMM Computer Communication Review", volume 33 Issue 5. Outubro de 2003.
- [70] CLARK, A. "Modeling the effects of burst packet loss and recency on subjective voice quality. Proc. IP Telephony Workshop 2001". Abril de 2001.
- [71] COLE, R.G.; ROSENBLUTH, J. "Voice over IP performance monitoring. Proc. ACM SIGCOMM Computer Communication Review", pp. 9-24. Abril de 2001.
- [72] SUN, L.F.; WADE, G.; LINES, B. M. et al. "Impact of packet loss location on perceived speech quality. Proc. IP Telephony Workshop 2001". Abril de 2001.
- [73] LAKANIEMI, A.; ROSTI, J.; RAISANEN, I. "Subjective VoIP speech quality evaluation based on network measurements. Proc. IEEE

- International Conference on Communications (ICC 2001)", pp. 748-752. Junho de 2001.**
- [74] MASUDA, M.; HAYASHI, T. **"Non-intrusive quality monitoring method of VoIP speech based on network performance metrics. IEICE Trans. Commun".** vol. E89-B, nº. 2, pp. 304-312. Fevereiro de 2006.
- [75] FRIEDMAN, T.; CACERES, R.; CLark, A. **"RTP control protocol extended reports (RTCP XR). IETF RFC 3611".** 2003.
- [76] ITU-T **"Recommendation P.564, Conformance testing for narrowband voice over IP transmission quality assessment models".** Julho de 2006.
- [77] DE LEON, P.; Sreenan, C. **"An adaptive predictor for media playout buffering. Proc. IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)",** pp. 3097-3100. Março de 1999.
- [78] FUJIMOTO, K.; ATA, S.; MURATA, M. **"Adaptive playout buffer algorithm for enhancing perceived quality of streaming applications. Proc. IEEE Globecom",** pp. 2463-2469. 2002.
- [79] ITU-T Recommendation G.107, **"The E-model, a computational model for use in transmission planning".** Maio de 2000.
- [80] MASUDA, M.; ORI, K. **"Network performance metrics in estimating the speech quality of VoIP. Proc. IEICE Asia-Pacific Symposium on**

- Information and Telecommunication Technologies (APSITT 2001)", pp. 333-337. Novembro de 2001.**
- [81] MASUDA, M.; ORI, K. **"Estimating the speech quality of VoIP by using network performance metrics. Proc. IEICE and IEEE International Workshop on Communications Quality & Reliability (2002 CQR)", pp. 29-33. Maio de 2002.**
- [82] ITU-T Recommendation P.830, **"Subjective performance assessment of telephone-band and wide-band digital codecs".** Fevereiro de 1996.
- [83] ITU-T Recommendation P.861, **"Objective quality measurement of telephone-band (300-3400 Hz) speech codecs".** Agosto de 1996.
- [84] ITU-T Recommendation P.862, **"Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of 3.1kHz handset telephony (narrow-band) networks and speech codecs".** Fevereiro de 2001.
- [85] ITU-T Recommendation G.113, **"General characteristics of general telephone connections and telephone circuits - transmission impairments".** Fevereiro de 1996.
- [86] CLARK, A. **"Echo in voice over IP systems".** Janeiro, 27-2006.
- [87] MAHAJAN, M.; PARASHAR, M. **"Managing QoS for multimedia applications in the differentiated services environment. Journal of Network and Systems Management",** volume 11 Issue 4. Dezembro de 2003. Disponível em: <http://www.springerlink.com>.

- [88] MICHAEL, M.; DIMITROS, K. **"Voice Quality Monitoring and Control for VoIP. IEEE Computer Society"**, pages: 35-42 (Julho, Agosto de 2005).
- [89] COSTA, J. P. R. et al. **"An Environment for Management of End to End QoS Over I-Based Heterogeneous Networks"**. Relatório interno do projeto NQM do LCS. Outubro de 2002.
- [90] ITU-T Recommendation Y.2011, **"General principles and general reference model for next generation networks"**. Outubro de 2004.
- [91] Antunes N., Rocha R., Pinto P. (1997) **"Analysis and Simulation of a Traffic Management Control Scheme for ATM switches with Loose Commitments"**, Int. Conf. on Networks and Distributed Systems Modeling and Simulation", Phoenix, 1997,
<ftp://marcel.inesc.pt/pub/papers/cndsmisc97.ps.gz>.
- [92] Busse I., Deffner B., Schulzrinne H. (1995). **"Dynamic QoS Control of Multimedia Applications based on RTP, Computer Communications"**, Vol. 19, Number 1, Jan. 96.
- [93] Campbell A., Coulson G. (1996). **"A QoS Adaptive transport System: Design, Implementation and Experience"**, ACM Multimedia '96, Boston, 1996, 117-127.
- [94] Campbell A., Coulson G., Hutchison D. (1994). **"A Quality of Service Architecture"**, ACM SIGCOMM 94, Computer Communication Review, Vol.24, April 1994, 6-27.

- [95] Clark., Tennenhouse D. (1990). "**Architectural Considerations for a New generation of protocols**", ACM SIGCOMM 90, Philadelphia, 1990, 200-208.
- [96] Correia M. Pinto P. (1995) "**Low- Level Multimedia Synchronization Algorithms on Broadband Networks**" ACM Multimedia '95, San Francisco, 1995, 423-434, <ftp://mariel.inesc.pt/pub/papers/mm95.ps.gz>.
- [97] Garcia F., Hutchison D., Mauthe A., Yeadon N. (1996). "**QoS Support for Distributed Multimedia Applications**", Proceed Int. Conf. In Distributed Processing (ICDP '96), Dresden 1996.
- [98] Han C., Shin K. (1995) "**Scheduling MPEG_Compresed Video Streams with Firm Deadline Constrains**" ACM Multimedia '95, San Francisco, 1995, 411-422.
- [99] ISO/IEC JTC1/SC21, (1995). "**Information Technology – Quality of Service Framework – Final CD**", July 1995
- [100] Kalkbrenner G. et al. (1994). "**Quality of Service (QOS) in Distributed Hypermedia Systems**", Proc. 2 nd Int'l Workshop on Principles of Document Processing, 1994
- [101] Nahrstedt K., Smith J. (1995). "**The QoS Broker**", IEEE Multimedia, Spring 1995
- [102] Riley M., Richardson E. (1994). "**Minimizing the Effect of Cell**

Losses on MPEG Video", BRIS'94, Hamburg 1994, 491-494

- [103] Schulzrinne H., Casner S., Frederick R., Jacobson V. (1996). **"RTP: A Transport Protocol for Real-Time Application"**, (RFC 1889) January 1996
- [104] Talley L., Jeffay K. (1994). **"Two-Dimensional Scaling Techniques for Adaptive, Rate-Based Transmission Control of Live Audio and Video Streams"**, ACM Multimedia '94, S. Francisco, 1994
- [105] Vogel A., Kerhervé B., Bochmann G., Gecsei J. (1995). **"Distributed Multimedia and QoS: A Survey"**, IEEE Multimedia, Vol.2, Numb2, 1995

ANEXOS

MATRIZ DE CONSISTENCIA

CALIDAD DE SERVICIO UTILIZANDO EL PROTOCOLO RTP/RTCP				
PROBLEMA	OBJETIVO	HIPOTESIS	VARIABLES	MÉTODOS
GENERAL	GENERAL	GENERAL		GENERAL
El mayor problema de la utilización del protocolo UDP/IP como mecanismo de transporte para aplicaciones en tiempo real (real-time applications) es la falta de garantía de entrega de los paquetes a sus destinos, además que los paquetes dañados o retrasados con respecto al momento de su reproducción también son descartados.	Implementar mecanismos que resuscan la detoriación o pérdida de paquetes originados principalmente por los factores: atrazo, jitter, ancho de banda, y pérdida de paquetes.	Manteniendo las variables en valores debajo de los siguientes: $IPDT \leq \text{Valor A}$ $IPDV \leq \text{Valor B}$ $IPLR \leq \text{Valor C}$ Banda = Códec utilizado Se puede garantizar QoS del servicio en estudio. Valores A, B, C, etc a ser determinados.	INDEPENDIENTES $IPDT = \text{Atrazo}$ $IPDV = \text{Variación de atrazo}$ $IPLR = \text{Pérdida de paquetes}$ $BANDA = \text{ancho de banda necesario}$. DEPENDIENTES: $QoS = \text{Calidad de servicio}$ $QoS = f(IPDT, BANDA, IPDV, IPLR)$ $IPDT = \text{Atrazo}$ $IPDV = \text{Variación del atrazo}$ $IPLR = \text{Tasa de paquetes perdidos}$.	Revisión de la literatura sobre el tema de evaluación de calidad de voz y de otros medios a ser utilizados, visando una solución que permita utilizar de preferencia recursos disponibles en las instalaciones tipo (RTP/RTCP) y otros de costo accesible.
ESPECÍFICOS	ESPECÍFICOS	ESPECÍFICOS		ESPECÍFICOS
1. Implementar QoS para aplicaciones multimedia.	1. Implementar mecanismos para mantener una calidad de servicio para señales de voz.	1. Monitoreando y controlando los parámetros de atrazo, jitter, ancho de banda, y pérdida de paquetes		Se aplicaron los métodos de investigación: Descriptivo y Experimental.
2. En particular para las aplicaciones que utilizan voz y video.	2. Implementar mecanismos para mantener una calidad de servicio para señales de video.	2. Mantener los parámetros siguientes: atrazo, jitter, ancho de banda, y pérdida de paquetes, por debajo de los valores de: atrazo \leq que 150 ms. Jitter \leq que 50 ms. Pérdida de paquetes \leq a 1 % y ancho de banda definido por el CODEC utilizado		Particularmente los resultados fueron obtenidos vía simulación y pruebas en redes "reales" para validación.

Anexo 2

EJEMPLO BÁSICO DE IMPLEMENTACIÓN DE QoS

Muchos servicios tales como videoconferencia, *streaming*, juegos *on-line*, *voice over IP (VOIP)* y negocios *on-line* tienen rigidos requisitos de **QoS** de red. Desafortunadamente, la *Internet* actual no ofrece una **QoS** satisfactoria a sus usuarios finales. Permanecen algunas inestabilidades que amenazan a la calidad del servicio. Varias formas de implementación de mecanismos de **QoS**, unos simples y otros mas sofisticados han sido propuestos, entre ellos "*adaptive rates*", mecanismos de filas, **InterService**, **DiffService**, **RSVP**, **MPLS**, etc. EL ejemplo, aquí tratado y a seguir presentado, es parte de los mecanismos de filas implementados en los *routers* de las redes donde cada *router* debe implementar alguna disciplina de filas para establecer como cada paquete es bufferizado mientras espera por ser transmitido. Existen varias disciplinas que pueden ser usadas para controlar cuales paquetes serán transmitidos y cuales serán descartados. Estas disciplinas también afectan la latencia que tendrá un paquete, por la determinación de cuanto un paquete espera para ser transmitido. Las formas mas comunes son: *First In First Out (FIFO)*, *Priority Queuing (PQ)* y *weighted-Fair-Queuing (WFQ)*. La filosofía del mecanismo **FIFO**, es que el primero paquete que llega al *router* es el primer paquete a ser transmitido dado que la cantidad de espacio de *buffer* en cada *router* es finita, si un paquete llega y el *buffer* esta lleno, el *router* descarta el paquete.

Esto es hecho sin considerar a que flujo el paquete pertenece o cuan importante es el paquete. Este es el tipo usado en la *Internet* actual.

PQ es una simple variación del mecanismo básico, **FIFO**. La idea es marcar cada *packet* con una prioridad, la marca normalmente es colocada en el campo IP "*Type of service*" (**TOS**) del paquete. Los *routers* implementan varias filas **FIFO**, una por cada una de las clases prioritarias. Dentro de cada prioridad los paquetes son manipulados de la misma manera que en **FIFO**. Esta disciplina de filas permite cortar en la frente de la línea.

WFQ es parte del mecanismo **FQ** (*Fair Queueing*), cuya filosofía es mantener una fila separada por cada flujo pasando a través de ese *router*. El *router* atiende a esas filas de una manera *round robin*. **FQ** permite que un "peso" sea asignado a cada flujo (fila). Este "peso" efectivamente controla el porcentaje de ancho de banda del enlace que cada flujo conseguirá. Normalmente son usados los *bits* del campo **TOS** del cabecal **IP**, para identificar este "peso". Cuando este peso es implementado, se toma el **FQ** en **WFQ**.

Los graficos a, b, c, d, e, f muestran las diferencias de las 3 disciplinas.

GRÁFICO A
IPDT PARA LOS 3 MECANISMOS DE FILAS

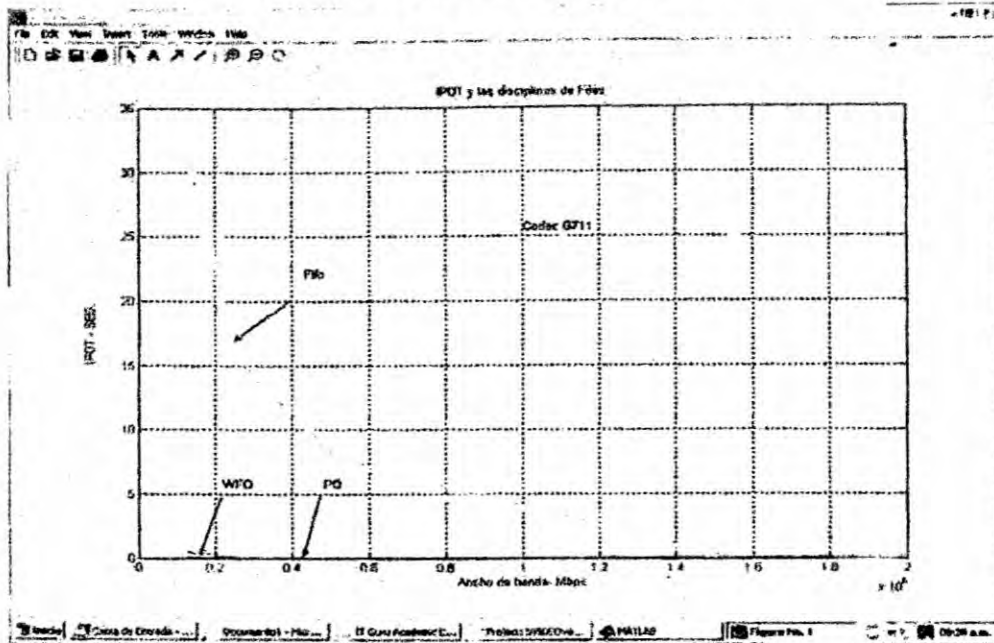


GRÁFICO B
DETALLE DEL IPDT PARA PQ Y WFO

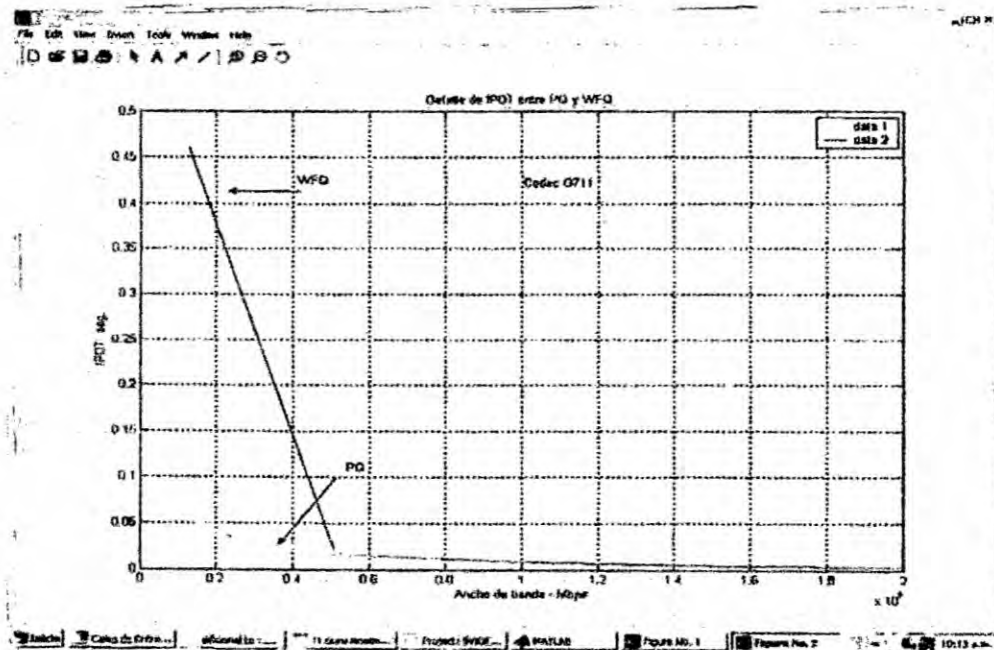


GRÁFICO C
IPDV PARA LOS 3 MECANISMOS

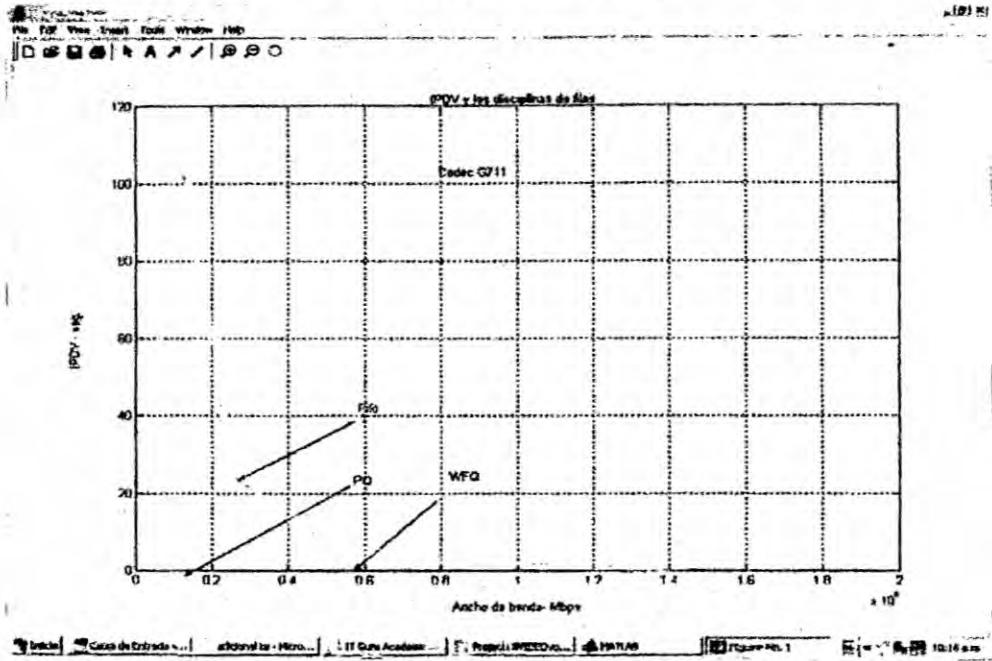


GRÁFICO D
DETALLE DE IPDV PARA PQ Y WFQ

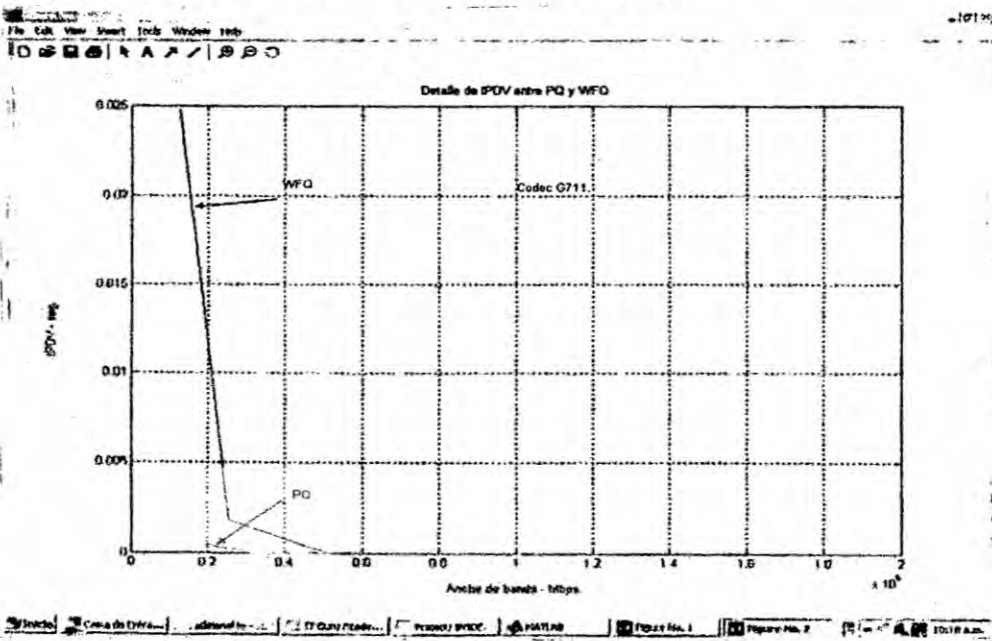


GRÁFICO E
IPLR Y LOS 3 MECANISMOS

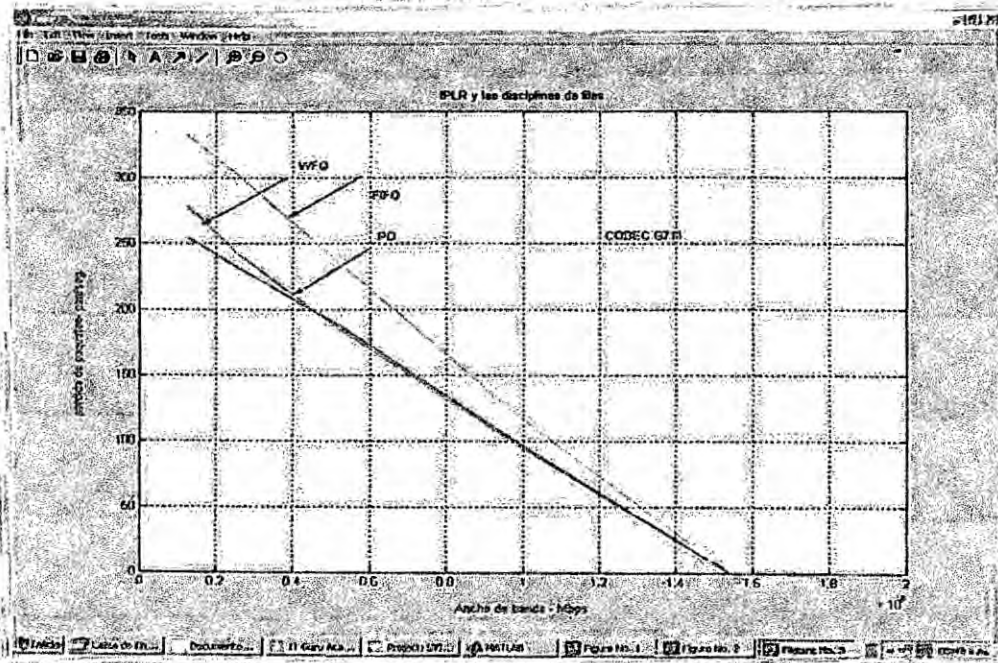
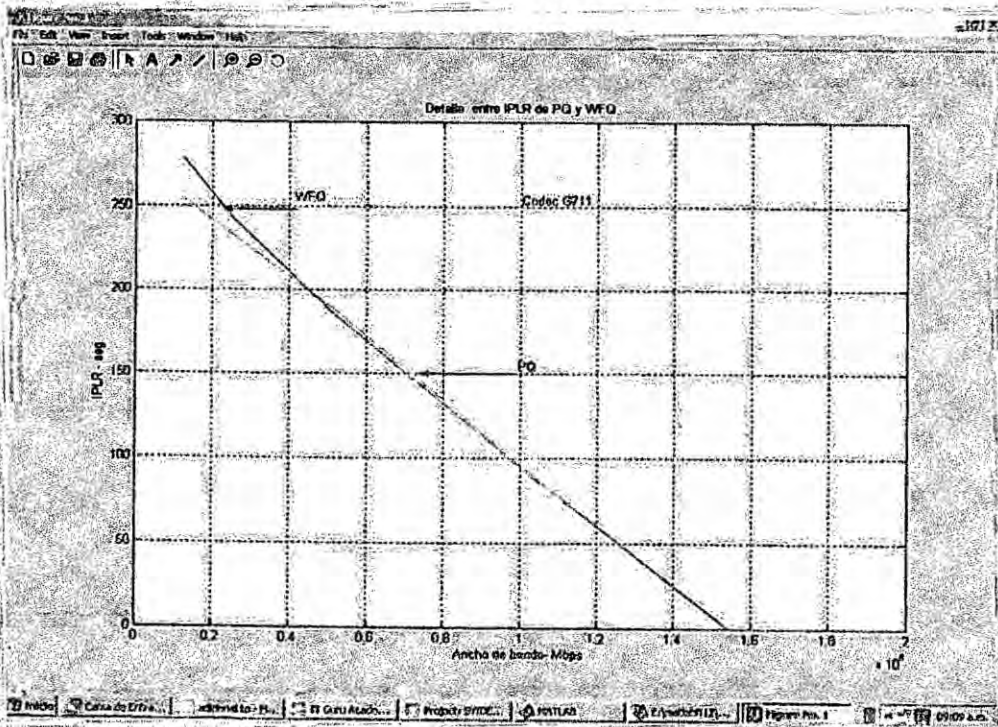


GRÁFICO F
DETALLE DO IPLR PARA PQ Y WFQ



Los gráficos anteriores, muestran el comportamiento de un flujo de voz G.711 cuando comparte un enlace de 2 **Mbps** con um flujo de video **UDP** y disciplina de filas **FIFO** (a, b), **PQ** (c, d) y **WFQ** (e, f).