



UNIVERSIDAD AUTÓNOMA  
DEL ESTADO DE HIDALGO



INSTITUTO DE CIENCIAS BÁSICAS E INGENIERÍA

CENTRO DE INVESTIGACIÓN EN TECNOLOGÍAS DE INFORMACIÓN Y SISTEMAS

---

INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

PROCESAMIENTO DIGITAL DE SEÑALES SÍSMICAS MEDIANTE LA  
TÉCNICA WAVELET

T E S I S

QUE PARA OBTENER EL TÍTULO DE INGENIERO EN  
ELECTRÓNICA Y TELECOMUNICACIONES

**PRESENTA:**

**JORGE ALBERTO DIAZ NERIA**

ASESORES:

DR. LUIS ENRIQUE RAMOS VELASCO

DR. ROBERTO ARMANDO HERNÁNDEZ GÓMEZ

PACHUCA HGO., MÉXICO 8 DE ABRIL DE 2008



Jorge Alberto Diaz Neria

**Procesamiento digital de  
señales sísmicas mediante la  
técnica wavelet**

*El presente trabajo,  
esta dedicado a mis padres,  
a quienes debo lo que ahora soy,  
porque me han brindado los medios para concluir  
de manera exitosa todos mis proyectos.  
A mis hermanos, por ser mis compañeros,  
ser uno de mis pilares de apoyo y por compartir  
conmigo sueños y proyectos.  
Por todo esto y mucho más, gracias.*

Gracias a

**PROMEP** por el apoyo recibido a través  
del proyecto P/CA-11 2006-14-18.

**UAEH** por los proyectos del programa  
anual de investigación con números  
PAI-31A.

# Agradecimientos

Siempre he soñado con lograr poder ser alguien que llene de orgullo y alegría a todas las personas que me rodean y que me han enseñado tantas cosas en mi corta vida ahora, he alcanzado una de mis metas pero no lo he logrado solo. Es tiempo de mostrar lo agradecido que estoy con todas las personas que me han dado su apoyo para la realización de esta meta y este trabajo, y es en estas páginas donde muestro su gran labor y sobre todo, la admiración y cariño que siento por todos ustedes.

Agradezco a esta, nuestra máxima casa de estudios la Universidad Autónoma del Estado de Hidalgo, y a los profesores que me han enseñado a formar en mí un profesional; por ser la fuente del saber que he adquirido en este tiempo y que me han permitido culminar de manera exitosa mis estudios de licenciatura.

A Dios, por darme la oportunidad de vivir y crecer en una familia tan grande y amorosa...

Por sobre todo a mis padres Mary y Rigo, quienes con sus oraciones y apoyo incondicional me han ayudado a ver siempre hacia adelante, por darme la vida y volver a dármele, por sus consejos y por darme la más grande herencia que pueda recibir...una carrera profesional; a mis hermanos Gabi y Dani por ser mis amigos, por esas horas de juego y de desvelo que han surgido en nuestras vidas, por compartir conmigo penas y alegrías y sobre todo, por el amor que me hacen sentir dentro de mi corazón, por estar a mi lado en los momentos difíciles, siempre dándome fuerzas para seguir adelante.

A mi gran familia; por todo su apoyo tanto en lo económico, moral y profesional en especial a mis abuelitos Eufemio y Pepi, Rosi y Gabi; a las familias Neria Gress, Monge Neria, Cruz Neria, Neria Hernández, Gutiérrez Neria, Neria Ruelas, Díaz Gutiérrez, Díaz Gálvez y mi tío Arturo Díaz...mil gracias.

En la elaboración y redacción de esta tesis he de agradecer de especial manera a mis asesores Dr. Luis Enrique Ramos Velasco y Dr. Roberto Armando Hernández Gómez, por todo el valioso tiempo que me brindaron para desarrollar este trabajo de investigación, por su paciencia, sus comentarios y enseñanzas. También al M. en C. Iván Razo por compartir un poco de su saber, que dió lugar a la realización de este proyecto.

Gracias...a todas las amistades ganadas durante mi carrera universitaria; en especial a: Nataly, Oscar, Ulises, Pawys, Neo, Abi y Cheque por su respaldo incondicional y la motivación que me dieron en los momentos que lo necesitaba; mi infinita gratitud a Susas, Mir, Paco, Aldo, Carmelo, Raúl, Maya y Ana por innumerables momentos

en los que disfrute de su compañía, comprensión, lealtad, horas de desvelo y diversión...a quienes no solo han sido mis amigos, sino también mis compañeros durante largos años, mis cuates Juan, Cheko y Cristian; al Klan, muy especialmente a Cynthia, Alma, Eliza, J Eliza, Isis, Migda, Alex y Julito por darme cabida en este gran grupo de camaradas que está siempre en las buenas y en las malas, ya que con ustedes no sólo conocí lugares en los viajes que realizamos, sino el gran valor que tiene cada momento que pasamos juntos y que son la chispa que nos une. Asimismo, quiero decirles que aunque la distancia nos separe se que este valioso tesoro perdurará por siempre; a mi cuñis Ange y a Nalle; agradezco el cariño y aprecio que sienten por mi.

A uno de mis mejores amigos el Pbro. Abdías A. Castillo por esas horas de plática, por compartir conmigo sus experiencias, por alentarme a ser cada día un mejor ser humano y mantenerme firme en mis principios, por su cariño y oportunidad de llamarle tío.

Ahora mostrarle a ese ser tan importante y especial que llego a mi vida cuando menos lo esperaba lo agradecido que me siento con ella, por que vino a darle un nuevo sentido a mi vida con su forma de ser, su mirada y sonrisa; quien me impulsa a mejorar cada día...gracias Ary.

Finalmente, pero no menos importante; a mis amigos del coro Luz y Vida; por sus buenos deseos para conmigo, su apoyo y comprensión ya que en varias ocasiones me tendieron su mano cuando no pude acompañarlos, el llevar a cabo uno de mis más grandes deseos...tocar la batería. Muchas gracias chavos: Marco, Angel, John, Bray, Felipe, Armando, Geovanni, Aure y chicas: Moni, Sandra, Nubis, Alida, Blanca, Gaby, Magos, Nadia, Marisol, Gaby, Gladis, Tommy y Edna

# Resumen

## Procesamiento digital de señales sísmicas mediante la técnica wavelet

En esta tesis se realizan experimentos empleando un DSP de la familia Texas Instruments *TMS320C6711* para el análisis de señales sísmicas con el fin de ofrecer nuevas alternativas para su tratamiento. Lo obtenido en los resultados, permite ver la factibilidad de este proyecto.

Cabe resaltar que para la realización de este proyecto de investigación se utilizaron nuevas técnicas, como lo es el uso de herramientas matemáticas tales como la transformada wavelet; que ofrece un mejor análisis y estudio de la señal.

La utilización de estas técnicas de estudio lleva a una mejor comprensión y desglosamiento de información mas detallada. De esta manera se puede ofrecer un nuevo panorama en el análisis de señales; en este caso, de señales de origen sísmico las cuáles al día de hoy tienen pocas formas de análisis que muestren sus diversas características.

La señal sísmica que es empleada se obtuvo de Virgo seismic monitor [2], la cuál se encuentra en formato *.dat*, dicha señal es procesada tanto en simulación numérica en MatLab y en emulación por el DSP.

# Índice general

Dedicatoria . . . . .	III
Agradecimientos . . . . .	VII
Resumen . . . . .	IX
Índice general . . . . .	XI
Índice de figuras . . . . .	XV
Índice de cuadros . . . . .	XVIII
<b>Acrónimos</b>	<b>XIX</b>
<b>Notación</b>	<b>XXI</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Objetivos de la tesis . . . . .	2
1.2. Planteamiento del problema . . . . .	2
1.3. Solución propuesta . . . . .	2
1.4. Alcances y limitaciones . . . . .	3
1.5. Justificación . . . . .	3
1.6. Sismo . . . . .	4
1.6.1. Registro de sismos . . . . .	6
1.6.2. Análisis de las señales sísmicas . . . . .	7
1.7. Procesador digital de señales . . . . .	8
1.8. Organización del trabajo . . . . .	9
<b>2. Ondas sísmicas</b>	<b>11</b>
2.1. Qué es un sismo y sus tipos de ondas . . . . .	11
2.2. Ondas de cuerpo . . . . .	14
2.2.1. Ondas P . . . . .	14
2.2.2. Ondas S . . . . .	15
2.2.3. Ondas convertidas . . . . .	16
2.2.4. Coda . . . . .	19
2.3. Ondas superficiales . . . . .	19



2.3.1.	Ondas de Rayleigh . . . . .	20
2.3.2.	Ondas de Love . . . . .	22
2.4.	Ondas guiadas . . . . .	22
2.5.	Modos propios de la tierra . . . . .	23
2.6.	Resumen . . . . .	23
<b>3.</b>	<b>Transformadas wavelet</b>	<b>25</b>
3.0.1.	Ondas y ondoletas (wavelets) . . . . .	25
3.0.2.	Análisis tiempo-frecuencia . . . . .	27
3.0.3.	Transformadas . . . . .	27
3.0.4.	Tipos de transformadas . . . . .	28
3.0.5.	Tipos de transformadas wavelet . . . . .	29
3.0.6.	Notación y definiciones . . . . .	30
3.1.	Transformada wavelet continua . . . . .	30
3.1.1.	Definición . . . . .	31
3.1.2.	La interpretación de bancos de filtros . . . . .	32
3.2.	La expansión en series wavelet . . . . .	35
3.2.1.	Wavelets díadicas . . . . .	35
3.2.2.	Definición . . . . .	35
3.2.3.	Wavelets díadicas compactas . . . . .	36
3.2.4.	Ejemplo: La transformada Haar . . . . .	37
3.3.	La transformada wavelet discreta . . . . .	38
3.3.1.	Teoría de banco de filtros . . . . .	39
3.3.2.	Filtros pasabanda ideales . . . . .	39
3.3.3.	Filtros pasabanda suaves . . . . .	42
3.3.4.	Análisis multiresolución . . . . .	43
3.3.5.	Codificación subbanda . . . . .	45
3.3.6.	La media banda inferior . . . . .	45
3.3.7.	La media banda superior . . . . .	48
3.3.8.	Translapando la media banda superior . . . . .	49
3.3.9.	Codificación y decodificación subbanda . . . . .	51
3.3.10.	El algoritmo de la transformada wavelet rápida . . . . .	51
3.3.11.	Funciones base . . . . .	53
3.3.12.	Diseño de la transformada wavelet discreta . . . . .	53
3.3.13.	Filtros espejo . . . . .	54
3.3.14.	El vector de escalamiento . . . . .	55
3.3.15.	El vector wavelet . . . . .	57
3.3.16.	Calculando la transformada wavelet . . . . .	58
3.3.17.	Ejemplos . . . . .	59
3.3.18.	Wavelets con soporte compacto ortonormal . . . . .	60

3.4. Transformadas wavelet biortogonales . . . . .	62
3.4.1. Implementación . . . . .	62
3.4.2. Wavelets biortogonales . . . . .	63
3.4.3. Construyendo wavelets biortogonales . . . . .	63
3.4.4. Selección de la wavelet . . . . .	64
<b>4. Resultados obtenidos</b>	<b>67</b>
4.1. La señal sísmica empleada . . . . .	67
4.1.1. Naturaleza de la señal . . . . .	68
4.2. Transformadas de Fourier y wavelet . . . . .	69
<b>5. Conclusiones</b>	<b>79</b>
<b>Glosario</b>	<b>81</b>
<b>Bibliografía</b>	<b>83</b>
<b>A. Transformadas</b>	<b>87</b>
A.1. Transformada Z . . . . .	87
A.2. Relación entre el plano Z y el plano S . . . . .	90
A.3. La transformada corta de Fourier (STFT) . . . . .	90
A.3.1. Resolución tiempo - frecuencia. . . . .	92
A.4. Transformada rápida de Fourier (FFT) . . . . .	95
A.4.1. Cálculo directo de la DFT . . . . .	97
A.4.2. Metodología para el cálculo de la DFT (“Divide y vencerás”) . . . . .	98
<b>B. Álgebra lineal</b>	<b>99</b>
B.1. Espacios vectoriales de dimensión finita . . . . .	99
B.1.1. Espacios con producto interno . . . . .	105
B.1.2. Espacios vectoriales de dimensión infinita . . . . .	107
<b>C. Especificaciones técnicas del DSP TMS320C6711</b>	<b>113</b>
C.1. Arquitectura de los dispositivos	
TMS320 C62x/C67x . . . . .	115
C.1.1. Unidad de procesamiento central (CPU) . . . . .	115
C.1.2. Caminos de datos del CPU . . . . .	117
C.1.3. Archivos de registros de propósito general ( <i>register files</i> ) . . . . .	117
C.1.4. Unidades funcionales . . . . .	119
C.1.5. Archivos de registros de control del TMS320 C62x/C67x . . . . .	119
C.1.6. Caminos entre archivos de registros ( <i>Register Files Cross Paths</i> ) . . . . .	119
C.1.7. Caminos de memoria, cargas y almacenamiento . . . . .	122

C.1.8. Caminos de direccionamiento de datos . . . . .	122
C.1.9. Mapeo entre instrucciones y unidades funcionales . . . . .	124
C.2. Modos de direccionamiento . . . . .	124
C.3. Interrupciones . . . . .	125
C.4. Periféricos . . . . .	126
<b>D. Programas realizados en el DSP y en MatLab</b>	<b>131</b>
D.1. Desarrollo de programas en MatLab . . . . .	131
D.2. Formato de archivos .dat . . . . .	132
D.3. Implementación de programas en el DSP . . . . .	134

# Índice de figuras

1.1. Localización de un epicentro. . . . .	5
1.2. Ondas P y S. . . . .	5
1.3. Principio básico de operación y registro de un sismógrafo que mide el desplazamiento vertical del terreno. . . . .	6
1.4. Conociendo la distancia ( $d$ ) de una estación sismológica al epicentro, ésta puede representarse como un círculo de radio $d$ con centro en dicha estación. Si se cuenta con observaciones al menos en tres estaciones, el punto de intersección ( $E$ ) de los círculos corresponde al epicentro. . .	7
2.1. Frentes de onda circulares propagándose hacia afuera. Las líneas punteadas (rayos), perpendiculares a los frentes de onda, indican la trayectoria de propagación. . . . .	12
2.2. Ondas senoidales sumadas para obtener un pulso. . . . .	13
2.3. Onda compresional propagándose a lo largo de un resorte con velocidad $v$ . $C$ indica compresión y $D$ indica dilatación. El desplazamiento de las partículas del resorte se produce en las direcciones indicadas por $d$ . . .	14
2.4. Onda de cizalla propagándose con velocidad $v$ a lo largo de una cuerda. El desplazamiento de las partículas de la cuerda se da en las direcciones indicadas por $d$ . . . . .	15
2.5. La onda S y sus componentes $SV$ y $SH$ . . . . .	16
2.6. $i$ rayo incidente, $r$ rayo reflejado, $R$ rayo refractado. . . . .	17
2.7. Fuente sísmica. . . . .	18
2.8. Telesismo registrado en sismómetros vertical ( $Z$ ) y horizontal en dirección Norte-Sur ( $N$ ) y Este-Oeste ( $E$ ). En (a) $d$ indica cómo es el desplazamiento de una partícula de la superficie de la Tierra al paso de una onda Rayleigh con velocidad $v$ . . . . .	19
2.9. Velocidades de grupo para ondas Rayleigh. . . . .	20
2.10. Velocidades de grupo para ondas Love. . . . .	21
2.11. Amplitudes de los modos propios de vibración para el modo fundamental y los dos primeros superiores de una onda de Rayleigh. . . . .	21

2.12. Modos propios de la Tierra. (a) Modos volumétricos: modo fundamental y los dos primeros modos superiores. (b) Primeros dos modos de cizalla. . . . .	24
3.1. Ondas y ondoletas. . . . .	26
3.2. Espacio tiempo-frecuencia: (a) señal, (b) representación. . . . .	27
3.3. Una wavelet. . . . .	32
3.4. Analogía de banco de filtros para la transformada integral wavelet de una señal. . . . .	34
3.5. Escalamiento binario y translación díadica de una wavelet. . . . .	35
3.6. Funciones base la transformada Haar. . . . .	38
3.7. Señal compuesta que contiene dos tonos bursts y ruido aleatorio: (a) los tres componentes, (b) espectro de amplitud y fase. . . . .	40
3.8. Generando una serie de filtros pasabanda por división del eje de frecuencia: (a) respuestas impulso, (b) funciones de transferencia. . . . .	41
3.9. Implantación de un banco de filtros pasabanda. . . . .	41
3.10. Salidas del filtro pasabanda. . . . .	42
3.11. Filtros pasabanda pasivos: (a) respuestas impulso ; (b) funciones de transferencia. . . . .	43
3.12. Salida del banco de filtros pasabanda pasivos. . . . .	44
3.13. Codificación subbanda, la media banda inferior: (a) una señal muestreada y su espectro de banda limitado, (b) el filtro pasabajas de media banda ideal, (c) filtrado pasabajas de la señal, (d) la función submuestreada, (e) muestreo singular reemplazado con ceros, (f) muestreo singular de puntos descartados. . . . .	46
3.14. Codificación subbanda, la media banda superior: (a) una señal muestreada y su espectro de banda limitado, (b) el filtro pasaaltas de media banda ideal, (c) filtrado pasaaltas de la señal, (d) la función submuestreada, (e) muestreo singular reemplazado con ceros, (f) muestreo singular de puntos descartados. . . . .	50
3.15. Translapando la media banda superior: (a) translapando el espectro en la Figura 3.14c, (b) resultado del translaje. . . . .	50
3.16. Codificación subbanda de dos bandas y reconstrucción. . . . .	52
3.17. El algoritmo de la transformada wavelet discreta. . . . .	52
3.18. La transformada wavelet discreta inversa. . . . .	53
3.19. Construyendo la función de escalamiento (después Daubechies). . . . .	56
3.20. Daubechies, (a) escalamiento de la función y (b) wavelet para $r = 2$ . . . . .	60
3.21. Wavelets ortonormales para: (a) $r = 3$ , (b) $r = 5$ , (c) $r = 7$ , y (d) $r = 9$ . . . . .	61
3.22. Sucesiones de filtros discretos para las wavelets ortonormales en la Fig.3.20. . . . .	61

3.23. Paso de una descomposición a un paso de reconstrucción de la transformada wavelet biortogonal. . . . .	63
3.24. Sucesiones de filtros discretos para las wavelets biortonormales (de [11] y [29]). . . . .	64
3.25. Ejemplos de wavelets biortogonales: (a) wavelet pirámide Laplaciana, (b) función wavelet spline lineal, (c) wavelet de fase lineal de 18 puntos. 65	65
A.1. Señal $x(t)$ y función tiempo-ventana centrada en $\tau$ . . . . .	92
A.2. (a) Representación de la señal $x(t)$ ; (b) Contenido espectral de la señal obtenido mediante la transformada rápida de Fourier. . . . .	93
A.3. (a) Representación de la señal $x_1(t)$ ; (b) Contenido espectral de la señal obtenido mediante la FFT, se observa que las amplitudes de ambas componentes han disminuido a la mitad de su valor real. . . . .	94
A.4. Representación tiempo - frecuencia con buena resolución en tiempo y mala resolución en frecuencia. . . . .	95
A.5. Representación tiempo - frecuencia con buena resolución en tiempo y mala resolución en frecuencia. . . . .	95
A.6. Función wavelet correspondiente a la familia Daubechies. . . . .	96
C.1. Diagrama de bloques del TMS320C62x/C67x. . . . .	116
C.2. Camino de datos del TMS320C67x. . . . .	118
C.3. Diagrama de bloques con los periféricos de los dispositivos 'C6211/'C6711.126	126

# Índice de cuadros

C.1. Unidades Funcionales y las operaciones realizadas. . . . .	120
C.2. Registros de Control . . . . .	121
C.3. Registros de control extendido para el TMS320C67x . . . . .	122
C.4. Mapeo de instrucciones de punto fijo y unidades funcionales . . . . .	123
C.5. Mapeo de instrucciones de punto flotante y unidades funcionales . . . . .	124
C.6. Generación de direccionamiento indirecto para Load/Store . . . . .	125
C.7. Periféricos de los dispositivos TMS320C6000 . . . . .	127
D.1. MATLAB: Visualización de la señal. . . . .	131
D.2. MATLAB: Programa para obtener el espectro de frecuencia de una señal. . . . .	131

# Acrónimos

<b>Siglas</b>	<b>Descripción</b>
CCS	Code Composer Studio, entorno de desarrollo para el DSP.
CPU	Unidad Central de Procesamiento.
CWT	Transformada Wavelet Continua.
DMA	Acceso Directo a Memoria.
DSP	Procesador Digital de Señales.
DWT	Transformada Wavelet Discreta.
FLOPS	Operaciones en Punto Flotante Por Segundo.
FWT	Transformada Rápida Wavelet.
IEEE	Instituto de Ingenieros Eléctricos y Electrónicos.
MAC	Acumulador de Múltiples Operaciones.
MIPS	Millones de Instrucciones Por Segundo.
PIM	Estación sismológica presa infernillo Michoacán.
PIO	Estación sismológica pinotepa nacional Oaxaca.
STFT	Transformada de Fourier de Tiempo Corto.
TAC	Estación sismológica Tacubaya.



# Notación

Símbolo	Descripción
$a \in A$	$a$ es un elemento de $A$ .
$A \triangleq B$	$A$ es por definición igual a $B$ .
$A \subset B$	$A$ es un subconjunto propio de $B$ .
$A \subseteq B$	$A$ es un subconjunto de $B$ .
$f : A \rightarrow B$	$f$ mapea el dominio $A$ en $B$ .
km	kilómetros.
km/seg	kilómetros / segundo.
mm	milímetros.
mts	metros.
$V_p$	velocidad de onda p.
$V_s$	velocidad de onda s.
$\mathbb{R}$	Campo de los números reales.
$\mathbb{R}^n$	Espacio vectorial de los reales de dimensión $n$ .
$\sum$	Sumatoria.
$t_s$	tiempo de onda s.
D	distancia.
$t_p$	tiempo de onda p.
$\hat{\mathbf{x}}$	$\mathbf{x}$ estimada.
$\ \mathbf{x}\ $	Norma de $\mathbf{x}$ en un espacio vectorial.

# Capítulo 1

## Introducción

En este capítulo se definen los objetivos a lograr en este trabajo, así como las metas a alcanzar y sus posibles repercusiones. Por otra parte, se darán a conocer algunos puntos importantes y necesarios para el análisis de señales sísmicas y sus resultados.

En este trabajo se aprovecha la versatilidad de un DSP el cual puede ser implementado en cualquier área del saber para la obtención y análisis de datos proporcionados por los diferentes tipos de señales que existen en nuestro entorno.

El hombre a llevado su saber a grandes desarrollos tecnológicos y científicos; pero en esta ocasión llevaremos este caudal de conocimientos al rubro de la seguridad de nuestra sociedad; enfocado a sismos y así lograr un análisis más profundo acerca del tratamiento de estas señales naturales.

Con el auxilio de circuitos integrados que abren nuevas áreas de desarrollo basadas en procesadores digitales de señales ( DSP ), aplicados a: sensores inteligentes, visión de robots y automatización, etc., es posible ayudar al tratamiento de estas señales para que brinde las bases para continuar los avances en el estudio de estas.

Para ello, es necesario aplicar diversos procedimientos tales como: filtrado, comprensión, análisis en frecuencia, entre otras; los procesadores digitales de señales, son microprocesadores rápidos con un especializado conjunto de instrucciones y una arquitectura apropiada para el procesamiento de señales, que han facilitado su implementación dentro de diversos dispositivos relacionados con las comunicaciones, control y procesamiento de imágenes.

Para el procesamiento digital de señales se optó por trabajar con las señales sísmicas como caso de estudio, con el afán de extraer información útil para su comprensión

siendo que es poco el análisis de este tipo de señales que se cree pueden ser del tipo de emisión de radiaciones infrarojas; sin embargo, esto aún no ha sido aceptado por la comunidad científica y se les puede incluir como una señal eléctrica.

Para ello, es necesario valernos de técnicas matemáticas tales como la transformada Z y transformada rápida de Fourier; así como el diseño y simulación de los algoritmos necesarios en el DSP. En base a esto se plantean los objetivos:

## 1.1. Objetivos de la tesis

### General

Realizar algoritmos numéricos en un DSP que permitan ayudar en el análisis de señales sísmicas.

### Específicos

1. Estudiar las diferentes modalidades para obtener información de una señal sísmica, bases matemáticas, diseño y estándares propios de dichas señales.
2. Aprender las diferentes técnicas de procesamiento digital de señales.
3. Puesta en operación del DSP.
4. Programación de los algoritmos numéricos. En este punto en particular, sólo se cargarán los algoritmos necesarios, ya que el DSP cuenta con rutinas precargadas necesarias para el procesamiento.

## 1.2. Planteamiento del problema

Teniendo un DSP y una señal sísmica se plantea: como hacer el procesamiento de está empleando técnicas modernas de procesamiento digital de señales; porque se pretende ofrecer una nueva visión para el análisis de señales producidas por sismos.

## 1.3. Solución propuesta

La solución propuesta es emplear la transformada wavelet para el tratamiento de la señal sísmica. Para lo cual, primeramente se hace un estudio de las diferentes

características de la señal sísmica, para posteriormente programar tanto en simulación numérica (en entorno MatLab) como en emulación en el DSP (software propio del DSP, CCS[28]). Es importante aclarar que el DSP de Texas Instruments contiene ya los programas con la transformada wavelet y solo se aplica el algoritmo a la señal sísmica.

## 1.4. Alcances y limitaciones

Para hacer posible la comprensión, tratamiento y análisis de las señales sísmicas es preciso saber que es lo que deseamos obtener de este tipo de señales. Es necesario que la información de esta señal sea de utilidad para determinado caso de estudio. Dado que este proyecto se enfoca a dar un panorama distinto al investigador o estudiante de este tipo de señales causadas por sismos se darán a conocer las características y el tipo de información que será recabada por el DSP.

Debido a que hay pocos avances en el estudio de las señales sísmicas la finalidad de este trabajo será la de brindar un nuevo panorama de estudio, mediante las herramientas usadas se brindarán resultados específicos de la señal; como pueden ser: fase, frecuencia de la señal y amplitud que son datos particulares de la señal.

Teniendo en cuenta ventajas y desventajas, algunas de las limitaciones de este trabajo son:

- Sólo se procesa la señal sísmica y los resultados obtenidos son presentados.
- No se realiza un estudio detallado de los resultados derivados del procesamiento.
- Dadas las dificultades de encontrar disponibles al público en general las señales sísmicas en México se recurrió a otras fuentes de información, obteniendo de esta forma la señal sísmica empleada.

Dicho lo anterior, la contrariedad radica en que no existe un dispositivo capaz de brindar a detalle características específicas de señales producidas por movimientos telúricos las cuáles podrían ser de gran utilidad para personas dedicadas al estudio de este tipo de señales.

## 1.5. Justificación

Actualmente el término seguridad no debe ser limitado a la protección de un bien o de un servicio, sino que también abarca el resguardo de nuestra sociedad. El análisis

de sismos y señales producidas por dichos movimientos telúricos serán la plataforma para alcanzar la meta, y la correcta implementación de la misma puede proveer al entorno de las personas que se dedican al estudio de estas señales un nuevo medio de análisis de este tipo de señales.

Esto gracias al rápido avance en la electrónica, particularmente en las técnicas de fabricación de circuitos integrados, que han tenido y sin duda alguna; continuarán teniendo un gran impacto en la industria y la sociedad.

Desgraciadamente, al día de hoy se tiene un avance pobre en el estudio y mejora de la información obtenida a partir de estas señales, para que con ello se pueda llegar si no a predecir, a reducir las consecuencias que tendría un sismo en nuestra infraestructura social.

A continuación se presenta un panorama general de un sismo.

## 1.6. Sismo

Un sismo es una serie de perturbaciones súbitas en el interior de la tierra que dan origen a vibraciones o movimientos del suelo; la causa principal y responsable de la mayoría de los sismos (grandes y pequeños) es la ruptura y fractura de las rocas en las capas exteriores de la tierra. Como resultado de un proceso gradual de acumulación de energía debido a los fenómenos geológicos que deforman la superficie de la tierra, dando lugar a las grandes cadenas montañosas.

En el interior de la tierra ocurre una fractura súbita cuando la energía acumulada excede la resistencia de las rocas. Al ocurrir la ruptura, se propagan (en el interior de la tierra) una serie de ondas sísmicas que al llegar a la superficie se siente como un temblor. Generalmente, los sismos ocurren en zonas de debilidad de la corteza terrestre que son llamadas fallas geológicas. Existen también sismos menos frecuentes causados por la actividad volcánica en el interior de la tierra, y temblores artificiales ocasionados por la detonación de explosivos. El sitio donde se inicia la ruptura se llama foco y su proyección en la superficie de la tierra, epicentro (Figura 1.1).

En el caso de la tierra existen fundamentalmente dos tipos de ondas sísmicas internas; es decir, vibraciones que se propagan en el interior de la tierra: ondas compresionales o longitudinales y ondas de corte o cizallamiento. Las ondas compresionales, llamadas P en la terminología sismológica, comprimen y dilatan el medio donde se propagan en una dirección de propagación del frente de ondas. Las ondas de sonido; por ejemplo, son ondas compresionales que se propagan en el aire. El segundo tipo de

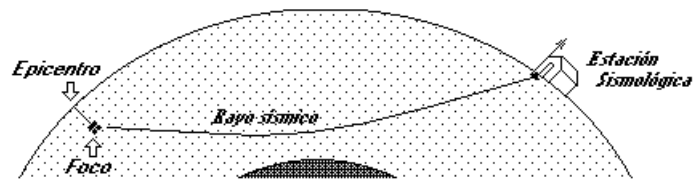


Figura 1.1: Localización de un epicentro.

ondas que se propagan en sólidos son las ondas de corte, llamadas ondas S. En este caso la deformación que sufre el sólido es en dirección perpendicular a la trayectoria del frente de ondas. La propagación de esta onda produce un esfuerzo cortante en el medio y de ahí el nombre de ondas de corte o cizallamiento como se muestra en la Figura 1.2.

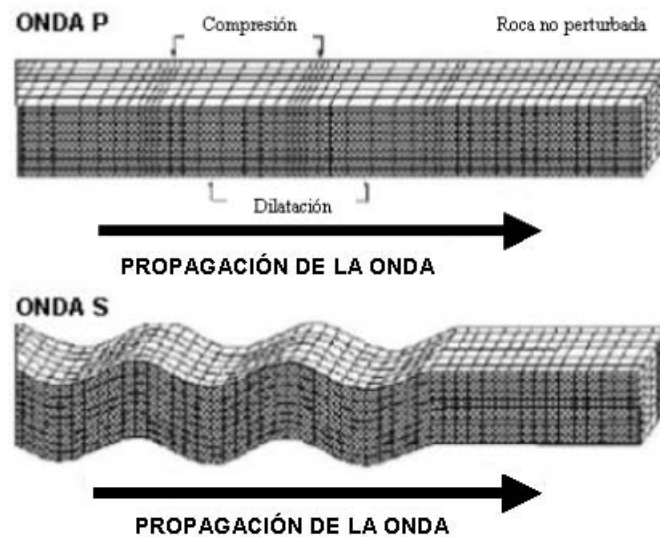


Figura 1.2: Ondas P y S.

La velocidad de propagación de las ondas en el interior de la tierra varía, dependiendo de la densidad y de las propiedades elásticas de las rocas. En rocas típicas de la corteza terrestre las ondas P se propagan a velocidades promedio de entre 4.5 y 6.5 km/seg, mientras que en el núcleo de la tierra, a profundidades de 6 000 km, la velocidad de las ondas P alcanzan los 15 km/seg. Las ondas S viajan a una velocidad más lenta que las ondas P; la relación aproximada entre ambas velocidades es:  $V_p = 3V_s$ . Como viajan más rápidamente, las ondas P son registradas antes que las ondas S; es por ello que en sismología a las ondas compresionales se les llama ondas primarias (P) y a las ondas de corte, que arriban más tarde, ondas secundarias (S). Además de las ondas internas P y S que han sido discutidas arriba existe otro tipo

de ondas que se propagan sobre la superficie de la tierra a velocidades menores que las de las ondas P y S. A esta clase de ondas se les llama ondas superficiales [7].

### 1.6.1. Registro de sismos

La vibración de la tierra debido a la ocurrencia de un temblor se observa experimentalmente con el auxilio de sismógrafos: instrumentos sumamente sensibles a los movimientos de la superficie de la tierra. Los primeros sismógrafos fueron construidos a finales del siglo pasado, empleando un sistema puramente mecánico. En la actualidad, estos se han modificado y perfeccionado, aunque el principio básico de operación es el mismo: una masa suspendida de un resorte sostenido por un soporte empotrado en el suelo. Cuando el suelo se mueve por el paso de las ondas sísmicas, también se mueve el soporte. Sin embargo, la inercia de la masa que es estable en su sitio, permite medir el desplazamiento relativo entre la masa y el suelo, como se ve en la Figura 1.3.

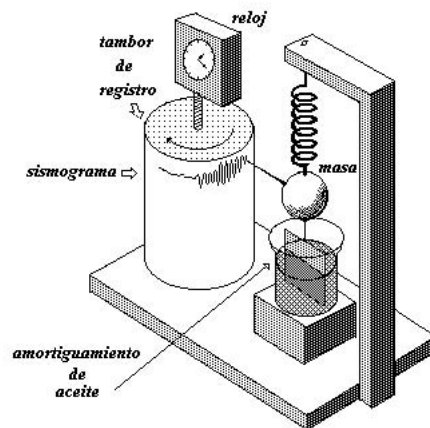


Figura 1.3: Principio básico de operación y registro de un sismógrafo que mide el desplazamiento vertical del terreno.

Normalmente, el desplazamiento relativo de la masa con respecto al suelo es tan pequeño que es necesario amplificarlo para poder medirlo. Inicialmente, la amplificación se hacía mecánicamente, por medio de palancas; en la actualidad se lleva a cabo electrónicamente y los instrumentos modernos amplifican el movimiento del terreno centenas de miles de veces. Una vez amplificado el movimiento del suelo, éste se registra en papel o en un cinta magnética. El registro obtenido en esta forma se llama sismograma. Además, los sismógrafos cuentan con un preciso control de tiempo que se inscribe directamente sobre los sismogramas para identificar exactamente el tiempo de llegada de las diferentes ondas sísmicas que arriban a una estación sismológica. El diseño y construcción de los sismógrafos se ha perfeccionado notablemente; en la

actualidad existen redes sísmicas de detección, transmisión y registro de temblores que envían las señales detectadas por medio de radio, microondas o satélite a grandes distancias. Este tipo de redes permiten estudiar la sismicidad de una amplia zona o de un región de difícil acceso.

## 1.6.2. Análisis de las señales sísmicas

### Determinación de epicentros.

Como las ondas sísmicas viajan a diferentes velocidades, las diferencias de tiempo de arribo entre las ondas P y S registradas en una estación sismológica están en función directa de su distancia al sitio donde ha ocurrido el temblor. Por ejemplo, para distancias regionales la diferencia en segundos entre el tiempo de arribo de las ondas P ( $t_p$ ) y las ondas S ( $t_s$ ) multiplicada por ocho, da la distancia aproximada al epicentro en kilómetros [ $distancia \sim 8 \times (t_s - t_p)$ ]. Sin embargo, es obvio que los datos de una sola estación no bastan para determinar el epicentro del sismo, puesto que la diferencia ( $t_s - t_p$ ) da la distancia, pero no la dirección; es necesario contar con un mínimo de tres estaciones sismológicas que registren el temblor para poder estimar la ubicación del epicentro, un ejemplo de esta configuración se muestra en la Figura 1.4. En esta figura, se representa el ejemplo de un sismo en las costas de Guerrero, localizado mediante registros de las estaciones: Tacubaya, D.F. ( TAC ); Presa Infiernillo, Mich. ( PIM ); y Pinotepa Nacional, Oax. ( PIO ).

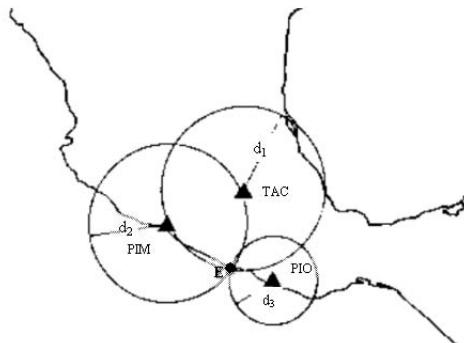


Figura 1.4: Conociendo la distancia ( $d$ ) de una estación sismológica al epicentro, ésta puede representarse como un círculo de radio  $d$  con centro en dicha estación. Si se cuenta con observaciones al menos en tres estaciones, el punto de intersección ( $E$ ) de los círculos corresponde al epicentro.

En términos generales, la localización es relativamente simple: basta ubicar las estaciones sismológicas en un mapa y trazar, para cada una de ellas, un círculo cuyo radio sea igual a la distancia de la estación al epicentro que se ha calculado con base



en las diferencias de llegada de las ondas P y S. El punto o región donde se intersectan los círculos trazados es el epicentro. En la práctica, el procedimiento para localizar epicentros en los observatorios sismológicos es más complicado, pues debe tomarse en consideración la estructura interna y la esfericidad de la tierra. Las localizaciones epicéntricas se hacen rutinariamente por medio de una computadora, aunque el principio general es el mismo ya descrito.

### **Intensidad y magnitud**

La escala más conocida para medir estas señales es la escala de Richter que maneja 10 grados. En la magnitud Richter, cada aumento de 1 grado equivale a 10 veces la intensidad y 32 veces la liberación de la energía en el grado anterior. Es decir, se trata de una escala logarítmica y no aritmética. Todo esto significa que si el suelo se mueve 1 mm en un sismo de escala 2, se moverá 10 mm en uno de escala 3; 100 mm en uno de magnitud 4; y 10 mts en uno de escala 6. En un terremoto de 8 grados, la tierra se moverá 10 mil veces más que en un sismo de 4 grados. En la actualidad se emplean una gran cantidad de dispositivos digitales para procesar la información de un sismo. Un dispositivo que actualmente se encuentra disponible en el mercado de la electrónica es el procesador digital de señales.

## **1.7. Procesador digital de señales**

Un procesador digital de señales DSP, es un microprocesador especializado y diseñado específicamente para procesar señales digitales en tiempo real. Este provee secuencias de instrucciones ultra-rápidas, tales como: ( MAC ) (multiply-accumulate operations), DEEP PIPELINING, la habilidad de actuar como un dispositivo de ( DMA ) (acceso directo a memoria), saturación aritmética, arquitectura Harvard, memoria de programas y memoria de datos separadas.[3]

La mayoría de los DSP son de punto fijo, porque en el mundo real del procesamiento digital, mucha precisión no es un requisito, sin embargo; DSP de punto flotante son comunes en los campos científicos y de otras aplicaciones donde la precisión es un requisito. Estos dispositivos serán de gran ayuda en relación a señales sísmicas; las cuales arrojan como resultado la amplitud de la onda sísmica que miden la magnitud del movimiento telúrico.

## **1.8. Organización del trabajo**

La distribución de la presente tesis es la siguiente:

Para el Capítulo 2, se dan a conocer los pormenores y datos importantes sobre las señales sísmicas en el que se pretende abarcar la información necesaria para el procesamiento de dicha información.

En el Capítulo 3 se ofrece un panorama acerca de la transformada wavelet y sus diferentes tipos. Los resultados derivados de las simulaciones realizadas en el procesador digital de señales; en el que se desarrolla el procesamiento digital de la señal sísmica aplicando herramientas matemáticas tales como la transformada rápida de Fourier y la transformada wavelet; además el resultado de los algoritmos desarrollados en el DSP son tratados en el Capítulo 4.

Por último, en el Capítulo 5 se muestran las conclusiones, a partir del análisis de señales sísmicas, tomando en cuenta sus características para obtener información de utilidad.

# Capítulo 2

## Ondas sísmicas

Como es sabido, un terremoto afecta áreas muy grandes comparadas con la extensión de la fuente sísmica. Esto indica que existe energía, liberada en la fuente, que es transmitida a través del terreno; esta energía se propaga en forma de ondas sísmicas.

El objetivo de este capítulo es abordar los conceptos de onda y de rayo, y se habla también acerca de los distintos tipos de ondas sísmicas y de su nomenclatura.

Es importante desglosar todas las características posibles sobre una onda sísmica, para así poder llegar al objetivo y ofrecer un buen panorama sobre la información recopilada.

### 2.1. Qué es un sismo y sus tipos de ondas

¿Qué es un sismo?. Un sismo es una serie de perturbaciones súbitas en el interior de la tierra que dan origen a vibraciones o movimientos del suelo; la causa principal y responsable de la mayoría de los sismos (grandes y pequeños) es la ruptura de las rocas en las capas más exteriores de la tierra.

Los sismos irradian señales en una amplia gama de frecuencias y amplitudes. En una escala logarítmica la banda de frecuencias sísmicas es mucho mayor que los tonos audibles. Únicamente un rango limitado de estas señales pueden ser desplegados en forma visible en un tiempo dado. De forma tal que cualquier sismógrafo que registre los datos en papel o película actúa como un filtro suprimiendo la mayor parte de la información útil. Actualmente la tecnología digital permite registrar todas las señales sísmicas de interés en un mismo medio y en un sólo flujo de datos, y de esta manera procesarla para su análisis. En este trabajo de tesis sólo se realizará el procesamiento de una señal sísmica en un DSP que previamente ha sido cargada en dicho dispositivo.

Estos equipos, conocidos como de banda ancha poseen las siguientes características: Sensibilidad suficiente para resolver señales al nivel de ruido mínimo del terreno dentro de todo el rango de frecuencias de interés. Rango de operación lo suficientemente sensible para registrar los grandes sismos a distancias regionales y telesísmicas. La resolución de señales pequeñas a altas frecuencias no es interferida por el ruido del terreno, ya sea natural o artificial.

Si se toma una barra de algún material elástico (metal, madera, piedra, etc.) por un extremo y es golpeada en el otro extremo, la energía del golpe se transmite a través de la barra y llega a nuestra mano. Esto sucede porque cada parte de la barra se deforma y luego vuelve a su forma original; al deformarse jala o empuja a las partes vecinas, las cuales, a su vez, mueven a sus propias partes vecinas, etc., lo que hace que la deformación viaje a lo largo de la barra. Como puede notarse la deformación es la que viaja y no las partículas o pedazos de la barra, los cuales sólo se desplazan un poco de su posición original y luego vuelven a ella. Una deformación que viaja a través de un medio como lo es la Tierra, se llama *onda sísmica*. Al conjunto de todos los puntos en el espacio que son alcanzados simultáneamente por una onda se le llama *frente de onda*. Un ejemplo familiar es el de las ondas formadas en la superficie de un lago al dejar caer en ella algún objeto; los frentes de onda son los círculos concéntricos que viajan alejándose de la fuente, es decir; del lugar donde se originó el disturbio.

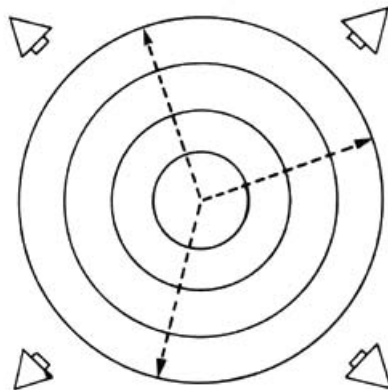


Figura 2.1: Frentes de onda circulares propagándose hacia afuera. Las líneas punteadas (rayos), perpendiculares a los frentes de onda, indican la trayectoria de propagación.

Si se trazan líneas (imaginarias) perpendiculares a los frentes de onda (indicadas por líneas punteadas en la figura), se ve que estas indican la dirección en la que viajan las ondas. Estas líneas son llamadas rayos, y son muy útiles para describir las trayectorias de la energía sísmica. La onda sísmica deforma el terreno a través del cuál

pasa, lo cuál indica que puede hacer trabajo, y, por lo tanto, corresponde a energía que se desplaza. En el caso de ondas generadas por explosiones, la energía es producto de las reacciones químicas o nucleares que causaron la explosión; en el caso de ondas generadas por sismos, es la que estaba almacenada como energía en las rocas[8].

### Onda periódica

Las ondas más sencillas son las senoidales (aquéllas cuyos valores varían en el tiempo y/o en el espacio como la función seno o función coseno), como las que se muestran en las tres primeras trazas de la Figura 2.2. Cada una se caracteriza por su frecuencia  $f$  (el número de veces que el movimiento se repite en cierto tiempo), expresada en Hertz (ciclos/segundo, abreviado Hz), o por su periodo  $T = 1/f$  (el tiempo que tarda en repetirse), expresado en segundos, su amplitud  $A$  (el máximo valor que puede tomar a partir del punto de equilibrio), expresada en unidades de longitud (usualmente micras o centímetros) y su fase (qué valor tiene la onda, es decir, en qué punto de su ciclo está, para un tiempo o lugar de referencia). Si una onda senoidal viaja con una velocidad  $V$ , al cabo de un periodo habrá recorrido una distancia  $\lambda = V \cdot T$ , denominada longitud de onda.

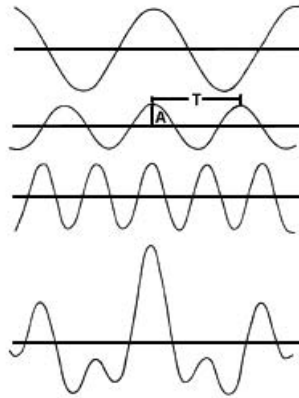


Figura 2.2: Ondas senoidales sumadas para obtener un pulso.

Para su representación en series de Fourier es necesario sumar las tres ondas senoidales, y así obtener la traza situada bajo ellas, la cual es menos regular que éstas y presenta un máximo donde los valores de las trazas componentes se suman (interferencia constructiva) y valores menores donde se anulan (interferencia destructiva).

De esta manera es posible construir una onda de cualquier forma, mediante la suma (en teoría infinita) de ondas senoidales con diferentes amplitudes y frecuencias (teorema de Fourier) [25]. La amplitud de cada onda senoidal componente indica qué

tanto de ella contiene la onda sísmica en cuestión. Se llama espectro de la señal sísmica al conjunto de sus componentes senoidales. Se dice que una onda sísmica es de alta o baja frecuencia (o de periodo corto o largo) según predominen en su espectro unas u otras componentes.

## 2.2. Ondas de cuerpo

La teoría de la elasticidad establece que son posibles dos tipos de ondas elásticas que viajan a través de la Tierra, y que son conocidas como *ondas de cuerpo* u *ondas internas*, las cuales pueden ser compresionales o de cizalla.

### 2.2.1. Ondas P

Las ondas compresionales son las que se transmiten cuando las partículas del medio se desplazan en la dirección de propagación, produciendo compresiones y dilataciones en el medio. Esto es fácil de visualizar si por ejemplo, en un resorte como el mostrado en la Figura 2.3, si se comprime un extremo del resorte (a) y luego se suelta, el material comprimido se extiende en la dirección indicada por la flecha pequeña, comprimiendo al material que está junto a él (b). Esa compresión y la dilatación (extensión) correspondiente viajan en la dirección indicada por las flechas gruesas, que es la misma (aunque puede variar el sentido) del desplazamiento de las partículas.

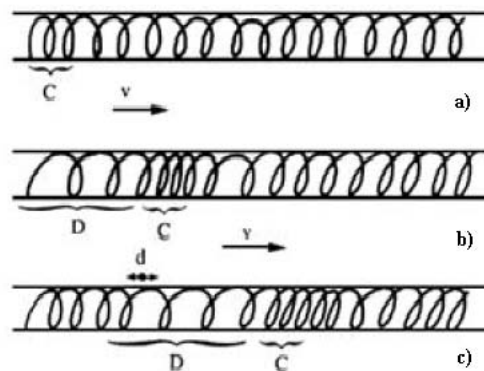


Figura 2.3: Onda compresional propagándose a lo largo de un resorte con velocidad  $v$ .  $C$  indica compresión y  $D$  indica dilatación. El desplazamiento de las partículas del resorte se produce en las direcciones indicadas por  $d$ .

Las ondas compresionales son las más veloces de todas las ondas sísmicas (más de 5 km/s en las rocas graníticas cercanas a la superficie, y alcanza más de 11 km/s en el interior de la Tierra) y, por lo tanto, es la primera en llegar a cualquier punto, en

ser sentida y en ser registrada en los sismogramas, por lo que se llamó onda Primera o Primaria y de allí el nombre de P (en inglés se asocia también con *push* que significa empujón o empujar).

### 2.2.2. Ondas S

Las ondas de corte o de cizalla, llamadas ondas S, son aquéllas en las cuales las partículas del medio se desplazan perpendicularmente a la dirección de propagación, por lo que están asociadas con deformaciones del terreno de tipo de cizalla. Se pueden visualizar si se piensa en las ondas que viajan por una cuerda tensa (Figura 2.4) y se mueve uno de sus extremos perpendicularmente a ella (a). Cada partícula de la cuerda se mueve, hacia arriba o hacia abajo en la dirección indicada por las flechas pequeñas, jalando a sus vecinas; de manera que la onda viaja en la dirección de la cuerda (indicada por la flecha grande) perpendicularmente a la dirección del desplazamiento de cada pedazo de cuerda (b-c).

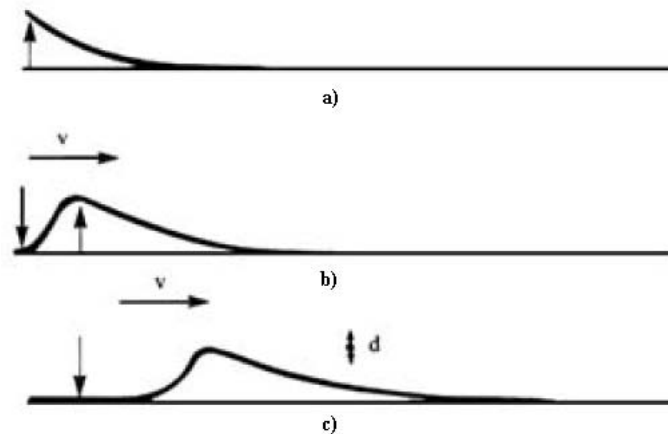


Figura 2.4: Onda de cizalla propagándose con velocidad  $v$  a lo largo de una cuerda. El desplazamiento de las partículas de la cuerda se da en las direcciones indicadas por  $d$ .

La onda S es más lenta que la onda P. En una amplia gama de rocas su velocidad,  $V_s$ , es aproximadamente igual a la velocidad de la onda P,  $V_p$ , dividida entre  $\sqrt{3}$  (esto es conocido como condición de Poisson)[29]. Como la onda S es la segunda en llegar se le llamó Secundaria, y de allí su nombre (en inglés se asocia con *shake*, que significa sacudir). Como los líquidos no pueden soportar esfuerzos cortantes, las ondas S no se propagan a través de ellos. El desplazamiento de las partículas en el terreno durante el paso de la onda puede ser en cualquier dirección perpendicular a la de propagación; pero, a veces, pueden desplazarse en una sola dirección, en cuyo caso se

dice que las ondas están polarizadas. La componente vertical de la onda S se denota a menudo por  $SV$ , mientras que la componente horizontal se denota por  $SH$  (Figura 2.5).

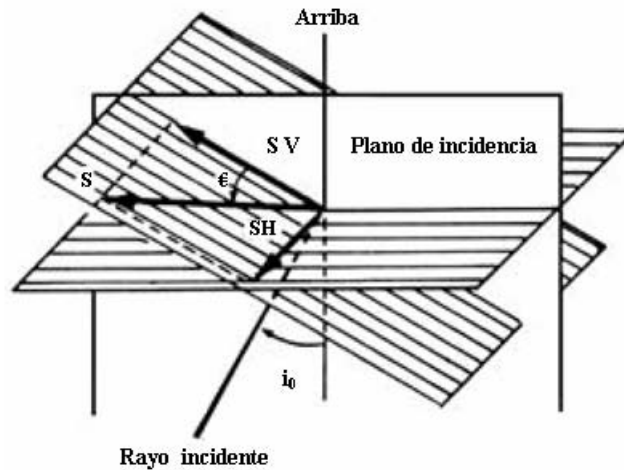


Figura 2.5: La onda S y sus componentes  $SV$  y  $SH$ .

Usualmente la onda S tiene mayor amplitud que la onda P, y se siente más fuerte que ésta.[1]

### 2.2.3. Ondas convertidas

Cuando una onda de cuerpo que viaja a través de un medio incide sobre una interfase (una superficie) que lo separa de otro medio con distintas propiedades elásticas, como se indica en la Figura 2.6, en general, parte de la energía es transmitida al segundo medio y parte es reflejada.

Si se traza una línea perpendicular a la interfase (la normal), y se miden los ángulos  $\alpha_1$  y  $\alpha_2$  que forman los distintos rayos con ella, se observa que están relacionados según la siguiente fórmula, conocida como ley de Snell:

$$\frac{V_1}{V_2} = \frac{\text{sen } \alpha_1}{\text{sen } \alpha_2} \quad (2.1)$$

donde  $V_1$  es la velocidad (P o S) del rayo en el primer medio y  $V_2$  su velocidad en el segundo medio. La aplicación de la ley de Snell permite saber cómo se comportan los rayos sísmicos cuando encuentran alguna de las discontinuidades que presenta la tierra



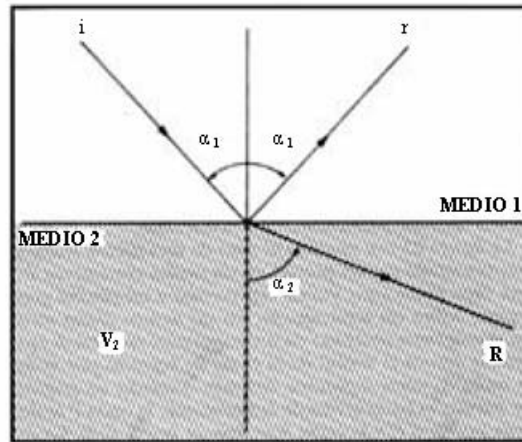


Figura 2.6:  $i$  rayo incidente,  $r$  rayo reflejado,  $R$  rayo refractado.

y que serán vistas someramente más adelante; se verán los nombres que se aplican a las ondas de cuerpo según la trayectoria que hayan recorrido. La ley de Snell enuncia que si un rayo pasa de un medio de menor velocidad a otro de mayor velocidad se aleja de la normal, mientras que, si pasa de un medio de mayor a otro de menor velocidad se acercará a ella. En particular, cuando  $\alpha_2$ , el ángulo de refracción es de  $90^\circ$ , y el rayo, llamado críticamente refractado viaja por el medio inferior, paralelamente a la interfase. La Figura 2.7 muestra el caso de una fuente sísmica (representada por un punto) en un medio consistente de una capa plana (que puede representar a un estrato geológico) sobre un semiespacio. En la Figura 2.7(a) se puede observar cómo sería un sismograma obtenido en un punto muy cercano al epicentro: vemos el arribo de la onda  $P_g$ , seguido por el de la onda  $S_g$  unos segundos después (cuando su ángulo de partida es hacia arriba de la horizontal, la onda P se denota por  $P_g$ , y la onda S se denota por  $S_g$ ). La diferencia de tiempo entre estos arribos es llamado a veces prefase, se indica generalmente por  $t_{sp}$  y está relacionado con la distancia D a la fuente como:

$$D = \frac{t_{sp} V_p V_s}{(V_p - V_s)} \quad (2.2)$$

Por lo tanto, para una gran cantidad de lugares en la Tierra (donde se cumple la condición de Poisson y  $V_p$  es aproximadamente igual a 6 km/s), si se cuenta el número de segundos entre las llegadas de P y S, y se multiplican por un valor constante igual a 8.2, se obtiene la distancia a la fuente en kilómetros.

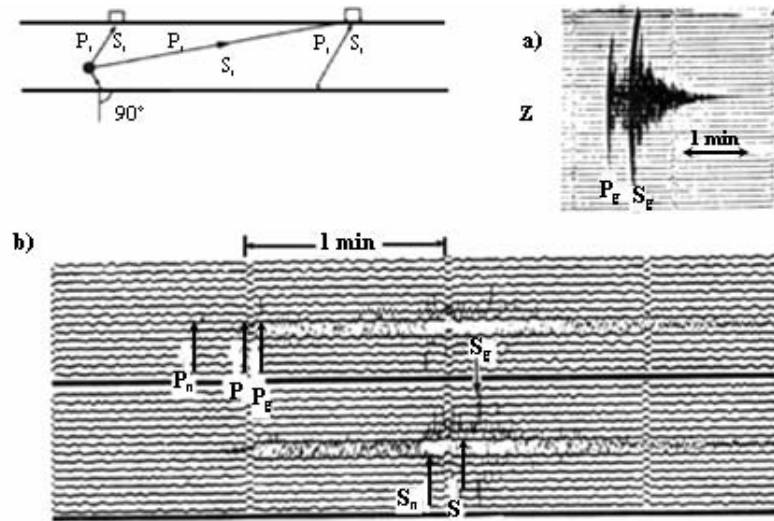


Figura 2.7: Fuente sísmica.

Además de  $P_g$  y  $S_g$ , llegarán rayos reflejados en la interfase. Éstos son rayos P reflejados como P o como S y rayos S reflejados como S o como P. Siempre llegarán después de  $P_g$ , pues tienen que recorrer un camino mayor, y parte de ese camino puede ser recorrido, en calidad de S.

En la Figura 2.7(b) se observa el sismograma correspondiente a un punto más lejano del epicentro, donde se ve que la onda P críticamente refractada en alguna interfase, llamada a veces  $P_r$ , llega antes que  $P_g$ . Esto sucede porque el camino de  $P_r$  es más largo pero más rápido. A partir de este punto, el primer arribo, llamado a veces FA, será el correspondiente a  $P_r$ . Si existe otro medio aún más rápido bajo la capa en que viaja ésta, puede dar lugar a otra que, para distancias más lejanas llegue aún más temprano. Como las ondas críticamente refractadas llegan a menudo a la cabeza del sismograma, son llamadas a veces ondas de cabeza.

Las ondas de cabeza asociadas con la discontinuidad de Conrad se denotan por  $P^*$  y  $S^*$ ; la velocidad de P es del orden de 6.5 a 6.8 km/s. Las ondas de cabeza refractadas por la discontinuidad de Mohorovicic se denotan por  $P_n$  y  $S_n$ ; la velocidad va de 7.8 a 8.3 km/s.

### 2.2.4. Coda

Después de la llegada de las ondas P y S, la señal vista en el sismograma decae poco a poco como lo indican las líneas punteadas en la Figura 2.7. Esta parte posterior de la señal se llama *coda*, y se debe a energía sísmica "dispersa" que llega hasta el sensor después de haber sido reflejada por las heterogeneidades propias del terreno. La forma como decae la coda sirve para estimar qué tan grande es la atenuación del terreno. También, es útil para la determinación de la magnitud de los sismos, sobre todo de los locales.

## 2.3. Ondas superficiales

Además de las ondas que viajan a través del terreno, existen otras que lo hacen por la superficie, esto es, su amplitud es máxima en ésta y nula en las grandes profundidades. Estas ondas pueden explicarse como causadas por la interferencia de las ondas de cuerpo (interacción de muchas de estas ondas que viajan en diferentes direcciones), y son más lentas que éstas. En el caso de los telesismos (los que ocurren a más de 1000 km de distancia del observador), como el mostrado en la Figura 2.8, las ondas superficiales llegan mucho después que las de cuerpo, y se aprecia que presentan dispersión; esto es, las ondas de diferentes frecuencias viajan con diferentes velocidades.

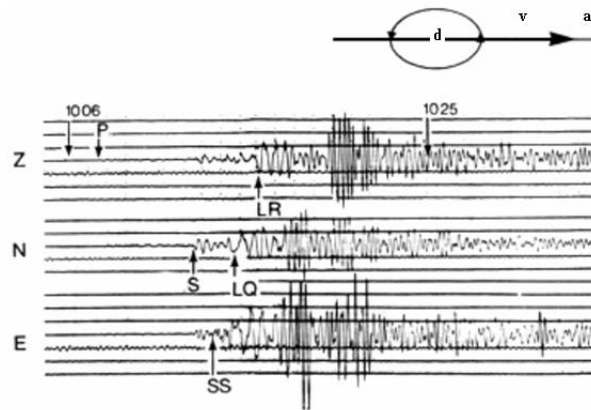


Figura 2.8: Telesismo registrado en sismómetros vertical (Z) y horizontal en dirección Norte-Sur (N) y Este-Oeste (E). En (a)  $d$  indica cómo es el desplazamiento de una partícula de la superficie de la Tierra al paso de una onda Rayleigh con velocidad  $v$ .

### 2.3.1. Ondas de Rayleigh

Éstas, denotadas usualmente por R, o LR cuando son de periodo muy largo se deben a la interacción entre las ondas P y las SV, y el movimiento de cada partícula de la superficie del terreno al paso de la onda se da en forma de elipse, según se muestra en la Figura 2.9.

Las ondas de Rayleigh son las ondas más lentas con velocidades de grupo (la velocidad con que viaja la energía) que van de 1 a 4 km/s, según se muestra en la Figura 2.9, que muestra varias curvas que corresponden a diversos modos de propagación de la onda de Rayleigh; donde cada modo propio, modo fundamental o eigenmodo es una forma en la cual puede vibrar el terreno de manera que se logre la interferencia constructiva que da lugar a las ondas superficiales. El modo cuya amplitud no cambia de signo con la profundidad (Figura 2.9) es llamado modo fundamental, el que cambia una vez de signo (Figura 2.10), primer modo superior, el que cambia de signo dos veces (Figura 2.11) segundo modo superior.

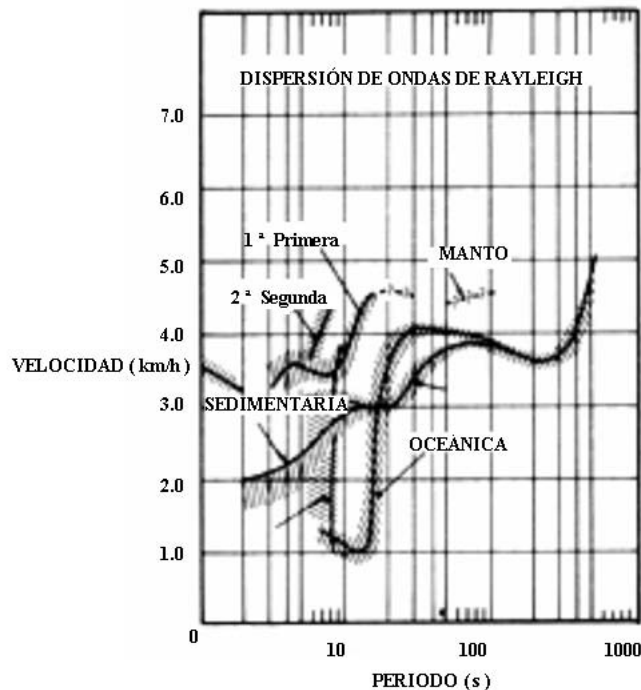


Figura 2.9: Velocidades de grupo para ondas Rayleigh.

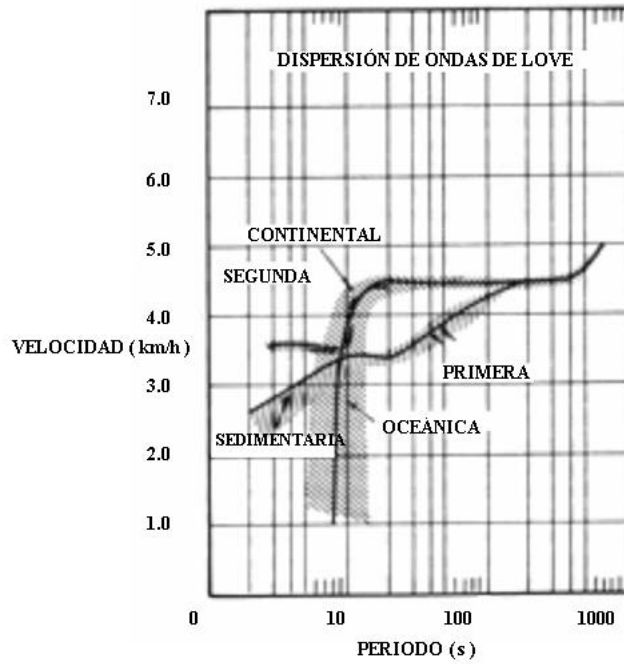


Figura 2.10: Velocidades de grupo para ondas Love.

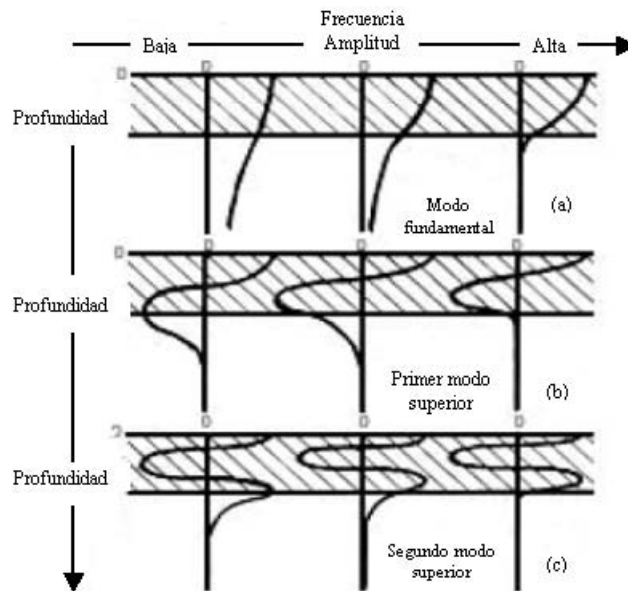


Figura 2.11: Amplitudes de los modos propios de vibración para el modo fundamental y los dos primeros superiores de una onda de Rayleigh.

Se ve que los modos de alta frecuencia tienen grandes amplitudes solamente cerca de la superficie del terreno, por lo que las propiedades del material profundo casi no influyen en ellos. En cambio, los modos de baja frecuencia tienen amplitudes considerables en profundidades mayores, por lo que su velocidad depende de las profundidades del medio cerca de la superficie y lejos de ella. La velocidad del terreno aumenta, usualmente, con la profundidad, lo que explica por qué, como se muestra en la Figura 2.11, las componentes de más baja frecuencia son usualmente las más rápidas. Sin embargo, la velocidad de grupo no disminuye siempre al aumentar la frecuencia, pues la transmisión de energía requiere de interferencia constructiva de los modos.

Otro efecto de la dependencia de los modos en la profundidad es que si la fuente sísmica ocurre a cierta profundidad, excitará más a aquellos modos cuyas amplitudes sean grandes y menos a aquellos cuyas amplitudes sean pequeñas (o, posiblemente, nulas) a dicha profundidad. Esto permite determinar la profundidad del foco de un terremoto, y, si es profundo, distinguirlo de una explosión nuclear que siempre es somera. En general, los sismos más someros generan mayores ondas superficiales que los profundos (para iguales momentos y áreas de ruptura).

### 2.3.2. Ondas de Love

En inglés *Love waves*, son las denotadas usualmente por L o G, o LQ si son de periodo muy largo. Se comportan de manera muy parecida a la descrita para las ondas de Rayleigh, pero se deben a interferencia constructiva de ondas SH solamente, por lo que no pueden existir en un semiespacio, sino que requieren al menos una capa sobre un semiespacio, donde pueda quedar atrapada parte de la energía sísmica. Son polarizadas horizontalmente (como las SH) y, por lo tanto, no se registran en los sensores verticales, como se muestra en la Figura 2.11.

Aunque más lentas que las ondas de cuerpo, las ondas Love tienen velocidades de 1 a 4.5 km/s son más veloces que las de Rayleigh, como se muestra en la Figura 2.10. En la Figura 2.10 se observan las curvas de dispersión de grupo para varios modos propios de las ondas Love. Se puede ver que, igual que con las ondas de Rayleigh, cada modo tiene una velocidad tope, y también existe una frecuencia tope por debajo de la cual no puede vibrar cada uno de los modos superiores.

## 2.4. Ondas guiadas

Cuando una capa o nivel de roca se encuentra rodeada de otras rocas con velocidades sísmicas superiores, algunas de las ondas que se encuentren dentro de ella no

podrán escapar a los medios circundantes y serán transmitidas a lo largo de la capa con muy poca pérdida de energía. Este tipo de estructura es conocido como guía de ondas. Existen varios tipos de guías de ondas; se puede considerar que la superficie de la Tierra es una guía de onda para las ondas de Rayleigh y de Love. En el océano existe una capa de agua de baja velocidad, conocida como canal *SOFAR*, que transmite ondas hasta grandes distancias; si un sismo genera ondas que se transmitan por este canal (a la velocidad del sonido en el agua), al alcanzar la tierra firme serán registradas después de la llegada de las ondas P y S, por lo que son conocidas como ondas T (de "Terceras").

Otros tipos de ondas guiadas son las ondas  $L_g$ , que son ondas de periodo corto (1 a 6 seg.), predominantemente transversales, guiadas en la corteza terrestre y observadas solamente en trayectorias puramente continentales; y las ondas de placa, que son ondas que viajan a lo largo de las placas subducidas.

## 2.5. Modos propios de la tierra

Cuando se golpea un objeto de tamaño finito, como por ejemplo, una campana, ésta comenzará a vibrar; pero no puede vibrar de cualquier manera, sólo puede vibrar en formas que sean combinaciones de ciertas formas de vibrar llamadas modos propios. Una campana o un diapasón tiene un modo que domina la vibración, y es lo que les da sus tonos característicos; sin embargo, no vibran de un sólo modo, su vibración total es la suma del total de sus modos, excitados cada uno en mayor o menor proporción.

La Tierra se comporta de manera similar, puede vibrar sólo de modos determinados, y toda vibración será una combinación de éstos. La Figura 2.12 muestra algunos de los modos de la Tierra. Cuando ocurre un sismo, excita algunos de estos modos, y se puede expresar cualquier onda sísmica como una combinación de ellos.

Sin embargo, para periodos muy largos podemos distinguir los modos individuales, y su estudio ha sido de gran valor para inferir varios datos acerca de la estructura del interior de la Tierra[17].

## 2.6. Resumen

En este capítulo se han dado a conocer los tipos de ondas sísmicas y sus principales características; así como algunos datos relevantes sobre las mismas.

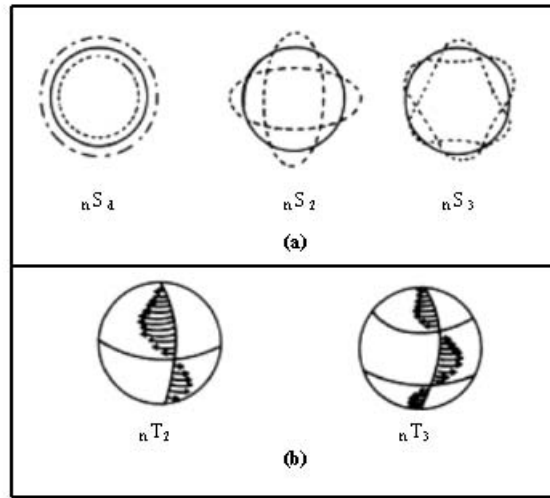


Figura 2.12: Modos propios de la Tierra. (a) Modos volumétricos: modo fundamental y los dos primeros modos superiores. (b) Primeros dos modos de cizalla.

Se abarcaron también los dispositivos usados para medir la intensidad de una onda sísmica; denominados sismógrafos que actualmente, la tecnología digital ofrece para monitorear todas las señales sísmicas de interés en un mismo medio y en un sólo flujo de datos.

Estos equipos, conocidos como de banda ancha poseen características como: Sensibilidad suficiente para resolver señales al nivel de ruido mínimo del terreno dentro de todo el rango de frecuencias de interés.

Es de vital importancia hacer notar que no todas las regiones geográficas poseen las mismas características, debido a esto las ondas sísmicas analizadas en este capítulo solo abarcan un ejemplo minúsculo comparado a toda la gama de ondas que pueden ser registradas.



# Capítulo 3

## Transformadas wavelet

En años recientes ha surgido un considerable interés con respecto a nuevas técnicas de transformación, que son especialmente dirigidos a problemas de compresión de imágenes, borde y detección de características, y análisis de textura. Estas técnicas se agrupan bajo el título de análisis multiresolución, análisis de tiempo-frecuencia, algoritmos piramidales, y transformadas wavelet.

En este capítulo, se revisan algunas de las limitaciones de la tradicional transformada de Fourier y transformadas similares y se definen tres tipos de transformadas wavelet que ofrecen una función mejorada para ciertas aplicaciones.

Nosotros mismos restringimos la transformación de valor real, medible, funciones integrales cuadradas de una dimensión, dado que estas abarcan las señales que son de interés para este trabajo de tesis. Se comienza por introducir los tres tipos básicos de transformadas wavelet. Entonces se ilustran algunas wavelets particulares y algunas aplicaciones de transformadas wavelet.

### 3.0.1. Ondas y ondoletas (wavelets)

Recordando que la transformada de Fourier, usa como sus funciones base ortonormal, **ondas** senosoidales, también recordemos que ellas se asemejan a las ondas del océano y ondas propagándose en otros medios. Para la transformada de Fourier, estas funciones se extienden infinitamente en ambas direcciones. Los vectores base de la transformada discreta de Fourier son también diferentes de cero sobre el dominio de los números enteros; esto es, ellos no tienen un soporte compacto.

En contraste, las componentes de una señal transitoria son diferentes de cero solo

durante un intervalo corto. Del mismo modo, muchas características importantes en imágenes (bordes, por ejemplo) son altamente localizados en posiciones especiales. Tales componentes no se asemejan a ninguna de las funciones base de Fourier, y ellas no son representadas en los coeficientes de la transformada (i.e., el espectro de frecuencia) como se discutirá subsecuentemente. Esto hace a Fourier y otras transformadas de onda, menos óptimas que en representaciones para análisis de señales comprimidas y analizar su contenido en los componentes transitorios o localizados.

En equidad, se nota que la transformada de Fourier puede representar cualquier función analítica (aún una señal transitoria limitada) como una suma de senosoidales.

Para combatir cada una de las deficiencias, matemáticos e ingenieros han explorado diversas aproximaciones usando transformadas, teniendo funciones base de duración limitada. Estas funciones base varían tanto en posición como en frecuencia. Ellas son ondas de duración limitada y son referidas como *wavelets*. Transformadas basadas en ellas son denominadas *transformadas wavelet*. Son también llamadas *ondoletas* en la literatura de la lengua española sobre el tema.

La Figura 3.1 ilustra la diferencia entre ondas y wavelets. Las dos curvas de la parte superior son ondas coseno que difieren en frecuencia pero no en duración. La parte inferior son dos wavelet que difieren ambas en frecuencia y posición a lo largo del eje.

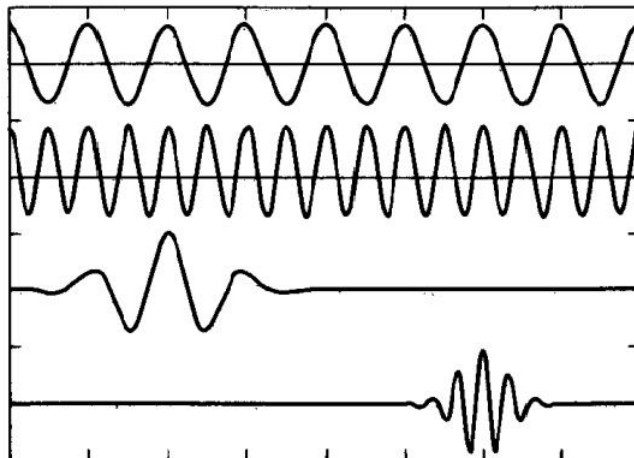


Figura 3.1: Ondas y ondoletas.

La transformada Haar es el ejemplo más claro de lo que ahora llamamos una

transformada wavelet. Esta difiere de otras transformadas en que todos sus vectores base son generados por translaciones y escalamientos de una función simple. La función Harr, la cual es un singular pulso rectangular, es la wavelet más antigua y sencilla.

### 3.0.2. Análisis tiempo-frecuencia

La literatura sobre procesamiento de señales incluye un trabajo considerable en el análisis de señales en términos de un espacio bidimensional tiempo-frecuencia. Éste método realmente precedió a las transformadas wavelet, pero ahora se encuentra dentro de el mismo marco moderno. Conforme a esto, cada componente transitorio de una señal mapea a una posición en el plano tiempo-frecuencia que corresponde a las componentes frecuenciales predominantes y tiempo de ocurrencia.

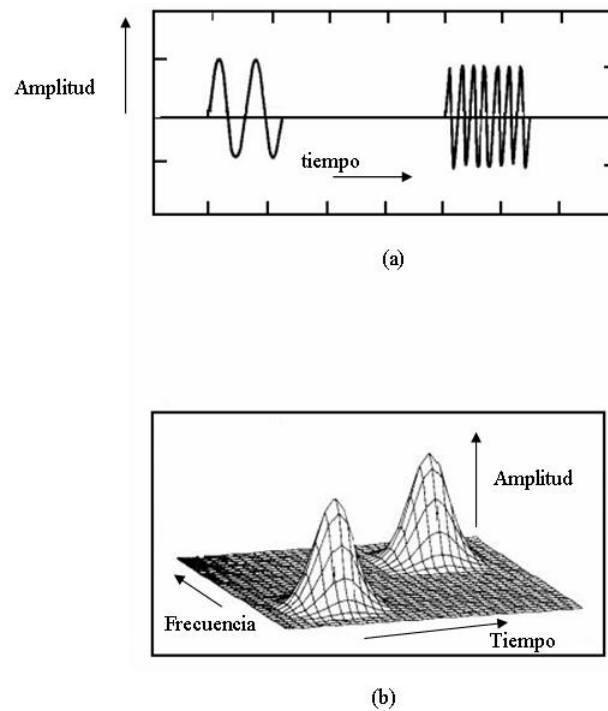


Figura 3.2: Espacio tiempo-frecuencia: (a) señal, (b) representación.

### 3.0.3. Transformadas

Cada uno de los coeficientes en una transformada es determinado por el producto interno entre la función de entrada y una de las funciones base. Este valor representa,

en algún sentido, el grado de similitud entre la función de entrada y la función base en particular. Si las funciones base son ortogonales (u ortonormales), entonces el producto interno tomado entre dos funciones es cero, indicando que estas son completamente disimilares. Entonces si la señal esta compuesta arriba de componentes que son similares a una, o a pocas, de las funciones base, entonces uno o pocos de los coeficientes serán pequeños.

De la misma manera, la transformada inversa puede ser vista como una reconstrucción de la señal original sumando funciones base que son ponderadas en amplitud por los coeficientes de la transformada. Entonces, sí la señal esta compuesta de componentes que son similares a una o a pocas de las funciones base, entonces esta sumatoria necesita tener sólo pocos términos de amplitud significativa. Muchos de los términos pueden entonces ser ignorados, y la señal puede ser representada de manera compacta sólo por pocos coeficientes de la transformada.

Además, si los componenetes de interés en la señal son similares a uno o pocos de las funciones base, entonces esos componentes se manifestarán en coeficientes grandes para aquellas (y solo aquellas) funciones base. Ellos asi serán fáciles de encontrar en la transformada. Finalmente, si un componente indeseable (ruido) es similar a una o pocas funciones base, entonces, también, será fácil de encontrar. Esto será también fácil de remover, simplemente reduciendo (o poniendo a cero) los correspondientes coeficientes de la transformada.

Concluimos de todo esto que existe un valor potencial usando transformadas con funciones base que son similares a los componentes esperados de las señales a ser transformadas. También se nota que los componentes transitorios no pueden ser similares a las funciones base de Fourier u otros tipos de transformadas de onda.

### 3.0.4. Tipos de transformadas

Existen tres diferentes, pero relacionadas técnicas de transformaciones de Fourier: la transformada integral de Fourier, la expansión en series de Fourier y la DFT. La transformada integral de Fourier asocia dos funciones continuas (una señal y su espectro). Ésta y su inversa están dadas en una dimensión por

$$F(s) = \int_{-\infty}^{\infty} f(x)e^{-j2\pi(xs)}dx \quad \text{y} \quad f(x) = \int_{-\infty}^{\infty} F(s)e^{j2\pi(xs)}ds \quad (3.1)$$

La expansión en series de Fourier representa una función periódica (o una función transitoria que puede ser considerada a ser un ciclo de una función periódica) como

una secuencia (finita o infinita) de coeficientes de Fourier. Ésta y su inversa son obtenidos haciendo  $s = n\Delta s$  una variable discreta tal que

$$F_n = F(n\Delta s) = \int_0^L f(x)e^{-j2\pi(n\Delta s)x} dx \quad y \quad f(x) = \Delta s \sum_{n=0}^{\infty} F_n e^{j2\pi(n\Delta s)x} \quad (3.2)$$

donde  $L$  es el periodo y  $\Delta s = 1/L$ .

La DFT representa una función simple por una muestra del espectro, y el número de muestras independientes (grados de libertad) es igual en ambos dominios. Ésta se obtiene haciendo  $x = i\Delta x$  una variable discreta. Si  $g(x)$  es de banda limitada y muestreada como requiere el teorema de muestreo, entonces  $g_i = g(i\Delta x)$ , y

$$G_k = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} g_i e^{-j2\pi k \frac{i}{N}} \quad y \quad g_i = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} G_k e^{j2\pi i \frac{k}{N}} \quad (3.3)$$

En las tres técnicas de transformación, senos y cosenos de diferentes frecuencias forma un conjunto de funciones base ortonormales. Además, cada coeficiente de la transformada es determinado por un producto interno de la función a ser transformada y una de las funciones base. Un producto interno discreto y funciones base discretas son usadas por la DFT, mientras un producto interno integral y funciones base continuas sirven para las otras dos transformadas. En cada caso, la transformada inversa consiste en la sumatoria de funciones base cuyas amplitudes son ponderadas por los coeficientes de la transformada. Esta sumatoria se convierte en una integral para la transformada continua de Fourier.

Las transformadas discretas discutidas anteriormente también usan funciones base ortonormales. Así, ellas se comportan de modo similar a la forma que se comporta la DFT.

### 3.0.5. Tipos de transformadas wavelet

Como con la transformada de Fourier, las mismas tres posibilidades existen para las transformadas wavelet: una transformada wavelet continua (TWC), una expansión en series de wavelet, y una transformada wavelet discreta (TWD). La situación es ligeramente mas compleja, ya que, desde las funciones base pueden o no ser ortonormales.

Un conjunto de funciones wavelet base pueden ser empleadas en una transformada igual si las funciones no son ortonormales. Esto significa, por ejemplo, que una expansión en serie de wavelets puede representar una función de banda limitada por muchos

coeficientes infinitos. Si esta secuencia de coeficientes es truncada para una duración finita, entonces podremos reconstruir solo una aproximación de la función original. Asimismo, una transformada wavelet discreta puede requerir más coeficientes de los que la función original tiene puntos de prueba en orden para reconstruirla exactamente, o igual para una aproximación aceptable.

### 3.0.6. Notación y definiciones

A continuación, se introducen algunas definiciones para aclarar el concepto de la transformada wavelet. De acuerdo con el escrito de la literatura sobre wavelets se usa  $j$  para representar como un índice de integración en este capítulo. También usaremos  $j$  para representar la unidad imaginaria  $\sqrt{-1}$ , teniendo cuidado de no usar ambos en la misma ecuación. La distinción del contexto debe ser clara.

La clase de funciones que se buscan para representar por una transformada wavelet son aquellas que son integrables cuadráticas sobre el eje real (i.e., el conjunto de todos los números reales, el eje  $x$ ). Esta clase es denotada como  $L^2(R)$ . Así, la notación  $f(x) \in L^2(R)$  significa

$$\int_{-\infty}^{\infty} |f(x)|^2 dx < \infty \quad (3.4)$$

En el análisis wavelet, se genera un conjunto de funciones base mediante escalamiento y traslación de una función prototipo,  $\psi(x)$ , el cual es llamado **wavelet básica**. Esto es, algunas funciones oscilatorias, usualmente centradas sobre el origen, que decae rápidamente a medida que  $|x| \rightarrow \infty$ . Así,  $\psi x \in L^2(R)$ .

## 3.1. Transformada wavelet continua

La transformada wavelet continua (también denominada transformada integral wavelet) fue presentada por Grossman y Morlet [19].

### 3.1.1. Definición

Si  $\psi(x)$  es una función real valuada cuyo espectro de Fourier,  $\psi(s)$ , satisface el criterio de admisibilidad [19, 10]

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\psi(s)|^2}{|s|} ds < \infty \quad (3.5)$$

donde  $\psi(x)$  se denomina **wavelet básica**. Nótese que, debido a que  $s$  está en el denominador, es necesario que

$$\psi(0) = 0 \Rightarrow \int_{-\infty}^{\infty} \psi(x) dx = 0 \quad (3.6)$$

Además, puesto que  $\psi(0) = 0$ , podemos ver que la amplitud del espectro de una wavelet admisible es similar a la función de transferencia de un filtro pasa banda. De hecho, ningún filtro pasabanda con respuesta al impulso con media cero significa (Ecuación 3.6) que decae a cero lo suficientemente rápido incrementando la frecuencia (Ecuación 3.5) puede servir como una wavelet básica para esta transformada.

Un conjunto de funciones wavelet base,  $\{\psi_{a,b}(x)\}$ , puede ser generada por translamiento y escalamiento de la wavelet base,  $\psi(x)$ , como

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}} \psi\left(\frac{x-b}{a}\right) \quad (3.7)$$

donde  $a > 0$  y  $b$  son números reales. La variable  $a$  refleja la escala (ancho) de una función base en particular, mientras que  $b$  especifica su traslado de posición a lo largo del eje  $x$ .

Normalmente la wavelet básica,  $\psi(x)$ , está centrada en el origen, entonces  $\psi_{a,b}(x)$  está centrado en  $x = b$ . La Figura 3.3 muestra un ejemplo de tal wavelet. Ésto, en particular esta dada por

$$\psi(x) = \frac{2}{\sqrt{3}\sqrt{\pi}} (1-x^2) e^{-\frac{x^2}{2}} \quad (3.8)$$

La transformada wavelet continua de  $f(x)$  con respecto a la wavelet  $\psi(x)$  es entonces

$$W_f(a,b) = \langle f, \psi_{a,b} \rangle = \int_{-\infty}^{\infty} f(x) \psi_{a,b}(x) dx \quad (3.9)$$

Los coeficientes de la transformada wavelet son dados una vez más como productos internos de la función a ser transformada con cada una de las funciones base. Grossman y Morlet [19] demostraron que la transformada wavelet inversa continua es

$$f(x) = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty W_f(a, b) \psi_{a,b}(x) db \frac{da}{a^2} \quad (3.10)$$

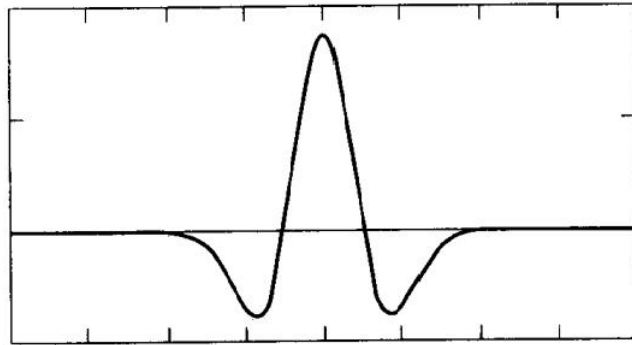


Figura 3.3: Una wavelet.

El factor escala en el lado derecho de la Ecuación 3.7, asegura que las normas de las funciones wavelet base son todas iguales, dado que

$$\left\| f\left(\frac{x-b}{a}\right) \right\| = \sqrt{\int_{-\infty}^\infty \left| f\left(\frac{x-b}{a}\right) \right|^2 dx} = \sqrt{a} \|f(x)\| \quad (3.11)$$

Puesto que la wavelet base tiene media cero (Ecuación 3.6), todos los escalamientos y translaciones de ésta (Ecuación 3.7) igualmente tendrán media cero, y la media de  $f(x)$  debe ser encontrada por separado.

### 3.1.2. La interpretación de bancos de filtros

El siguiente ejercicio ilustra una forma de ver la transformada wavelet continua. Primero se define la función wavelet base general con escala  $a$  como

$$\psi_a(x) = \frac{1}{\sqrt{a}} \psi\left(\frac{x}{a}\right) \quad (3.12)$$



Esta es la wavelet básica escalada por  $a$  y normalizada por  $a^{-\frac{1}{2}}$ . Ésta define un conjunto de funciones que llegan a ser amplias incrementando  $a$ . También definimos

$$\check{\psi}_a(x) = \psi_a^*(-x) = \frac{1}{\sqrt{a}}\psi^*\left(\frac{-x}{a}\right) \quad (3.13)$$

el cual es el complejo conjugado reflejado de la wavelet escalada. Si  $\psi(x)$  es real y par, como en muchas veces sucede, la reflexión y conjugación no tienen efecto.

Ahora se puede escribir la transformada wavelet continua (Ecuación 3.12) como

$$W_f(a, b) = \int_{-\infty}^{\infty} f(x)\check{\psi}_a(b-x)dx = f * \check{\psi}_a \quad (3.14)$$

Para  $a$  fija, entonces,  $W_f(a, b)$  es la convolución de  $f(x)$  con la wavelet conjugada reflejada con escala  $a$ .

La Figura 3.3 muestra la transformada integral wavelet integral como un banco de filtros lineal (convolución) actuando sobre  $f(x)$ . Cada valor de  $a$  define un filtro pasabanda diferente, y las salidas de todos los filtros, toman a la vez, la transformada wavelet comprendida. Entonces, la Ecuación 3.13 llega a ser

$$f(x) = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty [f * \check{\psi}_a](b)\psi_a(b-x)db \frac{da}{a^2} = \frac{1}{C_\psi} \int_0^\infty [f * \check{\psi}_a * \psi_a](x) \frac{da}{a^2} \quad (3.15)$$

lo cual implica que las salidas de los filtros, cada uno filtrado otravez por  $\psi_a(x)$  y propiamente escalado, combinados para reconstruir  $f(x)$ . Esto es una declaración de la identidad de Calderon [9, 24].

Recordando del teorema de similaridad que

$$\mathfrak{F}\{f(ax)\} = \frac{1}{|a|}F\left(\frac{s}{a}\right) \quad (3.16)$$

Esto significa que

$$\Psi_a(s) = \mathfrak{F}\{\psi_a(x)\} = \sqrt{a}\Psi(as) \quad (3.17)$$

y las frecuencias centrales de los filtros pasabanda decrecen cuando las funciones de transferencia llegan a ser muy limitadas con el incremento de  $a$ .

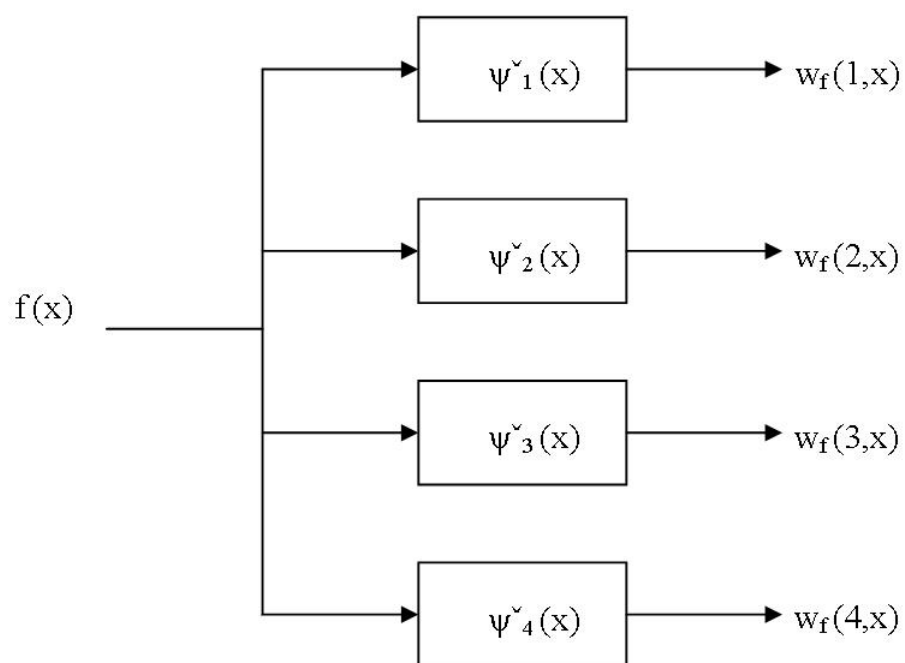


Figura 3.4: Analogía de banco de filtros para la transformada integral wavelet de una señal.

## 3.2. La expansión en series wavelet

### 3.2.1. Wavelets díadicas

El segundo tipo de transformada wavelet es algo más restrictiva que la primera. Nuevamente, una wavelet básica es escalada y trasladada para formar un conjunto de funciones base. Sin embargo, aquí, el escalamiento y la translación son especificados por enteros en vez de números reales.

En esta segunda definición, se restringen la formación de las funciones básicas por *escalamientos binarios* (disminuyendo por factores de dos) y *translaciones dyadic* de la wavelet básica,  $\psi(x)$ . Una translación *dyadic* es un desplazamiento de valor  $\frac{k}{2^j}$ , la cual es un múltiplo entero del factor de escala binario y así como también del ancho de la wavelet. Escalamientos binarios y translaciones *díadicas* son ilustradas en la Figura 3.5.

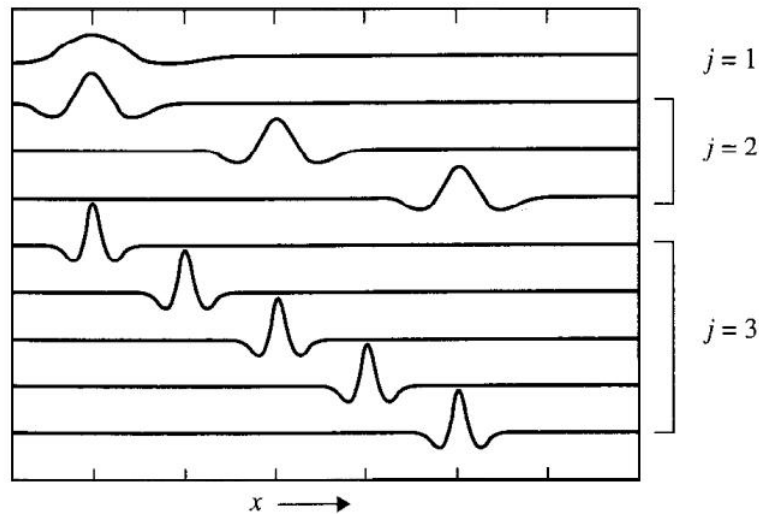


Figura 3.5: Escalamiento binario y translación díadica de una wavelet.

### 3.2.2. Definición

Una función  $\psi(x)$  es una *wavelet ortogonal* si el conjunto de funciones  $\{\psi_{j,k}(x)\}$  está definido por

$$\psi_{j,k}(x) = 2^{\frac{j}{2}} \psi(2^j x - k) \quad (3.18)$$

donde  $-\infty < j, k < \infty$  son enteros, formando una base ortonormal de  $L^2(\mathbb{R})$  [10]. El entero  $j$  determina la dilatación, mientras que  $k$  especifica la translación.

El conjunto de wavelets anteriores forman una base ortonormal si, primero

$$\langle \psi_{j,k}, \psi_{l,m} \rangle = \delta_{j,l} \delta_{k,m} \quad (3.19)$$

donde  $l$  y  $m$  son enteros,  $\delta_{j,k}$  es la función delta Kronecker, y  $\langle \cdot, \cdot \rangle$  indica el producto interno, y segundo, si cualquier función  $f(x) \in L^2(\mathbb{R})$  puede ser escrita como

$$f(x) = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} c_{j,k} \psi_{j,k}(x) \quad (3.20)$$

donde los coeficientes de la transformada son también dados por productos internos; esto es

$$c_{j,k} = \langle f(x), \psi_{j,k}(x) \rangle = 2^{\frac{j}{2}} \int_{-\infty}^{\infty} f(x) \psi(2^j x - k) dx \quad (3.21)$$

las Ecuaciones 3.27 y 3.28 especifican una *expansión en serie de wavelets* de  $f(x)$  relativa a la wavelet  $\psi(x)$  [10].

Nótese que aquí una función continua 3.27 es representada por una sucesión infinita doble, y en general, la transformada es otra vez sobrecompleta. Ya que las funciones base comúnmente se extienden hasta infinito en ambas direcciones, una reconstrucción completa debe incluir todos los términos.

No obstante, si  $\psi(x)$  es apropiadamente elegida, uno puede truncar las series sin aproximaciones erias de error. Si  $f(x)$  es de duración finita, y la wavelet básica está bien localizada, (i.e., ésta se aproxima rápidamente a cero alejándose del origen), entonces muchos de los coeficientes con  $|k|$  grande serán eliminados. Asimismo, coeficientes con  $|j|$  grande serán usualmente pequeños, ya que las funciones wavelet base llegan a ser extremadamente extensas o limitadas.

### 3.2.3. Wavelets díadicas compactas

Si además restringimos  $f(x)$  y la wavelet básica a funciones que son cero fuera del intervalo  $[0, 1]$ , entonces la familia de funciones base ortonormales pueden ser especificadas por un solo índice,  $n$ ; esto es

$$\psi_n(x) = 2^{\frac{j}{2}} \psi(2^j x - k) \quad (3.22)$$

donde  $j$  y  $k$  son realmente funciones de  $n$ , como sigue:

$$n = 2^j + k \quad \text{para} \quad j = 0, 1, \dots, k = 0, 1, \dots, 2^j - 1 \quad (3.23)$$

Para cualquier  $n$ ,  $j$  es el entero más grande, tal que  $2^j \leq n$ , y  $k = n - 2^j$ .

Ahora la transformada inversa es

$$f(x) = \sum_{n=0}^{\infty} c_n \psi_n(x) \quad (3.24)$$

donde se asume que  $\psi_0(x) = 1$ . Los coeficientes de la transformada están dados por el producto interno

$$c_n = \langle f(x), \psi_n(x) \rangle = 2^{\frac{j}{2}} \int_{-\infty}^{\infty} f(x) \psi(2^j x - k) dx \quad (3.25)$$

Aquí, una función continua es representada por una sola sucesión infinita, como en una representación de series de Fourier. La tremenda redundancia de la transformada integral wavelet es eliminada. En realidad, si uno o pocos de los  $\psi_n(x)$  son similares a  $f(x)$  (o sus mayores componentes), entonces uno puede truncar las series a pocos términos sin un error de aproximación apreciable.

Aquí se tienen las bases de la transformada wavelet discreta (TWD). Si  $f(i\Delta t)$  es una función discreta muestreada con  $N$  puntos, donde  $N$  es una potencia de dos, y si  $\psi(x)$  es una *wavelet diádica compacta*, entonces podemos calcular una transformada discreta usando versiones discretas de las Ecuaciones (3.27) y (3.28). Ambas ecuaciones llegan a ser sumatorias de  $N$  términos. La transformada Haar ofrece un ejemplo de esto.

### 3.2.4. Ejemplo: La transformada Haar

La transformada Haar es uno de los ejemplos más sencillos de lo que ahora llamamos, una transformada wavelet ortonormal diádica compacta. Esta difiere de otras transformadas en que sus funciones base son todas generadas por translaciones y dilataciones de una wavelet básica. La función Haar, la cual es un par de pulsos par, es la más simple y vieja wavelet ortonormal con soporte compacto.

La wavelet básica es progresivamente limitada (reducida en escala) por potencias de dos. Cada pequeña wavelet es entonces trasladada por incrementos igual a su ancho, tal que el conjunto completo de wavelets en alguna escala cubre completamente el intervalo. Como la wavelet base es escalada bajo potencias de dos, su amplitud es escalada sobre potencias de  $\sqrt{2}$ , para mantener la ortonormalidad. El resultado de

todo esto es un conjunto de funciones de bases ortonormales (Figura 3.6).

### 3.3. La transformada wavelet discreta

La DWT (o transformada wavelet discreta) es muy semejante a las transformadas unitarias. Ésta promete ser la más usada para compresión de imágenes, procesamiento y análisis. Dado un conjunto de funciones base ortonormales, uno puede calcular la transformada wavelet discreta justo como se hace en cualquier otra transformada unitaria, tal como la transformada Haar. Obtener una wavelet básica apropiada, requiere experiencia en la materia.

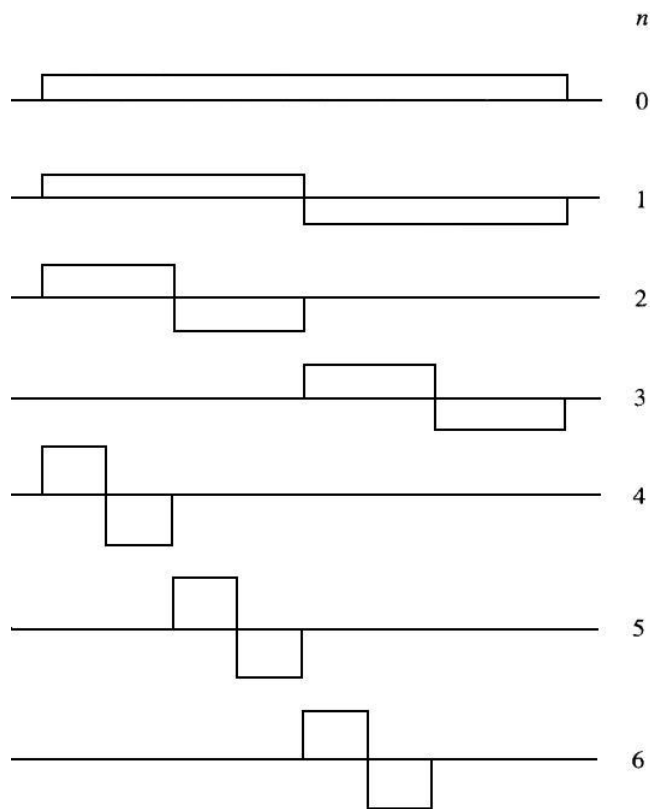


Figura 3.6: Funciones base la transformada Haar.

En ésta sección, revisaremos tres técnicas que han encabezado el desarrollo de la transformada wavelet discreta: (1) teoría de banco de filtros, (2) análisis multiresolución o análisis tiempo-escala, particularmente usando representaciones piramidales, y (3)

código sub-banda. Ésta discusión es seguida por una introducción a la transformada wavelet discreta.

### 3.3.1. Teoría de banco de filtros

Trabajos en el área de análisis de habla y procesamiento de señales acústicas han usado por mucho tiempo el concepto de banco de filtros pasabanda para descomposición de una señal en componentes a diferentes frecuencias. Es más, el método es un precursor del *análisis tiempo-frecuencia*, en el cual las componentes de las señales son mostrados en un *espacio de bidimensional* cuyas dimensiones son tiempo de ocurrencia y frecuencia de oscilación. Aquí, se revisarán las bases de los métodos como un paso principal hacia una discusión para la transformada discreta wavelet.

Supongase que se tiene una señal compuesta de *dos tonos desencadenados* (senosoides de corta duración) incrustado un ruido aleatorio, como se ilustra en la Figura 3.7a. Suponga además que se desea analizar esta señal para detectar el número, frecuencia y posición de los tonos desencadenados.

La transformada de Fourier reflejará por supuesto, contenido completo de la señal, pero no con frecuencia en una forma que es fácilmente interpretada. Información de posición, por ejemplo, es codificado en el espectro de fase de una manera complicada. Mientras el espectro de amplitud pueda mostrar distintos picos debido a cada componente transitoria de la señal, esta es seguro sólo para detección transitoria cuando sus componentes son suficientemente grandes en amplitud y duración para dominar el espectro. La Figura 3.7b, por ejemplo, manifiesta distintos picos en las frecuencias de dos tonos desencadenados. Sin embargo, la fase del espectro, da una nueva percepción en la ubicación de esos componentes en tiempo. Muchas veces, los componentes sin interés de la señal (ruido), complican el espectro hasta el punto en que el análisis en frecuencia es insuficiente para resolver los componentes de la señal.

### 3.3.2. Filtros pasabanda ideales

Supongase que se divide el eje de la frecuencia en un conjunto de intervalos disjuntos (adyacentes, no se traslapan) y se usa esta división para definir un conjunto de funciones de transferencia ideales pasabanda, como se muestra en la Figura 3.8b. La respuesta impulso correspondiente aparece en la Figura 3.8a. La Figura 3.9 muestra la implantación de un banco de filtros pasabanda. La señal de entrada alimenta a cada uno de los filtros pasabanda en paralelo. Las correspondientes salidas son  $g_i(x)$ . Las  $H_i(x)$  se construyen tal que su suma sea 1 para todas las frecuencias, y así, los

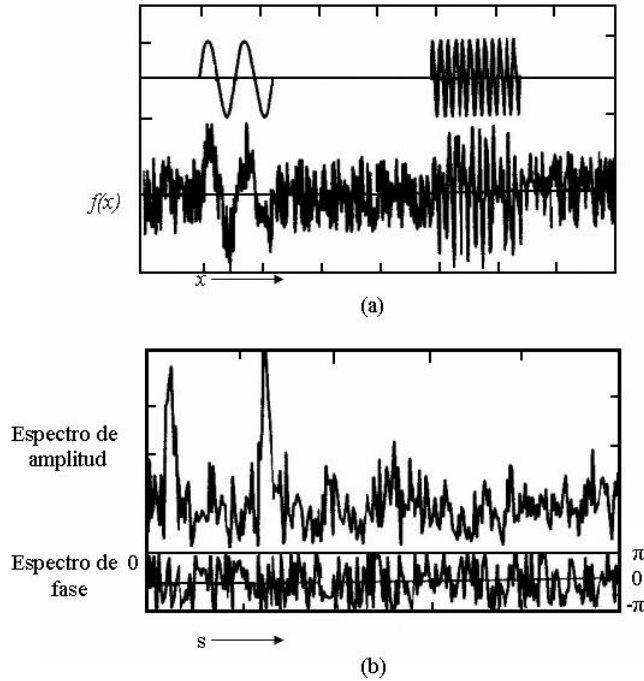


Figura 3.7: Señal compuesta que contiene dos tonos bursts y ruido aleatorio: (a) los tres componentes, (b) espectro de amplitud y fase.

$g_i(x)$  sumarán la forma  $f(x)$ . Esto es,

$$\sum_{i=1}^{\infty} H_i(s) = 1 \Rightarrow \sum_{i=1}^{\infty} g_i(x) = f(x) \quad (3.26)$$

La Figura 3.10 muestra la salida de 3 de los filtros pasabanda mostrados en la Figura 3.9. Nótese que las dos señales con tono bursts, (como en la Figura 3.7a) emerge desde filtros separados. Además, su localización a lo largo del eje del tiempo es evidente en sus salidas. Así, se tiene un método para descomponer la señal compuesta e identifica las componentes de interés.

Cada una de las salidas del filtro pasabanda esta formada por la convolución

$$g_i(x) = \int_{-\infty}^{\infty} f(t)h_i(t-x)dt \quad (3.27)$$

Puesto que  $H_i(s)$  es real y ,  $h_i(x)$  también lo será. Entonces la reflexión en la integral de convolución no tiene efecto, y las salidas del filtro pueden ser escritas como

$$g_i(x) = \int_{-\infty}^{\infty} f(t)h_i(t-x)dt = \langle f(t), h_i(t-x) \rangle \quad (3.28)$$



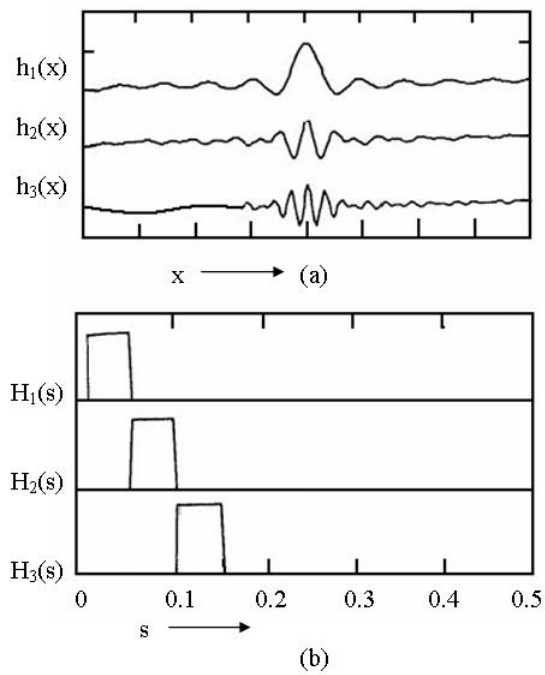


Figura 3.8: Generando una serie de filtros pasabanda por división del eje de frecuencia: (a) respuestas impulso, (b) funciones de transferencia.

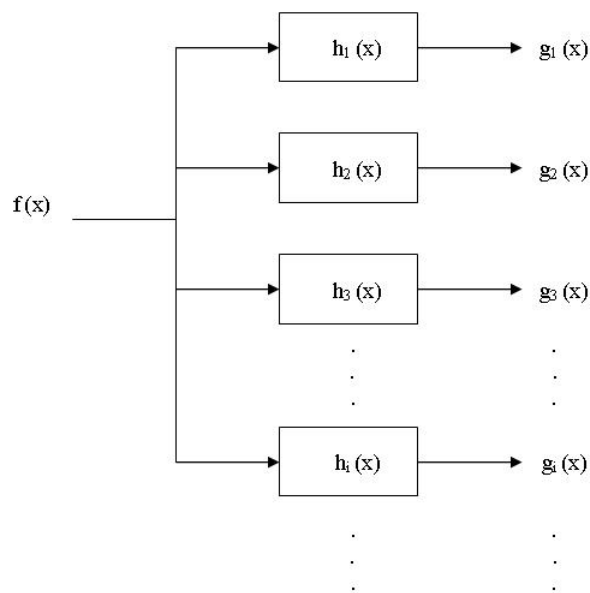


Figura 3.9: Implantación de un banco de filtros pasabanda.

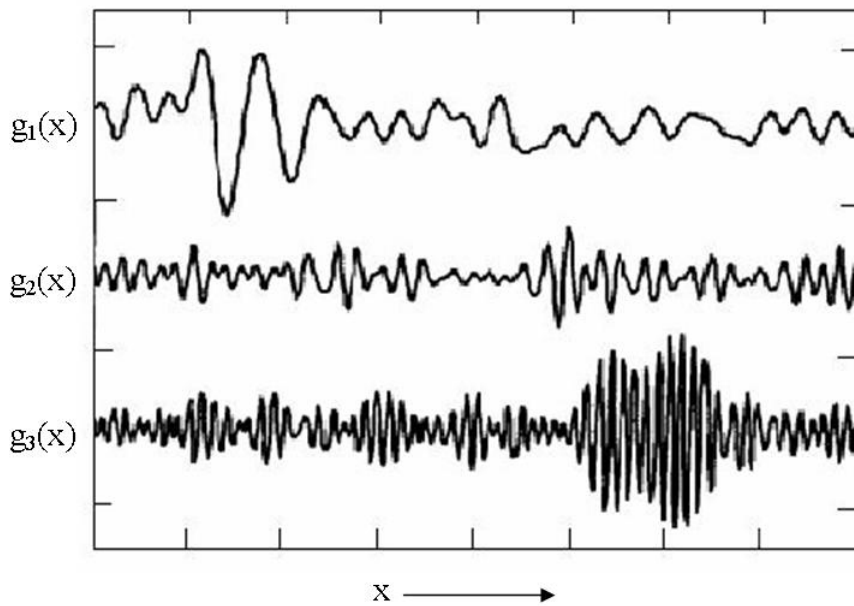


Figura 3.10: Salidas del filtro pasabanda.

Por consiguiente, cada punto sobre  $g_i(x)$  es el producto interno de  $f(t)$  con una versión de  $h_i(t)$  que ha sido desplazada  $x$ . También podemos ver  $g_i(x)$  como un conjunto (bidimensional) de coeficientes de la transformada wavelet, donde  $\{h_i(x)\}$  es el conjunto de wavelets. Además,  $g_i(x)$  es suficiente para reconstruir exactamente a  $f(x)$  en vista de la Ecuación 3.28.

### 3.3.3. Filtros pasabanda suaves

Las funciones  $h_i(x)$  en la Figura 3.8a carecen de una de las buenas características que las funciones wavelet base deberían tener: ellas no son bien localizadas. Esto es, no decaen rápidamente fuera de su región central. Esto significa que  $h_i(x - x_0)$  responderá a componentes fuertes que son localizadas en un punto distante a  $x_0$ . Estos bordes agudos de  $H_i(s)$  that give rise to the undesirable width of  $h_i(x)$ .

Diseñando las funciones  $H_i(s)$  para tener bordes suaves reducirá el ancho de  $h_i(x)$ . Puesto que  $H_i(x)$  deben sumar la unidad en todos lados, la función de transferencia resultante pasabanda translapará sus bordes. Tal construcción es mostrada en la Figura 3.11. Aquí, cada uno de los bordes pasabanda son subidos medio ciclo del coseno. La resultante estrecha de las respuestas impulso es evidente.

La Figura 3.12 muestra las salidas del banco de filtros con la señal de la Figura 3.7a como entrada y filtros pasabanda pasivos. Nótese el mejoramiento en localización. Así tenemos que tomar un conjunto hacia un análisis *tiempo-frecuencia* de la señal compuesta. Esto es, tenemos medios de localización de componentes transitorios de la señal en ambos tiempos (o posición) y frecuencia.

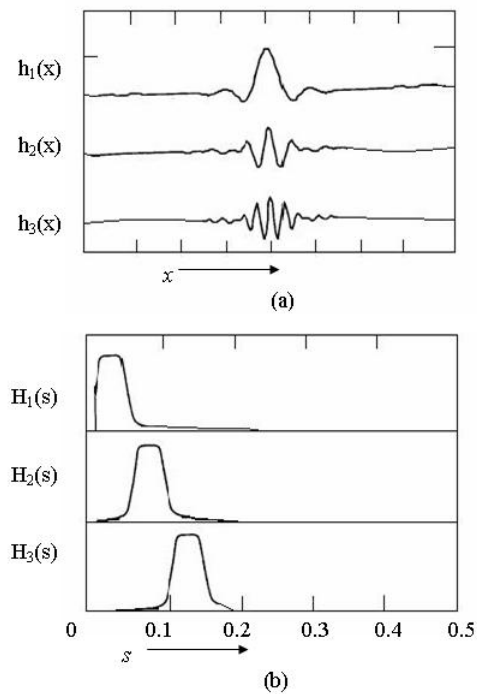


Figura 3.11: Filtros pasabanda pasivos: (a) respuestas impulso ; (b) funciones de transferencia.

### 3.3.4. Análisis multiresolución

Muchos de los desarrollos que preceden al análisis wavelet generalmente llamado *análisis multiresolución*. Estos desarrollos fueron pensados para combatir las limitaciones de Fourier mencionada en el inicio del capítulo. Ahora, este método se resume como un trabajo de campo dirigido al moderno análisis wavelet.

La teoría de banco de filtros ofrece un conveniente medio de representación de señales compuestas de componentes oscilatorios, tal como notas de música y ráfagas de tonos. Estas componentes incluyen varios (o muchos) ciclos de oscilación dentro de

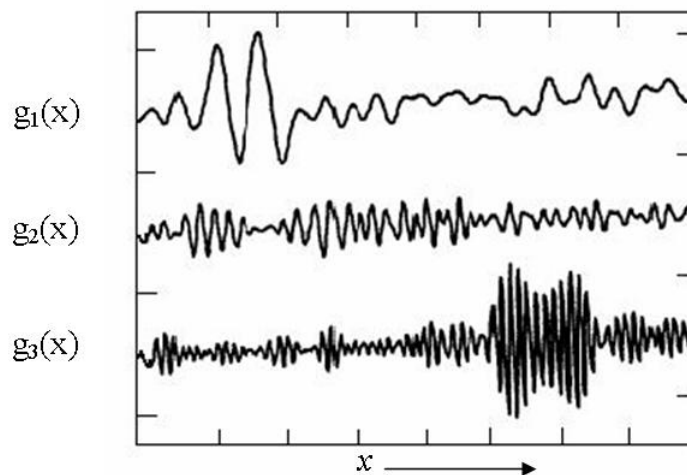


Figura 3.12: Salida del banco de filtros pasabanda pasivos.

su duración. Sin embargo, en análisis de imágenes, los componentes de interés localizados muchas veces no son realmente oscilatorios, en este caso incluyen solo un ciclo igual justo en la parte de un ciclo. Algunos ejemplos incluyen líneas, bordes y manchas.

Los objetos en una imagen ocurren a diferentes tamaños de escala. Un borde, por ejemplo, puede ser uno u otro una transición aguda desde el negro al blanco o uno que ocurre gradualmente sobre una distancia considerable. En general un método multiresolución busca la representación de la imagen para explotar esta idea.

La cartografía ilustra este método. Los mapas son comunmente dibujados a diferentes escalas. La *escala* de un mapa es la razón del tamaño de un terreno actual a la de su representación en un mapa. A grandes escalas, como un globo terráqueo, mayores características como los continentes y océanos son visibles, mientras detalles como una ciudad, calles, bajadas, bajan la resolución del mapa. A pequeña escala, los detalles llegan a ser visibles y las características grandes se pierden. Así, para ser posible navegar a un punto localizado a distancia, se necesita un conjunto de mapas dibujados a diferentes escalas.

Las transformadas wavelet han sido desarrolladas a lo largo de la línea de multiresolución. Como con el análisis tiempo frecuencia, es representada en un espacio bidimensional, pero aquí el eje vertical es la escala en lugar de la frecuencia. El escalamiento es alcanzado por dilatación y contracción, la wavelet básica para formar un conjunto de funciones base.

La wavelet básica  $\psi(x)$  es escalada como  $\psi(x/a)$  (la cual es expandida si  $a > 1$  y contraída si  $a < 1$ ) para formar un conjunto de funciones base. A mayor escala  $a$ , las funciones base buscadas para grandes características son dilatadas, mientras que para menor  $a$ , ellas se contraen para los detalles finos.

### 3.3.5. Codificación subbanda

Como antecedente adicional para la transformada wavelet discreta, ahora se describe una técnica tiempo-frecuencia denominada *codificación subbanda*. Originalmente desarrollada para codificación compacta de señales de música digitalizadas, la codificación subbanda busca la descomposición de la señal (o imagen) en una banda limitada (filtrado pasabanda) de componentes y representar estas, sin redundancia, de tal manera que sea posible reconstruir la señal original sin error [31, 15, 13].

Dada una señal de banda limitada  $f(t)$ , que es

$$\mathfrak{F}\{f(t)\} = F(s) = 0 \quad \text{para} \quad |s| \geq s_{max} \quad (3.29)$$

se puede muestrear la señal con un muestreo uniforme espaciando  $\Delta t$  a la forma

$$f(i\Delta t) \quad i = 0, 1, \dots, N - 1 \quad s_{max} \leq s_N = \frac{1}{2\Delta t} \quad (3.30)$$

(Figura 3.13a) donde  $s_N$  es la frecuencia de Nyquist.

Se comienza el análisis dividiendo el eje de la frecuencia en intervalos disjuntos. Mientras cualquier subintervalo largo pueda ser usado, ahora se elige  $s_N/2$ , como se muestra en la Figura 3.13b, por razones que llegarán a ser claras después. Aquí, el espectro  $F(s)$  es periódico con periodo  $2s_N$ .

### 3.3.6. La media banda inferior

La Figura 3.13b muestra un filtro pasabajos de media banda,  $h_0(i\Delta t)$ , llamado así porque pasa sólo la frecuencia de banda  $[-s_N/2, s_N/2]$ , la cual es la mitad de la baja frecuencia de la banda total de frecuencia  $[-s_N, s_N]$ . La respuesta impulso y función de transferencia de  $h_0$  son

$$h_0(t) = \text{sen} \left( \pi \frac{t}{2\Delta t} \right) \quad \text{y} \quad H_0(s) = \Pi \left( \frac{s}{s_N} \right) \quad (3.31)$$

donde el pulso rectangular es

$$\Pi(x) = \begin{cases} 1 & ; |x| < \frac{1}{2} \\ \frac{1}{2} & ; |x| = \frac{1}{2} \\ 0 & ; |x| > \frac{1}{2} \end{cases} \quad (3.32)$$

y

$$\text{sen}(x) = \frac{\text{sen}(x)}{x} \quad (3.33)$$

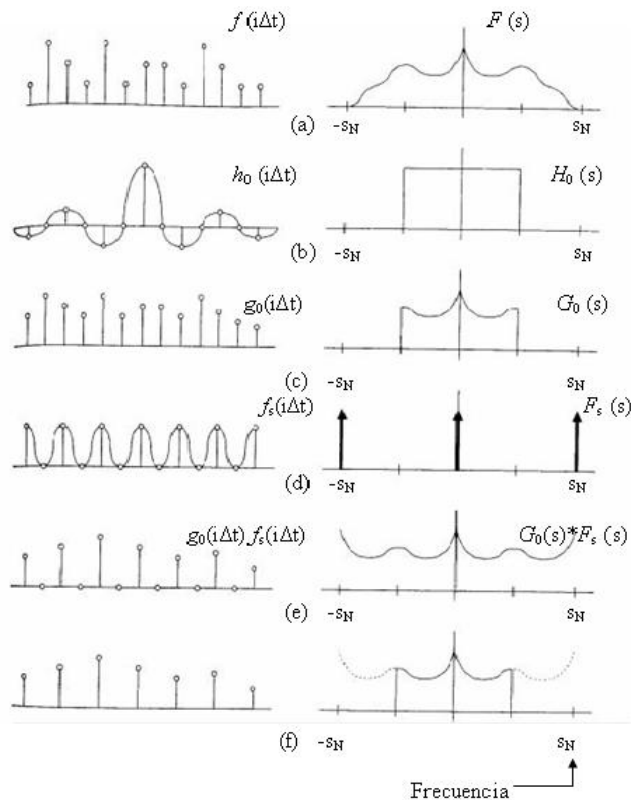


Figura 3.13: Codificación subbanda, la media banda inferior: (a) una señal muestreada y su espectro de banda limitado, (b) el filtro pasabajas de media banda ideal, (c) filtrado pasabajas de la señal, (d) la función submuestreada, (e) muestreo singular reemplazado con ceros, (f) muestreo singular de puntos descartados.

Aplicando estos filtros a  $f(i\Delta t)$  (Figura 3.13a) produce la señal  $g_0(i\Delta t)$  (Figura 3.13c) la cual es de banda limitada a  $s = s_N/2$ . Esto es una versión de baja resolución de  $f(i\Delta t)$ . Ésta retiene la forma básica de  $f(i\Delta t)$ , pero ha perdido los detalles.

Ya que  $g_0(i\Delta t)$  no tiene energía por encima de  $s_N/2$ , puede ser muestreada con muestreo de espaciamiento tan largo como  $2\Delta t$  sin introducir translapamiento. En efecto, podemos descartar cada segundo del muestreo y representar  $g_0$  sólo con los restantes  $N/2$  muestreos (Figura 3.13f). Este proceso es denominado *submuestreo* o *decimación*.

Podemos modelar el submuestreo primero multiplicando la señal por una función de submuestreo que maneja muestras impares numeradas a cero y descartando las muestras impares numeradas. Tal que la función de submuestreo

$$f_s(i\Delta t) = \frac{1}{2}[1 + \cos(2\pi s_N i\Delta t)] \quad (3.34)$$

y su espectro

$$F_s(s) = \frac{1}{2}[\delta(s) + \delta(s - s_N) + \delta(s + s_N)] \quad (3.35)$$

son mostrados en la Figura 3.13d.

Donde se multiplica la señal  $g_0(i\Delta t)$  por  $f_s(i\Delta t)$ , convolucionamos su espectro con  $F_s(s)$ . El resultado es hacer el espectro simétrico en tal forma que su periodo es reducido desde  $2s_N$  hasta  $s_N$  como se muestra en la Figura 3.17e. Su amplitud es también cortada por la mitad, escribimos

$$F_s * G_0(s) = \frac{1}{2}G_0(s) + \frac{1}{2}G_0(s + s_N) + \frac{1}{2}G_0(s - s_N) \quad (3.36)$$

Claramente, ahora se puede descartar los puntos de muestreo de impares numerados sin pérdida de información (Figura 3.13f). Esto reduce la frecuencia de Nyquist a  $s_N/2$  y deja una señal que es propiamente muestreada con espaciamiento de muestreo de  $2\Delta t$ .

La información no ha sido pérdida en el proceso de submuestreo  $g_0(i\Delta t)$ . Para ver esto, nótese que se puede recuperar  $g_0(i\Delta t)$  desde la señal submuestreada en la Figura 3.13f simplemente por: (1) calculando su espectro discreto (punto  $N/2$ ), (2) rellenando con ceros desde  $s_N/2$  hasta  $s_N$  para reconstruir  $G_0(s)$ , y (3) tomando la DFT inversa (punto  $N$ ) de  $G_0(s)$  para reconstruir  $g_0(i\Delta t)$ , la señal mostrada en la Figura 3.13c. Mientras que, este no es el método preferido, se argumenta que el submuestreo  $g_0(i\Delta t)$  no produce pérdida de información.

Una manera simple de recuperar  $g_0(i\Delta t)$  puede ser también vista en la Figura 3.13. Primero se sobremuestra la señal de banda baja (Figura 3.13f) insertando los ceros muestreo impares-numerado (para formar la Figura 3.13e). Entonces se filtra

la señal con  $2h_0(i\Delta t)$ , el filtro pasabanda de media banda ideal (Figura 3.13b). Este reconstruirá el espectro, y por lo tanto, la señal, en la Figura 3.13c, en relación con eso recobra  $g_0(i\Delta t)$ .

En el dominio de la frecuencia se escribe

$$F_s * G_0(s) \times H_0(s) = \left[ \frac{1}{2}G_0(s) + \frac{1}{2}G_0(s + s_N) + \frac{1}{2}G_0(s - s_N) \right] \times \prod \left( \frac{s}{s_N} \right) = \frac{1}{2}G_0(s) \quad (3.37)$$

Nótese que el filtro pasabajas de respuesta al impulso,  $h_0(i\Delta t)$ , es  $\text{sen}(\pi x/2\Delta t)$ , la cual tiene cruces por cero en múltiplos del muestreo espaciado, excepto en cero (Figura 3.13b). Así, sus valores de interpolación intermedios (impar- numerado) de  $g_0(i\Delta t)$ , donde los ceros son localizados, y dejan par- numerado de muestreos.

### 3.3.7. La media banda superior

Ahora volviendo a la media banda superior de  $f(i\Delta t)$  (Figura 3.14a), se puede aislar esa energía con un filtro pasabanda ideal de media banda (Figura 3.14b). Esta respuesta impulso del filtro y función de transferencia son, repectivamente

$$h_1(t) = \delta(t) - \text{sen} \left( \pi \frac{t}{2\Delta t} \right) \quad \text{y} \quad H_1(s) = 1 - \Pi \frac{s}{s_N} \quad (3.38)$$

donde  $\Pi(x)$  es como en la Ecuación 3.39.

El filtro produce una señal,  $g_1(i\Delta t)$ , cuyo espectro es diferente de cero sólo en la media banda superior (Figura 3.14c). Esta señal contiene exactamente la información de alta frecuencia que fue eliminada desde  $f(i\Delta t)$  por el filtro pasabajas en la Figura 3.13b. Así,  $g_0(i\Delta t)$ , y  $g_1(i\Delta t)$  contiene toda la información que estaba presente en la señal original,  $f(i\Delta t)$ . En efecto

$$f(i\Delta t) = g_0(i\Delta t) + g_1(i\Delta t) = f(i\Delta t) * h_0(i\Delta t) + f(i\Delta t) * h_1(i\Delta t) \quad (3.39)$$

donde

$$H_0(s) + H_1(s) = 1 \quad (3.40)$$



La Figura 3.14d muestra la función submuestreo  $f_s(i\Delta t)$  que fue usada en el análisis de la sección anterior. Cuando  $g_1(i\Delta t)$  (Figura 3.14c) es submuestreada por  $f_s(i\Delta t)$ , su espectro es convolucionado con  $F_s(s)$ . Esto, ocupa el intervalo  $[-s_N/2, s_N/2]$  con una replica de su espectro y produce el espectro mostrado en la Figura 3.14e. Se escribe

$$F_s(s) * G_1(s) = \frac{1}{2}G_1(s) + \frac{1}{2}G_1(s + s_N) + \frac{1}{2}G_1(s - s_N) \quad (3.41)$$

Este espectro es ahora periódico con período  $s_N/2$  puede ser muestreado en espacios  $2\Delta t$  sin traslapes. Así, ahora se tiene otra señal que es encerrada por la media banda inferior, y puede ser submuestreada como antes (Figura 3.14f).

Esto deja el punto  $N$  de la señal  $f(i\Delta t)$  en dos puntos  $N/2$  de la señal. Se ha visto que  $g_0(i\Delta t)$  puede ser recuperada desde la señal de banda baja. Sólo resta mostrar que  $g_1(i\Delta t)$  puede ser recuperada desde la señal de banda alta para ver que  $f(i\Delta t)$  (y de aquí  $f(t)$ ) puede ser reconstruida sin error.

La Figura 3.14e muestra la señal de banda alta sobremuestreada. Su espectro es idéntico al de la Figura 3.14f, excepto que después del sobremuestreo, la frecuencia es otra vez  $s_N$ . Se puede reconstruir  $G_1(s)$ , y así  $g_1(i\Delta t)$  (Figura 3.14c), simplemente filtrando esta señal sobremuestreada con  $2h_1(i\Delta t)$  (Figura 3.14b) para eliminar la energía de baja frecuencia. Esto es

$$F_s * G_1(s) \times H_1(s) = \left[ \frac{1}{2}G_1(s) + \frac{1}{2}G_1(s + s_N) + \frac{1}{2}G_1(s - s_N) \right] \times \left[ 1 - \Pi\left(\frac{s}{s_N}\right) \right] = \frac{1}{2}G_1(s) \quad (3.42)$$

Así, se tiene, en dos canales de codificación subbanda, una representación invertible de la señal en términos de dos filtros de salidas discretas submuestreadas, y está sin redundancia (i.e., no sobrecompleta).

### 3.3.8. Translapando la media banda superior

Claramente, submuestreando  $g_1(i\Delta t)$  descartando todos los otros puntos de muestreo resultarán en traslape. La energía en frecuencias entre  $s_N/2$  y  $s_N$  será aliased down para el intervalo  $[0, s_N]$  como se indica en la Figura 3.15a. Sin embargo, estos procesos, no son destructivos, puesto que el intervalo es ya libre. Esto produce el espectro mostrado en la Figura 3.15b, el cual es de banda limitada a  $s_N/2$  y contiene toda la energía de  $g_1(i\Delta t)$ .

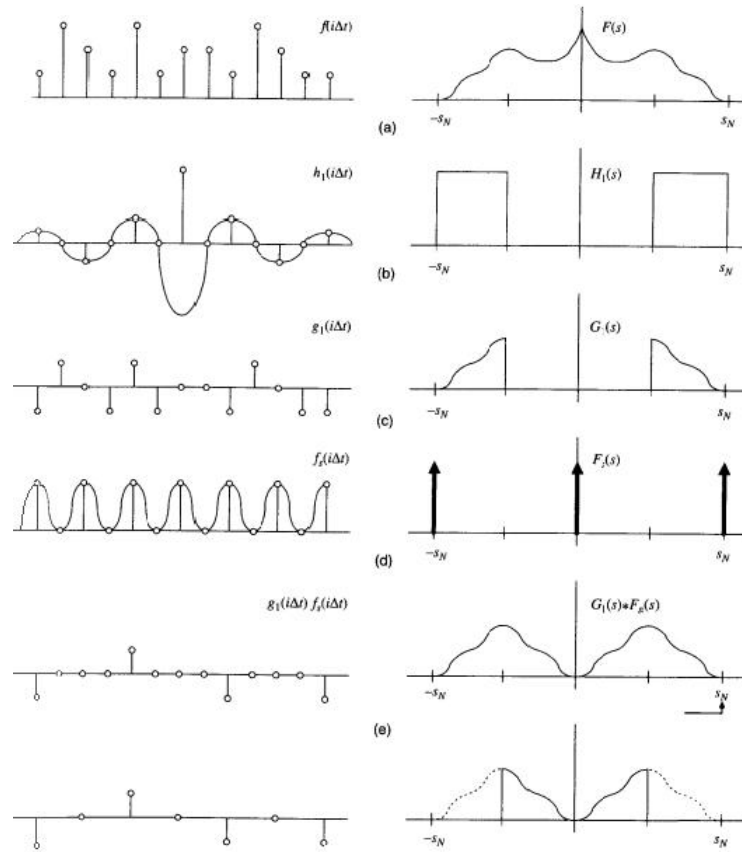


Figura 3.14: Codificación subbanda, la media banda superior: (a) una señal muestreada y su espectro de banda limitado, (b) el filtro pasaaltas de media banda ideal, (c) filtrado pasaaltas de la señal, (d) la función submuestreada, (e) muestreo singular reemplazado con ceros, (f) muestreo singular de puntos descartados.

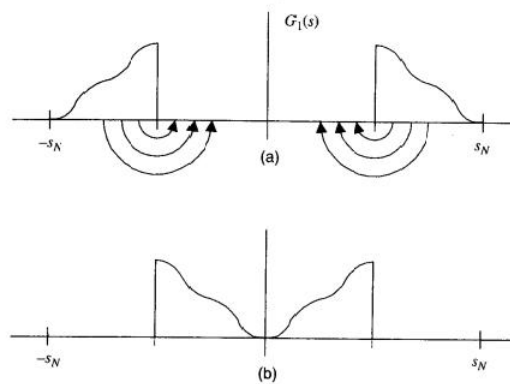


Figura 3.15: Translapando la media banda superior: (a) translapando el espectro en la Figura 3.14c, (b) resultado del translaje.

Para el submuestreo sólo es necesario  $g_1(i\Delta t)$  para obtener la señal de subbanda codificada. Además, sobremuestreando, seguido de un filtrado con  $2h_1(i\Delta t)$ , se recuperará  $g_1(i\Delta t)$ .

### 3.3.9. Codificación y decodificación subbanda

Entonces, la codificación subbanda de dos canales, requiere sólo el filtrado de  $f(i\Delta t)$  con  $h_0(i\Delta t)$  y  $h_1(i\Delta t)$ , seguido por el submuestreo de cada salida. Esto produce las dos señales subbanda de media longitud

$$g_0(k\Delta t) = \sum_i f(i\Delta t)h_0((-i + 2k)\Delta t) \quad (3.43)$$

y

$$g_1(k\Delta t) = \sum_i f(i\Delta t)h_1((-i + 2k)\Delta t) \quad (3.44)$$

La reconstrucción es efectuada mediante sobremuestreo las señales subbanda inferior y superior, interpolandolas con  $2h_0(i\Delta t)$  y  $2h_1(i\Delta t)$ , respectivamente, y sumandolas a la vez. Esto es dado por

$$f(i\Delta t) = 2 \sum_k [g_0(k\Delta t)h_0((-i + 2k)\Delta t) + g_1(k\Delta t)h_1((-i + 2k)\Delta t)] \quad (3.45)$$

y es ilustrado en la Figura 3.20.

Se tiene un ligero problema en el punto de media frecuencia  $s = s_N/2$  puesto que la decodificación ocasiona el filtrado doble de  $f(i\Delta t)$ , una vez de  $h_0(i\Delta t)$  y una con  $h_1(i\Delta t)$  y puesto que  $H_0(s_N/2) = 1/2$  y  $H_1(s_N/2) = 1/2$ . Este problema puede ser evitado usando  $\Pi(\pm\frac{1}{2}) = \sqrt{\frac{1}{2}}$  en la Ecuación 3.35. En la siguiente sección, se usan mas filtros pasabanda, y se puede manejar la situación explícitamente.

Se puede haber elegido para particionar el eje de la frecuencia en  $M$  intervalos cortos de longitud  $2s_N/M$ , produciendo  $M$  señales subbanda de  $N/M$  puntos, como es comunmente hecho en codificación subbanda. Entonces, diferentes componentes de frecuencia aparecen en canales subbanda separados. Dado que son movidas hacia la DWT, sin embargo, nos fijamos en la elección de dos subbandas ( $M = 2$ ).

### 3.3.10. El algoritmo de la transformada wavelet rápida

Mallat [23] definió un algoritmo para la transformada wavelet discreta que es más eficiente que calcula un conjunto total de productos internos. Esto aplica en codificación subbanda de doble banda en una forma iterativa y construye la transformada

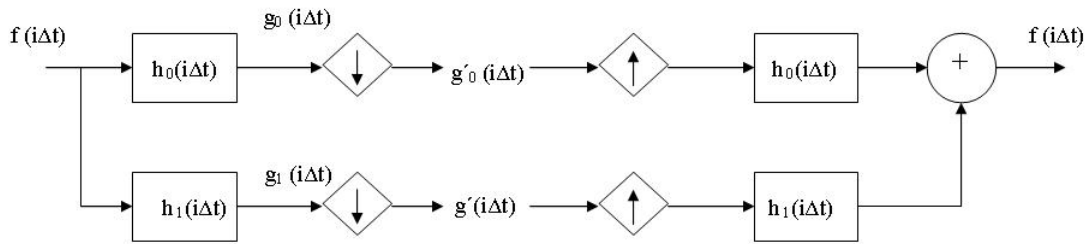


Figura 3.16: Codificación subbanda de dos bandas y reconstrucción.

wavelet desde la base de arriba, que es, calcular primero los coeficientes de menor escala.

Después del primer paso de codificación subbanda como se describió en la sección anterior, la señal subbanda inferior,  $g_0(i\Delta t)$ , es otra vez sujeta a codificación de subbanda de media banda. Esto deja el punto  $N/2$  por encima de la señal de media banda y dos puntos  $N/4$ , las señales subbanda corresponden al primer y segundo cuarto del intervalo  $[0, s_N]$ .

El proceso es continuo, en cada paso se retiene la señal de mediabanda superior y además la señal de media banda inferior, hasta que un punto de la banda baja de la señal es obtenido. Los coeficientes de la transformada son entonces el punto de banda baja y la colección de códigos subbanda más altos de señales de media banda. Esto es mostrado en la Figura 3.17. Los primeros coeficientes  $N/2$  llegan desde la media banda superior de  $F(s)$ , los siguientes puntos  $N/4$  vienen del segundo cuarto de banda etc.

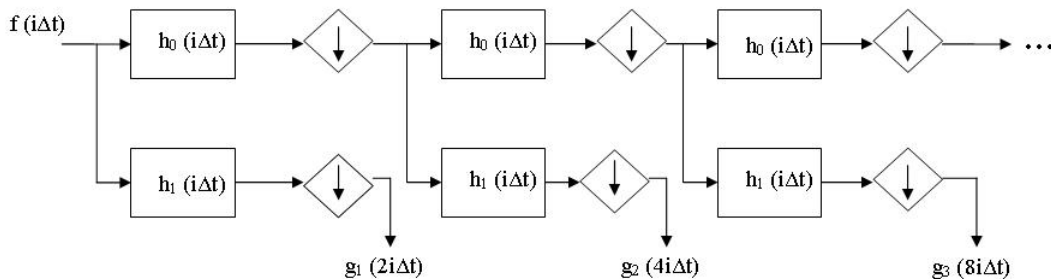


Figura 3.17: El algoritmo de la transformada wavelet discreta.

La respuesta impulso,  $h_j$ , dobla en escala a cada iteración. Así, se tiene una trans-

formada wavelet ortonormal. La wavelet básica es  $h(t) = \delta(t) - \text{sen}(at)$ , y las funciones base son  $2^{-j/2}h(2^jt - n)$  [18]. Así, la codificación subbanda, la cual es básicamente una técnica de transformación tiempo-frecuencia, ha sido empleada para definir una transformada wavelet tiempo-escala.

El algoritmo resultante es a veces referido como la *transformada rápida wavelet* (Fast Wavelet Transform, FWT), o algoritmo de Mallat, debido a la apariencia del diagrama en la Figura 3.17. La transformada inversa es obtenida revirtiendo el proceso, como se muestra en la Figura 3.18.

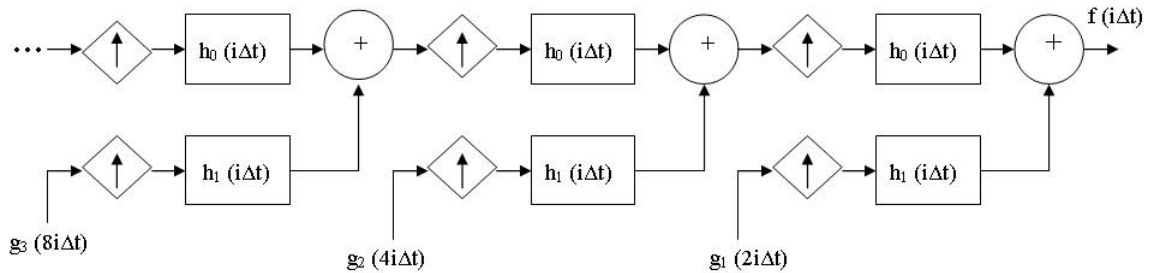


Figura 3.18: La transformada wavelet discreta inversa.

### 3.3.11. Funciones base

En la Figura 3.17 se puede ver que cada conjunto de coeficientes de la transformada es obtenido por convolución de  $f(i\Delta t)$  repetidamente con  $h_0(i\Delta t)$  y una vez con  $h_1(i\Delta t)$ . Así, las funciones base de esta transformada wavelet son  $h_1(i\Delta t)$  y otras funciones derivadas por convolución de  $h_1(i\Delta t)$  repetidamente con  $h_0(i\Delta t)$ . Esto es explorado con más detalle en la siguiente sección.

### 3.3.12. Diseño de la transformada wavelet discreta

Ahora se está preparado para entrar al diseño de la wavelet base para usar en una transformada wavelet discreta. Como en las secciones anteriores, no es necesario que los filtros implementados en un banco de filtros sean pasabajas o pasaaltas ideales. Similarmente, para la DWT, se puede usar cualquier par de filtros de codificación subbanda que permite mantener la Ecuación (3.44).

Escribiendo 3.48 en el dominio de la frecuencia, se tiene

$$F(s) = 2 \left[ \frac{1}{2} G_0(s) H_0(s) + \frac{1}{2} G_1(s) H_1(s) \right] = 2 \left[ \frac{1}{2} F(s) H_0(s) H_0(s) + \frac{1}{2} F(s) H_1(s) H_1(s) \right] \quad (3.46)$$

lo cual significa que

$$F(s) = F(s) [H_0^2(s) + H_1^2(s)] \quad (3.47)$$

y las dos funciones de transferencia para el filtro deben satisfacer la condición

$$H_0^2(s) + H_1^2(s) = 1 \quad \text{para} \quad 0 \leq |s| \leq s_N \quad (3.48)$$

Las funciones de transferencia son cuadradas porque  $f(t)$  es convolucionada dos veces con cada filtro, una vez durante la codificación y una vez durante la decodificación.

Suponga que  $H_0(s)$  es una función de transferencia pasabajas suave que se desea usar en una transformada wavelet. Claramente la correspondiente  $H_1(s)$  esta dada por

$$H_1^2(s) = 1 - H_0^2(s) \quad (3.49)$$

Así, un filtro pasabaja bien seleccionado es todo lo que se requiere para diseñar una transformada wavelet discreta.

### 3.3.13. Filtros espejo

Comparando la Figura 3.17b y 3.18b, se ve que, para el caso de un filtro pasabanda ideal,  $h_1(i\Delta t)$  puede ser visto como  $h_0(i\Delta t)$ , desplazada por una cantidad  $s_N$  a lo largo del eje de la frecuencia. Acorde al teorema de desplazamiento

$$\mathfrak{F}^{-1}\{H(s - a)\} = e^{j2\pi at} h(t) \Rightarrow \mathfrak{F}^{-1}\{H(s) s_N\} = e^{j2\pi(\frac{1}{2\Delta t} i\Delta t)} h(i\Delta t) = (-1)^i h(i\Delta t) \quad (3.50)$$

tal que un desplazamiento de medio período del espectro pueda ser desplazado simplemente cambiando el signo de las muestras impares de la señal. Nótese el uso de la unidad imaginaria en la Ecuación (3.53).

Se puede usar esta aproximación en el diseño de más filtros subbanda generales. Seleccionando  $h_1(i\Delta t)$  tal que

$$h_1((N - 1 - i)\Delta t) = (-1)^i h_0(i\Delta t) \quad (3.51)$$

donde  $N$  es la longitud de  $h_0(i\Delta t)$ , así se obtienen los correspondientes filtros pasaaltas. El filtro  $h_1(i\Delta t)$  es denominado *filtro espejo* de  $h_0(i\Delta t)$ . Si  $h_0(i\Delta t)$  es de duración corta, puede ser asegurado que  $h_1(i\Delta t)$  será corta también.

La propiedad de simetría que  $H_0$  debe tener en orden para mantener la Ecuación 3.56, y para su entera aproximación para trabajar, es

$$H_0^2\left(\frac{s_N}{2} + s\right) = 1 - H_0^2\left(\frac{s_N}{2} - s\right) \quad (3.52)$$

### 3.3.14. El vector de escalamiento

Entonces, para desarrollar una transformada wavelet discreta, solo se necesita un filtro pasabajas discreto con respuesta al impulso  $h_0(k)$  que reúna ciertas condiciones. Esta respuesta impulso es a veces llamada *vector de escalamiento*.

Puesto que  $h_0(k)$  puede generar una función relacionada  $\phi(t)$ , llamada función de escalamiento. También se puede generar  $h_1(k)$  y, desde esta y  $\phi(t)$ , la wavelet básica,  $\psi(t)$ . Si el vector de escalamiento tiene sólo un número finito de entradas diferentes de cero, entonces  $\phi(t)$ ,  $\psi(t)$ , y todas las wavelets resultantes tendrán soporte compacto. Esto es, ellos serán cero fuera de un relativamente corto intervalo sobre el eje  $t$ .

En realidad, si se tiene uno u otro  $h_0(k)$ , o  $\phi(t)$ , se puede usar para generar la otra. Esto es usualmente sencillo para iniciar con  $h_0(k)$ , lo cual debe satisfacer la Ecuación (56). Dejando al vector de escalamiento una sucesión tal que

$$\sum_k h_0(k) = \sqrt{2} \quad \text{y} \quad \sum_k h_0(k)h_0(k + 2l) = \delta(l) \quad (3.53)$$

Entonces existe una función de escalamiento

$$\phi(t) = \sum_k h_0(k)\phi(2t - k) \quad (3.54)$$

que pueden ser construidos como una copia de su suma ponderada de mitad de escala, usando  $h_0(k)$  como la ponderación. Desde la observación en una sección anterior,  $\phi(t)$  puede también ser calculada por convolución repetida de  $h_0(k)$  con versiones escaladas del pulso de función rectangular (Figura 3.23); que es

$$\phi(x) = \lim_{i \rightarrow \infty} \eta_i(x) \quad (3.55)$$

donde

$$\eta_i(x) = \sqrt{2} \sum_n h_0(n) \eta_{i-1}(2x - n) \quad (3.56)$$

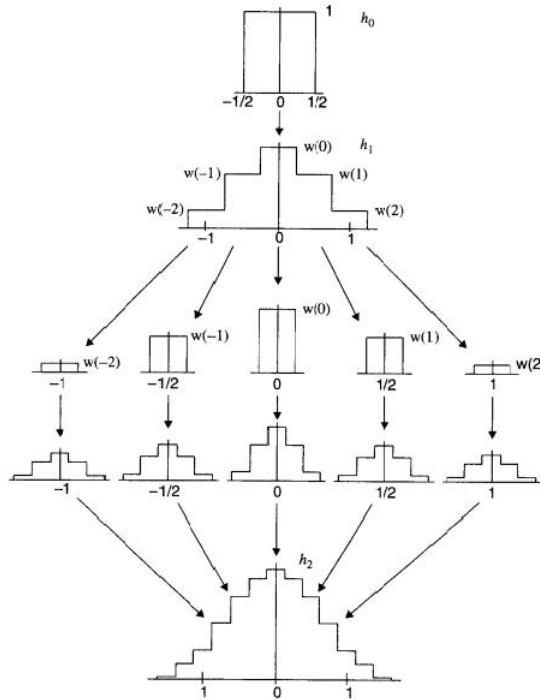


Figura 3.19: Construyendo la función de escalamiento (después Daubechies).

es un piecewise aproximación constante a  $\phi(t)$  y

$$\eta_0(x) = \Pi(x) = \begin{cases} 1 & ; |x| < \frac{1}{2} \\ \frac{1}{2} & ; |x| = \frac{1}{2} \\ 0 & ; |x| > \frac{1}{2} \end{cases} \quad (3.57)$$

Nótese que la primera iteración crea un trazo de función constante que teniendo los valores de  $h_0(k)$ . Además, la resolución de las aproximaciones dobles, y la aproximación llega a ser suave, con cada iteración. Por ejemplo, nueve iteraciones, tomarán una secuencia de cuatro puntos en una función muestreada de 1024 puntos, y ésta es adecuada para muchas implementaciones digitales. La función escalamiento  $\phi(t)$  es, entonces, una función continua que tiene la misma forma general como el filtro pasabajas discreto de respuesta impulso  $h_0(k)$ .



Por otro lado, si se inicia con una función de escalamiento  $\phi(t)$ , ésta debe ser ortonormal bajo cambios de unidades; esto es

$$\langle \phi(t - m), \phi(t - n) \rangle = \delta_{m,n} \quad (3.58)$$

Entonces  $h_0(k)$  puede ser calculada de

$$h_0(k) = \langle \phi_{1,0}(t), \phi_{0,k}(t) \rangle \quad (3.59)$$

donde

$$\phi_{j,k}(t) = 2^{j/2} \phi(2^j t - k) \quad j = 0, 1, \dots \quad k = 0, 1, \dots, 2^j - 1 \quad (3.60)$$

Si el vector de escalamiento tiene solo un número finito de entradas diferentes de cero, entonces las wavelets resultantes tendrán soporte compacto [14].

Si una función de escalamiento deseada  $\hat{\phi}(t)$  no es ortonormal, esta puede ser usada para generar una que es ortonormal por propia normalización de su espectro,  $\hat{\Phi}(s)$ . Esto es

$$\Phi(s) = \frac{C \hat{\Phi}(s)}{\sqrt{\sum_n^{\infty = -\infty} \hat{\Phi}(s - 2\pi n)}} \quad (3.61)$$

donde  $C$  es una constante.

### 3.3.15. El vector wavelet

Una vez que se tienen en mano  $\phi(t)$  y  $h_0(k)$ , se continua el desarrollo por definición a una respuesta impulso pasaaltas discreta llamado vector wavelet así

$$h_1(k) = (-1)^k h_0(-k + 1) \quad (3.62)$$

y, de esto, una wavelet básica

$$\psi(t) = \sum_k h_1(k) \phi(2t - k) \quad (3.63)$$

de la cual se obtiene un conjunto wavelet ortonormal

$$\psi_{j,k}(t) = 2^{j/2} \psi(2^j t - k) \quad (3.64)$$

### 3.3.16. Calculando la transformada wavelet

Dado un conjunto de wavelets ortonormales, la expansión en series de wavelet de la función continua de banda limitada  $f(t)$  es

$$c_{j,k} = \int_{-\infty}^{\infty} f(t)\psi_{j,k}(t)dt \quad y \quad f(t) = \sum_{j,k} c_{j,k}\psi_{j,k}(t) \quad (3.65)$$

y la transformada wavelet discreta de la función muestreada es

$$c_{j,k} = \sum_i f(i\Delta t)\psi_{j,k}(i\Delta) \quad y \quad f(i\Delta t) = \sum_{j,k} c_{j,k}\psi_{j,k}(i\Delta t) \quad (3.66)$$

Los coeficientes y sumatorias pueden también ser indexadas sólo por el entero  $n = 0, 1, \dots, N - 1$  donde

$$n = 2^j + k \quad \text{para} \quad j = 0, 1, \dots, \log_2(N) - 1, \quad k = 0, 1, \dots, 2^j - 1 \quad (3.67)$$

Nos referimos a esto como el algoritmo *top-down*, puesto que se computes primero con coeficientes de gran escala. En contraste, el algoritmo de Mallat computes primero con coeficientes de pequeña escala.

Entonces, la tarea de diseño, implica primero encontrar una secuencia  $h_0(k)$  que satisfaga la Ecuación 3.64 y entonces construir la correspondiente función de escalamiento, o eligiendo un escalamiento ortonormal de la función y determinando  $h_0(k)$  de la Ecuación (3.58). Una función de escalamiento puede ser hecha ortonormal por la Ecuación (3.60). Entonces el vector wavelet,  $h_1(k)$ , es obtenido de la Ecuación 3.65 y la wavelet básica de la Ecuación 3.66.

Una u otra, la transformada wavelet discreta puede ser implementada directamente, por la Ecuación 3.69, o con el algoritmo FWT. El último no requiere construcción explícita de la función de escalamiento y wavelet, y es más eficiente computacionalmente.

Para ser matemáticamente preciso, las condiciones en la Ecuación 3.69 establece que las wavelet  $\psi_{j,k}(t)$  constituye una *estructura firme* y así soportar una transformada invertible. Sin embargo, no son adecuadas, para garantizar que esas funciones base serán siempre ortonormales. Lawton [21, 22] y Cohen [12] dan estrictas condiciones de ortonormalidad en  $h_0(k)$ , pero las diferencias entre una estructura firme y una transformada ortonormal son tan despreciadas que las implementaciones digitales no son afectadas. Así, puede ser satisfecha usando la Ecuación (3.52).

### 3.3.17. Ejemplos

Se ilustra la construcción de una transformada wavelet con tres ejemplos.

**Ejemplo 1.** Usando filtros pasabajas y pasaaltas ideales, se tiene [18]

$$h_0(k) = \frac{1}{\sqrt{2}} \operatorname{sinc}\left(\pi \frac{k}{2}\right) \quad \text{y} \quad h_1(k) = \sqrt{2} \delta(k) - h_0(k) \quad (3.68)$$

y

$$\phi(t) = \operatorname{sinc}(\pi t) \quad \text{y} \quad \psi(t) = 2\phi(2t) - \phi(t) \quad (3.69)$$

Esto da una transformada wavelet discreta basada en *sinc wavelets* (Figura 3.11). Nótese que estas wavelets no tienen soporte compacto.

**Ejemplo 2.** Si se toma

$$h_0(k) = \begin{cases} \frac{1}{\sqrt{2}} & ; k = 0, 1 \\ 0 & ; \text{en otro caso} \end{cases} \quad (3.70)$$

y

$$h_1(k) = \begin{cases} \frac{1}{\sqrt{2}} & ; k = 0 \\ \frac{-1}{\sqrt{2}} & ; k = 1 \\ 0 & ; \text{en otro caso} \end{cases} \quad (3.71)$$

entonces  $\psi(t)$  es la función Haar, y nos conduce a la transformada Haar. Este vector de escalamiento tiene dos entradas diferentes de cero, y, como se esperaba, la transformada Haar tiene soporte compacto.

**Ejemplo 3.** La secuencia  $h_0(k)$ , tiene cuatro elementos diferentes de cero y están dados por

$$4\sqrt{2}h_0(k) = \begin{cases} 1 + \sqrt{3} & ; k = 0 \\ 3 + \sqrt{3} & ; k = 1 \\ 3 - \sqrt{3} & ; k = 2 \\ 1 - \sqrt{3} & ; k = 3 \\ 0 & ; \text{en otro caso} \end{cases} \quad (3.72)$$

satisfacen la Ecuación 3.55 y es así un vector de escalamiento. Esta es una función escalamiento y wavelet, construida por el procedimiento esbozado en la Figura 3.19,

y mostrado en la Figura 3.20. Esta es una de las familias de secuencias de longitud finita que da origen a wavelets ortonormales teniendo soporte compacto. La familia es mas discutida en la siguiente sección.

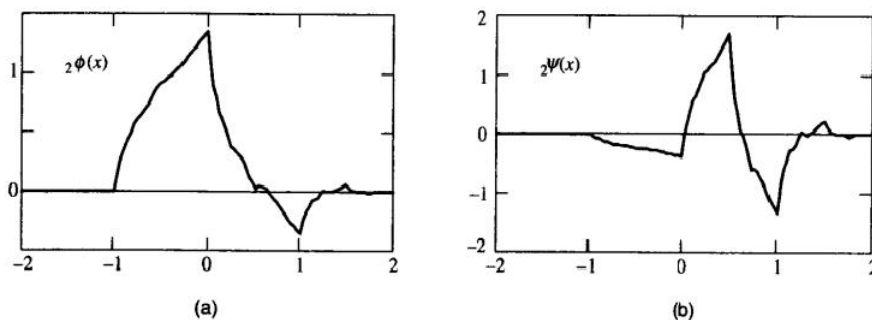


Figura 3.20: Daubechies, (a) escalamiento de la función y (b) wavelet para  $r = 2$ .

### 3.3.18. Wavelets con soporte compacto ortonormal

Daubechies [14] ha construido una familia,  $\{\psi(x)\}$ , de wavelets ortonormales teniendo soporte compacto. Para cada valor integral del índice  $r$  el conjunto de wavelets

$$\{r\psi_{j,k}(x)\} = 2^{j/2}\psi(2^j x - k) \quad (3.73)$$

donde  $j$  y  $k$  son enteros, y forman una base ortonormal. Además,  $r\psi(x)$  es cero fuera del intervalo  $[0, 2r - 1]$ , sus primeros  $r$  momentos desaparecen, esto es

$$\int_{-\infty}^{\infty} x^n \psi(x) dx = 0 \quad n = 0, 1, \dots, r \quad (3.74)$$

y su número de derivadas continuas es aproximadamente  $r/5$ . Esto describe en vez de un buen comportamiento, o *regular*, a un grupo de funciones. Interesantemente,  $1\psi(x)$  es la wavelet básica de la transformada Haar.

La Tabla 3.1 muestra las secuencias  $h_0(k)$  que generaron las wavelets ortonormales para  $r = 3, 5, 7$ , y  $9$ . La técnica de construcción fue descrita en la sección . La Figura 3.21 muestra gráficas de las wavelets correspondientes. Nótese que estas funciones llegan a ser amplias y más regulares con incremento de  $r$ .

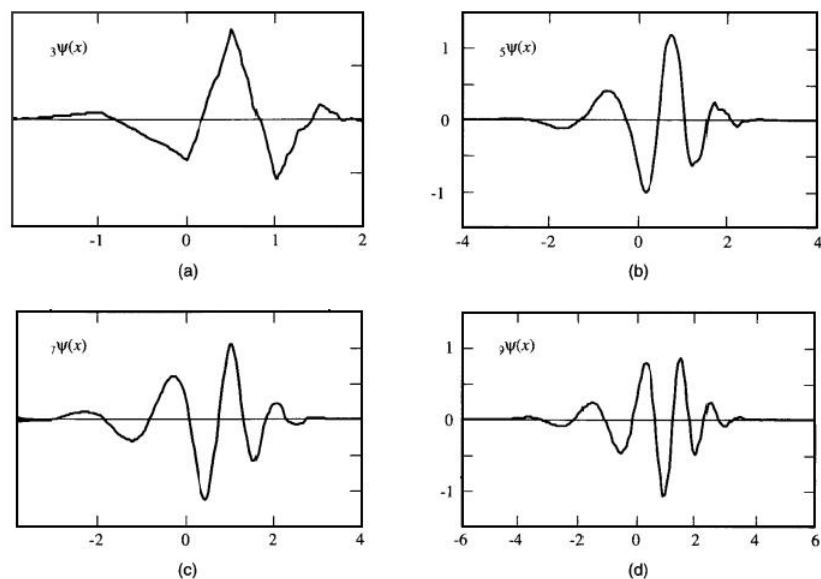


Figura 3.21: Wavelets ortonormales para: (a)  $r = 3$ , (b)  $r = 5$ , (c)  $r = 7$ , y (d)  $r = 9$ .

**Tabla 3.1 Sucesiones de filtros discretos para las wavelets ortonormales en la Fig.3.20**

---

0.3327	0.8069	0.4599	-0.1350	-0.0854	0.0352				
0.1601	0.6083	0.7243	0.1384	-0.2423	-0.0322	0.0776	-0.0062	-0.0126	0.0033
0.0779	0.3965	0.7291	0.4698	-0.1439	-0.2240	0.0713			
0.0806	-0.0380	-0.0166	0.0126	0.0004	-0.0018	0.0004			
0.0381	0.2438	0.6048	0.6573	0.1332	-0.2933	-0.0968	0.1485	0.0307	
-0.0676	0.0003	0.0224	-0.0047	-0.0043	0.0018	0.0002	-0.0003	0.0000	

---

Figura 3.22: Sucesiones de filtros discretos para las wavelets ortonormales en la Fig.3.20.

### 3.4. Transformadas wavelet biortogonales

Las funciones que califican como wavelets biortogonales con soporte compacto carecen de propiedades de simetría. Sería conveniente, por ejemplo, si  $\psi(t)$  pudiera ser una función par o impar. Usando dos wavelet base diferentes,  $\psi(x)$  y  $\tilde{\psi}(x)$  (una para descomposición (análisis) y otra para reconstrucción (síntesis)) se pueden tener wavelets simétricas con soporte compacto [10, 24, 11, 30]. Las dos wavelets son *duales* una de la otra, y las familias de wavelets  $\{\psi_{jk}(x)\}$  y  $\{\tilde{\psi}_{jk}(x)\}$  son biortogonales, esto es

$$\langle \{\psi_{j,k}, \tilde{\psi}_{l,m}\} \rangle = \delta_{j,l} \delta_{k,m} \quad (3.75)$$

Entonces se tiene

$$c_{j,k} = \langle f(x), \psi_{j,k}(x) \rangle \quad \text{y} \quad d_{j,k} = \langle f(x), \tilde{\psi}_{j,k}(x) \rangle \quad (3.76)$$

para la descomposición, y

$$f(x) = \sum_{j,k} c_{j,k} \tilde{\psi}_{j,k}(x) = \sum_{j,k} d_{j,k} \psi_{j,k}(x) \quad (3.77)$$

para la reconstrucción.

Cualquier wavelet puede ser usada para la descomposición, siempre que una o la otra sea usada para la reconstrucción. La transformada wavelet biortogonal permite el uso de wavelets simétricas (pares o impares) teniendo soporte compacto.

#### 3.4.1. Implementación

La transformada wavelet biortogonal requiere cuatro filtros discretos (vectores de respuesta al impulso). Se deben elegir filtros pasabajas (vectores de escalamiento),  $h_0(n)$  y  $\tilde{h}_0(n)$ , cuyas funciones de transferencia satisfacen

$$H_0(0) = \tilde{H}_0(0) = 1 \quad \text{y} \quad H_0(s_N) = \tilde{H}_0(s_N) = 0 \quad (3.78)$$

donde  $s_N = 1/2\Delta x$  es la frecuencia. De estos, se generan dos filtros pasabanda (vectores wavelet), como después, por cambios de medio periodo de sus funciones de transferencia

$$h_1(n) = (-1)^n h_0(1-n) \quad \tilde{h}_1(n) = (-1)^n \tilde{h}_0(1-n) \quad (3.79)$$

Ahora podemos implementar el algoritmo ( FWT ) usando estos cuatro filtros, como se muestra en la Figura 3.23

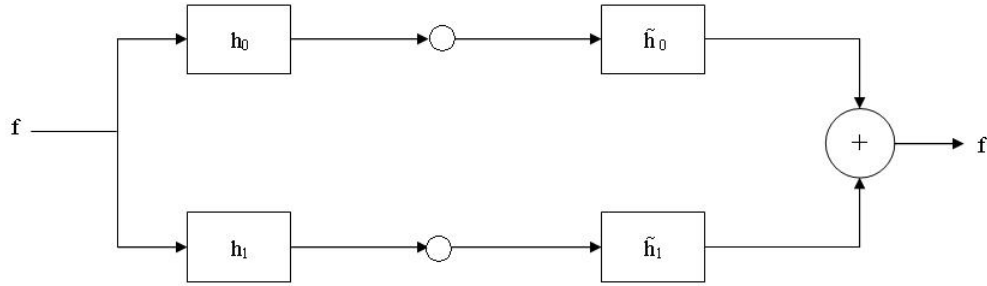


Figura 3.23: Paso de una descomposición a un paso de reconstrucción de la transformada wavelet biortogonal.

### 3.4.2. Wavelets biortogonales

Las condiciones en filtros wavelet biortogonales son

$$\sum_n h_0(n) = \sum_n \tilde{h}_0(n) = \sqrt{2} \quad \text{y} \quad \sum_n h_1(n) = \sum_n \tilde{h}_1(n) = 0 \quad (3.80)$$

y la propiedad de reconstrucción perfecta requiere que

$$H_0(s)\tilde{H}_0(s) + H_1(s)\tilde{H}_1(s) = H_0(s)\tilde{H}_0(s) + H_0(s - s_N)\tilde{H}_0(s - s_N) = 1 \quad (3.81)$$

Las dos funciones de escalamiento están dadas, en el dominio de la frecuencia, por

$$\Phi(2s) = \tilde{H}_0(s)\Phi(s) = \prod_{n=0}^{\infty} \tilde{H}_0(s/2^n) \quad \text{y} \quad \tilde{\Phi}(2s) = H_0(s)\tilde{\Phi}(s) = \prod_{n=0}^{\infty} H_0(s/2^n) \quad (3.82)$$

y entonces las wavelets son

$$\psi(x) = \sqrt{2} \sum_n h_1(n+1)\phi(2x-n) \quad \text{y} \quad \tilde{\psi}(x) = \sqrt{2} \sum_n \tilde{h}_1(n+1)\tilde{\phi}(2x-n) \quad (3.83)$$

### 3.4.3. Construyendo wavelets biortogonales

El diseño de wavelets biortogonales requiere el desarrollo de respuestas impulso discretas (vectores de escalamiento)  $h_0(n)$  y  $\tilde{h}_0(n)$  cuyas funciones de transferencia satisfacen las ecuaciones (3.73) y (3.76). Esto es un área activa de investigación, y varios autores han catalogado cada filtro y la correspondiente wavelet biortogonal.

Cohen, Daubechies, y Feauveau [11], por ejemplo, seleccionaron  $\phi(x)$  como una función *B-spline* (por ejemplo, la función triángulo) y desarrollaron  $H_0(s)$  como un polinomio en  $\cos(s)$ . Vetterli y Herley [30] presentaron propuestas basadas en la teoría de ecuaciones diofantinas y en la teoría de fracciones continuas. Generalmente, usando respuestas impulso largas dando un aumento a más wavelets regulares, esto es, éstos tendrán un número largo de momentos derivativos y disipados. La Tabla 3.2 presenta tres pares de vectores de escalamiento y la Figura 3.24 muestra las correspondientes wavelets biortogonales, construidas en el procedimiento esbozado en la Figura 3.19.

**Tabla 3.2 Sucesiones de filtros discretos para las wavelets biortonormales en la Fig. 3.22. ( de [11] y [29] )**

---

Filtro de análisis Laplaciano	$h_0 = \sqrt{2} [-.05 \ .25 \ .6 \ .25 \ -.05]$
Filtro de síntesis Laplaciano	$\tilde{h}_0 = \sqrt{2} [-.0107 \ -.0536 \ .2607 \ -.0536 \ -.0107]$
Filtro Spline 2	$h_0 = \sqrt{2} [.25 \ .5 \ .25]$
Filtro Spline 4	$\tilde{h}_0 = \sqrt{2} /128 [3 \ -6 \ -16 \ 38 \ 90 \ 38 \ -16 \ -6 \ 3]$
Filtro de análisis de 18 puntos	$h_0 = [.0012 \ -.0007 \ -.0118 \ -.0117 \ .0713 \ -.0310 \ -.2263 \ .0693 \ .7318$ $.7318 \ .0693 \ -.2263 \ -.0310 \ .0713 \ -.0117 \ -.0118 \ -.0007 \ .0012]$
Filtro de síntesis de 18 puntos	$\tilde{h}_0 = [.0012 \ .0007 \ -.0113 \ -.0114 \ .0235 \ .0017 \ -.0444 \ .2044 \ .6479$ $.6479 \ .2044 \ -.0444 \ .0017 \ .0235 \ -.0114 \ -.0113 \ .0007 \ .0012]$

---

Figura 3.24: Sucesiones de filtros discretos para las wavelets biortonormales (de [11] y [29]).

### 3.4.4. Selección de la wavelet

La wavelet básica ideal puede ser una función oscilatoria de breve duración (teniendo soporte compacto o amplitud pequeña fuera de un intervalo corto) donde todas las translaciones diádicas de escalamientos binarios de la función son ortonormales. La función Haar ilustra esto. Otras funciones wavelet disponibles pueden fallar para tener todos estos criterios.

Primero, mientras la wavelet básica debe ir a cero cuando  $|x| \rightarrow \infty$  al menos tan rápido como  $1/x$  para tener el criterio de admisibilidad, muchas wavelet pequeñas



tienen infinito soporte compacto. Esto significa que ellas son diferentes de cero sobre la línea real, excepto para su cruce en cero. Puede ser que translaciones dyadics de la wavelet a cada escala sean ortogonales, pero wavelets a diferentes escalas no lo son. Similarmente, esto puede ser que diferentes escalas de la wavelet son ortogonales, pero algunas o todas las translaciones díadicas en la misma escala no lo son.

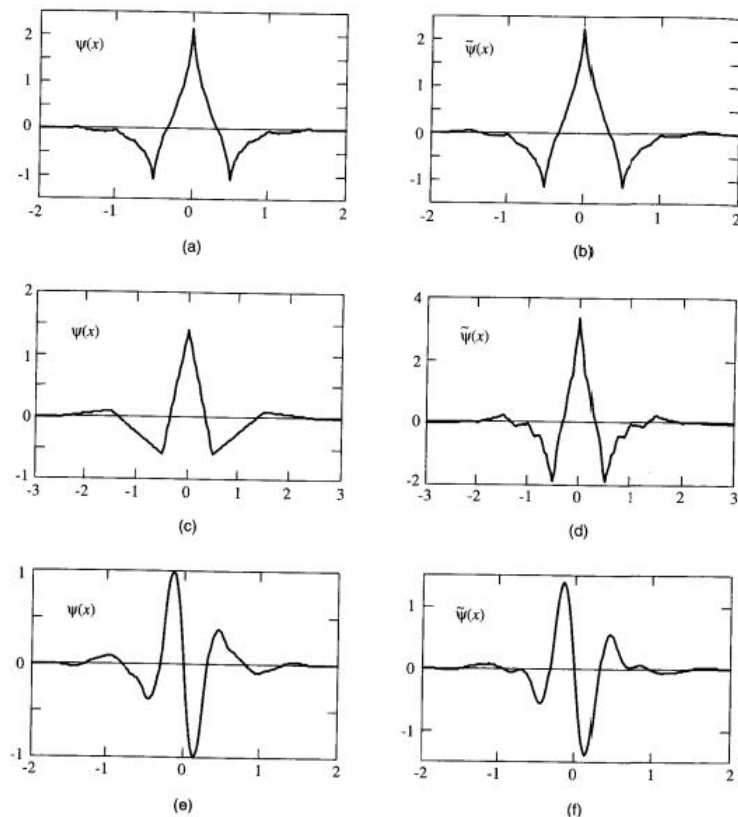


Figura 3.25: Ejemplos de wavelets biortogonales: (a) wavelet pirámide Laplaciana, (b) función wavelet spline lineal, (c) wavelet de fase lineal de 18 puntos.

Nótese que algunas transformadas wavelet (por ejemplo la ( CWT )) son sobrecompletas, mientras otras (como la DWT) no lo son. Para transformadas sobrecompletas, las restricciones en funciones base son relativamente moderadas. Para transformadas no redundantes, tal como la transformada wavelet discreta ortonormal, las restricciones localizadas en funciones base son mucho más severas.

La ( DWT ) biortogonal requiere dos vectores de escalamiento y dos vectores wavelet en vez de uno, pero esto no incrementa el cuidado computacional del proceso. Sin embargo, la transformada biortogonal proporciona una mucho más vasta elección

de la forma wavelet que la transformada ortonormal, entonces esto es preferible en muchas aplicaciones.

La elección de la función básica es usualmente gobernada por la aplicación. Para menor pérdida en la compresión, por ejemplo, una base ortonormal o biortogonal es conveniente o requerida, puesto que el objetivo es representar la función exacta o compactamente. Una transformada sobrecompleta incrementa la cantidad de datos requeridos para representar la función exactamente. Por otro lado, si la meta es pérdida en la compresión, la detección específica de componentes tal como bordes en imágenes, o remover ruido, entonces es más importante para seleccionar una wavelet que es similar a los componentes de interés.

### **Comentarios**

Una función wavelet básica es una función oscilatoria  $|x| \rightarrow \infty$ . Su espectro regenera la función de transferencia de un filtro pasabanda.

Un conjunto de funciones base para la transformada wavelet puede ser generado desde dilataciones y translaciones de una wavelet base.

La transformada wavelet discreta representa una señal como una función de dos variables: tiempo y escala.

La expansión en series de wavelet representa una señal periodica o de longitud finita con una serie de coeficientes.

La transformada Haar es la transformada wavelet discreta más simple.

La DWT puede ser implementada directa, o indirectamente, por el algoritmo de la transformada wavelet rápida (FWT).

Los sistemas wavelet biortogonales permiten a la DWT un uso menos restringido de wavelets con soporte compacto.

# Capítulo 4

## Resultados obtenidos

En este capítulo se muestran los resultados que se obtienen al aplicar las herramientas matemáticas, como la transformada de Fourier y la transformada wavelet de Daubechies 10 a las señales sísmicas obtenidas del sitio web de Virgo Seismic Monitor empleando para esto, medios digitales como el DSP TMS3206711 de Texas Instruments.

Mediante la transformada de Fourier se puede observar el espectro de una señal, lo cual quiere decir, que muestra la o las bandas de frecuencia que posee dicha señal; que, para el caso de las señales sísmicas analizadas son de baja frecuencia (del rango de los 0.03 - 50 Hz).

La transformada Wavelet brinda la descomposición de la señal sísmica en distintas aproximaciones y detalles, con lo cual se obtiene sólo información necesaria y útil de la señal, permitiendo así eliminar datos innecesarios (ruido).

### 4.1. La señal sísmica empleada

Las señales sísmicas obtenidas, fueron almacenadas en archivos con extensión .dat, que pueden ser interpretados a través de distintos entornos de simulación, para visualizar información detallada sobre los citados archivos. La simulación es una etapa importante dentro del diseño y construcción de sistemas, debido a que ofrece una perspectiva del funcionamiento del sistema, que permitirá ajustar su comportamiento y manipular las variables involucradas como mejor convenga.

En la base de datos del sitio Virgo Seismic monitor [4] fueron encontradas y extraídas las señales empleadas en esta tesis, las señales son descompuestas mediante

un software denominado Crimson Editor[6] debido a que las señales contenían cinco columnas de datos en las cuales; la primera es el tiempo (dado en GPS) y las siguientes cuatro corresponden a la velocidad de las muestras (en  $Microns/s^2$ ).

La velocidad del suelo es medida por un velocímetro tri-axial de banda amplia ( $Guralp40T - 30s$ ) situado cerca de la torre Virgo BS (en Pisa, Italia) y unido sólidamente a la losa de concreto del edificio.

Según datos obtenidos por Virgo, se han clasificado las bandas de frecuencia para una señal sísmica en las siguientes:

- Región de muy baja frecuencia ( $30 - 100mHz$ ): el movimiento del suelo en estas frecuencias muy bajas es excitado de vez en cuando por el componente lento de la onda de terremotos distantes pero muy intensos.
- Región de baja frecuencia ( $0.1 - 1.0Hz$ ): el movimiento del suelo en este rango es excitado principalmente por la acción de las ondas del mar (Microsismo oceánico). La excitación es más intensa en dirección horizontal.
- Región intermedia ( $1.0 - 4.0Hz$ ): el movimiento del suelo en esta banda de frecuencia es dominado por las explosiones sísmicas producidas por el tráfico (carros pesados) que cruzan puentes cercanos.
- Región de alta frecuencia ( $4.0 - 15.0Hz$ ): las variaciones del sismo de tierra aquí se asocian principalmente a la actividad humana en el sitio (es decir los coches, los trabajos de construcción, etc.).

#### 4.1.1. Naturaleza de la señal

Como se mencionó en el apartado anterior las señales sísmicas obtenidas fueron descargadas en tres archivos los cuáles a su vez, contenían cinco columnas de datos; donde, la primera contenía el tiempo y las siguientes cuatro la magnitud medida de acuerdo a cada valor de tiempo de la primer columna. Siendo así, se realizó el arreglo de los datos a partir de Crimson Editor, un editor que permitió el reacomodo de los datos, para así; poder ser analizados; tomando la primer columna (respectiva al tiempo) y la segunda (que contiene la velocidad) para formar una señal (Señal1.dat), la primer columna y la tercera para formar otra señal (Señal2.dat) y así sucesivamente hasta obtener cuatro señales de cada uno de los tres archivos descargados. En la tabla siguiente se presenta la composición de dichos archivos :

Nombre del archivo.	Nombre de las señales obtenidas.
<i>EmSEBDCE0450Hzrms.dat</i>	Señal1.dat, Señal2.dat, Señal3.dat, Señal4.dat
<i>EmSEBDCE0550Hzrms.dat</i>	Señal5.dat, Señal6.dat, Señal7.dat, Señal8.dat
<i>EmSEBDCE0650Hzrms.dat</i>	Señal9.dat, Señal10.dat, Señal11.dat, Señal12.dat

Para poder realizar el análisis de las señales antes mencionadas es necesario visualizar las señales originales y así, observar las características y los resultados de las transformaciones que se van a aplicar. Mediante el Algoritmo D.1 (dado en el Apéndice D) realizado en MatLab [5] es posible graficar las señales y observar su forma original.

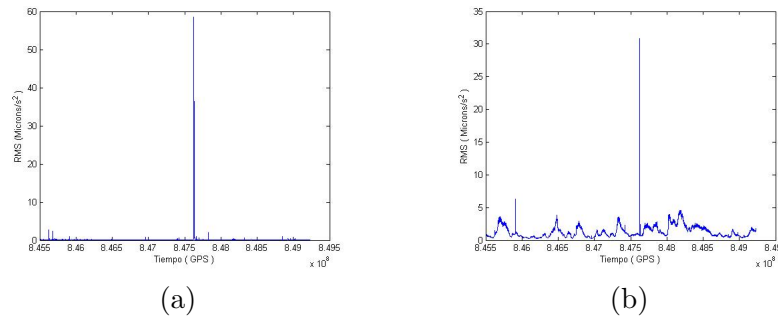


Figura 4.1: Señales originales: (a) Señal1.dat ; (b) Señal2.dat.

La Figura 4.1 muestra dos señales; la Señal1.dat (inciso a) y la Señal2.dat (inciso b). En ellas se aprecia en el eje horizontal (eje x) la magnitud del tiempo (dada en GPS) y en el eje vertical (eje y) la velocidad (expresada en RMS o  $\text{microns}/s^2$ ). Ahora si se realiza un análisis en la Señal1.dat es posible saber cuales son los rangos de tiempo y velocidad; los cuales son de:  $8.455 \times 10^8 \text{GPS} - 8.495 \times 10^8 \text{GPS}$  para el tiempo y de  $0 - 60 \text{microns}/s^2$  para la velocidad. Hecho este análisis se comprobaron las magnitudes obtenidas mediante MatLab y las que fueron mostradas por Crimson Editor al momento del reacomodo de los datos. El mismo análisis aplica en las Figuras 4.2, 4.3, 4.4, 4.5 y 4.6.

A partir de esto, se procede a aplicar las transformadas de Fourier y wavelet.

## 4.2. Transformadas de Fourier y wavelet

Las transformaciones matemáticas son aplicadas a las señales para obtener más información que aquella que se puede extraer de la señal original. Entre un gran número de transformaciones existentes una de las más conocidas es la transformada

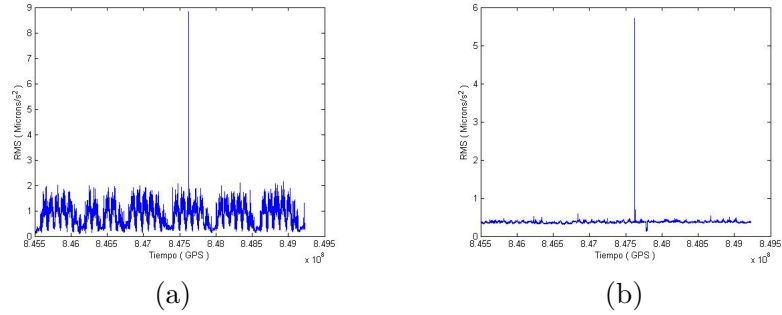


Figura 4.2: Señales originales: (a) Señal3.dat ; (b) Señal4.dat.

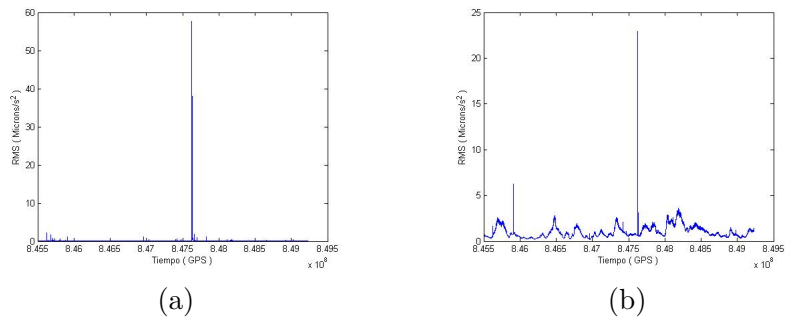


Figura 4.3: Señales originales: (a) Señal5.dat ; (b) Señal6.dat.

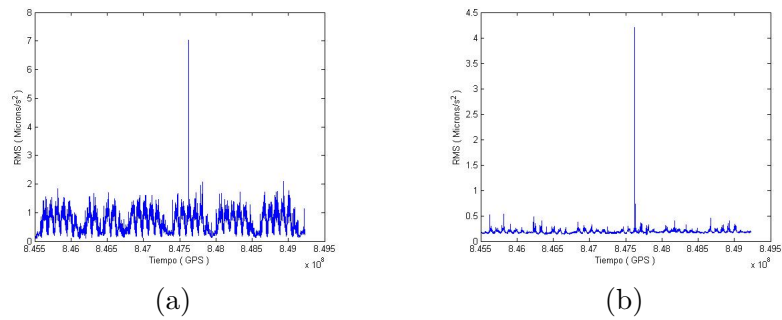


Figura 4.4: Señales originales: (a) Señal7.dat ; (b) Señal8.dat.

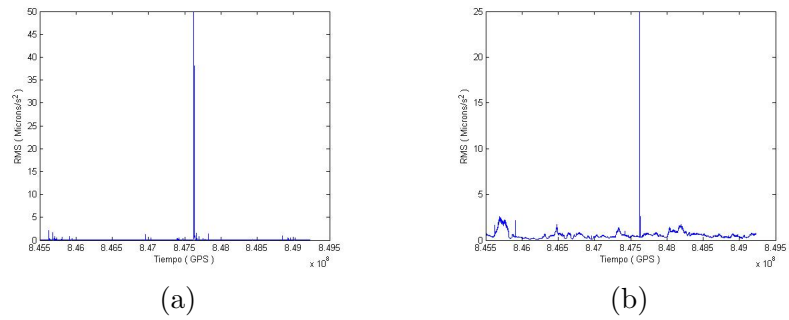


Figura 4.5: Señales originales: (a) Señal9.dat ; (b) Señal10.dat.

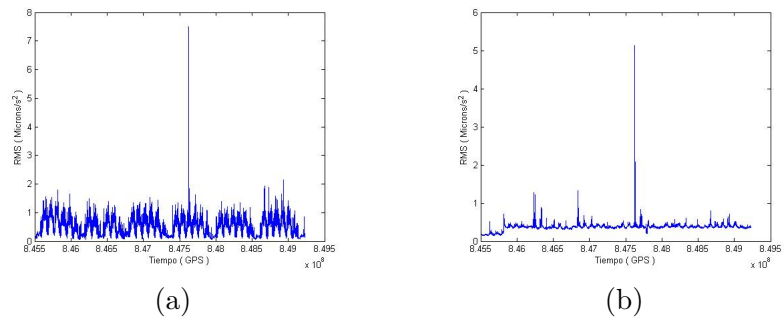


Figura 4.6: Señales originales: (a) Señal11.dat ; (b) Señal12.dat.

de Fourier. Esta transformación permite descomponer una señal en sus componentes sinusoidales de diferentes frecuencias, es decir; es una técnica matemática utilizada para transformar una señal desde el dominio del tiempo al dominio de la frecuencia. La FFT es una herramienta matemática que nos permite obtener información en el régimen de la frecuencia, tal es el caso de las señales sísmicas; donde, podemos ver que dicho método fue aplicado para conocer el rango de frecuencias de cada señal y así; saber en que puntos se encuentra la mayor cantidad de información de la señal que se está tratando. Con la ayuda de MatLab (análisis realizado por efectos de calidad de imagen) fue posible obtener las imágenes que muestran los espectros de frecuencia de las señales analizadas. El algoritmo empleado es el Algoritmo D.2 del Apéndice D. A continuación se pueden observar los resultados obtenidos de aplicar la transformada rápida de Fourier a cada una de las señales:

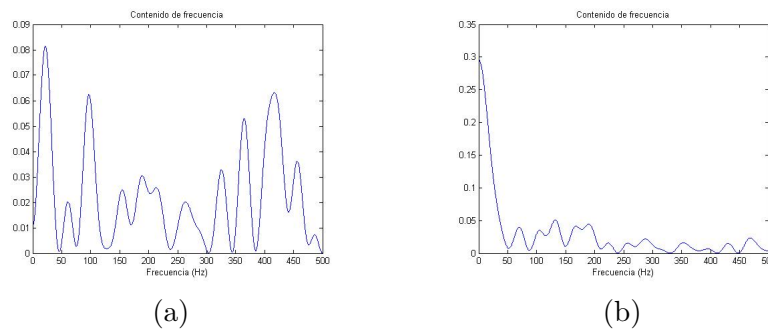


Figura 4.7: Gráficas de la transformada rápida de Fourier obtenidas a partir de MatLab: (a) Señal1.dat ; (b) Señal2.dat.

En estas figuras se ha logrado obtener la transformada rápida de Fourier que es una técnica muy empleada en el tratamiento y procesamiento de señales ya que nos permite realizar un análisis en régimen de la frecuencia, por tanto; muestran el espectro de frecuencia de cada archivo Señal.dat; particularmente se observa en las graficas de la Figura 4.7 la banda (o bandas) de frecuencia de cada señal, en (a) se distingue que la información esta contenida entre los  $0.03mHz$  a los  $500Hz$  (sobre el eje x). Así también; muestra los valores de amplitud que van desde  $0 - 0.09$  (sobre el eje y), y para (b) valores de  $0 - 0.35$  de amplitud en diferentes puntos desde  $0.03mHz$  hasta los  $500Hz$ . Siguiendo esta misma interpretación se obtienen las graficas de las señales restantes expuestas a continuación:

Cuando se observa una señal producto de una transformación de Fourier, resulta imposible determinar cuando ocurre un determinado evento o cuando está presente una determinada frecuencia, y estas características pueden resultar ser las más interesantes de la señal y la transformada de Fourier no es la adecuada para mostrarlas y/o analizarlas. En este caso el análisis mediante wavelets permite el uso de intervalos



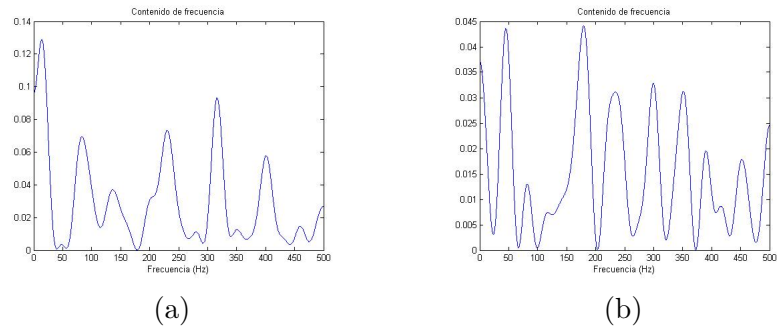


Figura 4.8: Gráficas de la transformada rápida de Fourier obtenidas a partir de MatLab: (a) Señal3.dat ; (b) Señal4.dat.

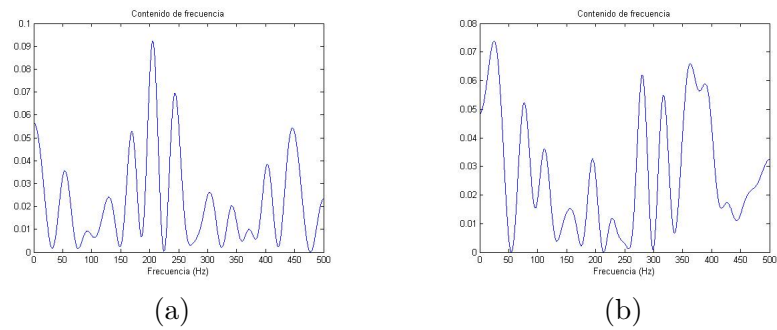


Figura 4.9: Gráficas de la transformada rápida de Fourier obtenidas a partir de MatLab: (a) Señal5.dat ; (b) Señal6.dat.

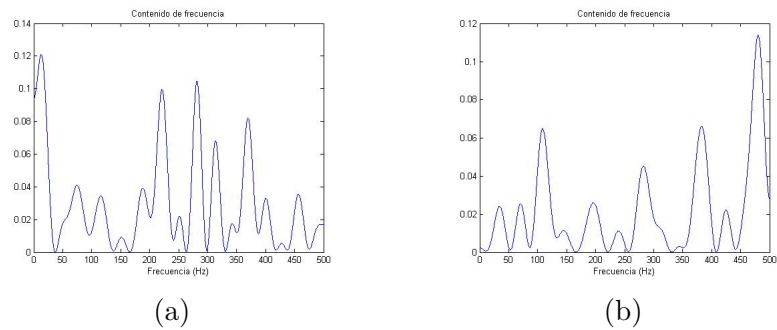


Figura 4.10: Gráficas de la transformada rápida de Fourier obtenidas a partir de MatLab: (a) Señal7.dat ; (b) Señal8.dat.

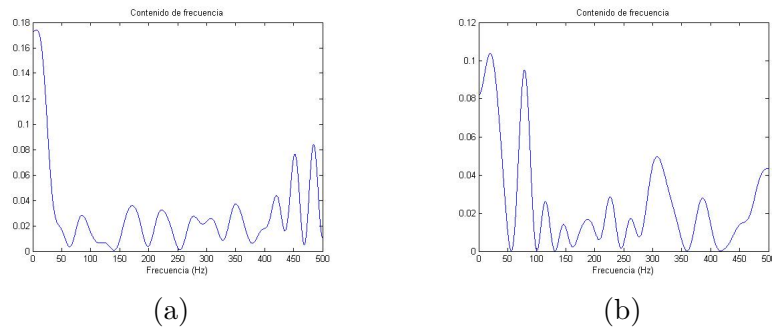


Figura 4.11: Gráficas de la transformada rápida de Fourier obtenidas a partir de MatLab: (a) Señal9.dat ; (b) Señal10.dat.

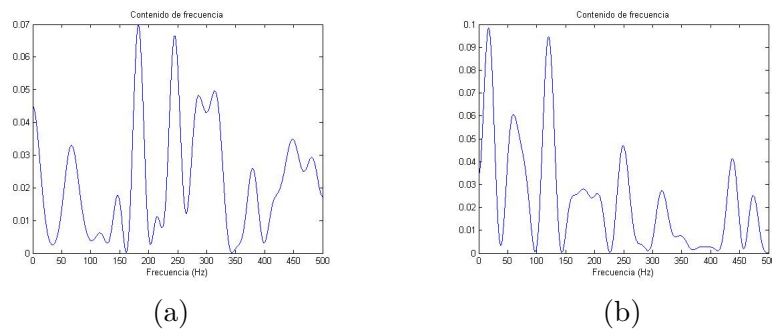


Figura 4.12: Gráficas de la transformada rápida de Fourier obtenidas a partir de MatLab: (a) Señal11.dat ; (b) Señal12.dat.

grandes de tiempo en segmentos donde se requiera mayor exactitud en baja frecuencia y regiones más pequeñas donde se requiere información en alta frecuencia. Esta nueva técnica utilizada en el estudio de señales permite mostrar datos que al investigador le permitan un análisis más preciso y mejor aún sin pérdida de referencias; que con otras técnicas no es posible realizar. La wavelet adecuada para el análisis de una señal depende de las características de la señal misma. Como siguiente punto y con la ayuda del DSP, se cargan los programas `load.asm`, `vectors.asm`, `wavtodo.h` y `wavelet.c` en dicho dispositivo; para obtener resultados respectivos a aplicar la transformada wavelet a cada una de las señales originales (`Señal1.dat`, `Señal2.dat`, `Señal3.dat...`). El procesador tiene pre-cargada una señal wavelet Daubechies 10 de cuarto nivel, lo que significa que se aplican filtros de alta y baja frecuencia que dividen la frecuencia de la señal sísmica en cuatro partes proporcionales, es decir que si la señal tiene un rango de frecuencias de  $0 - 50Hz$  se tendrán cuatro partes de la señal original:  $0 - 12.5Hz$ ,  $12.5 - 25Hz$ ,  $25 - 37.5Hz$  y  $37.5 - 50Hz$ . Conociendo lo anterior se aplica la señal wavelet a la señal sísmica original (en este caso a la señal `Señal1.dat`), obteniendo las siguientes gráficas:

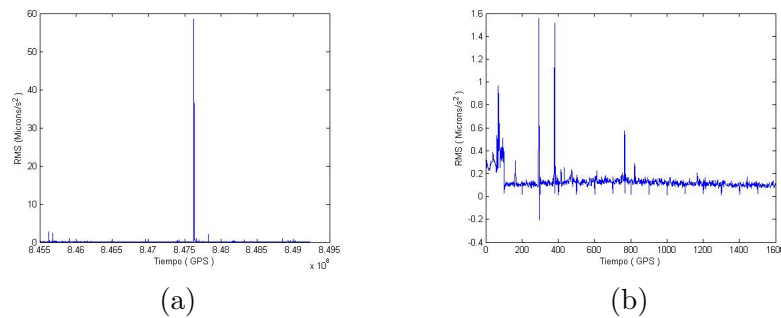


Figura 4.13: Señal obtenida a partir de la aplicación de la wavelet Daubechies(10) a un nivel de descomposición igual a 4 a la señal original `Señal1.dat`: (a) Señal original, `Señal1.dat` ; (b) `Señal1.dat` tratada con la transformación wavelet.

Una vez aplicada la transformada wavelet es visiblemente notoria la diferencia que existe entre ambas graficas; ya que es posible observar los datos de la señal con una mejor resolución y teniendo otra gama de valores que pueden ser analizados para obtener información relevante que lleve a un mejor estudio de señales de esta naturaleza o encontrar nuevas características en este tipo de señales que se presentan de manera súbita. La grafica del inciso b, muestra una señal similar a la original (inciso a); solo que en ésta se observa un eje que se denomina escala (eje horizontal, x) y otro que es el de amplitud (eje vertical, y) donde podemos decir que las bajas frecuencias (altas escalas) corresponden a la información global de una señal, mientras que las altas frecuencias (bajas escalas) corresponden a información detallada de patrones

ocultos en la señal (que usualmente tienen una duración reducida de tiempo). Fue necesario crear un archivo en el buffer de salida del DSP para graficar los resultados a partir de la transformada wavelet. A continuación se muestra el resto de señales obtenidas.

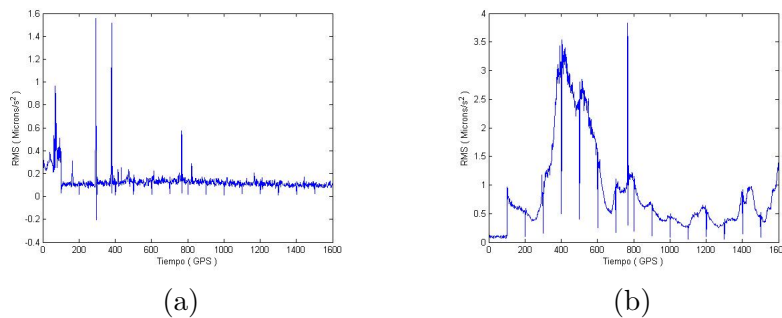


Figura 4.14: Señales sísmicas obtenidas a partir de la aplicación de la wavelet Daubechies(10) a un nivel de descomposición igual a 4: (a) Señal1.dat ; (b) Señal2.dat.

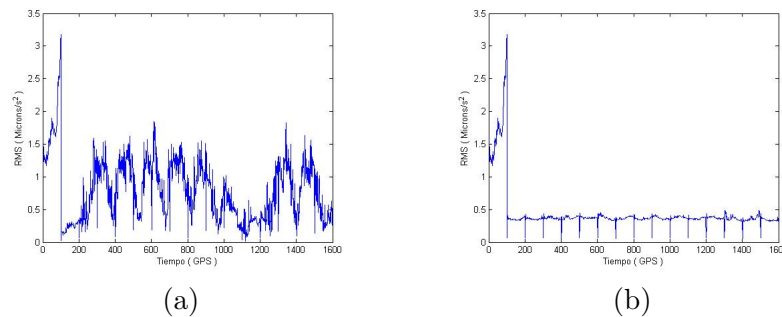


Figura 4.15: Señales sísmicas obtenidas a partir de la aplicación de la wavelet Daubechies(10) a un nivel de descomposición igual a 4: (a) Señal3.dat ; (b) Señal4.dat.

### *Resumen*

Este capítulo forma el corazón de la tesis, ya que muestra los resultados, tanto en simulación como en emulación, este último empleando un DSP. Es importante recalcar que en este trabajo de tesis, sólo se realiza el **procesamiento** de las señales sísmicas.

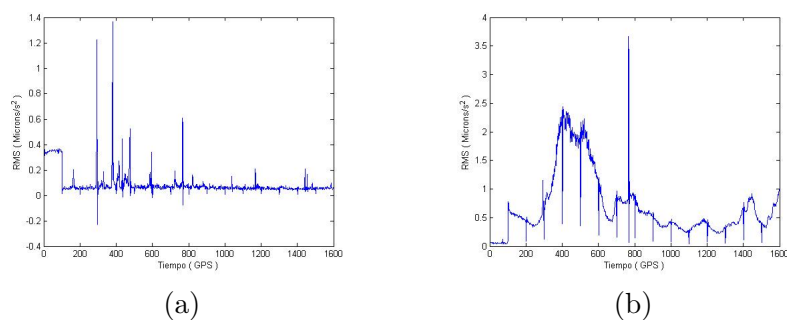


Figura 4.16: Señales sísmicas obtenidas apartir de la aplicación de la wavelet Daubechies(10) a un nivel de descomposición igual a 4: (a) Señal5.dat ; (b) Señal6.dat.

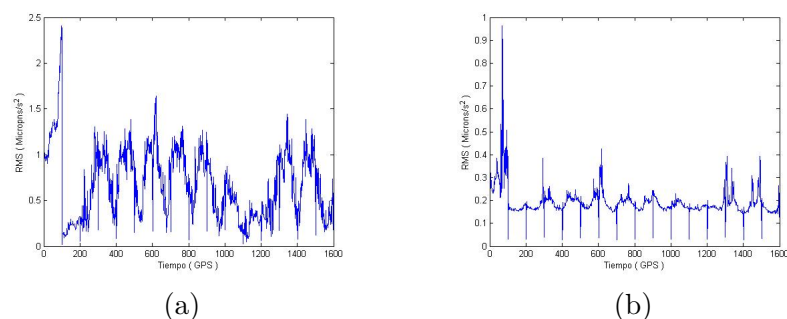


Figura 4.17: Señales sísmicas obtenidas apartir de la aplicación de la wavelet Daubechies(10) a un nivel de descomposición igual a 4: (a) Señal7.dat ; (b) Señal8.dat.

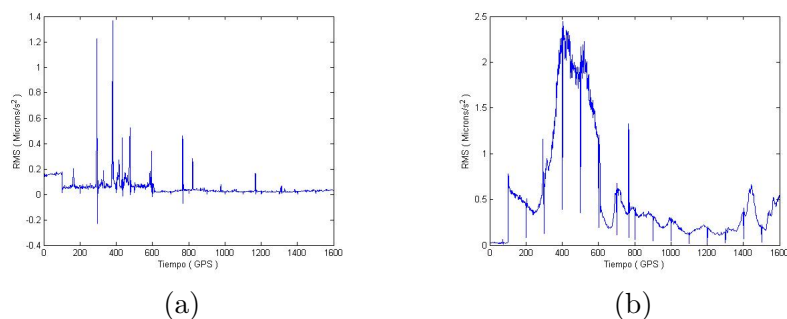


Figura 4.18: Señales sísmicas obtenidas apartir de la aplicación de la wavelet Daubechies(10) a un nivel de descomposición igual a 4: (a) Señal9.dat ; (b) Señal10.dat.

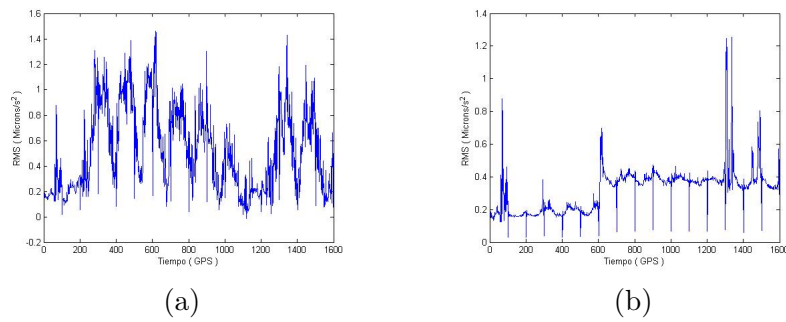


Figura 4.19: Señales sísmicas obtenidas apartir de la aplicación de la wavelet Daubechies(10) a un nivel de descomposición igual a 4: (a) Señal11.dat ; (b) Señal12.dat.

# Capítulo 5

## Conclusiones

Actualmente el procesamiento digital de señales ha tenido un fuerte auge tecnológico irrumpiendo en distintos campos científicos; el hombre, haciendo uso de todas sus capacidades y habilidades ha logrado con ello obtener resultados muy importantes y decisivos, aplicados en su vida diaria y para resolver diversos tipos de dificultades. Aunque, debido a la diversidad de campos de acción en el estudio y análisis de señales que se encuentran presentes en nuestro entorno; han manado incomparables técnicas de análisis basadas en herramientas matemáticas de análisis de señales que permiten estudiar a la señal en diferentes ámbitos como el análisis en tiempo y frecuencia (por nombrar un ejemplo).

Para llevar a cabo el análisis de una señal es necesario determinar puntos importantes que serán cruciales y se verán reflejados en los resultados obtenidos, tales como las características que se desean conocer de determinada señal. En este caso particular, el análisis de señales sísmicas juega un papel muy importante y fundamental ayudando en la investigación de sismos, que puedan ayudar en la prevención de desastres naturales provocados por sismos; mediante este estudio es posible ofrecer un panorama diferente en la comprensión de la información contenida en la señal. Es importante mencionar que en esta tesis sólo se realiza el procesamiento digital de señales de esta naturaleza, que son captadas por dispositivos especiales denominados sismógrafos; las señales fueron obtenidas de Virgo Seismic Monitor y los resultados obtenidos serán utilizados para diagnóstico en trabajos futuros de investigación. Lo novedoso en este trabajo es el uso de una nueva herramienta matemática: la transformada wavelet en conjunto con las nuevas tecnologías ofrecidas hoy en día como son los procesadores digitales de señales propiamente el *DSP3206711* de Texas Instruments.

Los resultados obtenidos al realizar el análisis de cada señal sísmica mediante la técnica wavelet, muestran características presentes en la señal que no son apreciables

a simple vista y que fueron posibles de observar debido a la descomposición realizada; con la wavelet de Daubechies 10. Los resultados obtenidos arrojan información que puede ser de utilidad a investigadores especialistas en el área de sismología, geología, etc., y serán revisados por ellos para conocer que datos e información de la señal son los necesarios para un mejor estudio y visualización de este tipo de señales.

Con esto se verificó la utilidad de la transformada wavelet como una herramienta adicional a las ya conocidas en el estudio y tratamiento de señales; es posible decir que se trata de un área relativamente reciente en la ciencia, pero que al día de hoy tiene una amplia aplicación en diversas ramas de la ciencia debido a su versatilidad, que es una de sus características principales. La perspectiva de este trabajo es global, dado que solo se realiza el *procesamiento* de la señal aplicando un algoritmo que contiene la señal wavelet aplicada.

Fue muy interesante haber podido realizar un estudio y análisis haciendo uso de tecnología de punta y de herramientas matemáticas relativamente nuevas que son utilizadas en el análisis de señales ya que con esto comprobamos cada vez mas que es necesario estar siempre al tanto del avance tecnológico, dado que ello; permite manejar y aprender de las diferentes ramas del saber.

Los resultados obtenidos al realizar el análisis de cada señal sísmica mediante la técnica wavelet, muestran características presentes en la señal que no son apreciables a simple vista y que fueron posibles de observar debido a la descomposición realizada; con la wavelet de Daubechies 10. Los resultados obtenidos arrojan información que puede ser de utilidad a investigadores especialistas en el área de sismología, geología, etc., y serán revisados por ellos para conocer que datos e información de la señal son los necesarios para un mejor estudio y visualización de este tipo de señales.

De acuerdo a nuestro conocimiento este trabajo es el primero realizado en el CITIS tratando el tema de análisis de señales sísmicas aplicando la técnica wavelet y haciendo uso del DSP.



# Glosario

$P_g$ . Onda P cuando su ángulo de partida es hacia arriba de la horizontal.

$P_r$ . Onda P críticamente refractada en alguna interfase.

$S_g$ . Onda S cuyo ángulo de partida es hacia arriba de la horizontal.

$V_p$ . Velocidad de onda P dada en km/seg.

$V_s$ . Velocidad de onda S expresada en km/seg.

**Coda**. Parte posterior de la señal sísmica.

**Discontinuidad de Mohorovicic** . Es la superficie de frontera o la pronunciada discontinuidad de la velocidad sísmica, que separa la corteza terrestre del manto superior. Esta discontinuidad fue descubierta por el sismólogo Andrija Mohorovicic, de origen Croata.

**Epicentro**. Es el punto exacto en la superficie que se localiza sobre el hipocentro de un sismo.

**FA**. Denominación dada al primer arribo.

**Manto**. Es la capa de roca que se encuentra entre la corteza y el núcleo exterior de la tierra. Tiene aproximadamente 2900 kilómetros de espesor y es la capa más grande de la tierra.

**Onda de Love**. Mayoría de ondas superficiales que tienen un movimiento horizontal y que son transversales o de corte a la dirección de propagación.

**Ondas guiadas**. Ondas que, cuando una capa o nivel de roca se encuentra rodeada de otras rocas con velocidades sísmicas superiores y algunas de las ondas que se encuentren dentro de ella no podrán escapar a los medios circundantes y serán transmitidas a lo largo de la capa con muy poca pérdida de energía.

**Ondas superficiales**. Son las ondas que se mueven sobre la superficie de la Tierra (por ejemplo, las ondas Love y Raleigh).

**Refracción.** Cambio de dirección o salto entre capas.

**Rocas graníticas.** Rocas cercanas a la superficie por las cuáles circulan las ondas compresionales.

**SH.** Componente horizontal de onda S denotada por SH.

**Sismograma.** Grabación escrita de un terremoto que se hace en un sismógrafo.

**SV.** Componente vertical de onda S denotada por SV.

**Telesismo.** Es un terremoto muy distante (más de 20 grados) desde la estación donde se registra.

# Bibliografía

- [1] <http://omega.ilce.edu.mx:3000/sites/ciencia/volumen1/ciencia2/34/html/terrem.html>. Ondas sísmicas.
- [2] <http://wwwcascina.virgo.infn.it/commissioning/monitoring-50hz/>. Página de la base de datos de donde fueron extraídas las señales sísmicas.
- [3] <http://www.monografias.com/trabajos17/procesamiento-digital/procesamiento-digital.shtml?monosearch>. Arquitectura estándar de un DSP.
- [4] <http://www.virgo.infn.it/>. Dirección electrónica de donde fueron obtenidas las señales sísmicas.
- [5] Matlab v 7.0. Ambiente utilizado para la obtención del rango de frecuencias, visualización de las señales sísmicas originales y de los resultados a partir de la transformada wavelet.
- [6] Software crimson editor v 3.7. Editor utilizado para la descomposición de las señales sísmicas.
- [7] Revista de geografía universal. 20(5), 1985.
- [8] K. Bullen. *An Introduction to the Theory of Seismology*. Cambridge Univ. Press, G. B., third edition, 1963.
- [9] A.P. Calderon. *Intermediate spaces and interpolation, the complex method*. Studia Math, 1964.
- [10] C.K. Chui. *An Introduction to Wavelets*. Academic Press Inc, Boston, MA, 1992.
- [11] I. Cohen, A.I. Daubechies and J.C. Feauveau. *Bi-orthogonal bases of compactly supported wavelets*. Comm. Pure and Applied Math., third edition, 1992.
- [12] L. Cohen. *Time-frequency distributions, a review*. Proc. IEEE., 1989.

- 
- [13] A. Croisier and D. Esteban. *Perfect channel splitting by use of interpolation, decimation, tree decomposition techniques*. Int. Conf. on information sciences and systems., 1976.
- [14] I. Daubechies. *Orthonormal bases of compactly supported Wavelets*. Commun on pure and appl. math., 1988.
- [15] D. Esteban and C. Galand. *Application of quadrature mirror filters to split-band voice coding systems.* ., 1977.
- [16] Harvey Gerber. *Álgebra lineal*. Gpo. Editorial Iberoamérica.
- [17] F. Gilbert. *Excitation of normal modes of the Earth by earthquake sources*. Geophys. J. R. astr. Soc.
- [18] R.A. Gopinath and C.S. Burrus. *Wavelet transforms and filter banks, A tutorial in theory and applications*. Academic Press., 1992.
- [19] A. Grossman and J. Morlet. *Descomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape*. SIAM J. Appl. Math., 1984.
- [20] Hwei Hsu. *Análisis de Fourier*. Addison-Wesley Iberoamericana., 1987.
- [21] W.M. Lawton. *Tight frames of compactly supported wavelets*. J. Math. Phys., 1990.
- [22] W.M. Lawton. *Necessary and sufficient conditions for existence of on wavelet bases*. J. Math. Phys., 1991.
- [23] S. Mallat. *A theory of multiresolution signal descomposition:The Wavelet representation*. IEEE Trans., 1989.
- [24] Y. Meyer. *Wavelets: Algorithms and applications*. Society for Industrial and Applied Mathematics., 1993.
- [25] Alan V. Oppenheim. *Señales y sistemas*. Prentice Hall, 1994.
- [26] Manolakis. Proakis. *Tratamiento digital de señales*. Prentice Hall, 1998.
- [27] Inc. Texas Instruments. *TMS320C6711 User's Guide*. Dallas, TX, 2001.
- [28] Texas Instruments, U.S.A. *Code Composer Studio*, Noviembre 2001. Ambiente de emulación del DSP.
- [29] A. Udías. *Introducción a la sismología y estructura interna de la Tierra*. Taller del I.G. y C. Madrid, España.

- [30] M. Vetterli and C. Herley. *Wavelets and filter banks: Theory and design*. IEEE Trans., third edition, 1992.
- [31] J.W. Woods and S.D. Ot'Neill. *Subband coding of images*. IEEE Trans., 1986.

# Apéndice A

## Transformadas

Las transformaciones matemáticas son aplicadas a señales para obtener información que no se encuentra disponible en la señal natural (dominio del tiempo). La mayoría de las veces, la información no puede ser fácilmente visible en el dominio del tiempo, entonces un método alternativo es verla en el dominio de la frecuencia. Las herramientas de análisis que se resumen en este capítulo, se aplican habitualmente a una gama muy amplia de señales que se encuentran en la práctica (por ejemplo, señales sísmicas, biológicas, electromagnéticas, etc.).

En general, el análisis en la frecuencia se realiza con el propósito de extraer información de la señal observada. Por ejemplo, en el caso de la señal de un sismógrafo, una señal sísmica. Las herramientas analíticas se utilizan para extraer información relevante para el tratamiento e interpretación de esta información.

### A.1. Transformada Z

La *Transformada Z* es utilizada para el análisis de señales discretas en el tiempo, es similar a la *Transformada de Laplace* para señales de tiempo continuo. Nosotros podemos usar la transformada de Laplace para resolver ecuaciones diferenciales que representan un filtro analógico, o la transformada Z para resolver una ecuación diferencial que representa un filtro digital. Considerando una señal analógica  $x(t)$  idealmente muestreada:

$$x_s(t) = \sum x(t)\delta(t - kT) \tag{A.1}$$

donde  $\delta(t - kT)$  es la función impulso (delta) retrasada por  $kT$ , y  $T = 1/F_s$  es el periodo de muestreo. La función  $x_s(t)$  es cero en cualquier punto excepto en  $t = kT$ . La transformada de Laplace de  $x_s(t)$  es:

$$X_s(s) = \int x_s(t)e^{-st}dt = \int x(t)\delta(t) + x(t)\delta(t - T) + \dots e^{-st}dt \quad (\text{A.2})$$

Donde la propiedad de la función impulso

$$X_s(s) = \int f(t)\delta(t - kT)dt = f(kT) \quad (\text{A.3})$$

$X_s(S)$  en ec. A.2 llega a ser:

$$X_s(S) = x(0) + x(T)e^{-sT} + x(2T)e^{-2sT} + \dots = \sum x(nT)e^{-nsT} \quad (\text{A.4})$$

Permitiendo  $z = e^{sT}$  en ec. A.3 que llegue a ser:

$$X(z) = \sum x(nT)z^{-n} \quad (\text{A.5})$$

Dejando implícito el periodo de muestreo  $T$ , después  $x(nT)$  puede ser escrito como  $x(n)$ :

$$X(z) = \sum x(n)z^{-n} = ZTx(n) \quad (\text{A.6})$$

que representa la transformada ( $ZT$ ) de  $x(n)$ . Esto es una correspondencia uno a uno entre  $x(n)$  y  $X(z)$ , haciendo la transformada  $Z$  una única transformación.

**Ejemplo** Determinar la Transformada Z de la función escalón unitario  $f(t)$ :

$$f(t) = \{1, t \geq 0; 0, t < 0\} \quad (\text{A.7})$$

Como se puede observar la función  $f(t)$  es continua en el tiempo, y debido a que la transformada Z es aplicable a funciones discretas obtenemos lo siguiente:

$$F(z) = Zf(t) = \sum f(kT)z^{-k} \quad (\text{A.8})$$

donde T representa el periodo de muestreo aplicado a la función continua en el tiempo. De la aplicación de la ecuación anterior a la función  $f(t)$  resulta:

$$F(z) = \sum f(kT)z^{-k} = \sum z^{-k} = 1 + z^{-1} + z^{-2} + \dots + z^{-k} \quad (\text{A.9})$$

Factorizando  $z^{-1}$  obtenemos:

$$F(z) = 1 + z^{-1} \underbrace{[1 + z^{-1} + z^{-2} + \dots + z^{-k} + \dots]} = 1 + z^{-1}F(z) \quad (\text{A.10})$$

Despejando para determinar  $F(z)$

$$F(z) - z^{-1}F(z) = 1(1 - z^{-1})F(z) = 1 \quad (\text{A.11})$$

Finalmente:

$$F(z) = \frac{1}{1 - z^{-1}} \frac{z}{z} = \frac{z}{z - 1} \quad (\text{A.12})$$



## A.2. Relación entre el plano Z y el plano S

Se sabe que:

$$f(t) = \sum f(kT)\delta(t - kT) \quad (\text{A.13})$$

y

$$F(s) = Lf(kT) = \sum f(kT)e^{-kTs} \quad (\text{A.14})$$

Resultando:

$$Z = e^{Ts} \quad (\text{A.15})$$

De la ecuación anterior se concluye que la expresión  $e^{Ts}$  representa en el plano de Laplace un retardo de unidad, el cual tiene su proyección dual en el plano discreto o plano Z mediante el operador Z, quedando de esta manera demostrado que el operador Z realiza la misma función que el operador S, solo que en el plano discreto.

## A.3. La transformada corta de Fourier (STFT)

Como ya es sabido, la transformada de Fourier constituye una herramienta mediante la cual podemos obtener información sobre como está distribuida la energía de una señal a través de sus distintas componentes de frecuencia, es decir, podemos conocer todas las componentes de frecuencia existentes en la señal y sus respectivos aportes energéticos. Todo lo anterior se puede resumir diciendo que la transformada de Fourier tiene una perfecta resolución en frecuencia lo que la hace una herramienta muy útil para el análisis de señales estacionarias. Sin embargo, ella no puede ser aplicada con el objeto de obtener información precisa de cuando o donde las diferentes componentes de frecuencia se encuentran en la señal como es el caso de señales estacionarias o no estacionarias cuyo contenido espectral varía con el tiempo. En otras

palabras, la transformada de Fourier posee una muy pobre resolución en tiempo. En un esfuerzo por resolver el problema de resolución en tiempo de la transformada de Fourier, Denis Gabor (1946) adaptó la transformada utilizando un procedimiento llamado ventanamiento. Este procedimiento consiste en dividir una señal  $x(t)$  en pequeños segmentos a través del tiempo de tal manera que podamos asumir que para cada segmento la señal es estacionaria y así calcular la Transformada de Fourier clásica para cada porción de la señal.

La forma de dividir la señal se realiza mediante lo que llamaremos una función tiempo-ventana  $h(t)$  cuyo ancho o soporte corresponde a la longitud de cada segmentación de la señal. Con la función ventana encuadramos la señal alrededor de un instante de tiempo  $\tau$  y calculamos su transformada de Fourier, luego trasladamos la función ventana hasta que no se sobrepone con la anterior cubriendo una nueva porción de la señal a la que volvemos a calcular su transformada de Fourier. Este proceso es repetido hasta que se ha cubierto la totalidad de la señal.

<sup>1</sup>

El resultado de lo expresado anteriormente se define en forma matemática de la siguiente manera:

$$STFT(t, \omega) = \int_{-1}^1 x(t)h * (\tau - t)e^{-i\omega t} dt \quad (\text{A.16})$$

y si consideramos a  $h(t)$  como una función ventana de valores sólo reales no complejos de tal manera que  $h(-t) = h^*(t)$  entonces nos queda:

$$STFT(t, x) = \int_{-1}^1 x(t)h(t - \tau)e^{-it} dt \quad (\text{A.17})$$

que calcula el producto interno entre la señal y la función tiempo-ventana trasladada y modulada. De acuerdo con las propiedades de la transformada de Fourier de translación en tiempo y frecuencia:

$$h(t - t_0) \Leftrightarrow H(\omega)e^{i\omega t_0}h(t)e - i\omega_0 t \Leftrightarrow H(\omega - \omega_0) \quad (\text{A.18})$$

y utilizando el teorema de Parseval podemos expresar también la ( STFT ) en términos de la transformada de Fourier de la señal y la transformada de Fourier de la función tiempo ventana, lo que nos da:

$$STFT(t, xi) = \frac{e^{-ib}}{2\pi} \int_{-\infty}^{\infty} X(t)H(\omega - xi)e^{i\tau\omega} dt \quad (\text{A.19})$$

<sup>1</sup>Short Time Fourier Transform Traducido directamente del inglés windowing

de tal manera que mientras la función tiempo-ventana modulada  $e^{it}h(t-\tau)$  en A.2 localiza la señal  $x(t)$  cerca de  $t = \tau$ , la correspondiente función frecuencia-ventana:

$$\frac{e^{ib}}{2\pi}H(\omega - xi)e^{-i\tau\omega} \quad (\text{A.20})$$

localiza el espectro  $X(\omega)$  de la señal en la vecindad de  $\omega = xi$  limitada por el soporte de la función frecuencia-ventana  $H(\omega)$ .

### A.3.1. Resolución tiempo - frecuencia.

Ahora bien, el soporte de la ventana constituye un parámetro de gran importancia ya que a través de este podemos establecer el grado de resolución tanto de tiempo como de frecuencia que deseemos. Si nuestra ventana es muy angosta analizaremos una porción muy pequeña de la señal lo que nos permite tener una buena resolución en tiempo pero una mala resolución en frecuencia ya que conoceremos sólo una mínima fracción del espectro total existente en la señal. Por otro lado, si nuestra ventana es muy ancha tendremos una buena resolución en frecuencia pero una mala resolución en tiempo, una ventana de ancho infinito es nada más y nada menos que la transformada de Fourier clásica.

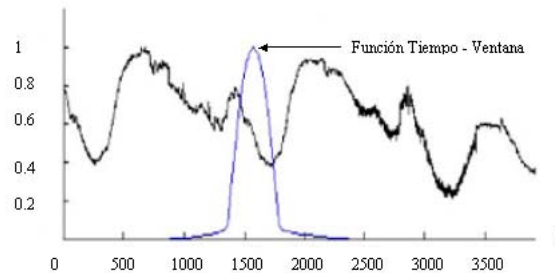


Figura A.1: Señal  $x(t)$  y función tiempo-ventana centrada en  $\tau$ .

Por lo tanto un defecto de la STFT es que no puede entregar una buena resolución tanto en tiempo como en frecuencia de manera instantánea ya que el soporte de la ventana es fijo. La raíz de este problema se basa en el principio de incertidumbre de Heisenberg el cual establece que es imposible conocer una representación exacta tiempo - frecuencia de una señal, es decir, no podemos saber que valor de frecuencia existe en un instante de tiempo determinado, sólo podemos conocer que componentes de frecuencia existen dentro de un intervalo de tiempo determinado. Con el fin de dejar más claro el concepto de resolución tiempo frecuencia utilizaremos un ejemplo. Primero supongamos que tenemos una señal  $x(t)$  dentro de un intervalo de tiempo igual a una décima de segundo. Esta señal está compuesta por sólo dos frecuencias,

una correspondiente a 250 Hz y la otra correspondiente a:

$$500\text{Hz}x(t) = 0.8 * \text{sen}(2\pi 250t) + \text{sen}(2\pi 500t) \quad (\text{A.21})$$

Como podemos observar en la figura anterior la transformada de Fourier nos entrega una resolución perfecta en frecuencia de dicha señal, como era de esperarse. Ahora, supongamos que tenemos otra señal  $x_1(t)$  con las mismas componentes de frecuencia sobre el mismo intervalo de tiempo, pero con la diferencia que las primeras 5 centésimas de segundo contienen a la frecuencia de 250 Hz y las otras 5 centésimas de segundo restante contienen a la frecuencia de 500 Hz;

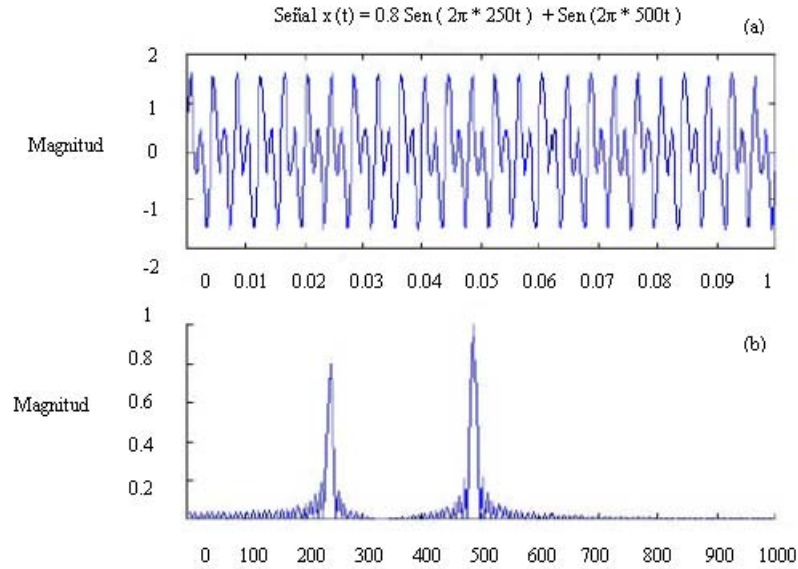


Figura A.2: (a) Representación de la señal  $x(t)$ ; (b) Contenido espectral de la señal obtenido mediante la transformada rápida de Fourier.

Si aplicamos la Transformada de Fourier sobre  $x_1(t)$  observamos que también podemos obtener las frecuencias existentes de la señal pero con una amplitud igual a la mitad de la amplitud real debido a que cada componente de frecuencia se encuentra sólo la mitad del tiempo de análisis de la señal como se ilustra en la Figura A.3.

Esta información errónea se debe a que la transformada de Fourier, como se expresó en un principio, no puede determinar en que momento dentro de la señal se encuentra una respectiva componente de frecuencia. Debido a esto vamos a volver a analizar la señal de A.5 con la transformada corta de Fourier (STFT), y la función tiempo - ventana utilizada será una función gaussiana simple de la forma:

$$h(t - \tau) = e^{-\frac{\pi}{a^2}(t-\tau)^2} \quad (\text{A.22})$$

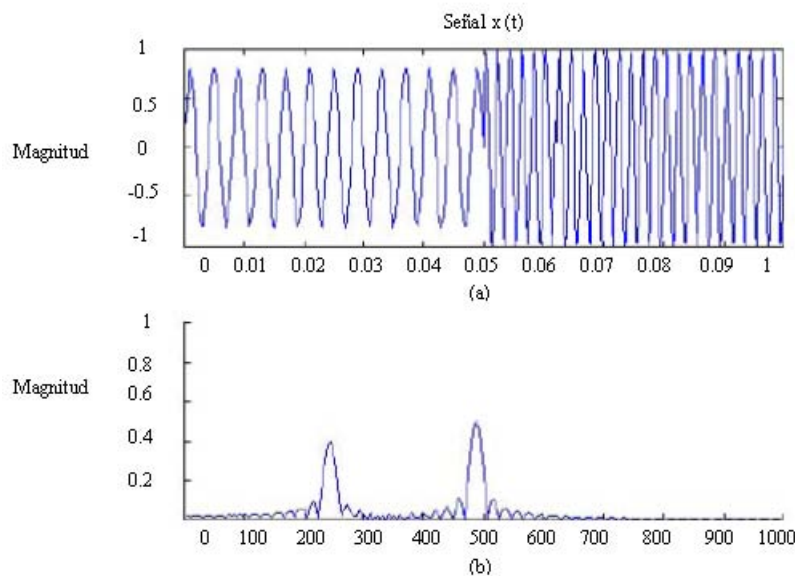


Figura A.3: (a) Representación de la señal  $x_1(t)$ ; (b) Contenido espectral de la señal obtenido mediante la FFT, se observa que las amplitudes de ambas componentes han disminuido a la mitad de su valor real.

donde  $a$  es el factor que controla el ancho o soporte de  $h$ . El primer análisis para un valor de  $a = 20$  se ilustra en la Figura A.4. Al ser angosta la ventana utilizada podemos observar que la resolución en el tiempo es buena ya que se diferencia claramente la posición en el tiempo de cada componente de frecuencia. Sin embargo, la resolución en frecuencia es bastante pobre ya que para cada componente se observa un ancho de banda amplio lo que impide una detección precisa del valor real de la frecuencia existente en el intervalo de tiempo donde se encuentra. El segundo análisis se efectúa para un valor de  $a = 250$  y se ilustra en la Figura A.5. Este aumento de  $a$  significa que nuestra función tiempo - ventana es más ancha y por lo tanto hemos mejorado nuestra resolución en frecuencia ya que el ancho de banda de cada componente ha disminuido permitiéndonos identificar claramente cada frecuencia. Por otro lado la resolución en tiempo se ha empobrecido, producto de la mejora en la resolución en frecuencia ya que no se observa una clara separación de la ubicación de cada componente en su respectivo intervalo de tiempo. Mediante este ejemplo se ha podido demostrar el problema implícito de resolución de la STFT lo que crea la interrogante ¿Es posible que la función ventana tenga un soporte dinámico y no estático?, es decir, una función ventana que tenga la capacidad de cambiar su soporte en forma automática dependiendo del contenido espectral del segmento de la señal analizado, ya que una situación ideal de análisis sería tener una buena resolución en tiempo para frecuencias altas y una buena resolución en frecuencia frente a contenido de frecuen-

cias bajas. Para la resolución de este problema existe una herramienta matemática denominada la *transformada continua wavelet* (Fig. A.6 que fue desarrollada como una alternativa de análisis frente a la STFT, el desarrollo teórico y práctico de esta herramienta constituye el foco principal de la siguiente.

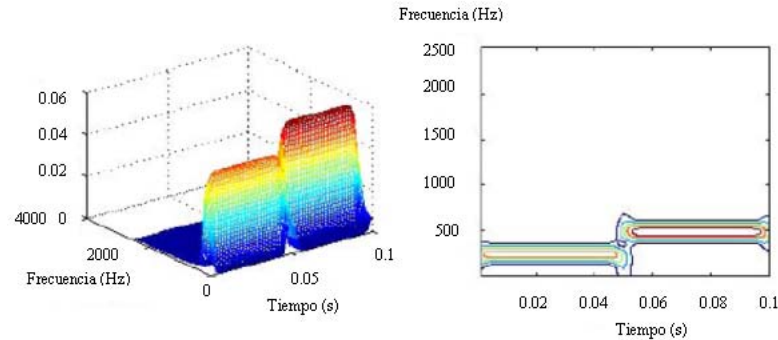


Figura A.4: Representación tiempo - frecuencia con buena resolución en tiempo y mala resolución en frecuencia.

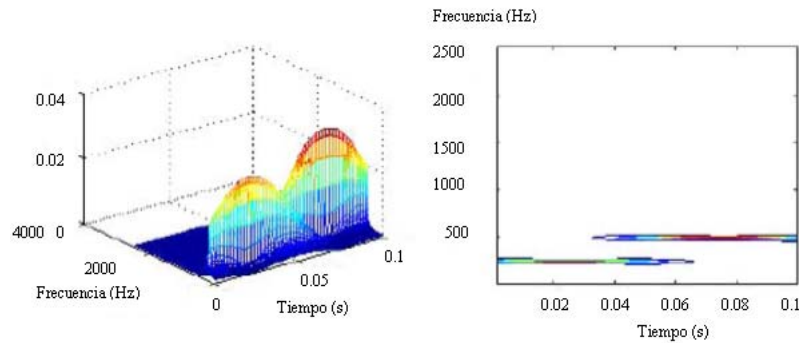


Figura A.5: Representación tiempo - frecuencia con buena resolución en tiempo y mala resolución en frecuencia.

## A.4. Transformada rápida de Fourier (FFT)

La Transformada Rápida de Fourier (Fast Fourier Transform) (FFT) es un algoritmo eficiente  $O(N \lg N)$  para calcular la DFT originalmente descubierta por Gauss a principios de 1800; redescubierta por Cooley y Tukey en IBM durante 1960; C.S. Burrus, de la Universidad de Rice University siendo jefe del departamento de Ingeniería, literalmente escribió el libro de los algoritmos de la rápida Transformada Discreta de Fourier DFT.

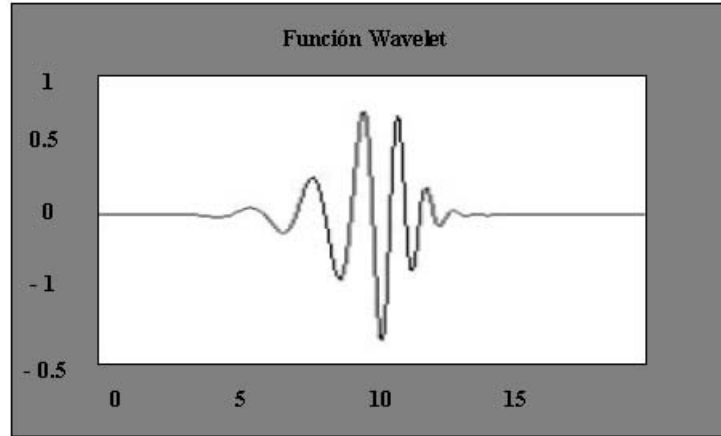


Figura A.6: Función wavelet correspondiente a la familia Daubechies.

La *Transformada Discreta de Fourier* (DFT) es de gran relevancia en aplicaciones de procesamiento digital de señales, como el análisis espectral. La DFT cuenta con algoritmos eficientes para el cálculo en dispositivos digitales.

Su metodología se basa en el "divide y vencerás"; así, una DFT de tamaño  $n$ , donde  $N$  es un número compuesto, se reduce al cálculo de DFTs más pequeñas a partir de las que se obtiene la DFT total. A partir de esta metodología han surgido un gran número de algoritmos para el cálculo de la DFT, que son los algoritmos para la Transformada Rápida de Fourier (FFT Fast Fourier Transform).

El problema dentro del cálculo de la DFT radica en determinar la secuencia  $F(k)$  de  $N$  números complejos dada la secuencia de datos  $f(n)$  de longitud  $N$  según la fórmula:

$$F(k) = \sum f(n)W_N^{kn} \quad (\text{A.23})$$

$$0 \leq k \leq N - 1$$

donde:

$$W_N = e^{-j2\pi/N} \quad (\text{A.24})$$

y la secuencia de  $f(n)$  es compleja.

Es posible ver que para cada valor de  $k$ , en el cálculo de  $F(k)$  se deben realizar  $N$  multiplicaciones complejas ( $4N$  multiplicaciones reales) y  $N - 1$  sumas complejas ( $4N - 2$  sumas reales). Por lo que para determinar los  $N$  valores de la DFT se requieren  $N^2$  multiplicaciones complejas y  $N^2 - N$  sumas complejas.

El cálculo directo de la DFT es ineficiente debido, a que no hace uso de las propiedades de simetría y periodicidad del factor de fase  $W_N$ , las cuáles son:

- Propiedad de simetría :  $W_N^{k+N/2} = -W_N^k$
- Propiedad de periodicidad :  $W_N^{k+N} = W_N^k$

En cambio los algoritmos computacionales eficientes, conocidos como algoritmos para la Transformada Rápida de Fourier, usan estas dos propiedades del factor de fase.[20]

#### A.4.1. Cálculo directo de la DFT

Para una secuencia  $f(n)$  de  $N$  puntos, la DFT se puede representar como:

$$F_{Re}(k) = \sum [f_{Re}(n) \cos \frac{2\pi kn}{N} + f_{Im}n \sen \frac{2\pi kn}{N}] \quad (\text{A.25})$$

$$F_{Re}(k) = - \sum [f_{Re}(n) \sen \frac{2\pi kn}{N} - f_{Im}n \cos \frac{2\pi kn}{N}] \quad (\text{A.26})$$

El calculo directo de las ecuaciones anteriores exige:

1.  $2N^2$  cálculos de funciones trigonométricas.
2.  $4N^2$  multiplicaciones reales.
3.  $4N(N - 1)$  sumas reales.
4. Numerosas operaciones de direccionamiento e indexado.



### A.4.2. Metodología para el cálculo de la DFT (“Divide y vencerás”)

El procedimiento principal de este método es la descomposición de una DFT de  $N$  puntos en DFTs más pequeñas. Es decir, considerando una DFT de  $N$  puntos, la idea es representar  $N$  como un producto de enteros de la forma:

$$N = LM \tag{A.27}$$

Por medio de esta notación la secuencia  $f(n), 0 \leq n \leq N - 1$ , ahora se puede representar mediante una matriz unidimensional indexada por  $n$ , o en una matriz bidimensional indexada por  $l$  y  $m$ , donde,  $0 \leq l \leq L - 1$  y  $0 \leq m \leq M - 1$ , por lo tanto  $f(n)$  se puede almacenar en una matriz rectangular de diferentes maneras, cada una depende de la correspondencia entre el índice  $n$  y los índices  $l$  y  $m$ . Así para la correspondencia:

$$n = l + mL \tag{A.28}$$

la disposición es que la primera columna contiene los primeros  $L$  elementos de  $f(n)$ , La segunda contiene los siguientes  $L$  elementos de  $f(n)$ , y así sucesivamente.

Una disposición similar puede usarse para almacenar los valores calculados de la DFT. Si se tiene la correspondencia del índice  $k$  a la pareja de índices  $(p, q)$ , donde  $0 \leq p \leq L - 1$  y  $0 \leq q \leq M - 1$ , de tal manera que:

$$k = Mp + q \tag{A.29}$$

la DFT se almacena por filas, donde la primera fila contiene los primeros  $M$  elementos de la DFT  $F(k)$ , la segunda fila los siguientes  $M$  elementos, y así sucesivamente. Actualmente la tecnología digital permite registrar todas las señales sísmicas de interés en un mismo medio y en un solo flujo de datos.[26]

# Apéndice B

## Álgebra lineal

### B.1. Espacios vectoriales de dimensión finita

Un espacio vectorial es un objeto compuesto que consta de un campo  $K$  y de un conjunto de *vectores*  $V$ , con dos operaciones binarias  $(+)$  y  $(\bullet)$ , que satisfacen ciertas propiedades especiales. El mismo conjunto de vectores puede ser parte de distintos espacios vectoriales. El nombre *vector* se da a los elementos del conjunto  $V$ , por conveniencia, pues existe una gran variedad de objetos que pueden ser vectores y que no se asemejan mucho al concepto que se tiene de vector. El origen de este nombre proviene del Ejemplo 1. En este apéndice se definirán algunos objetos concernientes a espacios vectoriales.

**Definición B.1 Espacio vectorial.** Un espacio vectorial consta de lo siguiente:

1. un campo  $K$  de escalares;
2. un conjunto  $V$  de objetos llamados vectores;
3. una operación binaria  $+ : V \times V \rightarrow V$  tal que  $(x, y) \mapsto x + y$ , satisface:
  - (a)  $x + y = y + x$
  - (b)  $x + (y + z) = (x + y) + z$
  - (c) existe un único vector  $0 \in V$ , llamado vector nulo, tal que  $x + 0 = 0 + x = x, \forall x \in V$
  - (d) para cada vector  $x \in V$ , existe un único vector  $-x \in V$ , tal que  $x + (-x) = 0$
4. una operación externa  $(\bullet) : K \times V \rightarrow V$  tal que  $(\alpha, x) \mapsto \alpha x$ , llamada multiplicación escalar, que satisface:

- (i)  $1x = x, \forall x \in V$
- (ii)  $(\alpha\beta)x = \alpha(\beta x)$  (iii)  $\alpha(x + y) = \alpha x + \alpha y$
- (iv)  $(\alpha + \beta)x = \alpha x + \beta x$ .

**Ejemplo 1. El espacio de n-tuplas,  $K^n$**  Sea  $K$  cualquier campo, y sea  $V$  el conjunto de todos los  $n$ -tuplas  $x = (x_1, x_2, x_3, \dots, x_n)$  de escalares  $x_i \in K$ . Si  $y = (y_1, y_2, y_3, \dots, y_n)$  con  $y_i \in K$ , la suma de  $x$  e  $y$  se define por  $x + y = (x_1 + y_1, x_2 + y_2, \dots, x_n + y_n)$ . El producto de un escalar  $c$  y el vector  $x$  se define por  $cx = (cx_1, cx_2, \dots, cx_n)$ . Usando las propiedades de la adición y multiplicación escalar de los elementos de  $K$  se prueban fácilmente las propiedades de espacio vectorial.

**Ejemplo 2. El espacio de matrices  $m \times n, K^{m \times n}$ .** Sea  $K$  cualquier cuerpo, y sean  $m$  y  $n$  enteros positivos. Sea  $F^{m \times n}$  el conjunto de todas las matrices  $m \times n$  sobre el campo  $K$ . La suma de dos vectores  $A$  y  $B$  en  $F^{m \times n}$  se define por  $(A+B)_{ij} = A_{ij} + B_{ij}$ . El producto de un escalar  $c$  y del vector  $A$  se define por  $(cA)_{ij} = cA_{ij}$ .

**Ejemplo 3. El espacio de funciones de un conjunto en un campo.** Sea  $K$  cualquier campo, y sea  $S$  cualquier conjunto no vacío. Sea  $V$  el conjunto de todas las funciones  $f : S \rightarrow K$ . La suma de dos vectores  $f$  y  $g$  de  $V$  es el vector  $f+g : S \rightarrow K$  definida por  $(f+g)(s) = f(s) + g(s)$ . El producto del escalar  $c$  y el vector  $f$  es la función  $cf$  definida por  $(cf)(s) = cf(s)$ .

**Definición B.2 Combinación lineal.** Un vector  $v \in V$  se dice combinación lineal de los vectores  $v_1, \dots, v_n \in V$ , si existen escalares  $c_1, \dots, c_n \in K$  tales que  $v = c_1v_1 + \dots + c_nv_n = \sum_{i=1}^n c_iv_i$ .

**Definición B.3 Subespacio de un espacio vectorial.** Sea  $V$  un espacio vectorial sobre el campo  $K$ . Un subespacio de  $V$  es un subconjunto  $W \subset V$  que con las operaciones de adición vectorial y multiplicación escalar sobre  $V$ , es él mismo un espacio vectorial sobre  $K$ .

**Teorema 1.**  $W \subset V, W \neq 0$ ; es un subespacio de  $V \Leftrightarrow \forall x; y \in W$  y  $\forall c \in K$ , el vector  $cx + y \in W$ . A veces se prefiere usar la propiedad  $cx + y$  como definición de un subespacio, lo que es sólo diferente. Lo importante es que si  $W$  contiene todos los  $cx + y$ , entonces será un espacio vectorial (con las propiedades heredadas de  $V$ ).

**Ejemplo 4.** El espacio de las funciones polinomios sobre el campo  $K$  es un subespacio del espacio de todas las funciones de  $F$  en  $F$ .

**Ejemplo 5.** Una matriz cuadrada  $n \times n$ , sobre el campo  $K$  es simétrica si  $A_{ij} = A_{ji}$  para todo  $i, j$ . Las matrices simétricas forman un subespacio del espacio de las matrices  $n \times n$  sobre  $K$ .

**Ejemplo 6.** Una matriz  $A$ , cuadrada  $n \times n$ , sobre el campo  $C$  de los números complejos es hermítica (o autoadjunta) si  $A_{jk} = \overline{A_{kj}}$  para todo  $j, k$ , donde el super-rayado indica conjugación compleja. Una matriz  $2 \times 2$  es hermítica si, y sólo si, tiene la forma:

$$z&x + iy&x - iy&w \quad (\text{B.1})$$

donde  $x, y, z, w$ , son números reales. El conjunto de todas las matrices hermíticas no es un subespacio del espacio de todas las matrices  $n \times n$  sobre  $C$ .

**Ejemplo 7.** El subconjunto  $\vec{0}$  que consta sólo del vector nulo es un subespacio de  $V$ , llamado subespacio nulo de  $V$ .

**Teorema 2.** Sea  $V$  un espacio vectorial sobre el campo  $K$ . La intersección de cualquier colección de subespacios de  $V$  es un subespacio de  $V$ .

**Demostración.** Sea  $(W_j)$  una colección de subespacios de  $V$ , y sea  $W = \bigcap W_j$ . Como cada  $W_j$  es un subespacio, cada uno contiene el vector nulo, luego  $W = \bigcap \neq 0$ . También cada  $W_j$  contiene  $(cx+y)$  lo que implica que  $(cx+y) \in W$ . Por el Teorema 1  $W$  es un subespacio de  $V$ .

**Definición B.4 Subespacio generado.** Sea  $S$  un conjunto de vectores de un espacio vectorial  $V$ . El subespacio generado por  $S$  se define como la intersección  $W$  de todos los subespacios de  $V$  que contienen a  $S$ .

**Teorema 3.** El subespacio generado por un subconjunto  $S$  no vacío de un espacio vectorial  $V$  es el conjunto de todas las combinaciones lineales de los vectores de  $S$ .

**Demostración.** Sea  $W = \langle S \rangle$  el subespacio generado por  $S$ , entonces  $W$  contiene toda la combinación lineal de vectores de  $S$ , de la forma  $x = c_1x_1 + c_2x_2 + \dots + c_mx_m$ . Así  $W$  contiene el conjunto  $L$  de todas las combinaciones lineales de vectores de  $S$ . Por otra parte  $S \subset L$  y  $L$  es no vacío. Si  $xy \in L$  entonces  $y = k_1y_1 + \dots + k_ny_n$ , y  $(\alpha x + y) = \sum_{i=1}^m (\alpha c_i)x_i + \sum_{j=1}^n k_jy_j \in L$ . Por lo tanto  $S \subset L \subset W \Rightarrow L = \bigcap W$  tal que  $S \subset W$ .

**Definición B.5 Dependencia lineal.** Sea  $V$  un espacio vectorial sobre  $K$ .  $S \subset$

$V$  se dice linealmente dependiente si existen vectores distintos  $x_1, x_2, \dots, x_n$  de  $S$  y escalares  $c_1, c_2, \dots, c_n$  de  $K$ , no todos nulos, tales que  $c_1x_1 + c_2x_2 + \dots + c_nx_n = 0$ . Un conjunto que no es linealmente dependiente se dice linealmente independiente. Como consecuencia de esta definición se tiene que:

- Todo conjunto que contiene el vector no nulo es linealmente dependiente.
- Un conjunto  $S$  es linealmente independiente  $\iff$  todo subconjunto finito de  $S$  es linealmente independiente.

**Definición B.6 Base de un espacio vectorial.** Sea  $V$  un espacio vectorial sobre  $K$ . Una base de  $V$  es un conjunto de vectores linealmente independientes de  $V$ , que genera el espacio  $V$ . El espacio  $V$  es de dimensión finita si tiene una base finita.

**Teorema 4.** Si  $W_1$  y  $W_2$  son subespacios de dimensión finita de un espacio vectorial, entonces  $\dim W_1 + \dim W_2 = \dim(W_1 \cap W_2) + \dim(W_1 + W_2)$

**Definición B.7 Transformación lineal.** Sean  $V$  y  $W$  dos espacios vectoriales sobre el cuerpo  $K$ . Una función  $T : V \longrightarrow W$  es una transformación lineal si  $T(cx + y) = c(Tx) + Ty, y \in V$  y  $\forall c \in K$ .

**Ejemplo 8.** Sea  $K$  un campo y sea  $V$  el espacio vectorial de las funciones polinomios de grado  $k$ . Sea  $D : V \longrightarrow V$  definida por  $(Df)(x) = c_1 + 2c_2x + \dots + kc_kx^{k-1}$ . Entonces  $D$  es una transformación lineal.

**Ejemplo 9.** Sea  $\mathfrak{R}$  el campo de los números reales y sea  $V$  el espacio de todas las funciones continuas de  $\mathfrak{R}$  en  $\mathfrak{R}$ . Se define  $T : V \longrightarrow V$  por  $(Tf)(x) = \int_0^x f(t)dt$ . Entonces  $T$  es una transformación lineal. La función  $(Tf)$  es continua y tiene primera derivada continua. La linealidad de las integraciones es una de sus propiedades fundamentales.

**Definición B.8 Espacio nulo y rango de una transformación lineal.** Sean  $V$  y  $W$  dos espacios vectoriales sobre el cuerpo  $K$  y sea  $T : V \longrightarrow W$  una transformación lineal. El espacio nulo de  $T$  es el conjunto de todos los vectores  $x$  de  $V$  tal que  $Tx = 0$ . Si  $V$  es de dimensión finita, el rango de  $T$  es la dimensión de la imagen de  $T$ , y la nulidad de  $T$  es la dimensión del espacio nulo de  $T$ .

**Teorema 5.** Sean  $V$  y  $W$  dos espacios vectoriales sobre el campo  $K$  y sea  $T : V \longrightarrow W$  una transformación lineal. Si  $\dim V < \infty$ , entonces  $\text{rango}(T) + \text{nulidad}(T) = \dim V$ . Si  $V$  es un espacio vectorial sobre el campo  $K$ , una transformación lineal

$F : V \longrightarrow K$  se llama función lineal sobre  $V$ .

**Ejemplo 10.** Sea  $n \in \aleph$  y  $K$  un campo. Si  $A \in K^{n \times n}$  la traza de  $A$  es el escalar  $trA = a_{11} + a_{22} + \dots + a_{nn}$ . La función traza es un funcional en el espacio de las matrices  $K^{n \times n}$ .

**Ejemplo 11.** Sea  $[a, b] \subset \mathfrak{R}$  y sea  $C([a, b])$  el espacio de las funciones reales continuas sobre  $[a, b]$ . Entonces  $L(g) = \int_a^b g(t)dt$  define un funcional lineal  $L$  en  $C[a, b]$ .

Si  $V$  es un espacio vectorial, el conjunto de los funcionales lineales sobre  $V$  forman, naturalmente, un espacio vectorial. Este es el espacio  $L(V, K)$ . Se designa este espacio por  $V^*$  y se llama espacio dual del espacio  $V$  :

$$V^* = L(V, K)$$

Si  $V$  es de dimensión finita  $dimV^* = dimV$ .

Sea  $\beta = v_1, v_2; \dots; v_n$  una base de  $V$ . Para cada  $i$  existe un funcional lineal único  $f_i$  en  $V$ , tal que  $f_i(v_j) = \delta_{ij}$ . De esta forma se obtiene de  $\beta$  un conjunto de  $n$  funcionales lineales distintos  $f_1, f_2, \dots, f_n$  sobre  $V$ . Estos funcionales son linealmente independientes y como  $dimV^* = n$ , deben ser tales que  $\beta^* = f_1, f_2, \dots, f_n$  es una base de  $V^*$ . Esta base se llama base dual de  $\beta$ .

**Teorema 6.** Sea  $V(K)$  un espacio vectorial con  $dimV = n$  y sea  $\beta = v_1, v_2, \dots, v_n$  una base de  $V$ . Entonces existe una única base dual  $\beta^* = f_1, f_2, \dots, f_n$  de  $V^*$  tal que  $f_i(v_j) = \delta_{ij}$ . Para cada funcional lineal  $f$  sobre  $V$  se tiene:

$$f = \int_{i=1}^n f(v_i)f_i \tag{B.2}$$

y para cada vector  $v \in V$  se tiene:

$$v = \int_{i=1}^n f_i(v)v_i \tag{B.3}$$

$$\tag{B.4}$$

Esta ecuación dice que si  $\beta = (v_1, v_2, \dots, v_n)$  es una base ordenada de  $V$  y  $\beta^* = f_1, f_2, \dots, f_n$  es la base dual, entonces  $f_i$  es precisamente la función que asigna a cada vector  $v \in V$  la  $i$ -ésima coordenada de  $v$  respecto a la base ordenada  $\beta$ .

**Ejemplo 12.** Sea  $V$  el espacio vectorial de todas las funciones polinomios de  $\mathfrak{R}$  en  $\mathfrak{R}$  que tienen grado  $\leq 2$ . Sean  $t_1, t_2, t_3$  tres números reales distintos arbitrarios, y sea

$$L_i(p) = p(t_i)$$

Entonces  $L_1, L_2, L_3$  son funcionales lineales sobre  $V$ . Estos funcionales son linealmente independientes, y como  $\dim V = 3$ , estos forman una base de  $V^*$ . La base  $p_1, p_2, p_3$  de  $V$  tal que  $L_1, L_2, L_3$  es su dual debe satisfacer:

$$L_i(p_j) = p_j(t_i) = \delta_{ij} \quad (\text{B.5})$$

$$(\text{B.6})$$

Estas funciones polinomios son:

$$p_1(x) = \frac{(x-t_2)(x-t_3)}{(t_1-t_2)(t_1-t_3)}, p_2(x) = \frac{(x-t_1)(x-t_3)}{(t_2-t_1)(t_2-t_3)}, p_3(x) = \frac{(x-t_1)(x-t_2)}{(t_3-t_1)(t_3-t_2)} \quad (\text{B.7})$$

$$(\text{B.8})$$

A las transformaciones lineales de  $V$  en  $V$  se les llama frecuentemente operadores lineales en  $V$ . Si  $T$  es un operador lineal en  $V$  tal que:

- (i) es uno a uno y
- (ii) aplica  $V$  sobre  $V$

se dice que  $T$  es regular o invertible. En este caso  $T^{-1}$  es lineal y  $T(T^{-1})(x) = T^{-1}(T(x)) = x$ .

### Definiciones:

1. Sea  $L(X, Y)$  el conjunto de todas las transformaciones lineales del espacio vectorial  $X$  en el espacio vectorial  $Y$ . Si  $T, F \in L(X, Y)$  y  $c, k$  son escalares, se define  $cT + kF$  por:

$$(cT + kF)(x) = cT(x) + kF(x), x \in X \quad (\text{B.9})$$

$$(\text{B.10})$$

2. Si  $X, Y, Z$  son espacios vectoriales, si  $T \in L(X, Y)$  y  $F \in L(Y, Z)$ , se define el producto  $FT$  por:

$$(FT)x = F(Tx), x \in X \quad (\text{B.11})$$

$$(\text{B.12})$$

Entonces  $FT \in L(X, Z)$ . Observemos que  $FT$  no es necesariamente igual, incluso cuando  $X = Y = Z$ .

3. Para  $T \in L(\mathfrak{R}^n, \mathfrak{R}^m)$ ; se define la norma  $\|T\|$  de  $T$  como el extremo superior de todos los números  $|Tx|$ , donde  $x$  varía en  $\mathfrak{R}^n$  con  $|x| \leq 1$ . Obsérvese que la desigualdad  $|Tx| \leq \|T\||x|$  se cumple para todo  $x \in \mathfrak{R}^n$ . Además, si  $\lambda$  es tal que  $|Tx| \leq \lambda|x|, \forall x \in \mathfrak{R}^n$ , entonces  $\|T\| \leq \lambda$ .

### B.1.1. Espacios con producto interno

Un producto interno sobre un espacio vectorial es una función con propiedades similares a las del producto escalar en  $\mathfrak{R}^3$ , y en términos de tal producto se puede también definir longitud y ángulo. La noción general de ángulo se restringirá al concepto de ortogonalidad de vectores.

**Demostración B.9** Sea  $K = \mathfrak{R}oC$  y  $V(K)$ . Un producto interno sobre  $V$  es una función que asigna a cada par ordenado de vectores  $x, y$  de  $V$  un escalar  $\langle x, y \rangle$  de  $K$ , de tal modo que  $\forall x, y \in V$  y  $\forall c \in K$  tenemos:

- (a)  $\langle x + y, z \rangle = \langle x, z \rangle + \langle y, z \rangle$
- (b)  $\langle cx, y \rangle = c\langle x, y \rangle$
- (c)  $\langle y, x \rangle = \overline{\langle x, y \rangle}$ , donde el super-rayado denota conjugación compleja
- (d)  $\langle x, x \rangle > 0, \text{ si } x \neq 0$ ; Obsérvese que (a), (b) y (c) implican:
- (e)  $\langle x, cy + z \rangle = \overline{c}\langle x, y \rangle + \langle x, z \rangle$

Es claro que si  $K = \mathfrak{R}$  la conjugación compleja no se acciona.

#### Ejemplo 13.

1. En  $K^n$  se define el producto interno canónico sobre  $x = (x_1, x_2, \dots, x_n)$  e  $y = (y_1, y_2, \dots, y_n)$  por  $\langle x, y \rangle = \sum_{j=1}^n x_j \overline{y_j} = xy^*$

Cuando  $K = \mathfrak{R}$ ,  $\langle x, y \rangle = \sum_{j=1}^n x_j y_j$ .

2. Sea  $V = K^{n \times n}$ , el espacio de las matrices  $n \times n$  sobre  $K$ .  $MV$  es isomorfo a  $K^{n^2}$ , luego  $\langle A, B \rangle = \sum_{j,k} A_{jk} \overline{B_{jk}}$  define un producto interno sobre  $V$ . Además, si se introduce la matriz transpuesta conjugada  $B^*$ , donde  $B_{kj}^* = \overline{B_{jk}}$ , este producto interno se puede expresar mediante la función traza  $\langle A, B \rangle = tr(AB^*) = tr(B^*A)$ .

3. Sea  $V$  el espacio vectorial de las funciones continuas de valor complejo en el



intervalo unitario,  $0 \leq t \leq 1$ . En  $V$  se define el producto interno:

$$\langle f, g \rangle = \int_0^1 f(t)\overline{g(t)}dt \quad (\text{B.13})$$

En el espacio de funciones reales se omite la conjugación.

Es útil saber que un producto interno sobre un espacio vectorial, real o complejo, está determinado por otra función, la llamada forma cuadrada determinada por el producto interno. Para definirla se representa primero la norma o longitud de  $x$  respecto al producto interno por  $\|x\| = \sqrt{\langle x, x \rangle}$ . La forma cuadrática determinada por el producto es la función que asigna a cada vector  $x$  el escalar  $\|x\|^2$ . Por las propiedades del producto interno se sigue:  $\|x \pm y\|^2 = \|x\|^2 \pm 2\text{Re}\langle x, y \rangle + \|y\|^2$ ,  $\forall x, y$  en el caso real

$$\langle x, y \rangle = \frac{1}{4}\|x + y\|^2 - \frac{1}{4}\|x - y\|^2 \quad (\text{B.14})$$

en el caso complejo

$$\langle x, y \rangle = \frac{1}{4}\|x + y\|^2 - \frac{1}{4}\|x - y\|^2 + \frac{i}{4}\|x + iy\|^2 - \frac{i}{4}\|x - iy\|^2 \quad (\text{B.15})$$

Estas igualdades se llaman identidades de polarización.

**Definición B.10 Espacio producto interno.** Es un espacio real o complejo junto con un producto interno definido en ese espacio. Un espacio producto interno real de dimensión finita se llama espacio euclideo. Un espacio con producto interno complejo se llama espacio unitario.

**Teorema 7.** Si  $V$  es un espacio producto interno, entonces  $\forall x, y \in V$  y  $\forall c \in K$  tenemos:

1.  $\|cx\| = |c|\|x\|$ ;
2.  $\|x\| > 0$ , para  $x \neq 0$ ;
3.  $\|\langle x, y \rangle\| \leq \|x\|\|y\|$ , (Desigualdad de Cauchy-Schwarz):
4.  $\|x + y\| \leq \|x\| + \|y\|$ .

**Definición B.11** Sea  $V$  un espacio producto interno y sean  $x, y \in V$ . Entonces  $x$  es ortogonal a,  $y$  si  $\langle x, y \rangle = 0$ ; y se dirá que  $x$  e  $y$  son ortogonales. Si  $S \subset V$  es un conjunto de vectores, se dice que  $S$  es un conjunto ortogonal siempre

que todos los pares distintos de  $S$  sean ortogonales. Un conjunto ortonormal es un conjunto ortogonal  $S$  tal que

$$\|x\| = 1, \forall x \in S \tag{B.16}$$

**Ejemplo 14.**

1. El vector cero es ortogonal a todo vector de  $V$  y es el único vector con esa propiedad.

2. La base canónica en  $\mathfrak{R}^n$  o  $C^n$  es un conjunto ortonormal con respecto al producto interno canónico (pic).

3. El vector  $(x, y)$  es ortogonal a  $(-y, x)$  con respecto al pic en  $\mathfrak{R}^2$ .

4. Sea  $V = C([0, 1])$ , espacio de funciones continuas de valor complejo o real en el intervalo  $0 \leq x \leq 1$  con el producto interno

$$\langle f, g \rangle = \int_0^1 f(x)\overline{g(x)}dx \tag{B.17}$$

Si  $f_n(x) = \sqrt{2\cos(2\pi nx)}$  y  $g_n(x) = \sqrt{2\sin(2\pi nx)}$ , entonces el conjunto de funciones  $1, f_1, g_1, f_2, g_2, \dots$  es un conjunto infinito ortonormal. En el caso complejo se pueden formar combinaciones de la forma  $\frac{1}{\sqrt{2}}(f_n + ig_n)$ ,  $n = 1, 2, 3, \dots$  obteniéndose un nuevo conjunto ortonormal  $S$  que consta de todas las funciones

$$h_n(x) = e^{i2\pi nx}, n = \pm 1, \pm 2, \dots \tag{B.18}$$

El número real  $\|f\| = \langle f, f \rangle^{\frac{1}{2}}$  es la norma de  $f$ . Si  $f, g$  son ortogonales, entonces

$$\|f + g\|^2 = \langle f + g, f + g \rangle = \langle f, f \rangle + \langle g, g \rangle + \langle f, g \rangle + \langle g, f \rangle = \|f\|^2 + \|g\|^2 \tag{B.19}$$

Si  $f, g$  son vectores cualesquiera con  $f \neq 0$ , entonces  $h = g - \frac{\langle f, g \rangle}{\langle f, f \rangle} f$  es ortogonal a  $f$ , y con estos se prueba la desigualdad de Schwarz

$$|\langle f, g \rangle| \leq \|f\|\|g\| \tag{B.20}$$

El problema de aproximación tiene una importante significación en espacios de dimensión infinita, sin embargo tiene un sencillo significado en espacios finito dimensionales.[16]

**B.1.2. Espacios vectoriales de dimensión infinita**

Una de las formas más fáciles de obtener un espacio vectorial de dimensión infinita es prolongando  $< n$ , el espacio de  $n$ -tuples de números reales, al espacio de sucesiones infinitas de números reales. Si  $x = (x_1, x_2, \dots)$  e  $y = (y_1, y_2, \dots)$  son sucesiones infinitas de números reales, diremos que  $x = y$  si  $x_i = y_i \forall i \in \mathfrak{N}$ . El conjunto de estas sucesiones

forma un espacio vectorial con la adición y multiplicación por un escalar conocidas. Pretendemos tener un producto interno en espacio, luego tenemos que restringir las sucesiones en alguna forma. Deseamos definir el producto interno

$$\langle x, y \rangle = x_1y_1 + x_2y_2 + \dots = \sum_{i=1}^{\infty} x_iy_i \quad (\text{B.21})$$

por tanto la norma es de la siguiente forma

$$\|x\| = \left( \sum_{i=1}^{\infty} x_i^2 \right)^{\frac{1}{2}} \quad (\text{B.22})$$

Como ahora estamos tratando con sucesiones infinitas, para asegurar la convergencia restringiremos las sucesiones a aquellas tales que  $\sum_{i=1}^{\infty} x_i^2 < \infty$ . Como se ha impuesto una restricción, es necesario verificar los axiomas de espacio vectorial. Sólo verificaremos cerradura de la suma. Si  $\sum_{i=1}^{\infty} x_i^2 < 1$  y  $\sum_{i=1}^{\infty} y_i^2 < 1$  consideremos:

$$0 \leq (|x_i| - |y_i|)^2 = x_i^2 + y_i^2 - 2|x_iy_i| \quad (\text{B.23})$$

tenemos que

$$2|x_iy_i| \leq x_i^2 + y_i^2 \quad (\text{B.24})$$

y entonces

$$(x_i + y_i)^2 = x_i^2 + y_i^2 + 2x_iy_i \quad (\text{B.25})$$

y

$$\sum_{i=1}^{\infty} (x_i + y_i)^2 \leq 2 \sum_{i=1}^{\infty} x_i^2 + 2 \sum_{i=1}^{\infty} y_i^2 < \infty \quad (\text{B.26})$$

La verificación de los otros axiomas se hace por cálculo directo. Para el producto interno tenemos que

$$|x_iy_i| \leq \frac{1}{2}(x_i^2 + y_i^2) \quad (\text{B.27})$$

y entonces

$$\sum_{i=1}^{\infty} |x_i y_i| \leq \frac{1}{2} \sum_{i=1}^{\infty} x_i^2 + \frac{1}{2} \sum_{i=1}^{\infty} y_i^2 \quad (\text{B.28})$$

lo que muestra que  $\sum_{i=1}^{\infty} x_i y_i$  converge absolutamente. Las cinco propiedades del producto interno son fáciles de verificar. Así mostramos que el conjunto de sucesiones es un espacio vectorial real con producto interno. Consideremos el conjunto infinito de vectores  $(e_i)$ , tenemos que  $x = \sum_{i=1}^{\infty} x_i e_i$ . Esta es una serie infinita de vectores, así que debemos definir lo que entendemos por convergencia de una de tales series. Sea  $x_n$  el vector de la suma parcial  $x_n = \sum_{i=1}^n x_i e_i$ .

Entonces  $\|x_n - x\| = (\sum_{i=n+1}^{\infty} x_i^2)^{\frac{1}{2}} \rightarrow 0$  cuando  $n \rightarrow \infty$  puesto que la serie  $\sum_{i=1}^{\infty} x_i^2$  converge.

**Definición B.12** Si  $V$  es un espacio de dimensión infinita con un producto interno, entonces  $v_1, v_2, \dots$  es una base ortonormal si:

$$\langle v_i, v_j \rangle = \delta_{ij}, \text{ con } i, j = 1, 2, \dots, n., \text{ la serie } \sum_{i=1}^{\infty} x_i v_i \text{ converge a } x, \forall x \in V$$

donde

$$x_i = \langle x, v_i \rangle \quad (\text{B.29})$$

es la coordenada de  $x$  respecto a  $v_i$ .

**Definición B.13** Si  $V$  es un espacio de dimensión infinita con su norma, una sucesión de vectores  $x_{n=1}^{\infty}$  es una sucesión de Cauchy si

$$\lim_{n, m \rightarrow \infty} \|x_n - x_m\| = 0, \quad (\text{B.30})$$

o en otra forma, si

$$\forall \epsilon > 0; \exists N \text{ tal que } \|x_n - x_m\| < \epsilon \quad (\text{B.31})$$

cuando  $n, m > N$

**Teorema 10.** Sea  $V(K)$  con  $\dim V = 1$  con una norma. Si una sucesión de vectores  $x_{n=1}^{\infty}$  converge a  $x$  en  $V$ , entonces la sucesión es una sucesión de Cauchy. El inverso de este teorema no es, en general, verdadero. Los espacios en que toda

sucesión de Cauchy converge a un vector en el espacio se llaman espacios completos.

**Definición B.14** Sea  $V$  un espacio vectorial con una norma.  $V$  es un espacio completo si toda sucesión de Cauchy en  $V$  converge a un vector en  $V$ . Un espacio vectorial normado y completo se llama espacio de Banach. Si la norma es derivada de un producto interno, el espacio se llama espacio de Hilbert.

**Teorema 11.** El espacio  $l^2\mathfrak{R} = \{x_n\}_{n=1}^{\infty} \mid \sum_{i=1}^{\infty} x_i^2 < \infty$ , con un producto interno  $\langle x, y \rangle = \sum_{i=1}^{\infty} x_i y_i$ , es completo. En espacios de dimensión infinita, se pueden encontrar infinitos conjuntos de funciones ortonormales.

**Ejemplo 15.** Encontrar un conjunto de polinomios ortonormales en  $C[-1, 1]$ . El producto escalar es  $\langle f, g \rangle = \int_{-1}^1 f(x)g(x)dx$

**Solución.** Comenzamos con una función constante  $h_1(x) = c$ .

$$\int_{-1}^1 h_1(x)h_1(x)dx = 1 \Rightarrow c = \frac{1}{\sqrt{2}} \quad (\text{B.32})$$

Seguimos con una función lineal  $h_2(x) = ax + b$ .

$$\int_{-1}^1 h_1(x)h_2(x)dx = 0 \Rightarrow \int_{-1}^1 \frac{ax + b}{\sqrt{2}}dx = 0; \sqrt{2}b = 0 \quad (\text{B.33})$$

y considerando que  $b = 0$  en  $h_2$

$$\int_{-1}^1 h_2(x)h_2(x)dx = 1 \Rightarrow \int_{-1}^1 a^2 x^2 dx = 1 \Rightarrow \frac{2a^2}{3} = 1 \quad (\text{B.34})$$

y así  $a = \frac{\sqrt{3}}{\sqrt{2}}$

Ahora consideremos una función cuadrática  $h_3(x) = rx^2 + kx + l$  Las constantes  $r, k, l$  se determinan desde las tres condiciones

$$\int_{-1}^1 h_3(x)h_3(x)dx = 1$$

$$\int_{-1}^1 h_1(x)h_3(x)dx = 0$$

$$\int_{-1}^1 h_2(x)h_3(x)dx = 0$$

$$\int_{-1}^1 h_1(x)h_3(x)dx = 0 \Rightarrow \int_{-1}^1 \frac{rx^2 + kx + l}{\sqrt{2}}dx = \frac{\sqrt{2}r}{3} + \sqrt{2}l = 0$$

$$\int_{-1}^1 \frac{\sqrt{3}}{\sqrt{2}} x(rx^2 + kx + l) dx = \frac{\sqrt{3}}{\sqrt{2}} k = 0$$

Así,  $r = -3l$ ,  $l^2 \int_{-1}^1 (3x^2 - 1)^2 dx = \frac{8}{5} l^2 = 1$ . Entonces  $l = \frac{\sqrt{5}}{2\sqrt{2}}$

Los tres primeros polinomios son:

$$h_1(x) = \frac{1}{\sqrt{2}}; h_2(x) = \frac{\sqrt{3}}{\sqrt{2}} x; h_3(x) = \frac{\sqrt{5}(3x^2 - 1)}{2\sqrt{2}} \tag{B.35}$$

Este proceso se puede continuar indefinidamente. En el  $n$ -ésimo paso hay  $n$  constantes por determinar de  $n - 1$  condiciones de ortogonalidad más una condición de normalización. El polinomio general es  $h_n = \sqrt{\frac{2n-1}{2}} P_{n-1}(x)$ , donde  $P_n(x)$  es el polinomio de Legendre dado por  $P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n$

**Teorema 9.** Si  $f_1, f_2, \dots$  es una base ortonormal, entonces la mejor aproximación  $\sum_{k=1}^n a_k f_k$  de  $g$  está dada tomando  $a_k = \langle g, f_k \rangle$ . El error cuadrático mínimo es  $E_n^2(g) = \|g\|^2 - \sum_{k=1}^n |a_k|^2$ . Los coeficientes  $a_k = \langle g, f_k \rangle$  se llaman coeficientes de Fourier de  $g$  con respecto a la sucesión ortonormal  $f_1, f_2, \dots$

La aproximación  $\sum_{k=1}^n a_k f_k$  se llama  $n$ -ésima aproximación de Fourier. Como  $E_n^2(g) \geq 0$  no es creciente,  $\lim_{n \rightarrow \infty} E_n^2(g)$  existe y también  $\sum_{k=1}^{\infty} |a_k|^2$ . De donde:

$$0 \leq \lim_{n \rightarrow \infty} E_n^2(g) = \|g\|^2 - \sum_{k=1}^{\infty} |a_k|^2 \tag{B.36}$$

luego  $\sum_{k=1}^{\infty} |a_k|^2 \leq \|g\|^2$ .

Esta última ecuación se conoce como Desigualdad de Bessel.

# Apéndice C

## Especificaciones técnicas del DSP TMS320C6711

Las familias de dispositivos TMS320C62x / TMS320C67x, ejecutan un máximo de 8 instrucciones, de 32 bits, por ciclo. El ( CPU ) contiene 32 registros de propósito general, de 32 bits y 8 unidades funcionales. Estos dispositivos tienen un conjunto completo de herramientas de desarrollo y optimización, que incluyen un compilador C eficiente, un optimizador de ensamblador para simplificar la planificación y programación del lenguaje ensamblador, y un depurador con interfase grafica, basada en Windows, para visualizar las características de ejecución en el código fuente.

Además, contiene una tarjeta de emulación de hardware compatible con la interfase del emulador TI XDS510. Estas herramientas cumplen con los estándares 1149.11990, revisión de acceso a puerto y arquitectura de verificación de límites, de la ( IEEE ).

Las características de los dispositivos C62x / C67x, incluyen:

1. Un CPU avanzado VLIW (very long instruction word) con 8 unidades funcionales, que incluyen 2 multiplicadores y 6 ALU's (unidades lógico aritméticas).
  - a) Ejecuta un máximo de 8 instrucciones por ciclo, 10 veces mas que los DSP's típicos.
  - b) Permite rápido tiempo de desarrollo en diseños con código RISC altamente efectivos.
2. Empaquetado de instrucción.
  - a) Obtiene el tamaño del código equivalente a las 8 instrucciones ejecutadas en serie o en paralelo.

- b) Reduce el tamaño del código, el consumo de energía y el *fetch* del programa.
- 3. Ejecución condicional de todas las instrucciones.
  - a) Reduce los saltos costosos.
  - b) Incrementa el paralelismo para mantener un alto desempeño.
- 4. Ejecuta el código programado, en unidades funcionales independientes.
- 5. Proporciona soporte eficiente de memoria para una variedad de aplicaciones de 8, 16 y 32 bits de datos.
- 6. Maneja operaciones aritméticas de 40 bits, adicionando precisión extra a vocoders y otras aplicaciones computacionalmente intensivas.
- 7. Proporciona soporte de normalización y saturación, en operaciones aritméticas claves.
- 8. Soporta operaciones comunes, halladas en aplicaciones de control y manipulación de datos, como: manipulación de campos, extracción de instrucción, activación, desactivación y conteo de bits.

Además, el C67x tiene las siguientes características:

- Un máximo de 1336 ( MIPS ) (millones de instrucciones por segundo), a 167 MHz.
- Un máximo de 1 G ( FLOPS ) (operaciones en punto flotante por segundo), a 167 MHz, para operaciones de precisión simple.
- Un máximo de 250 M FLOPS a 167 MHz, para operaciones de doble precisión.
- Un máximo de 688 M FLOPS a 167 MHz, para operaciones de multiplicación y acumulación.
- Soporte de hardware para operaciones de punto flotante, de simple y doble precisión (con formato IEEE).
- Multiplicación entera de 32 x 32 bits, con resultado de 32 o 64 bits.

Los dispositivos C62x / C67x tienen la siguiente variedad de opciones de memoria y periféricos:

- Amplia memoria RAM para ejecución rápida de algoritmos.



- Soporta interfases para memoria externa de 32 bits (SDRAM, SBSRAM, SRAM y otras memorias asíncronas), para aumentar el rango de memoria externa y maximizar el desempeño del sistema.
- Acceso a la memoria y periféricos de los dispositivos C62x / C67x a través del puerto *host*.
- Controlador del multicanal DMA.
- Puerto(s) serie multicanal.
- Timer(s) de 32 bits.

## C.1. Arquitectura de los dispositivos TMS320 C62x/C67x

Los dispositivos 'C6211, 'C6711, 'C6701, 'C6201 y 'C6202 operan a 150, 150, 167, 200 y 250 MHz respectivamente. Todos estos DSP's ejecutan un máximo de 8 instrucciones por ciclo. El DSP 'C6211 es de punto fijo mientras que el 'C6711 es de punto flotante. Los procesadores 'C62x/C67x consisten de tres partes: el CPU, los periféricos y la memoria. Ocho unidades funcionales operan en paralelo (seis ALU's y dos multiplicadores), con dos conjuntos similares de cuatro unidades funcionales básicas. Las unidades se comunican usando un camino cruzado entre dos clasificaciones de registros, cada una de los cuales contiene 16 registros de 32 bits. La Figura C.1 muestra el diagrama de bloque de los dispositivos TMS320C62x/C67x.

### C.1.1. Unidad de procesamiento central (CPU)

El CPU contiene:

- Unidad *fetch* de programa.
- Unidad de despacho de instrucción.
- Unidad de decodificación de instrucción.
- 32 registros de 32 bits.
- Dos caminos de datos (*path*), cada uno con cuatro unidades funcionales.
- Registros de control.
- Lógica de control.

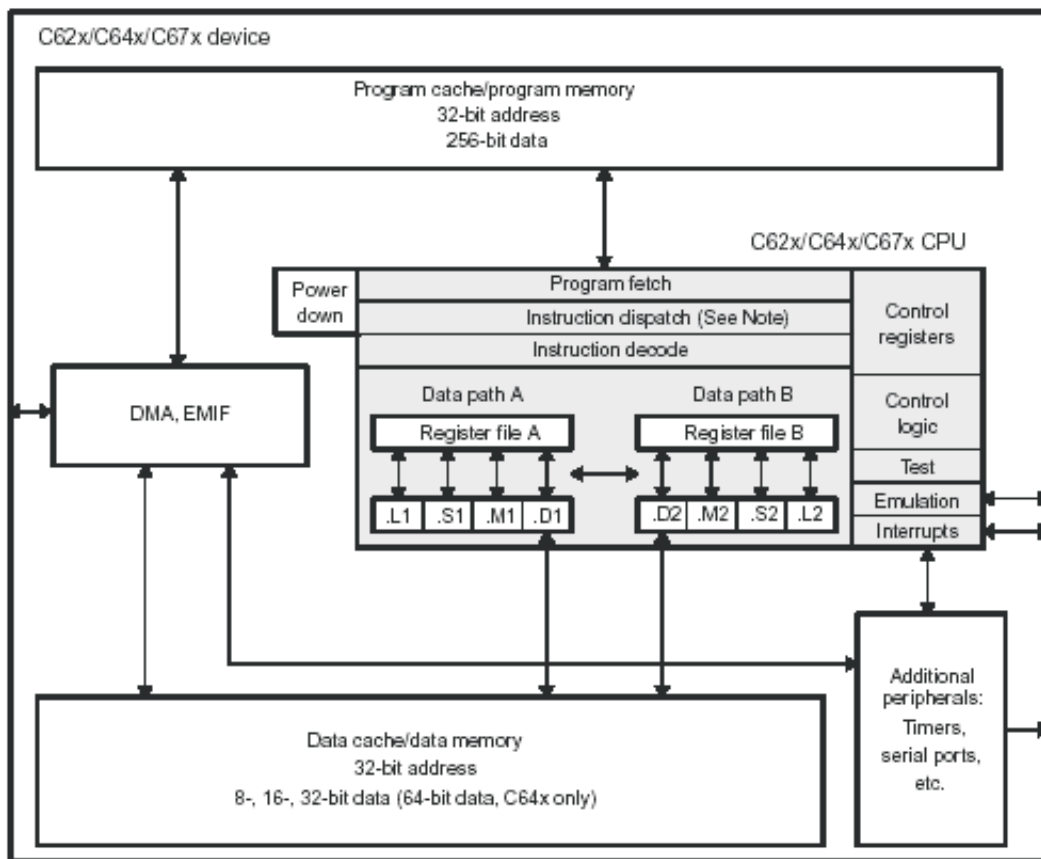


Figura C.1: Diagrama de bloques del TMS320C62x/C67x.

- Lógica de interrupción, emulación y prueba.

El CPU tiene dos caminos de datos (A y B), cada camino tiene cuatro unidades funcionales (.L, .S, .M y .D) y un archivo de registros que contiene 16 registros de 32 bits (*register file*). Las unidades funcionales ejecutan operaciones de lógica, corrimiento, multiplicación y direccionamiento de datos. Todas las instrucciones aceptan operaciones de carga y almacenamiento sobre los registros. Las dos unidades de direccionamiento de datos (.D1 y .D2) son exclusivamente responsables de toda la transferencia de datos entre los archivos de registros y la memoria.

### C.1.2. Caminos de datos del CPU

Los caminos de datos del CPU consisten de: dos archivos de registros de propósito general (A y B), ocho unidades funcionales (.L1, .L2, .S1, .S2, .M1, .M2, .D1 y .D2), dos caminos de lectura de memoria (LD1 y LD2), dos caminos de almacenamiento de memoria (ST1 y ST2), dos caminos cruzados entre los archivos de registros (1X y 2X) y dos caminos de direccionamiento de datos (DA1 y DA2). La Figura C.2 muestra el camino de los datos del CPU 'C67x.

### C.1.3. Archivos de registros de propósito general (*register files*)

Hay dos archivos de registros de propósito general (A y B) en los caminos de datos del 'C62x/'C67x. Cada uno de esos archivos contiene 16 registros de 32 bits (A0-A15) para el archivo A y B0-B15) para el archivo B. Los registros de propósito general pueden ser usados para manejar datos o punteros de direccionamiento de estos. Los registros A1, A2, B0, B1 y B2 pueden ser utilizados como registros de condición. Los registros 4-A7 y B4-B7 pueden ser usados para el direccionamiento circular.

Los archivos de registros de propósito general soportan datos de 32 y 40 bits de punto fijo. Los datos de 32 bits, pueden estar contenidos en cualquier registro de propósito general. Los datos de 40 bits están contenidos en dos registros; los 32 bits menos significativos del dato (LSB) son colocados en un registro par y los restantes ocho bits más significativos (MSB) son colocados en los ocho bits menos significativos del registro próximo superior (que es siempre un registro impar). El 'C67x también usa ese par de registros para colocar valores de punto flotante de doble precisión de 64 bits.

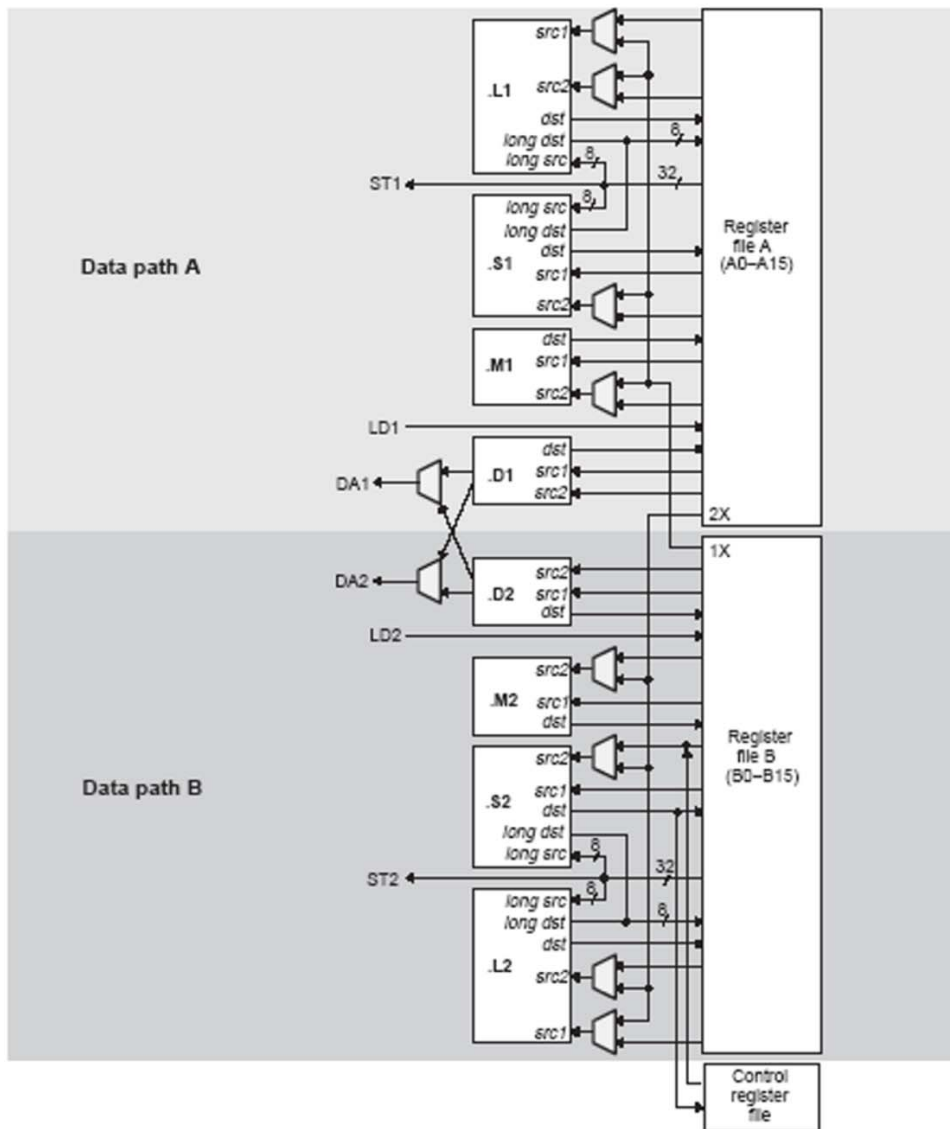


Figura C.2: Camino de datos del TMS320C67x.

### C.1.4. Unidades funcionales

Las ocho unidades funcionales en los caminos de datos del 'C62x/C67x pueden ser divididas en dos grupos de cuatro; cada unidad funcional, en un camino de datos, es casi idéntica a la unidad correspondiente, en el otro camino de datos (p. ej. .L1 es muy similar a .L2). Las unidades funcionales son descritas en el cuadro C.1.

La mayoría de los caminos en el CPU, soportan operaciones de 32 bits, y algunas soportan operaciones largas (40 bits). Cada unidad funcional tiene su propio puerto de escritura de 32 bits, en un archivo de registros de propósito general. Todas las unidades terminan en 1 (por ejemplo, .L1) cuando se refiere al archivo de registros A y en 2 cuando se refiere al archivo de registros B. Cada unidad funcional tiene dos puertos de lectura de 32 bits, para cada operando *src1* y *src2*. Cuatro unidades (.L1, .L2, .S1 y .S2) tienen un puerto extra de 8 bits para ejecutar operaciones de 40 bits. Debido a que cada unidad tiene su propio puerto de escritura de 32 bits, las ocho unidades pueden ser usadas en paralelo en cada ciclo.

### C.1.5. Archivos de registros de control del TMS320 C62x/C67x

Una unidad (.S2) puede leer de y escribir hacia los registros de control. Mostrados en la Figura C.2. El cuadro C.2, menciona y describe, los registros de control contenidos en el archivo de registros de control. A cada registro se accede con la instrucción MVC.

El 'C67x posee tres registros de configuración adicionales, para soportar operaciones de punto flotante (ver cuadro C.3). Los registros especifican los modos de redondeo de punto flotante para las unidades .M y .L. También contienen campos de bit para advertir si *src1* y *src2* son NaN (no es un número) o números desnormalizados. Además si resulta *overflow* o *underflow*, es inexacto, infinito o inválido. Hay campos que advierten si una división por cero fue efectuada o si una comparación fue ejecutada con un NaN.

### C.1.6. Caminos entre archivos de registros (*Register Files Cross Paths*)

Cada unidad funcional lee directamente de y escribe directamente hacia el archivo de registros, dentro de su propio camino de datos. Esto es, las unidades .L1, .S1, .D1 y .M1 escriben en el archivo de registros A y las unidades .L2, .S2, .D2 y .M2 escriben en el archivo de registros B. Los archivos de registros son conectados a las unidades funcionales del archivo de registros opuesto, a través de los caminos cruzados 1X y 2X. Esos caminos cruzados permiten a las unidades funcionales, de un camino de datos,

Unidad Funcional	Operaciones en punto fijo	Operaciones en punto flotante
Unidad .L(.L1, .L2)	Operaciones aritméticas y comparación de 32 y 40 bits. Cuenta de 0's o 1's Mas a la izquierda, para 32 bits. Normalización de 32 y 40 bits. Operaciones lógicas de 32 bits.	Operaciones aritméticas. Operaciones de conversión: DP→SP, INT→DP, INT→SP.
Unidad .S(.S1, .S2)	Operaciones aritméticas de 32 bits Corrimientos de 32/40 bits y operaciones campos de bits en 32 bits. Operaciones lógicas de 32 bits. Saltos, generación de constantes. Transferencia de registros de/hacia registros (solamente .S2)	Comparación recíproca. Operaciones de raíz cuadrada. Operaciones de valor absoluto. Operaciones de Conversión SP a DP
Unidad .M(.M1, .M2)	Operaciones de multiplicación de 16x16 bits.	Operaciones de multiplicación de 32x32 bits. Operaciones de multiplicación de punto flotante.
Unidad .D(.D1, .D2)	Sumas, restas y cálculos de direccionamiento circular de 32 bits. Carga y almacenamiento con <i>offset</i> constante de 5 bits. Carga y almacenamiento con <i>offset</i> constante de 15 bits. (solo .D2)	Lectura de palabras dobles con <i>offset</i> constante de 5 bits.

Cuadro C.1: Unidades Funcionales y las operaciones realizadas.

Abreviatura	Nombre	Descripción
AMR	Registro de modo de direccionamiento.	Especifica si utiliza direccionamiento lineal o circular para cada uno de los ocho registros; también contiene el tamaño para el direccionamiento circular.
CSR	Registro de control de estado.	Contiene el bit de interrupción global, los bits de control del cache y otros bits de control de estado diversos.
IFR	Registro de bandera de interrupción.	Despliega el estado de las interrupciones.
ISR	Registros para activar interrupción.	Permite activar interrupciones manualmente.
ICR	Registro para interrupción.	Permite limpiar interrupciones pendientes manualmente.
IER	Registro para retorno de interrupción	Permite habilitar / deshabilitar interrupciones individuales.
NRP	Puntero de retorno de interrupción no mascarable	Contiene la dirección de retorno de una interrupción no mascarable.
PCE1	Contador del programa, fase E1	Contiene la dirección del paquete <i>fetch</i> (contiene el paquete de ejecución del pipeline) en la etapa E1.

Cuadro C.2: Registros de Control

Abreviatura	Nombre	Descripción
FADCR	Registro de configuración del sumador del punto flotante	Específica el modo <i>underflow</i> , modo de redondeo, NaN y otras excepciones para la unidad .L
FAUCR	Registro de configuración auxiliar de punto flotante	Especifica modos de <i>underflow</i> , modos de redondeo y otras excepciones para la unidad .S
FMCR	Registro de configuración del multiplicador de punto flotante	Especifica modos de <i>underflow</i> , modos de redondeo y otras excepciones para la unidad .M

Cuadro C.3: Registros de control extendido para el TMS320C67x

acceder a operandos de 32 bits, del lado opuesto. El camino cruzado 1X permite a las unidades funcionales del camino de datos A, leer su operando fuente del archivo de registros B. El camino cruzado 2X permite a las unidades funcionales del camino de datos B, leer su operando fuente del archivo de registros A.

### C.1.7. Caminos de memoria, cargas y almacenamiento

Hay dos caminos de 32 bits, para leer los datos de memoria en los registros de almacenamiento: LD1 para el archivo de registros A y LD2 para el archivo de registros B. El 'C67x también tiene un segundo camino de carga de 32 bits para ambos archivos de registros A y B. Este segundo camino permite a la instrucción LDDW leer simultáneamente dos registros de 32 bits en los lados A y B. Existen además dos caminos de 32 bits, ST1 y ST2, para almacenar valores de los registros a la memoria, para cada archivo de registros. Los caminos de lectura largos .L y .S son compartidos con los caminos de almacenamiento.

### C.1.8. Caminos de direccionamiento de datos

Los caminos de direccionamiento de datos (DA1 y DA2) mostrados en la Figura C.2 colocados fuera de las unidades .D, permiten generar direcciones de datos de un archivo de registros. Con eso se sostienen cargas y almacenamientos en memoria, desde el otro archivo de registros. Sin embargo, las cargas y almacenamientos ejecutados en paralelo, deben cargar a y del mismo archivo de registros. Aunque también existe la alternativa de que ambos usen un camino cruzado al registro opuesto.



Unidad .L	Unidad .M	Unidad .S		Unidad .D	
ABS	MPY	ADD	SET	ADD	STB(15-bitt offset) Solo .S2
ADD	MPYU	ADDK	SHL	ADDAB	STH(15-bitt offset) Solo .S2
ADDU	MPYUS	ADD2	SHR	ADDAH	STW(15-bitt offset) Solo .S3
AND	MPYSU	AND	SHRU	ADDAW	SUB
CMPEQ	MPYH	B disp	SHRL	LDB	SUBAB
CMPGT	MPYHU	B IRP	SUB	LDBU	SUBAH
CMPGTU	MPYHUS	B NRP	SUBU	LDH	SUBAW
CMPLT	MPYHSU	B reg	SUB2	LDHU	ZERO
CMPLTU	MPYHL	CLR	XOR	LDW	
LMBD	MPYHLU	EXT	ZERO	LDB	
MV	MPYHULS	EXTU		LDBU	
NEG	MPYHSLU	MV		LDH	
NORM	MPYLH	MVC		LDHU	
NOT	MPYLHU	MVK		LDW	
OR	MPYLUHS	MVKH		MV	
SADD	MPYLSHU	MVLKH		STB	
SAT	SMPY	NEG		STH	
SSUB	SMPYHL	NOT		STW	
SUB	SMPYLH	OR			
SUBU	SMPYH				
SUBC					
XOR					
ZERO					

Cuadro C.4: Mapeo de instrucciones de punto fijo y unidades funcionales

Unidad .L	Unidad .M	Unidad .S	Unidad .D
ADDDP	MPYDP	ABSDP	ADDAD
ADDSP	MPYI	ABSSP	LDDW
DPINT	MPYID	CMPEQDP	
DPSP	MPYSP	CMPEQSP	
INTDP		CMPGTDP	
INTDPU	CMPGTSP		
INTSP		CMLTDP	
INTSPU	CMPLTSP		
SPINT		RCPDP	
SPTRUNC		RCPSP	
SUBDP		RSQRDP	
SUBSP		RSQRSP	
SPDP			

Cuadro C.5: Mapeo de instrucciones de punto flotante y unidades funcionales

### C.1.9. Mapeo entre instrucciones y unidades funcionales

El cuadro C.4 muestra el mapeo entre las instrucciones y las unidades funcionales para las instrucciones de punto fijo del TMS320C62x/C67x. El cuadro C.5 muestra el mapeo entre las instrucciones y las unidades funcionales para las instrucciones de punto flotante del TMS320C67x.

## C.2. Modos de direccionamiento

Los modos de direccionamiento son lineales por default para los 'C62x y 'C67x aunque también existe el modo de direccionamiento circular. El modo de direccionamiento se especifica con el registro modo de direccionamiento (AMR).

Con todos los registros se puede ejecutar el direccionamiento lineal. Solo en ocho de ellos se puede ejecutar el direccionamiento circular: del A4 a A7 (que son usados por la unidad .D1) y del B4 a B7 (que son usados por la unidad D2). Ninguna otra unidad puede ejecutar direccionamiento circular.

Las instrucciones LDB/LDH/LDW, STB/STH/STW, ADDAB/ADDAH/ADDAW, y SUBAB/SUBAH/SUBAW se apoyan en el registro AMR, para determinar que tipo de calculo del direccionamiento es ejecutado por esos registros.

Los CPUs 'C62x/'C67x tienen arquitectura de carga/almacenamiento, lo que significa que la única manera de acceder datos en memorias es con la instrucción de

Tipo de direccionamiento	Ninguna Modificación del registro de dirección	Preincremento o predecremento del registro de dirección	Postincremento o postdecremento del registro de dirección
Registro Indirecto	*R	*++R	*R++
		*- - R	*R - -
Registro Relativo	*+R[ucst5]	*++R[ucst5]	*R++[ucst5 ]
	*-R[ucst5]	*- -R[ucst5]	*R - -[ucst5]
Base + Index	*+R[offsetR]	*++R[offsetR]	*R++[offsetR]
	*-R[offsetR]	*- -R[offsetR]	*R- -[offsetR]

Cuadro C.6: Generación de direccionamiento indirecto para Load/Store

carga o almacenamiento. El cuadro (C.6) muestra la sintaxis de un direccionamiento indirecto, para su localización en memoria.

### C.3. Interrupciones

Los CPUs 'C62x/'C67x tienen 14 interrupciones. Estas son *reset*, la interrupción no mascarable (NMI) e interrupciones de la 4 a la 15. Estas interrupciones corresponden a las señales **RESET**, **NMI** e **INT4-INT15** respectivamente, sobre los límites del CPU. En los mismos dispositivos 'C62x/'C67x, estas señales pueden estar ligadas directamente a los pines del dispositivo, conectando periféricos al chip, o pueden ser desactivadas permanentemente, cuando están ligadas e inactivas en el chip. Generalmente, **RESET** y **NMI** son conectadas directamente a los pines del dispositivo. Las características del servicio de interrupción incluyen:

- El pin **IACK** del CPU es usado para confirmar la recepción de una petición de interrupción.
- Los pines **INUM0 INUM3** indican el vector de interrupción que esta siendo utilizado.
- Los vectores de interrupción son reubicables.
- Los vectores de interrupción consisten de un paquete *fetch*. Con los paquetes se proporciona un rápido servicio.

## C.4. Periféricos

Los periféricos disponibles en los dispositivos TMS320C6000 se muestran en el cuadro (C.7). Los periféricos que son accesibles al usuario se configuran con un conjunto de registros de control delimitados en memoria. El controlador del bus de periféricos realiza el arbitraje para el acceso a los periféricos. La lógica de configuración de Boot esta conectada por señales externas y la lógica de baja energía es accesible directamente por el CPU. La Figura C.3 muestra un diagrama de bloques, con los periféricos de los dispositivos 'C6211/'C6711.

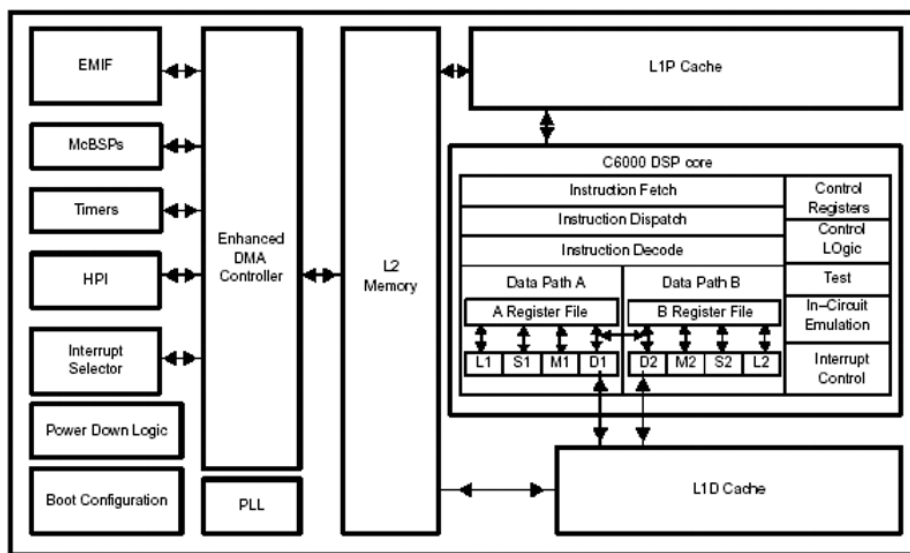


Figura C.3: Diagrama de bloques con los periféricos de los dispositivos 'C6211/'C6711.

1. **Controlador DMA.** El controlador DMA transfiere datos entre rangos de direccionamiento en el mapa de memoria, sin intervención del CPU. El controlador DMA tiene cuatro canales programables y cinco canales auxiliares.
2. **Controlador EDMA.** El controlador EDMA mejora las mismas funciones del controlador DMA. El EDMA tiene dieciséis canales programables, así como un espacio de RAM para soportar múltiples configuraciones de futuras transferencias.
3. **HPI.** El HPI es un puerto paralelo por medio del cual, un procesador host puede acceder directamente al espacio en memoria del CPU. El dispositivo host tiene facilidad de acceso debido a que es el maestro de la interfase. El host y el CPU pueden intercambiar información a través de la memoria interna o externa. En suma, el host tiene acceso directo a los periféricos de memoria mapeada.

Periféricos	C6201	C6202	C6211	C6701	C6711
Controlador de acceso directo a memoria (DMA)	Y	Y	N	Y	N
Controlador de acceso directo a memoria mejorado (EDMA)	N	N	Y	N	Y
Interfase al Puerto Host (HPI)	Y	N	Y	Y	Y
Bus de Expansión	N	Y	N	N	N
Interfase de Memoria Externa (EMIF)	Y	Y	Y	Y	Y
Configuración del Boot	Y	Y	Y	Y	Y
Puertos Seriales de Multicanal (McBSPs)	2	3	2	2	2
Selector de Interrupción	Y	Y	Y	Y	Y
Timers de 32 - bits	2	2	2	2	2
Lógica de Energía Baja	Y	Y	Y	Y	Y

Cuadro C.7: Periféricos de los dispositivos TMS320C6000

4. **Bus de expansión.** El bus de expansión es un reemplazo de para el HPI, así como una expansión del EMIF. La expansión proporciona dos áreas distintas de funcionalidad, (puerto host y puertos de Entrada/Salida) que pueden coexistir en un sistema. El puerto host del bus de expansión puede operar en modo asíncrono de forma (esclavo), similar al HPI o en modo síncrono (maestro/esclavo). Estos modos permiten la interfase de dispositivos para una variedad de protocolos del bus host. FIFOs síncronos y los dispositivos periféricos de Entrada/Salida asíncronos pueden conectarse al bus de expansión.
5. **EMIF.** El EMIF soporta una interfaz de baja adherencia (*glueless*) para varios dispositivos externos incluye:
  - a) SDRAM de ráfaga síncrona (SBSRAM).
  - b) DRAM síncrona (SDRAM).
  - c) Dispositivos asíncronos, incluyendo SRAM,ROM y FIFOs.
  - d) Dispositivo externo de memoria compartida.
6. **ConFiguración del Boot.** Los dispositivos TMS320C62x/C67x proporcionan una variedad de configuraciones del boot, que determinan las acciones de inicialización que ejecuta el DSP, después del reset del dispositivo. Estas incluyen: cargas de código de un espacio externo de ROM sobre el EMIF y cargas de código a través del HPI/bus de un host externo.

7. **McBSP.** El puerto serial multicanal con buffer (McBSP) esta basado en las interfaces estándar del puerto serie, encontrada en los dispositivos con plataformas TMS320C2000 y 'C5000. Resumiendo, el puerto puede almacenar muestras seriales en un buffer de memoria automáticamente, con la ayuda del controlador DMA/EDMA. Este también tiene capacidad de multicanal, compatible con los estándares de conexión a redes T1, E1, SCSA y MVIP. Proporciona:
- a) Comunicación full-Duplex.
  - b) Registros de datos de doble buffer para flujo continuo de datos.
  - c) Tramado independiente y temporización para dispositivos y transmisión.
  - d) Interfase directa a codecs estándar, chips de interfase analógica (AICs) y otros dispositivos A/D y D/A conectados en serie.
8. Tiene las siguientes capacidades:
- a) Interface directa a:
    - 1) Tramas T1/E1.
    - 2) Dispositivos conforme a  $ST - BUS^{TM}$ .
    - 3) Dispositivos conforme a IOM - 2.
    - 4) Dispositivos conforme a AC97.
    - 5) Dispositivos conforme a IIS.
    - 6) Dispositivos  $SPI^{TM}$ .
  - b) Transmisión y recepción multicanal de 128 canales.
  - c) Un selector del ancho del tamaño del dato, que incluye 8, 12, 16, 20, 24 y 32 bits.
  - d) Ley -  $\mu$  y Ley - A de compasión.
  - e) Transferencia inicial de 8 bits con LSB(bit menos significativo) o MSB(bit más significativo).
  - f) Polaridad programable para ambas tramas de sincronización y relojes de datos.
  - g) Reloj interno altamente programable y generación de trama.
9. **TIMER.** Los dispositivos 'C6000 tienen dos timer de propósito general que son usados para:
- a) Eventos del timer.
  - b) Eventos del contador.

- c) Generador de pulsos.
  - d) Interrupción del CPU.
  - e) Enviar eventos de sincronización a el contador DMA/EDMA.
10. **Selector de Interrupción.** El conjunto de periféricos del 'C6000 producen 14 a 16 fuentes de interrupción. El CPU tiene 12 interrupciones disponibles. El selector de interrupción permite elegir entre las 12 interrupciones, la que necesita su sistema. El selector de interrupción, también permite cambiar la polaridad de entrada para la interrupción externa.
11. **Lógica de bajo consumo de energía.** La lógica de bajo consumo de energía permite reducir el reloj para disminuir el consumo de energía. La mayoría de la potencia de operación de la lógica CMOS, se disipa durante la conmutación del circuito de un estado lógico a otro. Para prevenir algo o toda la lógica del chip de conmutación, se pueden realizar ahorros significativos de energía, sin perder datos ni contexto operacional.[27]

# Apéndice D

## Programas realizados en el DSP y en MatLab

### D.1. Desarrollo de programas en MatLab

A continuación se muestra el código en MatLab (realizado por efectos de calidad de imagen) para visualizar cada una de las imágenes originales:

MATLAB D.1: Visualización de la señal.
--

<pre>GPS=EmSEBDCE0450Hzrms(:,1); Microns=EmSEBDCE0450Hzrms(:,2); plot(GPS,Microns);</pre>
---

El programa realiza lo siguiente:

- la primera línea del programa realiza la carga de los datos contenidos en la primer columna de cada archivo .dat (magnitud en GPS)
- la segunda, carga los datos de la segunda, tercera, columna (tiempo)
- y por último, este comando permite ver la gráfica de la señal.

Los espectros de frecuencia de las señales analizadas fueron obtenidos a partir del código siguiente:



MATLAB D.2: Programa para obtener el espectro de frecuencia de una señal.
---

```

GPS=EmSEBDCE0450Hzrms(:,1);
Microns=EmSEBDCE0450Hzrms(:,2);
plot(GPS,Microns);
t=8.455:0.001:8.4923;
x=\sin(2*pi*1*t) + \sin(2*pi*15*t);
y=x + 2*randn(size(t));
plot(100000000 *t(1:38),y(1:38))
Y=fft(y,6200);
Pyy=Y.*conj(Y)/6200;
f=1000*(0:3100)/6200;
plot(f,Pyy(1:3101));

```

La obtención de los espectros de frecuencia se realiza de la siguiente manera:

- las primeras tres líneas realizan las funciones mencionadas anteriormente
- cuarta línea; se delimita el rango de tiempo, con incrementos de 0.001
- séptima línea; escalamiento
- octava línea; comando para obtener la FFT tomando las 6200 muestras
- novena línea; a partir del número de muestras se obtiene una frecuencia normalizada
- décima línea; gráfica del espectro de la señal.

## D.2. Formato de archivos .dat

El ambiente de desarrollo Code Composer Studio, lee y escribe datos utilizando un formato de archivos en específico. Un archivo de texto que contiene una línea de encabezado con la siguiente sintaxis:

1. Número mágico: valor fijo de 1651.
2. Formato: un número del 1 al 4, el cuál indica el formato de las muestras en el archivo (hexadecimal, entero, largo o flotante).
3. Dirección de inicio: la dirección de comienzo del bloque que está guardado.
4. Número de página: el número de página del bloque de donde se toman los datos.
5. Longitud: el número de muestras en el bloque.

Los datos almacenados como una muestra por renglón; pueden ser de alguno de los siguientes formatos:

- Hexadecimal
- Entero
- Largo
- Flotante

A continuación se muestra un ejemplo de como se almacenan las muestras de una señal en un archivo .dat:

```
1651  1  800  1  10
```

```
0x0000
```

```
0x0000
```

```
0x0000
```

```
0x0000
```

```
0x0000
```

```
0x0000
```

```
0x0000
```

```
0x0000
```

Particularmente, la sintaxis de las señales sísmicas empleadas es:

```
1651  2  800  1  6200
```

```
1.548e - 01
```

```
7.947e - 02
```

```
6.426e - 02
```

```
8.617e - 02
```

```
8.734e - 02
```

```
1.122e - 01
```

```
1.151e - 01
```

```
7.712e - 02
```

### D.3. Implementación de programas en el DSP

El procesamiento se realizó en tiempo real sobre el DSP, con el uso del Software Code Composer Studio ( CCS ) versión 2.1, para dicho proceso se creó un proyecto dentro del ambiente de simulación, en este proyecto se incluyen librerías propias para el manejo del DSP, así como algunas desarrolladas para el funcionamiento de los algoritmos, el código fuente fue desarrollado en lenguaje C desde donde se hace uso de las librerías y procedimientos en ensamblador. Este software, permite visualizar el código C y el de ensamblador permitiendo así, elegir el lenguaje de programación más cómodo.

En el proyecto credo se encuentran archivos de programa necesarios para el procesamiento de las señales sísmicas, los cuales son descritos a continuación:

#### wavtodo.h

Esta, es una librería desarrollada por Texas Instruments y es la idónea para las necesidades de procesamiento.

```
/* DSP BIOS 4.60.22 12-07-01 */
/* ===== wavtodo.h ===== */
#ifndef \_ \_ WAVELET\_H
#define \_ \_ WAVELET\_H
#ifndef TRUE
#define TRUE 1
#endif
#define BUFSIZE 0x3E8 //0x3E8 = 1000
#define FRAMESPERBUFFER 10
#define MINGAIN 1
#define MAXGAIN 10
#define MINCONTROL 0
#define MAXCONTROL 19
#define BASELOAD 1
struct PARMS \{
    int Beta;
    int EchoPower;
    int ErrorPower;
    int Ratio;
    struct PARMS *Link;
};
#endif /* \_ \_ WAVELET\_H */
```

**load.asm**

Este archivo contiene una rutina de carga, un simple ciclo (loop) que es invocado desde lenguaje C con un argumento.

```

; "@(#) DSP/BIOS 4.60.22 12-07-01 (barracuda-j15)"
;
; ===== load.asm =====
;   .ref _load
;   .text
N   .set 1000
;
; ===== _load =====
; This function simulates a load on the DSP by executing N * loopCount
; instructions, where loopCount is the input parameter to load().
;
;   void _load(int loopCount)
;
; The loop is using 8 instructions. One instruction for sub, nop and
; b, plus nop 5. The extra nop added after sub is to make the number
; of instructions in the loop a power of 2.
; _load:
;       mv a4, b0      ; use b0 as loop counter
[!b0]  b lend
;       mvk N,b1
;       mpy b1,b0,b0
;       nop
;       shru b0,3,b0   ; (loop counter)= (# loops)/8
loop:
;       sub b0,1,b0
;       nop
[b0]   b loop
;       nop 5
lend:  b b3
;       nop 5      ; return
;       .end

```

**vectors.asm**

```

; "@(\#) DSP/BIOS 4.60.22 12-07-01 (barracuda-j15)"
; ===== vectors.asm =====
; Plug in the entry point at RESET in the interrupt vector table
;
; ===== unused =====
; plug infinite loop -- with nested branches to
; disable interrupts -- for all undefined vectors
;
unused .macro id
    .global unused:id:
unused:id:
    b unused:id: ; nested branches to block interrupts
    nop 4
    b unused:id:
    nop
    nop
    nop
    nop
    nop
    .endm
    .sect ".vectors"
    .ref \_c\_int00    ; C entry point
    .align 32*8*4     ; must be aligned on 256 word boundary
RESET:                ; reset vector
    mvkl \_c\_int00,b0 ; load destination function address to b0
    mvkh \_c\_int00,b0
    b b0              ; start branch to destination function
    mvc PCE1,b0       ; address of interrupt vectors
    mvc b0,ISTP       ; set table to point here
    nop 3             ; fill delay slot
    nop
    nop

; plug unused interrupts with infinite loops to
; catch stray interrupts
unused 1
unused 2
unused 3
unused 4

```

```
unused 5
unused 6
unused 7
unused 8
unused 9
unused 10
unused 11
unused 12
unused 13
unused 14
unused 15
```

Los archivos fuentes *asm* contienen instrucciones de ensamble necesarias para ajustar el interruptor RESET para ramificar el programa C de punto de entrada.

#### **wavelet.c**

```
/****** wavelet.c *****/
#include <stdio.h>
#include "wavtodo.h"
/* Declaraciones Globales */
float inp\_buffer[BUFSIZE]; /* buffers para el procesamiento de datos*/
float out\_buffer[BUFSIZE];

/* VARIABLES UTILIZADAS EN LA DWT */
float SISMO[BUFSIZE];
float SISMO[BUFSIZE];
float AuxLow[BUFSIZE];
float AuxHigh[BUFSIZE];
float AuxSenal[BUFSIZE + 30];
int Cont,p,NumCoef,nC,n2C,nDes;
//Arreglos para coeficientes de Daubechies 10
float CoefL[20],CoefH[20];
float Rconv,AuxCoef,AuxSignal;
int CALH,CAS;
```

```

/* VARIABLES UTILIZADAS EN EL FILTRO */
float n2,n3,n4,n6,n7,n8,AuxInt,AuxOut;
float g1,g2,g3,g4,g5,g6,g7,r1;

/* Carga valor para rutina de procesamiento */
unsigned int processingLoad = BASELOAD;
struct PARMS str =
{
  2934,
  9432,
  213,
  9432,
  \&str
};

/* Funciones */
extern void load(unsigned int loadValue);
static int processing(float *input, float *output);
static void dataIO(void);
void DWT();
/* ===== main ===== */
void main()
\{
float *input = \&inp\_buffer[0];
float *output = \&out\_buffer[0];

/* Ciclo constante */
while(TRUE)
{
  /* Lee datos de entrada utilizando un probe-point
  conectado al Archivo Señal1.dat. */
  /* Escribe datos de salida a las graficas a trav'es
  del probe-point.*/
  dataIO();
  /* Procesamiento de la se~nal */
  processing(input, output);
  DWT();
}
}

```

```
/*
 * ===== processing =====
 * Funci\on : Aplica el algoritmo de filtrado a la se\~nal.
 * Par\ametros : Direcciones de los buffers de entrada y salida.
 * Valor Regresado: TRUE.

static int processing(float *input, float *output)
{
    int size = BUFSIZE;
    n2=n3=n4=n6=n7=n8=0;
    g1 = 0.17843;
    g2 = 0.53529;
    g3 = 0.53529;
    g4 = 0.17843;
    g5 = -0.084523;
    g6 = -0.33522;
    g7 = -0.0076958;

    while(size--)
    {
        AuxInt = *input++;
        r1 = AuxInt*g1 + n2*g2 + n3*g3 + n4*g4;
        AuxOut = *output++ = r1 + n6*g5 + n7*g6 + n8*g7;
        n4 = n3;
        n3 = n2;
        n2 = AuxInt;
        n8 = n7;
        n7 = n6;
        n6 = AuxOut;
    }
    /* Procesamiento adicional */
    load(processingLoad);
    return(TRUE);
}

void DWT()
{
    int size = BUFSIZE;
```



```

float *sis\_low = \&SISMOL[0];
float *sis\_high = \&SISMOH[0];
CALH = 0;
CAS = 0;
while(size--)
\{
    AuxSenal[CAS] = out\_buffer[CAS];
    AuxLow [CALH] = 100.0;
    AuxHigh[CALH] = 0.0;
    CAS++;
    CALH++;
\}
//Coeficientes de daubechies 10
NumCoef = 20;
CoefL[0]=0.0189;      CoefL[1]=0.1331;
CoefL[2]=0.3728;      CoefL[3]=0.4868;
CoefL[4]=0.1988;      CoefL[5]=-0.1767;
CoefL[6]=-0.1386;     CoefL[7]=0.0901;
CoefL[8]=0.0658;      CoefL[9]=-0.0505;
CoefL[10]=-0.0208;    CoefL[11]=0.0235;
CoefL[12]=0.0026;     CoefL[13]=-0.0076;
CoefL[14]=0.0010;     CoefL[15]=0.0014;
CoefL[16]=-0.0005;    CoefL[17]=-0.0001;
CoefL[18]=0.0001;     CoefL[19]=-0.0000;
n2C = 19;
for (nC=0;nC<=$NumCoef;nC++)
\{
    CoefH[nC]=CoefL[n2C];
    if (nC \% 2) // Modulo
        CoefH[nC] = CoefH[nC] * (-1); //Si es impar se cambia el signo
    AuxSenal[CAS++] = 0;
    n2C--;
\}
CAS = BUFSIZE + 20;
for(nDes = 0; nDes <$ 4; nDes++) // Descomposiciones
\{
    Cont = 0;
    // Low Pass

```

```
for (p=0;p<CAS;p+=2) //CAMBIO
{
    Rconv=0;
    for (nC=0;nC<=NumCoef;nC++)
    {
        if ((p - nC) >= 0)
        {
            AuxCoef = CoefL[nC];
            AuxSignal = AuxSenal[p-nC];
            Rconv = Rconv + ( AuxCoef * AuxSignal );
        }
    }
    AuxLow[Cont] = Rconv;
    Cont = Cont + 1;
}
Cont = 0;
// High Pass
for (p=0;p<CAS;p+=2)
{
    Rconv=0;
    for (nC=0;nC<=NumCoef;nC++)
    {
        if ((p - nC) >= 0)
        {
            AuxCoef = CoefH[nC];
            AuxSignal = AuxSenal[p-nC];
            Rconv = Rconv + ( AuxCoef * AuxSignal );
        }
    }
    AuxHigh[Cont] = Rconv;
    Cont = Cont + 1;
}
// Se suprimen detalles
for(p=0; p<Cont; p++)
{
    AuxSenal[p] = AuxLow[p];
}
CAS = Cont;
} // Fin de Descomposiciones
```

```
size = Cont;
CAS = 0;
CALH = 0;
while(size--)
{
    *sis\_low++ = AuxLow[CALH];
    *sis\_high++ = AuxHigh[CALH];
    CBufEcg++;
    CAS++;
    CALH++;
}
} // Fin de DWT
```

**Nota:** El archivo wavelet.c también puede ser visualizado por el software Code Composer Studio con un lenguaje ensamblador.