

ANÁLISIS Y SÍNTESIS DE VOZ HUMANA MEDIANTE PREDICCIÓN LINEAL LPC

TOMÁS MARTÍN ¹

Procesamiento Digital de Señales, Ingeniería de Sonido
Universidad Nacional de Tres de Febrero, Buenos Aires, Argentina.
¹tomasheto@hotmail.es

RESUMEN – En este trabajo se presenta el estudio de un método de compresión de audio dedicado a sintetizar voz humana conocido como LPC (Linear Predictive Coding). Se realiza este análisis con el fin de comprender cómo bajo el análisis de una señal de audio se puede lograr reducir considerablemente su tamaño y posibilitar así una transmisión con calidad aceptable por canales con ancho de banda reducido. Se utiliza para el mismo una estimación de los valores que mejor representan los coeficientes del filtro que surge por la influencia del tracto vocal bajo la minimización del error cuadrático. Para esto se implementa un código en Matlab que analiza archivos de voz humana y calcula los coeficientes de LPC mediante el método de autocorrelación utilizando el algoritmo de Levinson-Durbin para resolver el sistema de ecuaciones. A partir de los resultados se analiza la influencia del tipo y ancho de la ventana temporal y de la cantidad de coeficientes que se estiman.

1. INTRODUCCIÓN

Con el comienzo de la creación de redes de computadoras se vió la necesidad de generar métodos de transmisión de información eficiente por canales de ancho de banda reducida. Esto también es aplicado a la transmisión de voz previamente digitalizada. Por esta razón se hace imprescindible el uso de técnicas de compresión de las señales de audio que permitan reducir considerablemente el volumen de información necesario para representarlas. Es decir, en lugar de lograr una conexión dedicada a transmitir toda la señal de voz digitalizada entre dos usuarios remotos, se forman paquetes de datos con características representativas de la señal para su posterior recepción y reconstrucción de la señal original de voz y de este modo lograr altas tasas de compresión de información. Un claro ejemplo de esto es el algoritmo LPC (Linear Predictive Coding), el mismo hace corresponder la señal de audio con un modelo lineal y simple y obtiene los parámetros que mejor ajustan el modelo a la señal.

El objetivo de este trabajo de investigación es entonces diseñar y estudiar las características del método de compresión LPC. Una vez diseñado el algoritmo se estudia además la influencia del ancho de la ventana y de la cantidad de coeficientes que se estiman.

Lo que pretende el presente informe es describir metódicamente como se desarrolla el proceso de compresión de la señal de voz mediante LPC. Particularmente, aquí se resuelve mediante el método de autocorrelación utilizando el algoritmo de Levinson-Durbin para la resolución del sistema de ecuaciones. Para finalizar se realiza un posterior análisis de los resultados obtenidos y de la influencia de variar los distintos parámetros con los que se genera la representación de la señal. A continuación se detallan una serie de conceptos teóricos necesarios para comprender el algoritmo seguido de la explicación del mismo.

2. CONCEPTOS TEÓRICOS

2.1 Descripción del aparato fonador

El aparato fonador está formado por un conjunto de órganos con la función de producir la voz humana.

En primer lugar, los pulmones producen un flujo de aire que pasa por la laringe donde se ubican las cuerdas vocales. Es en este punto que se produce sonido debido a las vibraciones de las cuerdas vocales. Dependiendo de la tensión, elasticidad, altura, anchura, longitud y grosor de las mismas es que se define la altura tonal y la intensidad del sonido emitido.

Luego el sonido pasa a la faringe, a las cavidades oral y nasal. Es en esta zona en la que se produce el mecanismo de resonancia. En cuanto a las cavidades, la nasal es una cavidad rígida que no puede cambiar de tamaño, pero la oral puede cambiar de tamaño por lo que se encarga de la modulación del sonido. Por otro lado, la faringe (tubo encargado de distribuir el aire que llega desde la laringe) también puede modificar su tamaño y con ello determinar el timbre de la voz.

Por último están los órganos articuladores, estos son el paladar, lengua, dientes, labios entre otros. Estos son los encargados de convertir el sonido en fonemas necesarios para la comunicación humana.



Figura 1: Generación de sonido Sonoro

2.2 Frecuencias Formantes^[1]

Las cavidades que actúan como resonadores acústicos producen un fenómeno de resonancia en el sonido que las atraviesa. Este fenómeno produce un énfasis en determinadas frecuencias del espectro a las que se denomina Formantes. Existen tantas Formantes como resonadores posee el tracto vocal, sin embargo se considera que solo las tres primeras, asociadas a la cavidad oral, bucal y nasal proporcionan suficiente cantidad de información para poder diferenciar los distintos tipos de sonido.

2.3 Tipos de señales de voz^[1]

El sonido que producen las cuerdas vocales es llamado sonido sonoro (voiced), en contraste del sonido sordo (unvoiced) en cuya producción no vibran las cuerdas vocales.

Las señales sonoras se caracterizan por tener alta energía en el rango de los 300 Hz a 4000 Hz presentando cierta periodicidad (naturaleza cuasiperiódica). Por otro lado las señales sordas se caracterizan por tener una alta densidad de cruces por cero y baja energía comparadas con las señales del tipo sonora.

2.4 Modelo del tracto vocal^[2]

Se considera que el tracto vocal tiene un comportamiento similar al de un filtro, cuyos parámetros varían en el tiempo al pronunciar distintas palabras.

En la figura 2 se muestra un diagrama en bloques del modelo del tracto vocal.

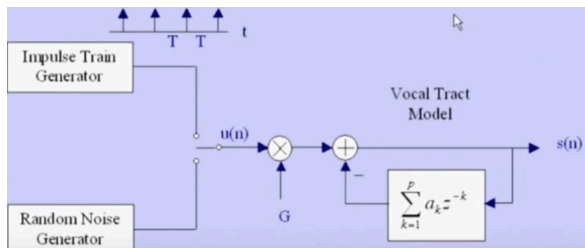


Figura 2: Modelo simple de tracto vocal

En el modelo de tracto vocal se consideran dos posibles entradas que dependen de si la señal a reproducir es sonora o sorda. Para la señal sonora, la excitación es un tren de impulsos de frecuencia controlada, mientras que para las señales sordas la señal generada es un ruido aleatorio. El espectro en frecuencia de la señal de voz de salida se obtiene a partir del producto del espectro de la señal generada por el espectro de la respuesta en frecuencia del filtro del tracto vocal. Además, cuando se habla del filtro de tracto vocal se entiende como un filtro todo polos *all-poles system*.

Las muestras $s[n]$ están relacionadas con la excitación $u[n]$ por la siguiente ecuación en diferencias:

$$s[n] = \sum_{k=1}^p a_k s[n-k] + Gu[n] \quad (1)$$

donde $s[n]$ es la muestra n de la secuencia de muestras de habla; $s[n-k]$ son las muestras pasadas, que se multiplican por la constante a_k y $G.u[n]$ es la excitación. Entonces, la muestra actual es igual a las muestras pasadas pesadas por una constante y sumadas a la excitación.

Entre pulsos de nota la excitación ($Gu[n]$) es cero. Solamente es distinto de cero cuando hay un comienzo de período. Entre excitación y excitación se puede decir que la muestra actual es una combinación lineal de muestras previas.

El modelo de producción de habla, modelo del tracto vocal, tiene función de transferencia:

$$H_{VT}(z) = \frac{S(z)}{U(z)} = \frac{G}{1 - \sum_{k=1}^p a_k z^{-k}} \quad (2)$$

2.5 Compresión de archivos de audio^[3]

El objetivo de minimizar el caudal de bits necesario para su transmisión preservando la calidad de la señal es de suma importancia cuando se intenta transmitir señales por anchos de banda reducidos.

Es entonces que resulta imprescindible el uso de técnicas de compresión de señales de audio que permiten reducir considerablemente el volumen de información necesario para representarlas. Algunos ejemplos de estas son LPC, CELP, MP3, MPEG, entre otros.

2.6 LPC (Linear Prediction Coding)^[2]

El algoritmo LPC es un método confiable y robusto para estimar los parámetros del sistema modelo del tracto vocal que combina el pulso de la glotis, el tracto mismo y las características de la radiación del habla.

Se suele llamar a LPC "filtro inverso", ya que intenta determinar el filtro "all-zero", que es el inverso del modelo del tracto vocal. El diagrama de bloques del predictor lineal es el siguiente:

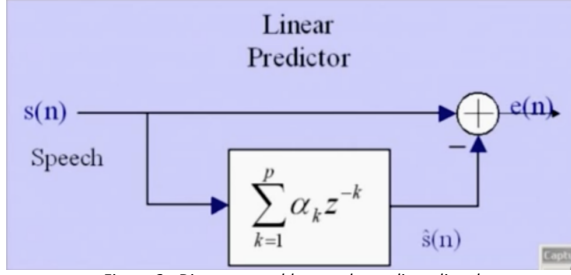


Figura 3: Diagrama en bloques de predictor lineal

El bloque de la transformada Z de los coeficientes alfa representa el pesaje de las muestras pasadas de $s[n]$ por los α_k , que implican un filtro. Tal pesaje de las muestras presente y pasadas de la señal entrega un estimador de $s[n]$ nomencalado como $\hat{s}[n]$. Por lo tanto se puede estimar la señal original con muestras pasadas de la misma. Vale la pena agregar que si el número de coeficientes aumenta la estimación se aproxima mejor a la señal original pero superadas los 12 coeficientes aproximadamente la gráfica del error en función de los coeficientes se aplanas asintóticamente, es decir que con mas de 12 coeficientes la estimación no será tanto mejor comparado con el costo computacional que esto significa.

El error $e[n]$ se define como la diferencia de la señal y su estimación.

Por lo tanto la transformada Z del error queda:

$$E(z) = S(z) \cdot (1 - \sum_{k=1}^p \alpha_k \cdot z^{-k}) \quad (4)$$

Y entonces, si se pasa a la función de transferencia del sistema se obtiene:

$$H_{LPC}(z) = \frac{E(z)}{S(z)} = 1 - \sum_{k=1}^p \alpha_k \cdot z^{-k} \quad (5)$$

Considerando el filtro LPC como

$$F(z) = \sum_{k=1}^p \alpha_k z^{-k} \quad (6)$$

La estimación entonces puede obtenerse al multiplicar la señal original por el filtro LPC

$$\hat{S}(z) = F(z) \cdot S(z) \quad (7)$$

Por otro lado, multiplicando las ecuaciones 2 y 4 se obtiene:

$$H_{VT}(z) \cdot H_{LPC}(z) = \frac{S(z)}{U(z)} \cdot \frac{E(z)}{S(z)} = \frac{E(z)}{U(z)} \quad (8)$$

$$\frac{E(z)}{U(z)} = G \cdot \frac{1 - \sum_{k=1}^p \alpha_k \cdot z^{-k}}{1 - \sum_{k=1}^p a_k \cdot z^{-k}}$$

si la predicción es correcta, los α_k serán iguales a los a_k del modelo del tracto vocal.

$$\frac{E(z)}{U(z)} = G \quad (9)$$

y antitransformando:

$$e[n] = G \cdot u[n] \quad (10)$$

Conociendo como obtener la señal estimada a partir de la ecuación 6, el problema radica en obtener los valores de los α_k . Para esto considerando que el error es la diferencia entre la señal y su estimación, lo que se busca es minimizarlo.

$$e[n] = s[n] - \sum_{k=1}^p \alpha_k s[n-k] \quad (11)$$

Además, debido a la naturaleza variable de la señal de voz en el tiempo, los coeficientes del predictor deben estimarse mediante un procedimiento de análisis a corto plazo basado en encontrar el conjunto de coeficientes de predictor que minimizan error cuadrático para cada segmento de corta duración de la señal.

2.7 Minimización de Error Cuadrático^[4]

El error cuadrático se define como:

$$E = \sum_m e[n]^2 = \sum_m (s[m] - \hat{s}[m])^2 \quad (12)$$

$$\sum_m (s[m] - \hat{s}[m])^2 = \sum_m \left(s[m]^2 - \sum_{k=1}^p \alpha_k s[m-k] \right)^2$$

Luego la ecuación de minimización:

$$\frac{dE}{d\alpha_k} = 0 \quad (13)$$

queda:

$$0 = \sum_m 2 \left(s[m] - \sum_{k=1}^p \alpha_k s[m-k] \right) (-s[m-i]) \quad (14)$$

y a partir de esta ecuación se llega a:

$$\sum_m s[m]s[m-i] = \sum_{k=1}^p \alpha_k \sum_m s[m-k]s[m-i] \quad (15)$$

al definir la autocorrelación como:

$$\varphi_{\hat{n}}[i, k] = \sum_m s_{\hat{n}}[m-i]s_{\hat{n}}[m-k], \quad (16)$$

donde y reemplazar en 15 se obtiene:

$$\sum_{k=1}^p \alpha_k \varphi_{\hat{n}}[i, k] = \varphi_{\hat{n}}[i, 0], \quad (17)$$

Estas p ecuaciones con p incógnitas definen una matriz de autocorrelaciones (\emptyset) simétrica y con elementos diagonales de izquierda a derecha constantes, es decir, una matriz de Toeplitz simétrica. Existen dos enfoques básicos para resolver este problema y con seguir los valores de a_k , el método de autocorrelación y el de covarianza (utilizado el presente informe).

Método de Autocorrelación^[4]

En el método de autocorrelación se toma una trama de la señal y se anulan los valores externos a la misma. Luego si se intenta calcular el error por autocorrelación de cada muestra lo que da un incremento considerable del error en los bordes de la trama por tener comparaciones entre la estimación y los valores ceros externos agregados. Es por esto que se realiza toma la trama utilizando una ventana que anule los extremos de la misma, como por ejemplo la ventana de Hamming.

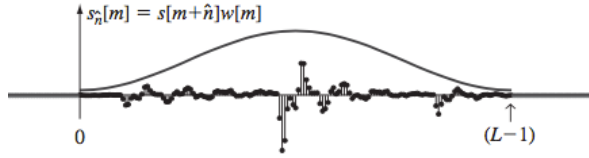


Figura 4: Ventana Hamming aplicado a señal.

Debido a que se anulan los valores externos a la trama es que la ecuación 16 puede escribirse como:

$$\varphi_{\hat{n}}[i, k] = \sum_{m=0}^{L-1+p} s_{\hat{n}}[m-i]s_{\hat{n}}[m-k] \quad \begin{cases} 1 \leq i \leq p \\ 0 \leq k \leq p, \end{cases} \quad (18)$$

que puede ser expresada como:

$$\varphi_{\hat{n}}[i, k] = \sum_{m=0}^{L-1-(i-k)} s_{\hat{n}}[m]s_{\hat{n}}[m+(i-k)] \quad \begin{cases} 1 \leq i \leq p \\ 0 \leq k \leq p. \end{cases} \quad (19)$$

Además, tal como puede verse, en este caso φ es idéntica a la función de autocorrelación de tiempo corto evaluada en $(i-k)$

$$\varphi_{\hat{n}}[i, k] = R_{\hat{n}}[i-k] \quad \begin{cases} 1 \leq i \leq p \\ 0 \leq k \leq p, \end{cases} \quad (20)$$

donde

$$R_{\hat{n}}[k] = \sum_{m=0}^{L-1-k} s_{\hat{n}}[m]s_{\hat{n}}[m+k]. \quad (21)$$

A partir de 20 y 21 se deducen las siguientes ecuaciones:

$$\varphi_{\hat{n}}[i, k] = R_{\hat{n}}[i-k] \quad \begin{cases} 1 \leq i \leq p \\ 0 \leq k \leq p. \end{cases} \quad (22)$$

$$\sum_{k=1}^p \alpha_k R_{\hat{n}}[i-k] = R_{\hat{n}}[i], \quad 1 \leq i \leq p. \quad (23)$$

Por último, la ecuación 23 puede escribirse de forma matricial de la siguiente manera:

$$\begin{bmatrix} R_{\hat{n}}[0] & R_{\hat{n}}[1] & R_{\hat{n}}[2] & \dots & R_{\hat{n}}[p-1] \\ R_{\hat{n}}[1] & R_{\hat{n}}[0] & R_{\hat{n}}[1] & \dots & R_{\hat{n}}[p-2] \\ R_{\hat{n}}[2] & R_{\hat{n}}[1] & R_{\hat{n}}[0] & \dots & R_{\hat{n}}[p-3] \\ \dots & \dots & \dots & \dots & \dots \\ R_{\hat{n}}[p-1] & R_{\hat{n}}[p-2] & R_{\hat{n}}[p-3] & \dots & R_{\hat{n}}[0] \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \dots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_{\hat{n}}[1] \\ R_{\hat{n}}[2] \\ R_{\hat{n}}[3] \\ \dots \\ R_{\hat{n}}[p] \end{bmatrix}$$

Esta matriz de $p \times p$ de los valores de autocorrelación es una matriz de Toeplitz (simétrica y con los elementos diagonales iguales). Para obtener los coeficientes solo es necesario entonces resolver este sistema. A los fines de esta práctica, para resolver este problema se utiliza el algoritmo de Levinson-Durbin^[4].

Por otro lado, el mínimo error del método de autocorrelación es:

$$\mathcal{E}_{\hat{n}} = \phi_{\hat{n}}[0, 0] - \sum_{k=1}^p \alpha_k \phi_{\hat{n}}[0, k], \quad (24)$$

3. MATERIALES Y MÉTODOS

El presente trabajo ha sido implementado en el lenguaje y software de Matlab_R2017b utilizando para esto una computadora Macbook Pro13 con un procesador de 2,5 GHz Intel Core i5, y dos bloques de

memoria RAM de un total de 8GB (1600 MHz DDR3).

Para el desarrollo del trabajo fue necesario implementar distintos algoritmos para la compresión de voz y su posterior síntesis y representación en espectrograma.

3.1 Interfaz Gráfica y Main.

Para poder utilizar los distintos algoritmos de compresión y síntesis de voz se implementa una interfaz gráfica donde puede modificarse el tipo y tiempo de ventana, la cantidad de coeficientes requeridos y el nombre del archivo audio a procesar. Luego se tienen cuatro botones que posibilitan al usuario a realizar el LPC, a sintetizar y exportar la señal, a reproducir y analizar en un espectrograma el archivo de audio de la señal sintetizada y por último un botón que cierre la aplicación.

Para poder enviar la información necesaria a cada función en particular, se implementa una función extra denominada *main*. Esta toma los datos de las opciones antes mencionadas y los envía a la función requerida por el usuario.

3.2 Modelo de predicción lineal.

La primer función a utilizar se encuentra en la interfaz gráfica como LPC. Esta es la encargada de, muestrear la señal con distintos tipos y tiempos de ventanas formando tramas y obtener para cada una de éstas:

- Conjunto de coeficientes
- Clasificación de trama como sonora o sorda
- Ganancia de la trama
- Frecuencia fundamental

Luego esa información es almacenada en una estructura de datos con menor cantidad de bits en relación a la trama original. Por último se exporta la información de todas las tramas a un archivo .txt para simular la transmisión del mensaje a otro dispositivo decodificador.

Cada trama es procesada de diversas formas a fin de determinar los parámetros del modelo. Estos procesos comprenden un filtro pre-énfasis; cálculo de los coeficientes del filtro LPC; cálculo de la Ganancia de la trama; la clasificación, mediante otra función local, de la trama en sonora o sorda para definir el tipo de estímulo y por último el cálculo del pitch de las tramas sonoras. Para terminar, el programa además de exportar el archivo .txt, realiza e informa un cálculo de reducción de cantidad de bits en command window de Matlab.

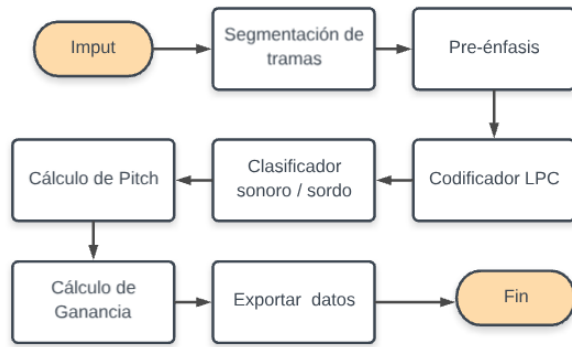


Figura 5: Diagrama de bloques de codificado LPC.

3.2.1 Pre-énfasis.

El espectro típico de la señal de voz tiene una atenuación característica en altas frecuencias debido al efecto de radiación del sonido en los labios^[5]. Esto puede afectar posteriormente en el cálculo de las componentes en frecuencia. Por ello es que se procesa la señal por un filtro pasa altos que incrementa la energía de las altas frecuencias. Es entonces que se diseña un filtro FIR con un cero cercano a la unidad con una ecuación en diferencias dada por:

$$y(n) = x(n) - 0.9378 \cdot x(n-1) \quad (25)$$

3.2.2 Cálculo de coeficientes del filtro.

Tal como se menciona en la introducción teórica, para el cálculo de los coeficientes se emplea el algoritmo de Levinson-Durbin que permite calcular recursivamente la solución al sistema de ecuaciones dado por la matriz de Toeplitz antes mostrada.

Para esto se realiza un ciclo *for* encargado de calcular la autocorrelación de la señal muestreada, desplazándola en p lugares según cuantos coeficientes se deseen obtener. A continuación se utiliza la función interna de Matlab "levinson" que entrega los valores de los coeficientes del filtro LPC.

3.2.3 Cálculo de ganancia.

Para el cálculo de la ganancia de trama se utiliza la siguiente fórmula^[4].

$$G^2 = R(0) - \sum_{k=1}^p a_k R(k) = \varepsilon_p \quad (26)$$

3.2.4 Clasificación de sonora o sorda.

Los parámetros fundamentales que se han tenido en cuenta para realizar esta clasificación son: Función Magnitud Suma (MSF) y Cruces por Cero (ZC). El primero de estos (MSF) es igual a la energía calculada por la trama. Esta función se expresa como:

$$MSF[m] = \sum_{n=m-N+1}^m |s[n]| \quad (27)$$

Debido a que la voz tiene concentrada mayor cantidad de energía en la región de bajas frecuencias es que se usa un filtro pasa bajos con una frecuencia de corte de 800 Hz. Un elevado valor de MSF está relacionado con una trama sonora mientras que un valor bajo con una trama sorda.

Para el segundo parámetro (ZC) la ecuación se define como:

$$ZC[m] = \frac{1}{2} \sum_{n=m-N+1}^m |\text{sgn}(s[n]) - \text{sgn}(s[n-1])| \quad (28)$$

Para una trama de voz sonora, la proporción de cruces por cero es relativamente baja, debido a la presencia de la componente de frecuencia del diapason. Mientras que para sonidos sordos, la proporción de cruces por cero es alta por tener una alta porción de energía en altas frecuencias.

3.2.5 Cálculo de Pitch.

Para el cálculo del pitch de las tramas sonoras se necesita implementar un algoritmo de estimación de período fundamental. El diseño del mismo es complejo y de su correcta determinación depende la calidad de la síntesis. Como en la práctica la trama en realidad no tiene una periodicidad perfecta y el ruido e interferencia siempre están presentes, la estimación del período fundamental se realiza como un compromiso entre la complejidad computacional y la calidad que se desea obtener. En este caso se utiliza un filtro pasa bajos, y se obtiene la frecuencia fundamental mediante autocorrelación de la trama.

3.3 Síntesis de la señal.

En esta segunda función del programa es donde se reciben las características de la señal y se implementa el modelo de producción de voz. Tal como se muestra en la figura dos, se genera, dependiendo de si la trama es sonora o no, un tren de impulsos con la frecuencia correspondiente para sonoras o ruido blanco para tramas sordas. Luego se multiplica por la ganancia y se filtra con los coeficientes del LPC. Por último se concatenan todas las tramas en un solo vector, se calcula el error entre la estimación y el archivo de audio original y se exporta un nuevo archivo de audio con la señal estimada.

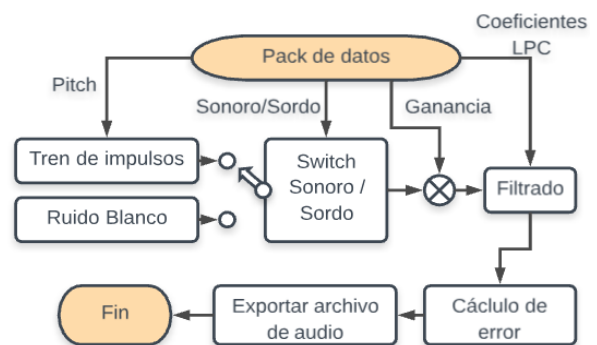


Figura 6: Diagrama de bloques de síntesis de voz con LPC.

3.4 Reproducción y representación.

Por finalizar, se implementa una última función encargada de reproducir el archivo de audio sintetizado y representar tanto este último como el archivo de audio original en dos espectrogramas para su análisis y comparación en respuesta en frecuencia. Para poder realizar esto se utilizan las funciones *sound* y *spectrogram* de Matlab.

4. RESULTADOS Y DISCUSIONES

Los algoritmos implementados en Matlab se prueban con una señal prueba de voz. Se cuantifica el error de predicción y la compresión de datos para distintos tipos y tamaños de ventana y también la influencia de modificar la cantidad de coeficientes en estos resultados. Por otro lado, se grafica un diagrama de polos y se comparan los resultados obtenidos de los coeficientes del filtro LPC con una función interna de Matlab. Además se presentan los espectrogramas de algunas de las señales estimadas.

En la siguiente tabla se muestra la comparación de 12 coeficientes en una trama calculados con el algoritmo implementado y los de la función interna de Matlab.

Tabla 1: Comparación de coeficientes de LPC

Algoritmo	Función Interna	Diferencia
1	1	0
-2.7969	-2.7969	$6,15 \times 10^{-12}$
4.7332	4.7332	$1,91 \times 10^{-11}$
-5.7398	-5.7398	$3,44 \times 10^{-11}$
5.0383	5.0383	$4,24 \times 10^{-11}$
-2.8975	-2.8975	$3,71 \times 10^{-11}$
0.1952	0.1952	$2,07 \times 10^{-11}$
1.8528	1.8528	$6,90 \times 10^{-13}$
-2.4966	-2.4966	$1,40 \times 10^{-11}$
1.8628	1.8628	$1,74 \times 10^{-11}$
-0.9600	-0.9600	$1,14 \times 10^{-11}$
0.2402	0.2402	$4,13 \times 10^{-12}$

Tal y como puede observarse en la columna de diferencias, los resultados obtenidos utilizando ambos métodos dan prácticamente iguales. Por lo tanto si considera que los coeficientes que se consiguen a través de la función interna de Matlab son verídicos, entonces son satisfactorios los coeficientes obtenidos con el algoritmo implementado.

A continuación se presentan dos diagramas de polos del filtro generado a partir de los coeficientes del LPC, el primero de ellos con los 12 polos antes mostrados y el segundo con 40 polos.

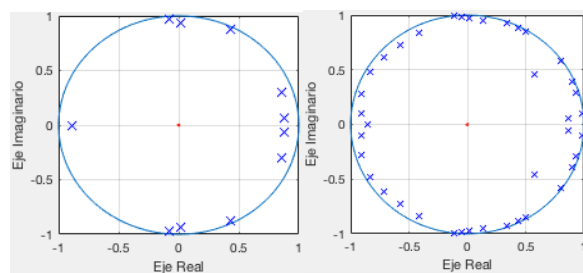


Figura 7: Diagrama de polos de LPC con 12 y 40 coeficientes.

En ambos diagramas puede observarse que todos los polos están dentro del círculo unidad y que

cada polo tiene su conjugado. Esto demuestra la estabilidad del filtro FIR.

A continuación se presentan las diferencias de compresión, y error de síntesis para distintas variaciones de los parámetros de entrada.

Tabla 2: Resultados de compresión y error bajo distintas condiciones

Parámetros de entrada	6 Coeficientes		12 Coeficientes	
	Error	Compresión (veces)	Error	Compresión (veces)
Ventana Hanning	0.1337	49.00	0.1298	29.41
Ventana Rectangular	0.1348	49.00	0.1368	29.41
Ventana Hamming	0.1351	49.00	0.1312	29.41
Ventana de 30 ms	0.1314	147.08	0.1323	88.30
Ventana de 20 ms	0.1381	97.96	0.1289	58.80
Ventana de 10 ms	0.1345	49.00	0.1333	29.40

Tabla 3: Resultados de compresión y error bajo distintas condiciones

Parámetros de entrada	24 Coeficientes		48 Coeficientes	
	Error	Compresión (veces)	Error	Compresión (veces)
Ventana Hanning	0.1293	16.33	0.1291	8.65
Ventana Rectangular	0.1341	16.33	0.1295	8.65
Ventana Hamming	0.1304	16.33	0.1285	8.65
Ventana de 30 ms	0.1329	49.08	0.1312	25.98
Ventana de 20 ms	0.1300	32.67	0.1291	17.30
Ventana de 10 ms	0.1327	16.34	0.1285	8.65

Vale la pena aclarar que para los resultados obtenidos de modificar el tipo de ventana, la cantidad de coeficientes fue de 12 y la duración de la ventana de 10 ms.

Como primer observación, cabe destacar que una vez superados los 12 coeficientes el error asociado a la estimación no disminuye lo suficiente comparado con el costo computacional que eso implica, observar que por otro lado la compresión disminuye drásticamente al aumentar la cantidad de coeficientes. Es por esto que 12 coeficientes ya se considerarían suficientes para estimar la señal de una

manera coherente. En cuanto al tipo de ventana, bajo las mismas condiciones, la ventana tipo Hanning fue la que obtuvo menor error, aunque al escuchar la estimación con los tres tipos de ventanas, la estimación de la ventana Hamming es mejor apreciada. Por otro lado, en cuanto a la duración de la ventana, se observan claros beneficios al reducir el tiempo. Sin embargo, es menester informar que por debajo de los 10 ms la síntesis de la señal comienza a estropearse debido probablemente a problemas de estimación del pitch.

Para finalizar se presentan a continuación tres espectrogramas obtenidos mediante una ventana Hamming de 10 ms modificando la cantidad de coeficientes en 6, 24 y 100.

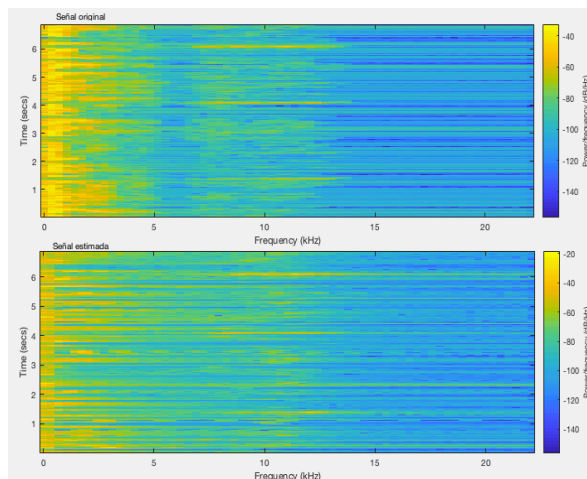


Figura 8: Diagrama de polos de LPC con 12 y 40 coeficientes.

En la figura 8 se encuentra el espectrograma de la señal original en el gráfico superior mientras que en el inferior, el espectrograma de la estimación. Puede observarse una marcada relación en la representación de la distribución energética frecuencial en ambas señales. Además, en la figura 9 puede observarse como la estimación mejora considerablemente al aumentar la cantidad de coeficientes a 24. En este caso la distribución energética se asemeja mucho mas a la de la señal original.

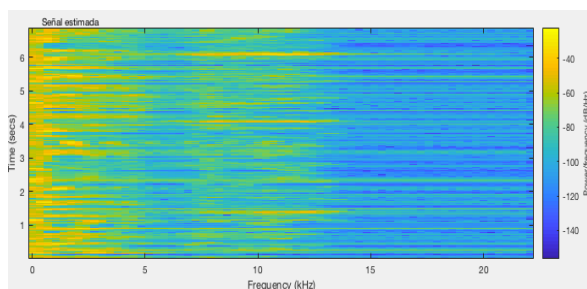
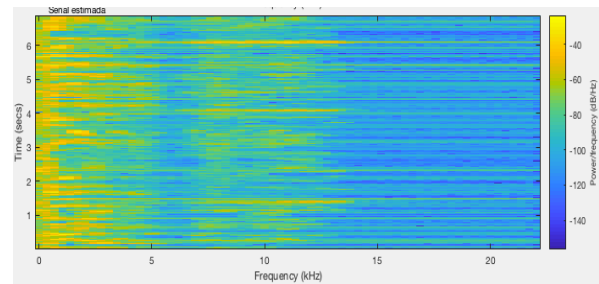


Figura 9: Diagrama de polos de LPC con 12 y 40 coeficientes.

Para finalizar en la figura 10, donde se encuentra el espectrograma de la estimación con 100 coeficientes, puede observarse como una vez superados los 24 coeficientes la distribución energética de la síntesis no tiene cambios significantes, mas no el costo computacional que esto implica. Si se mide el tiempo de síntesis con el algoritmo planteado, se consiguen 0.78 segundos y

2.37 segundos para 24 y 100 coeficientes respectivamente.



5. CONCLUSIONES

Se diseñaron e implementaron los algoritmos necesarios para la codificación y síntesis de voz LPC teniendo en cuenta todas las recomendaciones que ofrece el método LPC. Se comprueba además el correcto funcionamiento del mismo evaluando su efectividad al modificar los parámetros necesarios para la obtención y síntesis de la codificación de señales de voz. A partir de este trabajo se logra obtener una plataforma experimental para la evaluación académica de sistemas de compresión de señales de voz. Se perciben cambios considerables en la transmisión del mensaje al modificar los tiempos de ventana y la cantidad de coeficientes calculados según el tipo de voz que se desea transmitir. Esto es debido a características propias del registro de voz grabado. Para elevar la calidad de síntesis se pretende a futuro investigar en profundidad para conseguir una mejor estimación del Pitch y por otro lado estudiar las características de voz para poder determinar automáticamente el ancho de la ventana y la cantidad de coeficientes necesarios que optimizan la transmisión del mensaje en cada grabación.

6. REFERENCIAS

- [1] Reig Albiñana, David. "Implementación de algoritmos para la extracción de patrones característicos en Sistemas de Reconocimiento De Voz en Matlab". Trabajo Final de Carrera, Universidad Politécnica de Valencia, Escuela Politécnica Superior de Gandia, I.T. Telecomunicaciones. Ciudad de Gandia, España. 2014
- [2] Professor E. Ambikairajah. [UNSWelearning].(04/04/2010) Speech and Audio Processing 3: Linear Predictive Coding (LPC) - Professor E. Ambikairajah [Archivo de video]. <https://www.youtube.com/watch?v=IWH-Oh5KnNY&t=600s&index=4&list=PL8001C3949C24F14E>
- [3] II UNIDAD: Algoritmos de compresión de imágenes, audio y video. Recuperado de: <https://sistemamultimedia.wordpress.com/ii-unidad/>
- [4] Rabiner, Lawrence R. y Schafer, Ronald W. – Theory and Applications of Digital Speech Processing – Pearson 2011
- [5] Haykin, S., Communication Systems, 4º ed., Nueva York, Jhon Wiley & Sons, 2008.