



Universidad de Chile
 Facultad de Ciencias Físicas y Matemáticas.
 Departamento Ingeniería Eléctrica.



Capítulo 2.

Procesamiento Digital de Señales.

2.0 Transformada de Fourier

2.1 Muestreo de la Señal.

2.2 Análisis de Señales Discretas.

2.3 Filtros Digitales

2.4 DTF y FFT


2.5 Ventana de Hamming

2.6 LPC

2.7 Cuantización Vectorial.

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra



2.0 Transformada de Fourier

• **Significado matemático:**


Representación de una señal dada en términos de una suma infinita de exponenciales complejas ponderadas cada una por $F(\omega)$.

• **Significado en aplicaciones:**

Análisis de la señal en un dominio (frecuencia) en donde ciertas características son mucho más visibles que en el tiempo.

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra




2.0 Transformada de Fourier

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) \cdot e^{j\omega t} d\omega$$

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \cdot e^{-j\omega t} dt$$

EM756 Procesamiento de la Voz

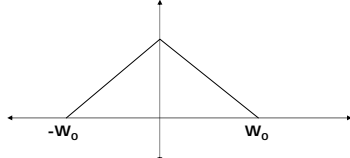
Profesor: Néstor Becerra



2.1. Muestreo de la señal.

○ Consideremos una señal $s(t)$, donde t es el tiempo.

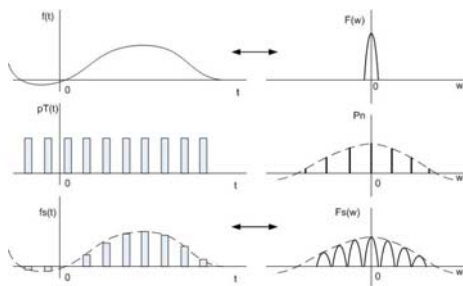
$$s(t) \leftrightarrow S(\omega)$$



EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.1 Muestreo de la señal

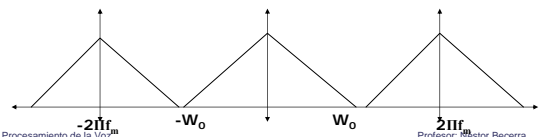


EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.1. Muestreo de la señal

- Supongamos que la señal $s(t)$ es muestreada a intervalos regulares T ($f_m = 1/T$, f_m es la frecuencia de muestreo). El espectro de la señal muestreada es representado por:



EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.1 Muestreo de la señal

- Para que no haya sobre posición,

$$2\pi f_m > 2W_0 = 2 \cdot 2\pi f_0$$

$$f_m > 2f_0$$

f_0 Es la máxima frecuencia de la señal.

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.2 Análisis de Señales Discretas

- Una señal muestreada podría ser representada por $S(nT)$, donde n es un número entero y T es el período de muestreo.
- En realidad las señales muestreadas se pueden representar como secuencias de números enteros (digitalizadas).

$$s(0), s(1), s(2), s(3), \dots$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.2 Análisis de Señales Discretas

- Todos los tipos de procesamiento en electrónica analógica se pueden implementar en el dominio discreto.
Ejemplos: Amplificadores, filtros, multiplicadores, etc.
- Otras funciones más complejas solo se pueden realizar mediante programas de computadores (procesamiento digital de señales).

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.3 Filtros Digitales

- Dos tipos: FIR, IIR

FIR (Finite Impulse Response):

- La salida es función únicamente de la entrada.
- El polinomio que describe este filtro:

$$Y(n) = a_0x(n) + a_1x(n-1) + a_2x(n-2) + \dots + a_kx(n-k)$$

- Un filtro FIR también se puede definir por:

$$A = [a_0 \quad a_1 \quad a_2 \quad \dots \quad a_k]$$

¿Porqué Respuesta Impulsiva Finita?

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.3 Filtros Digitales

IIR (Infinite Impulse Response):

- La salida en n depende de la entrada y de la salida hasta $n-1$.
- El polinomio que describe este filtro:

$$Y(n) = a_0x(n) + a_1x(n-1) + a_2x(n-2) + \dots + a_kx(n-k) + b_1y(n-1) + b_2y(n-2) + \dots + b_Ly(n-L)$$

- Un filtro IIR también se puede definir por:

$$A = [a_0 \quad a_1 \quad a_2 \quad \dots \quad a_k] \quad \text{y} \quad B = [b_1 \quad b_2 \quad b_3 \quad \dots \quad b_L]$$

¿Porqué Respuesta Impulsiva Infinita?

¿Estabilidad?

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

Anexo: Transformada Z

Se emplea para analizar sistemas discretos.
(De forma similar como opera Laplace para sistemas continuos).

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}$$

Mediante las propiedades de la transformada Z se puede concluir:

$$x(n) \leftrightarrow X(z)$$

$$x(n-k) \leftrightarrow X(z)z^{-k}$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

Anexo: Transformada Z

Ejemplo:

Considere el filtro FIR

$$Z\{y(n)\} = Z\{a_0x(n) + a_1x(n-1) + a_2x(n-2) + \dots + a_kx(n-k)\}$$

$$Z\{y(n)\} = Z\{a_0x(n)\} + Z\{a_1x(n-1)\} + Z\{a_2x(n-2)\} + \dots + Z\{a_kx(n-k)\}$$

$$Z\{y(n)\} = a_0Z\{x(n)\} + a_1Z\{x(n-1)\} + a_2Z\{x(n-2)\} + \dots + a_kZ\{x(n-k)\}$$

$$Y(z) = a_0X(z) + a_1X(z)z^{-1} + a_2X(z)z^{-2} + \dots + a_kX(z)z^{-k}$$

$$\frac{Y(z)}{X(z)} = a_0 + a_1z^{-1} + a_2z^{-2} + \dots + a_kz^{-k} = H(z)$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

Anexo: Respuesta en Frecuencia

- La respuesta en frecuencia de un sistema discreto se puede estimar reemplazando Z por:

$$z = e^{+j\omega} = \cos(\omega) + j\sin(\omega)$$

Ejemplo: Sea $H(z) = 1 + az^{-1}$

$$H(e^{+j\omega}) = 1 + ae^{-j\omega} = 1 + a[\cos(\omega) - j\sin(\omega)]$$

$$H(e^{+j\omega}) = 1 + a \cdot \cos(\omega) - j \cdot a \cdot \sin(\omega)$$

$$|H(e^{+j\omega})| = \sqrt{1 + 2a \cos(\omega) + a^2}$$

$$\angle H(e^{+j\omega}) = \arctan\left(\frac{-a \sin(\omega)}{1 + a \cos(\omega)}\right)$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.4 DFT y FFT

2.4.1 DFT (Discrete Fourier Transform)

Una secuencia de N muestras de valor complejo en el dominio de la frecuencia dado por:

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn} \quad 0 \leq k \leq N-1$$

Donde:

$$W_N = e^{-j\frac{2\pi}{N}}$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.4 DFT y FFT

2.4.1 DFT (Discrete Fourier Transform)

La transformada inversa se define como:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-kn} \quad 0 \leq n \leq N-1$$

Donde:

$$W_N = e^{-j\frac{2\pi}{N}}$$

- X(k) es una función periódica con periodo N.
- Las propiedades de la DFT son similares a las propiedades de la transformada de Fourier análoga.

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.4 DFT y FFT

2.4.2 FFT(Fast Fourier Transform)

- La DFT requiere N^2 multiplicadores.
- Empleo de simetría.
- Algoritmo mas eficiente de cálculo.
- Calcula N componentes de frecuencia a partir de N muestras de tiempo para $N=2^r$, con r cualquier entero positivo.

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.5 Ventana de Hamming

•Para reducir el efecto de discontinuidad en los bordes al mínimo debemos emplear tipos de ventana que tiendan a reducir a 0 los valores de las muestras en los extremos.

•Aunque existen varios tipos de ventana, una de las más comunes en el análisis de la voz es la que se conoce como ventana de Hamming.

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.5 Ventana de Hamming

$$JH(n) = \begin{cases} 0.54 - 0.46 \cos\left[\frac{2\pi n}{N-1}\right] & 0 \leq n < N \\ 0 & n < 0 \dots N \leq n \end{cases}$$

PREGUNTA:

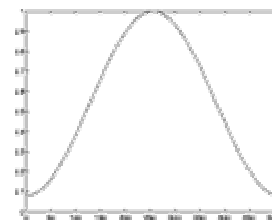
Que ancho de ventana de análisis le parece apropiado?

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.5 Ventana de Hamming

Forma genérica de la ventana de Hamming



EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

- En 1927 se propuso que una serie temporal de muestras correlacionadas podría ser generada a partir de una serie de muestras estadísticamente independientes (ruido blanco) procesadas a través de un filtro lineal.
- Existen métodos distintos de modelamiento que están basados en procesos estocásticos y/o procesos AR(auto regresivos).

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

Modelo Auto Regresivo.

- Es un proceso estocástico genérico.
- La idea principal se basa en que la voz puede modelarse a través de una combinación lineal de p muestras anteriores más una señal de excitación o ruido blanco.

$$x[n] = \sum_{k=1}^p a_k x[n-k] + e[n]$$

Donde $x(n)$ Coeficientes de la señal
 a_i Habitualmente son LPC
 $e[n]$ Ruido blanco

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

Aplicando transformada Z, a la formula anterior, también conocida como error de predicción.

$$x[n] = \sum_{k=1}^p a_k x[n-k] + e[n]$$

$$X(z) = H_I(z)E(z)$$

$$H_I = \frac{1}{\sum_{k=0}^p a_k z^{-k}}$$

$H_I(z)$ corresponde a un filtro IIR

Considerando:

$$\begin{aligned} a_0 &= 1 \\ a_k &= -\alpha_k \\ 1 \leq k \leq p \end{aligned}$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

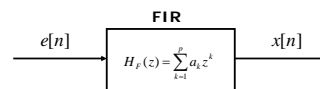
2.6 LPC (Linear Predictive Coding)

Modelo Auto Regresivo.

Un proceso regular genérico esta representado:

$$x[n] = \sum_{k=0}^p b_k e[n-k]$$

$$X(z) = H_F(z)E(z)$$



H_F corresponde a un filtro FIR de fase mínima y puede ser reemplazado por un filtro IIR como el H_I .

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

Modelo Auto Regresivo.

Un análisis de LPC puede ser vista como un modelamiento AR de una señal de voz en intervalos de tiempo donde la señal se puede considerar estacionaria. La señal multiplicada por una ventana de tiempo centrada en l esta dada por:

$$x(n, l) = v(n)w(l - l)$$

Simplificando la notación:

$$x(n) = v(n)w(n)$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

Para determinar los coeficientes del LCP se trabaja a partir de la ecuaciones de Yule-Walker. (Basadas error cuadrático medio)

$$E = \sum_{n=-\infty}^{\infty} e^2(n) = \sum_{n=-\infty}^{\infty} \left[x(n) - \sum_{k=1}^p \alpha_k x(n-k) \right]^2$$

$$\frac{\partial E}{\partial \alpha_k} = 0 \quad k = 1, 2, 3, \dots, p$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

Luego se tiene p ecuaciones lineales:

$$\sum_{n=-\infty}^{\infty} x(n-i)x(n) = \sum_{k=1}^p \alpha_k \sum_{n=-\infty}^{\infty} x(n-i)x(n-k) \quad i = 1, 2, 3, \dots, p$$

Definimos:

$$R(i) = \sum_{n=i}^{N-1} x(n)x(n-i) \quad i = 1, 2, 3, \dots, p$$

Reemplazando:

$$\sum_{k=1}^p \alpha_k R(i-k) = R(i) \quad (**)$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

De las ecuaciones anteriores tenemos que, la energía residual mínima o el error de predicción mínima E_p para un modelo de p polos:

$$E_p = R(0) - \sum_{k=1}^p \alpha_k R(k)$$

La ecuación (**) en forma matricial:

$$\begin{bmatrix} R(0) & R(1) & R(2) & \dots & R(p-1) \\ R(1) & R(0) & R(1) & \dots & R(p-2) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ R(p-1) & R(p-2) & R(p-3) & \dots & R(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(p) \end{bmatrix}$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

El sistema escrito en forma matricial se define:

$$R * A = r$$

Donde R matriz de $p \times p$. Invertiendo la matriz R se puede obtener el vector A de coeficientes del LPC.

Sin embargo existe un método más eficiente, que consiste en aprovechar las simetrías de la matriz.

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

Método de Levinson-Durbin

$$k_m = \frac{R(m) - \sum_{k=1}^{m-1} \alpha_k^{m-1} R(m-k)}{E_{m-1}}$$

$$\alpha_m^m = k_m$$

$$\alpha_k^m = \alpha_k^{m-1} - k_m \alpha_{m-k}^{m-1}$$

$$1 \leq k \leq m-1$$

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.6 LPC (Linear Predictive Coding)

Método de Levinson-Durbin

$$E_m = (1 - k_m^2) \cdot E_{m-1}$$

Donde inicialmente: $E_0 = R(0)$
 $\alpha_0 = 0$

Donde α_k^m indicando el coeficiente LPC en la m-ésima iteración.

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.7 Cuantización Vectorial

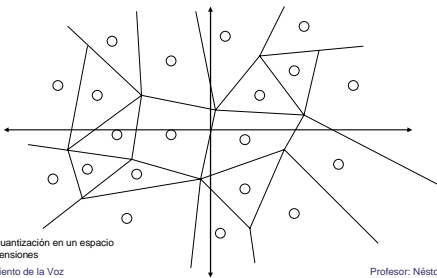
- La “**cuantización vectorial**” es un proceso de aproximación por el cual una señal cuya amplitud presenta valores continuos es representada por una nueva señal cuya amplitud se puede asumir con valores discretos.
- La cuantización es una herramienta muy útil para codificación pues el número de bits a representar es una variable que puede ser reducida cuanto se quiera cumpliendo una exigencia de distorsión máxima exigida.

EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.7 Cuantización Vectorial

$$q(T) = z_i \mid d(X, z_i) \leq d(X, z_j)$$



Ejemplo de cuantización en un espacio de varias dimensiones
EM756 Procesamiento de la Voz

Profesor: Néstor Becerra

2.7 Cuantización Vectorial

- Un conjunto de todos los z_i $1 \leq i \leq L$ es denominado el codebook (Libro de código).
- Un método comúnmente utilizado en la elaboración de un codebook es el algoritmo "**k-means**". Este método emplea una minimización de la distorsión global, definida como:

$$D = \sum_{i=1}^L D_i$$

Donde D_i es la distorsión dentro de la celda i .

2.7 Cuantización Vectorial

Donde $d(T, z_i)$ es la distancia entre un patrón de teste localizado en la celda C_i

$$D_i = \frac{1}{N_i} \sum_{T \in C_i} d(T, z_i)$$

$$z_i = \frac{1}{N_i} \sum_{T \in C_i} T$$

2.7 Cuantización Vectorial

- **Algoritmo de K-means.**

Paso1. Inicialización, escoger utilizando un método adecuado un codebook.

Paso2. Clasificación, clasificar cada vector T (cuadro)...completar....

Paso3. Actualización de Codebook recalculando cada code-word a través de la media aritmética de los elementos no interiores a la celda correspondiente.



2.7 Cuantización Vectorial

- **Algoritmo de K-means.**

Paso 4. Test de Convergencia completar.