



**Universidad del Azuay**

**Facultad de Administración**

**Escuela de Ingeniería de Sistemas**

*Análisis y evaluación para la selección de codecs de VoIP*

**Trabajo de graduación previo a la obtención del título de**

**Ingeniero de Sistemas**

**Autores: Paola Renata Montenegro Cantos**

**Verónica Carolina Mora Luna**

**Director: Ing. Pablo Esquivel**

**Cuenca, Ecuador**

**2007**

## Índice de Contenidos

Índice de Contenidos.....	ii
Resumen.....	iv
Abstract.....	v
Introducción.....	1
Capítulo 1. Marco Conceptual.....	2
Introducción.....	2
1.1. Codificación de voz.....	2
1.2. Concepto de codec.....	3
1.3. Historia de los codificadores de voz.....	3
1.4. Etapas para la conversión analógico - digital.....	4
Capítulo 2. Los Estándares de Compresión. Clasificación de los Codificadores de Voz.....	8
Introducción.....	8
2.1. Codificadores de la forma de onda.....	8
2.1.1. Codificadores en el dominio del tiempo.....	9
2.1.1.1. Modulación por codificación de impulsos (PCM).....	9
2.1.1.2. Modulación por codificación de impulsos diferencial (DPCM).....	9
2.1.1.3. Modulación por codificación de impulsos diferencial adaptativa (ADPCM).....	9
2.1.2. Codificación en el dominio de la frecuencia.....	10
2.1.2.1. Codificación en sub-bandas.....	10
2.1.2.2. Codificación por transformada.....	11
2.2. Vocoders.....	11
2.2.1. Vocoder por predicción lineal.....	11
2.2.2. LPC-10 (FS-1015).....	12
2.3. Codificadores híbridos.....	12
2.3.1. Codificación RPE-LTP (Regular Pulse Excitation Long Term Predictor) ...	12
2.3.2. Codificación CELP (FS-1016) (Code Excited Linear Prediction).....	13
2.4. Comparación de los algoritmos de compresión.....	14
Capítulo 3. Métodos de Análisis de la calidad de los codificadores.....	15
Introducción.....	15
3.1. Codecs Estándar.....	15
3.2. Mecanismos de reducción de ancho de banda.....	18
3.3. Calidad De Voz En Redes De Paquetes.....	19
3.4. Comparación codificadores de voz.....	26
Capítulo 4. Demostración Práctica.....	30
Introducción.....	30
4.1. Procedimiento a seguir para la evaluación de los codecs.....	30
4.1.1. Fórmulas.....	30
4.2. Aplicación.....	32
4.2.1. Aplicación de las fórmulas.....	32
4.2.2. Calculadoras de Ancho de Banda.....	41
Capítulo 5. Conclusiones y Recomendaciones.....	45
Conclusiones.....	45
Recomendaciones.....	45
Referencias.....	46
Glosario.....	46
Bibliografía.....	49
Anexos.....	51

Anexo 1: Openphone .....	51
Anexo 2: Ethereal.....	53

## **Resumen**

Este trabajo presenta un análisis de las características principales de los codificadores de voz estándar y varias formas de evaluar su eficiencia, según las diferentes situaciones en las que se les puede utilizar. A través del desarrollo del tema conseguiremos: conocer la historia y el funcionamiento de los codificadores de voz, a su vez elaboraremos una tabla que resuma las principales características de los codecs analizados y mediante métodos científicos, como fórmulas para calcular los factores que afectan a la calidad, herramientas adecuadas como analizadores de protocolos, calculadoras de ancho de banda y finalmente reglas y recomendaciones para seleccionar un codificador que obtenga resultados óptimos en la calidad según los recursos disponibles.

## **Abstract**

This paper presents an analysis of the main characteristics of standard voice encoders and the several ways to evaluate their efficiency according to the different situations where they can be used. Through the development of the topic, we got to know the history and the functioning of the voice encoders, and we made a chart that sums up the main characteristics of the codecs studied. For this research, we used several tools such as “sniffers”, bandwidth calculators, and formulas to calculate the factors that affect the quality in order to select the encoder that presents the best results regarding quality according to the available resources.



## **Introducción**

Las comunicaciones en Internet han variado enormemente en las últimas décadas. Hoy en día, el amplio despliegue de las redes de acceso de banda ancha, sumada a la optimización de los mecanismos de compresión y transmisión, abre la posibilidad de ofertar servicios de voz y video en tiempo real, a través de redes de conmutación de paquetes; redes que originalmente no estaban orientadas a dicho fin.

En este entorno se han desarrollado las soluciones de telefonía sobre IP. En el proceso de desarrollo de esta tecnología surgen nuevos horizontes a medida que los recursos de la red van incrementándose, lo cual conlleva a ofrecer algo más que un servicio básico de telefonía.

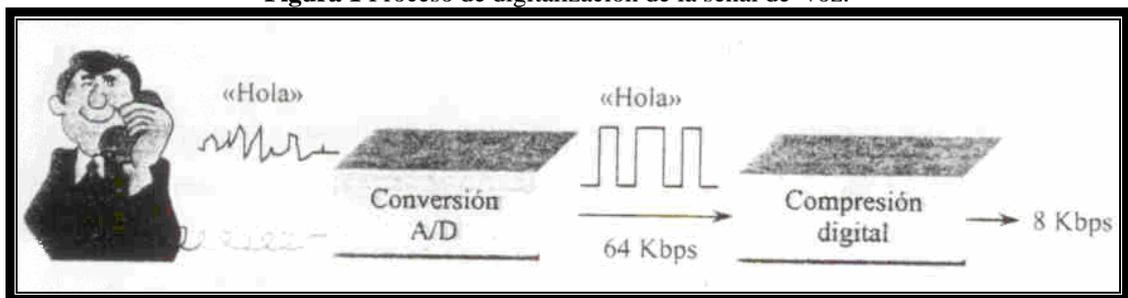
Para esto estudiaremos un componente muy importante en la comunicación que son los codecs de voz (abreviatura de CODificador/DECodificador), los cuales pueden ser hardware o software cuya funcionalidad es la de convertir la señal analógica en un conjunto de muestras digitales aptas para su transmisión por la red de paquetes. A través de la investigación a realizar se pretende obtener los mayores beneficios de la tecnología de Voz sobre IP; mediante una orientación adecuada para la elección del codificador de voz, según la situación de la empresa, en cuanto a recursos y expectativas de calidad que se requieran, sabiendo que se tiene que hacer un compromiso entre ellas.

## Capítulo 1. Marco Conceptual

### Introducción

El proceso de codificación de voz tiene mucha importancia para la comunicación, por tal razón hay que tener en cuenta la idea principal de este proceso donde: la señal de voz es transformada en una secuencia de bits por el codificador, esta es transmitida a través de un canal y finalmente convertida nuevamente en una señal audible por el decodificador para conseguir una representación digital (conversión analógico/digital - A/D-).

**Figura 1** Proceso de digitalización de la señal de voz.



### 1.1.Codificación de voz.

De modo genérico, la codificación consiste en realizar un conjunto de transformaciones a la señal que representa la información a transmitir con el fin de mejorar la eficiencia sin pérdida de calidad de la comunicación, compensando los efectos negativos de la presencia del canal (ruido, desvanecimientos, interferencias, etc.) .Para recuperar la información en el otro extremo será necesario un proceso inverso de decodificación (HUIDOBRO MOYA y ROLDAN MARTINEZ 2003 Pág.138...).

Desde el punto de vista de la transmisión de la señal de voz, la codificación de voz permite optimizar la utilización del canal de comunicación, transmitiendo el máximo de información. Desde el punto de vista de almacenar señal de voz en formato digital, la codificación de voz permite minimizar el número de bits necesarios para el

almacenamiento manteniendo un nivel de calidad adecuado. Desde el punto de vista de la comunicación, la señal de voz es transformada en una secuencia de bits por el codificador, es transmitida a través de un canal y finalmente convertida nuevamente en una señal audible por el decodificador. Como valor añadido al proceso, la codificación digital de voz permite incorporar algoritmos de cifrado para establecer comunicaciones privadas seguras o realizar grabaciones indescifrables para terceras personas.

### **1.2. Concepto de codec.**

Un codec es una abreviatura de COmpresor-DECompresor ó COdificador/DECodificador, el cual describe una especificación implementada en *software*, *hardware* o una combinación de ambos, capaz de transformar un archivo con un flujo de datos (*stream*) o una señal. Los codecs pueden codificar el flujo o la señal (a menudo para la transmisión o el almacenaje) y recuperarlo o descifrarlo del mismo modo para la reproducción o la manipulación en un formato más apropiado para estas operaciones. Los codecs son usados a menudo en videoconferencias y emisiones de medios de comunicación. Algunos codecs también apoyan la supresión del silencio, donde el silencio no se codifica ni se transmite. Para VoIP se los conoce también como *vocoders*. La voz ha de codificarse para poder ser transmitida por la red IP.

### **1.3. Historia de los codificadores de voz.**

Hace cincuenta años empezó la investigación en el campo de la codificación de la voz. El pionero fue Homer Dudley, que trabajaba en los laboratorios de *Bell Telephone*. La motivación a esta investigación surgió por la necesidad de transmitir voz por los cables de telegrafía de pequeño ancho de banda. La idea del *vocoder* (VOICE CODER, codificador de voz) era analizar la voz para extraer una serie de características y que el emisor enviase esas características, cuando éstas le llegasen al receptor reconstruiría la voz original. Este codificador recibió gran atención durante la Segunda Guerra Mundial, debido a su potencial en cuanto a eficiencia y posibilidad de encriptación se refiere.

Las primeras implementaciones del *vocoder* eran analógicas, sin embargo, con el nacimiento de los sistemas digitales y de las posibilidades que éstos ofrecen, pronto se pasó a las implementaciones digitales. Durante la década de los 40 hubo una gran actividad en la Codificación por Modulación de Impulsos (PCM). Este tipo de codificación no sigue la filosofía del *vocoder* de Dudley (y de los *vocoders* en general), sino que simplemente muestrea la voz. A partir del PCM se desarrollaron el DPCM y el ADPCM, que fueron propuestos como estándar por la CCITT (*International Consultative Committee for Telephone and Telegraph*).

Gracias a la flexibilidad de los sistemas digitales, se pudo experimentar con formas más sofisticadas de representación de la voz. Fant, a finales de los 50, trabajó en el modelo de producción de voz lineal. El surgimiento de la tecnología VLSI, tecnología de muy baja escala de integración, durante los 60 y 70 permitió nuevas soluciones al problema de la codificación de la voz. Así, por ejemplo, Flanagan y Golden propusieron una solución basada en la Transformada de Fourier.

Durante los 80 y 90, la investigación ha ido encaminada a conseguir codificadores que utilicen un ancho de banda cada vez menor mientras que la calidad de la voz sea cada vez mejor. Con esto se permite utilizar con más eficiencia y eficacia los canales de transmisión, se facilita la encriptación y se aprovechan mejor los sistemas de almacenamiento. Una de las principales aplicaciones de la codificación de voz es la telefonía móvil. En telefonía móvil, en Estados Unidos se utiliza un estándar de 8 Kbps (VSELP) y otro similar, a 6.7 Kbps, en Japón. En Europa, dentro del sistema GSM, se usa un codificador a 13 Kbps.

#### **1.4.Etapas para la conversión analógico - digital.**

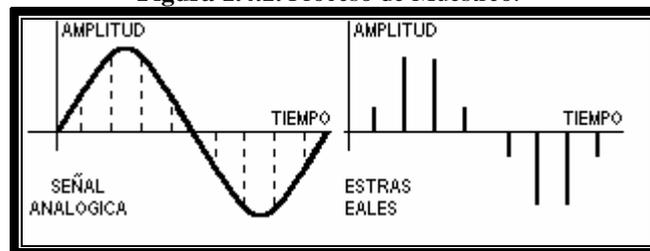
La transformación de la señal analógica a una señal digital se realiza mediante una conversión analógico-digital. Este proceso de conversión analógico digital o modulación por impulsos codificados (PCM) se realiza mediante tres pasos:

- Muestreo (*sampling*)
- Cuantificación (*quantization*)
- Codificación (*codification*)

## Muestreo

El proceso de muestreo consiste en tomar valores instantáneos de una señal analógica, a intervalos de tiempo iguales. A los valores instantáneos obtenidos se les llama muestras.

**Figura 1.4.1.** Proceso de Muestreo.



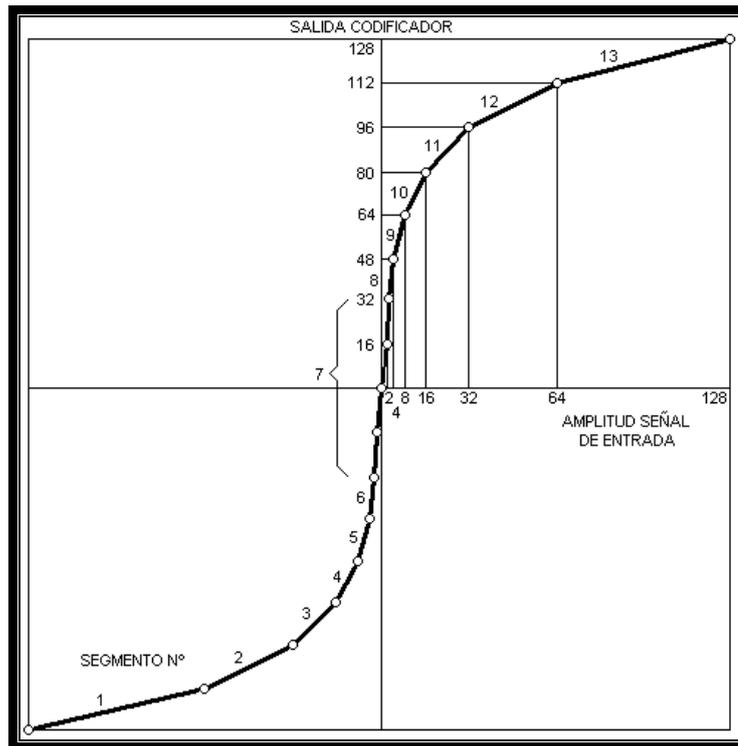
El muestreo se efectúa siempre a un ritmo uniforme, que viene dado por la frecuencia de muestreo  $f_m$  o *sampling rate*. La condición que debe cumplir  $f_m$  viene dada por el teorema del muestreo "Si una señal contiene únicamente frecuencias inferiores a  $f$ , queda completamente determinada por muestras tomadas a una velocidad igual o superior a  $2f$ ". De acuerdo con el teorema del muestreo, las señales telefónicas de frecuencia vocal (que ocupan la Banda de 300 a - 3.400 Hz), se han de muestrear a una frecuencia igual o superior a 6.800 Hz ( $2 \times 3.400$ ).

En la práctica, sin embargo, se suele tomar una frecuencia de muestreo o *sampling rate* de  $f_m = 8.000$  Hz. Es decir, se toman 8.000 muestras por segundo que corresponden a una separación entre muestras de:  $T=1/8000= 0,000125$  seg. = 125  $\mu$ s. Por lo tanto, dos muestras consecutivas de una misma señal están separadas 125  $\mu$ s que es el periodo de muestreo.

## Cuantificación

La cuantificación es el proceso mediante el cual se asignan valores discretos, a las amplitudes de las muestras obtenidas en el proceso de muestreo. En esta se logra: que cada amplitud instantánea (obtenida a partir del muestreo) sea representada por un byte de información.

Figura 1.4.2. Cuantificación.



## Codificación – Decodificación

La codificación es el proceso mediante el cual se representa una muestra cuantificada, mediante una sucesión de "1's" y "0's", es decir, mediante un número binario. Normalmente en telefonía se utilizan 256 intervalos de cuantificación para representar todas las posibles muestras (por ejemplo para G.711 tanto ley A como ley  $\mu$ ), por tanto se necesitarán números binarios de 8 bits para representar a todos los intervalos (pues  $2^8 = 256$ ). Otros codecs que usan ADPCM o cuantificación delta utilizan menos intervalos y por tanto menos bits. El dispositivo que realiza la cuantificación y la codificación se llama codificador.

La decodificación es el proceso mediante el cual se reconstruyen las muestras, a partir de la señal numérica procedente de línea. Este proceso se realiza en un dispositivo denominado decodificador. Al conjunto de un codificador y de un decodificador en un mismo equipo, se le llama codec.

## **Capítulo 2. Los Estándares de Compresión. Clasificación de los Codificadores de Voz**

### **Introducción**

En este capítulo se describen las técnicas de codificación más importantes usadas en transmisión de voz. En cada sección se explica cada una de las técnicas de codificación que son: forma de onda; los cuales son los más simples porque codifican la señal muestra por muestra, *vocoder*; los cuales incorporan el modelo de producción del habla en las técnicas de compresión y los híbridos que es una mezcla de los dos anteriores.

### **2.1. Codificadores de la forma de onda.**

Generalmente se diseñan para ser independientes a la señal, de tal forma que pueden ser usados para codificar una gran variedad de señales (no solamente voz, música, fax). Presentan una degradación aceptable en presencia de ruido y errores de transmisión. Sin embargo, para que sean efectivos, sólo se deben usar a *bit rates* medios de orden 32 kb/s. La codificación se puede llevar a cabo tanto en el dominio del tiempo como de la frecuencia.

Estos codecs se basan en almacenar información sobre la forma en el tiempo de la señal. Por lo general son los que más ancho de banda consumen, dado que no utilizan ninguna característica especial de la señal. Los codificadores de forma de onda dividen en dos grupos: Codificadores en el dominio del tiempo y Codificación en el dominio de la frecuencia.

### **2.1.1. Codificadores en el dominio del tiempo**

Dentro de este grupo tenemos los siguientes codificadores:

#### **2.1.1.1. Modulación por codificación de impulsos (PCM)**

La modulación por codificación de impulsos es la codificación de forma de onda más sencilla. Básicamente, consiste en el proceso de cuantificación. Cada muestra que entra al codificador se cuantifica en un determinado nivel de entre un conjunto finito de niveles de reconstrucción. Cada uno de estos niveles se hace corresponder con una secuencia de dígitos binarios, y esto es lo que se envía al receptor. Se pueden usar distintos criterios para llevar a cabo la cuantificación, siendo el más usado el de la cuantificación logarítmica. G.711/G.713 (abarcan la cuantificación y codificación).

#### **2.1.1.2. Modulación por codificación de impulsos diferencial (DPCM)**

Puesto que PCM no tiene en cuenta la forma de la onda de la señal a codificar, funciona muy bien con señales que no sean las de la voz, sin embargo, cuando se codifica voz hay una gran correlación entre las muestras adyacentes. Esta correlación puede aprovecharse para reducir el *bit-rate*. Una forma sencilla de hacerlo sería transmitir solamente las diferencias entre las muestras. Esta señal de diferencia tiene un rango dinámico mucho menor que el de la voz original, por lo que podrá ser cuantificada con un número menor de niveles de reconstrucción.

#### **2.1.1.3. Modulación por codificación de impulsos diferencial adaptativa (ADPCM)**

Implementa la misma técnica que DPCM pero utilizando un algoritmo de predicción más eficiente. La predicción se realiza en base a un algoritmo autoadaptativo dependiente de la actividad de la señal vocal. ADPCM (G.726/G.727) transmite una muestra (PCM) y sus diferencias (posteriormente diversas modificaciones del estándar permitían transmitir alguna muestra patrón periódicamente, para solucionar problemas de pérdida de datos).

Por lo tanto su algoritmo es muy sencillo, su retardo muy bajo ( $< 1$  ms) y su calidad muy buena (a 32 kbps, para la que fue diseñada inicialmente ADPCM, G.721), basada en el hecho, de que normalmente cuando una persona habla mantiene más o menos un nivel de voz. Cambios lentos en el nivel son recogidas rápida y perfectamente por el algoritmo. Con un cambio muy brusco se hace un cambio de escala (equivalente a transmitir una muestra completa) y posteriormente se vuelven a transmitir las diferencias. El intervalo de adaptación es tan pequeño que es imperceptible para un oído humano convencional.

La versión actual del estándar (G.727 E-ADPCM, *Enhanced ADPCM*, ADPCM mejorado) permite utilizar 2 bits (16 kbps), 3 bits (24kbps), 4bits (32 kbps) ó 5 bits (40 kbps) para cuantificar y codificar la diferencia entre la muestra anterior y la actual. Hoy día está en desuso pues hay algoritmos más complejos que requieren mucho menor ancho de banda y que tienen igualmente una excelente calidad. Sin embargo, sigue siendo una referencia por su sencillez, efectividad y porque no sólo se utiliza con la voz, sino que se adaptó a la música estéreo.

### **2.1.2. Codificación en el dominio de la frecuencia**

Este tipo de codificadores dividen la señal en distintas componentes en frecuencia y codifican cada una de éstas de forma independiente. El número de bits usados para codificar cada componente en frecuencia puede variar dinámicamente. Algunos codificadores son: codificación en sub-bandas, codificación por transformada.

#### **2.1.2.1. Codificación en sub-bandas**

Es el más sencillo de los métodos en el dominio de la frecuencia. La principal ventaja de la codificación en subbandas es que el ruido de cuantificación que se produce en cada banda queda confinado a la misma. La codificación en subbandas se usa mucho en señales de un gran ancho de banda, como puede ser en tele conferencia (estándar G.722 de la CCITT).

### **2.1.2.2. Codificación por transformada**

Consiste en una codificación por bloques. La señal de entrada se transforma en un dominio diferente y se codifican los coeficientes de la transformación. En el receptor, el decodificador calcula la transformada inversa para obtener la señal original reconstruida. La codificación por transformada se utiliza en la codificación de señales de banda ancha de imagen y sonido. Sin embargo, no se usa mucho en codificación de voz debido a su complejidad.

## **2.2. Vocoders**

En el grupo de *vocoders* están aquellos codificadores que sí tienen en cuenta la naturaleza de la señal a codificar, en este caso la voz y aprovechan las características de la misma para ganar en eficiencia. Permiten trabajar con muy bajos *bit rates*, pero la señal de voz que producen suena demasiado sintética. Los *vocoders* intentan producir una señal que suene como la voz original, independientemente de si la forma de onda se parece o no. Son las que menor ancho de banda utilizan (por debajo de 2.4 kbps, incluso se pretende llegar hasta 1 kbps), sin embargo la calidad de voz es baja, su retardo alto, igual que su carga de proceso. Utilizan complejos modelos matemáticos de la voz humana, a través de los cuales obtienen ciertos parámetros particulares de la conversación en curso y son transmitidos.

El resultado es, que se produce voz inteligible a muy bajo *bit rate*, consiguiéndose entender la conversación, pero no se reconoce a la persona con la que se habla (la sensación es la del habla de un robot). Generalmente no se usa comercialmente, y sí militarmente, donde un bajo régimen binario facilita una compleja encriptación (por motivos de seguridad) a mucho menor costo.

### **2.2.1. Vocoder por predicción lineal**

Es el tipo de *vocoder* más utilizado, este utiliza el mismo modelo de producción que otros *vocoders* pero difiere en la determinación del modelo del tracto vocal. Supone

que el tracto vocal funciona muy bien y es posible alcanzar una señal de voz sintetizada muy inteligible a *bit rates* del orden de 2.4 Kbps.

### **2.2.2. LPC-10 (FS-1015)**

Este algoritmo de codificación de voz fue desarrollado por el DOD (*Department Of Defense*) y permite la codificación de la señal de la voz a una velocidad de 2400 bps.

## **2.3. Codificadores híbridos**

En el tercer grupo, los codificadores híbridos, encontramos aquellos que combinando técnicas de los *vocoders* y de los codificadores de forma de onda unifican las ventajas de ambos, permitiendo una alta calidad de voz a bajos *bit rates* (inferiores a 8 Kb/s). Resultando en tasas de transmisión más altas que los vocales pero menor que los de forma de onda, la complejidad es menor que los vocales y mayor que los de forma de onda. En términos de calidad se aproximan a los codificadores de forma de onda.

También se les conoce como codificadores de análisis-por-síntesis. En el emisor se lleva a cabo un análisis que obtiene los parámetros de la señal para luego sintetizarla y conseguir el mayor parecido a la original. Las tres más utilizadas son las CELP (y sus variantes), y las VSELP (teléfonos móviles americanos).

### **2.3.1. Codificación RPE-LTP (Regular Pulse Excitation Long Term Predictor)**

Es el estándar europeo más utilizado en la telefonía celular. Una de sus ventajas sobre los demás codecs de esta categoría es su simplicidad (poca potencia necesaria, basta con 4 MIPS) y bastante popular. Se utiliza en Internet para audio-conferencias (*chats de voz*) y telefonía y en los teléfonos móviles GSM (es el modo FR – Full

Rate) a 13 kbps y su calidad es buena, similar a la línea analógica tradicional, para ese régimen binario. Su retardo de generación de trama habitual es de 20 ms.

Una posterior incorporación al estándar es G.723.1 que únicamente utiliza 5,33 kbps ó 6,40 kbps (técnicamente se convierte en Multi-Pulse ACELP, MP-ACELP). Se mantiene prácticamente la calidad en el caso de 5,33 kbps e incluso con 6,4 kbps aumenta un poquito respecto a la especificación inicial, a costa de aumentar por 4 la potencia de proceso necesaria y de triplicar su retardo. Por sus características, calidad y reducido ancho de banda requerido, esta G.723.1, en sus 2 variantes de 5,33 y 6,40kbps, es uno de los *codecs* más usados para VoIP.

### **2.3.2. Codificación CELP (FS-1016) (Code Excited Linear Prediction)**

Los codecs CELP (G.723) trabajan dividiendo la señal en segmentos del habla, denominados tramas. Usan un modelo del sistema vocal para remover la redundancia de la señal, permitiendo la transmisión a una tasa de datos más baja (típicamente entre 4 y 16 kbps para aplicaciones telefónicas). Los codecs CELP generalmente crean un retardo mayor que los codecs de forma de onda. Una trama de habla CELP no puede codificarse hasta que el codificador obtenga toda la información correspondiente a esa trama. Esto significa que hay un retardo de una trama entera antes de que el codec pueda empezar a procesar la señal.

Variantes:

- LD-CELP (*Low Delay CELP*) (G.728): Por su bajo retardo (5 ms) y relativamente alto régimen binario (16 kbps), precisa de unos 40 MIPS en el DSP, potencia muy elevada comparada con el resto de técnicas y aunque utilice un libro de códigos menor, las operaciones de análisis de la voz con las restricciones de bajo retardo/tiempo real características de LD-CELP, suponen una carga de proceso importante. Como ventaja se puede decir que con LD-CELP aún se pueden transmitir datos a 2.4 kbps.

- ACELP (Algebraic CELP)
- CS-ACELP (Conjugated Structure ACELP) (G.729, G.729a): Usado hoy en GSM a 12,2 kbps, aunque el estándar ITU-T G.729 define como régimen binario únicamente 8 kbps. 25 ms de generación de trama. Con la revisión del estándar de la ITU-T se mejoró ligeramente la calidad del sonido y se redujo a la mitad la potencia necesaria para su implementación (de 20 a 10 MIPS para 8 kbps). El retardo de generación de trama típico se sitúa en unos aceptables 25 ms. Bajo la capacidad de un teléfono móvil GSM de hoy día de utilizar EFR (*Enhanced Full Rate*) está usándose por debajo una codificación CS-ACELP a 12.2 kbps. Un aspecto muy importante de CS-ACELP es que sigue funcionando correctamente cuando se pierden paquetes debido a que utiliza interpolación para dichos paquetes.

#### 2.4. Comparación de los algoritmos de compresión

En las siguientes figuras se muestra las velocidades y calidad de cada uno de los métodos de codificación:

**Figura 2.4.1** Comparación de los algoritmos de codificación.

Codificador	Velocidad requerida	
	Kbps	MOS
PCM (G.711)	64	4.4
ADPCM (G.726)	32	4.2
LD-CELP (G.728)	16	4.2
CS-ACELP (G.729)	8	4.2
ACELP (G.723.1)	5.3	3.5
RPE-LTP	13	

## Capítulo 3. Métodos de Análisis de la calidad de los codificadores

### Introducción

Este capítulo está orientado a presentar las principales características de los codificadores estándar para voz sobre IP, recomendados por la ITU-T. También se mostrarán las diferentes formas de medir la calidad de voz percibida por el usuario final y todos los factores que se deben tomar en cuenta al momento de escoger el codificador de voz; el cual ayudará a realizar una implementación adecuada, según la necesidad y los recursos disponibles.

### 3.1. Codecs Estándar

#### G.711

Principal codec de la PSTN estandarizado por la ITU (*International Telecommunication Union*) en 1972, es utilizado para representar señales de audio con frecuencias de la voz humana, mediante muestras comprimidas de una señal de audio digital con una frecuencia de muestreo (*sampling rate*) de 8 kHz. Utiliza PCM (*Pulse Code Modulation*) para comprimir, descomprimir, codificar y decodificar y un *bit rate* de 64 kbps. Maneja un índice del muestreo de 8.000 muestras por segundo. La tolerancia en esa tarifa es  $\pm 50$  porciones por millón (PPM). Ocho dígitos binarios por muestra se utilizan.

Se utilizan dos leyes de codificación o algoritmos principales y éstos se refieren comúnmente como la Uno-ley (*A-law* usado en Europa y el resto del mundo) la cual codifica cada 14 muestras en palabras de 8 bits y la mu-ley ( *$\mu$ -law* usado en Norte América y Japón) que codifica cada 13 muestras en palabras de 8 bits. Ambos algoritmos son logarítmicos, pero el *A-law* fue específicamente diseñado para ser implementado en una computadora. G.711 es un algoritmo simple y de menor carga computacional y es la base del resto de estándares.

### **G.722**

Describe las características de un sistema de codificación de audio (50 a 7000 Hz) a velocidad binaria de 64 kbps (*bit rate*), el cual puede ser utilizado en una gran variedad de aplicaciones de voz de una mayor calidad. El sistema de codificación utiliza la modulación adaptativa diferencial de la subbanda para pulsos codificados (SB-ADPCM). En la técnica SB-ADPCM utilizada, la banda de frecuencia es dividida dentro de dos subbandas (mayor y menor) y las señales en cada subbanda son codificadas utilizando ADPCM.

El sistema tiene tres modos básicos de operación correspondientes a las velocidades de transmisión utilizadas para la codificación de audio de 7 khz: 64, 56 y 48 Kbps. Su velocidad de muestreo se dobla, a 16 Khz. Al aumentar el ancho de banda (50 - 7.000 Hz) se consigue naturalidad (bajas frecuencias) e inteligibilidad (altas frecuencias) tiene un retardo de 3 ms y una complejidad baja, se aplica en canales de 64 kbits/s de ISDN, para difusión de radio.

#### **G.722.1**

Describe un algoritmo *wideband digital* del codificador que proporciona un ancho de banda de 50 Hz a 7 kilociclos, funcionando en un índice binario de 24 kbit/s o de 32 kbit/s. El algoritmo se basa en la tecnología MLT. Tiene una duración de trama de 20 ms.

#### **G.722.2**

Codificación de la voz a 16 kbps utilizando la técnica *adaptive multi-rate wideband* (AMR-WB), algoritmo de *vocoder*. Utiliza la banda de audio (50 - 7000 Hz). Al tener un mayor ancho de banda se mejora la inteligibilidad y el *naturalness* del discurso perceptiblemente. La tarifa del muestreo es 16000 muestras por segundo. El esquema de codificación es ACELP.

### **G.723.1**

G.723 fue desarrollado en 1995 por la ITU para el uso en plataformas multimedia, tiene un ancho de banda de 5.3kbps y 6.3kbps. El codificador procesa señales con un tamaño de trama de 30 ms.

### **G.726**

Es un estándar de la ITU, también conocido como ADPCM (*Adaptive Differential Pulse Code Modulation*), sustituyó al obsoleto estándar G.721 en 1990. Permite trabajar a velocidades de 16 kbps, 24 kbps y 32 kbps. La gran ventaja de este codec es la disminución de ancho de banda requerido sin aumentar en gran medida la carga computacional.

### **G.728**

Codificación de la voz de buena calidad 16 kbps utilizando la técnica de bajo retardo CELP. Tiene una duración de la trama de 0,625 ms. Una característica de los algoritmos de CELP es que tienden para realizar más pobre que ADPCM en la presencia del ruido de fondo.

### **G.729**

El codificador G.729 es una de las tecnologías de codificación que más ha revolucionado los sistemas de comunicaciones. En redes de computadores, esta técnica ofrece una excelente solución al brindar un codificador de bajo retardo y de alta calidad. Este sistema, que está basado en los codificadores CELP, incorpora técnicas optimizadas que permiten disminuir la carga computacional requerida. Tiene un tiempo de duración de trama de 10 ms.

La música o los tonos tales como los tonos de DTMF o de fax no pueden ser transportados confiablemente con este codec, para estas se debe utilizar G.711 o métodos de señalización fuera de banda, para transportar esas señales. G.729 se usa mayoritariamente en aplicaciones de Voz sobre IP, por sus bajos requerimientos en ancho de banda. El estándar G.729 opera a una tasa de bits de 8 kbit/s, pero existen extensiones, las cuales suministran también tasas de 6.4 kbit/s y de 11.8 kbit/s para mejor o peor calidad en la conversación respectivamente.

## **G.729A**

Codec desarrollado por *France Telecom, Mitsubishi Electric Corporation, Nippon Telegraph and Telephone Corporation* (NTT) y la Universidad de *Sherbrooke*. Actualmente la propiedad intelectual es de la empresa SIPRO. Requiere 8 kbps de ancho de banda. La carga computacional de este algoritmo es elevada y también es necesaria una licencia para su uso comercial. G.729 A es una versión reducida de la complejidad del codificador G.729.

### **3.2. Mecanismos de reducción de ancho de banda**

#### **Supresión de Silencios y VAD (*Voice Activity Detection*)**

Es un mecanismo complementario al empleo de codecs compresores, para reducir el ancho da banda. Se pretende detectar periodos de silencio durante la conversación (mecanismos VAD, *Voice Activity Detection*) suprimiendo el envío de paquetes de voz mientras dure la situación. La señal de silencio igualmente se codifica, pero el software de supresión evita que se envíen dichos bloques de datos. Para evitar que el interlocutor piense que se ha cortado la comunicación durante los intervalos de silencio la ITU-T especifica dos posibles soluciones: enviar periódicamente paquetes de silencio (SID, *Silence Insertion Description*) durante la pausa. Estos paquetes proporcionan una indicación del nivel de ruido que existe en el origen para que el receptor lo simule en el terminal remoto mediante un algoritmo de CNG (*Comfort Noise Generation*) o generador de ruido, para evitar el envío de paquetes SID es posible marcar el bloque generado como NOTX (*No Transmission*). En el receptor se genera ruido ambiente a partir de una señal de ruido blanco o del muestreo del auricular.

La segunda solución es la elección del tamaño de los paquetes de voz, dado que las cabeceras que se añaden a los paquetes aumentan la tasa normal del codec. Si se desea minimizar el impacto de las cabeceras en el tráfico, es preciso enviarlas el menor número de veces posible y para ello se envían paquetes de gran tamaño con varios bloques de datos en cada paquete. Esta solución presenta un gran

inconveniente porque cuanto mayor es la carga útil menor es la tasa de envío (reduce la sobrecarga por cabeceras) pero mayor es el retardo (aumenta el tiempo de construcción del paquete). Como la compresión de voz aumenta el tiempo de empaquetado, cuanto más complejo sea el algoritmo de compresión, menor deberá ser el tamaño de los paquetes para que el retardo no sea excesivo.

### **3.3. Calidad De Voz En Redes De Paquetes**

En una red de paquetes (en donde se transmite voz y datos) se busca una elevada calidad de voz de extremo a extremo, ya que esta no está garantizada debido a varios factores que la degradan. Por ello la medida de la calidad de la voz se ha convertido en un aspecto fundamental dentro del entorno de la convergencia de redes. Entre los factores que influyen en la calidad de la voz, se encuentran: el retardo, el *jitter*, las pérdidas de paquetes y la claridad de la voz.

#### **Retardo:**

“Una de las características más importantes de la voz es su temporalidad. Desde este punto de vista, dos sílabas pertenecen a una misma palabra si se pronuncian dentro de un cierto intervalo de tiempo, por lo que dicho intervalo llega a ser tan importante como las propias sílabas. Por ello, si se introdujera, un retardo adicional entre dos sílabas de una palabra la melodía de la voz se perdería, afectando esto a la inteligibilidad del mensaje transmitido. El retardo causa dos problemas:

- **Eco:** Es consecuencia de las reflexiones que sufre la señal en el otro extremo. Cuando el retardo del eco supera un cierto umbral (establecido por la ITU en 5 ms), el hablante comienza a escuchar una versión retardada de sus propias palabras. Si dicho retardo tuviera niveles muy elevados, mantener una conversación podría llegar a ser imposible.
- **Solapamiento de la voz de los interlocutores:** En el transcurso de una conversación se producen pausas que invitan al otro interlocutor a contestar. Si la respuesta no llega en un intervalo de tiempo razonable, el hablante original podría continuar hablando de tal suerte que cuando llegara la respuesta esperada la voz de ambos se solaparía haciendo imposible la

comunicación. El umbral del retardo a partir del cual este fenómeno empieza a aparecer se encuentra en torno a los 150ms.” (HUIDOBRO MOYA y ROLDAN MARTINEZ 2003 Pág.154).

Algunas de las fuentes de retardo en una sola vía para una llamada hecha con paquetes de voz se describen a continuación:

- Retardo acumulado o algorítmico: Es causado por la necesidad de recolectar un rango de muestras de voz para que sean procesados por el codificador. Esto está relacionado con el tipo de codificador usado.
- Retardo de procesamiento: Es causado por el proceso de codificación y recolección de las muestras codificadas en paquetes para la transmisión sobre una red de paquetes. El retardo de codificación es una función del tiempo de ejecución del procesador y el tipo de algoritmo usado. A menudo se recolectan múltiples muestras.
- Retardo de red: Es causado por el medio físico y los protocolos usados para transmitir los datos de voz y por los *buffers* usados para remover el *jitter* en el lado receptor. El retardo de red es una función de la capacidad de los enlaces en la red y del procesamiento que ocurre a medida que los paquetes transitan por esta. Los *buffers* para *jitter* agregan retardo, que es utilizado para remover la variación de retardo a la que están sujetos los paquetes a medida que transitan en una red de paquetes.

**Jitter:**

El *jitter* se define técnicamente como la variación en el tiempo en la llegada de los paquetes, causada por congestión de red, pérdida de sincronización o por las diferentes rutas seguidas por los paquetes para llegar al destino. El *jitter* entre el punto inicial y final de la comunicación debiera ser inferior a 100 ms. Si el valor es menor a 100 ms el *jitter* puede ser compensado de manera apropiada, en caso contrario debiera ser minimizado. La solución más ampliamente adoptada es la utilización del *jitter buffer* el cuál consiste básicamente en asignar una pequeña cola o almacén para ir recibiendo los paquetes y sirviéndolos con un pequeño retraso. Si

alguno paquete no está en el *buffer* (se perdió o no ha llegado todavía) cuando sea necesario se descarta. Un aumento del buffer implica menos pérdida de paquetes pero más retraso. Una disminución implica menos retardo pero más pérdida de paquetes.

### **Pérdida de Paquete:**

La pérdida de paquetes es uno de los parámetros de red que más distorsión introduce en la calidad perceptiva de la voz. Adicionalmente, desde el punto de vista del codificador un paquete retardado lo suficientemente como para exceder el tamaño de *jitter* debe ser tratado como un paquete perdido. En todo caso el codificador deberá recuperar mediante algún método. Esto puede incluir simplemente no mostrar un paquete, repetir el último paquete o reemplazar el paquete perdido mediante alguna técnica de implementación específica. Los paquetes se pueden perder también debido a la congestión y a las técnicas de manejo de congestión, tradicionalmente el tráfico UDP no tiene un mecanismo de manejo de congestión como el TCP *slow start*. Debido a esto, el alto volumen del tráfico UDP tiende a desplazar el tráfico TCP el cual será perdido al momento de la congestión.

**Figura 3.3.1.** Rangos aceptables de los factores, para mantener la calidad.

Aplicación	Simetría	Retardo pta a pta	Jitter (ms)	Pérdida de paquetes
Conversación	Ida y vuelta	< 150 ms (Preferido) < 400 ms (Máximo)	< 1	< 3 %
Streaming de audio	Solo ida	< 10 s	< 1	< 1%
Mensajes de Voz	Solo ida	< 1 s (reproducir) < 2 s (grabar)	< 1	< 3%

### **Claridad de la voz:**

“La claridad de la voz es un parámetro subjetivo que puede definirse como la fidelidad con que la voz es percibida por el extremo remoto y depende de la distorsión introducida por los componentes de la red. Sin embargo, es independiente del retardo (aunque el *jitter* si ejerce gran influencia) y del eco (puesto que este es escuchado por el emisor y la claridad se avalúa en el receptor).” (HUIDOBRO MOYA y ROLDAN MARTINEZ 2003 Pág.133).

En las redes de conmutación de circuitos tradicionales (PSTN) se han venido empleando las siguientes técnicas de medida:

**Relación Señal / Ruido (SNR, *Signal to Noise Ratio*):**

“Mide los niveles de ruido relativos en las señales analógicas y la distorsión introducida durante el proceso de cuantificación de un codificador digital. Las pausas que se producen al hablar parecen deteriorar la SNR (debido a la ausencia de señal) pero, esto no es totalmente cierto porque para estos casos, se emplea la SNR segmental que se obtiene calculando la SNR en los intervalos hablados y no en los silenciosos. La SNR es muy útil cuando el proceso de codificación conserva la forma de onda de la señal de entrada. No obstante, cuando se emplean algoritmos de comprensión, la SNR la SNR segmental no guardan ninguna relación útil con la claridad de la voz.” (HUIDOBRO MOYA y ROLDAN MARTINEZ 2003 Pág.133...).

**Distorsión:**

Las Medidas de distorsión evalúan la distorsión no lineal introducida por equipos de procesamiento de señal (por ejemplo amplificadores).

**Tasa de error de bit:**

Es una medida de la calidad física de la transmisión sobre una red determinada. Sin embargo, puede ocurrir que un patrón de bits con una tasa de error muy alta tenga un impacto mínimo en la claridad de la voz percibida por un oyente humano.

Todas estas medidas son adecuadas cuando se conserva la forma de onda la señal de entrada, por esta razón en las redes integradas son necesarios otros tipos de medidas basados en la calidad de percepción, como son las medidas subjetivas y objetivas.

**Medidas Subjetivas**

Las medidas subjetivas son las más intuitivas y consisten en realizar una llamada telefónica, descolgar el receptor y escuchar qué tal se oye la conversación. La medida más empleada es la escala MOS (*Mean Opinión Store*); la cual se basa en que un grupo de personas evalúan la calidad de la voz percibida y mediante el cálculo de la media de las valoraciones dadas por los oyentes, se obtiene los valores de la escala,

sin embargo, resulta un método difícil de implementar debido a su alto costo por la gran cantidad de personas que se necesitan para su ejecución.

La escala MOS está recogida en la recomendación P.800 de la ITU. Existen varios tipos de escalas MOS, en función de la prueba que se lleve a cabo, estas son: la utilizada para medir la calidad de la voz percibida Figura 3.2.2. y la que evalúa el esfuerzo requerido para entender el significado del mensaje. Figura 3.2.3.

**Figura 3.3.2.** Escala MOS utilizada para medir la calidad de voz

<b>Puntuación</b>	<b>Calidad</b>
5	Excelente
4	Buena
3	Aceptable
2	Pobre
1	Mala

**Figura 3.3.3.** Escala MOS para la medida del esfuerzo de interpretación del mensaje

<b>Puntuación</b>	<b>Esfuerzo</b>
5	Relajación completa no es necesario ningún esfuerzo.
4	Necesario prestar atención no se requiere esfuerzo apreciable.
3	Esfuerzo moderado.
2	Esfuerzo considerable.
1	Imposible de entender.

### **Medidas Objetivas**

Las medidas objetivas se obtienen mediante los algoritmos PAMPS, PSQM y PESQ, este tipo de medidas resultan más baratas, fiables, eficientes y rápidas en comparación con las evaluaciones subjetivas, pero la correlación con esos resultados es baja cuando se consideran parámetros de red. Además, la gran mayoría de estos

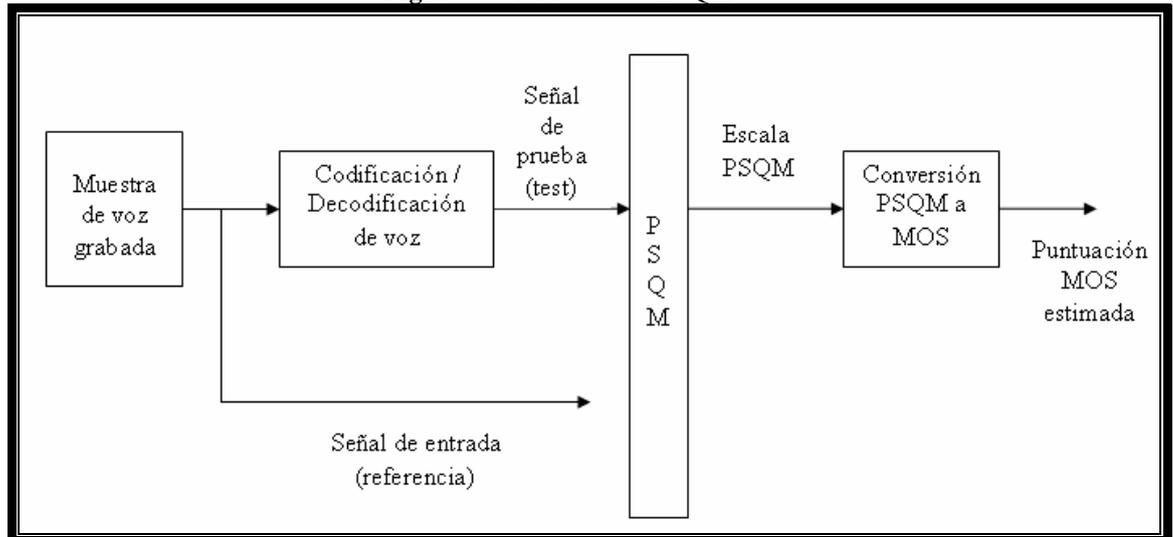
métodos no pueden utilizarse en tiempo real. Dentro de las medidas objetivas se distinguen dos grupos de medidas que se diferencian en su modo de interactuar con la red estas son: las medidas intrusivas que generalmente, consisten en la realización de una llamada de prueba, la transmisión de ficheros de voz pre grabados y la comparación entre la señal recibida y la transmitida, estos métodos suelen tener un coste de procesamiento relativamente alta y son ideales para la medida de las prestaciones de un codec en el laboratorio.

El otro gran grupo de medidas objetivas no intrusivas o pasivas, son más complejas que las intrusivas y son menos exactas; estas se efectúan mientras el sistema está en explotación, lo que supone el examen de las prestaciones de los dispositivos de la red. Se dispone de los siguientes estándares:

**PSQM** (*Perceptual Speech Quality Measurement*):

“Es un proceso matemático que proporciona una medida de la calidad de voz. El objetivo de PSQM es obtener una serie de puntuaciones que predigan los resultados de las pruebas subjetivas de manera fiable, especialmente de los métodos descritos en P.830. Sin embargo las escalas PSQM están organizadas según una distancia perceptual, es decir, que reflejan la diferencia entre una señal limpia y la señal de salida de la red telefónica. PSQM ha sido diseñada especialmente para anchos de banda telefónicos (300- 3400 Hz) y para codecs de voz. PSQM se aprobó como la recomendación P.861 de la ITU-T.”(HUIDOBRO MOYA y ROLDAN MARTINEZ 2003 Pág.154).

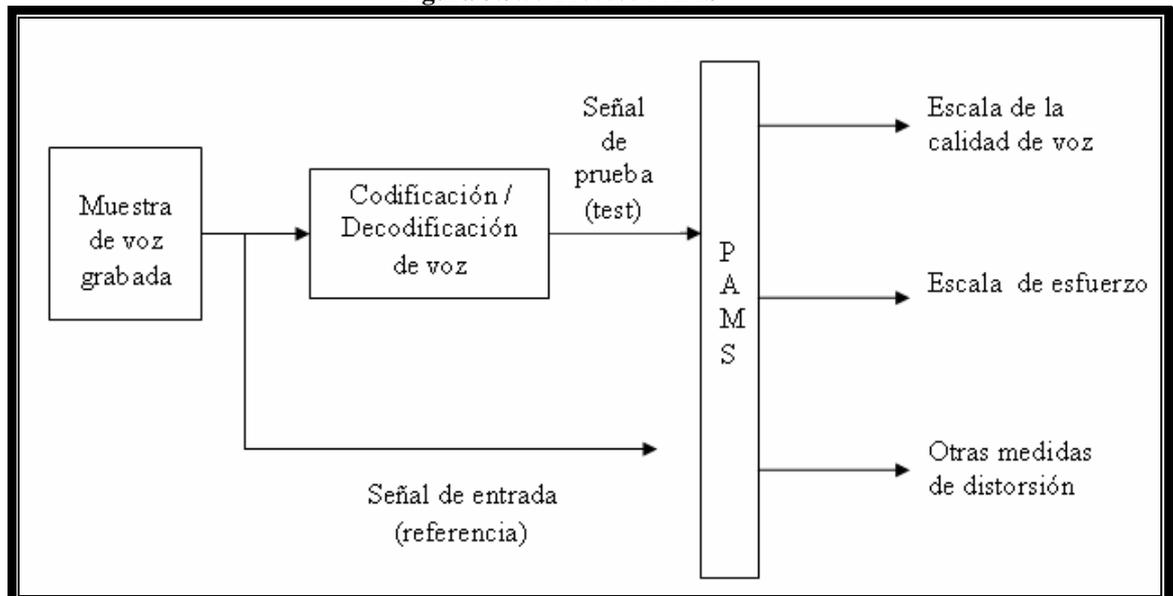
**Figura 3.3.4.** Proceso de PSQM



**PAMS:** (*Perceptual Analysis Measurement System*)

Emplea un modelo basado en los factores que influyen en la percepción humana para medir la claridad de la voz de la señal de salida percibida, en comparación con la de la señal de entrada. Aunque es similar a PSQM en muchos aspectos, utiliza técnicas de procesamiento diferentes y un modelo perceptual distinto.”(HUIDOBRO MOYA y ROLDAN MARTINEZ 2003 Pág.154).

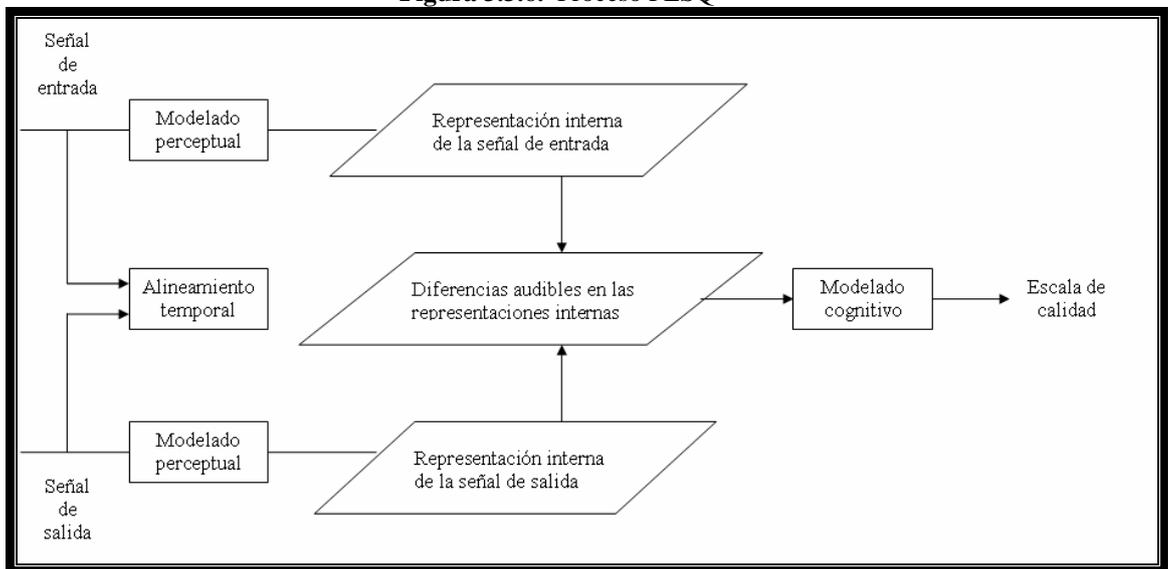
**Figura 3.3.5.** Proceso PAMS



### **PESQ** (*Perceptual Evaluation of Speech Quality*)

El algoritmo PESQ, desarrollado por la ITU en el año 2001, establece un método automático para la evaluación de la calidad perceptual. Este algoritmo presenta una exactitud aceptable en la estima de la calidad perceptual en un entorno de pérdida de paquetes para codificadores de voz y dado que existe una correspondencia, entre la nota PESQ y la puntuación MOS, es necesario exponer que los resultados de este estudio son mantenidos como nota PESQ. Al igual que PAMS Y PSQM, está optimizada para señales de ancho de banda telefónico.

**Figura 3.3.6.** Proceso PESQ



### **3.4. Comparación codificadores de voz.**

A continuación se muestra una tabla resumen con los codecs más utilizados actualmente:

**Bit Rate:** Indica la cantidad de información que se manda por segundo.

**Samplig Rate:** Indica la frecuencia de muestreo de la señal vocal (cada cuanto se toma una muestra de la señal analógica).

**Frame Size:** Indica cada cuantos milisegundos se envía un paquete con la información sonora.

**MOS:** Indica la calidad general del codec (valor de 1 a 5).

**Figura 3.4.1.**Cuadro Resumen de los codificadores de voz estándar

Nombre	Bit Rate kbps	Frecuencia de Muestreo (kHz)	MOS	Algoritmo	Muestras/s	Tamaño de trama ms	Lookahead(ms) Tiempo de procesamto.	Protocolos	Factor de Compresión	Uso
G.711	64	8	4.3/4.7	PCM	8000	0.125	0	H.320 H.323 H.321 H.310	1	Telefonía
G.721	32	8	4.0/4.2	ADPCM	8000	0.125	0			Telefonía
G.722	64 /56 / 48	16	3.16/4.5	ADPCM-SB	16000	0.125	1.5	H.320 H.323 H.321 H.310	3.5	Videoconferencia
G.722.1	24/32	16				20	20	H.320 H.323 H.321 H.311	4,6	Videoconferencia
G.722.2				ACELP	16000			H.320 H.323 H.321 H.312		Videoconferencia
G.723	24/40	8		ADPCM	8000	Muestreada		H.324 H.323		Telefonía Internet

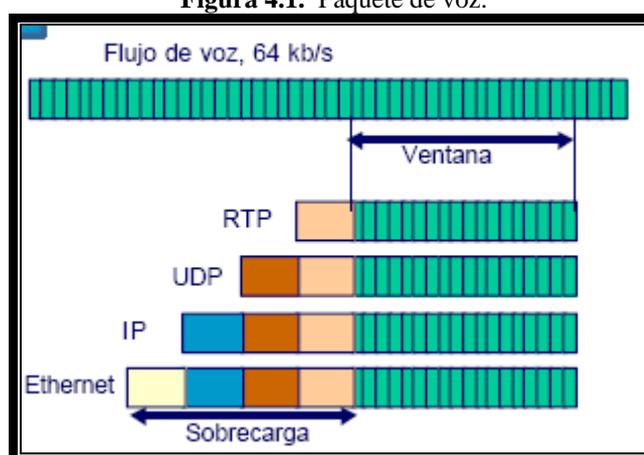
G.723.1	6,3	8	3.8/4.0	MP-MLQ	33,33	30 ms	7.5	H.324 H.323	10	Telefonía Internet
G.723.1	5,3	8	3.5/3.7	ACELP	33,33	30 ms	7.5	H.324 H.323	12	Telefonía Internet
G.726	16/24/32/40	8	3.7/4.3	ADPCM	8000	0.125	0		2	Telefonía
G.728	16	8	3,61	LD-CELP	1600	0.625	0	H.320 H.323 H.321 H310	4	Telefonía y Videoconferencia
G.729	8	8	3.9/4.2	CS-ACELP	100	10	5	H.325 H.323	8	Telefonía Internet
G.729A	8	8	3.7/4.2	CS-ACELP	100	10	5	H.325 H.324	8	Telefonía Internet

## Capítulo 4. Demostración Práctica

### Introducción

Para poder transmitir las muestras codificadas de voz sobre redes de datos, es necesario armar “paquetes”, (cada paquete se obtiene al juntar varias muestras) y se debe tener en cuenta el tamaño de carga útil y la sobrecarga por cabeceras. El tamaño de carga útil viene determinado por el tamaño de las tramas de información que entrega el codificador y por el número de tramas que se desean transportar en un paquete. En el cálculo de la sobrecarga por cabeceras se tienen en cuenta las cabeceras que añaden los sucesivos protocolos (RTP, UDP, IP y capas inferiores).

Figura 4.1. Paquete de voz.



En el caso más sencillo, por ejemplo en una sesión de voz entre dos terminales VoIP, la cabecera RTP se compone de 12 octetos, a los que hay que sumar los 8 de la cabecera UDP y los 20 de la IP. Los octetos de los niveles inferiores dependen de la tecnología concreta utilizada (por ejemplo 6 octetos para PPP).

### 4.1. Procedimiento a seguir para la evaluación de los codecs

#### 4.1.1. Fórmulas

A continuación se nombran algunas definiciones que serán utilizadas en las fórmulas

**Tamaño de trama:** Representa la longitud del tráfico de voz medida en tiempo

**Retardo de procesamiento:** Para correr el algoritmo de codificación sobre una trama.

**Tasa de bits:** Tasa de salida cuando la entrada es PCM a 64 kbps.

**Ancho de banda:** Velocidad de transmisión

a) Tamaño de trama:

$$\text{TamañoTrama} = \frac{\text{BitRate}(kbps) * \text{DuraciónTrama}(ms)}{8bits / byte} = \text{bytes} / \text{trama}$$

b) Tamaño de paquetes:

$$\text{TamañoPaquetes} = \text{TamañoTrama}(\text{bytes} / \text{trama}) * \text{NúmeroTramas} = \text{bytes} / \text{paquete}$$

c) Tiempo de duración del paquete:

$$\text{TiempoDuraciónPaquete} = \text{DuraciónTrama}(ms) * \text{NúmeroTramas} = ms$$

d) Tasa de paquetes:

$$\text{TasaPaquetes} = \frac{1}{\text{TiempoDuraciónPaquete}} = \text{paquetes} / \text{segundo}$$

e) Retardo de codificación:

$$\text{RetardoCodificación} = \text{DuraciónTrama} * \text{NúmeroTramas} + \text{TiempoLookAhead}(ms) + \text{TiempoProcesamiento} = ms$$

f) Retardo de decodificación:

$$\text{RetardoDecodificación} = \frac{\text{RetardoCodificación}}{2} = ms$$

g) Retardo Total:

$$\text{RetardoTotal} = \text{RetardoCodificación} + \text{RetardoDecodificación} + \text{Retardo Red} = ms$$

h) Ancho de Banda:

$$\text{AnchoBanda} = \frac{\text{TamañoPaquetes} + \text{H RTP} + \text{H UDP} + \text{H IP} + \text{H ETH}}{\text{TiempoDuraciónPaquete}} * 8bits / byte = kbps$$

## 4.2. Aplicación

### 4.2.1. Aplicación de las fórmulas

1) Usando el codificador G.729 con 6 tramas por paquete determinar:

- a) Tamaño de trama.
- b) Tamaño de paquetes.
- c) Tiempo de duración del paquete.
- d) Tasa de paquetes.
- e) Retardo de codificación.
- f) Retardo de decodificación.
- g) Ancho de Banda.

Datos:

Codec: G.729.

Duración trama: 10 ms.

Retardo Lookahead: 5 ms.

Bit Rate (Ancho de Banda): 8 kbps.

Cabecera RTP: 12 bytes.

Cabecera UDP: 8 bytes.

Cabecera IP: 20 bytes.

Cabecera Ethernet: 18 bytes.

Desarrollo:

- a) Tamaño de trama:

$$\text{TamañoTrama} = \frac{\text{BitRate}(kbps) * \text{DuraciónTrama}(ms)}{8bits / byte} = \text{bytes} / \text{trama}$$

$$\text{TamañoTrama} = \frac{8kbps * 10ms}{8bits / byte} = 10bytes / trama$$

- b) Tamaño de paquetes:

$$\text{TamañoPaquetes} = \text{TamañoTrama}(\text{bytes} / \text{trama}) * \text{NúmeroTramas} = \text{bytes} / \text{paquete}$$

$$\text{TamañoPaquetes} = 10\text{bytes} / \text{trama} * 6 = 60\text{bytes} / \text{paquete}$$

c) Tiempo de duración del paquete:

$$\text{TiempoDuraciónPaquete} = \text{DuraciónTrama} * \text{NúmeroTramas} = \text{ms}$$

$$\text{TiempoDuraciónPaquete} = 10\text{ms} * 6 = 60\text{ms}$$

d) Tasa de paquetes:

$$\text{TasaPaquetes} = \frac{1}{\text{TiempoDuraciónPaquete}} = \text{paquetes} / \text{segundo}$$

$$\text{TasaPaquetes} = \frac{1}{60\text{ms}} = 16.67 \text{ paquetes} / \text{segundo}$$

e) Retardo de codificación:

$$\text{RetardoCodificación} = (\text{DuraciónTrama} * \text{NúmeroTramas}) + \text{TiempoLookAhead} + \text{TiempoProcesamiento} = \text{ms}$$

$$\text{RetardoCodificación} = (10\text{ms} * 6) + 5\text{ms} + 10\text{ms} = 75\text{ms}$$

f) Retardo de decodificación:

$$\text{RetardoDecodificación} = \frac{\text{RetardoCodificación}}{2} = \text{ms}$$

$$\text{RetardoDecodificación} = \frac{75}{2} = 37.5\text{ms}$$

g) Ancho de Banda

$$\text{AnchoBanda} = \frac{\text{TamañoPaquetes} + \text{H RTP} + \text{H UDP} + \text{H IP} + \text{H ETH}}{\text{TiempoDuraciónPaquete}} * 8\text{bits} / \text{byte} = \text{kbps}$$

$$\text{AnchoBanda} = \frac{60\text{bytes} + 12\text{bytes} + 8\text{bytes} + 20\text{bytes} + 18\text{bytes}}{60\text{ms}} * 8\text{bits} / \text{byte} = 15.73\text{kbps}$$

2) Se utiliza un codificador G.729A con 4 tramas por paquete y se conoce que el retardo de la red es de 40 ms. Calcular el retardo total unidireccional.

a) Retardo de codificación.

$$\text{RetardoCodificación} = (\text{DuraciónTrama} * \text{NúmeroTramas}) + \text{TiempoLookAhead} + \text{TiempoProcesamiento} = \text{ms}$$

$$\text{RetardoCodificación} = (10\text{ms} * 4) + 5\text{ms} + 10\text{ms} = 55\text{ms}$$

a) Retardo de decodificación:

$$\text{RetardoDecodificación} = \frac{\text{RetardoCodificación}}{2} = \text{ms}$$

$$\text{RetardoDecodificación} = \frac{55}{2} = 27.5\text{ms}$$

c) Retardo total:

$$\text{RetardoTotal} = \text{RetardoCodificación} + \text{RetardoDecodificación} + \text{RetardoRed} = \text{ms}$$

$$\text{RetardoTotal} = 55\text{ms} + 27.5\text{ms} + 40\text{ms} = 122.5\text{ms}$$

122.5 ms < 150 ms que es el retardo aceptable según estándar europeo.

3) Calcular el ancho de banda sin comprimir las cabeceras y con compresión de cabeceras a 2 bytes.

Considerando el codificador G.729 de 8 kbps, si se transmite paquetes de 20 bytes cada 20ms, hasta el nivel IP.

Sin compresión:

$$\text{AnchoBanda} = \frac{\text{TamañoPaquetes} + \text{H RTP} + \text{H UDP} + \text{H IP}}{\text{TiempoDuraciónPaquete}} * 8\text{bits} / \text{byte} = \text{kbps}$$

$$\text{AnchoBanda} = \frac{20\text{bytes} + 12\text{bytes} + 8\text{bytes} + 20\text{bytes}}{20\text{ms}} * 8\text{bits} / \text{byte} = 24\text{kbps}$$

Con compresión:

$$\text{AnchoBanda} = \frac{20\text{bytes} + 2\text{bytes}}{20\text{ms}} * 8\text{bits} / \text{byte} = 8.8\text{kbps}$$

4) Cuál será el ancho de banda total que usa G.729 con 4 y 8 tramas

Datos

Codec: G.729

Trama: 10 ms

Velocidad: 8 Kbps

Ancho de banda con 4 tramas:

a) Tamaño de trama:

$$\text{TamañoTrama} = \frac{\text{BitRate}(\text{kbps}) * \text{DuraciónTrama}(\text{ms})}{8\text{bits} / \text{byte}} = \text{bytes} / \text{trama}$$

$$\text{TamañoTrama} = \frac{8\text{kbps} * 10\text{ms}}{8\text{bits} / \text{byte}} = 10\text{bytes} / \text{trama}$$

b) Tamaño de paquetes:

$$\text{TamañoPaquetes} = \text{TamañoTrama}(\text{bytes} / \text{trama}) * \text{NúmeroTramas} = \text{bytes} / \text{paquete}$$

$$\text{TamañoPaquetes} = 10\text{bytes} / \text{trama} * 4 = 40\text{bytes} / \text{paquete}$$

c) Ancho de Banda:

$$\text{AnchoBanda} = \frac{\text{TamañoPaquetes} + \text{H RTP} + \text{H UDP} + \text{H IP}}{\text{TiempoDuraciónPaquete}} * 8\text{bits} / \text{byte} = \text{kbps}$$

$$\text{AnchoBanda} = \frac{40\text{bytes} + 12\text{bytes} + 8\text{bytes} + 20\text{bytes}}{40\text{ms}} * 8\text{bits} / \text{byte} = 16\text{kbps}$$

Ancho de banda con 8 tramas:

a) Tamaño de trama:

$$\text{TamañoTrama} = \frac{\text{BitRate}(kbps) * \text{DuraciónTrama}(ms)}{8bits / byte} = \text{bytes} / \text{trama}$$

$$\text{TamañoTrama} = \frac{8kbps * 10ms}{8bits / byte} = 10bytes / trama$$

b) Tamaño de paquetes:

$$\text{TamañoPaquetes} = \text{TamañoTrama}(\text{bytes} / \text{trama}) * \text{NúmeroTramas} = \text{bytes} / \text{paquete}$$

$$\text{TamañoPaquetes} = 10bytes / trama * 8 = 80bytes / paquete$$

c) Ancho de Banda:

$$\text{AnchoBanda} = \frac{\text{TamañoPaquetes} + \text{H RTP} + \text{H UDP} + \text{H IP}}{\text{TiempoDuraciónPaquete}} * 8bits / byte = kbps$$

$$\text{AnchoBanda} = \frac{80bytes + 12bytes + 8bytes + 20bytes}{80ms} * 8bits / byte = 12kbps$$

5) Calcular el delta del *time stamp* del codificador G.729, con 4,8 y 16 tramas.

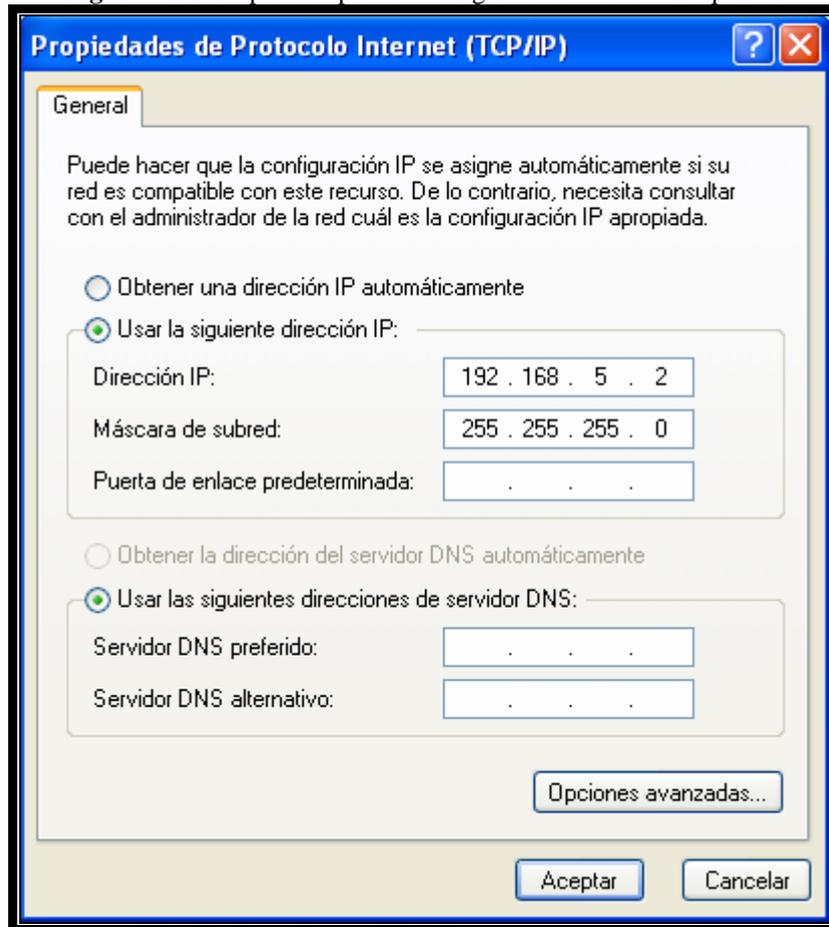
Para esta práctica utilizaremos la herramienta Ethereal (analizador de protocolos) y la herramienta Openphone (Softphone), a continuación se detallan los pasos a seguir:

5.1) Se debe tener configurado la red, para el ejemplo se utilizará una red punto a punto con las siguientes direcciones:

Máquina 1: 192.168.5.2

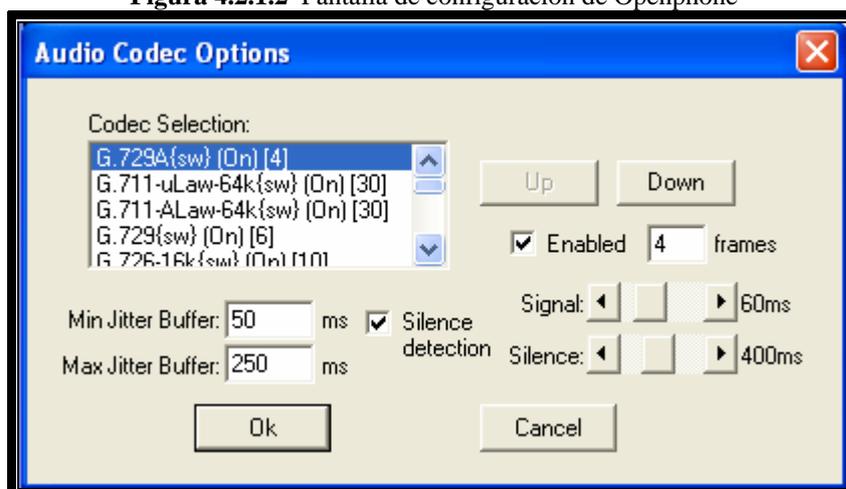
Máquina 2: 192.168.5.1

Figura 4.2.1.1 Opciones para la configuración de la red máquina 2



5.2) Configurar el Openphone (Ver Anexo 2: Openphone) para el ejemplo utilizaremos el codec G.729, en donde indicaremos el número de tramas.

Figura 4.2.1.2 Pantalla de configuración de Openphone



5.3) Al momento de realizar la llamada se debe capturar los paquetes con la herramienta Ethereal (Ver Anexo 3 Ethereal) y filtrar por el protocolo RTP.

4 tramas =  $480 - 160 = 320$  muestras

Figure 5.3.1. Captura Ethereal 4 tramas

No. -	Time	Source	Destination	Protocol
19	4.294497	192.168.5.1	192.168.5.2	RTP
20	4.328581	192.168.5.1	192.168.5.2	RTP
21	4.369506	192.168.5.1	192.168.5.2	RTP
23	4.409904	192.168.5.2	192.168.5.1	RTP

⊕ Frame 19 (94 bytes on wire, 94 bytes captured)				
⊕ Ethernet II, Src: QuantaCo_3f:83:56 (00:16:36:3f:83:56), Dst:				
⊕ Internet Protocol, Src: 192.168.5.1 (192.168.5.1), Dst: 192.168.5.2 (192.168.5.2)				
⊕ User Datagram Protocol, Src Port: 5000 (5000), Dst Port: 5000 (5000)				
⊖ Real-Time Transport Protocol				
⊕ [Stream setup by H245 (frame 9)]				
10.. .... = Version: RFC 1889 Version (2)				
..0. .... = Padding: False				
...0 .... = Extension: False				
.... 0000 = Contributing source identifiers count: 0				
1... .... = Marker: True				
Payload type: ITU-T G.729 (18)				
Sequence number: 2267				
Timestamp: 160				
Synchronization source identifier: 2321950806				
Payload: E9BE88C000FAD63D42FDA8DA0DDB71E1B3300D522D54B80B...				

Figure 5.3.2. Captura Ethereal 4 tramas

No. -	Time	Source	Destination	Protocol
19	4.294497	192.168.5.1	192.168.5.2	RTP
20	4.328581	192.168.5.1	192.168.5.2	RTP
21	4.369506	192.168.5.1	192.168.5.2	RTP
23	4.409904	192.168.5.2	192.168.5.1	RTP

⊕ Frame 20 (94 bytes on wire, 94 bytes captured)				
⊕ Ethernet II, Src: QuantaCo_3f:83:56 (00:16:36:3f:83:56), Dst:				
⊕ Internet Protocol, Src: 192.168.5.1 (192.168.5.1), Dst: 192.168.5.2 (192.168.5.2)				
⊕ User Datagram Protocol, Src Port: 5000 (5000), Dst Port: 5000 (5000)				
⊖ Real-Time Transport Protocol				
⊕ [Stream setup by H245 (frame 9)]				
10.. .... = Version: RFC 1889 Version (2)				
..0. .... = Padding: False				
...0 .... = Extension: False				
.... 0000 = Contributing source identifiers count: 0				
0... .... = Marker: False				
Payload type: ITU-T G.729 (18)				
Sequence number: 2268				
Timestamp: 480				
Synchronization source identifier: 2321950806				
Payload: 4DCBCFB266B60F6CBF1A17BECFAD0F0B9187A1712B384ED5...				

8 tramas = 160-800 = 640 muestras

Figure 5.3.3. Captura Ethereal 8tramas

No. ↓	Time	Source	Destination	Protocol
21	16.742205	192.168.5.1	192.168.5.2	RTP
23	16.816858	192.168.5.1	192.168.5.2	RTP
24	16.820555	192.168.5.2	192.168.5.1	RTP
25	16.882197	192.168.5.2	192.168.5.1	RTP

Filter: rtp

Frame 24 (134 bytes on wire, 134 bytes captured)

- Ethernet II, Src: Asiarock\_17:e3:36 (00:0b:6a:17:e3:36), Dst: ...
- Internet Protocol, Src: 192.168.5.2 (192.168.5.2), Dst: 192.168.5.1
- User Datagram Protocol, Src Port: 5004 (5004), Dst Port: 5000
- Real-Time Transport Protocol
  - [Stream setup by H245 (frame 12)]
    - 10.. .... = Version: RFC 1889 Version (2)
    - ..0. .... = Padding: False
    - ...0 .... = Extension: False
    - .... 0000 = Contributing source identifiers count: 0
    - 1... .... = Marker: True
    - Payload type: ITU-T G.729 (18)
    - Sequence number: 35562
    - Timestamp: 160
    - Synchronization source identifier: 460418932
    - Payload: A40FCCA000FADC9C409500E032483F221B11839074565040...

Figure 5.3.4 Captura Ethereal 8 tramas

No. ↓	Time	Source	Destination	Protocol
21	16.742205	192.168.5.1	192.168.5.2	RTP
23	16.816858	192.168.5.1	192.168.5.2	RTP
24	16.820555	192.168.5.2	192.168.5.1	RTP
25	16.882197	192.168.5.2	192.168.5.1	RTP

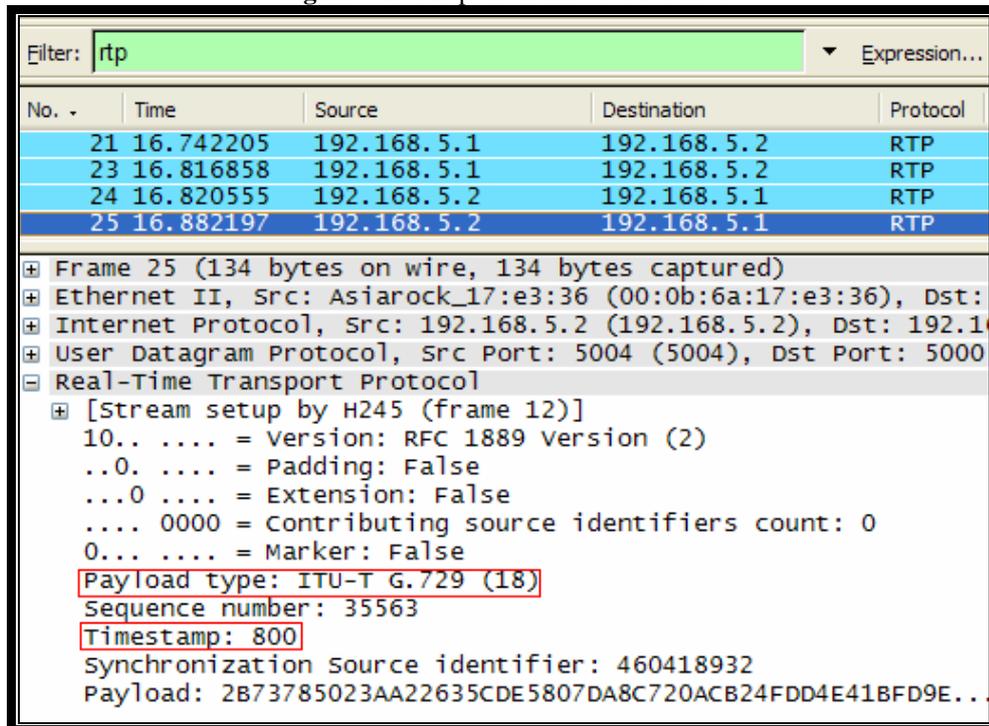
Filter: rtp

Frame 25 (134 bytes on wire, 134 bytes captured)

- Ethernet II, Src: Asiarock\_17:e3:36 (00:0b:6a:17:e3:36), Dst: ...
- Internet Protocol, Src: 192.168.5.2 (192.168.5.2), Dst: 192.168.5.1
- User Datagram Protocol, Src Port: 5004 (5004), Dst Port: 5000
- Real-Time Transport Protocol
  - [Stream setup by H245 (frame 12)]
    - 10.. .... = Version: RFC 1889 Version (2)
    - ..0. .... = Padding: False
    - ...0 .... = Extension: False
    - .... 0000 = Contributing source identifiers count: 0
    - 0... .... = Marker: False
    - Payload type: ITU-T G.729 (18)
    - Sequence number: 35563
    - Timestamp: 800
    - Synchronization source identifier: 460418932
    - Payload: 2B73785023AA22635CDE5807DA8C720ACB24FDD4E41BFD9E...

16 tramas = 1440-160 = 1280 muestras

Figure 5.3.5. Captura Ethereal 16 tramas



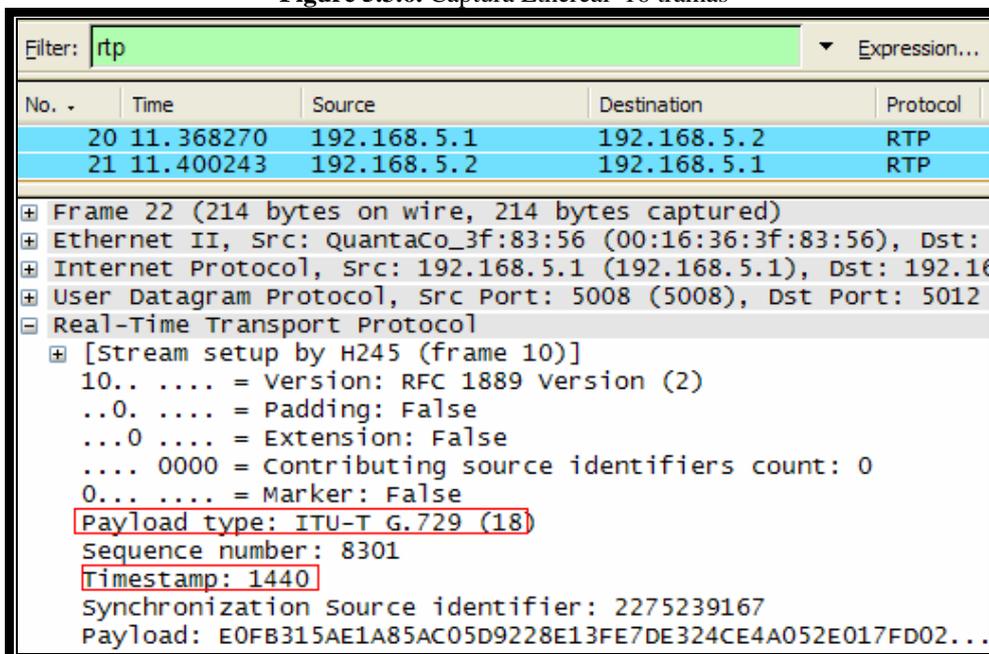
Filter: `rtp` Expression...

No. -	Time	Source	Destination	Protocol
21	16.742205	192.168.5.1	192.168.5.2	RTP
23	16.816858	192.168.5.1	192.168.5.2	RTP
24	16.820555	192.168.5.2	192.168.5.1	RTP
25	16.882197	192.168.5.2	192.168.5.1	RTP

Frame 25 (134 bytes on wire, 134 bytes captured)

- Ethernet II, Src: Asiarock\_17:e3:36 (00:0b:6a:17:e3:36), Dst: ...
- Internet Protocol, Src: 192.168.5.2 (192.168.5.2), Dst: 192.168.5.1 (192.168.5.1)
- User Datagram Protocol, Src Port: 5004 (5004), Dst Port: 5000
- Real-Time Transport Protocol
  - [Stream setup by H245 (frame 12)]
    - 10.. .... = Version: RFC 1889 Version (2)
    - ..0. .... = Padding: False
    - ...0 .... = Extension: False
    - .... 0000 = Contributing source identifiers count: 0
    - 0... .... = Marker: False
    - Payload type: ITU-T G.729 (18)
    - Sequence number: 35563
    - Timestamp: 800
    - Synchronization source identifier: 460418932
    - Payload: 2B73785023AA22635CDE5807DA8C720ACB24FDD4E41BFD9E...

Figure 5.3.6. Captura Ethereal 16 tramas



Filter: `rtp` Expression...

No. -	Time	Source	Destination	Protocol
20	11.368270	192.168.5.1	192.168.5.2	RTP
21	11.400243	192.168.5.2	192.168.5.1	RTP

Frame 22 (214 bytes on wire, 214 bytes captured)

- Ethernet II, Src: QuantaCo\_3f:83:56 (00:16:36:3f:83:56), Dst: ...
- Internet Protocol, Src: 192.168.5.1 (192.168.5.1), Dst: 192.168.5.2 (192.168.5.2)
- User Datagram Protocol, Src Port: 5008 (5008), Dst Port: 5012
- Real-Time Transport Protocol
  - [Stream setup by H245 (frame 10)]
    - 10.. .... = Version: RFC 1889 Version (2)
    - ..0. .... = Padding: False
    - ...0 .... = Extension: False
    - .... 0000 = Contributing source identifiers count: 0
    - 0... .... = Marker: False
    - Payload type: ITU-T G.729 (18)
    - Sequence number: 8301
    - Timestamp: 1440
    - Synchronization source identifier: 2275239167
    - Payload: E0FB315AE1A85AC05D9228E13FE7DE324CE4A052E017FD02...

5.4) Aplicar las siguientes fórmulas para verificar lo obtenido con el Ethereal.

a) 4 tramas

Frecuencia de Muestreo (*sampling rate*): 8KHz muestras/s

Duración Trama: 10ms

$$\Delta TS = \text{NúmeroTramas} * \text{DuraciónTrama}(ms) * \text{SamplingRate}(kmuestras / s) = \text{muestras}$$

$$\Delta TS = 4 * 10ms * 8kmuestras / s = 320muestras$$

b) 8 tramas

Frecuencia de Muestreo (*sampling rate*): 8KHzmuestras/s

Duración Trama: 10ms

$$\Delta TS = \text{NúmeroTramas} * \text{DuraciónTrama}(ms) * \text{SamplingRate}(kmuestras / s) = \text{muestras}$$

$$\Delta TS = 8 * 10ms * 8kmuestras / s = 640muestras$$

c) 16 tramas

Frecuencia de Muestreo (*sampling rate*): 8KHzmuestras/s

Duración Trama: 10ms

$$\Delta TS = \text{NúmeroTramas} * \text{DuraciónTrama}(ms) * \text{SamplingRate}(kmuestras / s) = \text{muestras}$$

$$\Delta TS = 16 * 10ms * 8kmuestras / s = 1280muestras$$

#### 4.2.2. Calculadoras de Ancho de Banda

Otra forma de comprobar el ancho de banda que se requiere según el codec seleccionado, es utilizando las calculadoras de ancho de banda disponibles en Internet. Entre las que consideramos más importantes tenemos las siguientes:

Sitio: <http://www.packetizer.com/iptel/bandcalc.html>

Figura 4.2.2.1 Calculadora Packetizer sin supresión de silencios.

# VoIP Bandwidth Calculator™

### Parameters

<input type="radio"/>	Payload is	G.729 8kbps	with <sup>2</sup>	80	ms or	8	frames <sup>3</sup> per packet.
<input type="radio"/>	RTP is	RTP (RFC 3550)					
<input type="radio"/>	UDP						
<input type="radio"/>	IP						
<input checked="" type="radio"/>	Link	ethernet 802.3					
<input type="checkbox"/>	Silence Suppression <sup>4</sup>	<input type="checkbox"/>	RTCP <sup>5</sup>	1	channel(s) <sup>6</sup>		

### Results

<p><i>Bandwidth</i></p> <p>Average<sup>7</sup>: 15.8 kbps</p> <p>Maximum<sup>8</sup>: 15.8 kbps</p> <p><i>Packet rate</i><sup>12</sup></p> <p>Average: 12.5 pps</p> <p>Maximum: 12.5 pps</p>	<p><i>Delay</i><sup>9</sup></p> <p>Frame: 10 ms</p> <p>Lookahead: 5 ms</p> <p>Algorithmic: 85 ms</p>	<p><i>Performance</i></p> <p>DSP MIPS<sup>10</sup>: 20 - 25</p> <p>MOS<sup>11</sup>: 3.9 - 4.2</p>
--	--	--

Como observamos, en la Figura 4.2.2.1. , en esta calculadora se escoge el codificador y se ingresa el numero de tramas, y además de obtener el ancho de banda se tienen resultados como el retardo y el desempeño.

Figura 4.2.2.2 Calculadora Packetizer con supresión de silencios.

# VoIP Bandwidth Calculator™

### Parameters

<input type="radio"/> Payload is	G.729 8kbps	with <sup>2</sup>	80	ms or	8	frames <sup>3</sup> per packet.
<input type="radio"/> RTP is	RTP (RFC 3550)					
<input type="radio"/> UDP						
<input type="radio"/> IP						
<input checked="" type="radio"/> Link	ethernet 802.3					
<input checked="" type="checkbox"/> Silence Suppression <sup>4</sup>	<input type="checkbox"/> RTCP <sup>5</sup>	1	channel(s) <sup>6</sup>			

### Results

<i>Bandwidth</i>	<i>Delay</i> <sup>9</sup>	<i>Performance</i>
Average <sup>7</sup> : 7.9 kbps	Frame: 10 ms	DSP MIPS <sup>10</sup> : 20 - 25
Maximum <sup>8</sup> : 15.8 kbps	Lookahead: 5 ms	MOS <sup>11</sup> : 3.9 - 4.2
<i>Packet rate</i> <sup>12</sup>	Algorithmic: 85 ms	
Average: 6.3 pps		
Maximum: 12.5 pps		

En la Figura 4.2.2.2., se observa una disminución importante del ancho de banda que se logra con la técnica de supresión de silencios.

Sitio: <http://www.erlang.com/calculator/lipb/>

Figura 4.2.2.3. Calculadora Erlang

The image shows a software window titled "Lines to VoIP Bandwidth Calculator". The window has a blue title bar with a close button (X) on the right. The main content area is divided into several sections. The first section is "Coding algorithm" with a dropdown menu showing "G.729A (CS-CELP) 8kbps compression". The second section is "Packet duration" with a dropdown menu showing "10 milliseconds (1 sample)". Below these are two side-by-side sections. The left section is "Voice paths" with a radio button for "Unknown" and a text box containing the number "1". The right section is "IP Bandwidth (kbps)" with a radio button for "Unknown" and a text box containing the number "40". At the bottom of the window, there are three buttons: "Calc.", "Results", and "Help".

La calculadora Erlang, tiene la opción de escoger el número de canales por el que se transmitirá la voz, en las fórmulas se ha calculado con un canal de voz.

## **Capítulo 5. Conclusiones y Recomendaciones**

### **Conclusiones**

Podemos concluir que la calidad de los codificadores viene dada por varios factores entre los que podemos mencionar: el algoritmo usado para su codificación, los mecanismos de reducción de ancho de banda como son la compresión y la supresión de silencios. Además hay que tener en cuenta el tamaño de la trama de los codificadores y el número de tramas al momento de formar los paquetes para la transmisión de voz. A través de herramientas como el analizador de protocolos Ethereal se puede capturar todos los paquetes que intervienen en una comunicación y esto representa una fuente de información con la cual se puede verificar los resultados obtenidos por medio de las fórmulas científicas aplicadas; también podemos comprobar estos valores con el uso de las calculadoras de ancho de banda. Finalmente podemos decir que debe existir un compromiso entre los recursos disponibles y las expectativas de calidad, ya que a mayores tasas de transmisión se obtiene mayor calidad de voz pero se necesitan mayores recursos.

### **Recomendaciones**

Como recomendación general podemos indicar que para la selección del codec que mejor se adapte a la necesidad de la empresa, se debe tener muy en cuenta los recursos con los que se cuenta, como el ancho de banda disponible y a través de las fórmulas y las herramientas expuestas realizar las comprobaciones respectivas, para verificar si es que el codec seleccionado cumple con las características de calidad deseada.

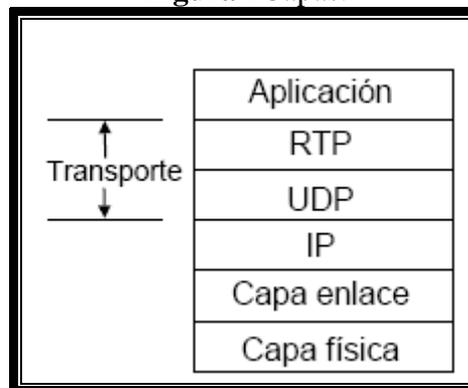
## Referencias

## Glosario

### RTP: Real Time Protocol

El protocolo RTP es utilizado generalmente sobre UDP, segmentos de datos (video ó audio) son generados por el transmisor de la aplicación multimedia, estos segmentos son encapsulados en paquetes RTP y cada paquete RTP es luego encapsulado en segmentos UDP. Como RTP provee servicios a la capa de aplicación, puede ser visto como una subcapa de la capa de transporte.

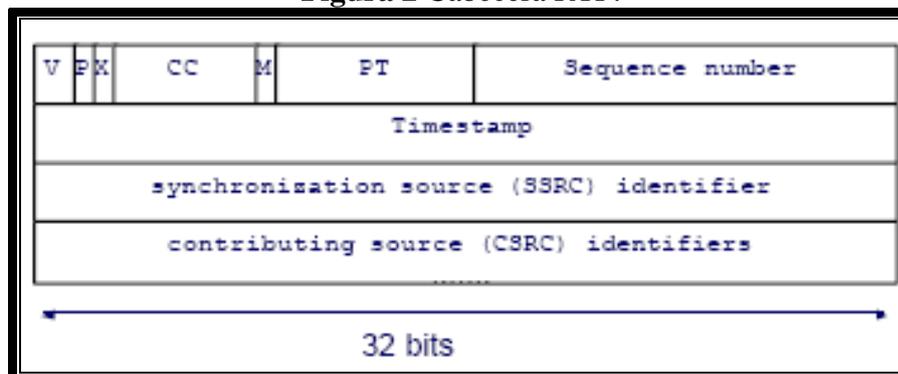
**Figura 1** Capas.



### Cabecera de los paquetes RTP

Los cuatro campos principales de la cabecera RTP son: Tipo de Carga Útil, Número de Secuencia, Marca de Tiempo y el Identificador de la Fuente.

**Figura 2** Cabecera RTP.



**Tipo de carga útil:** Este campo tiene 7 bits de largo, esto permite 27 o sea 128 tipos de carga útil diferentes. El tipo de carga útil indica por ejemplo para audio, el tipo de codificación usada como PCM para el codec G.723, etc. Si el transmisor decide cambiar el tipo de codificación en medio de una sesión, le informa al transmisor del cambio por medio de este campo. El transmisor podría cambiar la codificación para aumentar la calidad, o para reducir el flujo de bits de la sesión.

**Figura 3** Tipos de Carga Útil.

■ Payload type (7 bits)			
Payload Type	Formato	Medio	Clock Rate
0	PCM mu-law	Audio	8 kHz
3	GSM	Audio	8 kHz
4	G.723	Audio	8 kHz
8	PCM A-law	Audio	8 kHz
9	G.722	Audio	8 kHz
14	MPEG Audio	Audio	90 kHz
15	G.728	Audio	8 kHz
18	G.729	Audio	8 kHz
28	Motion JPEG	Video	90 kHz
31	H.261	Video	90 kHz
34	H.263	Video	90 kHz

**Número de Secuencia:** Este campo tiene 16 bits de largo. El número de secuencia se incrementa secuencialmente con cada paquete RTP enviado y es utilizado por el receptor para detectar paquetes perdidos. Si es que el receptor recibe paquetes no secuenciales sabe exactamente que paquetes se perdieron, se puede implementar en el receptor algún algoritmo que regenere los datos perdidos para que su incidencia sea mínima (por ejemplo, agregar ruido blanco, o "interpolar" los datos faltantes).

**Marca de Tiempo:** Este campo tiene 32 bits de largo, contiene el instante de creación del primer *byte* del paquete RTP. El receptor utiliza esta marca de tiempo para remover el  *jitter* de los paquetes y para mantener el sincronismo de los distintas cadenas RTP.

**Identificador de la fuente de sincronismo (SSRC):** El campo SSRC tiene 32 bits de largo e identifica la fuente de la cadena RTP. Cada cadena en una sesión RTP

tiene un SSRC distinto. El SSRC no es la IP del transmisor, sino un número aleatorio que el transmisor asigna a cada cadena.

### **RTCP: RTP Control Protocol**

El protocolo RTP consta de en realidad de dos protocolos, el RTP y el RTCP (establecido por RFC 3550). RTCP se basa en la transmisión periódica de paquetes de control a todos los participantes de la sesión, utilizando el mismo mecanismo de transporte que los paquetes RTP. Utiliza un puerto distinto que RTP, por lo general se utilizan puertos consecutivos donde el par es asignado a el flujo RTP y el impar al flujo RTCP. Este protocolo de control tiene las siguientes funcionalidades:

**Feedback RTCP:** Es el encargado de la distribución de la información de realimentación de la calidad de los datos, esta información se puede utilizar para tomar acciones para mejorar el rendimiento de la aplicación (por ejemplo cambiar la codificación, para disminuir la tasa de bits).

**Identificación:** Es necesario tener un identificador persistente a nivel de capa de transporte para las fuentes RTP. Este identificador se llama Nombre Canónico. Dado que el SSRC puede cambiar debido a un conflicto o debido a el reinicio del programa, los receptores utilizan el nombre canónico para mantener una lista de los participantes de la sesión. También se utiliza para agrupar distintas cadenas de un participante, por ejemplo para sincronizar audio con video.

**Control de Participantes:** Las funciones anteriores obligan a los participantes de la sesión a enviar periódicamente paquetes RTCP, la tasa de envío de esos paquetes tiene que ser controlada para permitir acceso a más participantes. Dado que los participantes envían los paquetes de control a todos los demás participantes, cada uno observa localmente el número de participantes y calcula a partir de este número la tasa de envío de paquetes de control.

## **Bibliografía**

### Libros

INTEGRACIÓN DE VOZ Y DATOS CALL CENTERS: TECNOLOGIA Y APLICACIONES.

HUIDOBRO MOYA, José Manuel.

ROLDAN MARTINEZ, David.

Grupo Editorial McGraw Hill, España, 2003. 12/01/2007

### Artículos de Internet

**AEQ Professional Broadcast** <http://www.aeq.es/spn/man/course.pdf> 15/01/2007

### **Fundación Escuela Latinoamericana de Redes**

<http://www.eslared.org.ve/articulos/ermannov/voip.pdf> 18/01/2007

### **Grupo de Investigación en Señales, Telemática y Comunicaciones, GSTC**

<http://ceres.ugr.es/~alumnos/luis/codif.htm> 20/01/2007

### **Universidad de Chile**

<http://www-scf.usc.edu/~busso/Documents/MasterTesis.pdf> 21/01/2007

### **Universidad Fasta**

<Http://web2.ufasta.edu.ar/Teleinformatica/Modulacion%20y%20Codificacion/PCM/etbpveu03.pdf> 22/01/2007

**Tech Faq** [www.tech-faq.com](http://www.tech-faq.com) 25/01/2007

**VoIP Foro** - [Codecs\\_g711, g729, iLBC, gsm.htm](#) 26/01/2007

**VoIP Foro** - [QoS - jitter - Causas, soluciones y valores recomendados.htm](#)

28/01/2007

**VoIP Foro** <http://www.voipforo.com/codec/codecs.php#g711> 02/02/2007

**Wikipedia** <http://es.wikipedia.org/wiki/G.729> 03/02/2007

**Wikipedia** <http://es.wikipedia.org/wiki/G.711> 04/02/2007

**Instituto de Ingeniería Eléctrica de la Universidad de la República**

<http://iie.fing.edu.uy/investigacion/grupos/artes/pqos/manual.pdf>

07/02/2007

**Erlang** <http://www.erlang.com/calculator/lipb/>

08/02/2007

**Packetizer** <http://www.packetizer.com/iptel/bandcalc.html>

08/02/2007

**Openphone** <http://www.openh323.org/>

08/02/2007

## Anexos

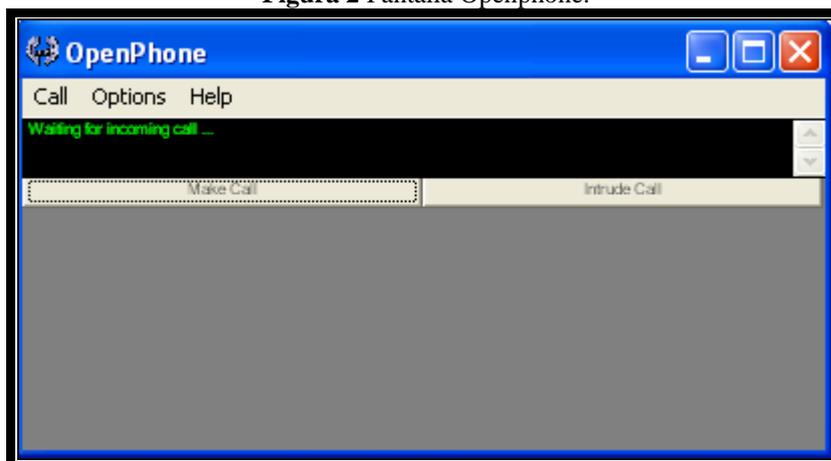
### Anexo 1: Openphone



El proyecto OpenH323 pretende crear una implementación completa, interoperable y de código abierto del protocolo de teleconferencia ITU H.323 que pueda ser utilizado por desarrolladores personales y usuarios comerciales sin ningún costo.

H.323 es un estándar que especifica los componentes, protocolos y procedimientos que proporcionan servicios de comunicación multimedia (audio, video y datos en tiempo real) sobre redes de paquetes no orientadas a la conexión y que no garanticen calidad de servicio.

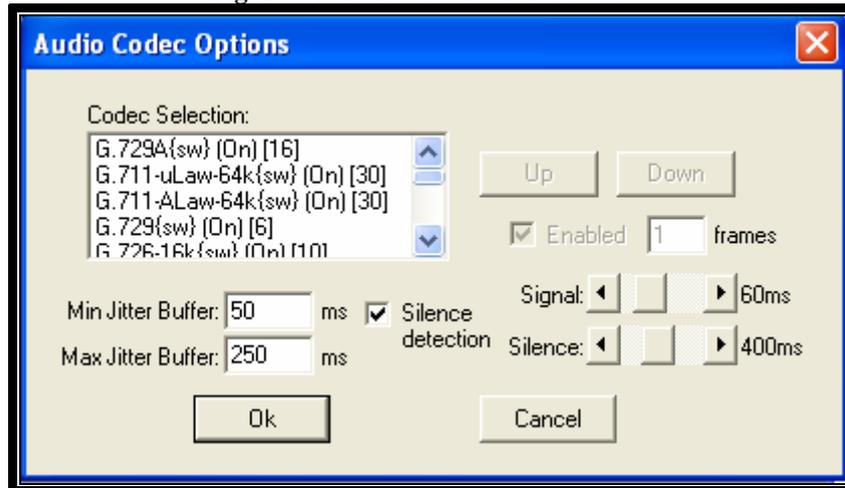
Figura 2 Pantalla Openphone.



#### Pasos para hacer una llamada:

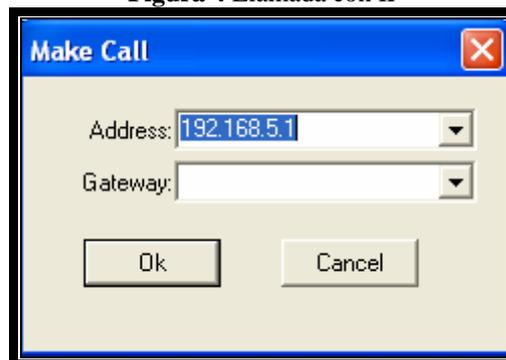
1. Seleccionar el codec de audio para la comunicación, en el menú: Options y luego escoger Audio Codec. En esta pantalla podemos ver los codecs disponibles, en la lista debemos ordenar los codecs para que en el momento de negociación del mismo sino se puede establecer la comunicación con el primer codec se va a seguir intentado con los siguientes de la lista. Además en esta pantalla debemos poner el número de tramas con los cuales vamos a trabajar, el retardo del buffer del jitter y la opción de detección de silencios.

**Figura 3** Pantalla de selección del codec.



2. En el menú la Call, escoger la opción Make Call, donde ingresamos la dirección IP o el alias de la máquina con la que nos queremos comunicar.

**Figura 4** Llamada con IP



**Figura 5** Llamada con alias.



## Anexo 2: Ethereal

Ethereal es una herramienta que nos permite observar y analizar el comportamiento de los protocolos de red, esta herramienta es capaz de monitorear el tráfico en la red y mostrarlo en una forma legible a esta herramienta se la denomina analizadores de protocolos de red, "packet sniffers" o simplemente "sniffers" (del inglés *sniff*, olfatear). Cuenta con una interfaz gráfica capaz de reconocer muchos protocolos distintos. Permite tanto revisar los paquetes de datos en una red activa como desde un archivo de captura previamente generado; es capaz de comprender diversos formatos de archivo propios de otros programas de captura, en particular el clásico tcpdump.

### Pasos para realizar una captura

1. Seleccionar el tipo de tráfico que se quiere capturar, se debe dar presionar el botón "List the available capture adapter" y luego escoger el medio e inmediatamente empieza a realizar las capturas has que se presione el botón "Stop".

Figura 1 Pantalla para selección del tráfico a capturar.

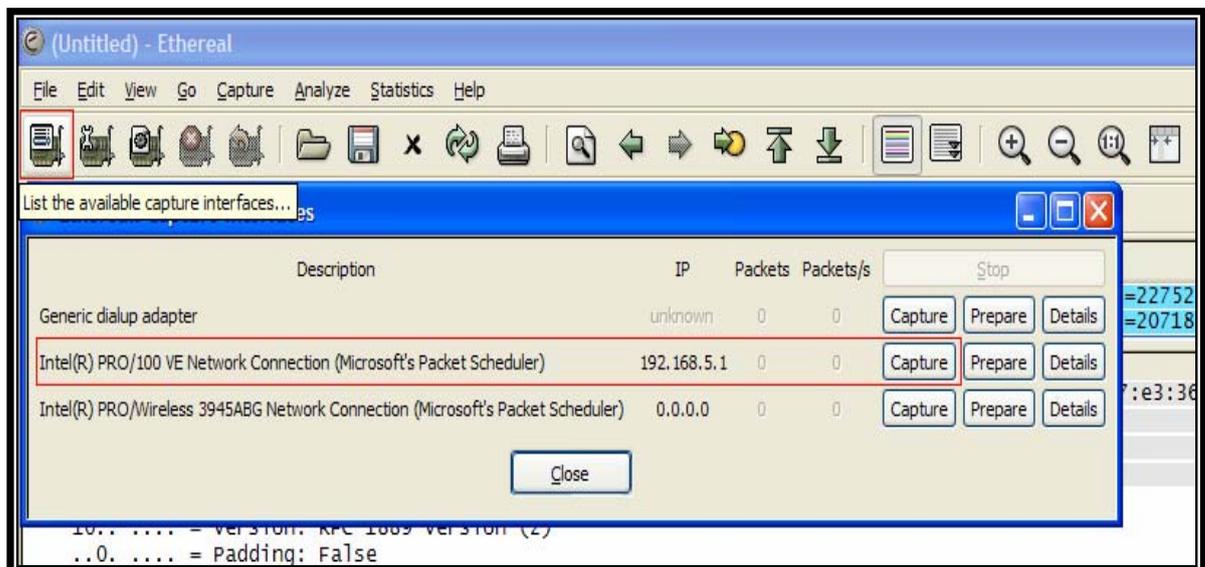
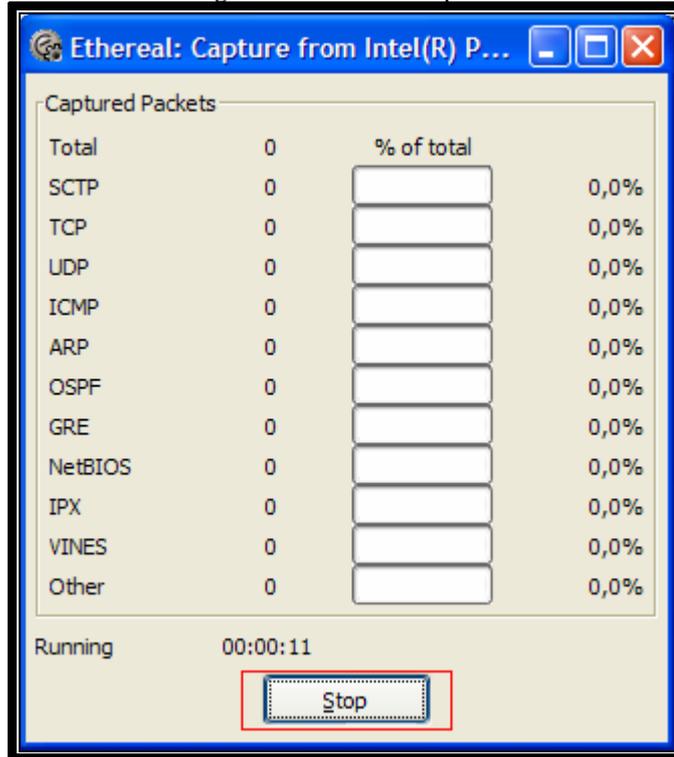


Figura 2 Pantalla de captura.



2. Una vez finalizado la captura se muestra la siguiente pantalla en donde se puede filtrar los diferentes protocolos.

Figura 3 Pantalla de resultados de la captura.

