UNIVERSIDAD TECNOLOGICA DE LA MIXTECA.

" Decodificador de Voz a 16 Kbit/s Usando la Técnica RPE-LTP. "

Tesis Profesional

Que para Obtener el TITULO de
INGENIERO EN ELECTRONICA
presenta :

Miguel Angel Ramírez Jiménez

Acatlima, Huajuapan de León, Oaxaca., Marzo '97.

Tesis presentada el 20 de Marzo de 1997
ante los siguientes sinodales:

Ing.	Hugo Suárez Onofre.
M. en C.	Hiram Ochoa Arriaga.
Ing.	Enrique Guzmán Ramírez.

Asesor:

M. en C. ***José Antonio Moreno Espinosa.***

Dedicatorias

A mi Madre , por su amor y confianza, compañera de toda mi vida.

A mi Padre, porque sin su ejemplo vivo, no sería quien ahora soy.

A Guillermo Ramírez J., por el inmenso apoyo que me brindó durante todo éste tiempo.

A Reyna V. Villagómez M., porque me brindó su apoyo moral en los momentos más difíciles para mi.

A Nestor, Jorge, Norma y Merced. A Lupe y Rafael.

A Cristhian Guadalupe y Maria de Jesús.

Miguel Angel.

Agradecimientos

A Memo por la confianza que depositó en mí desde el inicio de mi carrera.

A Nestor, Jorge, Norma y Merced. A Lupe y Rafael, por su esfuerzo diario y constante durante todos estos años tan difíciles para todos. Su ejemplo de unión y lucha estará siempre vivo en mí.

A mi gran amigo y compañero de Tesis, Alberto: por compartir conmigo los buenos momentos y acompañarme en los no tan buenos.

A Juan Carlos Alonso, por brindarme su amistad y apoyo incondicional.

A la familia López Tapia por brindarme su amistad. A ti Yadira por ser quien eres. A ti Yanet, mi gran amiga.

Un especial agradecimiento a mi Asesor por su disposición constante en ayudarme, por leer el documento y por aportar ideas que ayudaron a perfeccionar su contenido.

Un agradecimiento especial al jurado presente:

Ing. Hugo Suárez Onofre

M. en C. Hiram Ochoa Arriaga

Ing. Enrique Guzmán Ramírez

por el apoyo brindado en la fase de revisión del documento y por todas las recomendaciones y observaciones realizadas.

Miguel Angel. Ramírez Jiménez.

Resumen

El objetivo de este trabajo es proponer y al mismo tiempo implementar un *sistema decodificador de voz RPE-LTP* como complemento del proyecto *"Codificador de voz RPE-LTP"*. Como punto inicial, el sistema toma voz digital codificada a 16 Kbit/s para luego reconstruir y recuperar la voz original.

La voz reconstruida debe tener como característica principal la inteligibilidad. Aunado a lo anterior, la codificación de voz a bajo régimen con calidad de voz próxima a la telefónica comercial, es una necesidad para los nuevos servicios telemáticos siendo además una de las áreas de estudio a la que universidades y empresas de todo el mundo le dedican un esfuerzo considerable.

Gracias a ello, sobre las redes digitales de servicios integrados (ISDN) aparecen el correo de voz, la integración *voz-texto* y el almacenamiento de mensajes hablados. Ante este mercado necesitado de codificadores con reproducciones de voz fidedignas y velocidades de voz más bajas, el *Codificador-Decodificador RPE-LTP* se propone como una solución para reducir la tasa de transmisión a 16 Kbit/s.

CONTENIDO

Dedicatorias. iii
Agradecimientos. iv
Resumen. v
Introducción. ix
Contenido de la Tesis. xi

Capítulo 1 Antecedentes Generales. 1
1.1 Introducción. 1
1.2 Descripción del Aparato Auditivo. 2
1.3 Propiedades Auditivas. 5
1.4 Comportamiento de la Membrana Basilar. 7
1.5 Descargas Eléctricas en el Nervio Auditivo. 9
1.6 Modelos Auditivos. 12
1.7 Algunos Efectos Perceptuales. 15

Capítulo 2 Técnicas de Codificación de Formas de Onda. 19
2.1 Introducción. 19
2.2 Codificación en el Dominio Frecuencial. 23
2.2.1 Codificadores en Sub-Bandas (SBC). *23*
2.2.2 Codificadores de Transformada. *34*
2.3 Codificador por Predicción Adaptiva (APC). 39

Capítulo 3 Sistemas Basados en la Predicción Lineal. 47
3.1 Introducción. 47
3.1.1 Modelo TODO-POLOS. 50
3.1.1.1 Señal Determinística. 52
3.1.1.1.1 Método de Autocorrelación. 54
3.1.1.1.2 Método de Covarianza. *56*
3.1.1.2 Señal Aleatoria. *57*
3.1.1.2.1 Caso Estacionario. *58*

3.1.1.2.2 Caso No Estacionario. *58*
3.1.2 Cálculo de los Parámetros del Predictor. *59*
3.2 Excitación Multi-Pulso (MPE). 63
3.3 Excitación de Pulsos Regulares (RPE). 68
3.4 LPC de Excitación Residual con Vector de Cuantización (RELP-VQ) 76

Capítulo 4 Sintetizador RPE-LTP. 81
4.1 Introducción. 81
4.2 Decodificación de Parámetros para el Sintetizador. 83
4.2.1 LAR codificados, LARc. 95
4.2.2 Retardo de correlación, Nc. 96
4.2.3 Factor de ganancia, bc. 97
4.2.4 Selección de rejilla RPE, M. 97
4.2.5 Amplitud máxima codificada, xmaxc. 97
4.2.6 Muestras RPE normalizadas, xMc. 98
4.3 Reconstrucción del seleccionador de rejilla RPE. 98
4.4 Interpolación de LAR y transformación de LAR a PARCOR. 99
4.5 Filtro de Síntesis de Término Largo. 99
4.6 Filtro de Síntesis de Término Corto. 100
4.7 Pseudocódigo del Sintetizador. 101

Capítulo 5 Evaluación del Sistema RPE-LTP..... 113
Capítulo 6 Conclusiones y Perspectivas. 123
APENDICE. 131
GLOSARIO DE TERMINOS TECNICOS. 139
BIBLIOGRAFIA. 143

Introducción.

Dentro del contexto de la estandarización y de los futuros sistemas de radio móvil digital Pan-Europeo, el grupo especial móvil (GSM) CEPT desarrolló hace algunos años pruebas de codificadores subjetivos usando diferentes propuestas. Con esta contribución, *el Codificador-Decodificador de Voz(Vocoder)* describe un promedio mejor en la producción de voz de calidad. Este vocoder fué diseñado originalmente por el sistema de telefonía móvil digital experimental MATS-D[1] .

El esquema básico se extrajo primeramente a partir del conocido codificador RELP en banda base (RELP=Residual Pulse Excited Linear Prediction), combinado con la técnica Multi-Pulse Excitation-LPC(MPE-LPC).

La ventaja que presenta el codificador RELP en banda base es una complejidad relativamente baja mientras que la calidad de voz del mismo es limitada debido al tono de ruido la cual se introduce por el proceso de regeneración de frecuencias altas.

La técnica MPE-LPC, en contraste con la técnica anterior produce una excelente calidad de voz pero la complejidad es considerablemente alta.

En base a las consideraciones anteriores, se optó por equilibrar ambas técnicas surgiendo así una nueva llamada RPE-LPC. Para esta técnica existe su versión simplificada(RPE-LTP) la cual se implementa en este trabajo.

[1] K. Hellwig, R. Hofmann, R.J. Sluyter and P. Vary, "Mats-D Speech Codec: Regular-Pulse Excitation LPC", Second Nordic Seminar on Digital Land Mobile Radio Communication, 14-16 October, Stockolm, pp 257-261 (1986).

Justificación.

Desde hace ya varios años hasta hoy en nuestros días se sigue desarrollando enormemente la tecnología digital tanto como la tecnología de gran escala de integración (LSI, Large Scale Integration); por lo que debido a ello se ha puesto mucho énfasis en el desarrollo e implementación de métodos cada vez más eficientes para la codificación de voz en forma digital y para su transmisión.

Datos estadísticos revelan que típicamente, el costo de codificación de voz es mayor comparado con la complejidad del *codificador;* y a su vez, la complejidad es mayor comparada con la eficiencia del codificador y la utilización del canal.

Hasta ahora los codificadores digitales complejos(potencia-eficiencia) se han desechado debido a su alto costo.

En este trabajo se pretende la recuperación de la señal codificada obteniendo voz decodificada de ***buena calidad.*** Aunado a lo anterior y de manera implícita un *costo* mínimo requerido. Sabemos que esta meta fundamental no es nueva pero mediante la implementación de codificación de voz RPE-LTP se hacen posibles los dos objetivos primordiales haciendo uso de una de las técnicas con mayores ventajas respecto a las mencionadas en los siguientes apartados.

Objetivo.

Lograr la implementación de un sistema decodificador de voz como parte complementaria del codificador haciendo uso de la técnica RPE-LTP. El sistema contará con una reducción de la tasa de transmisión a 16 Kbit/s con características que le permitan:

Ser compatible con los sistemas de transmisión de la telefonía comercial.

Almacenar mensajes hablados y,

No degradar la calidad subjetiva de la señal de voz.

Un objetivo a tratar en el futuro es evaluar el comportamiento del sistema propuesto en tiempo real utilizando la tarjeta TAC-31C que cuenta con el procesador de señales TMS320C31 y paralelamente realizar la transmisión y recepción de voz entre dos computadores conteniendo en la primera el programa de codificación y en la segunda el programa decodificador cargados ambos en la memoria intermedia de la tarjeta TAC-31C.

Contenido de la Tesis.

En el capítulo 1 se menciona la descripción del oído desde el punto de vista de procesamiento de señales, se dan también algunas de las propiedades del oído, se habla un poco acerca del comportamiento de la membrana basilar y las descargas eléctricas del nervio auditivo. También se mencionan los modelos auditivos más importantes. Finalmente concluimos el capítulo con los comentarios sobre algunos efectos de percepción.

En el capítulo 2 se revisa de manera breve algunas técnicas utilizadas desde hace ya varios años para resolver el problema de la transmisión digital de la voz con el mínimo de información, pero cuidando un aspecto muy importante y este es la calidad de la señal, es decir, la inteligibilidad e identificación del interlocutor. De las investigaciones, desarrollos e

implementaciones de dichas técnicas mencionadas han aparecido diversas teorías de interés, no en la codificación de la forma de onda de la señal de voz analizada sino en la codificación de la fuente que la genera. Si bien, este es un concepto nuevo que revoluciona completamente la eficiencia en la codificación de la voz y como consecuencia origina nuevos métodos que van de acuerdo a las necesidades de transmisión, veremos en este capítulo que si se desea transmitir a muy bajas velocidades se deberá utilizar un vocoder de fuente sacrificando calidad de la señal de voz.

En contraparte, si lo que deseamos es una muy buena calidad podemos hacer uso de un vocoder de forma de onda con altas tasas de transmisión y baja complejidad hasta utilizar algún vocoder híbrido en la que utilizaremos una velocidad media y complejidad relativamente alta. Algunos de estos métodos se describen en este capítulo, el resto de las técnicas se estudiarán[5] . En el capítulo 3 se menciona ampliamente los fundamentos de la teoría de la técnica de Predicción Lineal. En el capítulo 4 se implementa el sintetizador RPE-LTP y al final se da el pseudocódigo del programa principal y todas las funciones utilizadas en el proceso de decodificación de la señal de voz; en el capítulo 5 se evalúa el vocoder completo utilizando pruebas subjetivas y en el capítulo 6 aparecen las conclusiones y perspectivas.

Finalmente, se proporciona un apéndice conteniendo el análisis de la cabecera de un archivo de sonido de extensión .WAV y el pseudocódigo de las dos funciones utilizadas para la extracción y colocación de cabecera en el proceso de Codificación(primera parte) y Decodificación (segunda parte) del proyecto global. Para finalizar el proyecto se muestran todas las fuentes bibliográficas que hicieron posible el contenido del documento.

[5] V. C. Alberto, Tesis, "Codificador de voz a 16 Kbit/s Usando la Técnica RPE-LTP". Universidad Tecnológica de la Mixteca., 1997.

Capítulo 1

ANTECEDENTES GENERALES.

1.1. Introducción.

Existe una gran variedad de vibraciones que pueden ser generadas por ondas mecánicas longitudinales. Particularmente las ondas sonoras, estas se encuentran restringidas a ciertos límites capaces de estimular el oído y el cerebro humano dando la sensación de sonido, lo que ocurre entre 20 Hz. y 20 KHz. aproximadamente y que además constituyen los límites audibles. Fuera de este rango, por encima o debajo de el se encuentran las ondas ultrasónicas e infrasónicas respectivamente en la que las vibraciones dejan de ser percibidas como sonido por el oído.

En el área de procesamiento de señales específicamente la *codificación de voz,* involucra tanto al analizador como al sintetizador para completar todo el proceso de codificación, es decir, Analizador(Codificador) y Sintetizador (Decodificador), teniéndose presente que la señal codificada puede enviarse mediante los medios de transmisión tales como vía cable, vía radio(espacio), limitando esta señal en un ancho de banda específico permisible.

Nuestro sistema a implementar corresponde al Sintetizador(Decodificador) para lo cual iniciaremos describiendo el aparato auditivo y algunas de sus propiedades auditivas desde el punto de vista de procesado de señales[2] .

1.2 Descripción del Aparato Auditivo.

En este apartado describimos el oído humano representado en la fig.(1.1) mediante 3 esquemas con los cuales nos ayudaremos para tal propósito.

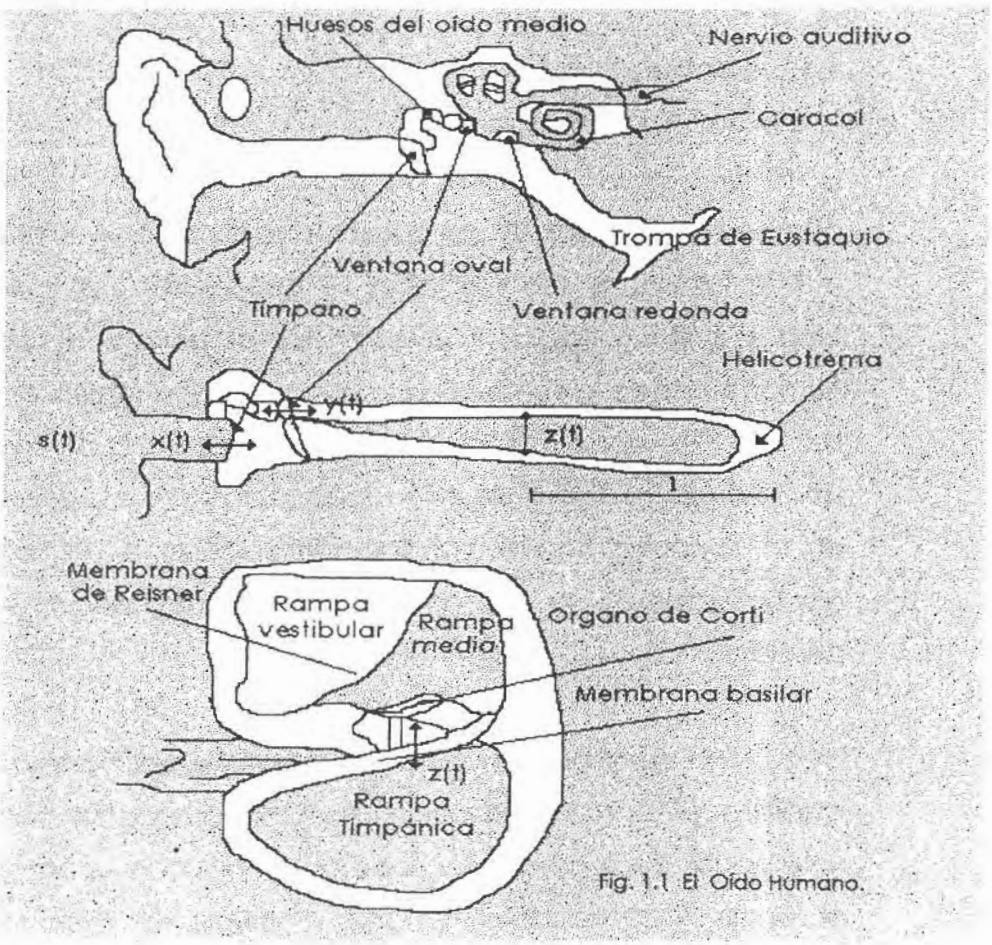


Fig. 1.1 El Oído Humano.

[2] R. García Gómez, "Tratamiento Numérico de la voz", Departamento de Señales, Sistemas y Radiocomunicaciones ETSI de Telecomunicación. Universidad Politécnica de Madrid. Madrid. Sep. 1991. Lección 2. pp. 1-11.

El esquema de la parte superior describe algunos detalles anatómicos, el del centro hace énfasis en la transformación que sufre la señal en su recorrido por el sistema auditivo y el inferior en su corte del caracol.

Sea $S(t)$ la señal que describe la presión acústica después de restarle el nivel medio de la presión atmosférica, estas variaciones provocan que la membrana timpánica tenga un desplazamiento en el sentido transversal, la amplitud de este desplazamiento es $X(t)$. Este movimiento es transmitido por la cadena de huesos del oído medio a la ventana oval cuyo desplazamiento es $Y(t)$. La ventana oval esta cerrada por una membrana flexible. Los desplazamientos de esta membrana provocan un desplazamiento de la linfa, una sustancia con propiedades de propagación del sonido parecidas a las del agua, que llena la cóclea o caracol.

En el centro de la fig.(1.1) tenemos representado el caracol desenroscado y en la parte inferior un corte transversal del mismo. Se observa que el caracol esta dividido longitudinalmente por dos membranas, la basilar y la de Reisner. Esta división no cierra el paso de la linfa por el extremo del caracol o helicotrema de tal manera que un desplazamiento de la membrana oval provoca un desplazamiento de la linfa y esta a su vez mueve la membrana que cierra la ventana redonda. Las olas de linfa provocan un movimiento de las membranas basilar y de Reisner. El movimiento relativo entre la membrana basilar y la tectorial, situada en el órgano de Corti, es detectado por las células nerviosas y transmitido al cerebro por el nervio auditivo. La señal que describe este movimiento relativo entre las membranas es $Z(t)$.

Si mediante un altavoz hacemos variar senoidalmente la presión ambiental $S(t)$, entonces observamos un desplazamiento también senoidal de la membrana basilar, es decir, de $X(t)$. Análogamente, tanto $Y(t)$ como $Z(t)$ son senoidales y de la misma frecuencia que $S(t)$. Este comportamiento es debido a que el sistema cuya entrada es $X(t)$ y salida $Z(t)$ es lineal e

invariante para niveles de sonido moderados. Por lo tanto, si variamos senoidalmente la presión acústica a la entrada del oído también el desplazamiento relativo entre la membrana basilar y la tectorial es senoidal y a la misma frecuencia.

Supongamos ahora que la presión acústica varía a la frecuencia de 1 KHz. A lo largo de la membrana basilar hay un punto que tiene una amplitud de desplazamiento mayor que todas las demás. Este punto esta situado a una distancia l del helicotrema. Si aumentamos la frecuencia del generador, el punto de máximo desplazamiento a lo largo de la membrana basilar esta a una distancia mayor del helicotrema. Si disminuimos la frecuencia, el punto de máximo desplazamiento se mueve hacia el helicotrema. A cada punto a lo largo de la membrana basilar le corresponde *una frecuencia característica*, aquella a la que su desplazamiento transversal es mayor que el de cualquier otro punto.

La correspondencia entre l y su frecuencia característica es aproximadamente logarítmica.

En la fig.(1.2) se representa el valor absoluto de la función de transferencia desde la entrada al caracol, es decir, considerando como entrada $Y(t)$, hasta el punto situado a una distancia l de la ventana oval, además podemos observar que no es simétrica con relación a la frecuencia de sintonía y la pendiente es mayor a frecuencias superiores que a frecuencias inferiores.

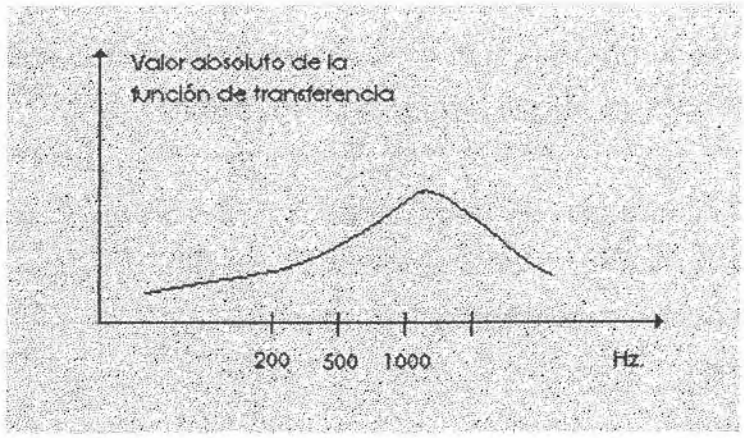


Fig. 1.2. Función de transferencia típica desde la entrada de la ventana oval a un punto situado a una distancia l de ella.

1.3 Propiedades Auditivas.

Oído externo: Esta constituído por la oreja y el canal auditivo. La oreja realiza un filtrado desde el punto de vista de transformación de señales. La función de transferencia de este filtro depende de la dirección de procedencia de la onda acústica. Un ejemplo interesante es la manera de localizar la procedencia de un sonido en el plano de simetría humano. Si la onda acústica viene en alguna dirección situada en este plano y esta llega a los dos oídos simultáneamente, la onda sufre un filtrado diferente dependiendo de la dirección de procedencia ya que nuestro cerebro decodifica el filtrado asignando una sensación de procedencia.

El canal auditivo es un tubo cerrado por un extremo mediante la membrana timpánica y abierto por el otro. La función de transferencia, considerando la entrada $S(t)$ y salida $X(t)$, enfatiza las frecuencias unos 12 dB entre los 3 a los 5 KHz. Un cambio de esta función de transferencia es detectado por

nuestro cerebro. Así cuando utilizamos auriculares, lo que implica cerrar este tubo acústico por el extremo abierto, la función de transferencia cambia significativamente. La nueva función de transferencia induce la sensación de que el sonido se genera en el interior del cerebro por lo que este efecto puede compensarse mediante técnicas de igualación.

Oído medio: Comienza en la membrana timpánica y finaliza en la membrana de la ventana oval. Estas dos membranas estan conectadas mediante una cadena de huesillos llamados:

* El martillo.
* El yunque y,
* El estribo.

El oído medio realiza tres funciones básicas.

La primera tiene que ver con la adaptación de impedancias, esta función es necesaria ya que en el oído externo el medio de propagación es el aire mientras que en el oído interno el medio es acuoso.

La segunda función tiene que ver con un filtrado pasa bajas cuya función de transferencia es aproximadamente plana hasta 1 KHz y después cae unos 15 dB/octava.

La tercera función es el reflejo acústico que protege el oído interno, actúa para señales con energía por debajo de unos 2 KHz y después de unos 60 a 120 ms. Este reflejo no protege frente a ruidos impulsivos o de alta frecuencia, además, actúa como un control automático de ganancia.

Oído interno: Transforma desplazamientos mecánicos de la membrana basilar en descargas eléctricas en el nervio auditivo.

El caracol da dos vueltas y media, desenrollado tiene una longitud de unos 35 mm. Las rampas vestibular y timpánica se comunican a través del helicotrema. El movimiento relativo de la membrana basilar y tectorial pueden producirse bien por el movimiento transmitido por la cadena de huesillos o bien por las vibraciones mecánicas de la cóclea transmitidas a través del sistema óseo. Este movimiento relativo es detectado por las células sensoriales. Cada fibra del nervio auditivo termina en unas 20 células sensoriales. El total de células sensoriales son unas 30,000 repartidas uniformemente.

1.4. Comportamiento de la Membrana Basilar.

Tal como hemos indicado en el apartado anterior, para una excitación senoidal a una frecuencia f_1, la membrana basilar tiene un desplazamiento máximo a una distancia l_1, para otra frecuencia f_2 el desplazamiento máximo es una distancia l_2. A cada punto a lo largo de la membrana basilar le corresponde una *frecuencia característica.*

Por lo tanto, a lo largo de la membrana basilar podemos establecer una escala en mm. o en Hz. Desde la ventana oval al punto situado a una distancia l del helicotrema podemos caracterizar el comportamiento mediante una función de transferencia tal como lo hemos representado en la fig.(1.2). Una particularidad de estos filtros es que tienen un factor de calidad Q sensiblemente constante. Es decir, la relación entre la frecuencia de resonancia y el ancho de banda es aproximadamente constante. Esto implica que los filtros que corresponden a frecuencias características bajas tienen anchos de banda menores que los de las altas. En otras palabras, el oído humano tiene mayor resolución frecuencial en bajas frecuencias que en altas frecuencias.

En la fig.(1.3) representamos la función de transferencia que corresponde a cuatro puntos diferentes de la membrana basilar. Los filtros sintonizados a frecuencias por debajo de los 1000 Hz tienen un ancho de banda de unos 100 Hz. Los sintonizados por encima de esta frecuencia tienen anchos de banda que crecen linealmente con la frecuencia. Estos anchos de banda se conocen como *"bandas críticas"*.

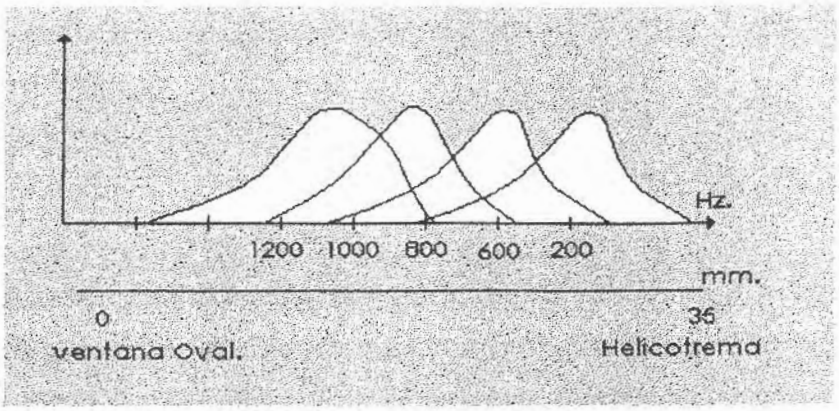


Fig. 1.3 Funciones de transferencia desde la ventana oval a distintos puntos de la membrana basilar. A cada punto de la membrana le corresponde una frecuencia caracteristica.

Otro aspecto interesante a observar es el desplazamiento a lo largo de la membrana basilar cuando la excitamos senoidalmente. En la fig.(1.4) representamos el desplazamiento transversal en dos instantes de tiempo determinados. El comportamiento es análogo al de una onda propagándose por un medio no homogéneo.

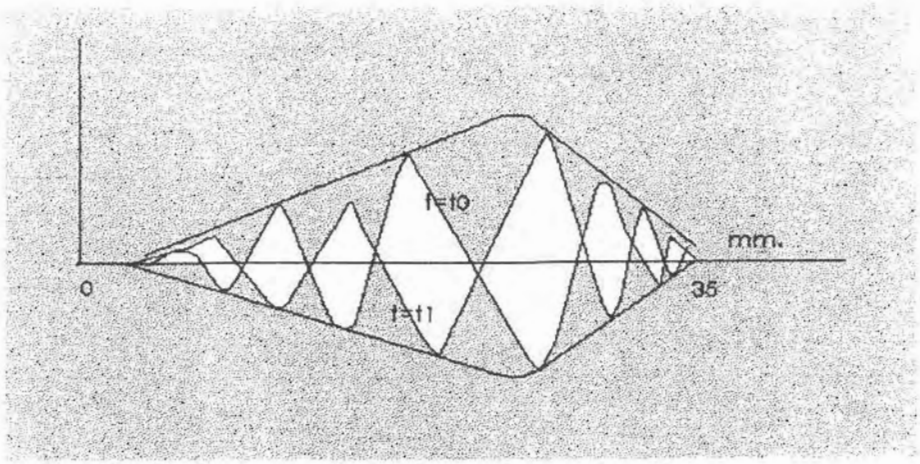


Fig.1.4 Desplazamiento transversal de la membrana basilar cuando la excitamos senoidalmente.

1.5. Descargas Eléctricas en el Nervio Auditivo.

Cuando no hay estímulos auditivos, las fibras nerviosas sufren descargas aleatorias espontáneas del orden de unas 10 a 50 por segundo. Cada descarga eléctrica es un impulso de una duración comprendida entre 0.5 y 1 ms.

El valor medio de pulsos en las distintas fibras nerviosas cambia en función del desplazamiento de la membrana basilar. Ante estos valores, es posible dibujar curvas de sintonía para cada terminación nerviosa. En la fig.(1.5) tenemos varias curvas de sintonía. Para sonidos senoidales, después de alcanzar un régimen permanente, se estableció el nivel de potencia a partir del cual el valor medio de las descargas supera el correspondiente de la actividad espontánea, estos niveles son los representados en la fig.(1.5). De nueva cuenta en estas curvas aparece claramente la selectividad asociada con cada posición a lo largo de la membrana basilar.

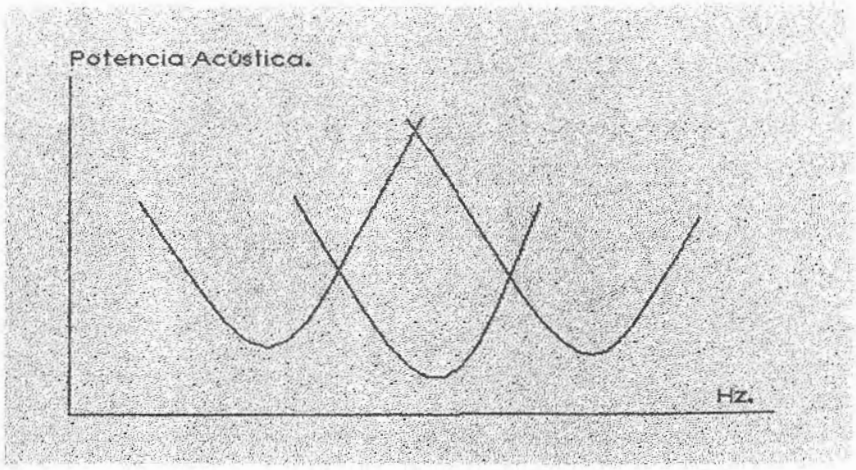


Fig. 1.5 Curvas de sintonía para tres terminaciones nerviosas.

Al aumentar la presión sonora por encima de un determinado nivel, estas curvas de sintonía tienden a ensancharse. Cuando hay excitación acústica, las descargas en las fibras nerviosas son más probables en unos determinados instantes que en otros. Los disparos estan sincronizados con los desplazamientos de la membrana basilar en uno de los dos sentidos posibles.

Consideremos una onda senoidal, dividamos cada periodo en un número fijo de intervalos. Si para cada intervalo temporal contamos el número de disparos que llegan al nervio acústico, con la suma de todas las terminaciones nerviosas obtendremos el *histograma de intervalo.* Este histograma tiene una forma equivalente a *la rectificación de media onda* de la señal periodica aplicada. En la fig.(1.6) representamos la onda periodica y el histograma de intervalo, que podemos interpretar como una función de probabilidad dependiente del tiempo.

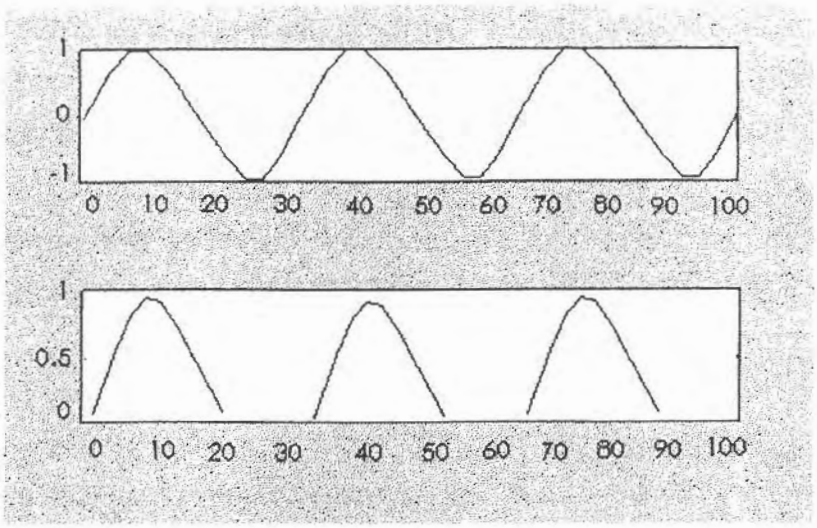


Fig. 1.6 Rectificación de media onda. Si la señal acústica varía senoidalmente, tal como representamos en la parte superior, el número medio de descargas en una fibra nerviosa esta representada en la gráfica inferior.

Cuando se alcanza el régimen permanente, a mayor intensidad de las senoidales, por encima del umbral indicado por las curvas de sintonía, la tasa media de disparo también se incrementa hasta que se alcanza el nivel de saturación. A medida que aumenta la intensidad por encima de este nivel los histogramas de intervalo tienden a tener un pico más pronunciado.

Cuando aplicamos un tono de suficiente amplitud, la tasa de disparos se incrementa inmediatamente. Después desciende en forma abrupta al comienzo(una constante de tiempo de unos 3 ms.) y más tarde un poco más lentamente(constante de tiempo de unos 40 ms.) hasta alcanzar el régimen permanente. Este régimen esta relacionado linealmente con el nivel del estímulo. Cuando suprimimos el tono, la tasa de disparos cae cerca del valor cero por debajo del nivel de disparo espontáneo y posteriormente se recupera hasta este nivel.

Este mecanismo sugiere que el cerebro puede interpretar tanto intensidades como cambios espectrales. La descripción simple que hemos dado permite definir algunos modelos computacionales. Estos modelos pretenden extraer

de la señal de voz la información equivalente a la que se envía a través del sistema nervioso al cerebro. Téngase en cuenta que la complejidad del sistema es elevada y que muchos mecanismos no se conocen en detalle, especialmente en la membrana basilar hacia el cerebro. Además la complejidad del modelo puede ser muy diferente dependiendo de los propósitos para los que ha sido diseñado.

1.6 Modelos Auditivos de la Voz.

Hay varias razones para interesarnos en el modelo del oído. Es interesante emular el proceso auditivo ya que este es tremendamente robusto. Somos capaces de entender a una persona aunque el ambiente en que se hable sea muy ruidoso, parece sensato intentar medir en la señal de voz aquellas características que mide el oído. Desde este punto de vista el modelo auditivo nos guiará en la extracción de parámetros para el reconocimiento de la voz.

La codificación de voz implica de una u otra forma la aproximación de la señal. Estamos interesados en aproximar aquellas características que sean perceptibles y descartar las otras. Un modelo auditivo podría ayudar a este objetivo.

Los distintos modelos difieren en el nivel de detalle pero suelen tener en cuenta el efecto de filtrado hasta el nivel de la membrana basilar, la transducción mecánica a eléctrica y algún tipo de procesado a niveles superiores. Este último aspecto es el menos conocido con diferencia.

Desde la fuente de sonido hasta un determinado punto situado a lo largo de la membrana basilar podemos emular las transformaciones que sufre la señal mediante un filtro lineal e invariante. Por razones computacionales no suelen modelarse las señales en muchos puntos de esta membrana, muchos

modelos incluyen entre 20 y 40. Cada uno de estos filtros es de tipo pasa-banda con su correspondiente función de transferencia. Cada filtro tendrá cierto solape frecuencial con los otros, de modo que cualquier señal, pueda producir algún efecto en al menos uno de los filtros. Los filtros centrados en bajas frecuencias se suelen diseñar con anchos de banda inferiores, para tener en cuenta este aspecto selectivo de la membrana basilar.

La relación existente entre el movimiento de la membrana basilar y el disparo de las neuronas es no lineal y bastante complicado. Los modelos suelen tener en cuenta la rectificación de la señal que hace el oído y un determinado control automático de ganancia para modelar la adaptación.

El modelo incluirá a la salida de cada uno de los filtros un rectificador de media onda y un control automático de ganancia. La salida de este, controlará la probabilidad de disparo de la fibra nerviosa. Esta probabilidad de disparo suele depender también de los disparos anteriores para tener en cuenta el tiempo de latencia neuronales y la adaptación a sonidos fuertes. Seleccionando adecuadamente los parámetros para este tipo de modelos, es posible simular los disparos que se observan en el nervio acústico.

Hay indicios de que los niveles superiores del aparato auditivo utilizan información relativa a las coincidencias de disparo entre diferentes neuronas. Por lo que algunos modelos incluyen algún procesado de la información que tenga en cuenta que cuando hay un pico espectral fuerte, varias fibras nerviosas cercanas a la posición característica que corresponde al pico se disparan síncronamente.

Entre los diferentes modelos auditivos que se han desarrollado se encuentran los siguientes:

a) Modelo Seneff.

En este modelo cada uno de los canales tiene una estructura de la forma:

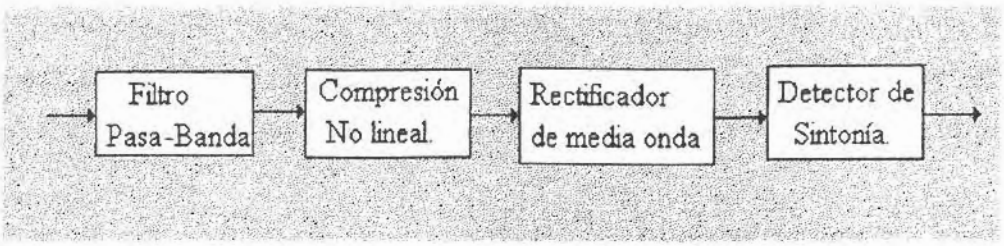


Fig. 1.7 Modelo de Seneff

El compresor no lineal consiste en la conexión en serie de dos controles automáticos de ganancia. Ambos idénticos, excepto su constante de tiempo. Si $X[n]$ es la señal de entrada a uno de ellos y $y[n]$ su señal de salida, entonces la relación entre ellos esta dada por :

$$y[n] = \frac{x[n]}{k + \langle |x[n]| \rangle_{\tau}}$$

τ : Constante de tiempo del integrador.

$\langle\ \rangle$: Símbolo del integrador.

x[n] : Secuencia de entrada.

k : Una constante.

b) Modelo de Ghitza.

El modelo que se muestra a continuación ha sido aplicado tanto a codificación como a reconocimiento de voz, dando buenos resultados en ambas aplicaciones.

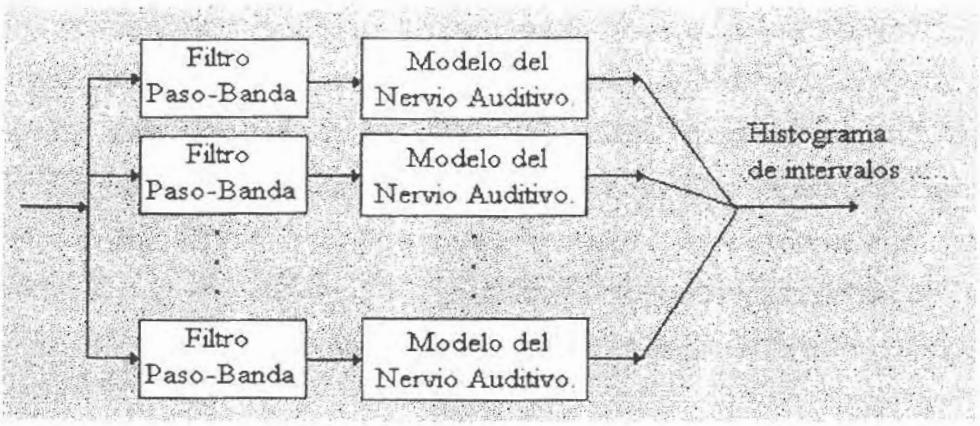


Fig. 1.8 Modelo de Ghitza.

El modelo esta representado en la fig.(1.8). La señal de voz $S[n]$ se procesa por un banco de 85 filtros pasa-banda. Estos filtros modelan el desplazamiento de la membrana basilar como respuesta a la señal logarítmica en todo el márgen dinámico de la señal. La utilización de niveles de cruce por umbrales positivos es equivalente a la rectificación de media onda. La salida de los detectores de cruce por los umbrales representa la actividad de disparo en el nervio auditivo.

Supone el modelo que el sistema nervioso procesa la información sobre los instantes de disparo de las fibras sin tener en cuenta su situación a lo largo de la membrana basilar. A esta teoría se le conoce como *Teoría frecuencial o no localista.*

1.7 Algunos Efectos Perceptuales.

A continuación describiremos algunos resultados perceptuales. En algunos casos indicaremos también las teorías que se han propuesto para explicarlos.

Sonoridad: La sonoridad percibida depende tanto del nivel de la señal como de su frecuencia. Comparando subjetivamente tonos de amplitud y

frecuencias diferentes se obtienen curvas de igual sonoridad, estas curvas se conocen como de Fletcher-Munson. La unidad de sonoridad es el *Fon*.
Los psicólogos también utilizan el nivel de sonoridad. Estas curvas se obtienen haciendo que un sujeto varíe la amplitud de un sonido de tal manera que su nivel sea la mitad del otro, el doble, una décima parte, etc. La escala de nivel de sonoridad es el *son*.
La relación entre el *son* y el *fon* se puede aproximar como:

$$N(son) = 0.063x_{10}^{0.03L(fon)}$$

Tono Fundamental. La unidad de periodo fundamental es el *mel*. El tono en *mels* se obtiene comparando subjetivamente dos tonos, unos en múltiplos del otro. La relación entre mel y frecuencia es:

$$Y(mels) = \frac{1000}{\log(2)}\log(1+\frac{f(Hz)}{1000})$$

Esta curva es aproximadamente lineal hasta 1 KHz y logarítmica por encima de esta frecuencia.

Enmascaramiento: Cuando un sonido evita la percepción de otros decimos que lo enmascara. El enmascaramiento depende del nivel relativo de ambos sonidos y de sus componentes frecuenciales. El enmascaramiento de un tono por otro es más efectivo cuando sus frecuencias son próximas.

También se ha estudiado el enmascaramiento de un tono senoidal por ruido blanco. Fletcher y Munson encontraron que cuando un tono se pretende enmascarar por un ruido con espectro plano y banda ancha centrada a la frecuencia del tono, el enmascaramiento es efectivo si la banda tiene un ancho superior a uno que denominan banda crítica. Las bandas críticas tienen una ancho de banda casi constante en la escala *mel*. El ancho de una banda crítica esta comprendido entre unos 100 *mels* a 50 *mels*, hasta

aproximadamente 250 *mels* a 3600 *mels*. El ancho de banda crítica esta relacionado con un buen número de efectos perceptuales, también se utiliza este concepto en algunos métodos de extracción de características para el reconocimiento de voz.

Percepción de la Voz. Aunque los mecanismos de percepción no estan claros y parece que falta mucho por aclararlos, algunos hechos son ampliamente aceptados.

Parece que nuestro cerebro hace una clasificación de los sonidos en voz y no voz. Cuando escuchamos sonidos sintéticos que gradualmente se hacen más semejantes a voces humanas hay un umbral a partir del cual la percepción es de voz. Parece que el cerebro procesa la voz de forma diferente a como lo hace con otros sonidos, incluso parece que la importancia de los dos hemisferios del cerebro juegan un papel diferente, el izquierdo es muy importante para la percepción de la voz.

El cerebro tiende a imponer una categorización de los sonidos que clasifica en *voz / no voz*. Estos sonidos vocálicos se clasifican a su vez (sonoros o fricativos).

La clasificación tiene que ver con umbrales de nuevo.

No hay una transición gradual, hasta cierto punto un sonido se decodifica. Por ejemplo, como /ba/ y si continuamos variando el parámetro adecuado cambiaríamos a /ga/.

Filtrado: Los resultados experimentales de filtrar la señal de voz mediante filtros pasa-bajas y pasa altas son los siguientes. Cuando utilizamos un filtro pasa-altas la *inteligibilidad*(porcentaje de sílabas correctamente reconocidas en una secuencia sin sentido) decrece a medida que la frecuencia de corte aumenta, la inteligibilidad no se deteriora para frecuencias de

corte inferiores a 400 Hz, en cambio, con una frecuencia de corte a 1700 Hz, la inteligibilidad se reduce al 50% y frecuencias de corte superiores a 6 KHz, la secuencia de sílabas se vuelve ininteligible.

Cuando el filtro es pasa-bajas, la inteligibilidad no se ve afectada para frecuencias de corte superiores a 6 KHz, el porcentaje de sílabas reconocidas es del 50% cuando la frecuencia de corte es de 1.5 KHz y a 400 Hz la voz se vuelve inteligible.

Recortes de la Señal. Consideremos la transformación:

$$y[n] = signo\{s[n]\}$$

donde $S[n]$ es la señal de voz. Observe que el efecto de esta transformación es el de obtener una señal con amplitudes +1 y -1 dependiendo de si $S[n]$ es positiva o negativa, es decir, se preservan los cruces por cero. Sorprendentemente esta transformación mantiene la inteligibilidad. Si los recortes se hacen con relación a un nivel diferente del de cero la inteligibilidad disminuye. También disminuye la inteligibilidad si los recortes son centrales, esto es:

$$y[n] = \begin{cases} s[n] - \mu & si \quad s[n] \geq \mu \\ 0 & -\mu \leq |s[n]| \leq \mu \\ s[n] + \mu & si \quad s[n] \leq \mu \end{cases}$$

Capítulo 2

TECNICAS DE CODIFICACION DE FORMAS DE ONDA.

2.1 Introducción.

Inicialmente, desde el año de 1982 se ha trabajado arduamente en estandarizar los sistemas y equipos de comunicación en Europa. Para tal fin se creó un grupo de trabajo, The European Telecommunications Administration CEPT-CCH-GSM(Groupe Speciale Mobile). La tarea asignada al GSM fué estudiar y desarrollar un estándar para los futuros sistemas móviles europeos, compatibles y además ofreciendo grandes servicios atractivos, bajo costo y trabajando todos en un sistema común.

El sistema en conjunto Codificador-Decodificador de voz a 16 Kbit/s usando RPE-LTP, cumple con las siguientes normas de calidad que el GSM estableció en ese entonces[3].

---

[3] M. Decina, G. Modena, "CCITT Standards on Digital Speech Processing", IEEE Journal on Selected Areas in Communications, Vol. 6, No. 2, Feb. 1988, pp 227-233.

El sistema se basa en una transmisión digital de voz y datos.

Expande las funciones y servicios ofrecidos por la Red Digital de Servicios Integrados(ISDN) dentro del sector móvil.

No presenta cambios significativos de la red fija telefónica.

La calidad de voz percibida es igual o mejor que la calidad a 900 MHz de los sistemas analógicos existentes.

El grupo de trabajo GSM, de acuerdo a sus lineamientos definidos seleccionó un esquema básico de codificador en base al estudio y análisis de 6 codificadores candidatos seleccionados, estableciendo como puntos principales los requerimientos de diseño y metodología de prueba para comparar los algorítmos de codificación candidatos.

Para la evaluación de los 6 codificadores se tomó en consideración la calidad de la voz generada, el retardo de transmisión y aspectos de implementación.

Finalmente el esquema básico obtenido fue el Codificador RPE-LPC. El resultado de la optimización de este codificador fue la selección de un algoritmo simplificado con predicción de término largo(RPE-LTP). Este análisis determinó la elección de nuestro codificador a implementar.

Los codificadores estudiados fueron:

Codificadores en Sub-Bandas, aquí hay 4 variantes:

- 3 variantes de codificador en Sub-Bandas con bloque de PCM Adaptivo (SBC-APCM).
- 1 codificador en Sub-Bandas usando PCM adaptivo diferencial (ADPCM).

- 2 codificadores basados en LPC.
1 de ellos es un codec RPE-LPC y
1 codec con excitación multi-pulso con predicción de término largo (MPE-LTP).

Hay tablas[4] que describen los parámetros principales de los codec, velocidad promedio, propiedades de retardo, memoria requerida y complejidad computacional.

Para mayor referencia acerca de los codec de forma de onda como PCM, consultar[5].

Se tienen varios elementos básicos objetivos a considerar para los codificadores de voz:

Calidad de voz: Los nuevos sistemas de codificación de voz requieren una calidad promedio para voz telefónica que sea al menos tan buena como la lograda por la primera generación de sistemas analógicos a 900 MHz sobre el rango de condiciones de operación práctica. La tasa de bit media y baja de los codecs tendrá su ejecución optimizada para la voz, previéndose también que ciertos esquemas pudiesen tener problemas con otras señales encontradas en telefonía pública. Por lo tanto, se identifican los siguientes requerimientos:

* El algoritmo de codificación tendrá que ser robusto a variaciones de voz espectral y niveles. Se espera un rango ancho de señales espectrales debido a variaciones en los parlantes, micrófonos y efectos de transmisión de la red telefónica. Por lo tanto, el nivel de voz promedio en la red telefónica se conoce y se varia en un rango de 30 dB.

[4] Jon E. Naturg, "Pan-European Speech Coding Standard for Digital Mobile Radio", Speech Communication 7 (1988). pag. 113-123.
[5] V.C Alberto, Tésis, "Codificador de voz a 16 Kbit/s usando la Técnica RPE-LTP", Universidad Técnológica de la Mixteca, 1997.

* El codificador será robusto a ruido ambiente y señales de voz múltiples.

Tasa de bit: La tasa de muestreo es de 8 KHz Basado en un análisis preliminar de los requerimientos en conflicto para la eficiencia del espectro y calidad de voz, la tasa de bit es de 16 Kbit/s.

Transcodificación: El estándar de codificación de voz básico para el sistema GSM se definió como un transcodificador entre PCM uniforme de 13 bits(8 KHz de frecuencia de muestreo) y una tasa de transmisión de 16 Kbit/s para sistemas de radio.

En el tamaño de la red, el codificador de voz GSM habrá de proporcionar una transcodificación de la Ley A o μ de PCM* . La conversión entre PCM de 64 Kbit/s de acuerdo a CCITT G.711 y al PCM uniforme ya se ha definido completamente en las recomendaciones del CCITT G. 721. Esta conversión resultará en 13 bits con PCM uniforme en el caso de PCM con Ley A. En la estación móvil los fabricantes tendrán que elegir entre los codecs disponibles con la ley A/μ seguido por la conversión a PCM uniforme de 13 bits o usar un A/D-D/A lineal para proporcionar el formato de 13 bits directamente.

* La Ley A y la Ley μ se describen ampliamente en la tesis denominada "Codificador a 16 Kbit/s usando RPE-LTP". Referencia [5].

2.2 Codificación en el Dominio Frecuencial.

El PCM y el ADPCM son codificadores en el dominio del tiempo en los que la señal de voz se procesa en el dominio del tiempo como una única señal de banda total.

La familia de codificadores que opera en el dominio frecuencial se divide a su vez en dos sub-familias:

i) Una basada en la utilización de un banco de filtros.

ii) La otra en una transformada.

2.2.1 Codificación en Sub-Bandas(SBC)[6] [7].

Como el mecanismo auditivo realiza un análisis de Fourier local, la división en *sub-bandas* o la utilización de transformadas con significado frecuencial permite la codificación más precisa en aquellas bandas o frecuencias en las que el oído es más sensible. Por otra parte, permite una forma natural de realizar una conformación del ruido de acuerdo con el espectro de la señal ya que el ruido en cada una de las bandas no afecta a las demás. A continuación se describe un codificador en el dominio de la frecuencia en el cual la señal de voz se divide en *sub-bandas* y cada una se codifica por separado.

El codificador es capaz de digitalizar la voz a 16 Kbit/s con una calidad comparable a la de 64 Kbit/s de un codificador PCM. Para asegurar tal realización, se explota la naturaleza cuasi-periodica de la señal de voz y una característica del mecanismo auditivo conocido como *enmascara- miento de ruido.*

---

[6] R. E. Crochiere, "On the Design of Sub-Band Coders For Low Bit rate Speech Communication", Bell System Tech. J. Vol. 56, No. 5, May-Jun 1977.

[7] R. E. Crochiere, S. A. Webber, J. L. Flanagan, "Digital Coding of Speech in Sub-Bands", Bell System Tech. J. Vol. 55, No. 8, Oct. 1976.

La periodicidad de los sonidos de voz se manifiesta por si mismos en el hecho de que las personas hablan con una frecuencia de Pitch característico. Esta periodicidad permite la predicción del Pitch y por lo tanto una reducción adicional en el nivel de predicción de error que requiere cuantización. Esto se compara con el PCM diferencial pero sin la predicción del Pitch.

El número de bits por muestra necesarios para transmitirse se reduce enormemente pero sin causar mucha degradación en la calidad de voz.

El número de bits por muestra se puede reducir más al hacer uso del fenómeno de *enmascaramiento de ruido* en la percepción. Esto es, el oído humano no percibe el ruido en una banda de frecuencia determinada, si el ruido esta alrededor de 15 dB por abajo del nivel de la señal en esa banda. Esto significa que una codificación de error relativamente grande(lo equivalente al ruido) se puede tolerar cerca de los *formantes* y por consiguiente el porcentaje de codificación se puede reducir.

En el contexto de producción de voz, *las frecuencias de los formantes* (o simplemente formantes) son las frecuencias de resonancia del tubo del tracto vocal. Los formantes dependen de la forma y dimensiones del tracto vocal.

En el codificador SBC el ancho de banda de la señal de voz se divide en *sub-bandas* y se hacen pasar a través de filtros *pasa-bandas*. Cada sub-banda se traslada a una banda baja, después se muestrea a una nueva frecuencia de Nyquist y por último se codifica digitalmente.

En la fig.(2.1) se tiene el diagrama del codificador SBC.

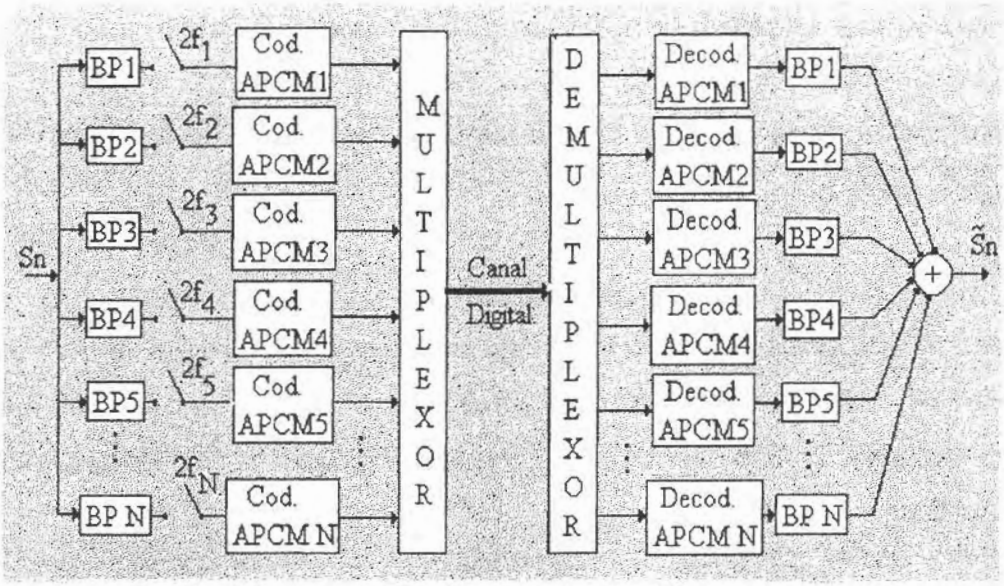


Fig. 2.1. Sistema de codificación en Sub-bandas (SBC)

De acuerdo con el mecanismo auditivo se desea que los filtros correspondientes a frecuencias bajas tengan un ancho de banda más pequeño que las centradas a frecuencias mayores. Esto suele complicar el diseño y muchas veces se utilizan anchos de bandas constantes.

Para disminuir el número de operaciones los bancos de filtros pueden diseñarse utilizando técnicas de *diezmado pasa-banda sin demodulación o* filtros *especulares en cuadratura* QMF. Los principios de funcionamiento de ambos bancos de filtros se describen a continuación.

Diezmado Pasa Banda sin Demodulación[2] .

Si se tiene una secuencia $X[n]$ considerada como pasa banda, la banda que ocupa esta en el intervalo $[(k+1)\frac{\pi}{M}, k(\frac{\pi}{M})]$. La fig.(2.2) representa este espectro.

[2] R. García Gómez, "Tratamiento Numérico de la voz", Departamento de Señales, Sistemas y Radiocomunicaciones ETSI de Telecomunicación. Universidad Politécnica de Madrid, Madrid. Sep. 1991. Lección 6. pp 9-14.

Tanto k como M se consideran como números enteros.

Si $Y[n]$ se obtiene mediante el diezmado de $X[n]$ por el factor M, la relación entre ambos espectros es:

$$Y(w) = \frac{1}{M}\sum_{l=0}^{M-1} X\left(\frac{w-2\pi l}{M}\right)$$

Para observar si este diezmado introduce solapamientos espectrales se dibujará a:

$$Z(w) = \sum_{l=0}^{M-1} X\left(w-\frac{2\pi}{M}\right)$$

Y se observa que:

$$Y(w) = \frac{1}{M} z\left(\frac{w}{M}\right)$$

Es decir, un cambio de escala en el eje independiente.

Si no hay solapes en $Z(W)$ tampoco lo habrá en $Y(W)$.

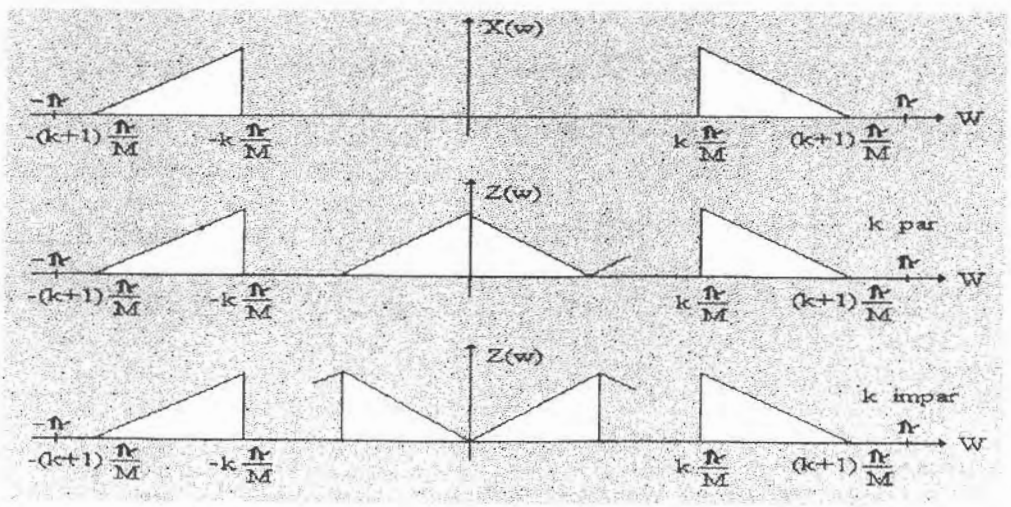


Fig. 2.2. Diezmado pasa-banda sin demodulación de X[n] por el factor M.

En la fig.(2.2) se observa que no hay solapes espectrales.

La única diferencia para K par o impar es que los espectros estan invertidos. Esta inversión puede compensarse simplemente multiplicando por $(1)^n$ a la señal interpolada.

Al no haber solapes espectrales se puede recuperar la señal intercalando $M-1$ ceros entre cada dos muestras realizando un filtrado pasa bandas. De esta forma se evita la modulación y demodulación del caso general.

El esquema de interpolación y diezmado para cada canal se representa en la fig.(2.3). El canal k-ésimo constará de filtros pasa bandas en el intervalo $[(k+1)\frac{\pi}{M}, k(\frac{\pi}{M})]$.

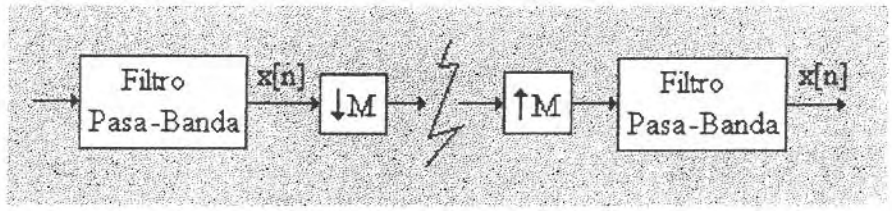


Fig. 2.3. Filtros especulares en cuadratura.

En la fig.(2.4) se considera un banco de dos ramas.

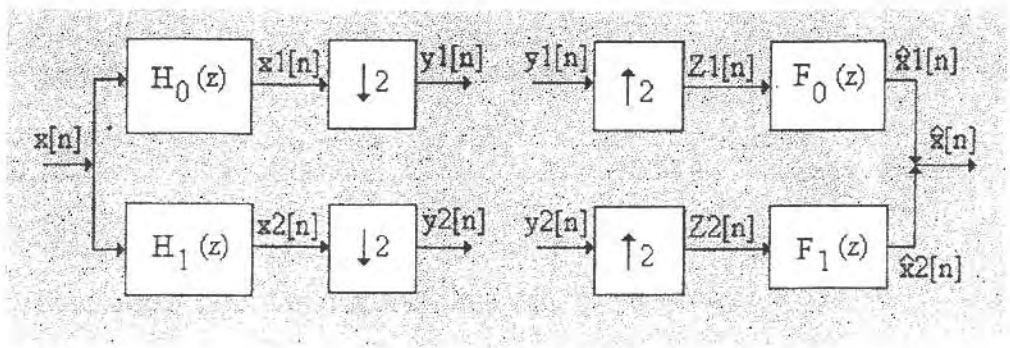
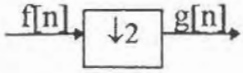


Fig.2.4. Banco de dos ramas.

En este esquema $H_0(z)$ y $F_0(z)$ son filtros pasa bajas, mientras que $H_1(z)$ y $F_1(z)$ son pasa altas. Para analizar el comportamiento de este banco, se tiene que las relaciones *entrada-salida* diezmado por 2 es:

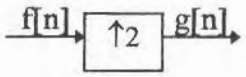


$$G(z) = 0.5F(z^{1/2}) + 0.5F(-z^{1/2})$$
$$G(jW) = 0.5F(z^{jW/2}) + 0.5F(-z^{jW/2})$$

$$G(z) = \frac{1}{2}F\left(z^{1/2}\right) + \frac{1}{2}\left(-z^{1/2}\right)$$

Si $f[n]$ es pasa bajas y tiene un ancho de banda inferior a $\pi/2$, entonces, no hay solapes espectrales. Lo mismo sucede si la señal es pasa altas y su ancho de banda es inferior a $\pi/2$. En ambos casos el espectro de la señal diezmada tiene el mismo aspecto.

Las relaciones de velocidad de un interpolador son:



$$G(z) = F(z^2)$$
$$G(e^{jW}) = F(e^{2jW})$$

En el banco de filtros que se analiza se tiene un diezmador seguido de un interpolador en cada uno de los canales(lo que normalmente es un reductor de velocidad por 2, seguido de un incrementador por el mismo factor).

En el banco de filtros considerado, se establece la condición de diseño:

$$X[n] = \hat{X}[n]$$

Si los filtros $H_0(z)$ y $H_1(z)$ se limitan en banda a $\pi/2$, uno pasa bajas y el otro pasa altas, el orden será muy elevado para obtener bandas de transición muy abruptas. En caso contrario se pierde información espectral,

las que eliminen las bandas de transición. Es por tanto, prácticamente irrealizable esta alternativa.

La alternativa en la que se basan los filtros QMF permite solapes espectrales en cada canal. Se observa que no se pretende recuperar la información canal a canal sino que se desea que la suma de ambas sea la señal original, tal como lo indica la condición de diseño anterior.

A continuación se tienen las condiciones de verificación de los filtros.

El esquema del banco tiene en cuenta las relaciones de diezmado e interpolación anteriores:

$$Y_1(z) = \frac{1}{2} X_1(z^{1/2}) + \frac{1}{2} X_1(-z^{1/2})$$

$$Y_1(z) = \frac{1}{2} X\ (z^{1/2}) H_0(z^{1/2}) + \frac{1}{2} X\ (-z^{1/2}) H_0(-z^{1/2})$$

Análogamente:

$$Y_2(z) = \frac{1}{2} X_2(z^{1/2}) + \frac{1}{2} X_2(-z^{1/2})$$

$$Y_2(z) = \frac{1}{2} X_2(z^{1/2}) H_1(z^{1/2}) + \frac{1}{2} X_2(-z^{1/2}) H_1(-z^{1/2})$$

Y tomando en cuenta que:

$$z_1(z) = Y_1(z^2)$$

$$z_2(z) = Y_2(z^2)$$

Tenemos:

$$X_1(z) = F_0(z)\{\frac{1}{2}X(z)H_0(z) + \frac{1}{2}X(-z)H_o(-z)\}$$

$$X_1(z) = F_1(z)\{\frac{1}{2}X(z)H_1(z) + \frac{1}{2}X(-z)H_1(-z)\}$$

Y la salida del banco de filtros será:

$$X(z) = \frac{1}{2}X(z)\{F_0(z)H_0(z) + F_1(z)H_1(-z)\} +$$

$$\frac{1}{2}X(-z)\{F_0(z)H_0(-z) + F_1(z)H_1(-z)\}$$

El segundo sumando, el que introduce solapes espectrales se puede anular haciendo que:

$$F_0(z) = H_1(-z)$$

$$F_1(z) = -H_0(-z)$$

Con lo cual se obtiene:

$$X(z) = \frac{1}{2}\{F_o(z)H_0(z) + F_1(z)H_1(-z)\}$$

y el banco de filtros es equivalente a un filtro lineal invariante de función de transferencia que es:

$$T(z) = \frac{1}{2}\{F_o(z)H_0(z) + F_1(z)H_1(-z)\}$$

$$T(z) = \frac{1}{2}\{H_1(-z)H_0(z) - H_0(-z)H_1(z)\}$$

Tomando a:

$$H_1(z) = H_0(-z)$$
$$H_1(e^{jW}) = H_0(e^{j(W-\pi)})$$

Al hacer $H_0(z)$ pasa bajas, $H_1(z)$ será pasa altas. Ambos filtros serán simétricos especularmente con relación a ρ/2 tal como se observa en la fig.(2.5), de aquí el nombre de QMB(Quadrature Mirror Filter Banks).

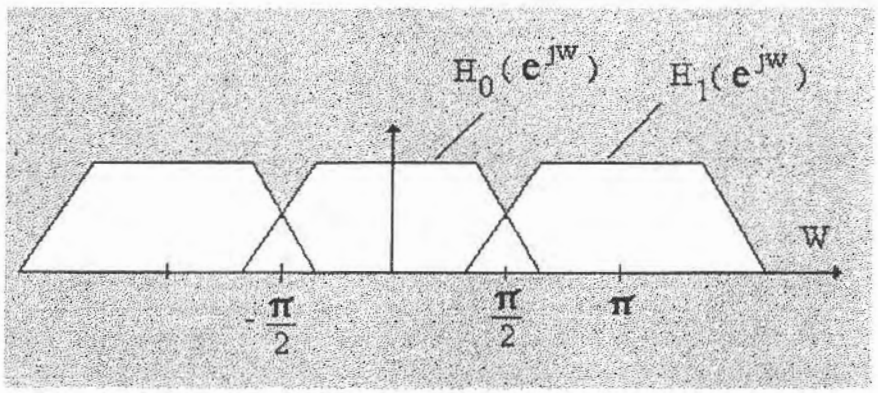


Fig.2.5 Simetría entre dos filtros especulares.

En la codificación adaptativa en sub-bandas(ASBC), la forma del ruido se acompaña de la asignación del bit adaptativo. En particular, el número de bits utilizados para codificar cada sub-banda se varía dinámicamente y se comparte con otras sub-bandas, tal que la exactitud de la codificación siempre se coloca en el dominio de la frecuencia donde es necesaria para caracterizar la señal. De hecho, las sub-bandas con poca o ninguna energía nó se pueden codificar.

Un diagrama a bloques de la Codificación Adaptativa en Sub-Bandas se muestra en la fig.(2.6). Específicamente, la banda de la señal de voz se divide en una cantidad determinada de bandas contiguas por un banco de filtros pasa bandas (normalmente de cuatro a ocho). La salida de cada

filtro pasa bandas se traslada en frecuencia para asumir una forma pasa bajas para un proceso equivalente a la modulación de banda lateral única. Después se muestrea(o se remuestrea) con una frecuencia ligeramente mayor a la de Nyquist(el doble del ancho de banda de cada sub-banda) y se codifica usando un ADPCM de predicción fija (normalmente de orden uno).

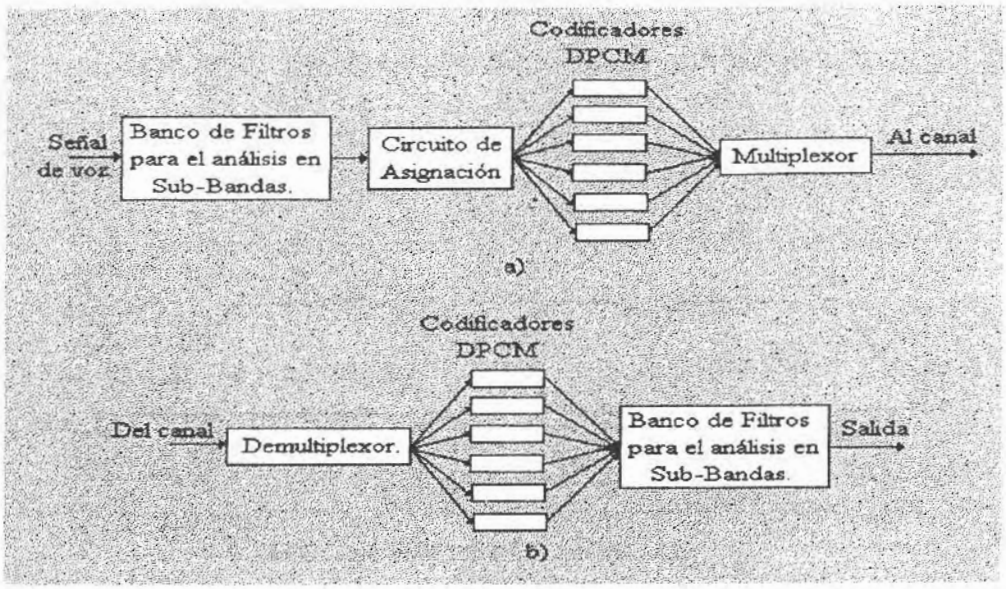


Fig. 2.6. Esquema de Codificación Adaptativo en sub-bandas
a) Transmisor, b) Receptor.

Se emplea una estrategia de codificación específica para cada sub-banda de acuerdo a un criterio perceptual asociado a esa banda. La asignación del bit de información se transmite al receptor, habilitándolo para decodificar en forma individual las señales en sub-bandas y modularlas para después regresarlas a sus posiciones originales en la banda de frecuencia. Finalmente, se asume que se produce una señal a la salida que proporciona una replica aproximada de la señal de voz original.

Si F_s denota la frecuencia de muestreo para la señal de entrada(ancho de banda) y N el número promedio de bits utilizado para codificar una muestra

de la señal, entonces, el porcentaje de bits correspondiente es NF_s bits por segundo.

Claramente, se puede escribir

$$N F_s = (MN)\left(\frac{F_s}{M}\right)$$

Donde M es el número de sub-bandas y suponiendo que tienen el mismo ancho de banda, entonces, la frecuencia de muestreo para cada una de las sub-bandas se reconoce que es F_s/M.

La implicación es un número total de MN bits por muestra para las M sub-bandas.

Para ilustrar el significado de este resultado considérese un esquema con M=4 sub-bandas de igual ancho de banda, con frecuencia de muestreo estándar de F_s=8 KHz para voz y N=2 bits por muestra. Entonces, la frecuencia de muestreo para cada sub-banda es de 2 KHz y el número total de bits por muestra para las cuatro sub-bandas es 8. Para cada segmento de voz para componentes predominantes de baja frecuencia, por ejemplo, se pueden utilizar los bits de asignación 5,2,1,0 (bits) para las cuatro sub-bandas en el incremento de la frecuencia. Por otra parte, para un segmento de voz con componentes predominantes de alta frecuencia, la asignación apropiada de bits puede ser de 1,1,3,3.

Así, en el esquema de codificación adaptativa en sub-bandas esta de acuerdo con el contenido espectral de la señal de voz de entrada, por tanto, ayuda a controlar la forma del espectro de cuantización de ruido como una función de frecuencia. Específicamente, se usan más niveles de representación(en promedio) para las bandas de frecuencia mas bajas donde se conserva la información del Pitch y el formante. Mas sin embargo,

si la energía de frecuencias altas es dominante en la señal de voz de entrada, el esquema automáticamente tendrá un gran número de niveles de representación para las componentes de frecuencias altas de la entrada. También, es notable que el ruido de cuantización que se permite dentro de cada sub-banda sea una entrada de voz de bajo nivel en un esquema de sub-banda y este no se puede ocultar con la cuantización del ruido producido en otra sub-banda.

La complejidad de un cuantificador adaptativo en sub-bandas de 16 Kbit/s es típicamente de cien veces más que la de un codificador PCM de 64 Kbit/s para producir aproximadamente la misma calidad subjetiva de voz. Sin embargo, como resultado de un gran número de operaciones aritméticas involucradas en diseñar el codificador adaptativo en sub-bandas, hay un retardo de procesamiento de 25 ms, por otra parte, tales retardos no se encuentran en el codificador PCM.

2.2.2 Codificación de Transformación Adaptativa (ATC)[2].

El esquema de principio de un codificador basado en la transmisión de las transformadas locales se representa a continuación:

[2] R. García Gómez, "Tratamiento Numérico de la voz", Departamento de Señales, Sistemas y Radiocomunicaciones ETSI de Telecomunicación. Universidad Politécnica de Madrid, Madrid, Sep. 1991. Lección 6, pp 15-17.

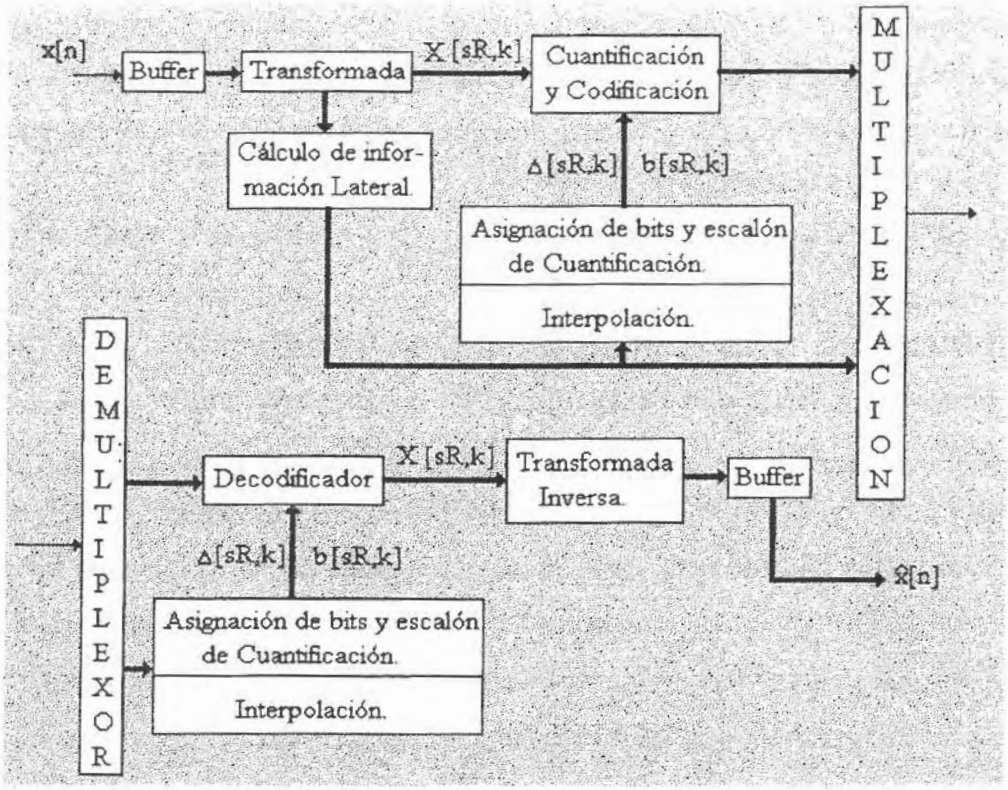


Fig. 2.7. Esquema de principio de un codificador de transformada.

Este codificador transmite la transformada de ventanas sucesivas de la señal. El desplazamiento de la ventana es de R muestras. Estos desplazamientos suelen corresponder a intervalos de 16 a 32 ms. Las ventanas sucesivas pueden estar un poco solapadas, este solape puede ser del orden de aproximadamente el 10% o menos.

Al calcular la transformada obtenemos una secuencia de M números $X[sR,k]$, $0 \leq k \leq M-1$. Cada número se codifica con $b[sR,k]$ bits, utilizando un cuantificador vectorial con un escalón $\Delta[sR,k]$.

Para seleccionar el escalón de cuantificación y el número de bits para cada componente de la transformada se utiliza la información de la envolvente espectral de la ventana que en ese momento se este cuantificando. Ya que sucesivas ventanas tienen un pequeño solape, sus transformadas pueden ser bastante diferentes. Por lo que no se pueden

utilizar esquemas de adaptación de los cuantificadores basándose en la cuantificación de las transformadas anteriores, es decir, la adaptación estará basada en la información lateral que se transmitirá explícitamente, tal como se indica en la figura anterior.

* Selección de la transformada.

Ya que el oído humano hace un análisis local de Fourier de la señal, parece natural elegir la transformada de Fourier para esta familia de codificadores, en su versión discreta la *DFT*. Si se considera la ventana de la señal *v[n]=x[n]w[n]*, la ventana *w[n]* normalmente se elige de forma trapezoidal.

La *DFT* de *v[sR,k]* viene dada por:

$$V[k] = \sum_{n=0}^{M-1} v[n] e^{-j(2\pi kn/M)} \qquad 0 \le k \le M-1$$

Si cada una de las $V[k]$ la cuantificamos con bastantes bits, entonces se modela la cuantificación mediante un ruido aditivo, por otra parte cuando la cuantificación se hace con pocos bits(incluso cuando algunos $V[k]$ pueden cuantificarse con cero bits, es decir, no se transmiten) este modelo de no cuantificador no es válido. El modelo adecuado incluye un término multiplicativo y otro aditivo.

$$\hat{V}[k] = V[k]Gv[k] + Ev[k]$$

Y después de la síntesis se tiene:

$$\hat{V}[n] = V[n] \otimes g_v[n] e_v[n]$$

Donde el símbolo ⊗ denota la *convolución circular*. Por lo tanto el resultado de la síntesis es la convolución circular de la señal original de la *DFT* inversa del factor multiplicativo más ruido aditivo.

Debido a esta convolución circular *el extremo derecho de la ventana de la señal afecta al extremo izquierdo y viceversa.* Tomando en cuenta que la convolución circular implica una repetición periodica de las señales. Este efecto produce ruidos molestos. Por esta razón suelen utilizarse alguna otra transformada con interpretación frecuencial. Particularmente la transformada del coseno, cuyas fórmulas directa e inversa son:

$$Vc[k] = \sum_{n=0}^{M-1} v[n]c[k]\cos\left(\frac{(2n+1)\pi k}{2M}\right)$$

$$v[n] = \frac{1}{M}\sum_{k=0}^{M-1} Vc[k]c[k]\cos\left(\frac{(2n+1)\pi k}{2M}\right)$$

$$c[k] = \begin{cases} 1 & \text{si } k = 0 \\ \sqrt{2} & \text{En otro caso} \end{cases} \qquad (2.1)$$

Esta transformada disminuye la influencia de un extremo de la ventana de la señal y en el otro mantiene una eficiencia alta ya que la transformada de M muestras de la señal es otro conjunto de M números. Para observar la razón se define a:

$$y[n] = \begin{cases} \frac{1}{2}v[n] & 0 \le n \le M-1 \\ \frac{1}{2}v[2M-n-1] & M \le n \le 2M-1 \end{cases}$$

Esta ecuación indica que la secuencia *y[n]* se construye a partir de *v[n]* tal como se representa en la figura siguiente.

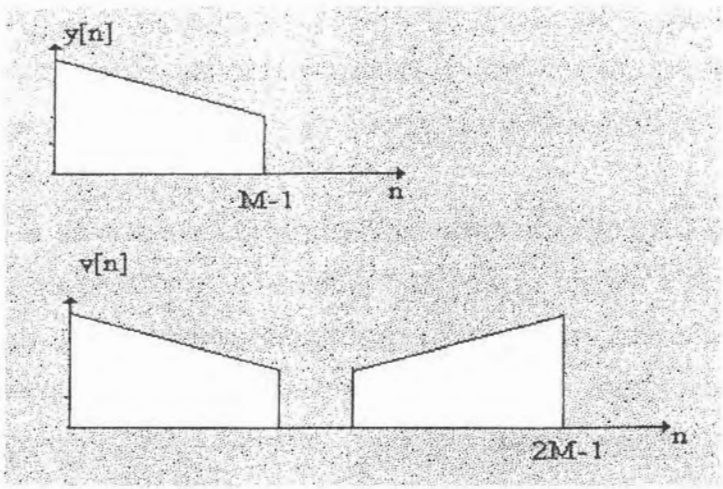


Fig. 2.8 Definición de Y[n] a partir de v[n].

La *DFT* de esta secuencia de *2M* muestras es:

$$Y[k] = \sum_{n=0}^{2M-1} y[n]\ e^{-j\frac{2\pi kn}{2M}} \qquad 0 \le k \le 2M-1$$

$$Y[n] = e^{j\frac{\pi k}{2M}} \sum_{n=0}^{2M-1} v[n] \cos\left(\frac{(2n+1)\pi k}{2M}\right) \qquad (2.2)$$

Comparando la ec.(2.2) con la ec.(2.1) se llega a la conclusión que la transformada del coseno se puede calcular a partir de la *DFT* de $Y[n]$.

$$V_c[k] = c[k] e^{j\frac{pk}{2M}} Y[k] \qquad (2.3)$$

Se observa que si cuantificamos $Y[k]$ se tendrán convoluciones circulares de *2M* puntos. Un extremo de la ventana de análisis no afecta al otro debido a la forma de construir $Y[n]$. La relación de la ec.(2.3) indica que la transformada del coseno debe verificar también esta propiedad.

2.3 Codificación por Predicción Adaptiva(APC).

La técnica APC mostrada en el diagrama a bloques de la fig.(2.9), estima o predice la muestra de entrada presente a partir de la historia pasada de la forma de onda, esto es:

$$\tilde{S}_n = P(S_{n-1}S_{n-2},\ldots)$$

La señal residual o señal de error $e_n = S_n - \tilde{S}_n$, originada por la función P, proporciona suficiente información al receptor para regenerar la entrada de una forma exacta. Sin embargo, la señal de error es sometida a un proceso de cuantización y por ello, no se envía en forma exacta, lo que ocasiona que la voz a la salida se distorsione. Por esta razón es importante la selección de un método de cuantización apropiado para lograr la mejor calidad.

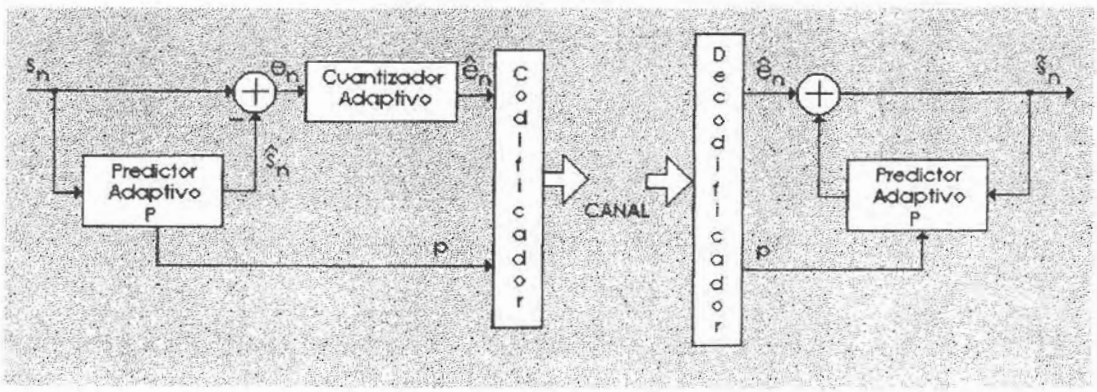


Fig. 2.9 Sistema APC simplificado.

Un cuantizador que opere sobre la señal residual, necesita un menor número de bits/muestra que un cuantizador que opere sobre la señal de entrada, lo cual se traduce en una menor tasa de información.

El predictor óptimo depende de las estadísticas de la señal y de aquí que, los parámetros de predicción necesitan adaptarse a los cambios de la señal. Tales parámetros se escogen para minimizar el error cuadrático medio e_n entre las muestras estimadas y las muestras reales, en un intervalo de análisis(o ventana) de longitud $T = 20 \text{ o } 25 \text{ ms}$ [8].

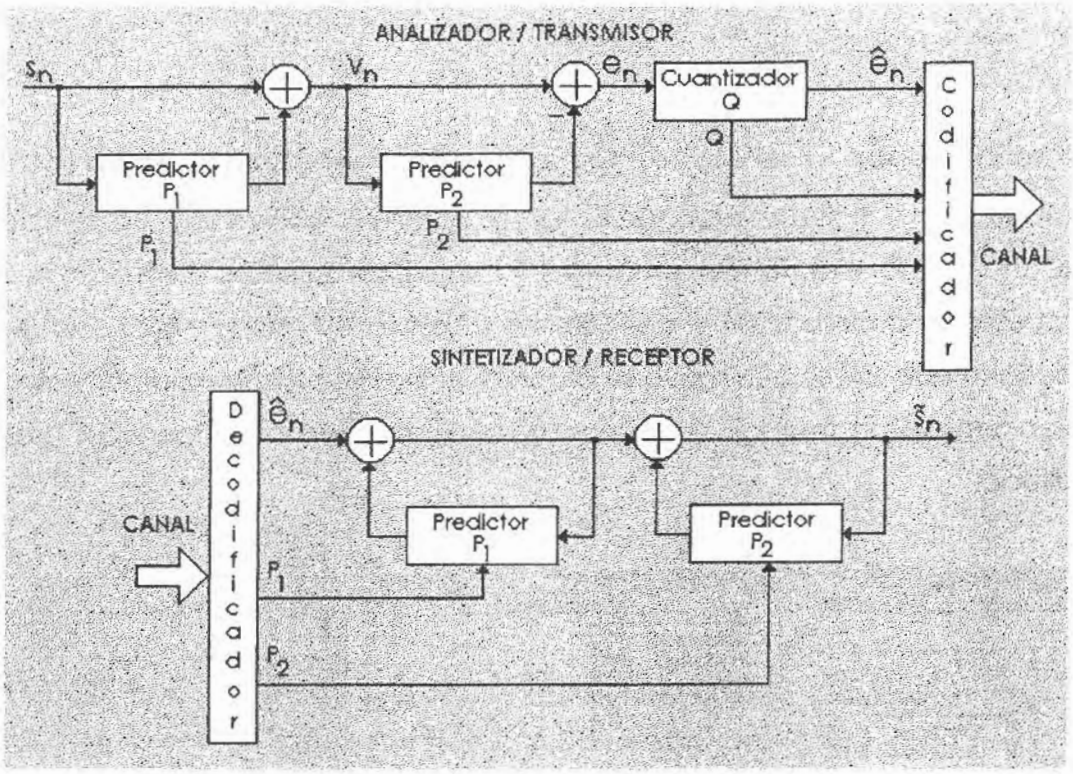


Fig. 2.10 Sistema APC para codificación de voz.

Una vez calculados los parámetros del predictor, se mantienen fijos durante todo el intervalo pero cambian de ventana a ventana, de acuerdo a las variaciones estadísticas de la entrada. Así durante el análisis de cada ventana, los algoritmos determinan los parámetros de predicción de los datos de entrada y forman la secuencia de la señal de error filtrando digitalmente la señal de entrada.

[8] A. J. Goldberg, H. L. Shaffer, "A Real-Time Adaptive Predictive Coder Using Small Computer", IEEE COM-23, No. 12, Dec. 1975, pp 1443-1451.

Para procesamiento de voz, la función predictor P se divide en dos predictores mas simples P_1 y P_2 como lo muestra la fig.(2.10). La forma de P_1 toma en cuenta que la voz es, a menudo, cuasi-periodica con periodo M y por tal razón estima la voz como:

$$\widetilde{S}_n = \alpha S_{n-M}$$

Donde la ganancia de tono fundamental α, indica que, o bien existe variación en la ganancia de ventana a ventana, o que la señal de voz no esta perfectamente correlacionada con el periodo M.

Para minimizar el error cuadrático total E

$$E = \sum_{n=1}^{T} (S_n - \alpha S_{n-M})^2$$

se diferencia respecto a α y se iguala a cero resultando:

$$\alpha = \sum_{n=1}^{T} S_n S_{n-M} / \sum_{n=1}^{T} S_{n-M}^2$$

Donde T es la longitud de la ventana en muestras.

Para sonidos periodicos como vocales, α es casi la unidad pero para ruido, como el producido por algunas consonantes, se tiene poca correlación con tan sólo algunas M muestras aparte y α es cercano a cero.

La forma reducida V_n es:

$$V_n = S_n - S_{n-M}$$

o primera señal de error, aún contiene bastante redundancia de modo tal que un segundo predictor P_2, de orden N, puede reducir la potencia de la

señal de salida, sobre todo si el segmento de voz es no periodico. Este segundo predictor utiliza una suma ponderada de N muestras pasadas de la señal de voz para calcular las muestras estimadas:

$$\tilde{V} = \sum_{i=1}^{N} a_i V_{n-i}$$

Donde las a_i se seleccionan para minimizar el error cuadrado U.

$$U = \sum_{n=1}^{T} \left(V_n - \tilde{V}_n\right)^2$$

y a su vez son la solución a la ecuación matricial $\phi a = c$ donde:

$$\phi_{ij} = \sum_{n=1}^{T} V_{n-i} V_{n-j}$$

$$C_i = \sum_{n=1}^{T} V_n V_{n-j}$$

Si consideramos la forma de onda reducida sólo dentro de una ventana, esto es, cero fuera de cierto intervalo $1 \le n \le T$, entonces se tienen ecuaciones normales de autocorrelación:

$$\phi_{ij} = \sum_{n=1}^{T-|(i-j)|} V_n V_{n-|(i-j)|} = R_{i-j}$$

De esta forma la ecuación matricial se convierte en:

$$\begin{bmatrix} R_0 & R_1 & R_2 & \cdots & R_{p-1} \\ R_1 & R_0 & R_1 & \cdots & R_{p-2} \\ R_2 & R_1 & R_0 & \cdots & R_{p-3} \\ \vdots & \vdots & \vdots & & \vdots \\ R_{p-1} & R_{p-2} & R_{p-3} & \cdots & R_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ \vdots \\ R_p \end{bmatrix}$$

Una matriz simétrica de Toeplitz, con elementos similares a lo largo de cualquier diagonal paralela a la diagonal principal.

Diversas soluciones para calcular los coeficientes de predicción (a_i) y la energía cuadrada media U, emplean las propiedades de la matriz de Toeplitz generando algoritmos muy eficientes.

Desafortunadamente los parámetros de predicción no tienen buenas propiedades para ser transmitidos, debido a que los errores de cuantización o los errores de transmisión pueden ocasionar que los polos del filtro sintetizador en el receptor cuya función de transferencia es:

$$H(z) = \frac{1}{(1-P_1)(1-P_2)}$$

se muevan fuera del circulo unitario en el plano Z, originando inestabilidades en la forma de onda a la salida.

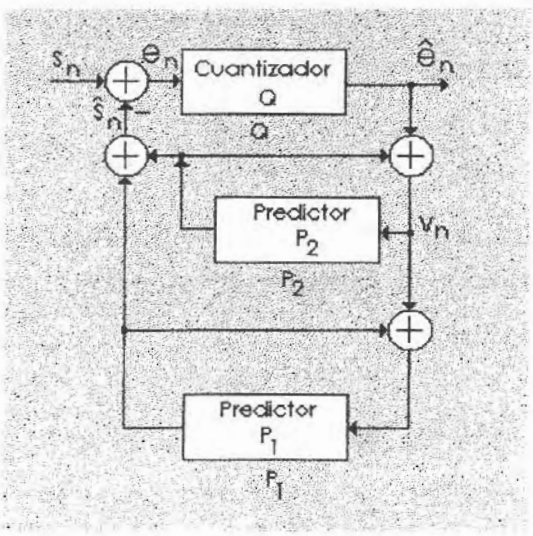


Fig. 2.11 Analizador APC con el cuantizador dentro del lazo de predicción.

En tal caso, a partir de los coeficientes de predicción, se calculan unos parámetros auxiliares conocidos como "Coeficientes de correlación parcial" (PARCOR) cuya magnitud absoluta es menor a la unidad y con ello se asegura la estabilidad del sistema, pues son además, menos susceptibles a los errores de cuantización.

En un diseño mejorado, en lugar de cuantizar la señal de error como se observa en la fig.(2.10), el cuantizador se coloca dentro del lazo del filtro analizador mostrado en la fig.(2.11). Sin este cuantizador, la configuración tendría la función de transferencia:

$$H(z) = (1 - P_1)(1 - P_2)$$

pero ahora se puede eliminar los efectos de pequeños errores de acumulación causados por el error de cuantización. Esto ocurre por que la predicción se efectúa sobre la señal de error cuantizada, luego entonces, el analizador genera la misma secuencia de predicción que lograría el sintetizador en ausencia de errores de transmisión. Además, ya que el analizador compara su predicción con la señal original de entrada, puede entonces corregir distorsiones causadas al cuantizar la señal de error, alterando la secuencia de la señal de error.

Existen varios métodos para cuantizar la señal de error. Sin embargo, para lograr tasas de transmisión a 16 Kbit/s es necesario utilizar cuantización de dos niveles (1 bit/muestra).

En este cuantizador, la señal de error se limita a $+Q$ ó $-Q$, dependiendo si el error es positivo o negativo. La magnitud de Q se determina en cada ventana, de tal forma que la diferencia media cuadrada entre la señal de error y el error cuantizado sea mínimo.

Esto conduce a:

$$Q = \frac{1}{T}\sum_{n=1}^{T}|e_n|$$

Desgraciadamente la ecuación anterior requiere calcular dos veces la señal de error, la primera para estimar a:

$$\sum_{n=1}^{T}|e_n|$$

y calcular Q, y la segunda para obtener la señal de error cuantizada con la disposición de la fig.(2.11). Como esto no es práctico se puede aproximar por:

$$\frac{1}{T}\sum_{n=1}^{T}|e_n| = C\left(\sum_{n=1}^{T}\frac{e_n^2}{T}\right)^{1/2}$$

Donde C es una constante seleccionada apropiadamente.

Por otro lado, U es:

$$\sum_{n=1}^{T}e_n^2 = U$$

y se obtiene durante la solución de los parámetros de predicción (a_i), entonces Q puede estimarse como:

$$\tilde{Q} = C\left(U/T\right)^{1/2}$$

Para encontrar un valor apropiado para C, se calcula la Q_i real y:

$$\tilde{Q} = C\left(U_i/T\right)^{1/2}$$

de un determinado número de ventanas de voz, originadas por distintos locutores y distintas frases. Entonces se escoge C para minimizar la diferencia media cuadrada entre Q_i y CQ_i.

Como resultado de la minimización de:

$$F = \sum_i \left(Q_i - C\tilde{Q}_i\right)^2$$

se obtiene:

$$C = \sum_i Q_i \tilde{Q}_i / \sum_i \tilde{Q}_i^2$$

Un valor típico para C es 0.72.

Después de calcular la ganancia de tono fundamental α, el periodo M, los coeficientes PARCOR k_i y el paso Q del cuantizador, el analizador filtra digitalmente la señal de voz y cuantiza la señal de error que, junto con los parámetros anteriores se envían en una trama al receptor, donde los coeficientes de predicción se regeneran a partir de los coeficientes PARCOR de la siguiente forma:

$$a_j^{(i)} = a_j^{(i-1)} - a_{i-j}^{(i-1)} K_i \qquad j = 1, 2, \ldots, i\text{-}1.$$

$$a_i^{(i)} = K_i \qquad i = 1, 2, \ldots, N.$$

Donde $a_i^{(i)}$ representa la a_i en la i-ésima iteración.

Finalmente, el sintetizador genera una forma de onda que se aproxima a la señal de voz original.

Capítulo 3

SISTEMAS BASADOS EN LA PREDICCION LINEAL.

3.1 Introducción.

Mediante LPC(Linear Prediction Coder)[9] , la señal esta modelada como una combinación de sus valores pasados y presentes y valores pasados de una entrada hipotética a un sistema cuya salida es la señal dada.

Análogamente en el dominio de la frecuencia es equivalente modelar el espectro de la señal por un espectro de *Polos-Ceros.* El sistema consiste de un filtro que es excitado por un tren cuasi-periodico de impulsos ó una fuente de ruido aleatorio como se describe en la fig.(3.1).

La fuente periodica produce sonidos de voz tales como vocales y consonantes nasales y la fuente de ruido produce sonidos no vocalizados(fricativos) tales como las letras(f, th, s, sh).

---

[9] J. Makhoul. "Linear Prediction: A Tutorial Review", Proc. IEEE, Vol63, No. 4, Apr. 1975, pp. 561-580.

Los parámetros del modelo se obtienen con el análisis de mínimos cuadrados en el dominio del tiempo. Los parámetros del filtro determinan la identidad del sonido en particular(características espectrales) de cada uno de los dos tipos de excitación.

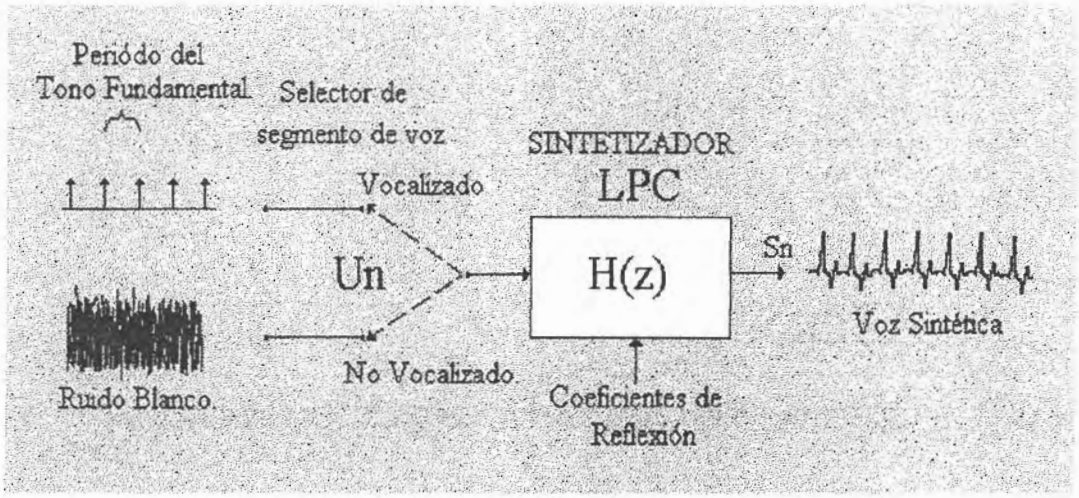


Fig. 3.1 Modelo de producción de voz discreta.

Para una señal de voz, es de interés determinar el tipo de sonido que se tiene, *vocalizado o no vocalizado*, y si se trata de un sonido vocalizado, determinar el periodo de excitación; esto es, la distancia entre un pulso y otro del tren de excitación.

Una herramienta de utilidad es la computadora por lo que en el tratamiento de la voz se ha trabajado ampliamente, lo cual exige muestrear la señal continua en el tiempo $S(t)$ generando una señal discreta en el tiempo $S(nT)$, es decir, aplicando análisis de series en el tiempo tomamos una señal continua en el tiempo $S(t)$; esta es muestreada para obtener una señal discreta en el tiempo $S(nT)$.

Donde n es una variable entera y T es el intervalo de muestra o periodo.

La frecuencia de la muestra es $f = \frac{1}{T}$.

Nota: Abreviamos $S(nT)$ por S_n.

En el modelo general de Predicción Lineal se tiene un modelo poderoso donde S_n se considera una salida de algún sistema con una entrada desconocida U_n de acuerdo con la siguiente relación:

$$S_n = -\sum_{k=1}^{p} a_k S_{n-k} + G \sum_{i=0}^{q} b_i U_{n-i} \qquad b_0 = 1 \qquad (3.1)$$

donde $1 \le k \le p$, b_i, $1 \le i \le q$

y la ganancia G son los parámetros del sistema hipotético.

La ec.(3.1) establece que la "salida" S_n es una función lineal de salidas pasadas y presentes y entradas pasadas. A la ec.(3.1) también podemos expresarla en el dominio de la frecuencia haciendo la transformada Z, $T-Z$ (Transformada Z), en ambos lados de la ec.(3.1). Si $H(z)$ es la función de transferencia del sistema como se muestra en la fig.(3.1), entonces, tenemos que:

$$H(z) = \frac{S(z)}{U(z)} = G \frac{1 + \sum_{i=1}^{q} b_i z^{-i}}{1 + \sum_{k=1}^{p} a_k z^{-k}} \qquad (3.2)$$

donde $$S(z) = \sum_{n=-\infty}^{\infty} S_n Z^{-n} \qquad (3.3)$$

$S(z) \Rightarrow$ T-Z de S_n

$U(z) \Rightarrow$ T-Z de U_n

$H(z) \Rightarrow$ Es el modelo general de la función de transferencia *Polos-Ceros.*

Las raíces del polinomio del numerador representan los *ceros;* mientras que las raíces del polinomio del denominador representan los *polos.*

Existen dos casos especiales del modelo que son de interés:

1) Modelo *Todo-Ceros.* donde los $a_k = 0, \quad 1 \le k \le p.$

conocido como modelo de "Promedio en movimiento" (MA,Moving Average).

2) Modelo *Todo-Polos.* donde los $b_i = 0, \quad 1 \le i \le q.$

conocido como modelo "Autoregresivo" (ARMA, Autoregressive moving Average).

3.1.1. Modelo Todo-Polos.

En nuestro modelo *LPC*, se supone que la señal S_n es determinada mediante una combinación lineal de los valores pasados y por algunas entradas U_n. Por ello usamos el modelo *Todo-Polos.*

$$S_n = -\sum_{k=1}^{p} a_k s_{n-k} + G U_n \qquad (3.4)$$

donde G es un factor de ganancia.

Tal modelo se representa en el dominio del tiempo y dominio de la frecuencia respectivamente, fig.(3.2).

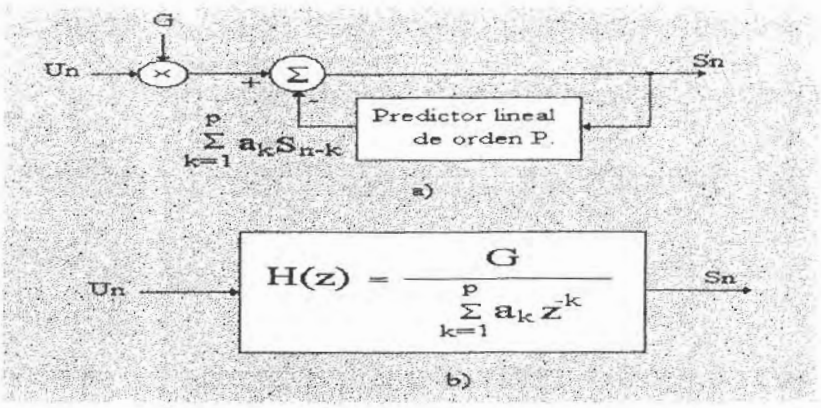


Fig. 3.2 a) Modelo Todo-Polos discreta en el dominio del tiempo
b) Modelo Todo-Polos discreto en el dominio de la frecuencia.

En base a la fig.(3.2), la función de transferencia $H(z)$ de la ec.(3.2) se reduce a una función de transferencia de *todo polos.*

$$H(z) = \frac{G}{1 + \sum_{k=1}^{p} a_k z^{-k}} \qquad (3.5)$$

Dada una señal particular S'_n el problema es determinar los coeficientes del predictor a_k y la ganancia G, de tal forma que el error de aproximación sea mínimo. Para tal minimización del error se utiliza una aproximación intuitiva del método de mínimos cuadrados. El modelo de mínimos cuadrados asume que la entrada U_n es desconocida. Por lo tanto la señal S'_n se puede predecir únicamente en forma aproximada mediante las sumas lineales de muestras pasadas.

Entonces, decimos que la aproximación de S_n es $\tilde{S}_n$.

$$\tilde{S}_n = -\sum_{k=1}^{p} a_k s_{n-k} \qquad (3.6).$$

Donde $\tilde{S}_n$ es la señal predicha de estimación de S_n. Entonces, el error entre el valor actual S_n y el valor predicho $\tilde{S}_n$ esta dado por:

$$e_n = S_n - \tilde{S}_n = S_n + \sum_{k=1}^{p} a_k S_{n-k} \qquad (3.7)$$

donde $e_n = \text{Re}siduo.$

En el método de mínimos cuadrados los parámetros a_k se obtienen como resultado de la minimización de la media o error cuadrático total con respecto a cada uno de los parámetros. El análisis debe desarrollarse a lo largo de dos líneas.

Primero considerando a S_n como señal determinística y después asumiendo que S_n es una muestra de un proceso aleatorio.

3.1.1.1. Señal Determinística.

Denotamos el error cuadrático total por medio de *E*, donde:

$$E = \sum_n e_n^2 = \sum_n (S_n + \sum_{k=1}^{p} a_k S_{n-k})^2 \qquad (3.8)$$

Minimizando a *E* sin especificar el rango de las sumas haciendo a :

$$\frac{\partial E}{\partial a_i} = 0, \qquad 1 \le i \le p. \qquad (3.9)$$

En un caso concreto, si $P = 4$, por ejemplo al desarrollar la ec.(3.8) se obtiene:

$$E = \sum_n (S_n^2 + a_1 S_n S_{n-1} + a_2 S_n S_{n-2} + a_3 S_n S_{n-3} + a_4 S_n S_{n-4} +$$

$$+ a_1 S_n S_{n-1} + a_1^2 S_{n-1}^2 + a_1 a_2 S_{n-1} S_{n-2} + a_1 a_3 S_{n-1} S_{n-3} + a_1 a_4 S_{n-1} S_{n-4} +$$

$$+ a_2 S_n S_{n-2} + a_1 a_2 S_{n-1} S_{n-2} + a_2^2 S_{n-2}^2 + a_2 a_3 S_{n-2} S_{n-3} + a_2 a_4 S_{n-2} S_{n-4} +$$

$$+ a_3 S_n S_{n-3} + a_1 a_3 S_{n-1} S_{n-3} + a_2 a_3 S_{n-2} S_{n-3} + a_3^2 S_{n-3}^2 + a_3 a_4 S_{n-3} S_{n-4} +$$

$$+ a_4 S_n S_{n-4} + a_1 a_4 S_{n-1} S_{n-4} + a_2 a_4 S_{n-2} S_{n-4} + a_3 a_4 S_{n-3} S_{n-4} + a_4^2 S_{n-4}^2 .)$$

al derivar parcialmente respecto a cada a_i resulta:

$$\frac{\partial E}{\partial a_i} = 2 \sum_n (\sum_{k=1}^{p} a_k S_{n-i} S_{n-k}) + 2 \sum_n S_n S_{n-i} = 0$$

de donde se obtiene:

$$\sum_{k=1}^{p} a_k \sum_n S_{n-i} S_{n-k} = - \sum_n S_n S_{n-i} \qquad 1 \le i \le p. \qquad (3.10).$$

La ec.(3.10) en terminología de mínimos cuadrados se conoce como la ECUACION NORMAL. Para cualquier definición de la señal S_n, la ec.(3.10) forma un conjunto de P ecuaciones con P incógnitas desconocidas, la cual se pueden resolver para los coeficientes del predictor $\{a_k, 1 \le k \le p\}$ que minimiza el error cuadrático total E en la ec.(3.8).

El error cuadrático total mínimo E_p, se obtiene a partir de la expansión de la ec.(3.8) y sustituyendo en la ec.(3.10). Por lo tanto,

$$E = \sum_n (S_n + \sum_{k=1}^{p} a_k S_{n-k})^2 = \sum_n [S_n^2 + 2 S_n \sum_{k=1}^{p} a_k S_{n-k} + (\sum_{k=1}^{p} a_k S_{n-k})^2]$$

$$E = \sum_n S_n^2 + 2 \sum_n \sum_{k=1}^{p} a_k S_n S_{n-k} + \sum_n (\sum_{k=1}^{p} a_k S_{n-k} \sum_{i=1} a_i S_{n-i})$$

$$E = \sum_n S_n^2 + 2 \sum_n \sum_{k=1}^{p} a_k S_n S_{n-k} + \sum_{k=1}^{p} a_k \sum_n \sum_{i=1}^{p} a_i S_{n-k} S_{n-i}$$

pero sabemos que:

$$\sum_{k=1}^{p} a_k \sum_n S_{n-k} S_{n-i} = -\sum_n S_n S_{n-i}$$

entonces,

$$E = \sum_n S_n^2 + 2\sum_n \sum_{k=1}^{p} a_k S_n S_{n-k} + \sum_{k=1}^{p} a_k \left(-\sum_n S_n S_{n-i} \right)$$

$$E = \sum_n S_n^2 + \sum_n \sum_{k=1}^{p} a_k S_n S_{n-k} \cong E_p$$

$$E_p = \sum_n S_n^2 + \sum_n \sum_{k=1}^{p} a_k S_n S_{n-k} \qquad (3.11).$$

Especificando ahora el intervalo de la suma sobre n en las ecs.(3.8), (3.9) y (3.10), se da origen a dos casos de interés que conducen a dos métodos distintos para la estimación de los parámetros:

a) Método de Autocorrelación.
b) Método de Covarianza.

3.1.1.1.1. Método de Autocorrelación.

En este método asumimos que el error en la ec.(3.8) se minimiza sobre una duración infinita, $-\infty < n < \infty$. Las ecs. (3.10) y (3.11) se pueden reducir a :

$$\sum_{k=1}^{p} a_k \sum_{n=-\infty}^{\infty} S_{n-k} S_{n-i} = -\sum_{n=-\infty}^{\infty} S_n S_{n-i} \qquad 1 \le i \le p.$$

$$\sum_{k=1}^{p} a_k R(i-k) = -R(i) \qquad 1 \le i \le p. \qquad (3.12).$$

$$E_p = \sum_{n=-\infty}^{\infty} S_n^2 + \sum_{k=1}^{p} a_k \sum_{n=-\infty}^{\infty} S_n S_{n-k}$$

$$E_p = R(0) + \sum_{k=1}^{p} a_k R(k) \qquad (3.13).$$

donde :

$$R(i) = \sum_{n=-\infty}^{\infty} S_n S_{n+i} \qquad (3.14).$$

que es la función de autocorrelación de la señal Sn y se cumple además que es una función PAR.

$$R(-i) = R(i) . \qquad (3.15).$$

A partir de los coeficientes $R(i-k)$ de la ec.(3.12) se forma lo que se conoce como matriz de autocorrelación. Una matriz de autocorrelación es una matriz simétrica de Toeplitz(la matriz de Toeplitz es aquella que todos sus elementos a lo largo de cada diagonal son iguales).

En la práctica, la señal S_n se conoce únicamente sobre un intervalo finito o sólo interesa el valor de la señal sobre ese intervalo finito.

Un método común es multiplicar la señal S_n por una función ventana W_n para obtener otra señal S'_n que es cero fuera de algún intervalo,

$$0 \le n \le N-1 .$$

Esto es: .

$$S'_n = \begin{cases} S_n W_n & 0 \le n \le N-1 \\ 0 & Otro \quad caso \end{cases} \qquad (3.16)$$

La función de autocorrelación se determina entonces por medio de:

$$R(i) = \sum_{n=0}^{N-i-1} S'_n S'_{n+i} \qquad i \geq 0 \qquad (3.17).$$

3.1.1.1.2. Método de Covarianza.

En contraste con el método anterior, aquí se supone que el error E en la ec.(4.8) se minimiza sobre un intervalo finito, $0 \leq n \leq N\text{-}1$. Por lo que las ecs.(3.10) y (3.11) se pueden reducir a:

$$\sum_{k=1}^{p} a_k \sum_{n=0}^{N-1} S_{n-k} S_{n-i} = -\sum_{n=0}^{N-1} S_n S_{n-i} \qquad 1 \leq i \leq p.$$

$$\sum_{k=1}^{p} a_k \Psi_{ki} = \Psi_{0i} \qquad 1 \leq i \leq p. \qquad (3.18).$$

$$E_p = \sum_{n=0}^{N-1} S_n^2 + \sum_{k=1}^{p} a_k \sum_{n=0}^{N-1} S_n S_{n-k}$$

$$E_p = \Psi_{00} + \sum_{k=1}^{p} a_k \Psi_{0k} \qquad (3.19).$$

Donde:

$$\Psi_{ik} = \sum_{k=0}^{N-1} S_{n-i} S_{n-k} \qquad (3.20).$$

es la covarianza de la señal en el intervalo dado.

Los coeficientes Ψ_{ki} en la ec.(3.18) forman la matriz de covarianza que cumple las condiciones de simetría ($\Psi_{ik} = \Psi_{ki}$) . Sin embargo, a diferencia de la matriz de autocorrelación, los términos a lo largo de cada diagonal no son iguales.

Esto puede ser parecido escribiendo de la ec.(3.20):

$$\Psi_{i+1,k+1} = \Psi_{ik} + S_{-i-1}S_{-k-1} - S_{N-1-i}S_{N-1-k} \qquad (3.21).$$

Notamos que en la ec.(3.21) los valores de la señal S_n se tienen que conocer en $-p \le n \le N-1$. Hay un total de $P+N$ muestras.

Este método reduce al método anterior como el intervalo en el cual n varia hasta infinito.

3.1.1.2. Señal Aleatoria.

Si se supone que la señal S_n es una muestra de un proceso aleatorio, entonces, el error e_n en la ec.(3.7) es:

$$e_n = S_n - \widetilde{S}_n = S_n + \sum_{k=1}^{p} a_k S_{n-k}$$

que es también una muestra de un proceso aleatorio.

En el método de mínimos cuadrados, minimizaremos el valor esperado del cuadrado del error. Por lo tanto,

$$E = \xi(e_n^2) = \xi\left(S_n + \sum_{k=1}^{p} a_k S_{n-k}\right)^2 \qquad (3.22).$$

Siguiendo el mismo procedimiento para obtener la ec.(3.10) aplicando $\frac{\partial E}{\partial a_i} = 0$, a la ec.(3.22) para obtener las ecs. normales,

$$\sum_{k=1}^{p} a_k \xi(S_{n-k}S_{n-i}) = -\xi(S_n S_{n-i}); \qquad 1 \le i \le p. \qquad (3.23).$$

El error promedio estará dado por:

$$E_p = \xi(S_n^2) + \sum_{k=1}^{p} a_k \xi(S_n S_{n-k}) \qquad (3.24).$$

Tomando los valores esperados en la ec.(3.23) y la ec.(3.24) dependiendo si el proceso S_n es estacionario o no, se tiene el caso estacionario y el caso no estacionario.

3.1.1.2.1. Caso Estacionario.

Para un proceso estacionario S_n , se tiene:

$$\xi(S_{n-k} S_{n-i}) = R(i-k) \qquad (3.25).$$

Donde $R(i)$ es la autocorrelación del proceso.

Las ecs.(3.23) y (3.24) se reducen a las ecs.(3.12) y (3.13) respectivamente. La única diferencia es que aquí la autocorrelación es de un proceso estacionario en lugar de una señal determinística.

3.1.1.2.2. Caso No Estacionario.

Para un proceso no estacionario S_n se tiene que :

$$\xi(S_{n-k} S_{n-i}) = R(n-k, n-i) \qquad (3.26)$$

Donde $R(t,t')$ es la autocorrelación no estacionaria entre los tiempos t y. t'. $R(n-k, n-i)$ es una función del tiempo indexado en n.

Sin pérdidas de generalidad, se puede asumir que estamos interesados en estimar los parámetros a_k en el instante *n=0.* Entonces, las ecs.(3.23) y (3.24) se reducen a:

$$\sum_{k=1}^{p} a_k R(-k,-i) = -R(0,-i). \qquad (3.27).$$

$$E_p' = R(0,0) + \sum_{k=1}^{p} a_k R(0,k). \qquad (3.28).$$

En la estimación de los coeficientes de autocorrelación no estacionarios de la señal S_n, se observa que los procesos no estacionarios no son ergódicos por lo que no puede sustituirse el promedio de ensamble por un promedio de tiempo. Sin embargo, para una cierta clase de procesos no estacionarios conocidos como procesos localmente estacionarios, es razonable estimar la función de autocorrelación con respecto a un punto en el tiempo como un promedio en corto tiempo.

3.1.2. Cálculo de los Parámetros del Predictor.

En cada una de las dos formulaciones de la Predicción Lineal, los coeficientes del predictor $\{a_k, 1 \le k \le p\}$, se pueden calcular por medio de resolver un conjunto de P ecuaciones con P incógnitas. Estas ecuaciones son, la ec.(3.12) para el método de autocorrelación(estacionario) y la ec.(3.18) para el método de covarianza(no estacionario).

Existen varios métodos típicos para desarrollar el cálculo necesario(por ejemplo, la reducción o eliminación de Gauss y el método de reducción de Crout). Estos métodos generales requieren $P^3/3 + 0(P^2)$ operaciones (multiplicaciones o divisiones) y P^2 localidades de almacenamiento. Sin embargo, observamos que las ecs.(3.12) y (3.18) ambas resultan ser matrices de covarianza, es decir, casi-definitivamente positivas, aunque en la

práctica usualmente lo son. Así, las ecs.(3.12) y (3.18) se pueden resolver más eficientemente por el método de descomposición de Cholesky.

Estos métodos requieren alrededor de la mitad del cálculo $P^3/6 + 0\ (P^2)$ y alrededor de la mitad de la memoria $P^2/2$ de la empleada por los métodos generales.

La ec.(3.12) se puede expander a una matriz de forma:

$$\begin{bmatrix} R_0 & R_1 & R_2 & R_3 & \cdots & R_{p-1} \\ R_1 & R_0 & R_1 & R_2 & \cdots & R_{p-2} \\ R_2 & R_1 & R_0 & R_1 & \cdots & R_{p-3} \\ R_3 & R_2 & R_1 & R_0 & \cdots & R_{p-4} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ R_{p-1} & R_{p-2} & R_{p-3} & R_{p-4} & \cdots & R_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ \vdots \\ R_p \end{bmatrix} \qquad (3.29)$$

Note que la matriz de autocorrelación PxP es simétrica y los elementos a lo largo de la diagonal son idénticos(matriz de Toeplitz).

Levinson derivó un procedimiento recursivo para resolver este tipo de ecuaciones. El procedimiento fué mas tarde reformulado por Robinson. El método de Levinson asume un vector columna en el lado derecho de la ec.(3.29) que es un vector columna general. Haciendo uso del hecho de que este vector columna consta de los mismos elementos encontrados en la matriz de autocorrelación, otro método atribuido a Durbin surge en el cual es dos veces tan rápido como el de Levinson.

El método requiere solamente $2p$ localidades de almacenamiento y $p^2 + O(P)$ operaciones. Un gran ahorro de los métodos generales.

El procedimiento recursivo de Levinson-Durbin se especifica como sigue:

$$E_0 = R(0) \qquad \text{(3.30a)}.$$

$$k_i = -\left[R(i) + \sum_{j=1}^{i-1} a_j^{i-1} R(i-j) \right] \Big/ E_{i-1} \qquad \text{(3.30b)}.$$

$$a_i^{(i)} = a_i$$

$$a_j^{(i)} = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)} \qquad 1 \le j \le i-1. \qquad \text{(3.30c)}.$$

$$E_i = \left(1 - k_i^2\right) * E_{i-1} \qquad \text{(3.30d)}.$$

Las ecs.(3.30b) y (3.30d) se resuelven recursivamente para $i=1, 2, 3, \ldots, p$.

La solución final esta dada por:

$$a_j = a_j^{(p)}. \qquad 1 \le i \le p. \qquad \text{(3.30e)}.$$

El modelo de Predicción Lineal para la producción de voz desempeña dos funciones básicas.

Por medio de un filtro lineal modela las características del conducto vocal y la forma espectral de la fuente vocal.

La segunda parte proporciona la excitación al filtro lineal. El modelo supone que la señal de voz se puede clasificar en dos tipos, *vocalizada* y *no vocalizada,* además de conocer el periodo del tono fundamental de excitación. Para segmentos *vocalizados,* la excitación es un tren de pulsos cuasi-periodicos y para segmentos *no vocalizados* la excitación es ruido blanco.

Con este modelo es difícil producir voz de alta calidad aún utilizando tasas de transmisión altas.

El problema central es la forma inflexible con la cual se genera la excitación. Hay más de dos modos con las cuales se excita al conducto vocal y a menudo esos modos son una mezcla.

Aún cuando la forma de onda de voz es claramente periodica, no es muy buena aproximación suponer que hay un solo punto de excitación en un periodo del tono fundamental completo.

Existe cierta evidencia de que además de la excitación principal que ocurre al cierre glotal, hay una segunda excitación no solo en la apertura glotal y durante la fase de apertura sino también después del cierre.

Estos resultados sugieren que la excitación para segmentos vocalizados debe consistir en varios pulsos en un periodo del tono fundamental, en lugar de uno solo en el inicio del periodo. A este modelo se le conoce como excitación multi-pulso ya que no hace suposiciones *a priori* sobre la naturaleza de la señal de excitación. La excitación consiste de una secuencia de pulsos para cualquier tipo de voz, incluyendo segmentos vocalizados o no vocalizados.

Es interesante notar que unos cuantos pulsos (típicamente 8 pulsos cada 10 ms.) son suficientes para generar diferentes tipos de sonidos con una pequeña distorsión audible. De esta forma se elimina el bloque de decisión entre sonidos vocalizados y no vocalizados y la determinación del periodo del tono fundamental.

Por supuesto, si el número de pulsos se incrementa arbitrariamente hasta un valor tal que hay un pulso en cada instante de muestreo, será posible duplicar la forma de onda original (a expensas de una mayor tasa de transmisión).

3.2 Excitación Multi-Pulso (MPE)[10] .

A continuación se muestra en la fig.(3.3) el diagrama a bloques de un sintetizador de voz LPC con excitación Multi-Pulso. La excitación para el filtro Todo-Polos se proporciona mediante un generador de excitación que produce una secuencia de pulsos localizados en los momentos $t_1, t_2, \cdots t_n, \cdots$ con amplitudes respectivas $\alpha_1, \alpha_2, \cdots \alpha_n$.

La señal sintetizada $\tilde{S}n$ se filtra para proporcionar una señal de voz continua en el tiempo $\tilde{S}(t)$.

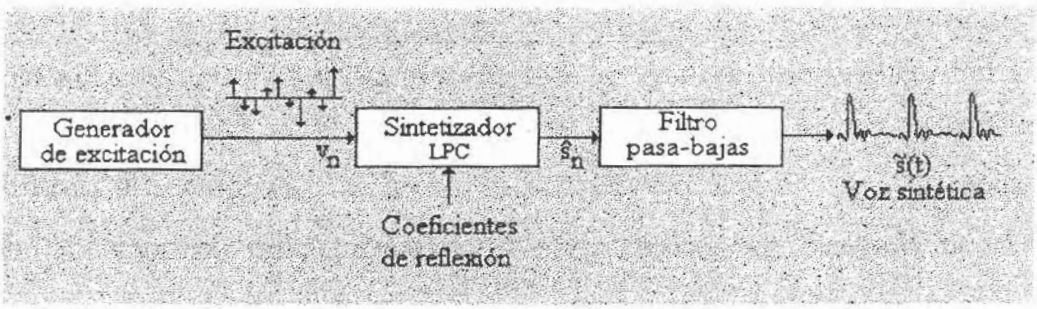


Fig. 3.3. Diagrama a bloques de un sintetizador LPC con excitación Multi-Pulso

Un procedimiento de análisis por medio de síntesis para determinar las posiciones y amplitudes de los pulsos se observa en la fig.(3.4).

El sintetizador LPC produce muestras $\tilde{S}n$ de señal de voz sintética como resultado a la excitación v_n. Las muestras de voz sintética se comparan con las correspondientes muestras de voz original para producir una señal de error e_n. Éste error no es muy significativo y debe modificarse para tomar en cuenta la forma en que la percepción humana trata el error.

[10] B. S. Atal, J. R. Remde, "A new model of LPC excitation for producing Natural-Sounding Speech at Low bit Rates", Proc. IEEE, Int. Conf. of Speech Signal Processing, Apr. 1982, pp. 614-617.

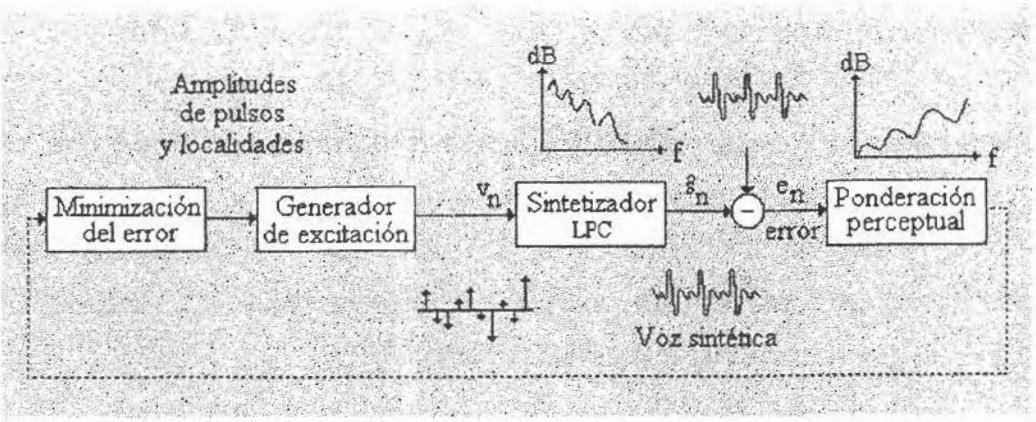


Fig 3.4. Diagrama a bloques del procedimiento de análisis por síntesis para determinar localidades y amplitudes para la excitación multi-pulso.

En términos generales, la señal de error debe desenfatizarse en las regiones espectrales donde se localizan los formantes por medio de un filtro lineal. De esta forma, la señal de error es ponderada para producir una medición significativa desde el punto de vista subjetivo de la diferencia entre las señales $\tilde{S}_n$ y S_n. El error ponderado se eleva al cuadrado y se promedia sobre un intervalo de tiempo corto (5 a 10 ms.) para producir la media cuadrada del error ponderado ε.

De esta forma, las posiciones y amplitudes de los pulsos de excitación se seleccionan para minimizar el error ε.

Para tener una mejor medida de la señal de error e_n se define al error ponderado en el dominio de la frecuencia como:

$$\xi = \int_0^{fs} \left| S(f) - \tilde{S}(f) \right|^2 W(f) df$$

Donde $S(f)$ y $\tilde{S}(f)$ son las transformadas de Fourier de la señal de voz original y sintética respectivamente. $W(f)$ es una función de ponderación

convenientemente escogida, f es la frecuencia y f_s la frecuencia de muestreo.

La función de ponderación $W(f)$ se escoge para desenfatizar las regiones de los formantes en el espectro del error.

Sea $1-P(z)$ el filtro *LPC* inverso en notación de transformada *Z*, entonces, la envolvente espectral de corto tiempo esta dada por:

$$S_e(f) = \left| \frac{k}{1-P(z)} \right|^2$$

Donde k es el error cuadrático medio de predicción. El filtro inverso $1-P(z)$ en términos de los coeficientes de predicción a_k es:

$$1-P(z) = 1-\sum_{k=1}^{p} a_k z^{-k}$$

Si $W(z)$ es la función de transferencia del filtro de ponderación, entonces, una selección apropiada para $W(z)$ es:

$$W(z) = \left[1-\sum_{k=1}^{p} a_k z^{-k}\right] \Big/ \left[1-\sum_{k=1}^{p} a_k \gamma^k z^{-k}\right]$$

Donde γ es una fracción entre 0 y 1 que controla el incremento de la señal de error en las regiones de los formantes. El filtro cambia de $W(z)=1$ para $\gamma=1$ a $W(z)=1-P(z)$ para $\gamma=0$. El valor de γ se determina por el grado que se desea desenfatizar al espectro del error en las regiones de los formantes.

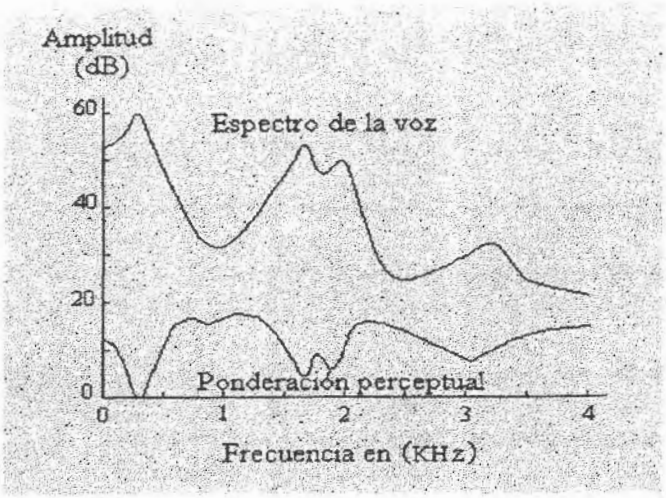


Fig. 3.5. Respuesta en frecuencia del filtro de ponderación perceptual.

El valor óptimo de γ se debe determinar por pruebas subjetivas apropiadas. Sin embargo, la selección no es tan crítica. Un valor típico para una frecuencia de muestreo de 8 KHz es 0.8.

La fig.(3.5) muestra un ejemplo del espectro de la voz y la respuesta en frecuencia del filtro de ponderación de la señal de error correspondiente.

Una solución eficiente para determinar la posición y amplitud de los pulsos es obtener la excitación calculando un pulso cada vez, de esta forma se convierte de un problema con varias incógnitas a uno con sólo dos incógnitas.

Una solución cercana para la amplitud del pulso se obtiene igualando a cero la derivada del error cuadrático medio respecto a la amplitud incógnita. El error cuadrático medio esta entonces en función únicamente de la localización del pulso. La localidad óptima se encuentra calculando el error para todas las localidades posibles de un intervalo de tiempo dado, y se posiciona en la de mínimo error. El procedimiento se puede mejorar observando que las amplitudes incógnitas para todos los pulsos se pueden

determinar en un paso simple, resolviendo un conjunto de ecuaciones lineales una vez que las posiciones de todos los pulsos son conocidas.

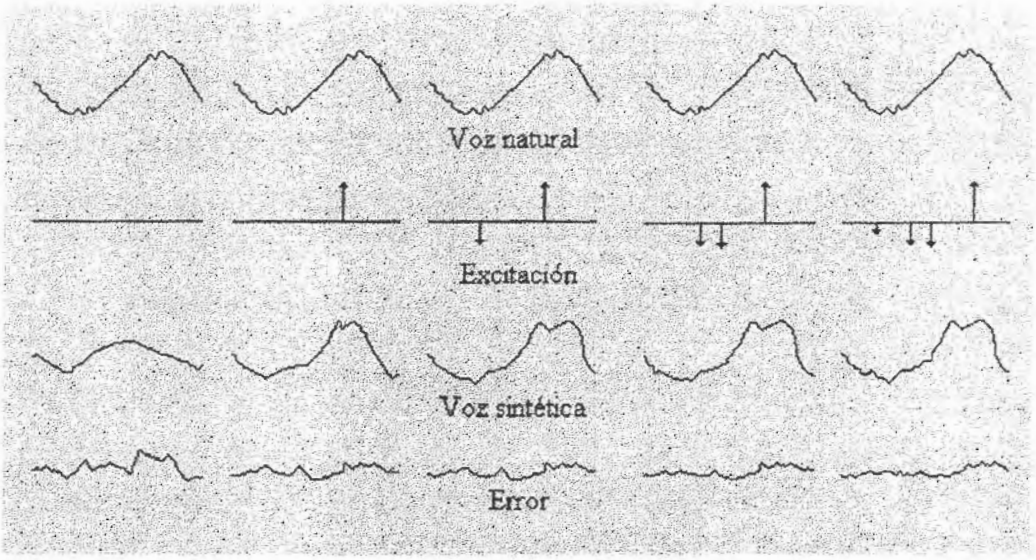


Fig. 3.6. Procedimiento para reducir el error de aproximación.

El procedimiento para encontrar las posiciones y amplitudes de los pulsos en determinado intervalo de tiempo se puede resumir de la siguiente manera:

En el inicio, sin ningún pulso de excitación la voz sintetizada se genera completamente por la memoria del filtro sintetizador de intervalos de síntesis anteriores. La contribución de esta memoria del pasado se resta de la señal de voz y así se determina la posición y amplitud de un sólo pulso que minimiza el error cuadrático medio ponderado. Una nueva señal de error se calcula ahora restando la contribución del pulso recién determinado. El proceso de localizar nuevos pulsos para reducir el error cuadrático medio ponderado continua hasta poder reducir el error en límites aceptables. La fig.(3.6) ilustra el procedimiento de minimización del error. El proceso de añadir pulsos puede continuar pero, en la práctica, después de colocar 8 pulsos en un intervalo de 10 ms, la mejoría es mínima.

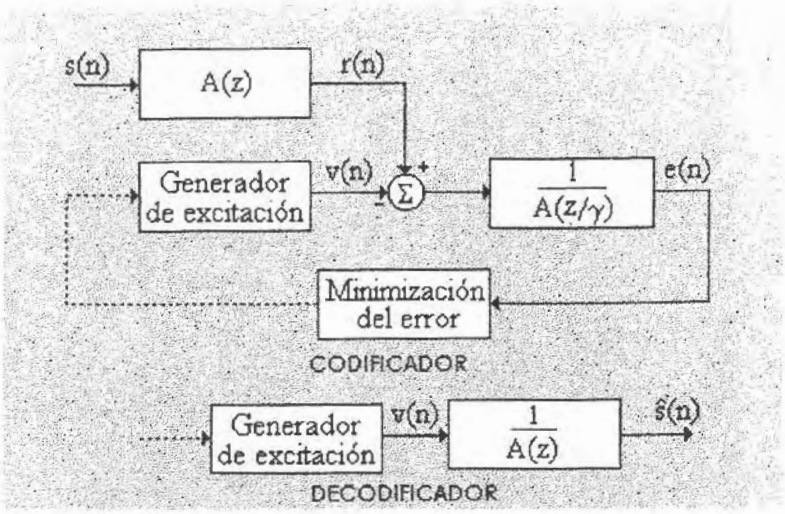


Fig. 3.7. Diagrama a bloques del codificador con excitación de pulsos regulares

3.3. Excitación de Pulsos-Regulares[11] .

La estructura básica del codificador se observa en la fig.(3.7). La señal residual r_n, se obtiene filtrando la señal de voz $S(n)$ a través del filtro variante en el tiempo $A(z)$ de orden p.

$$A(z) = 1 + \sum_{k=1}^{p} a_k z^{-k}$$

Este filtro se caracteriza mediante la técnica de Predicción Lineal descrita anteriormente. La diferencia entre la señal residual r_n y un cierto modelo de señal residual v_n (la cual se definirá a continuación), alimenta al filtro de ponderación $1/A(z/\gamma)$.

$$\frac{1}{A(z/\gamma)} = \frac{1}{1 + \sum_{k=1}^{p} a_k \gamma^k z^{-k}} \qquad 0 \le \gamma \le 1.$$

[11] P. Kroon, E. F. Deprettere, R. J. Sluyter, "Regular-Pulse Excitation: A Novel Approach to Effective and Efficient Multipulse Coding Speech", IEEE ASSP-34, No. 5, Oct. 1986, pp. 1054-1063.

Este filtro cuya función es ponderar el error, juega el mismo papel que el filtro de ponderación de error de los codificadores con excitación multi-pulso.

La diferencia ponderada resultante e_n, se eleva al cuadrado y se acumula; posteriormente es utilizada como medida para determinar la efectividad del modelo propuesto v_n de la señal residual r_n.

La secuencia de excitación v_n, se determina para segmentos con L muestras cada uno, de acuerdo al siguiente procedimiento.

Para cada segmento se requiere una *versión "re-muestreada"* de un cierto vector óptimo $b = (b(1),....,b(Q))$, con longitud $Q(Q < L)$. Cada segmento de la señal de excitación contiene Q muestras equidistantes de amplitud diferentes de cero, mientras que el resto de las muestras son iguales a cero. La separación entre muestras distintas de cero es $N = L / Q$.

La duración de un segmento de tamaño L es de 5 ms típicamente.

Cada segmento de excitación consta de N conjuntos de Q muestras equidistantes con magnitud distinta a cero originando N secuencias de excitación candidatas.

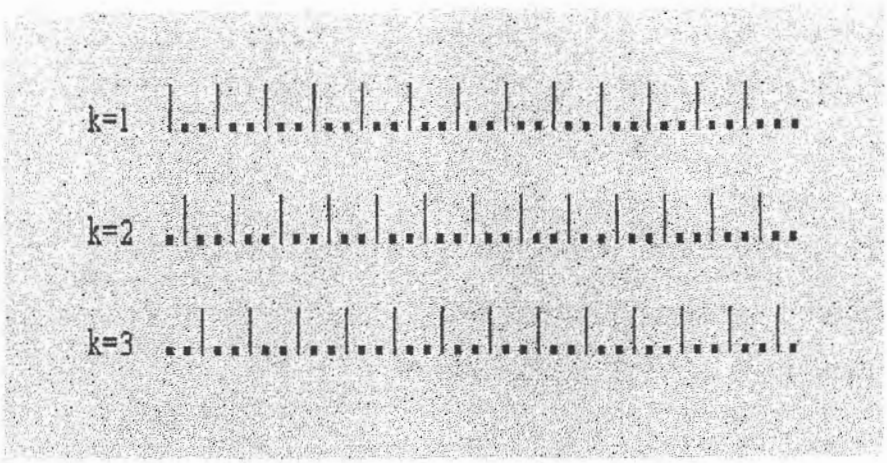


Fig. 3.8. Posibles rejillas de excitación con L=40 y N=3.

La fig.(3.8) muestra las posibles rejillas de excitación para un segmento con 40 muestras y separación $N=3$. En esta figura, las posiciones de los pulsos se marcan con una línea vertical y las muestras cero por puntos.

Si k $(k = 1,\ldots,N)$ denota la fase de la versión remuestreada del vector $b^{(k)}$, por ejemplo, la posición de la primera muestra con amplitud diferente a cero, entonces, es necesario calcular para cada valor de k las amplitudes $b^{(k)}(\bullet)$ que minimizan el error cuadrado acumulado.

El vector que logre el mínimo error se selecciona y transmite. El procedimiento de decodificación se observa en la fig.(3.7).

Algoritmo de Codificación.

Sea M_k la matriz Q por L con los elemento:

$$\begin{aligned} &m_{ij} = 1 \quad si \quad j = i * N + k - 1 \\ &m_{ij} = 0 \qquad Otro \quad caso \end{aligned} \qquad \begin{aligned} &0 \le i \le Q-1 \\ &0 \le j \le L-1 \end{aligned} \qquad (3.31)$$

El segmento de excitación o vector fila $v^{(k)}$ correspondiente a la k-ésima rejilla de excitación puede escribirse como:

$$v^{(k)} = b^{(k)} M_k \tag{3.32}$$

Sea H una matriz triangular superior de L por L cuya k-ésima columna $(j = 0, \cdots, L-1)$ contiene la respuesta truncada $h(n)$ del filtro de ponderación de error $1/A(z/\gamma)$ causada por un impulso unitario $\delta(n-j)$, esto es:

$$H = \begin{bmatrix} h(0) & h(1) & \cdots & h(L-1) \\ 0 & h(0) & \cdots & h(L-2) \\ 0 & 0 & \cdots & h(L-3) \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & h(0) \end{bmatrix} \tag{3.33}$$

Si e_0 denota la salida del filtro de ponderación debido al contenido en la memoria (por ejemplo, la salida como resultado del estado inicial del filtro) del intervalo anterior, entonces, la señal $e(n)$ producida por el vector de entrada $b^{(k)}$ se puede describir como:

$$e^{(k)} = e^{(0)} - b^{(k)} H_k \qquad k = 1, \cdots, N \tag{3.34}$$

Donde

$$e^{(0)} = e_0 + rH \tag{3.35}$$

$$H_k^{\cdot} = M_k H \tag{3.36}$$

y el vector r representa el residuo $r(n)$ del segmento actual.

El objetivo es minimizar el error cuadrado:

$$E^{(k)} = e^{(k)} e^{(k)t} \tag{3.37}$$

Donde t denota la transpuesta.

Para una determinada fase (k), las amplitudes óptimas $b^{(k)}(\bullet)$ se pueden calcular a partir de las ecs.(3.34) y (3.37) considerando que el producto $e^{(k)} H_k^t$, que de un modo mide el error, tiende a cero, de aquí:

$$e^{(k)} H_k^t = e^{(0)} H_k^t - b^{(k)} H_k H_k^t$$

o

$$b^{(k)} = e^{(0)} H_k^t \left[H_k H_k^t \right]^{-1} \tag{3.38}$$

Sustituyendo la ec.(3.38) en la ec.(3.34) y el resultado en la ec.(3.37), la expresión para el error E resulta:

$$e^{(k)} = e^{(0)} - e^{(0)} H_k^t [H_k H_k^t]^{-1} H_k$$

$$e^{(k)} = e^{(0)} [I - H_k^t [H_k H_k^t]^{-1} H_k]$$

$$E^{(k)} = e^{(0)} [I - H_k^t [H_k H_k^t]^{-1} H_k] e^{(k)t} \tag{3.39}$$

Se selecciona el vector $b^{(k)}$ que origina el mínimo valor de $E^{(k)}$ para toda k. El vector resultante de óptima excitación $v^{(k)}$ se caracteriza completamente por su fase k y el correspondiente vector de amplitudes $b^{(k)}$. El procedimiento completo comprende la solución de N conjuntos de ecuaciones lineales dadas en la ec.(3.38).

Aunque el efecto de ponderación de ruido se puede escuchar, el mecanismo real que hay detrás de este efecto no es muy claro.

El parámetro γ del filtro de ponderación determina la cantidad de potencia de ruido en las regiones de los formantes del espectro de la voz.

La ponderación del ruido reduce la relación señal a ruido pero mejora la calidad perceptual de la voz. Para una frecuencia de muestreo de 8 KHz, un valor típico para γ se encuentra entre 0.8 y 0.9.

Además del valor de γ, el orden del filtro de ponderación de ruido tiene cierta importancia. Como primera alternativa los coeficientes a_k y el orden p de $1/A(z/\gamma)$ son iguales a los del predictor $A(z)$, pero puede calcularse un predictor de orden $(q < p)$ y usar los q coeficientes resultantes para definir el filtro de ponderación.

El hecho de que el filtro de ponderación sea variante en el tiempo proporciona una contribución significante a la complejidad del procedimiento de análisis ya que el sistema de ecuaciones lineales a resolver se construye completamente a partir de la respuesta impulsional de este filtro.

Es evidente que la complejidad de cálculo seria considerablemente menor si el filtro de ponderación se eligiese de tal forma que la matriz a invertir no dependiera de los datos de corto tiempo. Para lograrlo, se escoge un filtro de ponderación $1/C(z/\gamma)$.

$$\frac{1}{C(z/\gamma)} = \frac{1}{1+\sum_{k=1}^{q} C_k \gamma^K Z^{-K}} \qquad (3.40)$$

Donde $\{C_K\}$ son los coeficientes del predictor fijo de bajo orden utilizado en los sistemas DPCM que se basan en las características espectrales promedio de la señal de voz.

El valor seleccionado para γ fue 0.8 y al utilizar los coeficientes de predicción $\{C_K\}$ para filtros fijos con diferentes órdenes ($q = 1 \quad a \quad 3$) [12] se observa que los efectos de los filtros $1/A(z/\gamma)$ y $1/C(z/\gamma)$ son casi equivalentes.

Este resultado se utiliza para reducir significativamente la complejidad del codificador. El procedimiento de análisis para el codificador *RPE* necesita la solución de N conjuntos de ecuaciones lineales, donde N representa el espacio entre pulsos sucesivos en un segmento del modelo de excitación.

Sin embargo, el problema de invertir las matrices $H_k H_k^t$ puede solucionarse de una manera más eficiente. El algorítmo puede reconfigurarse para forzar al producto H_k en H_k^t en la ec.(3.22) a una simple matriz de Toeplitz la cual es independiente de la fase *k*.

$$\text{Sea} \quad h(n) = \gamma^n g(n) \qquad n = 0,1,2,\cdots \tag{3.41}$$

La respuesta impulsional de filtro $1/A(z/\gamma)$ es $h(n)$, donde $g(n)$ es la respuesta impulsional del filtro todo-polos $1/A(z)$. Para valores de $|\gamma|$ menores a 1, $h(n)$ converge más rápidamente a cero que $g(n)$ y como resultado la matriz L por $2L$ construida con $h(n)$ se puede aproximar por una matriz de Toeplitz triangular superior H.

$$H = \begin{bmatrix} h(0) & h(1) & \cdots & h(L-1) & 0 & \cdots & 0 \\ 0 & h(0) & \cdots & h(L-2) & h(L-1) & \vdots & 0 \\ 0 & 0 & \cdots & h(L-3) & h(L-2) & & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & h(0) & \cdots & h(L-1) & 0 \end{bmatrix} \tag{3.42}$$

[12] J.L. Flanagan, M.R. Schroeder, B.S. Atal, R.E. Crochiere, N.S. Jayant, J.M. Tribolet, "Speech Coding", IEEE COM-27, No. 4, Apr. 1979, pag. 710-733.

Al sustituir la ec.(3.42) en las ecs.(3.35) y (3.36), los vectores e_0, $e^{(0)}$ y $e^{(k)}$ en las ecs.(3.35) y (3.34) serán ahora de longitud $2L$, mientras que los vectores $v^{(k)}$ y r en las ecs.(3.32) y (3.35) mantienen su longitud L.

Después de sustituir la ec.(3.42) en las ecs.(3.35) y (3.36) y descartando la aproximación de orden cero e_0 en la ec.(3.35), entonces la ec.(3.34) se convierte en:

$$e^{(k)} = e_0 + rH - b^{(k)} H_k \qquad \textit{Donde } e_0\text{=0.} \qquad \textit{Por lo tanto,}$$

$$e^{(k)} = rH - b^{(k)} H_k \qquad (3.43)$$

Y la ec.(3.38) en :

$$b^{(k)}[H_k H_k^t] = [e_0 + rH] H_k^t \qquad \textit{Donde } e_0\text{=0.} \qquad \textit{Por lo tanto,}$$

$$b^{(k)}[H_k H_k^t] = rH\, H_k^t\, M_k^t \qquad (3.44)$$

Si $S = H H^t$ (3.45)

Y enfatizando a: $H_k H_k^t \cong r_0 I$ (3.46)

Donde $r_0 = \sum_{i=0}^{L-1} h^2(i)$

entonces $b^{(k)} = \frac{1}{r_0} rS\, M_k^t$ (3.47)

Es simplemente una versión a escala de la señal residual r ya que r_0, S y M_k^t son constantes lo cual simplifica dramáticamente el cálculo de las rejillas de excitación.

De la misma forma, la selección de la rejilla basada en la minimización del error de aproximación de la ec.(3.39), se simplifica al sustituir la ec.(3.43) en la ec.(3.37).

$$E^{(k)} = [rH - b^{(k)} H_k][H^t r^t - H_k^t b^{(k)t}]$$

$$E^{(k)} = rH\, H^t r^t - rH\, H_k^t b^{(k)t} - b^{(k)} H_k H^t r^t + b^{(k)} H_k H_k^t b^{(k)t}$$

pero $H_k^t = H^t M_k^t \qquad H_k = M_k H$

$$E^{(k)} = rH\, H^t r^t - rH\, H^t M_k^t b^{(k)t} - b^{(k)} M_k H\, H^t r^t + b^{(k)} H_k H_k^t b^{(k)t}$$

de la ec.(3.44) $\qquad rH\, H^t M_k^t = b^{(k)} H_k H_k^t$

$$E^{(k)} = rH\, H^t r^t - b^{(k)} H_k H_k^t b^{(k)t} - b^{(k)} M_k H\, H^t r^t + b^{(k)} H_k H_k^t b^{(k)t}$$

$$E^{(k)} = rH\, H^t r^t - b^{(k)} M_k H\, H^t r^t$$

De la transpuesta de la ec.(3.44) $\qquad M_k H\, H^t r^t = H_k H_k^t b^{(k)t}$

$$E^{(k)} = rH\, H^t r^t - r_0 b^{(k)} b^{(k)t} \qquad (3.48)$$

De aquí:

$$min\{E^{(k)}\} = max\{b^{(k)} b^{(k)t}\} \qquad (3.49)$$

De esta forma el procedimiento anterior es ahora extremadamente simple.

3.4 LPC de Excitación Residual con Vector de Cuantización (RELP-VQ).

La digitalización de radio móvil presenta gran demanda en los componentes de sistemas, para este digitalizador de voz el porcentaje de datos usados para la transmisión de voz determina ampliamente los anchos de banda de frecuencias de los canales de voz y por consecuencia la economización de frecuencias del sistema de entrada.

Como una consecuencia, el porcentaje de datos se debe de conservar tan bajo como sea posible. En suma, es necesario usar códigos de protección de errores apropiados para compensar suficientemente las influencias de disturbios del canal de radio en la calidad de voz. Con ambos

requerimientos resulta un codificador de estructura compleja con diferentes bloques funcionales complementándose unos con otros.

La estructura que a continuación se describe combina un algorítmo de codificación óptimo basado en el vector de cuantización (VQ) con un procedimiento multi-pulso modificado.

Estructura del Codec RELP-VQ.

Todos los sistemas que usan predicción lineal para procesamiento de señales de voz se basan en la suposición de que la generación de voz se aproxima mediante un modelo en la cual se correla un tren de pulsos o ruido descorrelado por medio de un filtro autoregresivo de orden limitado llamado tracto vocal. Tanto la señal de excitación como el tracto vocal se modela por medio de un proceso autoregresivo, si bien se tienen propiedades de variación en el tiempo durante intervalos del orden de 10-30 ms, estos se consideran como invariantes en el tiempo. Durante la codificación de fuente es posible medir las correlaciones de corto tiempo con estos intervalos del filtro autoregresivo de la señal de voz. Un ajuste apropiado del filtro inverso permitirá que los componentes del modelo descrito se puedan separar.

En el caso de la estructura de excitación residual (RELP), la señal residual se modela como una función en el tiempo mediante una cuantificación y codificación óptima.

Para la transmisión, las tramas(frames) se conforman conteniendo los parámetros del filtro y la señal residual codificada.

La estructura RELP ofrece la ventaja de que los componentes de información que se transmitirán se pueden codificar en forma separada de una manera óptima. Estos componentes de información son:

-Coeficientes del filtro utilizados para describir el filtro de análisis y síntesis.

-La ganancia que controla la potencia de la señal residual.

-La posición de rejilla del re-muestreo adaptivo.

La fig.(3.8) muestra el diagrama de bloque de la estructura del codec RELP-VQ.

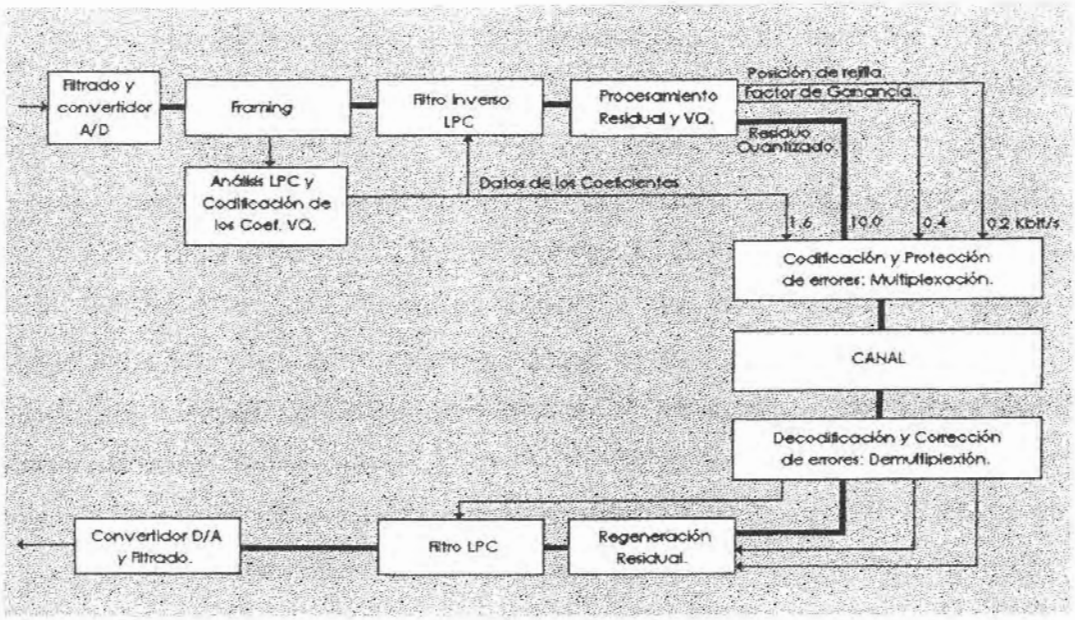


Fig. 3.8 Diagrama de bloque del codificador RELP-VQ

Vector de Cuantización (VQ).

El vector de cuantización (VQ) se aplica a señales discretas en el tiempo (secuencias de muestras) y otro conjunto de parámetros tales como los coeficientes del filtro de predicción. Ahora mencionaremos el principio y definición del (VQ).

Se pueden usar las correlaciones lineales y no lineales con una secuencia de muestras para la reducción de datos mediante la cuantización de bloque por bloque realizando una codificación óptima. Para este propósito, las muestras consecutivas k de una fuente de señal discreta en el tiempo se combina en bloques o vectores $X = (X_1, \cdots, X_k)$ y se codifica por medio de un vector de cuantización. El vector cuantizador se define como sigue:

Un cuantizador de nivel N, K dimensional se describe por medio de un mapa Q, la cual asigna a cada vector de entrada $X = (X_1, \cdots, X_k)$ un vector $Y = (Y_1, \cdots, Y_k)$ tomándolo de un alfabeto de producción finito C, por medio del alfabeto de producción o libro de código $C = (Y(n); n = 1, \cdots, N)$ y la partición $S = (S(n); n = 1, \cdots, N)$ del espacio del vector de entrada. A causa de Q, el mapa cuantizador de cada vector de entrada X dentro de un vector del libro de código y de tal forma que el error de cuantización evaluado por medio de una medida de distancia no negativa es mínima.

La siguiente fórmula aplica a:

$$Y = \min_{d(x,y)} Q(x)$$

La reducción del porcentaje de datos por medio del vector cuantizador se lleva a cabo informando al receptor únicamente la dirección del vector del libro de código de dimensión K en lugar del mismo vector. En el receptor el cual almacena el mismo libro de código como en el transmisor, el correspondiente vector del libro de códigos se sustituye por la dirección del libro de código actualizando la tabla.

El mapa Q se implementa por medio de un algorítmo de búsqueda.

Dado que las actividades de búsqueda y almacenamiento incrementa el tamaño del libro de código, es necesario encontrar una estructura del libro de código apropiada permitiendo así una selección y almacenamiento eficiente de los vectores del libro de código. Si un vector de entrada es comparado con todos los N vectores del Codebook, por ejemplo, como en el caso del método Full Search (FS), se requieren muchas de las actividades de búsqueda.

Esto se puede reducir mediante la generación de un libro de código con estructura de árbol multinivel. En varios pasos consecutivos (Tree Search, TS), únicamente los vectores elegidos en los nodos se usan para la comparación.

La reducción del proceso de búsqueda, sin embargo, se acompaña incrementando el volúmen de almacenamiento. En contraste, con el procedimiento FS los vectores de todos los niveles se tienen que almacenar.

Como se muestra por ejemplo, en la estructura del codebook primitivo representa un buen compromiso entre la calidad lograda y los requerimientos de almacenamiento. Para más información consultar[13].

[13] Gerhard, SCHRODER, " Residual-Excited LPC with Vector Quantization (RELP-VQ)", Speech Communication 7(1988), pag. 227-237.

Capítulo 4

SINTETIZADOR RPE-LTP.

4.1 Introducción.

Si bien es cierto, la codificación desarrollada en la primera parte del proyecto[5] donde se desarrolló el algorítmo de codificación de voz y siguiendo las normas de estandarización del GSM, la tasa de bit es de 16 Kbit/s utilizando tramas de longitud $T_o = 20\ ms$. En esta sección se llevó a cabo la segmentación de las muestras de la señal de voz, incluyendo el ventaneo y el proceso de normalización de la señal.

El analizador tomó muestras de voz a partir del cual extrajo la siguiente información:

1.- La información espectral contenida en la señal de voz la cual se cuantiza mediante el análisis de predicción lineal resultando por tanto un conjunto de parámetros del predictor.

---

[5] Tesis, "Codificador de voz a 16 Kbit/s usando la Técnica RPE-LTP", Primera parte del proyecto global, trabajo elaborado por V. C. Alberto, Universidad Tecnológica de la Mixteca, 1997.

2.- Información(LTP) Long Term Prediction (retardo y ganancia) describiendo la correlación de término largo de las muestras de voz.

3.- La información de excitación, mostrada por las posiciones de los pulsos y,

4.- Las amplitudes.

En la primera parte del proyecto toda esta información fué cuantizada, multiplexada y transmitida al sintetizador(trabajo a nuestra atención) por lo cual realizamos las siguientes operaciones:

1.- Decodificación de los parámetros para el sintetizador.
- LAR codificados, LARc.
- Retardo de correlación, Nc.
- Factor de ganancia, bc.
- Selección de rejilla RPE, M.
- Amplitud máxima codificada, xmaxc.
- Muestras RPE normalizadas, xMc.

2.- Reconstrucción del seleccionador de rejilla RPE.

3.- Interpolación de LAR y transformación de LAR a coeficientes de reflexión.

4.- Filtrado de Término Largo.

5.- Filtrado de Término Corto resultando la señal de voz reconstruida.

6.- Colocación del pseudocódigo de sintetizador.

La fig.(4.1) muestra el diagrama a bloques del decodificador RPE-LTP.[14]

[14] P. Vary, R.J. Sluyter, C. Galand and M. Rosso, "RPE-LTP Codec - The Candidate for the GSM Radio Communication System", *Internat. Conf. on Digital Land Mobile Radio Communications*, 30 june-3 july, Venice, pp.507-516 (1987).

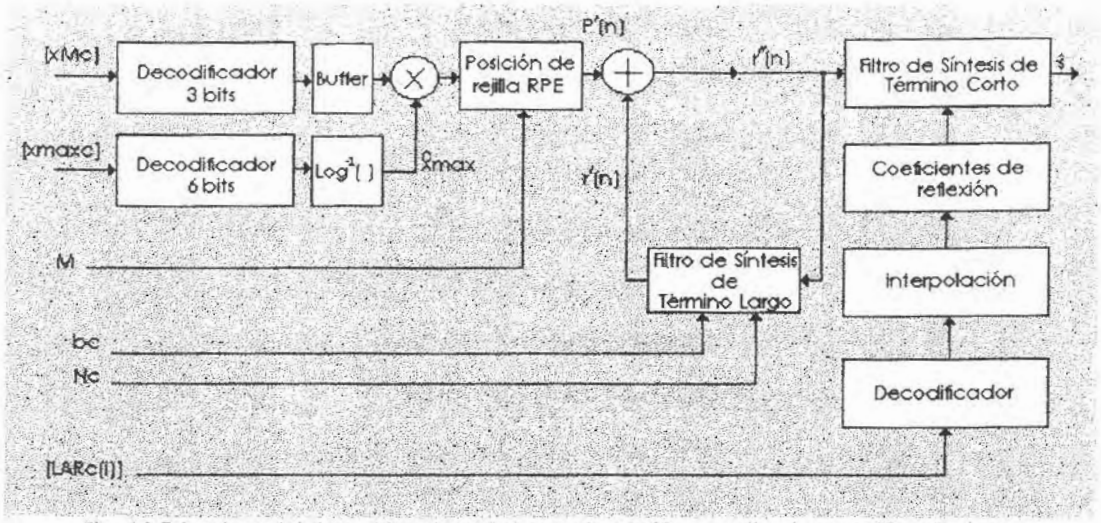


Fig. 4.1 Estructura del decodificador: [x] denota la versión cuantizada y codificada de x.

En base a la figura anterior se decodifican los parámetros recibidos en el sintetizador.

4.2 Decodificación de parámetros para el sintetizador.

Una vez recibida la trama de 260 bits(sin código de protección contra errores) y habiéndo logrado una tasa de transmisión neta de 13 Kbit/s, cada 20 ms. de la señal de voz(equivalente a 160 muestras tomadas cada 125 μs.) esta trama viene codificada de manera siguiente:

BIT	SIGNIFICADO.
0	Parámetro LAR 1, bit 0.
1	Parámetro LAR 1, bit 1.
2	Parámetro LAR 1, bit 2.
3	Parámetro LAR 1, bit 3.
4	Parámetro LAR 1, bit 4.
5	Parámetro LAR 1, bit 5.
6	Parámetro LAR 2, bit 0.
7	Parámetro LAR 2, bit 1.

8	Parámetro LAR 2, bit 2.
9	Parámetro LAR 2, bit 3.
10	Parámetro LAR 2, bit 4.
11	Parámetro LAR 2, bit 5.
12	Parámetro LAR 3, bit 0.
13	Parámetro LAR 3, bit 1.
14	Parámetro LAR 3, bit 2.
15	Parámetro LAR 3, bit 3.
16	Parámetro LAR 3, bit 4.
17	Parámetro LAR 4, bit 0.
18	Parámetro LAR 4, bit 1.
19	Parámetro LAR 4, bit 2.
20	Parámetro LAR 4, bit 3.
21	Parámetro LAR 4, bit 4.
22	Parámetro LAR 5, bit 0.
23	Parámetro LAR 5, bit 1.
24	Parámetro LAR 5, bit 2.
25	Parámetro LAR 5, bit 3.
26	Parámetro LAR 6, bit 0.
27	Parámetro LAR 6, bit 1.
28	Parámetro LAR 6, bit 2.
29	Parámetro LAR 6, bit 3.
30	Parámetro LAR 7, bit 0.
31	Parámetro LAR 7, bit 1.
32	Parámetro LAR 7, bit 2.
33	Parámetro LAR 8, bit 0.
34	Parámetro LAR 8, bit 1.
35	Parámetro LAR 8, bit 2.
36	Retardo de correlación 0, bit 0.
37	Retardo de correlación 0, bit 1.
38	Retardo de correlación 0, bit 2.
39	Retardo de correlación 0, bit 3.

40	Retardo de correlación 0, bit 4.
41	Retardo de correlación 0, bit 5.
42	Retardo de correlación 0, bit 6.
43	Factor de ganancia 0, bit 0.
44	Factor de ganancia 0, bit 1.
45	Fase de rejilla 0, bit 0.
46	Fase de rejilla 0, bit 1.
47	Pulso máximo de rejilla 0, bit 0.
48	Pulso máximo de rejilla 0, bit 1.
49	Pulso máximo de rejilla 0, bit 2.
50	Pulso máximo de rejilla 0, bit 3.
51	Pulso máximo de rejilla 0, bit 4.
52	Pulso máximo de rejilla 0, bit 5.
53	Pulso 0, rejilla 0, bit 0.
54	Pulso 0, rejilla 0, bit 1.
55	Pulso 0, rejilla 0, bit 2.
56	Pulso 1, rejilla 0, bit 0.
57	Pulso 1, rejilla 0, bit 1.
58	Pulso 1, rejilla 0, bit 2.
59	Pulso 2, rejilla 0, bit 0.
60	Pulso 2, rejilla 0, bit 1.
61	Pulso 2, rejilla 0, bit 2.
62	Pulso 3, rejilla 0, bit 0.
63	Pulso 3, rejilla 0, bit 1.
64	Pulso 3, rejilla 0, bit 2.
65	Pulso 4, rejilla 0, bit 0.
66	Pulso 4, rejilla 0, bit 1.
67	Pulso 4, rejilla 0, bit 2.
68	Pulso 5, rejilla 0, bit 0.
69	Pulso 5, rejilla 0, bit 1.
70	Pulso 5, rejilla 0, bit 2.
71	Pulso 6, rejilla 0, bit 0.

72	Pulso 6, rejilla 0, bit 1.
73	Pulso 6, rejilla 0, bit 2.
74	Pulso 7, rejilla 0, bit 0.
75	Pulso 7, rejilla 0, bit 1.
76	Pulso 7, rejilla 0, bit 2.
77	Pulso 8, rejilla 0, bit 0.
78	Pulso 8, rejilla 0, bit 1.
79	Pulso 8, rejilla 0, bit 2.
80	Pulso 9, rejilla 0, bit 0.
81	Pulso 9, rejilla 0, bit 1.
82	Pulso 9, rejilla 0, bit 2.
83	Pulso 10, rejilla 0, bit 0.
84	Pulso 10, rejilla 0, bit 1.
85	Pulso 10, rejilla 0, bit 2.
86	Pulso 11, rejilla 0, bit 0.
87	Pulso 11, rejilla 0, bit 1.
88	Pulso 11, rejilla 0, bit 2.
89	Pulso 12, rejilla 0, bit 0.
90	Pulso 12, rejilla 0, bit 1.
91	Pulso 12, rejilla 0, bit 2.
92	Retardo de correlación 1, bit 0.
93	Retardo de correlación 1, bit 1.
94	Retardo de correlación 1, bit 2.
95	Retardo de correlación 1, bit 3.
96	Retardo de correlación 1, bit 4.
97	Retardo de correlación 1, bit 5.
98	Retardo de correlación 1, bit 6.
99	Factor de ganancia 1, bit 0.
100	Factor de ganancia 1, bit 1.
101	Fase de rejilla 1, bit 0.
102	Fase de rejilla 1, bit 1.
103	Pulso máximo de rejilla 1, bit 0.

104	Pulso máximo de rejilla 1, bit 1.
105	Pulso máximo de rejilla 1, bit 2.
106	Pulso máximo de rejilla 1, bit 3.
107	Pulso máximo de rejilla 1, bit 4.
108	Pulso máximo de rejilla 1, bit 5.
109	Pulso 0, rejilla 1, bit 0.
110	Pulso 0, rejilla 1, bit 1.
111	Pulso 0, rejilla 1, bit 2.
112	Pulso 1, rejilla 1, bit 0.
113	Pulso 1, rejilla 1, bit 1.
114	Pulso 1, rejilla 1, bit 2.
115	Pulso 2, rejilla 1, bit 0.
116	Pulso 2, rejilla 1, bit 1.
117	Pulso 2, rejilla 1, bit 2.
118	Pulso 3, rejilla 1, bit 0.
119	Pulso 3, rejilla 1, bit 1.
120	Pulso 3, rejilla 1, bit 2.
121	Pulso 4, rejilla 1, bit 0.
122	Pulso 4, rejilla 1, bit 1.
123	Pulso 4, rejilla 1, bit 2.
124	Pulso 5, rejilla 1, bit 0.
125	Pulso 5, rejilla 1, bit 1.
126	Pulso 5, rejilla 1, bit 2.
127	Pulso 6, rejilla 1, bit 0.
128	Pulso 6, rejilla 1, bit 1.
129	Pulso 6, rejilla 1, bit 2.
130	Pulso 7, rejilla 1, bit 0.
131	Pulso 7, rejilla 1, bit 1.
132	Pulso 7, rejilla 1, bit 2.
133	Pulso 8, rejilla 1, bit 0.
134	Pulso 8, rejilla 1, bit 1.
135	Pulso 8, rejilla 1, bit 2.

136	Pulso 9, rejilla 1, bit 0.
137	Pulso 9, rejilla 1, bit 1.
138	Pulso 9, rejilla 1, bit 2.
139	Pulso 10, rejilla 1, bit 0.
140	Pulso 10, rejilla 1, bit 1.
141	Pulso 10, rejilla 1, bit 2.
142	Pulso 11, rejilla 1, bit 0.
143	Pulso 11, rejilla 1, bit 1.
144	Pulso 11, rejilla 1, bit 2.
145	Pulso 12, rejilla 1, bit 0.
146	Pulso 12, rejilla 1, bit 1.
147	Pulso 12, rejilla 1, bit 2.
148	Retardo de correlación 2, bit 0.
149	Retardo de correlación 2, bit 1.
150	Retardo de correlación 2, bit 2.
151	Retardo de correlación 2, bit 3.
152	Retardo de correlación 2, bit 4.
153	Retardo de correlación 2, bit 5.
154	Retardo de correlación 2, bit 6.
155	Factor de ganancia 2, bit 0.
156	Factor de ganancia 2, bit 1.
157	Fase de rejilla 2, bit 0.
158	Fase de rejilla 2, bit 1.
159	Pulso máximo de rejilla 2, bit 0.
160	Pulso máximo de rejilla 2, bit 1.
161	Pulso máximo de rejilla 2, bit 2.
162	Pulso máximo de rejilla 2, bit 3.
163	Pulso máximo de rejilla 2, bit 4.
164	Pulso máximo de rejilla 2, bit 5.
165	Pulso 0, rejilla 2, bit 0.
166	Pulso 0, rejilla 2, bit 1.
167	Pulso 0, rejilla 2, bit 2.

168	Pulso 1, rejilla 2, bit 0.
169	Pulso 1, rejilla 2, bit 1.
170	Pulso 1, rejilla 2, bit 2.
171	Pulso 2, rejilla 2, bit 0.
172	Pulso 2, rejilla 2, bit 1.
173	Pulso 2, rejilla 2, bit 2.
174	Pulso 3, rejilla 2, bit 0.
175	Pulso 3, rejilla 2, bit 1.
176	Pulso 3, rejilla 2, bit 2.
177	Pulso 4, rejilla 2, bit 0.
178	Pulso 4, rejilla 2, bit 1.
179	Pulso 4, rejilla 2, bit 2.
180	Pulso 5, rejilla 2, bit 0.
181	Pulso 5, rejilla 2, bit 1.
182	Pulso 5, rejilla 2, bit 2.
183	Pulso 6, rejilla 2, bit 0.
184	Pulso 6, rejilla 2, bit 1.
185	Pulso 6, rejilla 2, bit 2.
186	Pulso 7, rejilla 2, bit 0.
187	Pulso 7, rejilla 2, bit 1.
188	Pulso 7, rejilla 2, bit 2.
189	Pulso 8, rejilla 2, bit 0.
190	Pulso 8, rejilla 2, bit 1.
191	Pulso 8, rejilla 2, bit 2.
192	Pulso 9, rejilla 2, bit 0.
193	Pulso 9, rejilla 2, bit 1.
194	Pulso 9, rejilla 2, bit 2.
195	Pulso 10, rejilla 2, bit 0.
196	Pulso 10, rejilla 2, bit 1.
197	Pulso 10, rejilla 2, bit 2.
198	Pulso 11, rejilla 2, bit 0.
199	Pulso 11, rejilla 2, bit 1.

200	Pulso 11, rejilla 2, bit 2.
201	Pulso 12, rejilla 2, bit 0.
202	Pulso 12, rejilla 2, bit 1.
203	Pulso 12, rejilla 2, bit 2.
204	Retardo de correlación 3, bit 0.
205	Retardo de correlación 3, bit 1.
206	Retardo de correlación 3, bit 2.
207	Retardo de correlación 3, bit 3.
208	Retardo de correlación 3, bit 4.
209	Retardo de correlación 3, bit 5.
210	Retardo de correlación 3, bit 6.
211	Factor de ganancia 3, bit 0.
212	Factor de ganancia 3, bit 1.
213	Fase de rejilla 3, bit 0.
214	Fase de rejilla 3, bit 1.
215	Pulso máximo de rejilla 3, bit 0.
216	Pulso máximo de rejilla 3, bit 1.
217	Pulso máximo de rejilla 3, bit 2.
218	Pulso máximo de rejilla 3, bit 3.
219	Pulso máximo de rejilla 3, bit 4.
220	Pulso máximo de rejilla 3, bit 5.
221	Pulso 0, rejilla 3, bit 0.
222	Pulso 0, rejilla 3, bit 1.
223	Pulso 0, rejilla 3, bit 2.
224	Pulso 1, rejilla 3, bit 0.
225	Pulso 1, rejilla 3, bit 1.
226	Pulso 1, rejilla 3, bit 2.
227	Pulso 2, rejilla 3, bit 0.
228	Pulso 2, rejilla 3, bit 1.
229	Pulso 2, rejilla 3, bit 2.
230	Pulso 3, rejilla 3, bit 0.
231	Pulso 3, rejilla 3, bit 1.

232	Pulso 3, rejilla 3, bit 2.
233	Pulso 4, rejilla 3, bit 0.
234	Pulso 4, rejilla 3, bit 1.
235	Pulso 4, rejilla 3, bit 2.
236	Pulso 5, rejilla 3, bit 0.
237	Pulso 5, rejilla 3, bit 1.
238	Pulso 5, rejilla 3, bit 2.
239	Pulso 6, rejilla 3, bit 0.
240	Pulso 6, rejilla 3, bit 1.
241	Pulso 6, rejilla 3, bit 2.
242	Pulso 7, rejilla 3, bit 0.
243	Pulso 7, rejilla 3, bit 1.
244	Pulso 7, rejilla 3, bit 2.
245	Pulso 8, rejilla 3, bit 0.
246	Pulso 8, rejilla 3, bit 1.
247	Pulso 8, rejilla 3, bit 2.
248	Pulso 9, rejilla 3, bit 0.
249	Pulso 9, rejilla 3, bit 1.
250	Pulso 9, rejilla 3, bit 2.
251	Pulso 10, rejilla 3, bit 0.
252	Pulso 10, rejilla 3, bit 1.
253	Pulso 10, rejilla 3, bit 2.
254	Pulso 11, rejilla 3, bit 0.
255	Pulso 11, rejilla 3, bit 1.
256	Pulso 11, rejilla 3, bit 2.
257	Pulso 12, rejilla 3, bit 0.
258	Pulso 12, rejilla 3, bit 1.
259	Pulso 12, rejilla 3, bit 2.

Ahora bien, organizando estos parámetros nos quedan de la siguiente manera:

PARAMETRO.	No. DE BITS.
LAR 1.	6 bits.
LAR 2.	6 bits.
LAR 3.	5 bits.
LAR 4.	5 bits.
LAR 5.	4 bits.
LAR 6.	4 bits.
LAR 7.	3 bits.
LAR 8.	3 bits.
	36 bits.

PARAMETRO.	No. DE BITS.
Retardo de correlación 0.	7 bits.
Retardo de correlación 1.	7 bits.
Retardo de correlación 2.	7 bits.
Retardo de correlación 3.	7 bits.
	28 bits.

PARAMETRO.	No. DE BITS.
Factor de ganancia 0.	2 bits.
Factor de ganancia 1.	2 bits.
Factor de ganancia 2.	2 bits.
Factor de ganancia 3.	2 bits.
	8 bits.

PARAMETRO.	No. DE BITS.
Fase de rejilla 0.	2 bits.
Fase de rejilla 1.	2 bits.
Fase de rejilla 2.	2 bits.
Fase de rejilla 3.	2 bits.
	8 bits.

PARAMETRO.	No. DE BITS.
Pulso máximo de rejilla 0.	6 bits.
Pulso máximo de rejilla 1.	6 bits.
Pulso máximo de rejilla 2.	6 bits.
Pulso máximo de rejilla 3.	6 bits.
	24 bits.

PARAMETRO	BITS
Pulso 0, rejilla 0.	3 bits.
Pulso 1, rejilla 0.	3 bits.
Pulso 2, rejilla 0.	3 bits.
Pulso 3, rejilla 0.	3 bits.
Pulso 4, rejilla 0.	3 bits.
Pulso 5, rejilla 0.	3 bits.
Pulso 6, rejilla 0.	3 bits.
Pulso 7, rejilla 0.	3 bits.
Pulso 8, rejilla 0.	3 bits.
Pulso 9, rejilla 0.	3 bits.
Pulso 10, rejilla 0.	3 bits.
Pulso 11, rejilla 0.	3 bits.
Pulso 12, rejilla 0.	3 bits.
	39 bits.

PARAMETRO	BITS
Pulso 0, rejilla 1.	3 bits.
Pulso 1, rejilla 1.	3 bits.
Pulso 2, rejilla 1.	3 bits.
Pulso 3, rejilla 1.	3 bits.
Pulso 4, rejilla 1.	3 bits.
Pulso 5, rejilla 1.	3 bits.
Pulso 6, rejilla 1.	3 bits.
Pulso 7, rejilla 1.	3 bits.
Pulso 8, rejilla 1.	3 bits.
Pulso 9, rejilla 1.	3 bits.
Pulso 10, rejilla 1.	3 bits.
Pulso 11, rejilla 1.	3 bits.
Pulso 12, rejilla 1.	3 bits.
	39 bits.

PARAMETRO	BITS
Pulso 0, rejilla 2.	3 bits.
Pulso 1, rejilla 2.	3 bits.
Pulso 2, rejilla 2.	3 bits.
Pulso 3, rejilla 2.	3 bits.
Pulso 4, rejilla 2.	3 bits.
Pulso 5, rejilla 2.	3 bits.
Pulso 6, rejilla 2.	3 bits.
Pulso 7, rejilla 2.	3 bits.
Pulso 8, rejilla 2.	3 bits.
Pulso 9, rejilla 2.	3 bits.
Pulso 10, rejilla 2.	3 bits.
Pulso 11, rejilla 2.	3 bits.
Pulso 12, rejilla 2.	3 bits.
	39 bits.

PARAMETRO	BITS
Pulso 0, rejilla 3.	3 bits.
Pulso 1, rejilla 3.	3 bits.
Pulso 2, rejilla 3.	3 bits.
Pulso 3, rejilla 3.	3 bits.
Pulso 4, rejilla 3.	3 bits.
Pulso 5, rejilla 3.	3 bits.
Pulso 6, rejilla 3.	3 bits.
Pulso 7, rejilla 3.	3 bits.
Pulso 8, rejilla 3.	3 bits.
Pulso 9, rejilla 3.	3 bits.
Pulso 10, rejilla 3.	3 bits.
Pulso 11, rejilla 3.	3 bits.
Pulso 12, rejilla 3.	3 bits.
	39 bits.

por lo cual, resumimos estos datos en la tabla 4-1 que nos representa la distribución de los bits por cada trama recibida.

Distribución de bits por trama		
8	Parámetros LAR.	36 bits.
4	Retardo de correlación.	28 bits.
4	Factor de ganancia.	8 bits.
4	Fase de rejilla.	8 bits.
4	Pulso máximo de rejilla.	24 bits.
52	Muestras RPE.	156 bits.
*.	Protección contra errores.	60 bits.
	Número de bits por trama.	320 bits.
	Tasa Neta.	13.0 Kbps.
	Tasa Total.	16.0 Kbps.

Tabla 4-1. Distribución de bits para cada trama.
* La protección contra errores no esta presente en el proyecto.

4.2.1 LAR(Razones de Area Logarítmica) codificados, LARc.

Para la codificación de los parámetros LAR tenemos el antecedente de que los coeficientes de correlación *r(i)* fueron transformados a parámetros LAR con la finalidad de poder hacer una reducción considerable de la sensibilidad a los efectos de cuantización.

Por lo cual, los LAR están definidos como sigue:

$$LAR(i) = \log_{10}\left(\frac{1+r(i)}{1-r(i)}\right)$$

donde *r(i)* son los coeficientes de reflexión.

Dado que las características de compansión de esta transformación es de interés, se utilizó la siguiente aproximación con 5 segmentos lineales:

$$LAR(i) = \begin{cases} r(i), & |r(i)| < 0.675, \\ sign[r(i)] * [2|r(i)| - 0.675], & \\ & 0.675 \le |r(i)| < 0.950, \\ sign[r(i)] * [8|r(i)| - 6.375] & \\ & 0.950 \le |r(i)| < 1.000 \end{cases}$$

y como los coeficientes LAR tienen diferentes rangos dinámicos y diferentes distribuciones de amplitudes asimétricas $P_i(x)$. Por esta razón los coeficientes LAR se cuantizan con distintos cuantizadores uniformes como lo muestra la tabla 4-2.

No. LAR	MINIMO	MAXIMO	STEP SIZE	BITS/LAR
1-2	-1.60	1.55	0.05	6
3	-1.00	0.55	0.05	5
4	-0.55	1.00	0.05	5
5-6	-0.70	0.80	0.10	4
7-8	-0.30	0.40	0.10	3

TABLA 4-2. Cuantización de los parámetros LAR.

Por lo tanto, en base a esta forma de cuantización y codificación se decodifican dichos parámetros para restaurarlos a sus valores previos.

4.2.2 Retardo de correlación, Nc.

El segundo parámetro recibido desde el analizador es el retardo de correlación representado por $M(k)(k=0,..., 3)$ que tomó valores en el rango de $(40, ..., 120)$ y se codificó usando 7 bits.

La expresión utilizada para el cálculo de este parámetro fué:

$$M(k) = L.$$
$$Nc'(k) = M(k) - 40. \qquad k = 0, ..., 3.$$

donde L : Es la posición del pico de función de correlación cruzada en el intervalo de 40 a 120.

$Nc'(k)$: Es el retardo de correlación codificada.

Por lo tanto, en el sintetizador asumiendo que la transmisión estuvo libre de error, se decodifica este parámetro guardando el retardo actual definido en la siguiente expresión:

$$Nc(k) = Nc'(k) + 40, \qquad k = 0, ..., 3.$$

4.2.3 Factor de ganancia, bc.

Al igual que el retardo de correlación se recibió el factor de ganancia de predicción de término largo codificado y denotado por *bc(i)(i=0, ..., 3)* que fué codificado usando 2 bits para cada uno. Tomando el nivel de decisión y cuantización que proporciona el cuantizador de ganancia por lo consiguiente, se decodifica este parámetro en base a la tabla 4-3.

	Nivel de decisión	Nivel de cuantización
i	DLB(i)	QLB(i)
0	0.2	0.10
1	0.5	0.35
2	0.8	0.65
3	1.0	1.00

Tabla 4-3. Tabla de cuantización para la ganancia LTP.

4.2.4 Selección de rejilla, M.

El parámetro selección de rejilla RPE, representado por M también se decodifica, una vez que fué codificado con 2 bits y transmitido. Este parámetro representa cual de las 3 posibles rejillas fué seleccionada; donde M nos representa la fase marcada como (0, 1, ó 2).

4.2.5 Amplitud máxima codificada, xmaxc.

Este parámetro se determinó en base a la relación siguiente.

$$xmaxc = \max_{m} \sum_{i=1}^{\frac{1}{12}k} X_m^2(I), \qquad m=0,\ 1,\ 2.$$

por lo cual nos representa el pulso máximo de rejilla de la trama analizada. Ahora se decodifica con 7 bits.

4.2.6 Muestras RPE normalizadas, xMc.

El último parámetro se decodifica conteniéndo 52 muestras la cual fué definido en la primera parte del proyecto como:

$$xMc(l) = x(k_0 + m + 3l)$$

donde $l = 0,1,\ldots,\frac{1}{12}k,$ $m = 0, 1, 2.$

k_0 define el inicio de la trama actual y,

m denota la fase de la decimación de la rejilla y corresponde a las $\frac{1}{4}k$ muestras.

Esta decodificación se realiza con 3 bits para cada muestra.

4.3 Reconstrucción del seleccionador de rejilla RPE.

En este bloque los pulsos RPE se decodifican y desnormalizan, luego la tasa de muestreo se incrementa por un factor de 3 en el posicionador de secuencias insertando dos muestras de amplitud cero y colocando los pulsos en la posición temporal de rejilla M correcta.

4.4 Interpolación de LAR y Transformación de LAR a PARCOR.

Una vez que los parámetros LAR se decodifican pasan a un proceso de interpolación lineal ya que pueden ocurrir transitorios no deseados debido a que los coeficientes del filtro de síntesis de término corto pudieran cambiar abruptamente; esta interpolación lineal se realiza en la transición entre dos conjuntos de parámetros sucesivos.

Después de este proceso, los parámetros LAR interpolados son transformados ahora en coeficientes de reflexión para derivar los coeficientes $a_i(i = 0,\ldots,8)$ que caracterizaran al filtro de síntesis de término corto.

4.5 Filtro de Síntesis de Término Largo.

Una vez obtenida la señal de excitación $p'(n)(n = 1, ..., 160)$ estas son procesadas por medio de sub-bloques de 40 muestas. Para cada sub-bloque de muestras de excitación denotadas por $p'(n)(n = 1, ..., 40)$ se agrega una estimación $r'(n)(n = 1, ..., 40)$ de la señal para dar la señal residual reconstruida $r''(n)(n = 1, ..., 40)$.

$$r''(n) = r'(n) + p'(n). \qquad n = 1, ..., 40.$$

Las muestras estimadas $r'(n)$ son calculadas de las muestras residuales reconstruidas previamente $r''(n)$ con un filtro combinado ajustando el retardo de correlación $Nc(k)$ LTP y la ganancia $bc(k)$ LTP del sub-bloque actual.

$$r'(n) = bc(k) * r''(n - Nc(k)). \qquad n = 1, ..., 40.$$

4.6 Filtro de Síntesis de Término Corto.

Finalmente, los coeficientes de forma directa *a(i)* junto con la señal residual reconstruida *r"(n)* son utilizados en el filtro que modela el tracto vocal *(1/A(z))* para generar las muestras de voz reconstruidas *r"(n)* de acuerdo a la siguiente relación:

$$s'(n) = r''(n) - \sum_{i=1}^{8} a(i)s'(n-i)$$

$n=1,...,160$

La última tarea de procesamiento digital consiste en grabar las muestras así como obtener sus valores en 16 bits y colocarle su cabecera tipo RIFF al archivo de datos generado para completar finalmente el archivo de voz reconstruida.

La tasa de bit de 16 Kbit/s correspondiente a 320 bits disponibles para cada bloque de 20 ms. Estos bits estan distribuidos para la diferente información transmitida de acuerdo a la distribución mostrada previamente en la tabla 4-2.

4.7 Pseudocódigo del Sintetizador.

4.7.1 DESCRIPCION DE VARIABLES UTILIZADAS EN EL DECODIFICADOR.

```
*S          /*[0..159]   Muestras                          IN*/
*LARc       /*[0..7]     Coeficientes LAR                  OUT */
*Nc         /*[0..3]     Orden del LTP                     OUT */
*bc         /*[0..3]     Ganancia LTP codificada           OUT */
*Mc         /*[0..3]     Selección de rejilla RPE          OUT */
*xmaxc      /*[0..3]     Amplitud máxima codificada        OUT */
*xmc        /*[13*4]     Muestras RPE normalizadas OUT */
*d          /*[0..159]    Señal residual                   IN/OUT */
*LARcr      /*[0..7]                                       IN */
*Ncr        /*[0..3]                                       IN */
*bcr        /*[0..3]                                       IN */
*Mcr        /*[0..3]                                       IN */
*xmaxcr     /*[0..3]                                       IN */
*xMcr       /*[0..13*4]                                    IN */
*s          /*[0..159]                                     OUT */
nrp         /*40 */       /* long_term.c, syntesis     */
v[9]        /*  short_term.c, Syntesis                    */
msr         /*  decoder.c,  Postprocesing                 */
```

Nota: *Toda variable seguida del caracter " * " representa un puntero.*

** es un operador monario que devuelve el valor de la variable situada en la dirección de memoria de la variable.*

PSEUDOCODIGO DEL PROGRAMA PRINCIPAL

```
Inicio
      Definir(variables)
      Leer(archivo_de_entrada)
      fpr ← Abrir_archivo(input_file,"lectura_binaria")
      Si (fpr = NULO) entonces
            Escribir("No se pudo habrir archivo")
            Salir(0)
      fin-si
      Leer(archivo_de_salida)
```

```
        fpa ← abrir_archivo(output_file,"agregar_binario")
        r ← Gsm_create( )          { función que reserva memoria }
        Si ( NOT(r) ) entonces
              Escribir("Error en la asignación de memoria")
              Salir(0)
        fin-si
        CC ← Leer_archivo(s, Tamaño(gsm_byte), BUFSZ, fpr)
        Mientras (CC > 0) hacer
              cont ← cont + 1
              Llamar-a Gsm_decode(r, s, d)
              cd ← Escribir_en_archivo(d, Tamaño(word), BUFSZ, fpa)
              Si (cd <> BUFSZ) entonces
                    Escribir("error de escritura en archivo de salida")
                    Llamar-a Gsm_destroy( r )
                    Regresar (-1)
              fin-si
        fin-mientras
        Cerrar-archivo(fpr)
        Cerrar-archivo(fpa)
        Escribir("Tramas procesadas")
        Llamar-a Gsm_destroy(r)          { función que libera memoria }
        Regresar(0)
fin


Función Gsm_decode(r, s, d)
      Definir(variables)
      CC ← ((*c >> 4) AND 0x0F)
      Si (CC <> GSM_MAGIC) entonces
            Regresar(-1)
      sino
            Asigna-info-a (LARc[0..7])
            Asigna-info-a (Nc[0..3])
            Asigna-info-a (bc[0..3])
            Asigna-info-a (Mc[0..3])
            Asigna-info-a (xmaxc[0..3])
            Asigna-info-a (xmc[0..51])
            Llamar-a Gsm_decoder(s, lARc, Nc, bc, Mc, xmaxc, xmc, target)
            Regresar(0)
      fin-si
fin-función
```

```
Función Gsm_destroy( s )
        Si (S) entonces
                Liberar(S)
        fin-si
fin-función


Función Gsm_decoder(S, LARcr, Ncr, bcr, Mcr, xmaxcr, xMcr, s)
        *drp ← s-> dp0 + 120
        Desde j←0 hasta j<=3 hacer
                Llamar-a Gsm_RPE_Decoding(S,*xmaxcr,Mcr,*xMcr,erp)
                Llamar-a Gsm_Long_Term_Syntesis_Filtering(s,*Ncr,*bcr,
                        erp,drp)
                Desde k←0 hasta k<=39 hacer
                        wt[j*40+k] ← drp[k]
                        k ← k + 1
                fin-desde
                j ← j + 1
                xmaxcr ← xmaxcr + 1
                bcr ← bcr + 1
                Ncr ← Ncr + 1
                Mcr ← Mcr + 1
                xMcr ← xMcr + 13
        fin-desde
        llamar-a Gsm_Short_Term_Syntesis_Filter( S, LARcr, wt, s)
        llamar-a Postprocessing(S, s)
fin-función


Función Postprocessing(S, s)
        msr ← s -> msr
        Desde k←160 hasta k-- hacer
          .     tmp ← GSM_MULT_R(msr, 28180)
                msr ← GSM_ADD(*s, tmp)
                *s ← GSM_ADD(msr, msr) AND OxFFF8
                s ← s + 1
        fin-desde
        s -> msr ← msr
fin-función
```

```
Función Gsm_RPE_Decoding(S, xmaxcr, Mcr, xMcr, erp)
        Llamar-a APCM_quantization_xmaxc_to_exp_mant(xmaxcr, &exp,
        &mant)
        Llamar-a APCM_inverse_quantization(xMcr, mant, exp, xMp)
        Llamar-a RPE_grid_Positioning(Mcr, xMp, erp)
fin-función


Función APCM_quantization_xmaxc_to_exp_mant(xmaxc,exp_out, mant_out)
                { Calcula el exponente y la mantisa de la versión
                decodificada xmaxc  }
        exp ← 0
        Si (xmaxc > 15) entonces
                exp ← SASR(xmaxc, 3) - 1
        sino
                mant ← xmaxc - (exp<<3)
        fin-si
        Si (mant=0) entonces
                exp ← -4
                mant ← 7
        sino
                Mientras (mant <=7) hacer
                        mant ← mant << 1 OR 1
                        exp ← exp - 1
                fin-mientras
                mant ← mant - 8
        fin-si
        assert(exp >=-4 && exp <= 6)
        assert(mant >= 0 && mant <= 7)
        * exp_out ← exp
        *mant_out ← mant
fin-función


Función APCM_inverse_quantization(xMc, mant, exp, xMp)

        {  Este bloque decodifica las muestras de la frecuencia
        xMc[0..12] para obtener el arreglo de la secuencia
        xMp[0..12]  La tabla 4.6 se utiliza para obtener la mantisa
        de xmaxc  FAC[0..7]  }
        assert(mant>=0 && mant<=7)
```

```
        temp1 ← gsm_FAC[mant]
        temp2 ← gsm_sub(6, exp)
        temp3 ← gsm_asl(1, gsm_sub(temp2, 1) )
        Desde i←13 hasta i-- hacer
                assert(*xMc<=7 && xMc>=0)          { 3 bits sin signo }
                temp ← (*xMc++ << 1) - 7           {  restaura el signo }
                assert(temp <= 7 && temp >= -7)    {  4 bits con signo }
                temp ← temp<<12
                temp ← GSM_MULT_R(temp1, temp)
                temp ← GSM_ADD(temp, temp3)
                *xMp++ ← gsm_asr(temp, temp2)
        fin-desde
fin-función


Función RPE_grid_positioning(Mc, xMp, ep)

        { Este procedimiento calcula la señal residual de término largo
        ep[0..39] reconstruida para el filtro de análisis LTP. Las entradas
        son los Mc, los cuales son las posiciones de la rejilla seleccionada
        y las muestras de xMp[0..12] decodificadas son remuestreadas por un
        un factor de 3 para inserción de valores cero.  }

        i ← 13
        assert(0<=Mc && Mc<=3)
        Caso-de (Mc) hacer
                caso 3:  *ep++ ← 0
                caso 2:  Hacer
                                *ep++ ← 0
                                *ep++ ← 0
                                *ep++ ← *xMp++
                         mientras(--i)
                         Romper()
                caso 1:  *ep++ ← 0
                caso 0:  *ep++ ← *xMp++
                         i ← i - 1
                        Hacer
                                *ep++ ← 0
                                *ep++ ← 0
                                *ep++ ← *xMp++
```

```
                mientras(--i)
                Romper()
        fin-caso
        Mientras(++Mc < 4) hacer
                *ep++ ← 0
        fin-mientras
fin-función
```

```
Función Gsm_Long_Term_Syntesis_Filtering(S, *Ncr, *bcr, erp, drp)
        {  Procedimiento que utiliza los parámetros bcr y los Ncr para
      realizar la síntesis en el filtro de término largo.
      El decodificador de parámetros bcr requiere la tabla 4.3b    }

        {  Verifica losl límite de Nr    }
      Si (Ncr<40 OR Ncr>120) entonces
              Nr ← s-> nrp
      sino
              Nr ← Ncr
      fin-si
      s-> nrp ← Nr
      assert(Nr>=40 && Nr<=120)
        {  Decodificación de la ganancia de parámetros bcr del LTP   }
      brp ← gsm_QLB[bcr]
        {  Cálculo de la señal residual de término corto drp[0..39]
           reconstruida   }
      assert(brp <> MIN_WORD)
      Desde k←0 hasta k<=39 hacer
              k ← k + 1
              drpp ← GSM_MULT_R(brp, drp[k-Nr])
              drp[k] ← GSM_ADD(erp[k], drpp)
      fin-desde
        {  Actualización de la señal de término corto drp[-1..-120]
           reconstruida    }
      Desde k←0 hasta k<=119 hacer
              k ← k + 1
              drp[-120+k] ← drp[-80+k]
      fin-desde
fin-función
```

```
Función Gsm_Short_Term_Syntesis_Filter(s, LARcr, wt, s)
        *LARpp_j ← s-> LARpp[s->j]
        *LARpp_j_1 ← s-> LARpp[s->j^=1]
        Llamar-a Decoding_of_the_coded_Log_Area_Ratios(LARcr, LARpp_j)
        Llamar-a Coeficients_0_12(LARpp_j_1, LARpp_j, LARp)
        Llamar-a LARp_to_rp(LARp)
        Llamar-a Short_Term_Syntesis_Filtering(S, LARp, 13, wt, s)
        Llamar-a Coeficients_13_26(LARpp_j_1, LARpp_j, LARp)
        Llamar-a LARp_to_rp(LARp)
        Llamar-a Short_Term_Syntesis_Filtering(s, LARp, 14, wt+13, s+13)
        Llamar-a Coeficients_27_39(LARpp_j_1, LARpp_j, LARp)
        Llamar-a LARp_to_rp(LARp)
        Llamar-a Short_Term_Syntesis_Filtering(s, LARp, 13, wt+27, s+27)
        Llamar-a Coeficients_40_159(LARpp_j, LARp)
        Llamar-a LARp_to_rp(LARp)
        Llamar-a Short_Term_Syntesis_Filtering(S, LARp, 120, wt+40,st40)
fin-función
```

```
Función Decoding_of_the_coded_Log_Area_Ratios(LAR, LARpp)
        { Este procedimiento requiere dos tablas para una implementación
          eficiente }
        { Se repite el siguiente procedimiento para cada uno de los valores
          de la siguiente tabla. }
```

TABLA 1-1.

B	MIC	INVA
0	-32	13107
0	-32	13107
2048	-16	13107
-2560	-16	13107
94	-8	19223
-1792	-8	17476
-341	-4	31454
-1144	-4	29708

```
        Desde i←1 hasta i<=8 hacer
                temp1 ← GSM_ADD(*LARc++, MIC)<<10
                temp1 ← GSM_SUB(temp1, B<<1)
                temp1 ← GSM_MULT_R(INVA, temp1)
                *LARpp++ ← GSM_ADD(temp1, temp1)
```

```
        fin-desde
fin-función


Función Coeficients_0_12(LARpp_j_1, LARpp_j, LARp).
        {   Cálculo de los coeficientes de reflexión cuantizados.
        Interpolación de los LARpp[1..8] para obtener los LARp[1..8].
        Dentro de cada trama de 160 muestras de voz analizada, el filtro
        de análisis de término corto y el filtro de síntesis operan con 4
        conjuntos diferentes de coeficientes, derivados del conjunto
        anterior y actual de parámetros LARs (LARpp(j-1)) decodificados

        Desde i←1 hasta i<=8 hacer
                i ← i + 1
                LARp ← LARp + 1
                LARpp_j_1 ← LARpp_j_1 + 1
                LARpp_j ← LARpp_j + 1
                *LARp ← GSM_ADD(SASR(*LARpp_j_1, 2),SASR(*LARpp_j, 2)
                *LARp ← GSM_ADD(*LARp, SASR(*LARpp_j_1, 1))
        fin-desde
fin-función


Función Coeficients_13_26(LARpp_j_1, LARpp_j, LARp)
        Desde i←1 hasta i<=8 hacer
                *LARp ← GSM_ADD(SASR(*LARpp_J_i, 1), SASR(*LARpp_j, 1) )
                i ←i + 1
                LARpp_j_1 ← LARpp_j_1 + 1
                LARpp_j ← LARpp-j + 1
                LARp ← LARp + 1

        fin-desde
        Desde i←1 hasta i<8 hacer
                i ← i + 1
                Escribir(LARp[i])
        fin-desde
fin-función
```

```
Función Coeficients_27_39(LARpp_j_1, LARpp_j,LARp)
      Desde i←1 hasta i<=8 hacer
            i ← i + 1
            LARpp_j_1 ← LARpp_j_1 + 1
            LARpp_j ← LARpp_j + 1
            LARp ← LARp + 1
            *LARp ← GSM_ADD(SASR(*LARpp_j_1, 2), SASR(*LARpp_j, 2))
            *LARp ← GSM_ADD(*LARp, SASR(*LARpp_j, 1))
      fin-desde
fin-función
```

```
Función Coeficients_40_159(LARpp_j, LARp)
      Desde i←1 hasta i<=8 hacer
            i ← i + 1
            LARp ← LARp + 1
            LARpp_j ← LARpp_j + 1
            *LARp ← *LARpp_j
      fin-desde
fin-función
```

```
Función LARp_to_rp(LARp)
        { La entrada al procedimiento es el arroglo LARp[0..7]
      interpolado. Los coeficientes de reflexión rp[i] se utilizan en el
      filtro de análisis y síntesis   }

      Desde i←1 hasta i<=8 hacer
            i ← i + 1
            LARp ← LARp + 1
            Si (*LARp<0) entonces
                  Si (*LARp=MIN_WORD) entonces
                        temp ← MAX_WORD
                  sino
                        temp ← -(*LARp)
                  fin-si
                  Si (temp<11059) entonces
```

```
                    *LARp ← -(temp<<1)
                sino
                    Si (temp<20070) entonces
                        *LARp ← -(temp+11059)
                    sino
                        *LARp ← -(GSM_ADD(temp>>26112))
                    fin-si
                fin-si
            sino
                Si (temp< 11059) entonces
                    *LARp ← temp<<1
                sino
                    Si (temp <20070) entonces
                        *LARp ← temp + 11059
                    sino
                        *LARp ← GSM_ADD(temp>>2, 26112)
                    fin-si
                fin-si
            fin-si
        fin-desde
fin-función


Función Short_Term_Syntesis_Filtering(S, rrp, k, wt, sr)
        *v ← s -> v
        Mientras (k--) hacer
            sri ← *wt++
            Desde i←8 hasta i-- hacer
                tmp1 ← rrp[i]
                tmp2 ← v[i]
                Si (tmp1=MIN_WORD AND tmp2=MIN-WORD) entonces
                    tmp2 ← MAX_WORD
                .sino
                    tmp2 ← 0x0FFFF & ((Longword)tmp1 *
                    (Longword)tmp2 + 16384)>>15))
                fin-si
                sri ← GSM_SUB(sri, tmp2)
                Si(tmp1=MIN_WORD AND sri=MIN_WORD) entonces
                    tmp1 ← MAX_WORD
                sino
```

```
                    tmp1 ← 0x0FFFF & ((longword)tmp1 *
                    (longword)sri + 16384) >> 15))
                fin-si
                v[i+1] ← GSM_ADD(v[i], tmp1)
            fin-desde
            v[0] ← sri
            *sr++ ← v[0]
        fin-mientras
fin-función
```

Capítulo 5

EVALUACION DEL SISTEMA RPE-LTP.

5.1 Introducción.

Para determinar la degradación que la señal de voz sufre por el proceso de compresión, son necesarias varias pruebas. En este capítulo se estimará la calidad del sistema, con el propósito de compararla con otros métodos. La medición se divide en dos partes:

La Subjetiva y,
La Objetiva.

A continuación se describe un método subjetivo para evaluar codificadores de voz de tasa media, sin considerar su aplicación a la telefonía digital móvil.

Método aplicado para efectuar pruebas de opinión sobre la escucha[15].

Un criterio subjetivo utilizado corrientemente es el esfuerzo que requiere la escucha, para lo cual se emplea la siguiente escala.

5	AUDICION PERFECTA;	Ningún Esfuerzo.
4	CIERTA ATENCION ES NECESARIA;	Ningún Esfuerzo Apreciable.
3	ESFUERZO MODERADO.	
2	ESFUERZO CONSIDERABLE.	
1	SIGNIFICADO INCOMPRENSIBLE;	Aún con el Mayor Esfuerzo.

En esta prueba, las personas que escuchan muestran una propensión especial a lo que se conoce como efecto de "mejora", es decir, que sus criterios de apreciación se hallan expuestos a una gran influencia de la gama de cualidades y niveles de escucha que se producen en la misma prueba y especialmente dentro de la misma pasada.

Por consiguiente, es importante que la comprensibilidad intrínseca de los grupos y listas de frase no ofrezcan variaciones demasiado amplias y que ningún participante oiga la misma frase más de una vez por experimento, ya que es evidente que se reduciría el esfuerzo necesario para comprender una frase con lo que se estará familiarizado.

Las frases suelen estar grabadas, lo que permite su reproducción con un nivel determinado. Las grabaciones a estos efectos deben realizarse y reproducirse cuidadosamente para evitar la aparición de degradaciones no controladas. Las pruebas pueden efectuarse en presencia de ruido de circuito telefónico y de ruido ambiente, y sus efectos se deben tomar en cuenta.

[15] A. E. Coleman, N. Gleiss, P. Usai, "A Subjetive Testin Methodology for Digital Mobile Radio Codecs", CSELT Technical reports, Vol. XVI, No. 6, Oct. 1988, pp. 573-584.

Una vez terminada la prueba, se repite nuevamente pero esta vez el sujeto votará de acuerdo a la siguiente escala de calidad:

5	EXCELENTE
4	BUENO
3	REGULAR
2	MEDIOCRE
1	INSATISFACTORIO

Las notas de opinión para voces masculinas a menudo difieren de las femeninas o de las infantiles, ello se debe a que los codificadores dependen de los parámetros de articulación o de producción de la voz.

Para reducir el riesgo de que los resultados dependan fuertemente de las particularidades de las voces seleccionadas, es esencial usar más de una voz masculina y más de una femenina[16] .

Se ha encontrado que la opinión subjetiva, en términos de la proporción de notas en cada una de las cinco categorías(excelente, bueno, regular, mediocre, insatisfactorio) para una condición de transmisión dada, depende de diversos factores tales como el grupo de sujetos, la gama de condiciones presentadas en la prueba y el hecho de que la prueba se haya realizado durante conversaciones en laboratorio o comunicaciones telefónicas normales[17] .

El material grabado consistió de enunciados cortos, significativos y fáciles de entender, escogidos al azar(de literatura común) con los cuales se construyên listas en orden aleatorio, de tal forma que no hay conexión obvia entre el significado de un enunciado y el siguiente.

[16] CCITT, Supl. No. 3, "Libro Rojo", Vol. V (1985).
[17] CCITT, Supl. No. 2, "Libro Rojo", Vol. V (1985).

Deben evitarse enunciados muy cortos o muy largos, el propósito es que cada frase al enunciarse tenga una duración de 3 segundos aproximadamente.

Cada muestra a evaluar consiste de dos enunciados y se graban de acuerdo al siguiente patrón:

1 seg.	3 seg.	1 seg.	3 seg.	5 seg.
(1)	(2)	(3)	(4)	(5)

(1)	Para permitir que el ruido ambiente este presente antes de la voz.
(2)	Primer enunciado.
(3)	Al igual que (1).
(4)	Segundo enunciado.
(5)	Para proporcionar el voto.

Cada participante proporciona 4 notas de opinión sobre el esfuerzo requerido de 4 muestras de locutores con diferentes características (voces masculinas y femeninas).

Posteriormente se repite el experimento con las mismas voces y frases diferentes para determinar 4 notas de calidad.

A las 8 notas proporcionadas se les agrega la edad del participante con el propósito de considerar en un futuro el efecto de apreciación.

5.2 Procedimiento utilizado.

CONDICIONES DE GRABACION.

Se grabaron 16 frases con 3 voces masculinas y 3 femeninas utilizando un micrófono dinámico Sunn modelo M158. Para la edición de los enunciados,

la grabación original de la voz se realizó en un sistema multimedia mediante la grabadora de sonidos incluida en Windows. El formato de los archivos obtenidos fué un formato PCM a 8 KHz. de frecuencia de muestreo con 16 bits por muestra. Al decodificar los segmentos nuevamente se graban en un formato .WAV para poder ser escuchados en la misma grabadora de sonidos del sistema multimedia.

Las frases empleadas son:

1	La oportunidad llega a su fin.
2	Gran exhibición anual de muebles.
3	Mañana en su nuevo horario.
4	La grandeza de la nación.
5	Ballet Nacional de Cuba.
6	Tu imaginación a través del cine.
7	Sociedad de poetas muertos.
8	Fortaleza y orgullo del pasado.
9	Visite la Ciudad de Cholula.
10	Rica mermelada de zarzamora
11	Una ventana al conocimiento.
12	El otro lado de la etiqueta.
13	Caluroso apretón de manos.
14	Ocho artistas desconocidos.
15	El toreo como arte y fiesta.
16	Orquesta Filarmónica de Florencia.

CONDICIONES DE ESCUCHA.

Habiendo grabado las 16 frases, se indica el procedimiento de evaluación a cada participante. Una vez que ha entendido todas las instrucciones se dispone a escuchar la grabación.

A continuación se describen las instrucciones dadas a los participantes:

"En este experimento se esta probando un sistema que pretende ser utilizado para la comunicación entre dos personas y se divide en dos partes:

PRIMERA PARTE

Usted va a escuchar 4 muestras de voz. Cada muestra consiste de dos enunciados separados por una pausa de 1 seg. aproximadamente.

Después de escuchar cada muestra, por favor indique su opinión acerca del esfuerzo requerido para entender el significado de los enunciados. Al término de la muestra usted dispone de 5 seg. para emitir su opinión.

Para indicar su opinión se requiere el uso de la siguiente escala:

5	AUDICION PERFECTA;	Ningún Esfuerzo.
4	CIERTA ATENCION ES NECESARIA;	Ningún Esfuerzo Apreciable.
3	ESFUERZO MODERADO.	
2	ESFUERZO CONSIDERABLE.	
1	SIGNIFICADO INCOMPRENSIBLE;	Aún con el Mayor Esfuerzo.

Por favor marque su respuesta en la siguiente tabla.

MUESTRA.	OPINION.				
1	5 □	4 □	3 □	2 □	1 □
2	5 □	4 □	3 □	2 □	1 □
3	5 □	4 □	3 □	2 □	1 □
4	5 □	4 □	3 □	2 □	1 □

SEGUNDA PARTE.

Usted va a escuchar nuevamente 4 muestras de voz. Después de escuchar cada muestra completa, por favor indique su opinión sobre la calidad del sonido. Si usted oye cualquier ruido o interferencia en la pausa entre los dos enunciados, debe considerar el efecto de esa interferencia en su juicio sobre la calidad. Al término de la muestra usted dispone de 5 seg. para emitir su opinión.

Para indicar su opinión requiere de la siguiente escala:

5	EXCELENTE.
4	BUENO.
3	REGULAR.
2	MEDIOCRE.
1	INSATISFACTORIO.

Por favor marque su respuesta en la siguiente tabla.

MUESTRA.	OPINION.				
1	5 □	4 □	3 □	2 □	1 □
2	5 □	4 □	3 □	2 □	1 □
3	5 □	4 □	3 □	2 □	1 □
4	5 □	4 □	3 □	2 □	1 □

Su colabóración nos ayudará a mejorar este sistema de comunicación, Muchas Gracias".

Después de observar 80 opiniones(10 participantes) sobre 16 enunciados articulados por 3 voces masculinas y 3 femeninas, las notas promediadas de esfuerzo y calidad son:

	OPINION DE ESFUERZO	DESVIACION ESTANDAR
Voz masculina 1	4.0	0.7406
Voz masculina 2	4.3	0.7680
Voz masculina 3	4.2	0.8890
Voz femenina 1	4.0	0.6324
Voz femenina 2	3.8	0.7480
Voz femenina 3	3.9	0.8420
Total.	4.0333	0.7799

Otra medida que ayuda a determinar la calidad del sistema de una manera objetiva, es la relación señal a ruido.

Este resultado se debe considerar con reservas, puesto que es una medida objetiva y no toma en cuenta la respuesta del oído.

La señal de error de reconstrucción *r(n)* en una codificación digital que se define como la diferencia entre la entrada al codificador *x(n)* y la salida *y(n)*; y la relación señal a ruido *S/R* [18].

$$S/R = \sigma_x^2/\sigma_r^2$$

es la relación de la varianza de la señal de entrada entre la varianza de la señal de error de reconstrucción. En el caso especial de señales con media cero, el valor de la varianza medida sobre una secuencia de longitud M apropiada es:

$$\sigma_u^2 = \frac{1}{M}\sum_{n=1}^{M} u^2(n) \qquad u = x, y \text{ ó } r.$$

[18] N. S. Jayant, P. Noll, "Digital Coding of Waveforms Principles and Applications to Speech and Video", Prentice Hall, 1984.

En esta medición M, la ventana de analisis tiene una longitud de 20 ms.

La relación señal a ruido generalmente se expresa en dB, sin embargo, para determinar la relación S/R en todo el segmento de voz, se promedia sobre los valores de cada ventana sin expresarlos en dB y después a este promedio se convierte a dB.

$$S/R = 10\log\left(\frac{1}{v}\left[S_1/R_1 + S_2/R_2 + \ldots + S_v/R_v\right]\right)$$

Donde v es el número de ventanas en el segmento considerado.

Para segmentos vocalizados, se espera que la relación señal a ruido sea mayor que para segmentos no vocalizados.

El valor final de la relación señal a ruido será entonces un promedio donde se considera la actividad estadística de la voz.

Para esta evaluación se utilizó un segmento de voz de 3 seg. con aproximadamente 154 ventanas, la relación señal a ruido promediada para todas las ventanas fue:

Señal/Ruido = 10.3 dB.

Capítulo 6

CONCLUSIONES Y PERSPECTIVAS.

En este trabajo se describió el algoritmo y la implementación del decodificador RPE-LTP. Se marcó en un inicio la gran variedad de técnicas que intentan preservar la calidad de la señal de voz con una transmisión a 16 Kbit/s por lo que nuestro codificador en conjunto RPE-LTP tiene una velocidad de operación de 13 Kbit/s neto.

Comparando la eficiencia del algoritmo RPE-LTP con el MPE-LTP concluimos inmediatamente que el primero consume mucho menos ciclos de reloj y nos permite además obtener una calidad de voz aceptable en 13 Kbit/s.

La decisión más importante para tomar y realizar la implementación de esta técnica es el hecho de que actualmente el algoritmo es considerado por el GSM como el candidato único para el estándar Europeo en las comunicaciones de radio móvil digital celular.

Este trabajo es complemento de la parte del analizador por lo que la codificación en general mediante RPE-LTP es una técnica que por un lado modela el comportamiento del tracto vocal y por el otro, codifica una parte de la señal residual. Esta doble función permite mejorar la calidad de un codificador de fuente y disminuir la cantidad de información de un codificador de forma de onda.

Una ventaja de gran importancia para este algoritmo es el hecho de que comparado con otros codificadores "híbridos", por citar alguno, la codificación por predicción lineal con excitación multipulso(MPE), su complejidad resulta ser menor sobre todo si se considera el cálculo necesario para determinar las amplitudes y posiciones de los pulsos de la señal de excitación. Por lo tanto, se considera que el algoritmo es de complejidad media.

Los sistemas basados en la codificación por predicción lineal para sintetizar con buena aproximación, necesitan que los parámetros que modelan la articulación y producción de la voz, se determinen con gran exactitud. Esto llega a ser una gran limitante si se desea que estos sistemas funcionen con alta calidad. Sin embargo, la poca información con la que puede codificarse un segmento de voz, los hace eficientes para transmitir a muy bajas velocidades.

Existen otros sistemas de codificación a velocidades más bajas que la presentada por el algoritmo RPE-LTP, tal es el caso de la codificación por predicción lineal con excitación codificada(CELP); pero este algoritmo actualmente es muy complejo para operar en tiempo real. Por lo que tomamos en consideración la complejidad del algoritmo junto con la eficiencia del mismo y el grado de calidad subjetiva que nos proporciona.
En las tablas 1, 2 y 3 se muestran las características principales de los codificadores candidatos que el GSM evaluó junto con el codificador RPE-LTP.

CODIFICADOR	RPE-LPC Esquema básico	MPE-LTP Predictor	RPE-LTP Nuevo esquema
Tasa neta	14.8 Kbit/s	13.2 Kbit/s	13.0 Kbit/s
Tiempo Codificado por trama	19.5 ms.	20 ms.	20 ms.
Ventana	24.375 ms. Hamming traslapada	20 ms. Rectangular sin traslape	20 ms. Rectangular sin traslape
Orden del filtro de análisis	12	8	8
Algoritmo	Schur	Le Roux-Guegen	Schur
Codificación de coeficientes	52 bits	28 bits	36 bits
Tipo de filtro análizador/sintetizador	Celosía	Forma directa	Celosía
Modelo de excitación	Pulsos regulares (Método simplificado)	Multi-pulso con predictor de término largo	Pulsos regulares (Método simplificado)
Número de pulsos	52 / trama 4 bits / pulso	24/ trama 3 bits / pulso	52 / trama 3 bits / pulso

Tabla 1 Características de los codificadores RPE-LPC, MPE-LTP y del nuevo esquema RPE-LTP.

PARAMETRO	SBC-APCM	SBC-APCM	SBC-APCM	SBC-ADPCM
Señales de las Sub-Bandas	14 Kbit/s	14 Kbit/s	10 Kbit/s	15 Kbit/s
Codificación	Max	Max	Max	ADPCM
Asignación de bits	Adaptiva 1-4	Adaptiva 0, 2-5	Adaptiva 0-5	Banda 1-6: 4,3,2,2,2,2
Información lateral	1 Kbit/s	1 Kbit/s	3 Kbit/s	0 Kbit/s
Representación espectral	Cuantización vectorial	Cuantización vectorial	PCM logarítmico	
Tamaño del "recetario"	128x6 bits	128+64x8+4+4 bits	14x3 bits	
Tasa neta	15 Kbit/s	15 Kbit/s	13 Kbit/s	15 Kbit/s
Sincronía				1 Kbit/s
Protección contra errores	1 Kbit/s	1 Kbit/s	3 Kbit/s	
Longitud de la trama	15 ms	20 ms	16 ms	1 ms
No. de bandas Total/en uso	8/6	16/14	16/14	8/6
Tipo de filtro	QMF	Paralelo	Paralelo	QMF
Etapas	32,24,16	64	80	32,24,16
Ganancia del bloque	PCM Ley A	Log PCM	Log PCM	

Tabla2: Parámetros de los algoritmos SBC.

PARAMETRO	RPE-LPC	MPE-LTP
Tasa Neta	14.77 Kbit/s	13.2 Kbit/s
Protección contra errores	1.23 Kbit/s	2.8 Kbit/s
Trama de voz	19.5 ms	20 ms
Ventana	24.375 ms	20 ms
Orden del filtro de análisis	12	8
Algoritmo	Schur	Le Roux Gueguen
Codificación de coeficientes	52 bits	28 bits
Estructura del filtro	Celosía	FIR directa
Modelo de excitación	Pulsos regulares (Método simplificado)	Multi-pulso con predicción de término largo
No. de pulsos/trama	52	24

TABLA 3. Parámetros de los codificadores con excitación de pulsos

Esta comparación mostrada en las tablas anteriores son de aquellos codificadores que transmiten a 16 Kbit/s

Para el restablecimiento de la señal codificada se llevaron a cabo simulaciones en MatLab mismas que sirvieron de comparación con la señal original dándonos así información respecto a la eficiencia del algoritmo implementado. Las características propias de MatLab dan flexibilidad de análisis de la señal ya que a través de sus funciones predefinidas nos permitió simular nuestra información de voz. Un ejemplo claro fue el

realizado con los filtros de síntesis de término corto y largo. Para ello se hizo uso de las funciones predefinidas de filtros digitales en su sección de funciones para el área de comunicaciones, el resto de las funciones se implementaron dentro de MatLab ya que este software permite poder crear nuestras propias funciones debido a la facilidad que presenta en la codificación del programa.

Los problemas encontrados en la implementación del decodificador fueron en primer término:

El algoritmo RPE; este en su forma original presenta variantes respecto al algoritmo RPE simplificado y estas variantes afectan de manera directa en la degradación o distorsión de la propia señal modificando en forma considerable la relación Señal a Ruido.

Podemos decir que el algoritmo original utiliza gran cantidad de operaciones matriciales tanto directas como inversas para que al final se obtenga un vector fila que contiene el vector de excitación óptimo el cual se toma para la excitación del filtro de síntesis. Esto por consiguiente aumentaba la complejidad del algoritmo. Este inconveniente se resolvió utilizando la versión simplificada evitando así el uso de operaciones matriciales.

Respecto a las pruebas subjetivas realizadas en codificadores tales como el algoritmo RPE-LPC, su nota de opinión promedio es de 3.54[4] ; ahora bien para que un codificador pueda considerarse de buena calidad deberá tener una nota de opinión promedio entre 4 y 4.5. Por lo tanto, el codificador RPE-LTP tiene una nota de opinión de 4.3 trabajando en condiciones libres de error en la transmisión, en nuestra implementación se obtuvo una nota real de 4.0333. Debido a lo anterior, el GSM(Groupe Special Mobile) lo tomó

[4] Jon E. Natvig, "Pan-European Speech Coding Standard for Digital Mobile Radio", Speech Communications, Jul. 1988, pp. 113-123.

como estándar para la transmisión de voz aplicada a la telefonía móvil digital.

El único inconveniente que presenta el decodificador y que es materia para mejorar es el hecho de que la voz sintetizada ha sufrido una ligera atenuación debido al propio proceso de codificación y regeneración de la voz por lo que queda abierta la alternativa de mejorar y eficientar aun mas el algoritmo.

Finalmente, considero que el algoritmo RPE-LTP es una muy buena alternativa de codificación de voz y este trabajo puede continuarse a la siguiente etapa de implementación en una tarjeta de procesamiento de señales para correr en tiempo real pudiendo hacer uso de la tarjeta TAC-31C con el procesador de señales TMS320c.

Apéndice

INTRODUCCIÓN.

A continuación se presenta la descripción y pseudocódigo de dos programas auxiliares utilizados para el proceso de codificación de la voz en su conjunto.

En el proceso de Codificación de voz a 16 Kbit/s se utilizaron dos programas auxiliares, mismos que sirvieron para eliminar el código de cabecera del archivo original de voz analizado y posteriormente agregar dicha cabecera, es decir, todo archivo de voz (.WAV) posee una cabecera de identificación RIFF la cual identifica que es un archivo de voz[19][20] . Esta cabecera se elimina dejando así únicamente la sección de datos misma que es analizada por el algoritmo de codificación RPE-LPC y finalmente a estos datos resultantes reincorporar nuevamente su cabecera de identificación de archivo de voz (.WAV).

---

[19] La producción de música por medios digitales, "Personal Computing", México, año 7, No. 83. pp. 38-42.

[20] RIFF WAVE(.WAV) File Format. Rob Ryan <ST02200@brownvm.brown.edu> Organization: Brown University.

Análisis de la cabecera de un archivo de sonido con extensión .WAV

En procesamiento digital de señales se hace uso común de una grabadora de sonidos para analizar voz o cualquier otro tipo de sonido, es por eso que se necesita analizar la cabecera de un archivo .WAV para saber donde empieza cada elemento del *Chunk*.

Referente al audio digital, este se incorporó definitivamente a Windows a partir de su versión 3.1, a través de los archivos con extensión .WAV *(waveform)* que almacenan la información digitalizada en forma de onda de diferentes sonidos. Existe un conjunto de funciones del API para controlar la reproducción de sonido, aunque el alcance de esta técnica y la adopción de los archivos .WAV va más allá de Windows.

Los archivos .WAV se utilizan para guardar audio digital y pertenecen a una familia más extensa de archivos creados por Microsoft, principalmente para su ambiente Windows. Esta familia es de los archivos RIFF (Resource Interchange File Format). Todos estos archivos siguen una misma convención en su formato, la cual permite crear nuevos tipos de archivos sin alterar la convención original, y es similar en estilo al estándar TIFF que se usa para imágenes.

Existe cierta confusión en la literatura sobre el origen de este tipo de archivos pues hay referencias al IFF de una compañía llamada Electronic Arts, como el origen de toda esta familia. De cualquier forma, estos formatos tratan de satisfacer criterios de extensión y flexibilidad. A cambio de crecer pueden extenderse prácticamente en forma ilimitada, Además, todo tipo de software puede ignorar las partes que no sea de su interés en forma eficiente.

La unidad básica de un archivo RIFF es el *chunk* o *segmento*. Se utiliza este término para denominar a estas unidades, a reserva de que en el futuro surja un término equivalente que resulte mas práctico. Un chunk esta compuesto de tres elementos :

a) En primer término, una etiqueta de cuatro caracteres; si se usan menos de cuatro el resto se rellena con espacios.
b) Sigue un indicador del tamaño del segmento, como un entero positivo de 32 bits.
c) Inmediatamente después el área de datos con la longitud anunciada.

Si un programa identifica un segmento que desea ignorar, basta que consulte su longitud y haga un *seek* al principio del segmento siguiente.

Un segmento puede contener en su área de datos cualquier cosa, incluyendo otros segmentos (subsegmentos). Esta estructura refuerza la facilidad de identificar secciones completas de información. Siguiendo esta idea, un archivo RIFF contiene varios segmentos y subsegmentos, encerrados todos en un gran segmento con el identificador RIFF y la longitud del archivo al principio.

Es importante subrayar que la longitud de un segmento es del área de datos, sin contar el espacio ocupado por la etiqueta y el entero que indica el tamaño. Un archivo .WAV empieza con el identificador RIFF, seguido del tamaño del segmento en cuatro bytes.

Lo primero en el área de datos del segmento RIFF es un identificador mas WAVE, que reconoce al archivo como un RIFF-waveform, o sea, WAV. Después de esta etiqueta sigue un segmento marcado como "fmt " (nótese el espacio antes de la segunda comilla), con una longitud indicada de 16

bytes. A continuación se da una descripción del formato, que incluye los siguientes elementos :

♦ Una bandera que indica si es PCM u otro formato, siempre se llena con 1. (2 Bytes)

♦ El número de canales 1 : monoaural
2 : estéreo (2 bytes)

♦ El número de muestras por segundo (4 bytes)

♦ Número promedio de bytes por segundo. Esta información se usa principalmente para estimar el tamaño del buffer que se requerirá para procesar eficientemente el archivo. (4 bytes)

♦Una sugerencia del tamaño del bloque más eficiente para leer los datos. (2 bytes)

♦ Número de bits por muestra (8 o 16). (4 bytes)

La información anterior constituye el segmento "fmt ".

Le sigue el identificador "fact" que esta constituido por :

a) La etiqueta "fact". (4 bytes)
b) Un entero constante 4. (4 bytes)
c) Tamaño[1] del segmento de datos. (4 bytes)

Y por ultimo, el segmento de datos el cual contiene :

i) La etiqueta "data". (4 bytes)
ii) Tamaño[1] del segmento de datos. (4 bytes)

[1] Nota : El tamaño del segmento de datos es igual a la cantidad total de bytes del archivo menos cincuenta y ocho bytes correspondientes a los bytes que ocupa la cabecera.

En el segmento de datos se encuentran las muestras del sonido, con la longitud anunciada (8 o 16 bits) indicada en la especificación del formato.

A continuación se muestra la estructura de la cabecera de un archivo .WAV

	Identificador	Valor	Tipo
1	RIFF	1179011410	Entero largo
2	Tamaño del archivo	variable	Entero largo
3	WAVE	1163280727	Entero largo
4	fmt_	544501094	Entero largo
5	18__	18	Entero largo
6	1_	1	Entero
7	1_	1	Entero
8	8000	8000	Entero largo
9	8000	8000	Entero largo
10	1_	1	Entero
11	8___	8	Entero largo
12	fact	1952670054	Entero largo
13	4___	4	Entero largo
14	Tamaño del archivo - 58 bytes	variable	Entero largo
15	data	1635017060	Entero largo
16	Tamaño del archivo - 58 bytes	variable	Entero largo

1. Identificador de la familia a la que pertenecen los archivos.
3. Identificador que reconoce al archivo como RIFF-waveform.
4. Etiqueta del segmento formato.
6. Bandera que indica que es formato PCM.
7. Número de canales, 1 : Monoaural.
   2 : Estéreo.
8. Número de muestras por segundo.
9. Número promedio de bytes por segundo.
10. Tamaño del bloque más eficiente para leer los datos.
11. Número de bits por muestra.
12. Etiqueta que almacena información acerca del contenido del archivo.
15. Etiqueta del segmento de datos.

```
Función Extraction_of_head()
Inicio
Definición-de estructura con variable, Chunk_header
Definición-de estructura con variable, Waveformat
Definición-de estructura con variable, Dataformat
Lectura-de archivo de voz
Lectura-de bloque de datos de cabecera, fread(&ck, sizeof(Chunk_header))
Escritura-de bloque mediante, writebin((char*)&ck.ident, sizeof(dword))
Lectura-de bloque format id=wave, fread(&tag, sizeof(dword), 1, fp)
Escritura-de bloque mediante writebin((char*)&tag, sizeof(dword))
Lectura-de bloque FMT Chunk mediante fread(&ck, sizeof(Chunk_header), 1, fp)
Escritura-de bloque FMT Chunk mediante writebin((char*)&ck.ident,
sizeof(dword))
Guardando datos restantes en nuevo archivo
Fin
```

```
Función Set_header()
Inicio
      Definición-de estructura con variable, Chunk_header
      Definición-de estructura con variable, Waveformat
      Definición-de estructura con variable, Dataformat
      Definición-de estructura con variable, Fmatrix
      Abrir (archivo de salida)
      Determinación-de tamaño del archivo de salida (bytes), size_file
      ck.ident ← 1179011410
      ck.size ← size_file
      Escribir bloque de datos, fwrite(&ck, sizeof(Chunk_header), 1, fpa)
      tag ← 1163286727
      Escribir bloque de datos, fwrite(&ck, sizeof(udword), 1, fpa)
      ck.ident ← 544501094
      ck.size ← (udword)18
      Escribir bloque de datos, fwrite(&ck, sizeof(Chunk_header), 1, fpa)
      fmt.tag ← (uword)1
      fmt.canales ← (uword)1
      fmt.muestras ← (udword)8000
      fmt.bprom ← (udword)16000
      fmt.alineam ← (udword)2
      fmt.databits ← (udword)16
      Escribir bloque, fwrite(&fmt, sizeof(Waveformat), 1, fpa)
      ck.ident ← 1952670054
      ck.size ← (udword)4
      Escribir bloque, fwrite(&ck, sizeof(Chunk_header), 1, fpa)
      dk.inidat ← (udword)size_file - 58
      dk.data ← 1635017060
      dk.findat ← (udword)size_file - 58
      Escribir bloque, fwrite(&dk, sizeof(Dataformat), 1, fpa)
      Grabar archivo de salida
Fin
```

GLOSARIO DE TERMINOS TECNICOS

RPE-LTP : Abreviatura del codificador de predicción de término largo con excitación de pulsos regulares, Regular-Pulse Excitation-Long Term Prediction.

ISDN : Siglas de la red mundial de comunicaciones llamada Red Digital de Servicios Integrados (Integrated Services Digital Network), dicha red tiene diversos servicios tales como transmisión de voz y datos, encriptamiento de datos, etc.

BIT : Abreviatura de dígito binario. Un bit puede tomar el valor de uno o cero.

Kbit/s : Unidad de medida para la tasa de transmisión equivalente a miles de bits transmitidos o recibidos en un segundo.

GSM : Siglas para el Grupo Móvil Especial(Groupe Special Mobile) fundado en 1982 por la administración de Telecomunicaciones Europea con la finalidad de estudiar y desarrollar un estándar para los futuros sistemas móviles Pan-Europeo.

CEPT : (Conférence Européenne des Administrations des Postes et Telécommunications, CEPT) este es un comité europeo.

RELP : Codificador básico(RELP, Residual Excited Linear Prediction) codificador de predicción Lineal de excitación residual.

Vocoder : Abreviatura de Codificador-Decodificador de voz. (Voice Coder Decoder) también se utiliza el término *Codec*.

Vocoder de fuente : Es el tipo de codificador que extrae los parámetros más importantes de una señal de voz. Estos parámetros representan las características espectrales de la señal de voz. Las técnicas de codificación tales como LPC, MPE, MPE-LTP, CELP y RELP son ejemplos de codificadores de fuente.

Vocoder de forma de onda : Este otro tipo de codificadores se enfocan a reproducir la señal de voz de una manera lo más fidedigna posible a través de la propia forma de onda de la señal. Ejemplo de ellos tenemos PCM, DM, DPCM, ADM y ADPCM.

Banda base : Es la frecuencia que ocupa una señal cuando es generada inicialmente.

Compansión : Término utilizado para indicar el proceso mediante el cual la voz es comprimida y descomprimida dentro de su rango dinámico. El término es una contracción de las palabras compresión y expansión.
Existen dos leyes de compansión que rigen a todo el mundo, estas leyes estan estandarizadas por el CCITT. La ley μ es utilizada en EE.UU. y Japón, la ley A se utiliza en Europa y el resto del mundo.

Ley μ : Es la primera de las dos leyes de compansión, la característica del compresor *C(x)* *es continua*, aproximando una dependencia lineal en *X* para bajos niveles de entrada y una logarítmica para altos niveles. Esta ley se describe en términos algebraicos como :

$$\frac{c(x)}{x_{max}} = \frac{Ln(1+\mu|x|/x_{max})}{Ln(1+\mu)} \qquad 0 \le \frac{|x|}{x_{max}} \le 1$$

Ley A : Es la segunda ley de compansión *A*, la característica del compresor *C(x)* *es discreta*, formada por un segmento lineal para bajos niveles de entrada y un segmento logarítmico para altos niveles.
Esta ley se describe como :

$$\frac{C(x)}{x_{max}} = \begin{cases} \dfrac{A(x)/x_{max}}{1+Ln(A)} & 0 \le \dfrac{|x|}{x_{max}} \le \dfrac{1}{A} \\ \dfrac{1+Ln(A|x|/x_{max})}{1+Ln(A)} & \dfrac{1}{A} \le \dfrac{|x|}{x_{max}} \le 1 \end{cases}$$

S/N : Es la proporción de ruido que existe en la señal o relación Señal-Ruido.

Cuantización : Proceso que transforma muestras de voz analógica en un número finito de posibles valores.

Enmascaramiento de ruido : El enmascaramiento se da cuando una señal evita la percepción de otra. Cuando una señal no ruidosa evita la percepción de otra señal con ruido decimos que existe enmascaramiento del ruido. El enmascaramiento depende del nivel relativo de ambos sonidos y de sus componentes frecuenciales. El enmascaramiento de un tono es mas efectivo cuando sus frecuencias son próximas.

Trama : Segmento de voz con una cierta cantidad de muestras de la señal para análisis o síntesis y esta normalmente se encuentra en un rango de 15 ms a 50 ms.

Filtro de Síntesis : Es un filtro de análisis inverso. El filtro de análisis es un filtro de predicción cuya entrada es la secuencia de muestras de voz original y su salida es el error de predicción. Luego entonces, el filtro de síntesis toma como entrada el error de predicción cuya salida será la señal de voz regenerada llamada señal sintética.

Voz Sintética : Es la señal de voz producida a la salida del filtro de síntesis, caracterizada con los parámetros extraídos de la señal de voz original.

REFERENCIAS BIBLIOGRAFICAS

[1] K. Hellwig, R. Hofmann, R. J. Sluyter and P. Vary, "Mats-D Speech Codec: Regular-Pulse Excitation LPC", Second Nordic Seminar on Digital Land Mobile Radio Communication, 14-16 October, Stockholm,pp. 257-261, (1986).

[2] R. García Gómez, "Tratamiento Numérico de la voz", Departamento de señales, Sistemas y Radiocomunicaciones ETSI de Telecomunicación. Universidad Politecnica de Madrid, Madrid, Sep. 1991, Lección 2, pp 1-11; Lección 3, pp 1-18; Lección 5, pp 1-23; Lección 6, pp 1-36.

[3] M. Decina, G. Modena, "CCITT Standards on Digital Speech Processing", IEEE Journal on Selected Areas in Communications, Vol. 6, No. 2, Feb 1988, pp 227-233.

[4] Jon E. Natvig, "Pan-European Speech Coding Standard for Digital Mobile Radio", Speech Communications, jul. 1988, pp 113-123.

[5] V. C. Alberto, Tésis, "Codificador de Voz a 16 Kbit/s Usando RPE-LTP", Universidad Tecnológica de la Mixteca, Marzo,1997.

[6] R. E. Crochiere, "On the Design of Sub-Band Coders For Low Bit rate speech Communication", Bell System Tech. J. Vol. 56, No. 5, May-Jun 1977.

[7] R. E. Crochiere, S. A. Webber, J. L. Flanagan, "Digital Coding of Speech in Sub-Bands", Bell System Tech. J. Vol. 55, No. 8, Oct. 1976.

[8] A. J. Goldberg, H. L. Shaffer, "A Real-Time Adaptive Predictive Coder Using Small Computer", IEEE COM-23, No. 12, Dec. 1975. pp 1443-1451.

[9] J. Makhoul, "Linear Prediction: A Tutorial Review", Proc. IEEE, Vol. 63, No. 4, Apr. 1975, pp 561-580.

[10] B. S. Atal, J. R. Remde, "A new model of LPC excitation for producing Natural-Sounding Speech at Low Bit Rates", Proc. IEEE, Int. Conf. on Acoustics, Speech Signal Processing, Apr. 1982, pp. 614-617.

[11] P. Kroon, E. F. Deprettere, R. J. Sluyter, "Regular-Pulse Excitation A Novel Approach to Effective and Efficient Multipulse Coding Speech", IEEE ASSP-34, No. 5, Oct. 1986, pp 1054-1063.

[12] J. L. Flanagan, M. R. Schroeder, B. S. Atal, R. E. Crochiere, N. S. Jayant, J. M. Tribolet, "Speech Coding", IEEE COM-27, No. 4, Apr. 1979, pag 710-733.

[13] G. Schroder, "Residual-Excited with Vector Quantization (RELP-VQ)", Speech Communication, Jul. 1988, pp 227-237.

[14] P. Vary, R. J. Sluyter, C. Galand and M. Rosso, "RPE-LTP Codec- The candidate for the GSM Radio Communication System", Int., Conf. on Digital Land Mobile Radio Communications, 30 june- 3 july, Venice, pp. 507-516. (1987).

[15] A. E. Coleman, N. Gleiss, P. Usai, "A subjetive testing methodology for digital mobile radio codecs", CSELT Technical reports, vol. XVI, No. 6, Oct. 1988, pp 573-584.

[16] C.C.I.T.T. "Libro Rojo", Vol. 3, Rec. G. 721, 1985.

[17] C.C.I.T.T. Supl. No. 2, "Libro Rojo", Vol. V, 1985.

[18] N. S. Jayant, P. Noll, "Digital Coding of Waveforms, Principles and Applications to Speech and Video " Prentice Hall 1984.

[19] La producción de musica por medios digitales. "Personal Computing, México", Año 7, No. 83, pp. 38-42, Abril 1995.

[20] RIFF WAVE (.WAV) file format
Rob Ryan <ST02200@brownvm.brown.edu> Organization: Brown University.

Bibliografía Auxiliar.

[21] P. Vary, R. Hofmann, K. Hellwing, R. J. Sluyter, "A Regular-Pulse Excited Linear Predictive Codec", Speech Communication, Jul. 1988, pp 209-215.

[22] L. R. Rabiner, R. W. Schafer, "Digital processing of Speech Signals", Englewood Cliffs, N. J. Prentice Hall, 1978.

[23] P. M. Embree, B. Kimble, "C Language Algoritms for Digital Signal Processing", Prentice Hall Englewood Cliffs, New Jersey, 1991.

[24] A. Openheim and R. Schafer, "Digital Signal Processing", Prentice-Hall, Englewood Cliffs, New Jersey 1975.

[25] A. Openheim and R. Schafer, "Discrete-Time Signal Processing", Prentice-Hall, Englewood Cliffs, New Jersey 1989.

[26] The Math Works Inc., "The Student Edition MatLab: Student User Guide", Prentice-Hall, Englewood Cliffs, New Jersey 1992.