

Impreso el 20 de marzo de 2009
Reportar errores y sugerencias a:
d.milone@ieee.org
lrufiner@bioingenieria.edu.ar

Introducción a las Señales y los Sistemas Discretos

Diego H. Milone
Hugo L. Rufiner
Rubén C. Acevedo
Leandro E. Di Persia
Humberto M. Torres

Índice general

Prólogo	9
Prefacio	11
1. Introducción a señales	15
1.1. Introducción	16
1.2. Clasificación de las señales	18
1.2.1. Clasificación Fenomenológica	20
1.2.2. Clasificación Morfológica	22
1.3. Ruido en señales	25
1.4. Teoría de la comunicación	27
1.4.1. Teoría de la señal	27
1.4.2. Teoría de la información y de la codificación	28
1.5. Procesamiento de señales	29
1.5.1. Análisis de Señales	32
1.6. Operaciones elementales con señales	35
1.6.1. Operaciones unarias	35
1.6.2. Operaciones binarias	37
1.7. Preguntas	37
1.8. Trabajos prácticos	40
Bibliografía	42
2. Espacios de señales	43
2.1. Introducción	44
2.1.1. Desarrollo intuitivo	44
2.2. Señales, vectores y álgebra lineal	46
2.2.1. Normas	46
2.2.2. Producto interno	48
2.3. Espacios vectoriales y señales	52

2.3.1.	Conjunto de señales	52
2.3.2.	Espacios de señales	53
2.3.3.	Espacios vectoriales	54
2.4.	Bases y transformaciones	56
2.4.1.	Dependencia lineal y conjuntos generadores	56
2.4.2.	Bases	57
2.4.3.	Ortogonalidad y ortonormalidad	58
2.4.4.	Aproximación de señales	59
2.4.5.	Cambio de base	63
2.4.6.	Transformaciones lineales	67
2.5.	Preguntas	68
2.6.	Trabajos prácticos	70
	Bibliografía	73
3.	Transformada Discreta de Fourier	75
3.1.	Introducción	76
3.2.	Familia de bases de Fourier	77
3.2.1.	Series seno	77
3.2.2.	Series coseno	77
3.2.3.	Serie exponencial de Fourier	79
3.2.4.	Transformada de Fourier de tiempo discreto	80
3.2.5.	Transformada Continua de Fourier	81
3.3.	Exponenciales complejas discretas	81
3.4.	Transformada Discreta de Fourier	86
3.5.	Propiedades de la TDF	87
3.6.	Relación entre la TCF y la TDF	89
3.7.	Utilización de ventanas	92
3.8.	Resolución temporal y frecuencial	94
3.9.	Representación matricial de la TDF	97
3.10.	Transformada Rápida de Fourier	99
3.11.	Preguntas	102
3.12.	Trabajos prácticos	105
	Bibliografía	108
4.	Introducción a sistemas	109
4.1.	Introducción	110
4.2.	Interconexión de sistemas	110
4.3.	Propiedades de los sistemas	112
4.4.	Ecuaciones en diferencias	114

4.5. Representación de sistemas LTI discretos	116
4.6. Preguntas	118
4.7. Trabajos prácticos	118
Bibliografía	122
5. Convolución discreta	123
5.1. Introducción	124
5.2. Convolución lineal	124
5.3. Convolución circular	130
5.4. Relación entre convolución lineal y circular	130
5.5. Deconvolución	132
5.6. Preguntas	136
5.7. Trabajos prácticos	136
Bibliografía	139
6. Transformada Z	141
6.1. Introducción	142
6.2. Definición de Transformada Z	142
6.2.1. Convergencia de la Transformada Z	144
6.2.2. La Transformada Z inversa	146
6.3. Propiedades de la Transformada Z	147
6.4. Representación de sistemas discretos mediante la Transformada Z	148
6.4.1. Transformación de Euler	150
6.4.2. Transformación bilineal	153
6.5. Trabajos prácticos	155
Bibliografía	158
7. Identificación de sistemas mediante predicción lineal	159
7.1. Introducción	160
7.1.1. Técnicas convencionales	161
7.1.2. Técnicas no convencionales	161
7.2. Análisis de la respuesta para sistemas continuos	162
7.3. Métodos de predicción lineal	164
7.3.1. El modelo ARMA	164
7.3.2. El modelo AR	164
7.3.3. Cuadrados mínimos	165
7.3.4. Sistema de Wiener-Hopf para señales determinísticas	167
7.3.5. Sistema de Wiener-Hopf para señales aleatorias	169

7.3.6. Resolución del sistema de Wiener-Hopf	170
7.3.7. Determinación de la constante de ganancia G	173
7.4. Estimación del orden	173
7.4.1. Error de predicción final	174
7.4.2. Criterio de Akaike	175
7.5. Preguntas	175
7.6. Trabajos prácticos	176
Bibliografía	180
8. Identificación de sistemas no lineales mediante algoritmos genéticos	183
8.1. Introducción	184
8.2. Estructura de un AG	184
8.3. Diseño de la solución de un problema mediante AGs	186
8.4. Representación de los individuos	187
8.5. Función de fitness	188
8.6. Selección	189
8.6.1. Rueda de ruleta	189
8.6.2. Ventanas	190
8.6.3. Competencias	190
8.7. Reproducción y operadores de variación	190
8.7.1. Mutaciones	191
8.7.2. Cruzas	191
8.8. Características principales	192
8.9. Introducción a los fundamentos matemáticos	195
8.10. Trabajos prácticos	196
Bibliografía	201
A. Octave (v2.1.36)	203
A.1. Generales	203
A.2. Comandos del sistema	203
A.3. Matrices y rangos	203
A.4. Algunas variables predefinidas	204
A.5. Operaciones aritméticas y operadores de incremento	204
A.6. Operadores booleanos y de comparación	204
A.7. Sentencias	205
A.8. Manipulaciones básicas de matrices	205
A.9. Funciones trigonométricas	206
A.10. Álgebra Lineal	206

A.11. Procesamiento de Señales	206
A.12. Procesamiento de Imágenes	206
A.13. Funciones de entrada/salida	206
A.14. Misceláneas	207
A.15. Polinomios	207
A.16. Estadística	207
A.17. Gráficos básicos	207
A.18. Otras funciones de graficación	208
B. Comandos de SciLab (v2.6)	209
B.1. Señales	209
B.2. Sonido	210
B.3. Control	210
B.4. Funciones elementales	210
B.5. E/S a archivos	212
B.6. Creación de funciones	212
B.7. Gráficos	213
B.8. Operaciones lineales de matrices	214
B.9. No lineal	214
B.10. Polinomios	214
B.11. Programación	214
B.12. Cadenas de caracteres	215
B.13. Utilidades	215
C. Comandos de MatLab (v4.2)	217
C.1. Comandos de propósito general	217
C.1.1. Comandos básicos	217
C.1.2. Manejo de variables y del espacio de trabajo	217
C.1.3. Trabajo con archivos y con el sistema operativo	217
C.1.4. Control de la ventana de comandos	218
C.1.5. Iniciando y terminando MatLab	218
C.2. Gráficos bidimensionales	218
C.2.1. Gráficos X-Y elementales	218
C.2.2. Gráficos X-Y especializados	218
C.2.3. Referenciación de gráficos	218
C.3. Análisis de señales	219
C.3.1. Operaciones básicas	219
C.3.2. Operaciones con vectores	219
C.3.3. Correlación	219

C.3.4. Filtrado y convolución	219
C.3.5. Transformadas de Fourier	219
C.4. Funciones mat. elementales	220
C.4.1. Trigonométricas	220
C.4.2. Exponenciales	220
C.4.3. Complejos	220
C.4.4. Numéricas	220
C.5. Matrices elementales y manipulación de matrices	221
C.5.1. Matrices elementales	221
C.5.2. Constantes y variables especiales	221
C.5.3. Horas y fechas	221
C.5.4. Manipulación de matrices	221
C.6. Toolbox de procesamiento de señales	222
C.6.1. Generación de formas de onda	222
C.6.2. Análisis e implementación de filtros	222
C.6.3. Diseño de filtros digitales IIR	222
C.6.4. Selección del orden de filtros IIR	222
C.6.5. Diseño de filtros FIR	222
C.6.6. Transformadas	223
C.6.7. Procesamiento estadístico de señales	223
C.6.8. Ventanas	223
C.6.9. Modelado paramétrico	223
C.6.10. Operaciones especializadas	223
C.6.11. Prototipos de filtros analógicos pasabajo	224
C.6.12. Traslaciones en frecuencia	224
C.6.13. Discretización de filtros	224

Prólogo de la primera edición

Escribir la presentación de un libro suele ser una empresa de gran responsabilidad y en este caso se suma el hecho de que uno de los autores (H.L.R.) ha sido uno de mis mejores alumnos. No obstante, esta responsabilidad ha sido facilitada por la excepcional calidad de esta obra que, además de ser clara y concisa, posee como valor principal el de mostrar las técnicas de procesamiento de señales y de estudio de sistemas que han constituido la base para el desarrollo de la tecnología actual. La misma ha revolucionado nuestra sociedad y continúa evolucionando rápidamente, principalmente en el campo digital.

Resulta sorprendente descubrir que todos los seres vivos realizan, sin sospecharlo, análisis de señales similares a los aquí tratados. Por ejemplo, tanto una planta carnívora como un tigre pueden distinguir fácilmente sus respectivas presas, ya sea por su olor, su aspecto, sus movimientos o sus sonidos característicos. Sin embargo sólo nuestra especie ha podido comprender estos fenómenos y describirlos en términos de un lenguaje matemático. Una paloma puede distinguir, con mayor precisión que nosotros, diferencias sutiles en el canto de otras aves de su propia especie, pero el hombre es el único que puede preguntarse el por qué y el cómo de todos estos procesos. Podríamos decir que varios de ellos involucran técnicas semejantes al análisis de Fourier, no obstante sólo nosotros somos capaces de plantear un modelo o teoría como ésta que permita estudiarlos.

Además, a medida que descubrimos sus limitaciones, podemos proponer nuevas teorías para mejorar nuestro entendimiento del mundo.

En este libro se revisan los fundamentos de diferentes métodos para el estudio de los sistemas y la extracción de información útil de las señales, aplicables en campos tan disímiles como la biología, la geología, la economía o la astronomía, entre muchos otros.

Mis felicitaciones a los autores, que hago extensivas a la editorial de la Universidad Nacional de Entre Ríos, por haber realizado este esfuerzo, cada vez más raro en estos tiempos:

Luis F. Rocha

Prefacio

Se puede decir que los conceptos de señal y sistema permiten encarar el estudio de cualquier problema del mundo físico mediante un *modelo* adecuado de la realidad. Estos modelos se han difundido enormemente en los tiempos actuales, debido principalmente a las bases matemáticas de la teoría de la comunicación y los avances en el área informática que han permitido llevar las soluciones al campo digital, invadiendo casi todas las actividades de la sociedad moderna.

Este libro pretende brindar una breve introducción a los fundamentos de esta teoría para *comprender* el mundo. El mismo es el fruto de unos diez años de trabajo impartiendo cursos relacionados con el tema, y surgió originalmente como una necesaria introducción para un curso de modelización de sistemas biológicos, en la carrera de grado en Bioingeniería (o Ingeniería Biomédica), de la Facultad de Ingeniería de la Universidad Nacional de Entre Ríos, Argentina. Más recientemente, el material fue ampliado y utilizado como primer bloque de fundamentos de un curso de procesamiento digital de señales en la carrera de grado en Ingeniería Informática, en la Facultad de Ingeniería y Ciencias Hídricas de la Universidad Nacional del Litoral.

Se supone que el lector cuenta con algunas nociones básicas de física, matemática e informática. Las nociones de física aportan la base conceptual para transcribir la realidad concreta a una versión abstracta y simplificada de la misma. Como requisito, el lector deberá tener conocimientos de física elemental para entender la aplicación de los conceptos mostrados mediante algunos ejemplos simples de sistemas mecánicos y eléctricos. Las nociones de matemática aportan la base formal para la descripción de las señales y sistemas discretos. Se presupone que el lector conoce los fundamentos de álgebra lineal, cálculo vectorial, ecuaciones diferenciales y variable compleja. Las nociones de informática permiten la implementación computacional de prácticamente *todo* lo que pretende transmitir este libro.

La informática es la herramienta básica utilizada para llevar a la práctica y terminar de comprender los detalles de cada tema. No se requiere un gran dominio de la computación sino más bien algunos conocimientos mínimos de programación como estructuras condicionales y repetitivas, vectores y matrices, subprogramas y graficación básica.

Se ha tratado de conservar parte del enfoque heredado del Ing. Luis F. Rocha, quien originalmente nos acercó estas ideas como parte introductoria de un curso denominado Bioingeniería I. También se ha tratado de rescatar el carácter universal de varias de las ideas planteadas en la teoría de señales y sistemas. Especialmente, los conceptos vertidos en el Capítulo 2, Espacio de señales, forman una visión genérica de un conjunto muy amplio de técnicas utilizadas en el procesamiento digital de señales. Ver a las señales como vectores, interpretar geoméricamente operaciones básicas como el producto interno en un cambio de base y extender su aplicación a las transformaciones lineales, constituye un núcleo conceptual que permite visualizar operaciones complejas desde una perspectiva muy simple pero a la vez genérica. Si bien aún restan algunos capítulos con un enfoque más bien clásico, es nuestra intención completar esta tarea en futuras ediciones intentando formar un único hilo conductor, conceptual y didáctico, a lo largo de todo el libro.

El libro está organizado de la siguiente forma. En el Capítulo 1 se presenta una introducción general al tema de señales, con ejemplos de varios campos de aplicación. Se discuten los conceptos de señal, sistema, información y ruido. En el Capítulo 2 se presenta el estudio de las señales, ya no de forma aislada, sino en el marco de conjuntos de señales que cumplen con determinadas propiedades de interés. Como anticipamos, este enfoque desde los espacios de señales permite sentar las bases para las transformaciones lineales y provee una perspectiva clave, en lo conceptual y didáctico, para reinterpretar el resto del material presentado en el libro. En el Capítulo 3, como una continuación natural y aplicación del capítulo anterior, se presenta la Transformada Discreta de Fourier. Se ha dedicado un capítulo entero a esta transformación debido al papel fundamental que juega actualmente en las aplicaciones. Se revisa la relación que existe entre todos los miembros de la familia de bases de Fourier y se presentan las ideas principales detrás de la Transformada Rápida de Fourier. En el Capítulo 4 se completan los conceptos sobre sistemas presentados en el primer capítulo, pero orientado principalmente a las propiedades de los sistemas discretos lineales e invariantes en el tiempo. El Capítulo 5 discute las ideas detrás de la sumatoria de convolución, mostrando su conexión natural con los sistemas lineales e invariantes en el tiempo. En el Capítulo 6 se presentan

los fundamentos de la Transformada Z, que juega para los sistemas de tiempo discreto un papel análogo al de la Transformada de Laplace para el caso de los sistemas de tiempo continuo. Los Capítulos 7 y 8 tratan tópicos especiales de aplicación a la identificación de sistemas. El Capítulo 7 presenta las nociones básicas de identificación de sistemas discretos lineales, tanto para el caso de sistemas invariantes como para los variantes en el tiempo. Finalmente, en el Capítulo 8 se presenta la técnica de algoritmos genéticos, que es un método general para la optimización y búsqueda de soluciones. En el contexto de este libro se presenta como una alternativa para la identificación de sistemas no lineales. Esto se debe a que, aunque el libro está orientado principalmente a sistemas lineales, los sistemas no lineales están apareciendo cada vez más en las aplicaciones.

Los conceptos se refuerzan en cada capítulo con una serie de preguntas y ejercicios prácticos que están pensados para desarrollarse en sesiones adicionales a las clases teóricas, que en nuestro esquema docente denominamos: clases de coloquio y prácticas de laboratorio. Las clases de coloquio están íntimamente relacionadas con los temas desarrollados en las clases teóricas, pero la modalidad de trabajo es más flexible, participativa y personalizada. Las preguntas pretenden ser una guía para la discusión de los aspectos relevantes de cada tema o aquellos que presentan mayor dificultad para el aprendizaje. Las actividades prácticas se centran en la resolución de problemas mediante un lenguaje de programación. Si bien no se sugiere ningún lenguaje en particular, los apéndices finales proveen una lista de comandos para algunos lenguajes de cálculo numérico muy utilizados en la actualidad. Los ejercicios prácticos que poseen un poco más de dificultad dentro de cada capítulo, o que necesitan un poco más de tiempo para resolverse, están marcados con (*), o con (**), de acuerdo con el grado de dificultad relativo. La bibliografía de consulta para cada tema se provee al final del libro, separada en bibliografía de carácter general y por cada capítulo.

Queremos agradecer las innumerables sugerencias aportadas por los alumnos, que podemos decir han sido los primeros revisores de este material. También debemos destacar el aporte sustancial de varias personas que han influido de diversas formas en los contenidos actuales de este libro. Entre ellas queremos agradecer especialmente a Daniel Zapata, con quien hemos mantenido extensas discusiones conceptuales acerca de varios de los temas incluidos. En la lista de agradecimientos contamos también al Ing. Carlos Montalvo, que fue responsable de varias de las notas iniciales tomadas de las clases del Ing. Rocha y a los Bioingenieros Carlos Pais y César Martínez, que han realizado revisiones de este material. Por último, queremos volver

a destacar la fuerte influencia del Ing. Rocha como formador y maestro de todos nosotros, pues modificó nuestra concepción de la ciencia y la ingeniería.

Diego H. Milone, Hugo L. Rufiner

Capítulo 1

Introducción a señales

Hugo Leonardo Rufiner

Temas a tratar

- Definiciones básicas de señales.
- Clasificación de las señales.
- Contexto de la teoría de la señal.
- Descripción de los procesamiento de señales más usuales.
- Operaciones elementales sobre y entre señales.

Objetivos

- Operar con señales discretas y reconocer las características y propiedades generales de las mismas.
- Aprender a aplicar en ejemplos sencillos las herramientas y conceptos en estudio.
- Generar y manipular señales digitales en forma de vectores por medio de un lenguaje de programación.

1.1. Introducción

La teoría de las comunicaciones ha invadido prácticamente nuestra vida diaria, con aplicaciones tecnológicas en campos tan diversos como el comercio, la medicina, la educación y la política, entre muchos otros. Los conceptos de *señal*, *sistema* e *información*, soportan esta teoría y están íntimamente relacionados. Los mismos poseen un carácter universal que permite describir prácticamente cualquier problema del mundo real en términos de estos conceptos, e inmediatamente tener a mano toda la teoría para intentar resolverlos.

Podemos decir que las señales transportan información acerca del sistema que las produjo, contenida o codificada en un patrón de variaciones de alguna magnitud física. Desde el punto de vista matemático las señales son descritas por medio de funciones, y los sistemas en términos de transformaciones. Estas transformaciones modifican a las denominadas *señales de entrada* para dar lugar a otras *señales de salida* del sistema. En este capítulo nos ocuparemos principalmente de las señales y dejaremos los aspectos específicos de los sistemas para tratarlos en el Capítulo 4. La descripción matemática de la cantidad de información contenida en una señal no será explorada de manera directa en este libro.

La palabra señal proviene del latín *signale*, que significa: marca que se pone o hay en una cosa para darla a conocer o distinguirla de otras. Otras acepciones tradicionales pueden ser: signo, imagen o representación de una cosa. Como hemos visto, una señal es un fenómeno que representa información. En general se consideran señales eléctricas, pero la teoría de la señal puede ser aplicada a cualquier clase de señal (lumínica, sonora, magnética, etc.), sin importar su naturaleza física. Pueden definirse infinidad de señales en otros campos como el económico, social, biomédico, etc.; por lo tanto las técnicas aquí desarrolladas son de aplicación general.

En la Figura 1.1 se muestra la señal de evolución del índice Merval en los años 2002 y 2003. Se puede apreciar una lenta recuperación del mercado argentino luego de la difícil situación económica de fines del año 2001. Con este ejemplo queda evidenciado cómo podemos extraer información útil acerca de un sistema determinado analizando las señales producidas por el mismo.

En el ámbito biomédico, las señales provenientes del registro a nivel de la piel de la actividad eléctrica del corazón (ECG), son de uso diario. Estas señales junto con otras como la de presión, constituyen parámetros básicos para análisis y control del estado del sistema cardiovascular. Por ello están

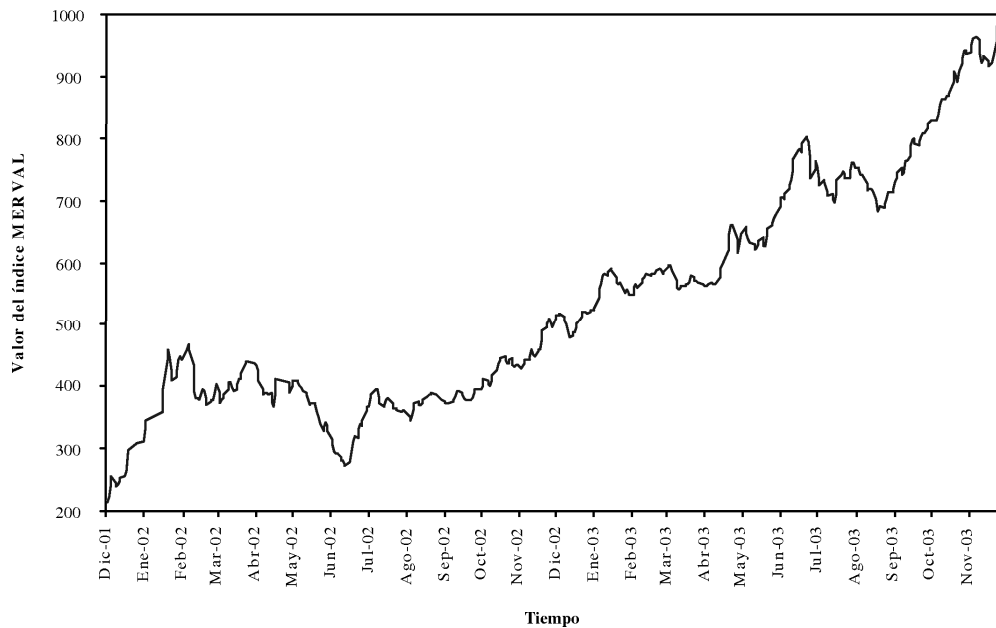


Figura 1.1. Evolución del índice argentino Merval entre 2002 y 2003.

incorporadas en los monitores de cabecera y otros instrumentos médicos (ver Figura 1.2).

Otro ejemplo de señal biológica puede ser la señal de voz, que es producida por el *aparato fonador* humano a través de un complicado mecanismo en el que intervienen varios órganos para modificar las propiedades acústicas del tracto vocal y de los estímulos sonoros implicados. De esta forma se producen los patrones de variación de la presión sonora característicos que constituyen la base de la comunicación humana. En la Figura 1.3 se puede apreciar el *sonograma* de una señal de voz proveniente de una frase del idioma inglés. En la misma pueden apreciarse marcas verticales



Figura 1.2. Señal de ECG tal como aparece registrada por un electrocardiógrafo de uso médico. El eje horizontal corresponde al tiempo y el vertical a la tensión.

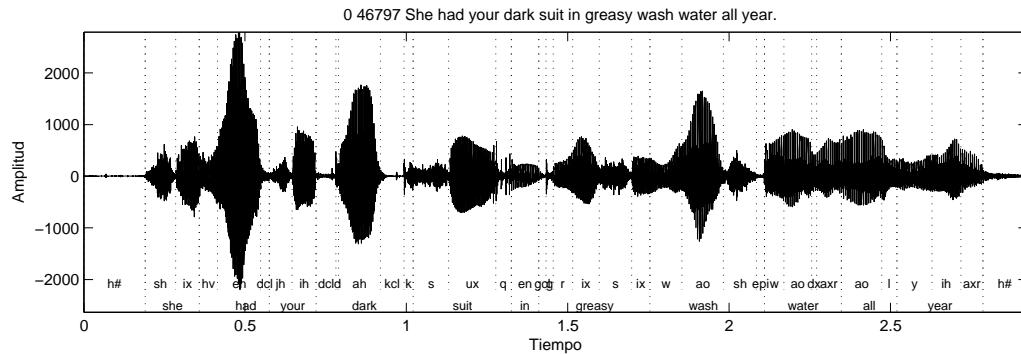


Figura 1.3. Señal de voz de una frase del idioma inglés (tomada de la base de datos TIMIT)

correspondientes a los trozos de la señal que representan distintas unidades acústico-fonéticas, como los fonemas y las palabras.

Generalmente, se toma como variable independiente al tiempo, aunque puede ser también alguna dimensión espacial, como en el caso de las imágenes (que constituyen señales bidimensionales). Un caso de particular interés son las imágenes médicas, en la Figura 1.4 se puede apreciar una imagen del cerebro producida por medio de la técnica de resonancia magnética. La representación de los valores tomados por alguna magnitud física en el espacio tridimensional constituye otro ejemplo. En la Figura 1.5 se pueden apreciar los contornos de presión para un fluido que pasa a través de una boquilla, simulados mediante la técnica de *elementos finitos*.

1.2. Clasificación de las señales

Como en cualquier ámbito, una adecuada taxonomía de los distintos tipos de señales permite el estudio sistemático de las mismas. Las señales se pueden clasificar de acuerdo a los siguientes criterios:

- Dimensional: basado en el número de variables independientes del modelo de la señal.
- Energético: de acuerdo a si poseen o no energía finita.
- Espectral: basado en la forma de la distribución de frecuencias del espectro de la señal.
- Fenomenológico: basado en el tipo de evolución de la señal, predefinido o aleatorio.



Figura 1.4. Imagen en tonos de grises proveniente de un estudio de resonancia magnética cerebral (National Institute of Health, U.S.A.)

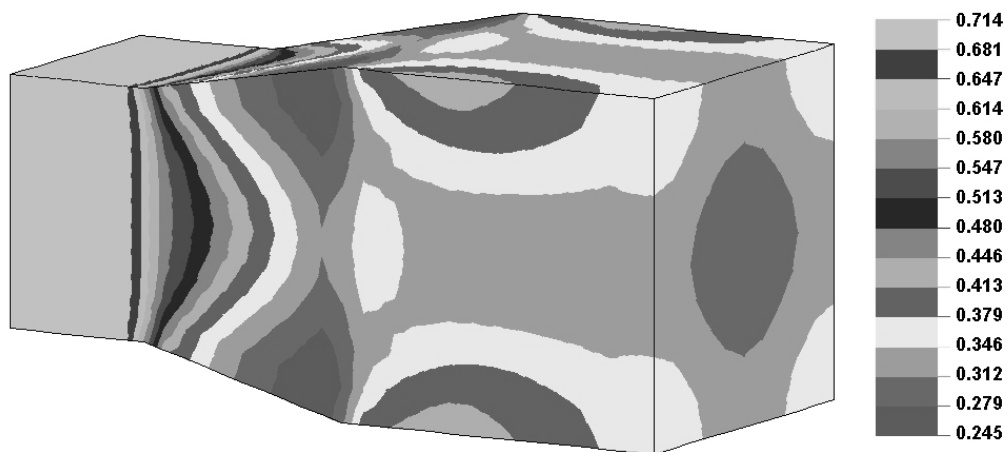


Figura 1.5. Contornos de presión para un fluido compresible que pasa a través de una boquilla cuadrada divergente (Engineering Mechanics Research Corporation, U.S.A.)

- Morfológico: basado en el carácter continuo o discreto de la amplitud de la señal o de la variable independiente.

Desarrollaremos a continuación principalmente las dos últimas, debido a la importancia de las mismas, y a que el resto resulta de la aplicación de conceptos sencillos ya conocidos o enunciados.

1.2.1. Clasificación Fenomenológica

En la Figura 1.6 se muestra en esquema de la clasificación fenomenológica cuyos elementos describiremos en las siguientes secciones.

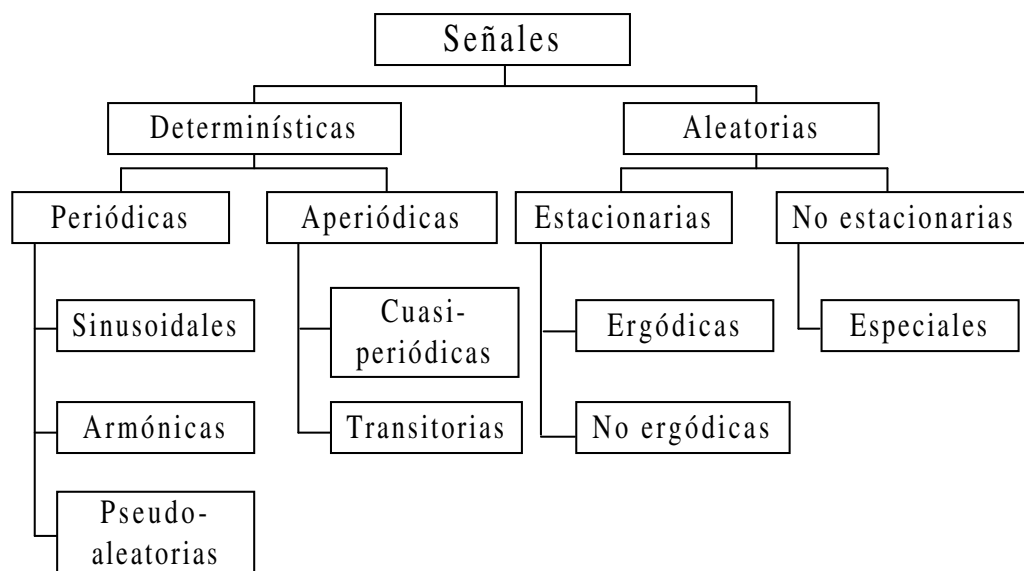


Figura 1.6. Clasificación *fenomenológica* de las señales

Señales determinísticas

Una señal se puede definir como *determinística* si sus valores son conocidos de antemano o pueden ser predichos exactamente. Por lo tanto, los próximos valores de una señal pueden ser determinados si son conocidas todas las condiciones anteriores de la señal. Así, ésta puede ser representada completamente por las ecuaciones que la definen.

A su vez, las señales determinísticas se pueden subdividir en periódicas y aperiódicas. Se dice que una señal continua es *periódica* si y sólo si

$x(t + T) = x(t)$ para todo $t \in (-\infty, \infty)$. El valor positivo más chico de T para el cual se cumple la ecuación anterior se llama período de la señal. Además, esta ecuación va a seguir siendo cierta si T es reemplazado por kT .

Cualquier señal que no es periódica se dice que es *aperiódica*. Algunas señales aperiódicas tienen propiedades únicas y son conocidas como funciones singulares, porque poseen derivadas discontinuas o son discontinuas ellas mismas. Entre estas señales se encuentra la tipo escalón, delta de Dirac, etc.

Otro tipo de señales que no poseen las propiedades que hemos nombrado anteriormente son las señales *transitorias*. Estas señales son aquellas que agotan su energía dentro del período de observación. Esta clasificación no depende tanto de la señal en sí, como de la escala temporal desde la cual se observa a la misma.

Señales estocásticas

Hay señales en las que existe casi siempre alguna *incerteza* acerca de los valores que puede tomar en los siguientes instantes. Estas señales son llamadas estocásticas o aleatorias y pueden ser descriptas solamente desde un punto de vista estadístico. Por ejemplo, se puede considerar que la señal de tensión del tendido eléctrico es determinística y hasta periódica, pero por otro lado, si se tienen en cuenta las pequeñas perturbaciones electromagnéticas esta misma señal puede ser considerada estocástica. Las señales aleatorias son más difíciles de manejar que las determinísticas, una señal con valores al azar es una realización de un proceso aleatorio. Una realización de un proceso aleatorio difiere de las otras en su descripción temporal, pero sin embargo poseen las mismas propiedades estadísticas.

Se puede dividir a las señales aleatorias en dos tipos: estacionarias y no estacionarias.

Un proceso aleatorio $\bar{X}(t)$ *estacionario* es aquel en el cual las propiedades estadísticas de la señal no varían con el tiempo. Por ejemplo, para un proceso de este tipo el valor esperado de $\bar{X}(t_i)$ a lo largo de todas las realizaciones sería igual para cualquier instante t_i . Dentro de este tipo de señales aleatorias se encuentran las de tipo *ergódicas*: para estas señales, las estadísticas a lo largo de una realización cualquiera son iguales a las estadísticas a lo largo de todas las realizaciones. Para clarificar estos conceptos se representa en la Figura 1.7 un proceso aleatorio $\bar{X}(t)$ de n realizaciones junto con las condiciones necesarias para establecer su estacionariedad y ergodicidad.

La estacionariedad y ergodicidad son propiedades que permiten el uso de métodos de procesamiento prácticos, un proceso que es no estacionario

(y por lo tanto no ergódico) es muy difícil de procesar.

$\bar{X}(t)$ = Proceso Aleatorio

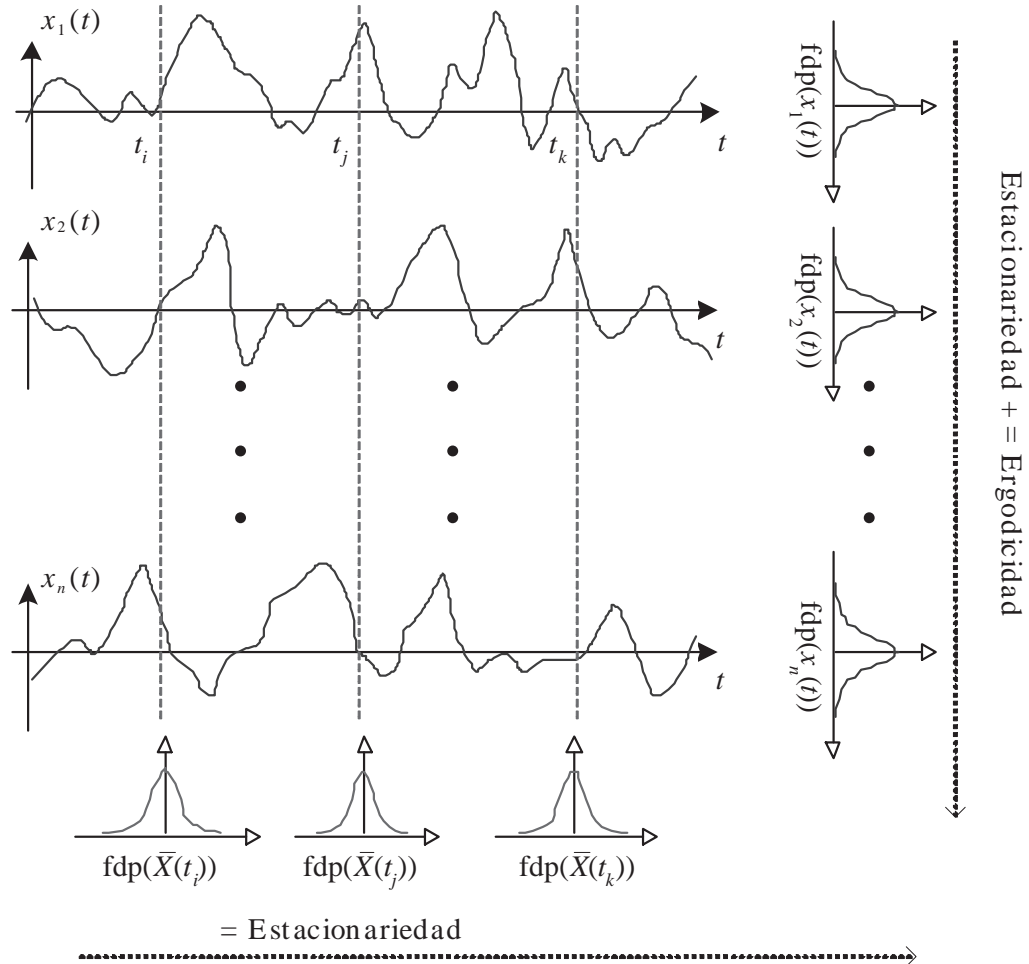


Figura 1.7. Esquema conceptual de un proceso aleatorio

1.2.2. Clasificación Morfológica

Señales continuas y discretas

Desde el punto de vista morfológico hay dos tipos básicos de señales: señales continuas y señales discretas. *En el caso de una señal continua la variable*

independiente es continua, por lo tanto estas señales están definidas para un continuo de valores de la variable independiente. Por otro lado, las señales discretas están definidas únicamente para una sucesión discreta de valores y consecuentemente para estas señales la variable independiente toma valores únicamente en el conjunto de los números enteros.

Para distinguir entre señales continuas y discretas normalmente se utiliza el símbolo t para denotar una variable continua y n para una variable discreta. Además, para señales de tiempo continuo se encerrará la variable independiente entre paréntesis (p.e. $y(t) = \text{sen}(\omega t)$), mientras que en el caso de las de tiempo discreto se encerrará entre corchetes (p.e. $y[n] = \text{sen}(nT)$).

Una señal discreta $x[n]$ puede representar un fenómeno para el cual la variable independiente es inherentemente discreta. Señales como estas pueden ser la relación especie-abundancia, o los datos demográficos tomados a determinados intervalos de tiempo. También las fotos en los diarios realmente consisten de una grilla de puntos muy fina y cada uno de estos puntos representa un muestreo del brillo del punto correspondiente de la imagen original. No importa el origen de los datos, de todas formas la señal $x[n]$ está definida únicamente para valores enteros de n .

En el caso en que la amplitud y la variable independiente sean continuas, entonces la señal es *analógica*; en cambio si la amplitud es discreta y la variable independiente también, la señal es *digital*.

Señales analógicas y digitales

La mayoría de las señales de interés en la naturaleza son analógicas. Sin embargo, es importante analizar las señales digitales, ya que la tecnología moderna en términos de software y hardware, hace al procesamiento en tiempo discreto más ventajoso que el procesamiento en tiempo continuo. Las ventajas son tales que normalmente es conveniente convertir la señal analógica en una digital de forma tal que se pueda llevar a cabo el procesamiento en tiempo discreto.

La conversión es llevada a cabo por sistemas de *conversión analógica a digital* (A/D), que muestrean, retienen cada muestra por un instante de tiempo, y cuantizan la señal en valores discretos. Normalmente el muestreo es llevado a cabo en instantes de tiempo uniformemente espaciados, sin embargo, también puede ser llevado a cabo un muestreo no-uniforme para tomar ventaja de algunas propiedades de la señal.

Las señales digitales son señales de tiempo discreto cuyos valores en amplitud son cuantizados. La salida de un conversor A/D, que muestrea

una señal de entrada continua y genera una secuencia de números binarios de longitud finita es una señal típicamente digital. En la parte superior de la Figura 1.8 se muestra un diagrama esquemático de un conversor A/D. Si se utiliza una frecuencia de muestreo determinada (por ejemplo 1 MHz, es decir una muestra cada $1 \mu s$.) el cuantizador tiene una relación entrada/salida como la de la parte inferior de la Figura 1.8. Si además se da una función continua $x(t)$ que tiene la forma que se ve en la Figura 1.9 (a), entonces las correspondientes señales en tiempo discreto $x_1(nT)$ y la señal digital de salida $x(nT)$, tomarían las formas representadas en la figuras (b) y (c).

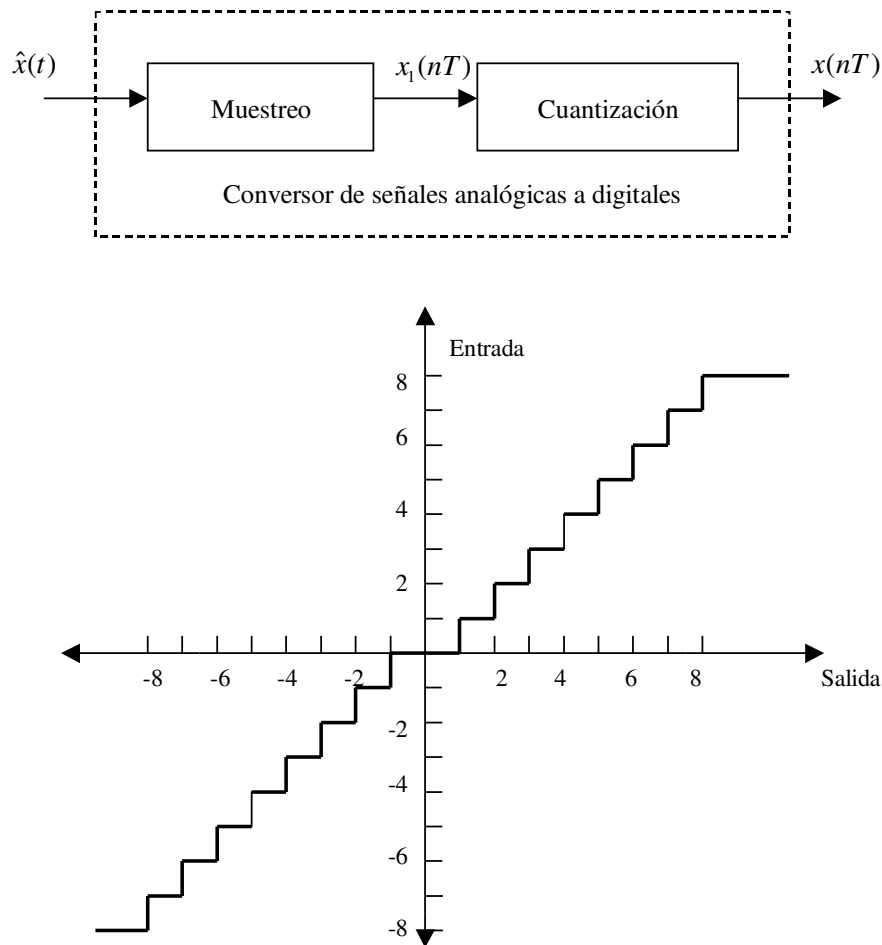


Figura 1.8. Conversor A/D y función de transferencia del cuantizador. Arriba: diagrama de bloques de un conversor A/D. Abajo: función de transferencia del cuantizador.

En términos estrictos, las computadoras pueden manejar únicamente

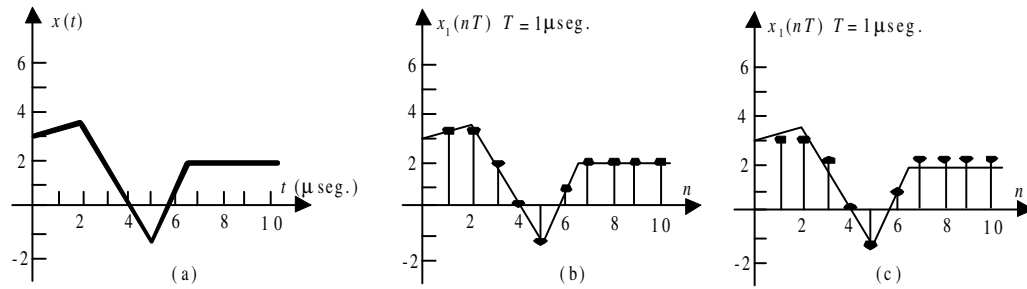


Figura 1.9. Señal continua, en tiempo discreto y digital. (a) Onda continua de entrada al sistema, (b) señal en tiempo discreto y (c) señal digital.

señales digitales, ya que las señales discretas pueden ser discretas en el tiempo pero pueden no serlo en amplitud. Como en una señal digital sólo hay un número finito de niveles, los errores están presentes en cualquier sistema que opere con este tipo de señales. Por lo tanto, una de las consideraciones de diseño de cualquier sistema que maneje señales digitales es el número de bits o el número de niveles de cuantización que se necesita para representar a la señal de una forma fidedigna. Cuanto más grande sea el número de bits usados, mayor va a ser la precisión en la representación de la señal, y más costoso va a ser el sistema digital.

Además de los efectos de la cuantización, el hecho de discretizar la señal mediante un conversor A/D, también puede introducir errores importantes en la señal resultante. Es fácil imaginarse que si muestreamos la señal a una velocidad más lenta que la de la mayor frecuencia presente en la señal podemos perder información importante. Esto puede producir cambios morfológicos significativos en la señal considerada. Este efecto, que conduce a una *confusión* acerca de cuáles son las frecuencias que componen la señal y se denomina *aliasing*, será tratado con detalle en el Capítulo 3.

1.3. Ruido en señales

Generalmente las señales están contaminadas con perturbaciones no deseadas que dificultan el análisis o proceso de la señal de interés, dichas perturbaciones se denominan ruido. Estrictamente, se denomina ruido a cualquier fenómeno o proceso (interferencia, distorsión aleatoria, etc.) que perturba la percepción o interpretación de una señal. Comparte la misma denominación que los efectos acústicos análogos y siempre está presente en la obtención de cualquier señal real.

Cuando se está en presencia de una señal contaminada con ruido se define

una medida de cuánto una señal esta contaminada por ruido, dicha medida se denomina relación señal-ruido (S/R o SNR). Ésta se define como la razón ξ entre la potencia de la señal P_s y la potencia del ruido P_r :

$$\xi = \frac{P_s}{P_r}$$

Antes de continuar, es conveniente hacer una distinción entre el ruido generado por disturbios de la señal puramente aleatorios (y por lo tanto impredecibles) y la interferencia causada por la recepción no deseada de otra señal útil (como puede ser la causada por el acoplamiento de las líneas de alimentación).

Las fuentes de ruido pueden clasificarse en dos grandes grupos:

- Fuentes de ruido colocadas fuera de cualquier sistema de procesamiento (externas) y actuando en él por susceptibilidad.
- Fuentes de ruido dentro del sistema (internas) que generan ruido independiente a las condiciones externas.

Aunque siempre es posible mejorar el diseño de un sistema de procesamiento hasta reducir las interferencias a un nivel aceptable, es absolutamente imposible eliminar la contribución de las fuentes de ruido internas.

Normalmente un sistema se va a comportar correctamente únicamente cuando el nivel útil de señal es más alto que el nivel de ruido. Sin embargo, algunos métodos de procesamiento más elaborados permiten trabajar con pequeñas SNR, gracias a la información de propiedades de la señal conocida a priori.

Las fuentes de ruido externas al sistema pueden ser divididas en dos grandes grupos: las fuentes de interferencias generadas por artefactos eléctricos y fuentes de interferencias del tipo electromagnético. Dentro de las primeras se puede agrupar a los motores eléctricos, las bobinas (reactancias) de los fluorescentes, transformadores, rectificadores, etc. Dentro de las interferencias electromagnéticas se pueden citar las ondas electromagnéticas de comunicación, radiocomunicación, TV, etc.

Las fuentes de ruido interno también pueden ser divididas en dos grandes grupos: perturbaciones del tipo impulsivas generadas por la conmutación de corrientes y ruido de fondo generado en los cables y componentes electrónicos debido a la naturaleza electrónica de los mecanismos de conducción. Este último tiene varios orígenes, como una generalización se puede decir que

se produce como consecuencia del movimiento aleatorio de las partículas cargadas en equilibrio térmico (movimiento Browniano) o bajo influencia de campos aplicados a los mismos. Bajo condiciones estables pueden ser vistos como procesos estacionarios. Sus tres principales constituyentes en componentes electrónicos son: el ruido térmico, el ruido tipo disparo y el ruido de aleteo o *flicker*.

El ruido térmico es el encontrado más frecuentemente, y es causado por la vibración aleatoria de los portadores de carga provocada por la temperatura en los conductores. El ruido tipo disparo aparece en válvulas, transistores, diodos, fotodiodos, etc; donde existe una barrera de potencial que deben atravesar los portadores. El ruido de aleteo es atribuible a las propiedades de la superficie de un material.

1.4. Teoría de la comunicación

El estudio de las señales se encuentra contenido en lo que se denomina Teoría de la Comunicación, la cual se encarga del estudio de los sistemas de comunicación, tanto artificiales como biológicos o naturales. En la Figura 1.10 se pueden ver dos ejemplos de este tipo de sistemas.

Esta teoría está compuesta a su vez por dos grandes ramas: la Teoría de la Señal y la Teoría de la Información y Codificación (ver Figura 1.11).

1.4.1. Teoría de la señal

La descripción matemática de las señales es el objetivo fundamental de la teoría de la señal. Ésta proporciona el modo de enfatizar (de forma matemáticamente conveniente) las características fundamentales de una señal, como pueden ser su distribución espectral de energía o su distribución estadística de amplitudes. También provee los métodos para analizar la naturaleza de las modificaciones impuestas a la señal mientras ésta pasa por algún bloque del tipo eléctrico o electrónico.

Una de las herramientas básicas y fundamentales de la teoría de la señal es la expansión en términos de funciones ortogonales, siendo la expansión de Fourier el caso más interesante, y cuya forma generalizada es conocida como la Transformada de Fourier. Debido a su importancia dedicaremos un capítulo completo a revisar las bases de esta transformación y su aplicación al caso de las señales discretas.

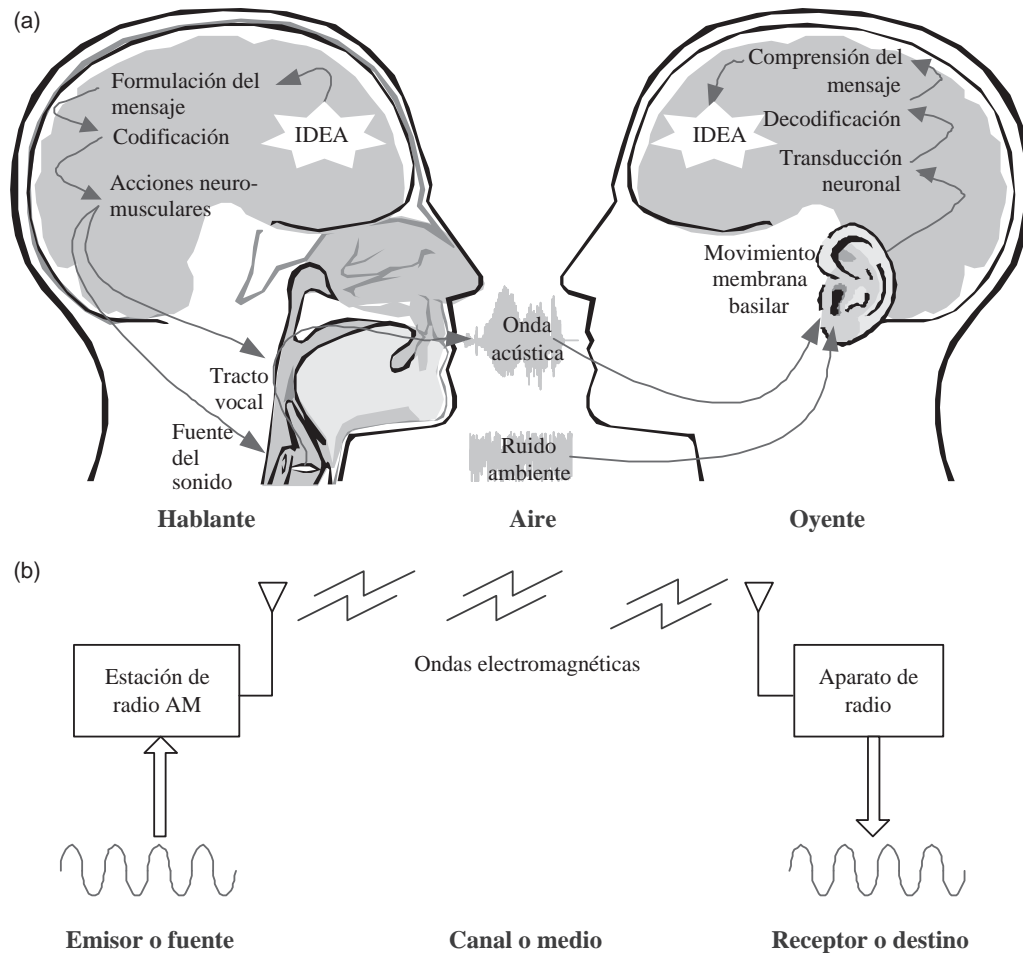


Figura 1.10. Sistemas de comunicación: (a) Humano por medio del habla, (b) Artificial por medio de un sistema de radio de amplitud modulada (AM)

1.4.2. Teoría de la información y de la codificación

La información está muy ligada al concepto de comunicación, es decir, transferencia de mensajes desde una fuente a un destinatario. La teoría de la información es una teoría probabilística de los mensajes, que tiene en cuenta sus propiedades estadísticas sin importar su significado. Provee un conjunto de conceptos que permiten la evaluación del desempeño de los sistemas de transferencia de información, especialmente cuando la señal está afectada por ruido.

Todo esto conduce al estudio de los métodos de codificación de la información. Las técnicas de codificación poseen tres objetivos fundamentales:

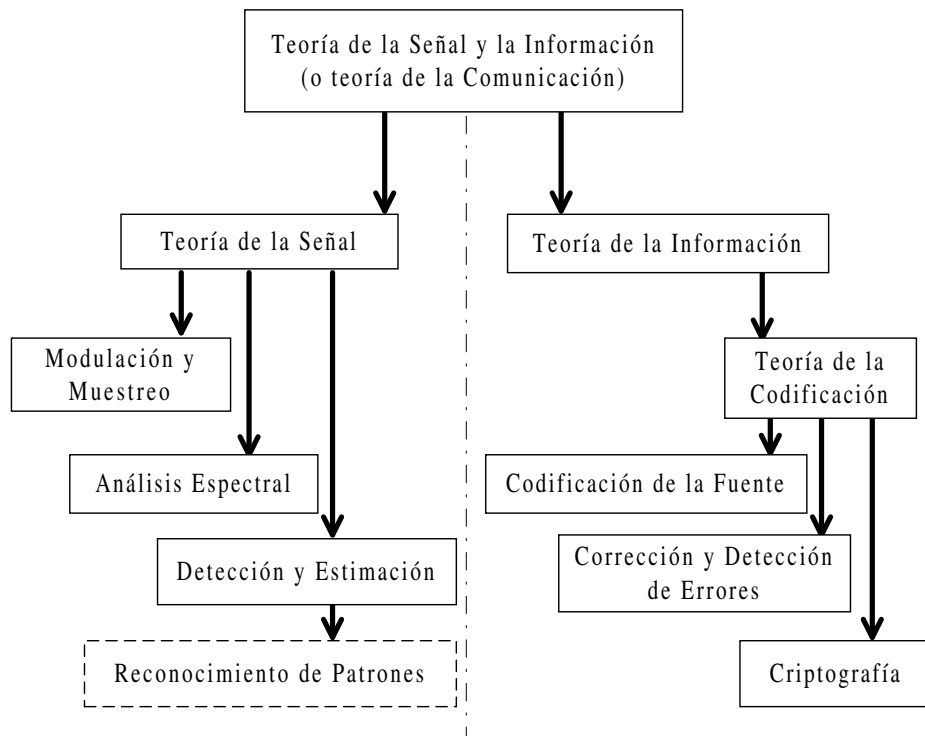


Figura 1.11. Esquema de la Teoría de la Comunicación

el primero es incrementar la densidad de la señal (compactar la señal lo más posible) eliminando la redundancia inútil, lo que se denomina codificación de fuente. El segundo objetivo es incrementar la confiabilidad de la señal, teniendo en cuenta las características con respecto al ruido. Esto se puede lograr incluyendo alguna redundancia, inteligentemente estructurada para permitir la posterior detección y corrección de los verdaderos errores, esto se denomina codificación de canal. Finalmente, el último objetivo de la codificación de la información es tratar de asegurar la privacidad de la comunicación (criptografía).

Aunque se ha hecho mención a la teoría de la información, el enfoque que interesa principalmente en este libro es el de la teoría de la señal.

1.5. Procesamiento de señales

Como ya se ha visto, la descripción matemática (o modelización) de la señal es el cometido de la Teoría de la Señal, y el procesamiento de la señal es la

disciplina técnica que, basada en los métodos de la teoría de la información y la señal, se encarga de la elaboración o interpretación de señales que acarrean información, con la ayuda de la electrónica, la computación y la física aplicada.

Podemos ver que las relaciones del hombre con su ambiente natural, o los sistemas que él mismo construye, están caracterizados por grandes niveles de intercambio de información. La observación (medición) de los fenómenos físicos o el diálogo (comunicación) entre los hombres, entre los hombres y las máquinas o entre máquinas, es hecho mediante señales (funciones del tiempo) o percepciones visuales (imágenes), cuya naturaleza es realmente compleja y puede ser enmascarada por disturbios indeseables (ruido de fondo, efectos atmosféricos, interferencias, etc.).

Las personas realizan complicados análisis de señales a través de los sistemas neurosensoriales y extraen información útil acerca de su entorno en forma prácticamente “transparente” para ellos. El sistema auditivo humano logra descifrar el mensaje “escondido” en los patrones de variación sonora producidos por el aparato fonador. Entendemos el mensaje codificado en el habla de manera asombrosamente “sencilla”, en forma casi independiente de factores como la identidad del hablante o el ruido de fondo. Por el contrario, los dispositivos artificiales que han tratado de emular estos aspectos distan mucho de poseer actualmente estas capacidades. A continuación revisaremos los procesamiento básicos que debemos perfeccionar para acercarnos más a estas capacidades “naturales”.

El extraer la información útil que se encuentra en estas señales (mediante análisis, filtrado, regeneración, medición, detección, e identificación) y mostrar los resultados correspondientes en la forma apropiada para el hombre o la máquina es uno de los objetivos principales del procesamiento de señales. En la Figura 1.12 se muestran los distintos tipos de procesamiento de una señal.

La generación de señales debe ser también considerada, permitiendo el estudio del comportamiento físico del sistema (p. ej., respuesta al impulso), o la transmisión y almacenamiento (síntesis, modulación y traducción a frecuencias, y codificación para reducir el efecto del ruido o la redundancia de información).

Para medir una señal, y especialmente una del tipo aleatorio, se trata de estimar el valor de una variable característica, que está vinculada a la misma con un determinado nivel de confianza. Un ejemplo es la medición de la señal de variación de la temperatura corporal a nivel cutáneo.

El filtrado es una función bien conocida, que consiste en eliminar o

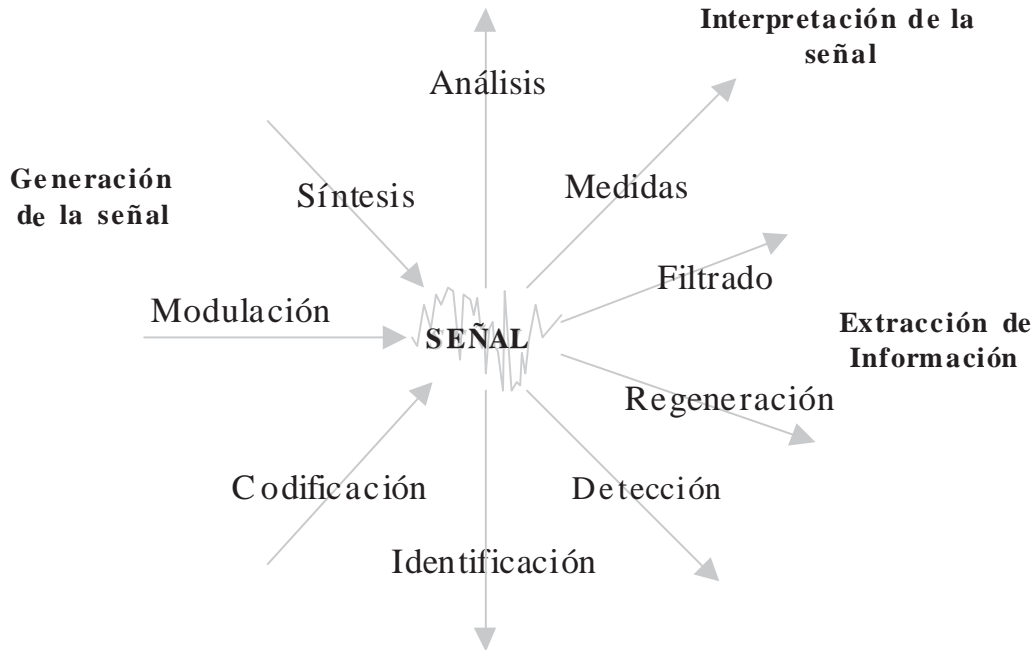


Figura 1.12. Tipos de procesamiento de la señal

disminuir algunas componentes no deseadas de la señal. Un ejemplo típico en el área biomédica es el de la eliminación del ruido de línea de 50 Hz previo a la adquisición del ECG.

La regeneración es la operación mediante la cual tratamos de retornar la señal a su forma inicial, después que ésta haya soportado algún tipo de distorsión. Por ejemplo la deconvolución de una imagen “desenfocada”.

Con un método de detección, tratamos de extraer una señal útil de un ruido de fondo de grandes dimensiones. La obtención de la señal de los denominados *Potenciales Evocados* podría encuadrarse dentro de este tipo de procesamiento. Algunas veces queremos recuperar la señal, otras simplemente saber si está presente o no en el registro considerado. Las técnicas de *correlación* pueden emplearse con este fin. Mediante los denominados filtros de correlación es posible detectar eventos de forma óptima, como una patología dentro de un electrocardiograma o la presencia de un eco en la señal del radar o del sonar.

Mediante el análisis, se trata de aislar los componentes del sistema que tienen una forma compleja para tratar de entender mejor su naturaleza u origen. Debido a la importancia del análisis de señales dedicaremos la

siguiente sección para ampliar este tópico.

La identificación es frecuentemente un proceso complementario, que permite clasificar la señal observada. Las técnicas de correlación son también frecuentemente usadas con este fin. Podemos comparar el canto de un ruiseñor con el de otro ruiseñor, debido a que la correlación es alta. Sin embargo, éste se correlaciona débilmente con el de una paloma. Para establecer las comparaciones se deben “construir” previamente una serie de *plantillas* adecuadas.

La síntesis es la operación opuesta al análisis, consiste en crear una señal con una forma apropiada mediante la combinación, por ejemplo, de un número de señales elementales. Este proceso es en general menos complejo que el de análisis, ya que puede verse como el *problema directo* de *armar* la señal en base a un conjunto de *partes*. Desde este punto de vista el análisis de una señal constituye precisamente el *problema inverso* que suele ser más difícil de resolver. Como ejemplo se podría mencionar la síntesis del habla partiendo de formas de onda sencillas, como senos, cosenos u ondas cuadradas.

El codificar una señal (además de su función de traducir una señal analógica a un lenguaje digital) es frecuentemente usado para minimizar los efectos del ruido, o tratar de conservar el ancho de banda o el volumen de memoria de una computadora, mediante la reducción de redundancia en una señal. Un ejemplo es la compresión del electrocardiograma para su almacenamiento en un dispositivo de registro continuo (Holter).

La modulación y traducción a frecuencias son las formas principales de adaptar una señal a las características de una línea de transmisión, de un filtro analizador, o de un medio de registro. Como ejemplo se pueden mencionar las técnicas clásicas para transmisión de señales de radio por medio de amplitud o frecuencia modulada (AM o FM).

1.5.1. Análisis de Señales

La palabra análisis proviene de la base griega *analyo* que significa desatar. Podemos definirla como: “Distinción y separación de las partes de un todo hasta llegar a conocer los principios o elementos de éste”¹. Como ya dijimos, el análisis de una señal consiste en aislar aquellas componentes que poseen una forma compleja para tratar de comprender mejor su naturaleza u origen. En este contexto llamamos *ruido* a cualquier fenómeno que perturba la percepción o interpretación de una señal. Es decir que analizar una señal

¹Diccionario General de la Lengua Española, Edición 1991, VOX.

consiste en encontrar y aislar aquellas partes características o *componentes ocultas* que mejor permitan describirlas, minimizando los efectos del ruido.

El análisis de fenómenos físicos posee elementos análogos, debido a que como hemos visto las señales constituyen manifestaciones del mundo físico. Su aparición es bastante anterior a este siglo, casi con el comienzo de la ciencia, y de hecho sentó las bases para el desarrollo de las teorías que sustentan el análisis de señales. En este sentido, podemos citar como ejemplo cercano el análisis de la luz visible mediante un prisma, que permite descomponerla en sus componentes fundamentales. Estas componentes están *ocultas* en la luz blanca y se manifiestan en su interacción con los objetos del mundo físico. Este fenómeno fue descubierto y estudiado por Newton como uno de sus primeros aportes a la óptica en 1670. Newton diseñó y construyó el primer telescopio reflector y concluyó que la luz blanca no era una única entidad después de observar la aberración cromática de su telescopio y de realizar el experimento del prisma en donde pudo observar el espectro (término que proviene de *spectrum*, o fantasma) de los componentes individuales de la luz blanca y recomponerlo con un segundo prisma.

Aunque Newton no reconoció el concepto de *frecuencia*, el parecido de este espectro con el de Fourier no es casual. Fourier conocía los trabajos de Newton y desarrolló las bases de su análisis cuando estudiaba la conducción del calor en los cuerpos sólidos. En 1807 Fourier difundió el primer esbozo de su *Teoría analítica del calor*, en la cual demostró que la conducción del calor en cuerpos sólidos se podía expresar como una suma infinita de términos trigonométricos cada vez más pequeños. Estos términos constituían las *componentes ocultas* que había podido descubrir en este fenómeno. Claramente Fourier había encontrado el *prisma* adecuado para analizar a los fenómenos de conducción del calor y como resultado había desarrollado la teoría del análisis armónico para descomponer funciones periódicas arbitrarias en términos de funciones sinusoidales. A pesar del aporte tremendo que constituiría su teoría fue fuertemente criticada por notables matemáticos de la época, como por ejemplo Laplace. Por su importancia en el contexto del procesamiento de señales, nosotros dedicaremos un capítulo completo al análisis de Fourier.

Como ejemplo de aplicación de las ideas de análisis a señales podemos retomar el caso de la señal de voz. Los primeros intentos para aplicar a ella el análisis de Fourier fueron realizados con dispositivos mecánicos como los basados en cuerdas, resonadores o filtros. Estos dispositivos realizaban una descomposición de los sonidos análoga a la propuesta por Fourier pero mediante principios mecánicos. De hecho pueden también establecerse

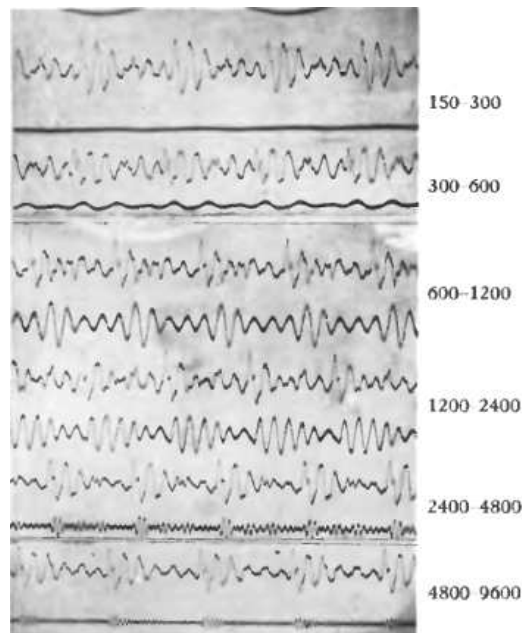


Figura 1.13. Descomposición de la /a/ como en “father” en oscilogramas obtenidos por medio de filtros dispuestos en octavas (reproducido del artículo original de Trendelenburg 1935).

análogas con el funcionamiento de la cóclea dentro nuestro oído, donde la membrana basilar constituye un complejo “anizador espectral”. También se utilizaron dispositivos de tipo estroboscópico. Con el advenimiento de los medios electrónicos “modernos” comenzaron a publicarse algunos trabajos que intentaron evidenciar las características y componentes fundamentales de esta señal. Por ejemplo en 1935, con dispositivos oscilográficos bastante sencillos y un banco de filtros analógicos dispuestos en octavas, se logró obtener los resultados de la figura 1.13 para una vocal del inglés. Por la naturaleza cuasiperiódica de las vocales pronunciadas en forma aislada, la /a/ era la que más fácilmente se ajustaba a un análisis de este tipo.

La aparición de las computadoras y la tecnología digital permite volver a aplicar el análisis de Fourier a la señal del habla. La digitalización de las señales de sonido permite introducirlas en la computadora para realizar cálculos con ellas. Un problema inicial era que los cálculos para obtener la transformada discreta de Fourier de una señal como esta demandaban mucho tiempo. En 1965 Cooley y Tukey publican un trabajo acerca de un algoritmo para el cálculo rápido de la Transformada Discreta de Fourier

mediante una computadora. Esto da lugar al resurgimiento de los estudios basados en espectros para aprovechar la flexibilidad y potencialidades de esta nueva herramienta.

Como vemos el análisis de Fourier ocupa un papel importantísimo dentro de las técnicas convencionales de análisis de señales, especialmente para aquellas señales derivadas de sistemas lineales e invariantes en el tiempo, es por ello que dedicaremos el Capítulo 3 a su estudio. Recientemente han surgido una serie de limitaciones de este tipo de análisis, por lo que se han dedicado esfuerzos importantes para desarrollar técnicas alternativas. Sin embargo los fundamentos de estas nuevas técnicas quedan fuera del alcance de la presente obra.

1.6. Operaciones elementales con señales

Para la realización de los procesamiento antes mencionados se requieren diversas operaciones sobre las señales en cuestión. En esta sección discutiremos aquellas operaciones elementales que permiten modificar a las señales. Estas operaciones pueden clasificarse en unarias y binarias.

1.6.1. Operaciones unarias

Una operación unaria involucra a una única señal, mientras que las binarias requieren dos señales. Algunas de las operaciones unarias son las operaciones de rango, las operaciones de dominio, muestreo e interpolación.

Operaciones de rango

Las operaciones de rango, las cuales modifican el rango de las señales, son definidas como:

$$x_{nuevo}(t) = \rho(x_{viejo}(t)) = (\rho \circ x_{viejo})(t)$$

Entre este tipo de operaciones, se pueden nombrar las operaciones de amplificación, rectificación y cuantización. Un ejemplo es la cuantificación uniforme, la cual se define como:

$$\rho(x) = \begin{cases} 0 & x < 0 \\ H \text{ int}(x/H) & 0 \leq x < (N-1)H \\ (N-1)H & x \geq (N-1)H \end{cases}$$

donde $\text{int}(\cdot)$, denota la parte entera del argumento.

Otro ejemplo es la rectificación de onda completa, la cual se define como:

$$\rho(x) = |x|$$

Operaciones de dominio

Las operaciones de dominio, que modifican la variable independiente, son definidas como:

$$x_{\text{nuevo}}(t) = x_{\text{viejo}}(\tau^{-1}(t))$$

Entre este tipo de operaciones, se pueden nombrar las operaciones de expansión, compresión, reversión, traslación, las cuales tienen la forma:

$$\tau^{-1}(t) = \alpha t$$

si:

- $\alpha > 1 \Rightarrow$ compresión,
- $0 < \alpha < 1 \Rightarrow$ expansión,
- $\alpha = -1 \Rightarrow$ reversión.

Otro ejemplo de las operaciones de dominio es la traslación, la cual se define como:

$$\tau^{-1}(t) = t + \theta$$

donde θ es una constante real.

Muestreo

Esta operación pasa la variable independiente de un dominio continuo a otro discreto. El muestreo puede ser uniforme (cuando el dominio es discretizado en forma uniforme) o no uniforme.

Interpolación

La interpolación consiste en pasar una señal cuya variable independiente pertenece a un dominio discreto, a una señal cuya variable independiente pertenece a un dominio continuo. Esta puede ser expresada como:

$$x(t) = \sum_n x^*(nT) I\left(\frac{t - nT}{T}\right)$$

donde I es la función interpolante. Existen varias funciones interpolantes posibles, entre las cuales podemos nombrar:

Función de interpolación escalón:

$$I_{escalon}(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 0 & \text{en otro caso} \end{cases}$$

Función de interpolación lineal:

$$I_{lineal}(t) = \begin{cases} 1 - |t| & |t| < 1 \\ 0 & \text{en otro caso} \end{cases}$$

Función de interpolación sinc: $I_{sinc}(t) = \text{sinc}(\pi.t)$ donde:

$$\text{sinc}(t) = \begin{cases} \frac{\sin(t)}{t} & t \neq 0 \\ 1 & t = 0 \end{cases}$$

1.6.2. Operaciones binarias

Las operaciones binarias se realizan punto a punto entre dos señales. Entre ellas se puede nombrar a la adición, sustracción, multiplicación y a la división.

1.7. Preguntas

1. ¿Cómo intervienen los criterios prácticos y los errores de medición y cómputo en la clasificación fenomenológica de señales reales?
2. Clasifique las siguientes señales según todos los criterios que conoce:
 - a) la velocidad del viento en Mabuji-Maye (Zaire),
 - b) $y(t) = \sin(2\pi 100t); t, y \in \mathbb{R}$,
 - c) la intensidad de luz del pixel (34,178) cuando se proyecta la película “El Nombre de la Rosa” (versión cinematográfica del famoso libro de Umberto Eco),
 - d) la corriente que circula por el cable del teclado de su computadora,

e) el delta de Dirac, definido como:

$$\delta[n] = \begin{cases} 1 & \text{si } n = 0 \\ 0 & \forall n \neq 0 \end{cases}; n \in \mathbb{Z}, -\infty < n < \infty.$$

- f) la variación anual de habitantes de origen asiático en Buenos Aires,
- g) la altura de las aguas en la costa del río Genil (España),
- h) el consumo de combustible por minuto que requiere un F1 durante el Gran Premio de Mónaco,
- i) la suma de todas las emisiones con contenido futbolístico en Radio Mitre,
- j) las ganancias mensuales y la cantidad de fallas en los productos de Microsoft en los últimos 10 años,
- k) el tango Adiós Nonino (Astor Piazzola), interpretado por el autor y su Quinteto Tango Nuevo, en Viena (1983),
- l) la cantidad de veces que se abre la puerta del aula por día a lo largo de un año,
- m) la cantidad de letras “a” por cada página de “El Aleph” (libro de J. L. Borges),
- n) el contenido temático de un canal de televisión por cable,
- \tilde{n}) la cantidad de moléculas 2 Fe(OH)_3 que se forman por día,
- o) el monto en dólares de la deuda externa de Argentina en los últimos 20 años.

3. Realice una lista de las señales que pueden medirse en el cuerpo humano y clasifíquelas según los criterios morfológico, fenomenológico y dimensional.
4. ¿Por qué razón se necesitan muchas realizaciones de una señal aleatoria para poder comprobar experimentalmente su estacionariedad?
5. Describa el proceso de verificación de la ergodicidad de una señal aleatoria asumiendo que el promedio y la desviación estándar son medidas suficientes para su caracterización estadística.
6. Enuncie las hipótesis que se han hecho sobre la señal en el punto anterior y generalice la prueba de ergodicidad mediante el uso de medidas aptas para el caso más general.

7. ¿Qué beneficio práctico brinda poder asumir que una determinada señal es ergódica?
8. ¿Por qué decimos que el `random` de la computadora es *pseudo* aleatorio? ¿Puede una computadora digital generar una señal realmente aleatoria?
9. Analice el proceso de discretización en tiempo de un período de la señal $x(t) = \cos(2\pi 10t)$. Incremente el período de muestreo hasta no poder reconstruir la señal continua a partir de la señal de tiempo discreto.
10. Si posee una señal con la forma $x(t) = \sin(2\pi 100t) + r(t)$, donde $r(t)$ es una señal aleatoria con distribución uniforme en $[-0,1 \dots 0,1]$, ¿cómo procedería para calcular la relación señal ruido?

1.8. Trabajos prácticos

Ejercicio 1: Genere y grafique las siguientes señales:

1. senoidal
2. sync
3. onda cuadrada
4. onda triangular
5. delta de Dirac
6. ruido aleatorio

Ejercicio 2: Realice las siguientes operaciones básicas sobre una señal senoidal:

1. expansión
2. compresión
3. inversión
4. rectificación
5. cuantización en 8 niveles
6. traslación

Ejercicio 3: Discretice una señal senoidal con frecuencia 5 Hz. y duración 1 seg. Utilice las siguientes frecuencias de muestreo: 1000, 100, 25, 10, 4, 1 y 0,5 Hz. Grafique y analice el resultado en cada uno de los casos.

Ejercicio 4: Discretice una señal senoidal con frecuencia 4000 Hz. y duración 2 seg., utilizando una frecuencia de muestreo de 129 Hz. Grafique el resultado y estime la frecuencia de la onda sinusoidal que se observa en la figura. Analice y obtenga conclusiones.

Ejercicio 5: Discretice una señal arbitraria con frecuencia de muestreo de 10 Hz y sobremuestreela mediante distintos tipos de interpoladores a 4 veces la frecuencia de muestreo.

Ejercicio 6: Genere una señal compleja del tiempo y gráfiquela en 3 dimensiones.

Ejercicio 7: (*) Genere una señal aleatoria con distribución gaussiana y verifique su ergodicidad.

Ejercicio 8: Lea dos señales sonoras desde archivos (p. ej., en formato WAV) y luego súmelas. Resultaría interesante que usted grabara la voz de dos personas pronunciando una frase corta. Puede hacerlo con un micrófono y la Grabadora de Sonidos (en Windows) o el comando `rec` en Linux. Guarde el resultado en un archivo del mismo formato y oiga las tres señales.

Ejercicio 9: Utilice una señal sonora conocida y súmele un ruido aleatorio. Oiga el resultado y compare con la señal original.

Ejercicio 10: Calcule la relación señal ruido de la mezcla del ejercicio anterior y vuelva a *ensuciar* la señal con relación señal ruido de 0 dB y 100 dB. Grafique, oiga y compare los resultados.

Ejercicio 11: Utilice una señal sonora conocida y multiplique cada uno de sus elementos por una constante. Oiga el resultado y compare con la señal original.

Ejercicio 12: Utilice una señal sonora conocida y multiplique cada uno de sus elementos por una recta decreciente que tenga valor 1 en el primer elemento y 0 en el último. Oiga el resultado y compare con la señal original.

Bibliografía

- [1] E. Brigham. *The Fast Fourier Transform and its applications*. Prentice-Hall, 1988.
- [2] L. E. Franks. *Teoría de la Señal*. Reverté S.A., 1975.
- [3] W. Kaplan. *Matemáticas Avanzadas*. Addison-Wesley Iberoamericana, 1986.
- [4] H. Kwakernaak, R. Sivan, and T Strijbos. *Modern signals and systems*. Prentice-Hall, 1991.
- [5] A. V. Oppenheim and R. W. Schaffer. *Digital Signal Processing*. Prentice-Hall, 1975.
- [6] A. Papoulis and M. Bertrán. *Sistemas y Circuitos Digitales y Analógicos*. Marcombo, 1989.
- [7] R. Roberts and R. Gabel. *Señales y sistemas lineales*. Limusa, 1994.
- [8] N. K. Sinha. *Linear Systems*. Wiley, 1991.
- [9] H. Skilling. *Circuitos en Ingeniería Eléctrica*. Cia. Ed. Continental, México, 1987.

Capítulo 2

Espacios de señales

Diego Milone, Leandro Di Persia

Temas a tratar

- Señales y vectores.
- La relación entre el álgebra lineal y las señales.
- Espacios de señales y espacios vectoriales.
- Bases y transformaciones lineales.

Objetivos

- Ver a las señales como elementos de un espacio vectorial.
- Reinterpretar conceptos básicos del álgebra lineal en el contexto del procesamiento de señales.
- Valorar la importancia del producto interno en el procesamiento de señales.
- Presentar los fundamentos generales de las transformadas lineales más usadas.
- Aplicar las herramientas en estudio en problemas sencillos.

2.1. Introducción

La mayoría de las personas están acostumbradas a extraer información con gran sensibilidad de señales representadas como una colección complicada de puntos en un marco simple, como es el espacio bidimensional de las imágenes. En esta unidad incorporaremos a las señales en un marco más estructurado: el espacio vectorial. Considerando a las señales como vectores de un espacio n -dimensional, podremos aprovechar todas las propiedades de la estructura algebraica de los espacios vectoriales e interpretar el procesamiento de las señales desde una perspectiva conceptual muy sencilla.

2.1.1. Desarrollo intuitivo

Suponga que tomamos mediciones de temperatura a intervalos de 1 minuto, simplemente con un termómetro. Al cabo de dos minutos habremos obtenido dos valores de temperatura, por ejemplo, 2 y 3 grados. Estamos acostumbrados a representar estos valores en una gráfica en la que el eje de las abscisas indica el tiempo y el de las ordenadas la magnitud de la temperatura.

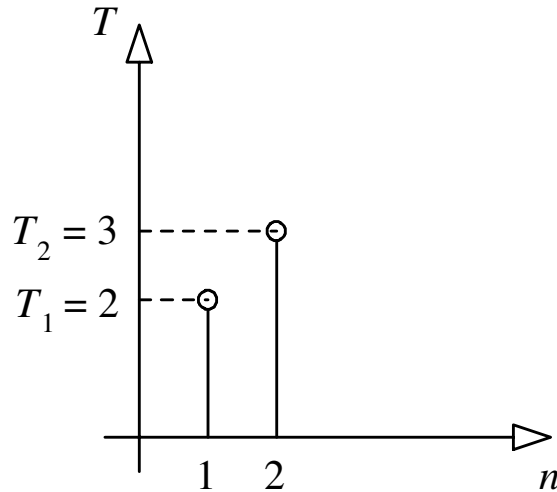


Figura 2.1. Gráfica de una serie de mediciones en función del tiempo.

Sin embargo, podemos también representar estos valores como un vector en un espacio de dos dimensiones, tal como se observa en la Figura 2.2.

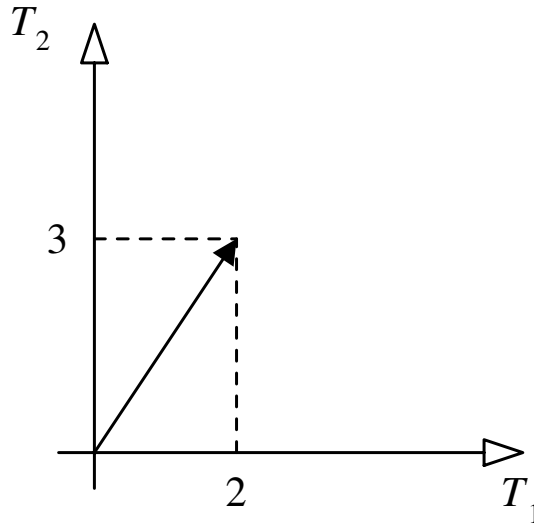


Figura 2.2. Una señal de 2 muestras (mediciones) en \mathbb{R}^2 .

Así, vemos que esta señal de dos muestras puede ser representada mediante un vector de dos componentes reales, es decir, un vector en \mathbb{R}^2 .

Un minuto después habremos obtenido una nueva medición, supongamos 1 grado, por lo que nuestra señal ya tendrá tres muestras y podemos representarla con un vector en tres dimensiones como el de la Figura 2.3.

Siguiendo con esta idea, vemos que al cabo de una hora tendremos una señal de 60 muestras, que podría ser interpretada como un vector en \mathbb{R}^{60} (pero no podremos representarlo gráficamente).

¿Es posible aplicar estas ideas a señales continuas? ¿Cómo representaría la señal $s(t) = \sin(\omega t)$? ¿En qué dimensión estaría el vector?

Al tratarse de una señal continua, en cualquier intervalo que consideremos habrá infinitos valores. Así podríamos ver a las señales continuas como vectores en \mathbb{R}^∞ , es decir, vectores con infinitos elementos o muestras. Realizando las consideraciones necesarias, es posible entonces “extrapolar” algunas de las nociones de espacios de dimensiones finitas a los de dimensiones infinitas de interés práctico.

Viendo a las señales como vectores podemos aprovechar muchas herramientas del álgebra lineal para entender el procesamiento de señales. A continuación haremos un repaso de ciertos conceptos de álgebra lineal y estudiaremos su aplicación a la Teoría de Señales.

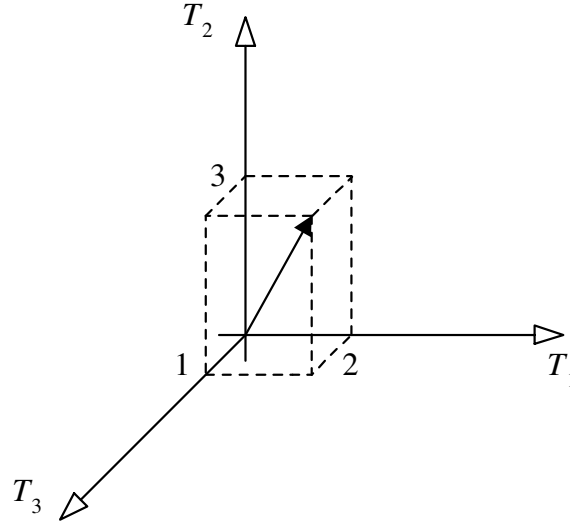


Figura 2.3. Una señal de 3 muestras en \mathbb{R}^3 .

2.2. Señales, vectores y álgebra lineal

Durante los cursos de álgebra acostumbramos tratar con vectores. Definíamos a estos objetos como colecciones o arreglos de datos que forman una entidad independiente. Solemos asociar los vectores con la representación gráfica de puntos en el espacio bidimensional y tridimensional. Como vimos anteriormente, estas ideas pueden aplicarse a las señales y esta interpretación geométrica nos brindará un enfoque simple para comprender procesos complicados en señales.

En forma general, podemos decir que una señal en el espacio N -dimensional es un vector $[x_1, x_2, \dots, x_N]$, definido como una N -upla ordenada de números. Para estos vectores utilizamos la notación:

$$\mathbf{x} = [x_n]; \quad n \in \mathbb{N}; x_n \in \mathbb{R}; \mathbf{x} \in \mathbb{R}^N \quad (2.1)$$

De forma similar, para el caso de señales continuas utilizamos la notación:

$$\mathbf{x} = [x(t)]; \quad t \in \mathbb{R}; x(t) \in \mathbb{R}; \mathbf{x} \in \mathbb{R}^\infty \quad (2.2)$$

2.2.1. Normas

Generalmente es útil tener alguna medida del tamaño de las señales. La norma provee este tipo de medida. La norma de un vector \mathbf{x} es un número

real positivo que toma el valor 0 sólo cuando $\mathbf{x} = \mathbf{0}$. Existen muchas normas y cada una define un tipo especial de medida para un vector. De acuerdo al tipo de problemas que se estén tratando, algunas serán mas apropiadas que otras. Una norma muy utilizada es la *norma-p*, definida como:

$$\|\mathbf{x}\|_p = \begin{cases} \left(\sum_{n=1}^N |x_n|^p \right)^{1/p}, & 1 \leq p < \infty \\ \sup_{n \in [1, N]} |x_n|, & p = \infty \end{cases}$$

Para el caso de señales continuas se define esta norma según:

$$\|\mathbf{x}\|_p = \begin{cases} \left(\int_{-\infty}^{\infty} |x(t)|^p dt \right)^{1/p}, & 1 \leq p < \infty \\ \sup_{t \in \mathbb{R}} |x(t)|, & p = \infty \end{cases}$$

Si $p = 1$, tenemos la norma 1, también conocida como *acción* de la señal:

$$\|\mathbf{x}\|_1 = \sum_{n=1}^N |x_n| \quad \text{o} \quad \|\mathbf{x}\|_1 = \int_{-\infty}^{\infty} |x(t)| dt$$

Si $p = 2$, tenemos la norma 2:

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{n=1}^N |x_n|^2} \quad \text{o} \quad \|\mathbf{x}\|_2 = \sqrt{\int_{-\infty}^{\infty} |x(t)|^2 dt}$$

La norma 2 da una idea del tamaño del objeto en un sentido físico, específicamente en el caso de vectores se trata de la longitud de éstos. Esta norma está directamente relacionada con la *energía* de la señal, que se define como:

$$E(\mathbf{x}) = \|\mathbf{x}\|_2^2$$

Si $p = \infty$, tenemos la norma infinito:

$$\|\mathbf{x}\|_{\infty} = \sup_{n \in [1, N]} |x_n| \quad \text{o} \quad \|\mathbf{x}\|_{\infty} = \sup_{t \in \mathbb{R}} |x(t)|$$

que en el análisis de señales corresponde a la *amplitud* de una señal positiva:

$$A(\mathbf{x}) = \|\mathbf{x}\|_{\infty}$$

En la Figura 2.4 se pueden observar la energía y la amplitud de una señal en \mathbb{R}^2 .

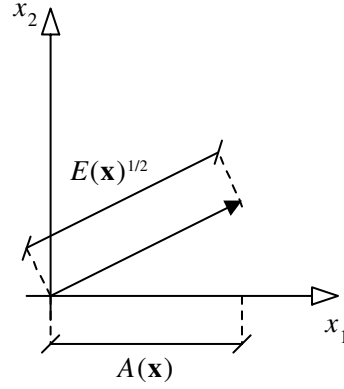


Figura 2.4. Representación gráfica de la energía y amplitud de una señal.

En la Figura 2.5 se muestra una representación gráfica de la norma- p de señales en \mathbb{R}^2 para diferentes valores de p .

Existen otras medidas de interés para caracterizar las señales y, en algunos casos, estas medidas están directamente relacionadas con la norma. Cuando la energía de una señal no es finita, es útil definir su *potencia* o valor cuadrático medio como:

$$P(\mathbf{x}) = \lim_{N \rightarrow \infty} \frac{1}{2N} \sum_{n=-N}^N |x_n|^2 \quad \text{o} \quad P(\mathbf{x}) = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T |x(t)|^2 dt.$$

Otra medida muy útil es la raíz del valor cuadrático medio (RMS, del inglés *root mean square*), definida como:

$$RMS(\mathbf{x}) = \sqrt{P(\mathbf{x})}.$$

Por último, el *valor medio* de una señal se define como:

$$m(\mathbf{x}) = \lim_{N \rightarrow \infty} \frac{1}{2N} \sum_{n=-N}^N x_n \quad \text{o} \quad m(\mathbf{x}) = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T x(t) dt.$$

2.2.2. Producto interno

Dados dos vectores $\mathbf{x}, \mathbf{y} \in \mathbb{R}^N$, se define su *producto interno* $\langle \mathbf{x}, \mathbf{y} \rangle \in \mathbb{R}$ como:

$$\langle \mathbf{x}, \mathbf{y} \rangle = x_1 y_1^* + x_2 y_2^* + \dots + x_N y_N^* = \sum_{i=1}^N x_i y_i^*$$

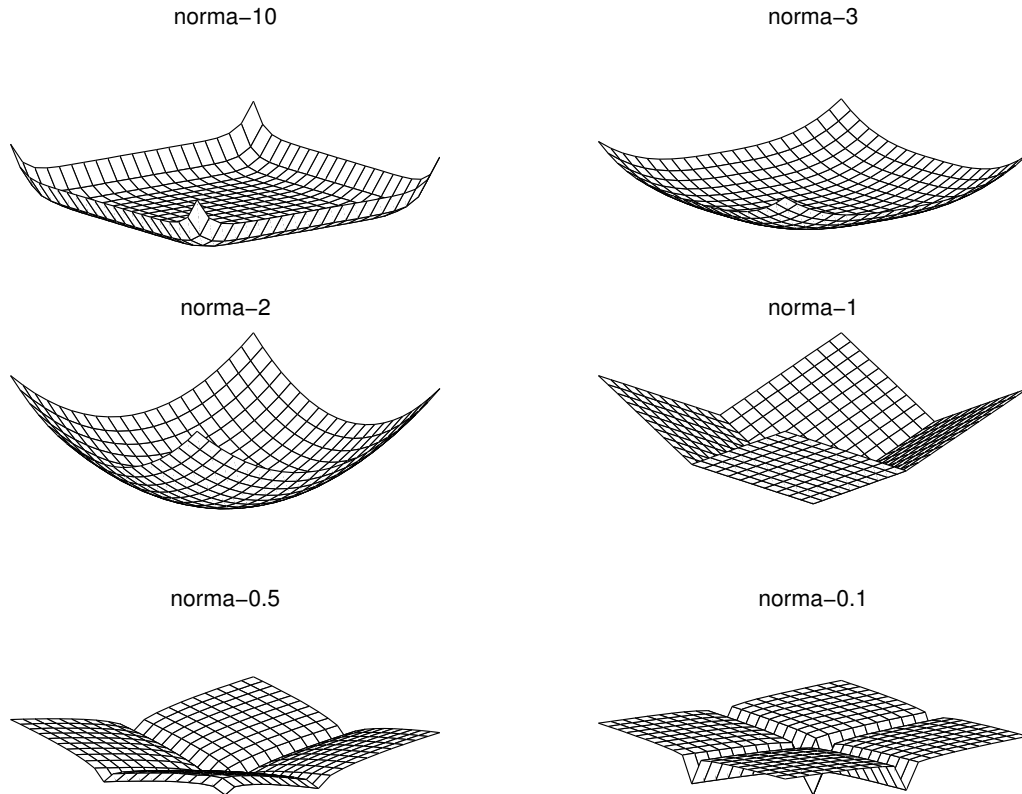


Figura 2.5. Varios ejemplos de la norma- p para señales en \mathbb{R}^2 . En el centro de cada gráfica las señales tienen ambas componentes nulas.

donde el $*$ representa el conjugado en caso de tratarse de valores complejos. En la bibliografía se pueden encontrar otras notaciones como $\mathbf{x} \cdot \mathbf{y}$ o también como un producto matricial \mathbf{xy}^T . Cuando tratamos con señales continuas el producto interno queda definido como:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \int_{-\infty}^{\infty} x(t)y^*(t)dt$$

De esta ecuación podemos ver que el producto interno de un vector consigo mismo es igual al cuadrado de su norma 2 (es decir la energía de la señal):

$$\|\mathbf{a}\|_2^2 = \langle \mathbf{a}, \mathbf{a} \rangle = \mathbf{a} \cdot \mathbf{a}$$

El producto interno de vectores tiene una clara interpretación geométrica relacionada con la proyección o componente de un vector sobre otro.

Definimos la proyección de \mathbf{x} sobre \mathbf{y} como:

$$\text{proy}_{\mathbf{y}}(\mathbf{x}) = \|\mathbf{x}\|_2 \cos(\phi)$$

donde ϕ es el ángulo que forman los vectores. Otra forma de calcular el producto interno es:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \|\mathbf{x}\|_2 \|\mathbf{y}\|_2 \cos(\phi),$$

y así obtenemos:

$$\text{proy}_{\mathbf{y}}(\mathbf{x}) = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{y}\|_2}.$$

Podemos ver que el producto interno nos da una idea del aporte de una señal en otra. Para independizarnos de la energía de la señal con la que estamos comparando, dividimos el producto interno por la norma 2 de ésta (en un espacio euclídeo). Esta interpretación del producto interno como proyección de una señal en otra puede verse gráficamente en la Figura 2.6.

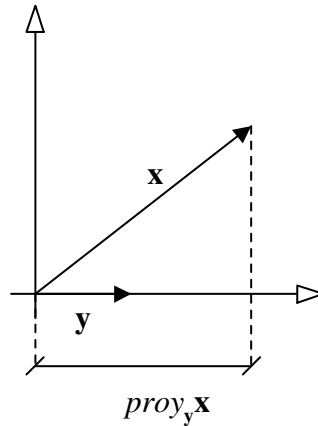


Figura 2.6. Proyección de la señal \mathbf{x} en la dirección de \mathbf{y} .

En el caso particular de que la señal sobre la que estamos proyectando tenga norma 2 unitaria, el producto interno es directamente una medida del *parecido* entre ambas señales. En la Figura 2.7 se muestran tres casos importantes para vectores en dos dimensiones y su equivalente en señales continuas.

Una vez aclarados estos conceptos básicos sobre vectores y su interpretación en el ámbito de las señales, pasaremos a formalizar lo relacionado con espacios vectoriales.

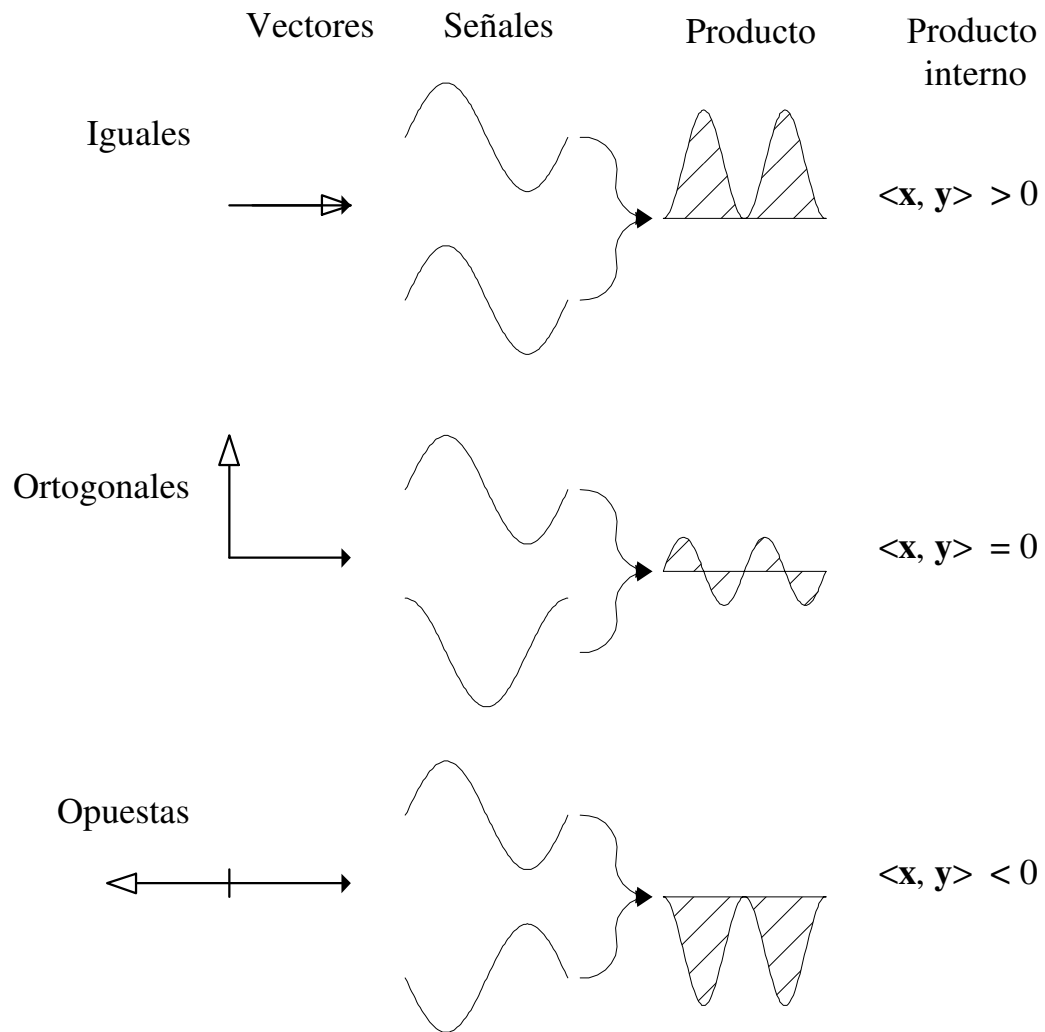


Figura 2.7. El producto interno y su significado en teoría de señales.

2.3. Espacios vectoriales y señales

2.3.1. Conjunto de señales

Consideremos un conjunto de señales S . Para determinar si una señal pertenece al conjunto S utilizamos una propiedad o prueba P . Un elemento x pertenecerá a S si cumple con esta propiedad, lo que podemos expresar como: $S = \{x/P\}$. La elección de P debe necesariamente adaptarse al problema en cuestión. A continuación se darán algunos ejemplos de conjuntos de señales que se encuentran frecuentemente en los problemas de análisis de la señal.

Señales sinusoidales:

$$S_s = \{\mathbf{x}/x(t) = A \sin(2\pi ft + \theta)\}$$

donde $t, A, f, \theta \in \mathbb{R}$, t representa al tiempo, A a la amplitud, f a la frecuencia y θ a la fase. De esta forma S_s contiene todas las posibles sinusoides, contemplando todos los valores posibles de amplitud, fase y frecuencia.

Señales periódicas:

$$S_p = \{\mathbf{x}/x(t) = x(t + T)\}$$

donde $t, T \in \mathbb{R}$, t representa al tiempo, y T es el período de la señal.

Señales acotadas en amplitud:

$$S_a = \{\mathbf{x}/\|\mathbf{x}\|_\infty \leq K\}$$

donde $K \in \mathbb{R}^+$ es una constante. Este es el conjunto de señales cuyos valores instantáneos están acotados en magnitud por K .

Señales de energía limitada:

$$S_e = \{\mathbf{x}/\|\mathbf{x}\|_2^2 \leq K\}$$

donde $K \in \mathbb{R}^+$ es una constante. Este es el conjunto de señales con energía menor o igual a K .

2.3.2. Espacios de señales

Habiendo reunido en un conjunto todas las señales que presentan alguna propiedad común, estamos en condiciones de examinar las características distintivas de los elementos dentro del conjunto. Una señal en particular sólo interesa en relación con las demás señales del conjunto. Por ejemplo, podemos preguntar acerca de una señal respecto a las demás: ¿tiene más energía?, ¿dura más?, ¿fluctúa más rápidamente?, ¿tiene más ceros?, ¿tiene un valor de pico mayor?, etc. Un método general para caracterizar la diferencia entre dos elementos de un conjunto consiste en asignar a cada par de elementos un número real positivo. Éste se interpreta como distancia entre los elementos y el propio conjunto comienza a tomar un carácter geométrico. Para definir una distancia se necesita un funcional $d : \{x, y\} \rightarrow \mathbb{R}$ que se aplique a todos los pares de elementos del conjunto. Dicho funcional se denomina *métrica* si cumple las siguientes propiedades:

- I) $d(x, y) \geq 0 \quad \wedge \quad d(x, y) = 0 \Leftrightarrow x = y,$
- II) $d(x, y) = d(y, x) \quad (\text{simetría}),$
- III) $d(x, z) \leq d(x, y) + d(y, z) \quad (\text{desigualdad del triángulo}).$

A un conjunto S al que le hemos asociado una métrica particular d le llamamos espacio métrico. En el caso de que el conjunto S contenga señales, denominamos al par (S, d) *espacio de señales*. Cabe destacar que estas definiciones no implican exigirle ninguna propiedad extra a los elementos del conjunto, simplemente se define la manera de medir las distancias entre los elementos. Por ejemplo, podemos definir el espacio métrico (\mathbb{R}, d) a partir del conjunto \mathbb{R} de números reales y una métrica $d(x, y) = |x - y|$; $x, y \in \mathbb{R}$. Esta es la métrica usual sobre \mathbb{R} .

Debe notarse que dos métricas diferentes, definidas sobre el mismo conjunto de señales, forman dos espacios de señales diferentes. Si al conjunto S_s de señales sinusoidales, le asociamos la métrica definida por la norma 1:

$$d_a(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_1$$

obtenemos el espacio de señales (S_s, d_a) . Si al mismo conjunto S_s le asociamos otra métrica definida por la norma 2:

$$d_b(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2$$

obtenemos otro espacio de señales *diferente* (S_s, d_b) .

2.3.3. Espacios vectoriales

Una vez que hemos definido un espacio de señales, es decir, un conjunto particular de señales con una métrica asociada, estamos interesados en manipularlo, para lo cual necesitamos una estructura algebraica simple. Dicha estructura la proporciona un espacio vectorial, el cual se define a continuación.

Un *espacio vectorial* \mathcal{S} es un cuádruple $(S, K, +, \cdot)$ que posee un conjunto de elementos llamados vectores, un campo de escalares, una operación de adición y una operación de producto por un escalar, que satisfacen las siguientes propiedades:

- I. La adición es cerrada:
 $\mathbf{x} + \mathbf{y} \in S; \quad \forall \mathbf{x}, \mathbf{y} \in S$
- II. La adición es conmutativa:
 $\mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x}; \quad \forall \mathbf{x}, \mathbf{y} \in S$
- III. La adición es asociativa:
 $\mathbf{x} + (\mathbf{y} + \mathbf{z}) = (\mathbf{x} + \mathbf{y}) + \mathbf{z}; \quad \forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in S$
- IV. Existe un único elemento $\mathbf{0} \in S$ que es neutro respecto a la adición:
 $\mathbf{x} + \mathbf{0} = \mathbf{x}; \quad \forall \mathbf{x} \in S$
- V. El producto por un escalar es cerrado:
 $\alpha \mathbf{x} \in S; \quad \forall \mathbf{x} \in S \wedge \forall \alpha \in K$
- VI. El producto por un escalar es asociativo:
 $\alpha(\beta \mathbf{x}) = (\alpha\beta) \mathbf{x}; \quad \forall \mathbf{x} \in S \wedge \forall \alpha, \beta \in K$
- VII. El producto por un escalar es distributivo según:
 $\alpha(\mathbf{x} + \mathbf{y}) = \alpha \mathbf{x} + \alpha \mathbf{y}; \quad \forall \mathbf{x}, \mathbf{y} \in S \wedge \forall \alpha \in K$
- VIII. El producto por un escalar es distributivo según:
 $(\alpha + \beta) \mathbf{x} = \alpha \mathbf{x} + \beta \mathbf{x}; \quad \forall \mathbf{x} \in S \wedge \forall \alpha, \beta \in K$
- IX. Existe un único elemento $1 \in K$ que es neutro respecto al producto por un escalar:
 $1 \mathbf{x} = \mathbf{x}; \quad \forall \mathbf{x} \in S$

Para que un conjunto de vectores dado constituya un espacio vectorial, según esta definición deben cumplirse *todas* las propiedades anteriores. Basta

con que una sola no se cumpla para que el conjunto no constituya un espacio vectorial. Si bien en esta definición nos referimos a espacios vectoriales como conjuntos de vectores sin importar su naturaleza, en el contexto que nos interesa nos centraremos en los vectores como señales discretas o continuas según se definieron en las ecuaciones (2.1) y (2.2). En este caso, el campo de escalares es generalmente $K = \mathbb{C}$. Además, la operación de adición se define como:

$$\mathbf{x} + \mathbf{y} = [x_i + y_i]_{i \in [1, N] \subset \mathbb{N}} \quad \text{o} \quad \mathbf{x} + \mathbf{y} = [x(t) + y(t)]_{t \in \mathbb{R}}$$

y el producto por un escalar $\alpha \in K$ queda definido según:

$$\alpha \mathbf{x} = [\alpha x_i]_{i \in [1, N] \subset \mathbb{N}} \quad \text{o} \quad \alpha \mathbf{x} = [\alpha x(t)]_{t \in \mathbb{R}}$$

La ventaja de demostrar que un conjunto de señales es un espacio vectorial radica en la existencia de toda una serie de propiedades que se cumplen para cualquier espacio vectorial, sin importar su naturaleza. Es decir, una vez que demostramos que un conjunto de señales es un espacio vectorial, podemos dar por sentadas muchas propiedades y aplicarlas sin necesidad de demostrarlas.

Subespacios

Cuando sabemos que un conjunto de señales constituye un espacio vectorial \mathcal{V} , podemos demostrar que un subconjunto no vacío de éstas también constituye un espacio vectorial \mathcal{V}_0 , simplemente verificando que este subconjunto sea cerrado ante la adición y el producto por un escalar definidos en \mathcal{V} . A este subconjunto de señales, que a su vez es un espacio vectorial, se lo denomina *subespacio vectorial*.

Por ejemplo, sea el conjunto de las señales senoidales de frecuencia fija $f = 5$ Hz:

$$\mathcal{S}_{S_5} = \{\mathbf{x}/x(t) = A \sin(2\pi 5t + \theta)\}$$

donde $t, A, \theta \in \mathbb{R}$, t representa al tiempo, A a la amplitud y θ a la fase. \mathcal{S}_{S_5} junto con las operaciones de suma y multiplicación por un escalar constituye un espacio vectorial \mathcal{S}_{S_5} . Ahora podemos definir el subconjunto de las señales sinusoidales de fase constante $\theta = 2$ radianes:

$$\mathcal{S}_{S_{52}} = \{\mathbf{x}/x(t) = A \sin(2\pi 5t + 2)\}$$

el cual claramente es cerrado frente a la adición y a la multiplicación por un escalar, lo que demuestra que es un subespacio vectorial de \mathcal{S}_{S_5} .

Espacios normados

Dados un espacio vectorial y una definición de norma, se dice que éste es un espacio normado si la norma es finita para todos sus elementos. Basándonos en la definición de *norma-p*, podemos definir un espacio normado en base al conjunto:

$$\{\mathbf{x} / \|\mathbf{x}\|_p < +\infty\}$$

cuando se trata de señales discretas se utiliza la notación $\ell^p(\mathbb{R})$ y cuando las señales son continuas $L^p(\mathbb{R})$. En el caso particular de que utilicemos la norma 1 queda definido el espacio de las señales *absolutamente integrables* $L^1(\mathbb{R})$. Cuando se utiliza la norma 2 se define el espacio de las señales *cuadrado integrables* o de energía finita $L^2(\mathbb{R})$. De forma similar, estos ejemplos son aplicables a las señales discretas en $\ell^1(\mathbb{R})$ y $\ell^2(\mathbb{R})$. Como vimos anteriormente el producto interno permite adoptar una perspectiva geométrica y utilizar terminología familiar de los espacios clásicos de dimensión finita. Cada producto interno en un espacio vectorial da lugar a una norma de la siguiente forma:

$$\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}.$$

Si el espacio es *completo* con respecto a esta norma entonces constituye un espacio de *Hilbert*. Una definición exacta de completitud a esta altura nos desviaría demasiado de nuestros objetivos, basta con decir aquí que esta propiedad asegura la continuidad del espacio, en el sentido de que no presentaría “agujeros”.

2.4. Bases y transformaciones

2.4.1. Dependencia lineal y conjuntos generadores

Dado un conjunto de N vectores $X_0 = \{\mathbf{x}_i\}$, con $N < \infty$, se llama *combinación lineal* de los vectores \mathbf{x}_i a una expresión de la forma:

$$\mathbf{x} = \sum_{i=1}^N \alpha_i \mathbf{x}_i$$

donde los α_i son escalares.

Se dice que un vector \mathbf{x} es linealmente dependiente del conjunto de vectores X_0 si y sólo si se puede escribir a \mathbf{x} como una combinación lineal de los vectores \mathbf{x}_i . En caso contrario se dice que el vector \mathbf{x} es linealmente independiente del conjunto de vectores X_0 .

Al variar los coeficientes α_i , es decir, al generar todas las combinaciones lineales posibles de los \mathbf{x}_i , el resultado es un conjunto X de nuevos vectores \mathbf{x}_j que a su vez heredan muchas de las propiedades de los \mathbf{x}_i que los generaron. Si el nuevo conjunto X constituye a su vez un espacio vectorial \mathcal{X} , entonces se dice que el conjunto X_0 es un conjunto generador de este espacio, si para todo vector $\mathbf{x} \in X$ existe el conjunto de escalares $A = \{\alpha_i\}$ tales que \mathbf{x} se pueda expresar como una combinación lineal de los elementos de X_0 .

Un conjunto de vectores se dice que es *linealmente independiente* si la relación

$$\sum_{i=1}^N \alpha_i \mathbf{x}_i = \mathbf{0}$$

sólo puede satisfacerse siendo nulos todos los escalares α_i . Dicho de otro modo, un conjunto es linealmente independiente si ninguno de sus vectores puede expresarse como combinación lineal de los demás vectores del mismo conjunto.

2.4.2. Bases

Dado un espacio vectorial \mathcal{X} , se dice que el conjunto de vectores X_0 constituyen una *base* de \mathcal{X} si X_0 es linealmente independiente y genera a \mathcal{X} . Es decir, para que X_0 sea una base del espacio vectorial, cualquier vector perteneciente a \mathcal{X} debe poder escribirse como una combinación lineal de los vectores de X_0 y además, ninguno de los vectores de X_0 debe poder escribirse como una combinación lineal de los demás.

Se llama dimensión D de un espacio vectorial \mathcal{X} al número de vectores que tiene una base de dicho espacio. Se puede demostrar que cualquier subconjunto de $N > D$ vectores de \mathcal{X} , será linealmente dependiente. Cuando hablamos de señales nos interesa especialmente el espacio \mathbb{R}^N y un resultado útil en este caso es que todo conjunto de N vectores linealmente independientes en \mathbb{R}^N es una base para \mathbb{R}^N . Dado que todas las bases de un espacio vectorial tienen el mismo número de vectores, todas las bases para generar señales arbitrarias de N elementos deberán tener N señales de N elementos cada una.

2.4.3. Ortogonalidad y ortonormalidad

Se dice que un conjunto X_0 es *ortogonal* si se verifica que para todos sus elementos:

$$\begin{aligned}\langle \mathbf{x}_i, \mathbf{x}_j \rangle &= 0 & \forall i \neq j & \text{ y} \\ \langle \mathbf{x}_i, \mathbf{x}_j \rangle &= k & \forall i = j\end{aligned}$$

donde k es una constante escalar distinta de cero. En particular, si la constante $k = 1$, se dice que el conjunto es *ortonormal*. Si X_0 es además una base para espacio vectorial \mathcal{X} , esta base posee la ventaja de que cuando se quiere expresar un vector como una combinación lineal de los elementos de la base, los coeficientes α_i se puede obtener simplemente mediante el producto interno entre el vector y cada uno de los elementos de la base. Por ejemplo, si queremos expresar \mathbf{x} como una combinación lineal de la base ortonormal en \mathbb{R}^3 formada por $X_0 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$:

$$\mathbf{x} = \alpha_1 \mathbf{x}_1 + \alpha_2 \mathbf{x}_2 + \alpha_3 \mathbf{x}_3$$

Si quisiéramos obtener α_1 , se puede hacer el producto interno a ambos lados por \mathbf{x}_1 :

$$\langle \mathbf{x}, \mathbf{x}_1 \rangle = \alpha_1 \langle \mathbf{x}_1, \mathbf{x}_1 \rangle + \alpha_2 \langle \mathbf{x}_2, \mathbf{x}_1 \rangle + \alpha_3 \langle \mathbf{x}_3, \mathbf{x}_1 \rangle$$

Pero por ser una base ortogonal $\langle \mathbf{x}_2, \mathbf{x}_1 \rangle = \langle \mathbf{x}_3, \mathbf{x}_1 \rangle = 0$ y por ser *ortonormal* $\langle \mathbf{x}_1, \mathbf{x}_1 \rangle = 1$. De esta forma se puede obtener α_1 simplemente mediante la proyección $\alpha_1 = \langle \mathbf{x}, \mathbf{x}_1 \rangle$. Es decir, para cualquier i tenemos:

$$\alpha_i = \langle \mathbf{x}, \mathbf{x}_i \rangle = \sum_n x_n x_{in} \quad (2.3)$$

En esta forma, cada coeficiente es una medida del *parecido* entre el vector y el elemento correspondiente de la base. Como vimos antes, conceptualmente $\alpha_i = \langle \mathbf{x}, \mathbf{x}_i \rangle$ es la *componente* de la señal \mathbf{x} en \mathbf{x}_i . Para hacer una extensión a señales continuas, suponga que se quiere representar la señal $x(t)$ con una combinación lineal del conjunto ortogonal de señales $X_0 = \{x_1(t), x_2(t), \dots, x_N(t)\}$:

$$x(t) \approx \alpha_1 x_1(t) + \alpha_2 x_2(t) + \dots + \alpha_N x_N(t)$$

De forma similar al ejemplo anterior, haciendo el producto interno a ambos lados con una de las señales del conjunto:

$$\begin{aligned}\int x(t) x_i(t) dt &= \alpha_1 \int x_1(t) x_i(t) dt + \alpha_2 \int x_2(t) x_i(t) dt + \dots + \\ &+ \alpha_i \int x_i(t) x_i(t) dt + \dots + \alpha_N \int x_N(t) x_i(t) dt\end{aligned}$$

Por ser X_0 un conjunto ortogonal se cumple $\int x_i(t)x_j(t)dt = 0 \forall i \neq j$ y así se puede obtener:

$$\alpha_i = \frac{\int x(t)x_i(t)dt}{\int x_i(t)x_i(t)dt} = \frac{\langle \mathbf{x}, \mathbf{x}_i \rangle}{\langle \mathbf{x}_i, \mathbf{x}_i \rangle}$$

Si además el conjunto es ortonormal:

$$\alpha_i = \langle \mathbf{x}, \mathbf{x}_i \rangle = \int x(t)x_i(t)dt \quad (2.4)$$

2.4.4. Aproximación de señales

Queremos aproximar una señal $\mathbf{y} \in \mathbb{R}^M$ mediante una combinación lineal de los elementos del conjunto ortogonal $X_0 = \{\mathbf{x}_1, \dots, \mathbf{x}_k, \dots, \mathbf{x}_N\}$. En esta aproximación se desea encontrar los α_k de forma que el *error* entre la combinación lineal y la señal sea mínimo. Para realizar esta minimización es necesario definir un criterio para medir el error. Una forma intuitiva de medir ese error es el cuadrado de la longitud del vector diferencia entre la señal y su aproximación. Esta medida del error también se conoce como *error cuadrático total*:

$$\epsilon = \|\mathbf{e}\|_2^2 = \|\mathbf{y} - \tilde{\mathbf{y}}\|_2^2 = \left\| \mathbf{y} - \sum_{i=1}^N \alpha_i \mathbf{x}_i \right\|_2^2 = \sum_{j=1}^M \left(y_j - \sum_{i=1}^N \alpha_i x_{ij} \right)^2$$

Para encontrar el mínimo es necesario hacer $\nabla_{\alpha} \epsilon = 0$:

$$\begin{aligned}
 0 &= \frac{\partial \epsilon}{\partial \alpha_k} \\
 0 &= \frac{\partial \sum_{j=1}^M \left(y_j - \sum_{i=1}^N \alpha_i x_{ij} \right)^2}{\partial \alpha_k} \\
 0 &= 2 \sum_{j=1}^M \left(y_j - \sum_{i=1}^N \alpha_i x_{ij} \right) \underbrace{\frac{\partial \left(y_j - \sum_{i=1}^N \alpha_i x_{ij} \right)}{\partial \alpha_k}}_{=0 \quad \forall i \neq k} \\
 0 &= -2 \sum_{j=1}^M \left(y_j - \sum_{i=1}^N \alpha_i x_{ij} \right) x_{kj} \\
 0 &= \sum_{j=1}^M \left(y_j x_{kj} - \sum_{i=1}^N \alpha_i x_{ij} x_{kj} \right) \\
 0 &= \sum_{j=1}^M y_j x_{kj} - \sum_{j=1}^M \sum_{i=1}^N \alpha_i x_{ij} x_{kj} \\
 \sum_{j=1}^M \sum_{i=1}^N \alpha_i x_{ij} x_{kj} &= \sum_{j=1}^M y_j x_{kj} \\
 \sum_{i=1}^N \alpha_i \sum_{j=1}^M x_{ij} x_{kj} &= \langle \mathbf{y}, \mathbf{x}_k \rangle \\
 \sum_{i=1}^N \alpha_i \underbrace{\langle \mathbf{x}_i, \mathbf{x}_k \rangle}_{=0 \quad \forall i \neq k} &= \langle \mathbf{y}, \mathbf{x}_k \rangle \\
 \alpha_k \langle \mathbf{x}_k, \mathbf{x}_k \rangle &= \langle \mathbf{y}, \mathbf{x}_k \rangle \\
 \alpha_k &= \frac{\langle \mathbf{y}, \mathbf{x}_k \rangle}{\langle \mathbf{x}_k, \mathbf{x}_k \rangle}
 \end{aligned}$$

Además, si el conjunto X_0 es ortonormal obtenemos $\alpha_k = \langle \mathbf{y}, \mathbf{x}_k \rangle$. Con este resultado se generalizan las ecuaciones (2.3) y (2.4) demostrando que el conjunto de coeficientes obtenidos mediante proyecciones ortogonales minimiza el criterio del error en un espacio euclídeo.

Por ejemplo, suponga que queremos aproximar \mathbf{v}_1 con \mathbf{v}_2 en \mathbb{R}^2 , es decir, $\tilde{\mathbf{v}}_1 = \alpha \mathbf{v}_2$. A partir de las ecuaciones anteriores podemos calcular α como:

$$\alpha = \frac{\langle \mathbf{v}_1, \mathbf{v}_2 \rangle}{\langle \mathbf{v}_2, \mathbf{v}_2 \rangle} = \frac{\langle \mathbf{v}_1, \mathbf{v}_2 \rangle}{\|\mathbf{v}_2\|_2 \|\mathbf{v}_2\|_2} = \frac{\|\mathbf{v}_1\|_2 \cos(\theta)}{\|\mathbf{v}_2\|_2}$$

Es decir, el valor de α es la proyección ortogonal de \mathbf{v}_1 sobre \mathbf{v}_2 normalizada por la longitud de \mathbf{v}_2 . Esto se puede apreciar en el diagrama de vectores de la Figura 2.8, donde se pueden ver $\mathbf{v}_1, \mathbf{v}_2$ y $\tilde{\mathbf{v}}_1$, la aproximación de \mathbf{v}_1 en la dirección de \mathbf{v}_2 . También se aprecia el error de la aproximación ortogonal $\epsilon = \|\tilde{\mathbf{v}}_1 - \mathbf{v}_1\|$ y otros errores de proyecciones no ortogonales que —como las que se indican en líneas punteadas— siempre son mayores.

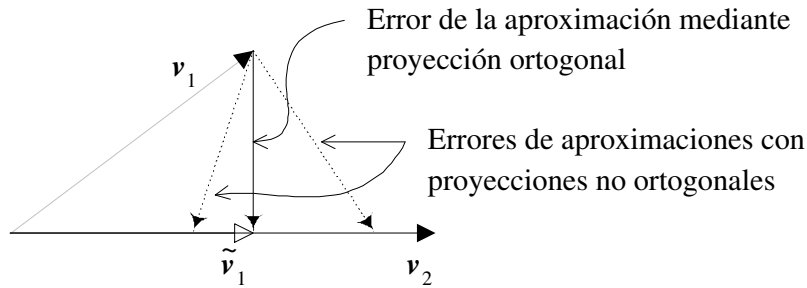


Figura 2.8. Aproximación de vectores utilizando proyecciones ortogonales. Se puede observar que la proyección es la que minimiza el error en el sentido de la norma euclídea .

Note que no estamos diciendo que el vector \mathbf{v}_2 sea el mejor para aproximar \mathbf{v}_1 , sino que la mejor manera de calcular α es a través de una proyección ortogonal.

Esto mismo que hemos demostrado para señales discretas, se verifica para señales continuas. Por ejemplo, suponga que queremos representar una señal continua, definida en el intervalo $[-1, 1]$ según:

$$y(t) = \begin{cases} -1 & \forall t < 0 \\ 1 & \forall t \geq 0 \end{cases}$$

Para representar esta señal podemos usar un conjunto de funciones de Legendre, que son ortonormales en el intervalo $[-1, 1]$. Estas funciones se pueden calcular a partir de las siguientes ecuaciones:

$$\begin{aligned}
 \phi_0(t) &= \frac{1}{\sqrt{2}} \\
 \phi_1(t) &= \sqrt{\frac{3}{2}} t \\
 \phi_2(t) &= \sqrt{\frac{5}{2}} \left(\frac{3}{2} t^2 - \frac{1}{2} \right) \\
 \phi_3(t) &= \sqrt{\frac{7}{2}} \left(\frac{5}{2} t^3 - \frac{3}{2} t \right) \\
 &\vdots \\
 \phi_n(t) &= \sqrt{\frac{2n+1}{2}} \frac{1}{2^n n!} \frac{d^n}{dt^n} (t^2 - 1)^n
 \end{aligned}$$

Utilizando las cuatro primeras funciones y el producto interno se puede encontrar una representación aproximada de la señal propuesta:

$$\begin{aligned}
 \alpha_0 &= \int_{-1}^1 \sqrt{\frac{1}{2}} y(t) dt = 0 \\
 \alpha_1 &= \int_{-1}^1 \sqrt{\frac{3}{2}} t y(t) dt = \sqrt{\frac{3}{2}} \\
 \alpha_2 &= \int_{-1}^1 \sqrt{\frac{5}{2}} \left(\frac{3}{2} t^2 - \frac{1}{2} \right) y(t) dt = 0 \\
 \alpha_3 &= \int_{-1}^1 \sqrt{\frac{7}{2}} \left(\frac{5}{2} t^3 - \frac{3}{2} t \right) y(t) dt = -\sqrt{\frac{7}{32}}
 \end{aligned}$$

La aproximación utilizando estos α_k queda de la forma:

$$\begin{aligned}
 y(t) &\approx \sqrt{\frac{3}{2}} \left(\sqrt{\frac{3}{2}} t \right) + \left(-\sqrt{\frac{7}{32}} \right) \left(\sqrt{\frac{7}{2}} \left(\frac{5}{2} t^3 - \frac{3}{2} t \right) \right) \\
 &= \frac{45}{16} t - \frac{35}{16} t^3
 \end{aligned}$$

Se puede demostrar que si aumentamos el número de funciones aproximantes, el error se irá reduciendo. En el caso de señales muestreadas, tendremos vectores de \mathbb{R}^M y la aproximación será exacta si se usan M

vectores linealmente independientes en \mathbb{R}^M . Como ya se mencionó, M vectores linealmente independientes de dimensión M generan el espacio \mathbb{R}^M , y por lo tanto constituyen una base.

2.4.5. Cambio de base

Para un espacio vectorial dado existen infinitas bases. De todas estas bases, cuando representamos una señal simplemente mediante $\mathbf{x} = [4, 8, 9]$, estamos utilizando implícitamente la base canónica:

$$X_e = \{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}$$

ya que la señal \mathbf{x} puede escribirse como:

$$\mathbf{x} = \mathbf{e}_1\alpha_1 + \mathbf{e}_2\alpha_2 + \mathbf{e}_3\alpha_3 = \mathbf{e}_14 + \mathbf{e}_28 + \mathbf{e}_39$$

Para hacer una referencia explícita a la base en que se representa la señal usaremos la notación $\mathbf{x}_{X_e} = [\alpha_1, \alpha_2, \alpha_3]$. Sin embargo, también podríamos representar esta misma señal en otra base de \mathbb{R}^3 . Por ejemplo, si utilizáramos la base *ortonormal*:

$$X_1 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\} = \left\{ \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \end{bmatrix}, \begin{bmatrix} -1/\sqrt{6} \\ 1/\sqrt{6} \\ 2/\sqrt{6} \end{bmatrix}, \begin{bmatrix} 1/\sqrt{3} \\ -1/\sqrt{3} \\ 1/\sqrt{3} \end{bmatrix} \right\}$$

podemos expresar la señal \mathbf{x} como $\mathbf{x} = \mathbf{x}_1\beta_1 + \mathbf{x}_2\beta_2 + \mathbf{x}_3\beta_3$ donde simplemente $\beta_i = \langle \mathbf{x}, \mathbf{x}_i \rangle$ (por ser una base ortonormal). Utilizando este producto interno encontramos la representación de la señal \mathbf{x} en la base X_1 como:

$$\mathbf{x}_{X_1} = [\beta_1, \beta_2, \beta_3] = [6\sqrt{2}, \frac{11}{3}\sqrt{6}, \frac{5}{3}\sqrt{3}]$$

Hay que tener en cuenta que tanto \mathbf{x}_{X_1} como \mathbf{x}_{X_e} son la misma señal \mathbf{x} vista desde diferentes perspectivas. Ambas representaciones contienen la misma información pero se ha efectuado un cambio de base. Podemos expresar \mathbf{x}_{X_e} a partir de \mathbf{x}_{X_1} como:

$$\mathbf{x}_{X_e} = \mathbf{x}_1\beta_1 + \mathbf{x}_2\beta_2 + \mathbf{x}_3\beta_3 = \mathbf{x}_16\sqrt{2} + \mathbf{x}_2\frac{11}{3}\sqrt{6} + \mathbf{x}_3\frac{5}{3}\sqrt{3} = [4, 8, 9]$$

que en la forma matricial nos queda:

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \end{bmatrix} \begin{bmatrix} -1/\sqrt{6} \\ 1/\sqrt{6} \\ 2/\sqrt{6} \end{bmatrix} \begin{bmatrix} 1/\sqrt{3} \\ -1/\sqrt{3} \\ 1/\sqrt{3} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix}$$

o simplemente $\mathbf{x}_{X_e} = \mathbf{M}\mathbf{x}_{X_1}$, donde la matriz \mathbf{M} contiene como columnas los vectores de la base X_1 . Ahora observe que en el cálculo de los coeficientes $\beta_i = \langle \mathbf{x}, \mathbf{x}_i \rangle$ también podemos expresar los productos internos como un producto matricial:

$$\begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \end{bmatrix} \\ \begin{bmatrix} -1/\sqrt{6} & 1/\sqrt{6} & 2/\sqrt{6} \end{bmatrix} \\ \begin{bmatrix} 1/\sqrt{3} & -1/\sqrt{3} & 1/\sqrt{3} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix}$$

o simplemente $\mathbf{x}_{X_1} = \mathbf{M}^T \mathbf{x}_{X_e}$.

Podemos generalizar este ejemplo mediante la definición de cambio de base. Sean $X_0 = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ y $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$ dos bases *ordenadas* para el mismo espacio vectorial \mathcal{V} de dimensión N (una base ordenada de \mathcal{V} es una base de \mathcal{V} en la cual se ha establecido un orden), para cualquier vector \mathbf{v} en \mathcal{V} las coordenadas de \mathbf{v} en la base X_0 , \mathbf{v}_{X_0} , y las coordenadas de \mathbf{v} en la base Y_0 , \mathbf{v}_{Y_0} , se relacionan por:

$$\begin{aligned} \mathbf{v}_{X_0} &= \mathbf{M}\mathbf{v}_{Y_0} \\ \mathbf{v}_{Y_0} &= \mathbf{M}^{-1}\mathbf{v}_{X_0} \end{aligned}$$

donde \mathbf{M} es la matriz no singular de $N \times N$ cuyos elementos están dados por:

$$\mathbf{y}_i = m_{1i}\mathbf{x}_1 + \dots + m_{Ni}\mathbf{x}_N$$

es decir, los elementos m_{ki} son los coeficientes que permiten expresar el vector \mathbf{y}_i de la base Y_0 , como combinación lineal de los vectores \mathbf{x}_k de la base X_0 .

La matriz \mathbf{M} se denomina matriz de *transición* o matriz de *cambio de base* X_0 a Y_0 . Su inversa será la matriz de transición de la base Y_0 a la base X_0 . En el ejemplo anterior debe notarse que en lugar de \mathbf{M}^{-1} hemos usado \mathbf{M}^T . Esto fue posible porque la base era ortonormal, y en este caso puede demostrarse que la inversa de la matriz de transición es igual a su transpuesta. En la Figura 2.9 se resume el proceso del cambio de base desde una perspectiva más cercana al procesamiento de señales mediante transformaciones lineales.

Siempre que las funciones de la base cumplan con ciertas condiciones, es posible extender *conceptualmente* esta definición a señales continuas teniendo en cuenta que:

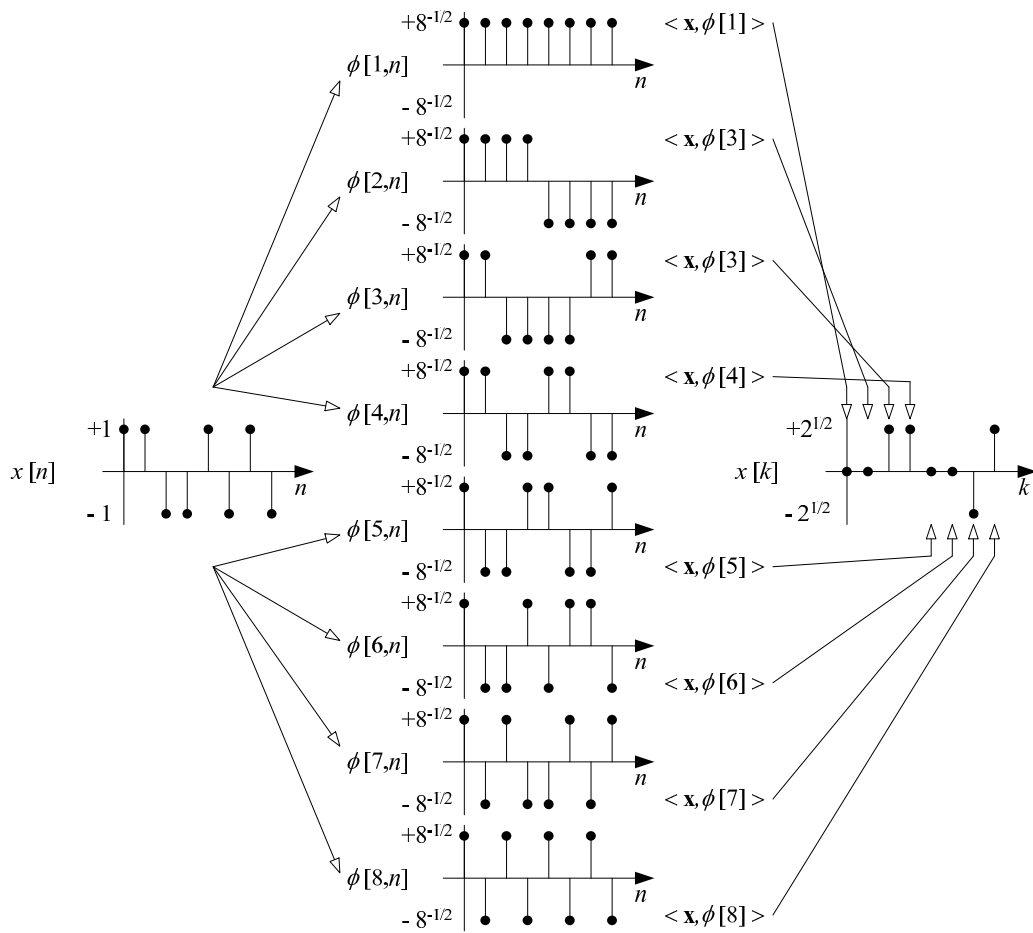


Figura 2.9. El cambio de base visto desde la perspectiva del procesamiento de señales.

1. cada elemento de la base es ahora una señal en \mathbb{R}^∞ , que puede representarse como $\mathbf{x}_i = [x_i(t)]$,
2. la base debe contar con infinitos elementos para poder generar el espacio \mathbb{R}^∞ , ahora resultará más adecuado representar los elementos de esta base como $[\phi(f, t)]$, que equivaldría a pensar en una matriz de $\infty \times \infty$ elementos que se acceden con los *índices continuos* f y t ,
3. en el cambio de base deberemos hacer infinitos productos internos, cada uno de los cuales se calcula mediante:

$$y(f) = \int_{-\infty}^{\infty} x(t)\phi(f, t)dt.$$

Como ya mencionamos la señal en cualquier base es la misma. Un cambio de base no modifica la información presente en la señal, sólo la forma en que ésta es presentada, ya que simplemente se trata de una proyección en términos de otra base. En particular podemos verificar qué sucede con la energía de la señal en ambas bases:

$$\begin{aligned} E(\mathbf{x}_{X_e}) &= \|[4, 8, 9]\|_2^2 = 161 \\ E(\mathbf{x}_{X_1}) &= \|[6\sqrt{2}, \frac{11}{3}\sqrt{6}, \frac{5}{3}\sqrt{3}]\|_2^2 = 161 \end{aligned}$$

Generalizando estas ideas se llega a la relación de Parseval, según la cual la energía se conserva ante un cambio entre bases ortonormales:

$$E(\mathbf{x}) = \sum_{i=1}^n \alpha_i^2 = \sum_{i=1}^n \beta_i^2$$

Si alguna de las bases es solamente ortogonal, entonces no siempre se cumple que el producto interno entre dos de sus elementos es 1. Si $\langle \mathbf{x}_i, \mathbf{x}_i \rangle = k_i$ entonces tendríamos:

$$E(\mathbf{x}) = \sum_{i=1}^n \alpha_i^2 = \sum_{i=1}^n \beta_i^2 k_i$$

En el caso de que se quieran aprovechar todas las ventajas de la ortonormalidad, existen diferentes métodos que permiten obtener una base ortonormal; entre estos se encuentra el proceso de ortonormalización de Gram-Schmidt.

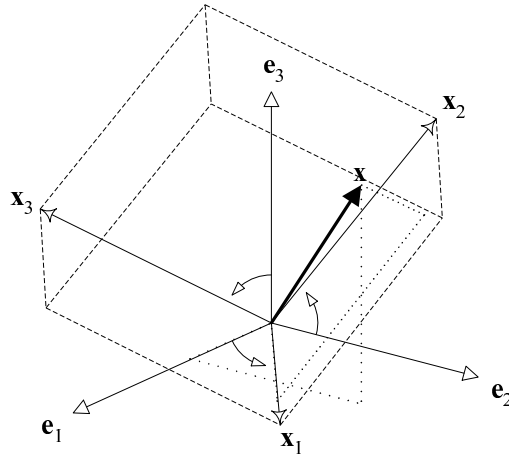


Figura 2.10. Un cambio de base no modifica al vector. Aquí se pueden observar la base canónica y la base ortonormal del ejemplo desarrollado en el texto. En este caso el cambio de base implica simplemente una rotación de los ejes coordenados.

2.4.6. Transformaciones lineales

Una transformación lineal entre dos espacios vectoriales \mathcal{X} y \mathcal{Y} es una correspondencia que asigna a cada vector \mathbf{x} de \mathcal{X} un vector $\mathcal{T}(\mathbf{x})$ en \mathcal{Y} de manera tal que:

$$\mathcal{T}(a_1\mathbf{x}_1 + a_2\mathbf{x}_2) = a_1\mathcal{T}(\mathbf{x}_1) + a_2\mathcal{T}(\mathbf{x}_2)$$

para todos los vectores \mathbf{x}_1 y \mathbf{x}_2 de \mathcal{X} y los escalares a_1 y a_2 .

Nótese que una transformación lineal mapea señales en un espacio \mathcal{X} dentro de otro espacio \mathcal{Y} . Nótese también que \mathcal{X} y \mathcal{Y} pueden ser cualquier espacio vectorial, incluso puede ser que $\mathcal{X}=\mathcal{Y}$.

Hay dos resultados que son especialmente interesantes. En primer lugar, para que una transformación lineal sea uno a uno (es decir, a cada vector en \mathcal{X} le corresponda un solo vector en \mathcal{Y} , y por lo tanto la transformación sea invertible) basta con verificar que al transformar el vector $\mathbf{0}$ del espacio \mathcal{X} , se obtiene un solo vector, el vector $\mathbf{0}$ en \mathcal{Y} . El otro resultado que nos será de utilidad es que para conocer el efecto de una transformación lineal sobre cualquier vector en \mathcal{X} basta con conocer el efecto de la transformación en los vectores de una base de \mathcal{X} . Esto se puede deducir dado que cualquier vector \mathbf{x} en \mathcal{X} se puede escribir como combinación lineal de los vectores de su base y, por la propiedad de linealidad de la transformación lineal, el vector transformado será una combinación lineal de los vectores de la base transformados. Esto es:

$$\begin{aligned}
 \mathbf{x} &= \alpha_1 \mathbf{x}_1 + \alpha_2 \mathbf{x}_2 + \cdots + \alpha_N \mathbf{x}_N \\
 \mathcal{T}(\mathbf{x}) &= \mathcal{T}(\alpha_1 \mathbf{x}_1 + \alpha_2 \mathbf{x}_2 + \cdots + \alpha_N \mathbf{x}_N) \\
 \mathbf{y} &= \alpha_1 \mathcal{T}(\mathbf{x}_1) + \alpha_2 \mathcal{T}(\mathbf{x}_2) + \cdots + \alpha_N \mathcal{T}(\mathbf{x}_N) \\
 \mathbf{y} &= \alpha_1 \mathbf{y}_1 + \alpha_2 \mathbf{y}_2 + \cdots + \alpha_N \mathbf{y}_N
 \end{aligned}$$

Este último resultado nos permite asegurar que dada una transformación lineal $\mathcal{T} : \mathbb{R}^N \rightarrow \mathbb{R}^M$ existe una sola matriz \mathbf{A} de $M \times N$ tal que:

$$\mathcal{T}(\mathbf{v}) = \mathbf{A}\mathbf{v},$$

En la sección anterior, estudiamos los cambios de bases. Nótese ahora, que los cambios de base son un tipo especial de transformaciones lineales con características muy interesantes para el procesamiento de señales: son uno a uno, son invertibles y el espacio vectorial \mathcal{X} es igual al espacio vectorial \mathcal{Y} . Además, al trabajar con señales discretas, encontramos que las transformaciones siempre tendrán una representación matricial y la matriz de transformación o de cambio de base será cuadrada ($N \times N$ para un espacio \mathbb{R}^N). La transformación inversa se obtendrá simplemente a partir de la inversa de dicha matriz.

2.5. Preguntas

1. ¿Cómo se puede interpretar que las señales digitales de N muestras son puntos en un espacio \mathbb{R}^N ?
2. ¿Por qué decimos que las señales continuas son puntos en el espacio \mathbb{R}^∞ ?
3. ¿Cuáles son las ventajas de poder ver a las señales como puntos en un espacio \mathbb{R}^N ?
4. ¿Para qué sirven las normas? ¿Por qué hay distintas normas?
5. ¿Cómo se puede definir una norma- p para $p = 0$? ¿Qué utilidad tendría una norma como ésta?
6. ¿Cómo puede interpretarse gráficamente la norma $p = \infty$?

7. ¿Cuál es la relación que existe entre las normas p y las medidas físicas de acción, energía, potencia, raíz del valor cuadrático medio y amplitud?
8. ¿Cómo se relacionan las definiciones de normas y producto interno en el caso de las señales discretas y las continuas? ¿Cuál es el equivalente discreto del dt que aparece en el caso de señales continuas?
9. ¿En qué casos es necesario utilizar el conjugado en la definición del producto interno?
10. ¿Por qué decimos que el producto interno mide el parecido entre dos señales? Analízelo primero para señales en \mathbb{R}^2 y luego extienda el análisis a señales continuas.
11. ¿Cual es la relación entre producto interno y proyección?
12. ¿Qué diferencia hay entre conjunto y espacio de señales?
13. Un mismo conjunto de señales con dos métricas diferentes ¿conforma dos espacios de señales diferentes?
14. ¿Qué utilidad tiene definir un espacio vectorial en el análisis de señales?
15. ¿Cómo se verifican las propiedades de cerradura en un espacio vectorial?
16. ¿Qué ventajas tiene el hecho de que una base sea ortonormal?
17. ¿Puede una base estar formada por señales linealmente dependientes? ¿y por señales no ortogonales?
18. Demostrar que las proyecciones ortogonales minimizan el criterio del error cuadrático en la aproximación de señales.
19. ¿Por que decimos que un cambio de base es un caso particular de transformación lineal?
20. ¿Bajo qué condiciones se puede asegurar que un cambio de base es simplemente una rotación de las señales de la base canónica?
21. ¿Cómo se puede interpretar geoméricamente el teorema Parseval a partir de considerar a una transformación con base ortonormal como un simple cambio de base? ¿Qué sucede si la base es simplemente ortogonal?

22. Exprese la transformación lineal de una señal discreta y su correspondiente transformación inversa mediante productos matriciales.
23. Indique cómo se aplica en esta transformación la idea de parecido entre señales y su medida a través del producto interno.
24. Escriba las ecuaciones y dé ejemplos de una base para transformaciones lineales en las que:
 - a) La señal en el dominio original y la señal transformada son discretas.
 - b) La señal en el dominio original es continua y la señal transformada discreta.
 - c) La señal en el dominio original es discreta y la señal transformada continua.
 - d) La señal en el dominio original y la señal transformada son continuas.
25. En cada uno de los casos anteriores indique en qué espacio ($\mathbb{R}^?$) están las señales de la base y en qué espacio ($\mathbb{R}^{? \times ?}$) está la matriz de la transformación.
26. Muestre con un ejemplo sencillo que para las transformaciones lineales en las que los vectores de la base son ortonormales, la matriz de inversión es la transpuesta de la matriz de transformación.

2.6. Trabajos prácticos

Ejercicio 1: Obtener los siguientes valores de una señal senoidal, una rampa, una onda cuadrada y una señal aleatoria:

1. valor medio,
2. máximo,
3. mínimo,
4. amplitud,
5. energía,
6. acción,
7. potencia media y

8. raíz del valor cuadrático medio.

Ejercicio 2: Defina matemáticamente el espacio de las señales senoidales y compruebe numéricamente si se trata de un espacio vectorial.

Ejercicio 3: Compruebe en el espacio del ejercicio anterior que el producto interno mide el grado de parecido entre dos señales (genere senoidales de distinta frecuencia y realice el producto interno entre ellas).

Ejercicio 4: Defina un espacio vectorial de señales complejas (formado por señales no necesariamente periódicas) y verifique que se trata de un espacio vectorial. Utilice el producto interno para medir el grado de parecido en este espacio.

Ejercicio 5: (*) Calcule el error cuadrático total de aproximación en el ejemplo con funciones de Legendre (página 61) bajo las siguientes condiciones:

1. con los coeficientes calculados en el ejemplo,
2. con pequeñas variaciones en torno a estos coeficientes α , construyendo una gráfica en 3D con la variación en los coeficientes en x, y y el error cuadrático total en z ,
3. con más coeficientes α , para comprobar cómo se reduce el error cuadrático total al aumentar los coeficientes.

Ejercicio 6: (*) Genere una señal como combinación lineal del conjunto de señales senoidales con frecuencias de 1, 2, 3, 4, 5, 6, 7, 8, 9 y 10 Hz y luego:

1. mida el grado de parecido con dichas senoidales representando el resultado en un gráfico de barras,
2. vuelva a medir el grado de parecido pero con una combinación lineal en la que se varía la fase de las senoidales y
3. realice el gráfico de barras para el caso de una señal cuadrada de 5,5 Hz.

Ejercicio 7: ^(**) En el archivo `te.txt` se encuentra la señal registrada al discar un número telefónico en una línea ruidosa y se requiere determinar el número que se ha discado. La señal se digitalizó con una frecuencia de muestreo de 11025 Hz y se sabe que cada número del teléfono está codificado mediante la suma de dos señales senoidales cuya frecuencia indica la posición en el teclado. De arriba hacia abajo las frecuencias son 697, 770, 852 y 941 Hz; de izquierda a derecha son 1209, 1336 y 1477 Hz. Por ejemplo: el número 2 se codifica con la suma de dos senos con frecuencias 697 y 1336 Hz; el número 7 se codifica con 852 y 1209 Hz. Se necesita determinar el número que se ha discado. (Sugerencia: utilice el producto interno).

Bibliografía

- [1] L.E. Franks. *Teoría de la señal*. Reverté, Barcelona, 1975.
- [2] R.A. Gabel and R.A. Roberts. *Señales y sistemas lineales*. Ed. Limusa S.A., México, 1975.
- [3] S. Grossman. *Álgebra lineal*. Grupo Ed. Iberoamericana, 1988.
- [4] H. Kwakernaak, R. Sivan, and R.C.W. Strijbos. *Modern Signals and Systems*. Prentice Hall, New Jersey, 1991.
- [5] B. Lathi. *Modern Digital and Analog Communication Systems*. Holt, Rinehart & Winston, 1983.
- [6] B.Ñoble and J. Daniel. *Álgebra lineal aplicada*. Prentice-Hall, 1989.

Capítulo 3

Transformada Discreta de Fourier

Leandro Di Persia, Diego Milone

Temas a tratar

- Series y transformadas de Fourier
- Teorema del muestreo y fenómeno de alias
- Ventanas temporales y resolución en tiempo–frecuencia
- Transformada Rápida de Fourier

Objetivos

- Aplicar los conceptos de producto interno y transformaciones lineales al caso de la Transformada Discreta de Fourier (TDF).
- Reinterpretar el fenómeno de alias desde la perspectiva del análisis frecuencial.
- Aplicar la TDF a ejemplos sencillos y aplicaciones con señales reales.

3.1. Introducción

La serie y la Transformada de Fourier desempeñan un rol importante en la representación de señales y en el análisis de sistemas lineales en el mundo analógico. Con el advenimiento de la computadora digital y las facilidades que esta proporciona para el diseño de algoritmos, fue deseable y necesario extender el uso de este tipo de herramientas matemáticas al mundo discreto. Como resultado de esto se desarrolló lo que se conoce como la Transformada Discreta de Fourier (TDF). A continuación vamos a analizar cómo surge esta transformación a partir de conceptos previamente estudiados.

Supongamos que se desea analizar una señal de N muestras $x[n]$, pero no podemos extraer la información que nos interesa en la base que está representada. Entonces nos planteamos la posibilidad de realizar un *cambio de base* de manera tal que la información se represente de otra forma. Como se estableció en el capítulo anterior, un cambio de base es un caso particular de *transformación lineal* uno a uno, invertible. También se pudo apreciar que, desde un punto de vista conceptual, este tipo de transformaciones funciona esencialmente como una *rotación* de los ejes de coordenadas (vectores de la base). Vimos que un conjunto de vectores, para constituir una base, debe ser linealmente independiente (es decir, que ninguno de ellos debe poder escribirse como combinación lineal de los restantes). También vimos que debe ser un conjunto generador (o sea que cualquier vector del espacio vectorial pueda escribirse como combinación lineal de ellos).

La señal $x[n]$ está en \mathbb{R}^N , y ya sabemos que en dicho espacio puede ser *generado* por N vectores linealmente independientes. Entonces, lo que nos está faltando para poder formar una base para el cambio de base es hallar N señales linealmente independientes en \mathbb{R}^N . Una propiedad interesante de la independencia lineal es que si un conjunto de señales es ortogonal, entonces puede demostrarse que también es linealmente independiente.

Nuestro interés al hacer esta transformación es lograr representar la información que contiene la señal $x[n]$, de tal manera que resulte más sencillo analizarla, extraer datos de ella, o incluso procesarla de manera de eliminar información que no nos interesa (por ejemplo, ruido). O sea que las señales que constituyan nuestra base no pueden ser cualquiera, sino que deben permitirnos representar nuestra señal en una forma más útil.

3.2. Familia de bases de Fourier

El análisis de Fourier abarca diversas formas de transformaciones que son aplicables a señales de diferente tipo, tanto continuas como discretas, ya sea en dominio temporal o frecuencial, y para señales periódicas y no periódicas. Todas ellas consisten en la representación de la señal original en función de una base sinusoidal compleja completa. Vamos a hacer un breve repaso de las diversas transformaciones existentes.

3.2.1. Series seno

Mediante las series seno es posible descomponer (transformar) señales continuas, reales, periódicas y con simetría *impar*. La base para la expansión está dada por:

$$\phi[k](t) = \sin(2\pi k f_0 t)$$

y las ecuaciones de transformación y reconstrucción son:

$$X[k] = \frac{2}{T_0} \int_{-T_0/2}^{T_0/2} x(t) \sin(2\pi k f_0 t) dt$$

$$x(t) = \sum_{k=0}^{\infty} x[k] \sin(2\pi k f_0 t)$$

donde T_0 es el período de la señal y $f_0 = 1/T_0$ es su frecuencia.

De las ecuaciones se ve que en el dominio frecuencial se trata de una representación *discreta* ya que sólo tienen definidos sus valores a intervalos múltiplos enteros de la frecuencia fundamental f_0 (ver Figura 3.1).

3.2.2. Series coseno

Mediante esta serie se pueden descomponer señales continuas, reales, periódicas y con simetría *par*. La base para la expansión está dada por:

$$\phi[k](t) = \cos(2\pi k f_0 t)$$

y las ecuaciones de transformación y reconstrucción son:

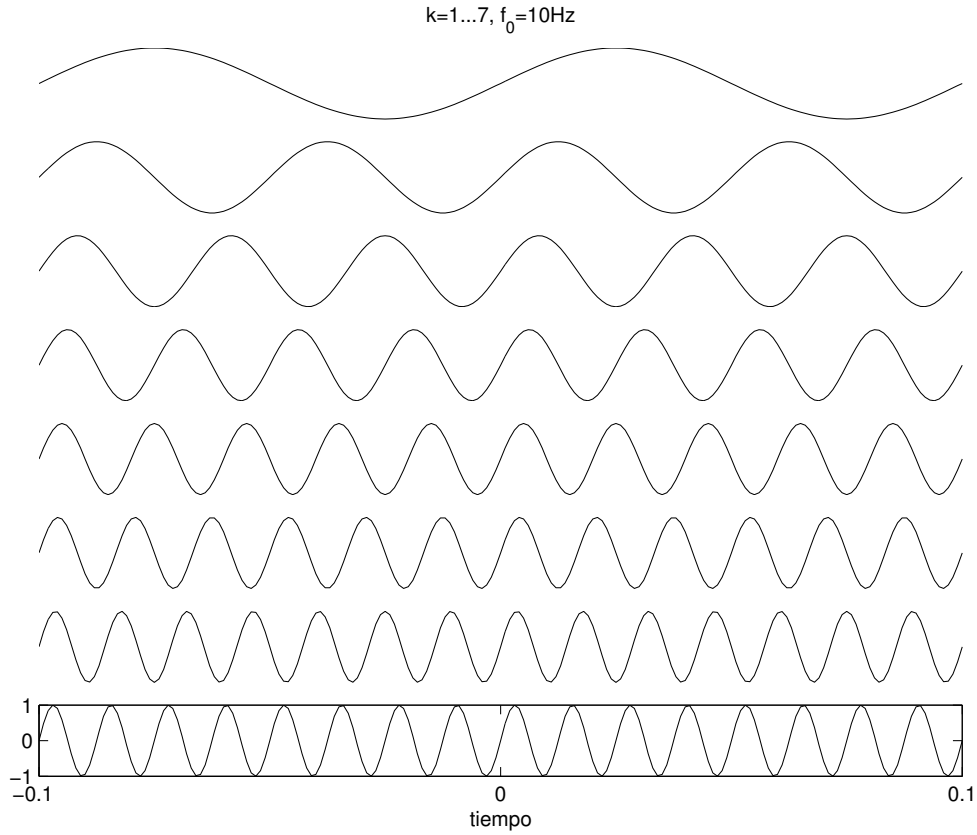


Figura 3.1. Algunos elementos de la base para la serie seno.

$$X[k] = \frac{2}{T_0} \int_{-T_0/2}^{T_0/2} x(t) \cos(2\pi k f_0 t) dt$$

$$x(t) = \sum_{k=0}^{\infty} x[k] \cos(2\pi k f_0 t)$$

donde T_0 es el período de la señal y $f_0 = 1/T_0$ es su frecuencia.

Al igual que la serie seno, se trata de una representación discreta en el dominio frecuencial (ver Figura 3.2).

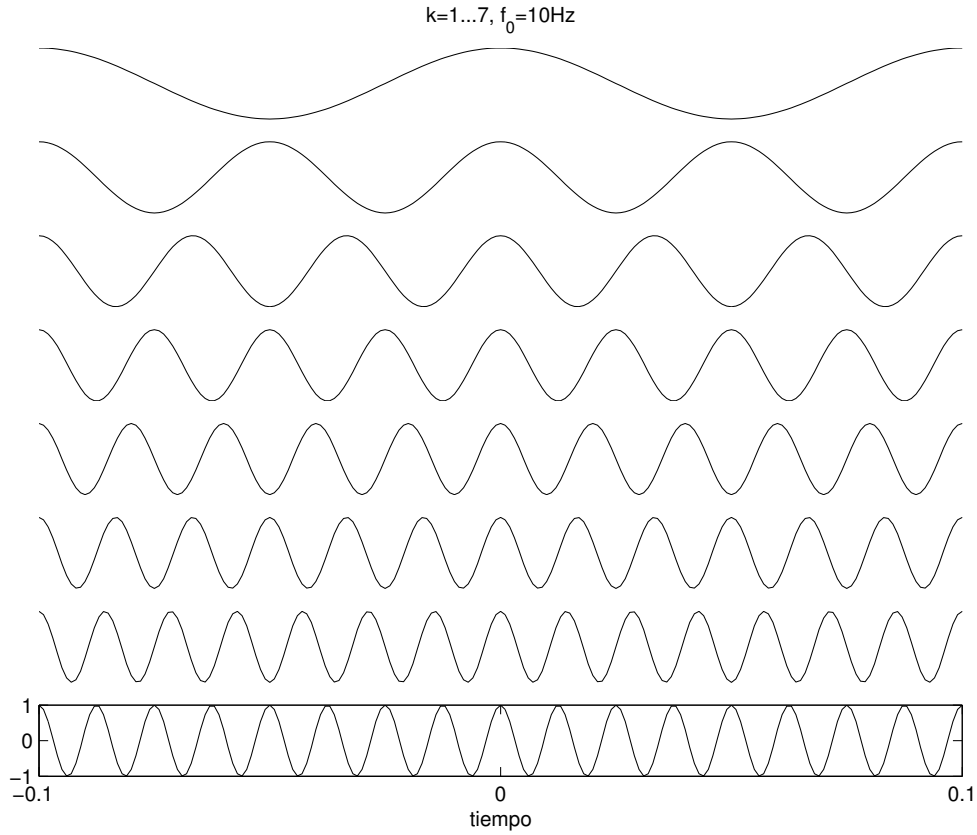


Figura 3.2. Algunos elementos de la base para la serie coseno.

3.2.3. Serie exponencial de Fourier

La serie exponencial de Fourier permite descomponer señales continuas y periódicas en general, sin importar su paridad. La base para la expansión está dada por:

$$\phi[k](t) = e^{j2\pi k f_0 t}$$

y las ecuaciones de transformación y reconstrucción son:

$$X[k] = \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} x(t) e^{-j2\pi k f_0 t} dt$$

$$x(t) = \sum_{k=-\infty}^{\infty} x[k] e^{j2\pi k f_0 t}$$

donde T_0 es el período de la señal y $f_0 = 1/T_0$ es su frecuencia fundamental.

Al igual que las series seno y coseno, se trata de una representación discreta en el dominio frecuencial (ver Figura III.3).

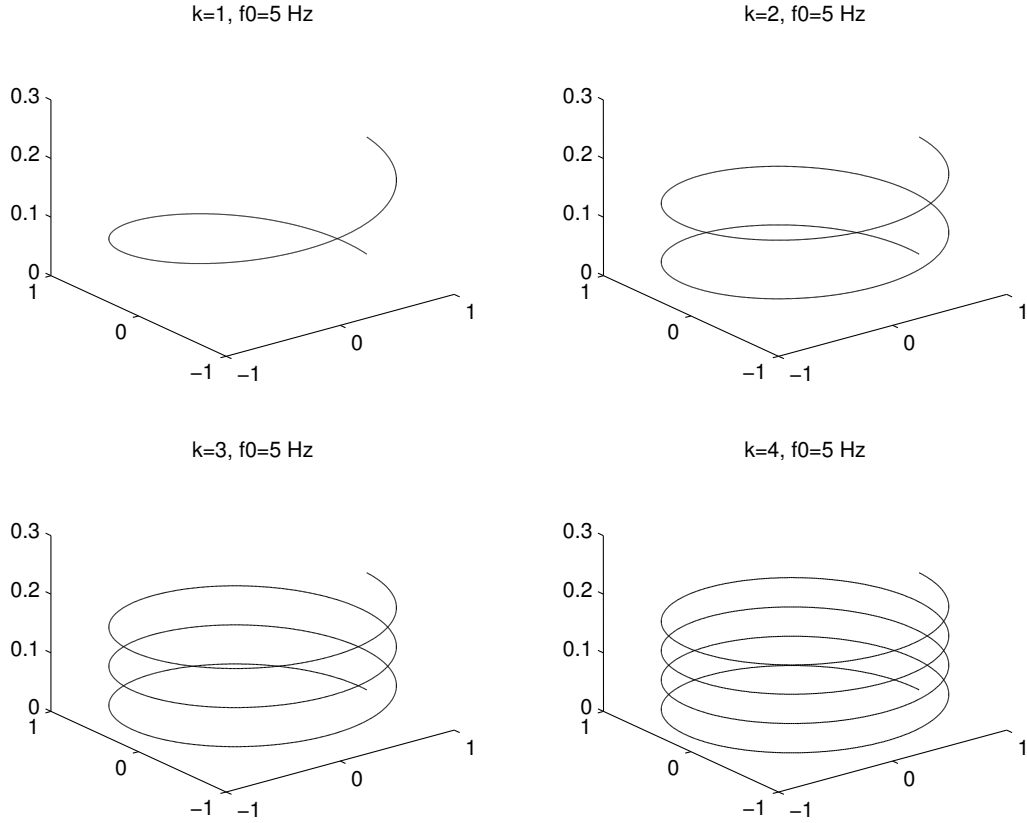


Figura 3.3. Cuatro elementos de la base para la serie exponencial Fourier.

3.2.4. Transformada de Fourier de tiempo discreto

La Transformada de Fourier de Tiempo Discreto (TFTD) se aplica a señales muestreadas en el tiempo, no periódicas y de duración infinita. La base para la TFTD es:

$$\phi[f](n) = e^{j2\pi fn}$$

Suponiendo que la señal $x[n]$ ha sido muestreada a intervalos regulares de tiempo T , las ecuaciones de transformación y reconstrucción son,

respectivamente:

$$\begin{aligned} X(f) &= \sum_{n=-\infty}^{\infty} x[n]e^{-j2\pi fn} \\ x[n] &= \int_{-\frac{1}{2T}}^{\frac{1}{2T}} X(f)e^{j2\pi fn} df \end{aligned}$$

Vemos claramente que en el dominio temporal la señal es *discreta*. Por otro lado, en el dominio frecuencial $X(f)$ es continua y *periódica* (nótese que la integral se hace sobre un período).

3.2.5. Transformada Continua de Fourier

La Transformada Continua de Fourier (TCF) se aplica a señales continuas no periódicas, aunque puede extenderse a señales periódicas utilizando funciones generalizadas como el $\delta(t)$. La base para esta transformación es:

$$\phi(f, t) = e^{j2\pi ft}$$

Las ecuaciones de transformación y reconstrucción son, respectivamente:

$$\begin{aligned} X(f) &= \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft} dt \\ x(t) &= \int_{-\infty}^{\infty} X(f)e^{j2\pi ft} df \end{aligned}$$

En este caso, se ve claramente que las señales, tanto en el dominio temporal como en dominio frecuencial, son continuas.

3.3. Exponenciales complejas discretas

En vista de las transformaciones presentadas en la sección anterior, sería deseable contar con una base generada a partir de exponenciales complejas que permita trabajar con señales digitales. Para esto, es necesario que dicha base sea discreta tanto en el dominio temporal como en el frecuencial.

Como estamos trabajando con señales en \mathbb{R}^N nuestra base debe consistir en N vectores de N muestras cada uno. Una exponencial compleja continua está dada por:

$$\phi(\omega, t) = e^{j\omega t}$$

donde ω es la frecuencia angular. En este caso, para cada valor de ω la exponencial generada es diferente de las demás.

Para una exponencial compleja discreta tenemos la siguiente ecuación:

$$\phi[k, n] = e^{j\Omega_k n}$$

donde Ω_k es la frecuencia angular discreta. A diferencia de las exponenciales continuas, no siempre una exponencial discreta de N muestras resulta periódica. Esto es debido a que, para una frecuencia de muestreo dada, no necesariamente habrá coincidencia con la frecuencia de la exponencial y el muestreo no retomará desde la misma parte de la exponencial antes de las N muestras. Si analizamos la condición de periodicidad para la exponencial discreta obtenemos:

$$e^{j\Omega_k(n+N)} = e^{j\Omega_k n} e^{j\Omega_k N} = e^{j\Omega_k n}$$

lo que implica que:

$$e^{j\Omega_k N} = 1$$

Para que esta última ecuación sea correcta, debe verificarse que $\Omega_k N$ sea un múltiplo entero de 2π , es decir $\Omega_k N = 2\pi k$, o bien:

$$\Omega_k = \frac{2\pi k}{N} \quad \text{con } k \in \mathbb{Z}$$

Para ejemplificar esto, analicemos lo que sucede con la parte real de la exponencial. A partir de la relación de Euler podemos escribir:

$$\text{real}(e^{j\Omega_k n}) = \cos(\Omega_k n)$$

Veamos qué pasa para valores específicos de Ω_k . En la Figura 3.4 se puede ver claramente que si $\Omega_k = 2\pi$ se obtiene una señal periódica, pero si se tiene $\Omega_k = 2,15\pi$ la señal no resulta periódica. Dado que la parte real de la exponencial de este ejemplo no resulta periódica, tampoco lo será la exponencial.

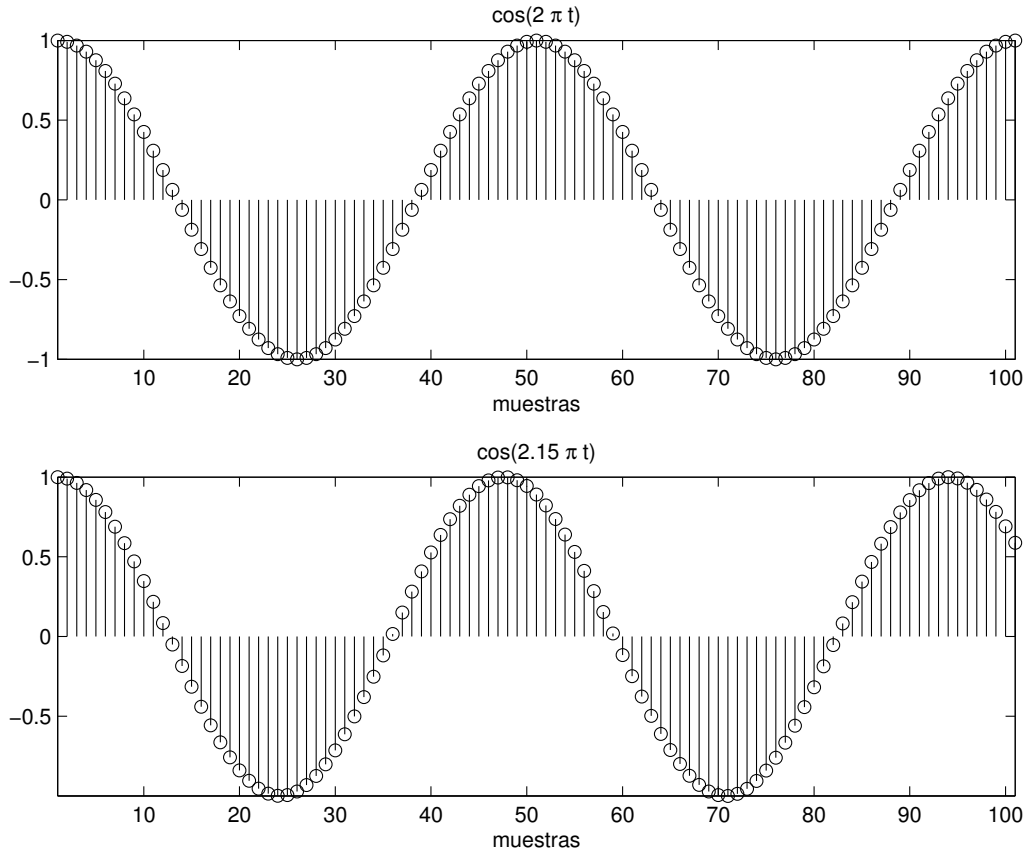


Figura 3.4. Funciones coseno para una base en \mathbb{R}^{100} con: (arriba) $\Omega_k = 2\pi$ (arriba), lo que resulta en una señal periódica y (abajo) $\Omega_k = 2,15\pi$, dando como resultado una señal no periódica.

Hemos obtenido así exponenciales complejas con período N , cuyas frecuencias sólo pueden tomar valores determinados por el entero k . La forma general es entonces:

$$\phi[k, n] = e^{j \frac{2\pi k}{N} n}$$

Pero ¿cuántos exponenciales diferentes pueden encontrarse al variar k ?

Si hacemos:

$$\begin{aligned}
 \phi[0, n] &= e^{j\frac{2\pi 0}{N}n} = 1 \\
 \phi[1, n] &= e^{j\frac{2\pi 1}{N}n} = e^{j\frac{2\pi}{N}n} \\
 &\vdots \\
 \phi[N-1, n] &= e^{j\frac{2\pi(N-1)}{N}n} \\
 \phi[N, n] &= e^{j\frac{2\pi N}{N}n} = e^{j2\pi n} = 1 \\
 \phi[N+1, n] &= e^{j\frac{2\pi(N+1)}{N}n} = e^{j\frac{2\pi N}{N}n} e^{j\frac{2\pi}{N}n} = e^{j\frac{2\pi}{N}n}
 \end{aligned}$$

podemos ver que:

$$\begin{aligned}
 \phi[N, n] &= \phi[0, n] \\
 \phi[N+1, n] &= \phi[1, n] \\
 &\vdots \\
 \phi[k, n] &= \phi[(k \bmod N), n]
 \end{aligned}$$

es decir, que sólo existen N exponenciales periódicas diferentes, por lo que tomamos $0 \leq k \leq N-1$.

Ahora bien, encontramos N exponenciales complejas en \mathbb{R}^N y, si pretendemos usarlas como base, sería conveniente que fueran ortogonales. Si hacemos el producto interno de dos de ellas, $\langle \phi[k_1], \phi[k_2] \rangle$, acotadas a N muestras, tenemos:

$$\begin{aligned}
 \langle \phi[k_1], \phi[k_2] \rangle &= \sum_{n=0}^{N-1} e^{j\frac{2\pi k_1}{N}n} e^{-j\frac{2\pi k_2}{N}n} \\
 &= \sum_{n=0}^{N-1} e^{j\frac{2\pi(k_1-k_2)}{N}n} \\
 &= \begin{cases} 0 & \forall k_1 \neq k_2 \\ N & \forall k_1 = k_2 \end{cases}
 \end{aligned}$$

Esta última ecuación resulta así porque, si $k_1 = k_2$, la exponencial vale 1 y se suma N veces este 1. Por otro lado, si $k_1 \neq k_2$ entonces $k_1 - k_2$ resulta en un número entero m , y haciendo un cambio de variables:

$$\langle \phi[k_1], \phi[k_2] \rangle = \sum_{n=0}^{N-1} e^{j\frac{2\pi m}{N}n} = 0$$

la cual es una sumatoria sobre una cantidad entera de períodos, de una exponencial compleja.

Resumiendo, hemos encontrado un conjunto de N señales en \mathbb{R}^N que son ortogonales y por tanto linealmente independientes. Estas propiedades hacen que también estas exponenciales complejas periódicas constituyan un conjunto generador para \mathbb{R}^N . Esta base para el espacio vectorial de las señales en \mathbb{R}^N puede expresarse de la siguiente forma:

$$F = \left\{ \phi / \phi[k, n] = e^{j\frac{2\pi kn}{N}}, 0 \leq k \leq N-1, 0 \leq n \leq N-1 \right\}$$

En la Figura 3.5 se han representado cuatro exponenciales complejas que forman parte de una base para señales en \mathbb{R}^{20} . En este ejemplo se han tomado las exponenciales correspondientes a $k = 1, k = 2, k = 3$ y $k = 4$.

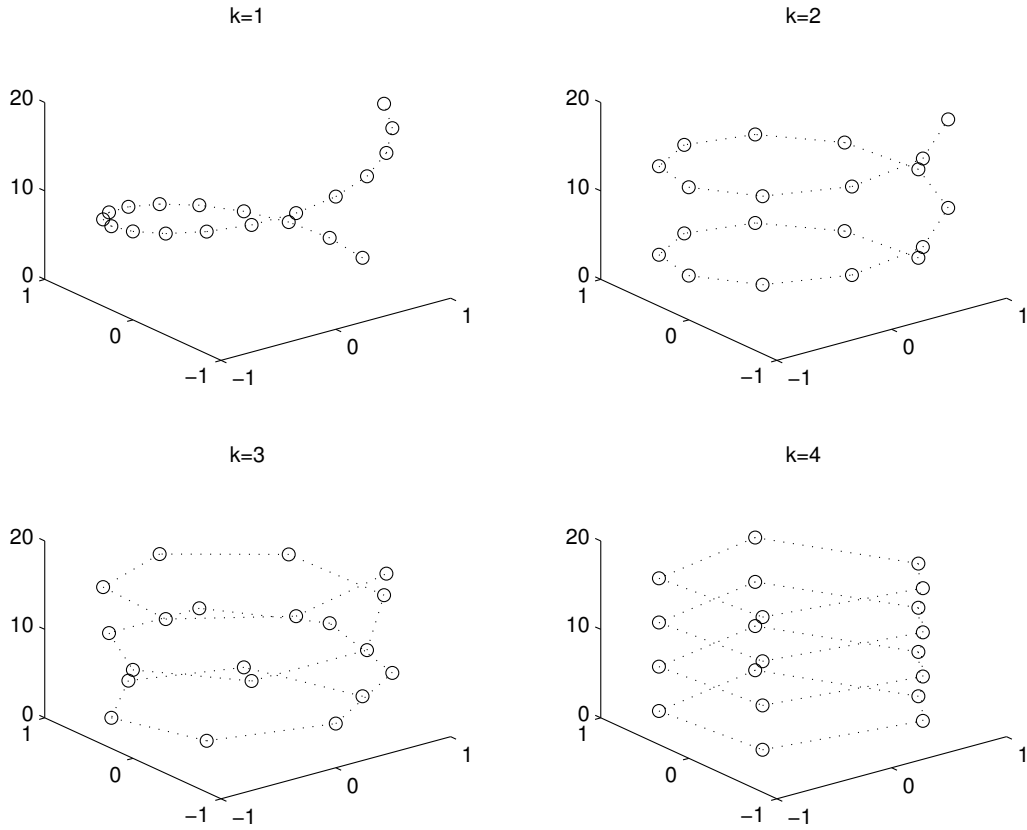


Figura 3.5. Cuatro elementos de la base para la Transformada Discreta de Fourier en \mathbb{R}^{20} . Los valores discretos se representan con círculos, y si bien se han unido con líneas punteadas, recuérdese que las muestras de las señales de la base son solamente los valores de los círculos marcados.

3.4. Transformada Discreta de Fourier

Habiendo determinado una base para nuestra transformación, nos queda ahora establecer cómo realizarla. Esto es sencillo utilizando la perspectiva de las transformaciones lineales. Una señal se puede representar en otra base ortogonal mediante:

$$\mathbf{x} = \sum_{k=0}^{N-1} X[k] \phi[k]$$

donde

$$X[k] = \frac{\langle \mathbf{x}, \phi[k] \rangle}{\langle \phi[k], \phi[k] \rangle} \quad (3.1)$$

Utilizando estas ecuaciones y la expresión para la base de exponenciales complejas discretas se llega a la definición de la Transformada Discreta de Fourier (TDF) de la siguiente manera:

$$\begin{aligned} X[k] &= \langle \mathbf{x}, \phi[k] \rangle \\ &= \sum_{n=0}^{N-1} x[n] \phi^*[k, n] \end{aligned}$$

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j \frac{2\pi kn}{N}}$$

(3.2)

Esta es la definición más utilizada para la TDF y en ella sólo se considera el numerador de (3.1), sin normalizar por el producto interno de las funciones de la base. Sin embargo, esto no es un problema dado que esos productos internos son todos iguales a N y este factor puede incluirse directamente en la fórmula de reconstrucción:

$$\begin{aligned} x[n] &= \sum_{k=0}^{N-1} \frac{X[k]}{\langle \phi[k], \phi[k] \rangle} \phi[k] \\ &= \sum_{k=0}^{N-1} \frac{X[k]}{N} \phi[k] \end{aligned}$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j \frac{2\pi k n}{N}}$$

(3.3)

Debe tenerse mucho cuidado en no confundir la TDF con la TFTD. La diferencia más importante radica en el hecho de que la TDF se aplica a señales discretas periódicas de duración infinita o a extensiones periódicas de señales de duración finita. Además, tanto el dominio como la imagen de la TDF son discretos mientras que en el caso de la TFTD la imagen es continua.

Es interesante analizar qué es lo que realmente está sucediendo al aplicar esta transformación. Ya que el producto interno conceptualmente mide el *grado de parecido* de una señal con respecto a la otra, lo que estamos haciendo entonces al implementar la TDF es comparar la señal de interés con las N exponenciales complejas que forman la base de \mathbb{R}^N . Intentamos determinar *qué tanto* de dicha exponencial debe usarse en una combinación lineal para sintetizar la señal original o *qué tanto* de dicha exponencial *hay* en la señal original. Dicho de otra forma, descomponemos la señal en una serie de N componentes con diferentes frecuencias, lo que nos permite estudiar características frecuenciales de la señal o el sistema que la generó.

Nótese que, si bien sólo hemos considerado N muestras de las exponenciales complejas, estos valores se repiten en las N muestras siguientes, lo que implica que al hacer una combinación lineal de ellas el resultante será una función de período N . Cuando trabajamos con señales discretas de longitud finita usualmente no es este el caso, sino que se trata de una secuencia de valores muestreados a partir de una señal continua. Fuera del intervalo de muestreo, dicha señal continua habrá tenido valores que en realidad desconocemos y estamos suponiendo que son iguales a los N elementos que conocemos. Es por esto que nunca hay que olvidar que cuando vemos el espectro obtenido por la TDF, en realidad vemos el espectro de la *extensión periódica* de las N muestras.

3.5. Propiedades de la TDF

La TDF presenta varias propiedades muy útiles e interesantes, similares a las propiedades de su análoga continua. Para analizar dichas propiedades

vamos a utilizar la siguiente notación:

$$x[n] \xleftrightarrow{\mathcal{F}} X[k]$$

donde \mathcal{F} representa el operador de transformación, para indicar el par formado por una señal $x[n]$ y su TDF $X[k]$. También expresamos:

$$X[k] = \mathcal{F} \{x[n]\}$$

para indicar que $X[k]$ es la TDF de $x[n]$, y

$$x[n] = \mathcal{F}^{-1} \{X[k]\}$$

para indicar que $x[n]$ es la TDFI de $X[k]$.

Para los siguientes pares: $x[n] \xleftrightarrow{\mathcal{F}} X[k]$ y $y[n] \xleftrightarrow{\mathcal{F}} Y[k]$ se verifican las siguientes propiedades (el símbolo \otimes representa el operador de convolución periódica (o circular) que se aplica cuando ambas señales son periódicas de período N):

1. *Linealidad:*

$$x[n] + y[n] \xleftrightarrow{\mathcal{F}} X[k] + Y[k]$$

2. *Simetría:*

$$\frac{1}{N} \mathcal{F} \{x[n]\} [k] \xleftrightarrow{\mathcal{F}} \mathcal{F}^{-1} \{X[k]\} [-n]$$

3. *Desplazamiento temporal (retardo):*

$$x[n - i] \xleftrightarrow{\mathcal{F}} X[k] e^{-j\frac{2\pi ki}{N}}$$

4. *Desplazamiento frecuencial (modulación):*

$$x[n] e^{j\frac{2\pi in}{N}} \xleftrightarrow{\mathcal{F}} X[k - i]$$

5. *Convolución en el tiempo:*

$$x[n] \otimes y[n] \xleftrightarrow{\mathcal{F}} X[k] Y[k]$$

6. *Convolución en frecuencia:*

$$x[n] y[n] \xleftrightarrow{\mathcal{F}} \frac{1}{N} X[k] \otimes Y[k]$$

7. *Teorema de correlación cruzada:*

$$\sum_{i=0}^N x[i]y[n+i] \xLeftrightarrow{\mathcal{F}} X[k]Y[k]^*$$

8. *Teorema de Parseval (conservación de la energía):*

$$\sum_{n=0}^{N-1} x[n]^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X[k]|^2$$

En la Tabla 3.1 se muestran algunas propiedades relacionadas con la simetría de los pares transformados en ambos dominios. Muchas veces es posible aprovechar esas simetrías para aumentar la eficiencia computacional en los algoritmos para el cálculo de las transformadas, así como para interpretar la información que se obtiene.

Tiempo	Frecuencia
$x[n]$ real	$X[-k] = X[k]^*$ (sim. conjugada)
$x[n]$ imaginaria	$X[-k] = -X[k]^*$
$x[n]$ par	$X[-k] = X[k]$ (sim. par)
$x[n]$ impar	$X[-k] = -X[k]$ (sim. impar)
$x[n]$ real y par	$X[k]$ real y par
$x[n]$ real e impar	$X[k]$ imag. e impar
$x[n]$ imag. y par	$X[k]$ imag. y par
$x[n]$ imag. e impar	$X[k]$ real e impar

Tabla 3.1. Correspondencias entre las simetrías en el dominio temporal y frecuencial.

3.6. Relación entre la TCF y la TDF

En esta sección desarrollaremos un procedimiento gráfico para interpretar la obtención de la TDF a partir de una señal muestreada, su relación con la TCF y los artefactos como el alias y el rizado que pueden presentar por este proceso.

Supongamos que tenemos una señal $x(t)$ continua de duración infinita y cuya TCF está limitada en banda, es decir, que por encima de cierta

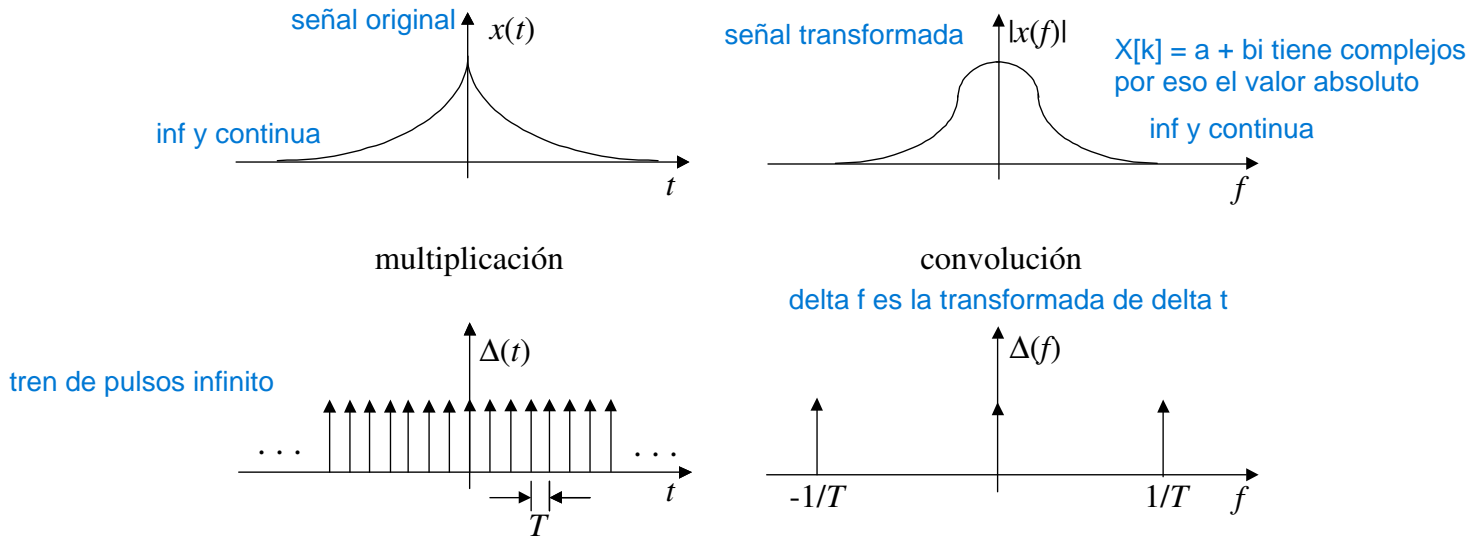


Figura 3.6. Muestreo de una señal continua

frecuencia f_{max} no contiene energía. Queremos muestrear dicha señal continua, para lo cual la multiplicamos por un tren de impulsos $\Delta(t)$ espaciados por un valor T . En el dominio frecuencial, el efecto de multiplicar en el tiempo es equivalente a una convolución entre la TCF de $x(t)$ y la del tren de impulsos. Pero la TCF de un tren de impulsos espaciados por T es otro tren de impulsos espaciados por $1/T$ (ver Figura 3.6).

Al convolucionar $x(f)$ con la TCF del tren de impulsos, $\Delta(f)$, el resultado es una función periódica con período $1/T$ y la forma de $x(f)$.

Obsérvese que si la frecuencia de muestreo $1/T$ no es superior al doble de la frecuencia máxima de la señal f_{max} , entonces habrá un solapamiento entre las versiones desplazadas de $x(f)$. En estas zonas se suman los aportes de ambas imágenes de $x(f)$. Esto se conoce como fenómenos de *alias* o *aliasing* y puede evitarse respetando el teorema de muestreo de Nyquist que puede resumirse en (Figura 3.7):

Si la frecuencia de muestreo es mayor al doble de la frecuencia máxima de la señal, entonces la señal continua puede reconstruirse exactamente a partir de su versión discreta utilizando el interpolador sincrónico.

Nótese que en el dominio temporal tenemos una señal continua muestreada y de duración infinita, mientras que en el dominio frecuencial se

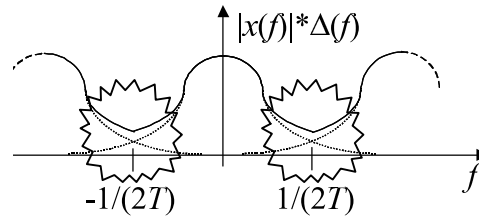


Figura 3.7. Fenómeno de alias

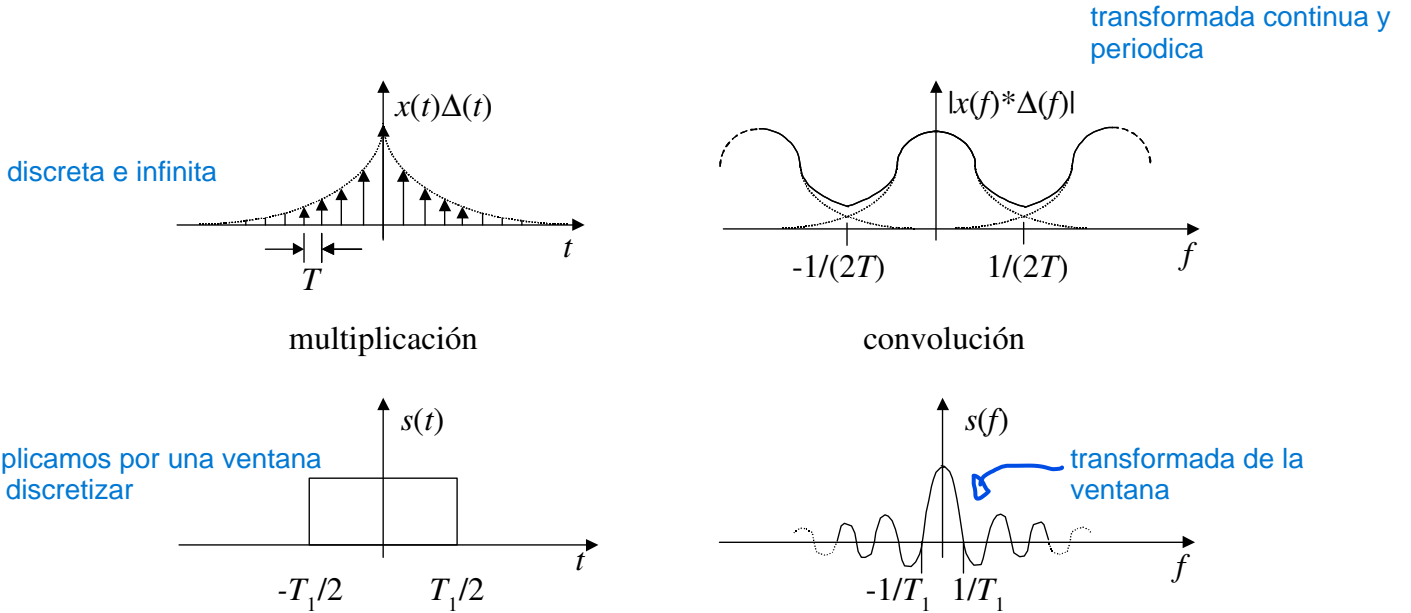


Figura 3.8. Multiplicación por una ventana cuadrada.

trata de una señal continua y periódica. Lo que tenemos en frecuencias es la TFTD que se mencionó en la sección anterior (ver Figura 3.8).

El siguiente paso es multiplicar $x(t)\Delta(t)$ por una ventana cuadrada $s(t)$ para acotarla en el tiempo. En el dominio frecuencial esto se corresponde a una convolución de la TCF de la ventana, $s(f)$, que es una función sinc. El efecto de dicha convolución será generar un rizado o *ripple* en el dominio frecuencial.

Ahora necesitamos muestrear en el dominio frecuencial, de manera de obtener un espectro discreto. Para ello multiplicamos por un tren de impulsos $\Delta_2(f)$ separados por $1/T_2$ lo que en el dominio temporal se traduce como una convolución con un tren de impulsos espaciados en T_2 . El resultado es una señal *periódica* de período T_1 (ver Figura 3.9). Nótese que se eligió el valor

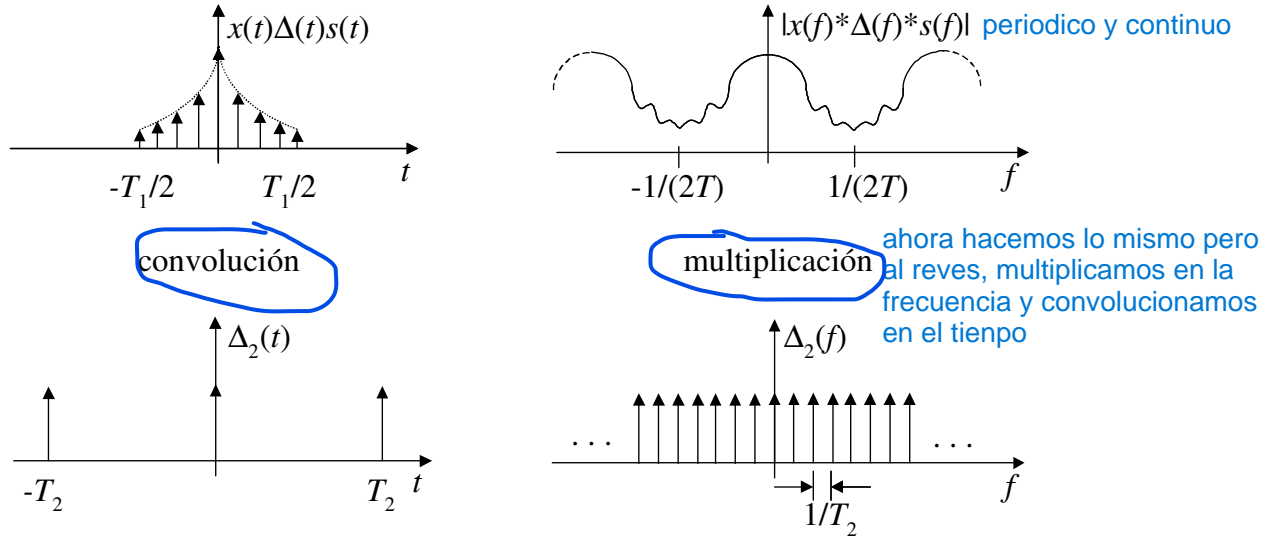


Figura 3.9. Muestreo en el dominio de la frecuencia

T_2 igual al ancho T_1 de la ventana temporal aplicada en el paso anterior, de manera que al hacer la convolución no se solapen las imágenes de la señal en el tiempo (alias temporal).

Finalmente, sólo nos resta tomar N muestras en el dominio temporal y N en el dominio frecuencial. Así, hemos obtenido la TDF de una señal discreta muestreada. Nótese que se asume que ambas señales son periódicas y se trabaja con un período completo de ambas (ver Figura 3.10).

3.7. Utilización de ventanas

En la sección anterior analizamos la TDF como una consecuencia del proceso de muestreo y limitación en la duración de la señal. Uno de los pasos críticos en dicho desarrollo se da al utilizar una ventana cuadrada en el dominio temporal para limitar la duración de la señal. Como se vio, esto en frecuencia es equivalente a la convolución con una función sinc, lo que genera la aparición de cierto “rizado” en el espectro. Para evitar este efecto no deseado, existe la alternativa de utilizar ventanas diferentes de la cuadrada, de manera que su espectro frecuencial presente lóbulos laterales de menor amplitud. Se han desarrollado diversas ventanas que cumplen con este requisito y analizaremos algunas de ellas (ver Figura 3.11). Las más utilizadas se definen para $0 < n \leq N$ según:

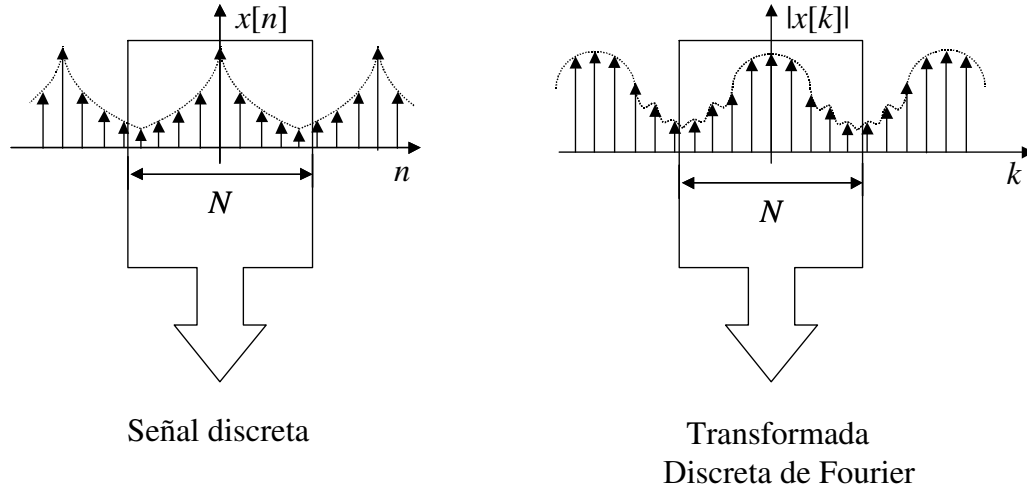


Figura 3.10. Par de la transformada discreta de Fourier.

i) Ventana rectangular:

$$\omega_R[n] = 1$$

ii) Ventana de Hanning:

$$\omega_h[n] = \frac{1}{2} - \frac{1}{2} \cos(2\pi n/N)$$

iii) Ventana de Hamming:

$$\omega_H[n] = \frac{27}{50} - \frac{23}{50} \cos(2\pi n/N)$$

iv) Ventana de Bartlett:

$$\omega_B[n] = \begin{cases} 2n/N & \text{si } 0 < n \leq N/2 \\ 2 - 2n/N & \text{si } N/2 < n \leq N \end{cases}$$

v) Ventana de Blackman:

$$\omega_K[n] = \frac{21}{50} - \frac{1}{2} \cos(2\pi n/N) + \frac{2}{25} \cos(4\pi n/N)$$

Estas ventanas pueden ser caracterizadas por el tamaño de los lóbulos de la magnitud de su espectro de frecuencias. La ventana rectangular posee el lóbulo central con menor ancho de banda pero la magnitud de los lóbulos laterales decae muy lentamente. La ventana de Blackman posee la mínima amplitud en sus lóbulos laterales pero su lóbulo principal tiene un ancho de banda tres veces mayor al de la rectangular.

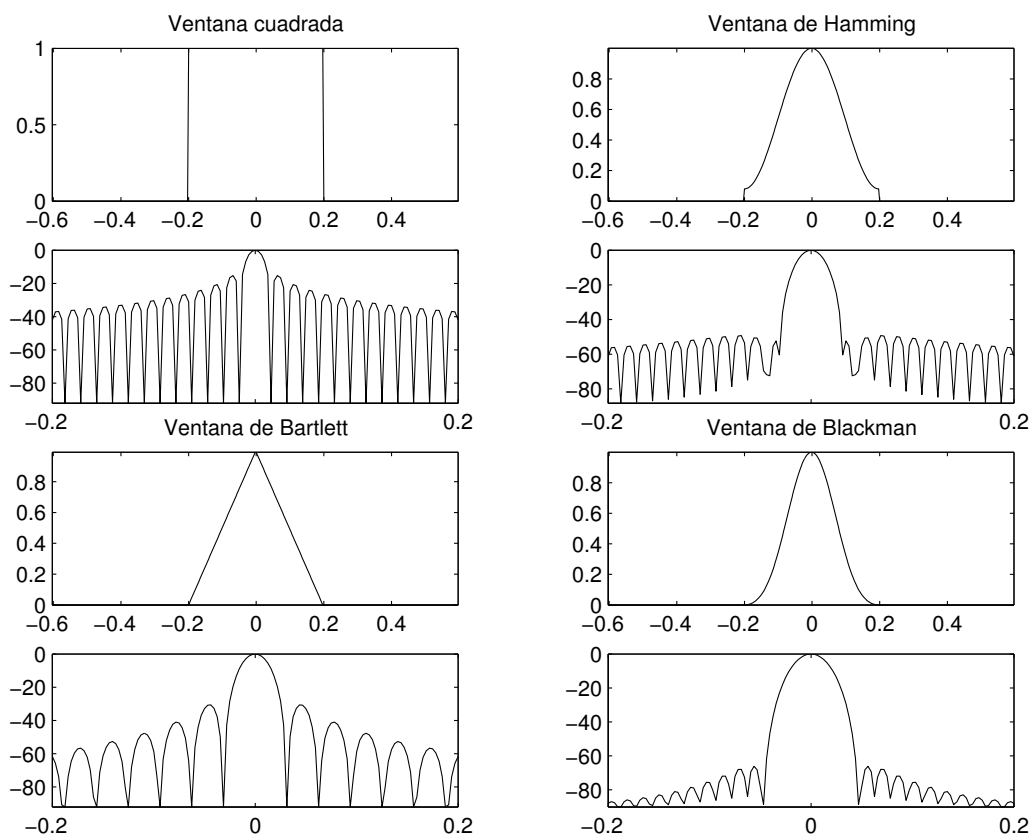


Figura 3.11. Las ventanas temporales más utilizadas y sus respectivos espectros en frecuencia.

3.8. Resolución temporal y frecuencial

Nos referimos a resolución cuando tratamos de especificar con qué precisión podemos determinar un valor. Resolución es sinónimo de separación, es decir, la máxima resolución es la separación que nos permitirá reconocer dos eventos o sucesos como diferentes, e identificar individualmente las informaciones que transportan. Esta definición es general y las unidades en las que medimos dicha separación dependerán de los fenómenos particulares que originan los sucesos. Por ejemplo, en óptica se denomina resolución a la menor separación entre dos puntos que, a cierta distancia, permite distinguirlos como puntos independientes. En el caso de señales que varían en el tiempo, se tratará del mínimo tiempo entre dos eventos de los que podemos conocer con exactitud sus características. Cuando se muestrea una señal con una frecuencia de muestreo f_m , automáticamente queda limitada

la resolución temporal, ya que no podemos dar información de lo que sucede entre dos muestras. Por lo tanto, la resolución temporal resulta ser:

$$\Delta t = T = \frac{1}{f_m}$$

Por otra parte, al muestrear además nos limitamos a un número finito N de muestras, que abarcarán un tiempo NT . Al usar la TDF, en ambos dominios la señal resulta discreta y periódica, con período N muestras. Además, del análisis anterior sabemos que el período en el dominio frecuencial será igual a la frecuencia de muestreo f_m . Por lo tanto, podemos establecer que la resolución frecuencial estará dada por el período en el dominio frecuencial dividido por N :

$$\Delta f = \frac{f_m}{N} = \frac{1}{NT} = \frac{1}{T_0}$$

donde T_0 es la duración total en tiempo de la señal muestreada finita (o lo que es lo mismo, el período de su extensión periódica).

Un concepto fundamental en el análisis es el principio de incertidumbre de Heisenberg, que podemos expresar en este contexto como sigue:

$$\Delta f \Delta t = \frac{1}{T_0} T = \frac{1}{NT} T$$

$$\Delta f \Delta t = \frac{1}{N}$$

Esta ecuación establece que, dado un número N de muestras a procesar (por ejemplo, nuestro algoritmo procesa bloques de 128 muestras), no podemos aumentar independientemente la resolución temporal y la frecuencial, ya que si aumentamos una la otra tiene que disminuir necesariamente para mantener la igualdad.

Podemos analizar lo que sucede al jugar con estos tres parámetros:

- Para una f_m fija:
 - Si $\uparrow N$ entonces $\uparrow T_0$ y $\downarrow \Delta f$ es decir, aumenta la resolución frecuencial
 - Si $\downarrow N$ entonces $\downarrow T_0$ y $\uparrow \Delta f$ es decir, disminuye la resolución frecuencial
- Para una N fija:

- Si $\uparrow f_m$ entonces $\downarrow T_0$ y $\uparrow \Delta f$ es decir, disminuye la resolución frecuencial
- Si $\downarrow f_m$ entonces $\uparrow T_0$ y $\downarrow \Delta f$ es decir, aumenta la resolución frecuencial

Muchas veces nos encontramos con la necesidad de mejorar la resolución frecuencial de la TDF. Es común pensar que basta aumentar f_m para lograr una mayor resolución frecuencial. Sin embargo, si se aumenta f_m , para un mismo T_0 aumentará proporcionalmente N y la resolución frecuencial Δf no cambiará. ¿Qué podemos hacer para disminuir Δf ?

- Disminuir la frecuencia de muestreo f_m para un N constante.
- Aumentar el número de puntos N si tenemos fijada f_m .

Estas dos opciones involucran remuestrear la señal, lo cual no siempre es posible. Se podría pensar en *aumentar* la duración de la señal, agregando ceros al final, como si realmente hubiéramos medido esos valores:

$$\mathbf{x}' = [x[0], x[1], \dots, x[N-1], 0, 0, \dots, 0]$$

De esta forma, para una misma frecuencia de muestreo se tienen más muestras y así se reduce Δf . Por ejemplo, si a una señal de N muestras la completamos con ceros hasta lograr $2N$ muestras y nos queda:

$$\Delta f' = \frac{1}{T'_0} = \frac{1}{2NT} = \frac{1}{2} \frac{1}{NT} = \frac{\Delta f}{2}$$

Hay que recalcar que esto aumenta la resolución por el aumento de N , pero no agrega más información de la señal en cuestión, es decir que en realidad implementa una interpolación en el dominio de la frecuencia. Si la señal original de la que fue tomada la secuencia de N muestras en verdad hubiera tomado valores iguales a ceros desde la muestra N a la $2N-1$, la TDF sería la correcta, y efectivamente se habría realizado un aumento de resolución. Pero recuérdese que al tratar con señales muestreadas a partir de una señal continua, realmente desconocemos los valores de las muestras posteriores a las N consideradas. Por lo tanto, el aumento de resolución es aparente dado que las nuevas muestras del espectro son aproximaciones a las que se obtendrían si en vez de cero se tuvieran los $2N$ valores de la señal temporal.

Ejemplo: sea

$$x[n] = \sin(2\pi 0,5n/f_m) + \sin(2\pi 1n/f_m) + \sin(2\pi 1,5n/f_m)$$

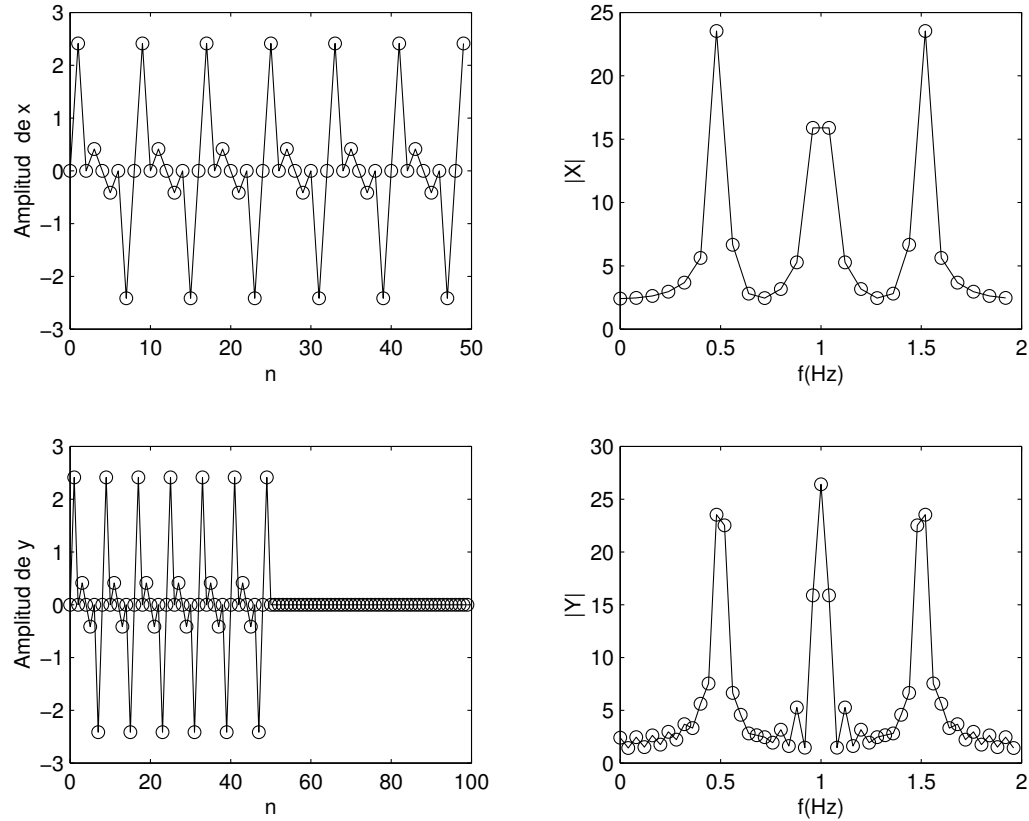


Figura 3.12. Efecto del agregado de ceros en el dominio temporal.

donde $N = 50$ y $f_m = 4$. También se define:

$$y[n] = \begin{cases} x[n] & \text{para } 0 \leq n \leq 50 \\ 0 & \text{para } 50 < n \leq 100 \end{cases}$$

En la Figura 3.12 se pueden observar los efectos de rellenar con ceros.

3.9. Representación matricial de la TDF

Como ya se mencionó, la TDF constituye una transformación lineal que genera un cambio de base sobre los vectores de \mathbb{R}^N . Sabemos que tal transformación siempre admite una representación matricial de la forma $\mathbf{y} = \mathbf{A}\mathbf{x}$. Estamos interesados, entonces, en determinar qué forma tiene la matriz de transformación para el caso de la TDF:

$$\mathbf{X} = \mathbf{W}\mathbf{x}$$

Sabemos que la transformación mapea señales en \mathbb{R}^N a señales en \mathbb{R}^N , por lo tanto la matriz W debe ser cuadrada. La ecuación (3.2) se puede reescribir como el producto interno entre un vector fila con las exponenciales y el vector columna \mathbf{x} :

$$X[k] = \left[e^{-\frac{j2\pi(k)(0)}{N}} \quad e^{-\frac{j2\pi(k)(1)}{N}} \quad \dots \quad e^{-\frac{j2\pi(k)(N-1)}{N}} \right] \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-1] \end{bmatrix}$$

Si disponemos los vectores de la exponencial como filas de una matriz, obtenemos la W deseada:

$$\begin{bmatrix} X[0] \\ X[1] \\ \vdots \\ X[N-1] \end{bmatrix} = \begin{bmatrix} e^{-\frac{j2\pi(0)(0)}{N}} & e^{-\frac{j2\pi(0)(1)}{N}} & \dots & e^{-\frac{j2\pi(0)(N-1)}{N}} \\ e^{-\frac{j2\pi(1)(0)}{N}} & e^{-\frac{j2\pi(1)(1)}{N}} & \dots & e^{-\frac{j2\pi(1)(N-1)}{N}} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-\frac{j2\pi(N-1)(0)}{N}} & e^{-\frac{j2\pi(N-1)(1)}{N}} & \dots & e^{-\frac{j2\pi(N-1)(N-1)}{N}} \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-1] \end{bmatrix}$$

Si utilizamos la notación $E = e^{-j2\pi/N}$ podemos escribir el producto de forma más compacta:

$$W = \begin{bmatrix} E^{(0)(0)} & E^{(0)(1)} & \dots & E^{(0)(N-1)} \\ E^{(1)(0)} & E^{(1)(1)} & \dots & E^{(1)(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ E^{(N-1)(0)} & E^{(N-1)(1)} & \dots & E^{(N-1)(N-1)} \end{bmatrix}$$

Ahora podemos hallar la matriz que invierte la transformación. A partir de (3.3) tenemos:

$$\begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-1] \end{bmatrix} = \frac{1}{N} \begin{bmatrix} E^{-(0)0} & E^{-(0)1} & \dots & E^{-(0)(N-1)} \\ E^{-(1)0} & E^{-(1)1} & \dots & E^{-(1)(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ E^{-(N-1)0} & E^{-(N-1)1} & \dots & E^{-(N-1)(N-1)} \end{bmatrix} \begin{bmatrix} X[0] \\ X[1] \\ \vdots \\ X[N-1] \end{bmatrix}$$

3.10. Transformada Rápida de Fourier

Supongamos que la señal $x[n]$ posee $N = 4$ muestras. Su TDF se puede escribir utilizando la notación matricial como:

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \end{bmatrix} = \begin{bmatrix} E^0 & E^0 & E^0 & E^0 \\ E^0 & E^1 & E^2 & E^3 \\ E^0 & E^2 & E^4 & E^6 \\ E^0 & E^3 & E^6 & E^9 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix}$$

Para implementar esta ecuación, se requieren N^2 multiplicaciones complejas y $N(N-1)$ sumas complejas. Pero teniendo en cuenta que $E^0 = 1$ se puede reescribir la ecuación como:

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & E^1 & E^2 & E^3 \\ 1 & E^2 & E^4 & E^6 \\ 1 & E^3 & E^6 & E^9 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix}$$

Como segunda simplificación, observemos que $E^{nk} = E^{(nk \bmod N)}$, donde $nk \bmod N$ es el resto de la división de nk por N . Por ejemplo, si $n = 2$ y $k = 3$, tenemos:

$$\begin{aligned} E^{nk} &= E^6 \\ &= e^{-j\frac{2\pi}{4}6} = e^{-j\pi 3} = e^{-j\pi} = e^{-j\frac{2\pi}{4}2} \\ &= E^2 = E^{6 \bmod 4} \\ &= E^{(nk) \bmod N} \end{aligned}$$

Utilizando este hecho, la ecuación queda:

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & E^1 & E^2 & E^3 \\ 1 & E^2 & E^0 & E^2 \\ 1 & E^3 & E^2 & E^1 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix}$$

El elemento E^0 no se ha reemplazado por 1 para generalizar el algoritmo en pasos posteriores.

Realizando un reordenamiento de las filas de las matrices y reescribiendo la matriz como un producto de dos matrices, tenemos:

$$\begin{bmatrix} X[0] \\ X[2] \\ X[1] \\ X[3] \end{bmatrix} = \begin{bmatrix} 1 & E^0 & 0 & 0 \\ 1 & E^2 & 0 & 0 \\ 0 & 0 & 1 & E^1 \\ 0 & 0 & 1 & E^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & E^0 & 0 \\ 0 & 1 & 0 & E^0 \\ 1 & 0 & E^2 & 0 \\ 0 & 1 & 0 & E^2 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix}$$

Esta factorización es la base del algoritmo de Transformada Rápida de Fourier (TRF, que en inglés se conoce como *Fast Fourier Transform*, FFT). Nótese que se han intercambiado las filas 1 y 2 de $X[k]$. Podemos separar esta operación en dos etapas, realizando primero el producto de \mathbf{x} y la matriz de la derecha, obteniéndose un resultado intermedio, \mathbf{x}^1 :

$$\begin{bmatrix} x_1[0] \\ x_1[1] \\ x_1[2] \\ x_1[3] \end{bmatrix} = \begin{bmatrix} 1 & 0 & E^0 & 0 \\ 0 & 1 & 0 & E^0 \\ 1 & 0 & E^2 & 0 \\ 0 & 1 & 0 & E^2 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix}$$

Desarrollando esta ecuación, tenemos:

$$\begin{aligned} x_1[0] &= x[0] + E^0 x[2] \\ x_1[1] &= x[1] + E^0 x[3] \\ x_1[2] &= x[0] + E^2 x[2] \\ x_1[3] &= x[1] + E^2 x[3] \end{aligned}$$

Ahora bien, como una simplificación más, $E^2 = -E^0$, lo que es fácil de verificar reemplazando por la exponencial y calculando. Por lo tanto, las ecuaciones quedan:

$$\begin{aligned} x_1[0] &= x[0] + E^0 x[2] \\ x_1[1] &= x[1] + E^0 x[3] \\ x_1[2] &= x[0] - E^0 x[2] \\ x_1[3] &= x[1] - E^0 x[3] \end{aligned}$$

Aquí podemos ver que los segundos términos en la primera y tercera ecuación son iguales (excepto por el signo negativo) y lo mismo para los segundos términos en la segunda y cuarta ecuación. Por lo tanto, para el cálculo del resultado intermedio \mathbf{x}^1 sólo se necesitan cuatro sumas y dos multiplicaciones entre números complejos.

Analicemos ahora el segundo producto matricial:

$$\begin{bmatrix} x_2[0] \\ x_2[1] \\ x_2[2] \\ x_2[3] \end{bmatrix} = \begin{bmatrix} 1 & E^0 & 0 & 0 \\ 1 & E^2 & 0 & 0 \\ 0 & 0 & 1 & E^1 \\ 0 & 0 & 1 & E^3 \end{bmatrix} \begin{bmatrix} x_1[0] \\ x_1[1] \\ x_1[2] \\ x_1[3] \end{bmatrix}$$

que desarrollado queda:

$$\begin{aligned}x_2[0] &= x_1[0] + E^0 x_1[1] \\x_2[1] &= x_1[0] + E^2 x_1[1] \\x_2[2] &= x_1[2] + E^1 x_1[3] \\x_2[3] &= x_1[2] + E^3 x_1[3]\end{aligned}$$

Con el mismo razonamiento de antes, $E^2 = -E^0$ y $E^3 = -E^1$ y por lo tanto para el cálculo de esta ecuación también se han usado cuatro sumas y dos multiplicaciones entre números complejos. En total se requirieron 8 sumas y 4 productos complejos ya que:

$$\begin{bmatrix} X[0] \\ X[2] \\ X[1] \\ X[3] \end{bmatrix} = \begin{bmatrix} x_2[0] \\ x_2[1] \\ x_2[2] \\ x_2[3] \end{bmatrix}$$

Para obtener el $X[k]$ sólo tenemos que intercambiar la segunda y la tercera filas.

En este ejemplo con $N = 4$ y $r = 2$ ha quedado claro que se ha reducido la cantidad de operaciones, de 16 multiplicaciones complejas y 12 sumas complejas, a sólo 4 productos complejos y 8 sumas complejas. El algoritmo que se ha ejemplificado es conocido como TRF *de base 2*, es decir con $N = 2^r$, que consiste en factorizar la matriz de transformación en r matrices más sencillas con la propiedad de minimizar el número de productos y sumas necesarios. Para este algoritmo, se consigue una reducción importante del número de operaciones: en el caso general se requieren $rN/2$ multiplicaciones complejas y rN sumas complejas. El costo computacional es prácticamente proporcional a la cantidad de multiplicaciones. En este caso se puede estimar un factor reducción relativa del costo computacional como:

$$\frac{N^2}{rN/2} = \frac{2N}{r}$$

Por ejemplo si $N = 1024 = 2^{10}$, el tiempo de cálculo será $2 \times 1024/10 = 204,8$ veces menor si se usa el algoritmo de la TRF en lugar del cálculo directo de la TDF a partir de su definición. Esta relación se ilustra en la Figura 3.13.

Existen otras variaciones del algoritmo en las que en vez de usar $N = 2^r$ se toma $N = b^r$, dando origen a los denominados *algoritmos de base 4, 8, 16*,

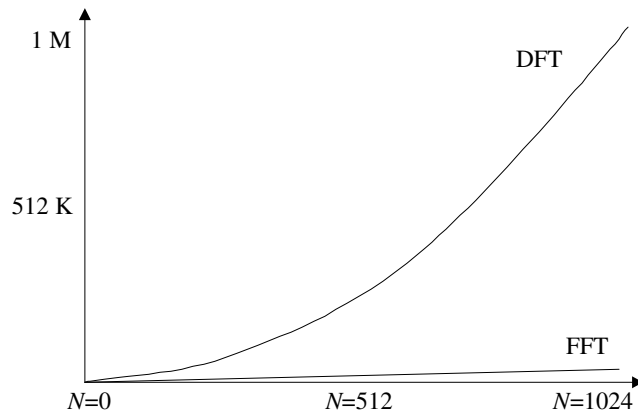


Figura 3.13. Comparación del costo computacional de la TDF y el algoritmo de la TRF.

etc. La Tabla 3.2 muestra la cantidad de operaciones requeridas por varios de estos algoritmos para $N = 4096$. Las bases diferentes de las de 2 no son muy utilizadas en la práctica ya que limitan cada vez más la cantidad de muestras de la señal a transformar.

Algoritmo	Nº de productos	Nº de sumas
TDF directa	16777216	16773120
TRF base 2	81924	139266
TRF base 4	57348	126978
TRF base 8	49156	126978
TRF base 16	48132	125442

Tabla 3.2. Operaciones requeridas para el cálculo de la TDF directamente a partir de su definición y utilizando el algoritmo de la TRF con diferentes bases.

3.11. Preguntas

1. ¿Qué unidades tiene la Transformada de Fourier?
2. ¿Es siempre útil realizar el análisis de una señal mediante la Transformada de Fourier?
3. ¿Se puede utilizar una base que no sea ortogonal para realizar el análisis de una señal? ¿Cuál es la ventaja de la ortogonalidad?

4. ¿Existe la Transformada de Fourier de una señal senoidal?
5. ¿Por qué no se puede analizar cualquier señal con una base constituida sólo por funciones sinusoidales?
6. ¿Qué relación tiene la propiedad de desplazamiento frecuencial de la TF con la modulación de señales?
7. ¿Qué inconvenientes aparecen cuando pasamos de las señales analógicas a las digitales?
8. ¿Cómo se manifiesta el aliasing en una señal del tiempo desde el punto de vista temporal y frecuencial?, ¿y en una imagen?
9. ¿Cómo se pueden disminuir los efectos del aliasing?
10. ¿Cuál es el resultado más evidente de muestrear una señal (desde el punto de vista frecuencial)?
11. ¿Por qué se dice que la función sincrónica constituye el interpolador ideal?
12. ¿Cuál es la diferencia entre la Transformada de Fourier de una señal muestreada y la TDF?
13. ¿Qué distorsiones incorpora el uso de ventanas rectangulares en el espectro de la señales resultantes?, ¿cómo puede minimizarse este efecto?
14. ¿Qué ventajas puede tener plantear a la TDF como un producto de vectores y matrices?. ¿Qué significa que una transformación sea unitaria y cómo se logra?
15. ¿Qué diferencia existe entre la TDF y la TRF?
16. ¿En qué ideas se basa la TRF para lograr su objetivo?
17. ¿Qué es la resolución frecuencial y como puedo aumentarla?
18. ¿Qué significa relleno de ceros (zero-padding)?
19. Cuando se utiliza la TRF, ¿dónde se *guardan* las *frecuencias negativas*?
20. ¿Qué es la autocorrelación y la correlación cruzada? ¿Para qué sirven?

21. ¿Cómo se puede utilizar la TRF para acelerar el cálculo de la convolución y de la correlación?
22. ¿Cómo se estima el espectro de una señal aleatoria?
23. ¿Qué es el ruido blanco?
24. ¿Qué diferencias existen entre el espectro de un delta Dirac y un ruido blanco?
25. ¿Cómo construiría un analizador de respuesta en frecuencia mediante un osciloscopio y oscilador controlado por tensión?
26. ¿Y con un banco de filtros?
27. ¿Por qué decimos que la cóclea realiza un *análisis espectral*?
28. ¿Qué ventajas posee realizar un filtrado directamente en el dominio frecuencial?
29. Suponga que tenemos una señal de banda angosta pero de frecuencia central desconocida y que la energía del ruido se distribuye uniformemente en todas las frecuencias. ¿Cómo implementaría un filtro para *limpiar* esta señal?
30. Suponga que tiene que implementar un sistema digital que traduzca la señal analógica generada por el discado telefónico. ¿Cómo lo haría usando la TDR?
31. ¿De qué manera podría implementar un sistema de encriptación digital mediante la TDF?
32. ¿Por qué la Transformada de Fourier no es adecuada para el análisis de señales no estacionarias? ¿Cómo se puede adecuar esta herramienta a este caso particular?
33. Describa cómo sería la base de la TDF bidimensional empleada para el análisis de las imágenes.

3.12. Trabajos prácticos

Ejercicio 1: Genere una señal $s(t) = \sin(2\pi f_1 t) + 4\sin(2\pi f_2 t)$, con $f_1 = 10$ y $f_2 = 20$ Hz, y obtenga su versión discreta $s[n]$ con período de muestreo $T_m = 0,001$ seg. en el intervalo de tiempo $t = [0 \dots 1]$ seg. A continuación:

1. Calcule la TDF $S[k]$ de la señal $s[n]$ y grafique el espectro de magnitud de $S[k]$.
2. Verifique la relación de Parseval para la TDF:

$$E_s = \sum_{n=1}^N s[n]^2 = \frac{1}{N} \sum_{k=1}^N |S[k]|^2$$

donde N es la cantidad de muestras de $s[n]$.

Realice los siguiente cambios y analice los resultados obtenidos:

1. Modifique $s[n]$ de forma tal que:

$$s(t) = \sin(2\pi f_1 t) + 4\sin(2\pi f_2 t) + 4$$

y observe los cambios en el espectro de magnitud de $S[k]$.

2. Modifique las frecuencias de las señales seno de forma tal que $f_1 = 10$ Hz y $f_2 = 11$ Hz y observe los cambios en el espectro de magnitud de $S[k]$.
3. Modifique nuevamente las frecuencias de las señales seno de forma tal que $f_1 = 10$ Hz y $f_2 = 10,5$ Hz. ¿Qué ocurre en el espectro de magnitud de $S[k]$?
4. Modifique el intervalo de tiempo de análisis de la siguiente manera $t = [0 \dots 0,72]$ seg. y observe los cambios en la TDF.

Ejercicio 2: Utilizando el programa didáctico *SigTeach* realice los siguientes ejercicios:

- Genere las siguientes señales:
 1. Seno real de período T .
 2. Cuadrada real de período T .
 3. Armónica compleja de período T .

4. Seno de período 3 o 4 veces menor que T .
 5. Armónica compleja de período 3 o 4 veces menor que T .
 6. Suma de la primera (1) y la cuarta (4).
- Determine si son ortogonales los pares de señales 1 y 4, 1 y 2, 3 y 5, 1 y 6.
 - Obtenga las TDF de todas las señales y repita el punto anterior para las transformadas de las señales 1 y 4, 3 y 5, 1 y 6. Discuta los resultados obtenidos.
 - Utilice las animaciones didácticas de la TDF para varias señales sencillas con 128 muestras.

Ejercicio 3: En los archivos de datos `ecg.txt`, `eeg.txt`, `emg.txt`, `presion.txt` y `respiracion.txt` se encuentran almacenados registros homónimos. Cargue los archivos mencionados y calcule las respectivas TDF. Luego grafique en una ventana los registros y en otra los espectros de magnitud de cada TDF ordenados de forma tal que el ancho de banda vaya aumentando.

Ejercicio 4: (*) En la Figura 3.14 se observa el espectro de magnitud obtenido aplicando la TDF a una señal generada mediante la función $g(t) = \sin(2\pi f_1 t) + 4\sin(2\pi f_2 t)$ y luego digitalizada.

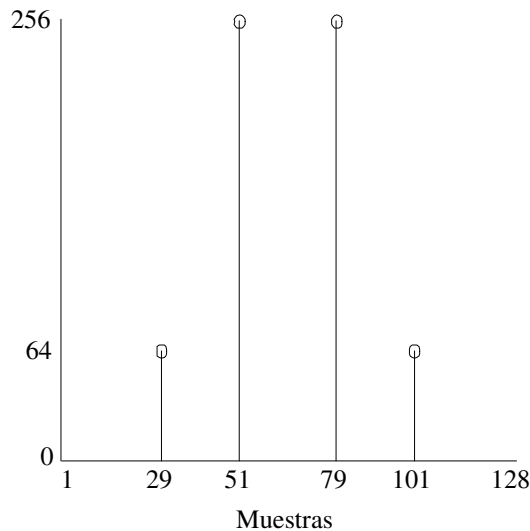


Figura 3.14. Espectro de magnitud obtenido mediante la TDF.

Siendo T la duración total de la señal adquirida y f_m la frecuencia de muestreo, indique cuáles de los siguientes conjuntos de parámetros puede ser correcto:

1. $T = 249$ ms, $f_m = 512$, $f_1 = 112$, $f_2 = 200$ Hz.
2. $T = 498$ ms, $f_m = 128$, $f_1 = 56$, $f_2 = 200$ Hz.
3. $T = 993$ ms, $f_m = 128$, $f_1 = 100$, $f_2 = 50$ Hz.
4. $T = 498$ ms, $f_m = 256$, $f_1 = 56$, $f_2 = 868$ Hz.
5. $T = 993$ ms, $f_m = 128$, $f_1 = 100$, $f_2 = 78$ Hz.
6. $T = 124,5$ ms, $f_m = 1024$, $f_1 = 3872$, $f_2 = 5520$ Hz.

Ejercicio 5: (*) Un método simple de interpolación consiste en agregar ceros en la TDF de la señal en cuestión y luego utilizar la TDF inversa. Investigue los detalles de implementación de este método para el caso de la señal `merval1.txt`, correspondientes al Índice Merval del último día de cada mes. El primer valor corresponde al índice del 30 de enero de 2001, el segundo al 28 de febrero de 2001 y así sucesivamente. Suponiendo que los valores se encuentran separados uniformemente a un mes, obtenga una estimación para los valores a mediados de cada mes. En el archivo `merval2.txt` se encuentran los valores tomados tanto a fines como a mediados de mes para poder comparar y obtener conclusiones.

Ejercicio 6: (*) La señal que se encuentra en el archivo `necg.txt` corresponde al registro de la actividad eléctrica del corazón de un paciente. Esta señal se ha digitalizado a razón de 360 muestras por segundo. Se sabe que el registro ha sido contaminado con un ruido en la banda de 40 a 180 Hz y se necesita eliminarlo para poder realizar un diagnóstico adecuado. Utilice la TDF para filtrar la señal (*).

Ejercicio 7: (*) Se desea detectar en qué intervalos de tiempo el instrumento grabado en el archivo `nshima.txt` ha ejecutado la nota LA. La frecuencia de muestreo con que se tomó el registro es de 11025 Hz.

Bibliografía

- [1] T. Aljama, M. Cadena, S. Charleston, and O. Yañez. *Procesamiento Digital de Señales*. Ed. UAM, México.
- [2] E. Brigham. *The Fast Fourier Transform and its applications*. Prentice-Hall, 1988.
- [3] J. Deller, J. Proakis, and J. Hansen. *Discrete-time processing of speech signals*. Prentice-Hall, 1993.
- [4] L. E. Franks. *Teoría de la Señal*. Reverté S.A., 1975.
- [5] W. Kaplan. *Matemáticas Avanzadas*. Addison-Wesley Iberoamericana, 1986.
- [6] H. Kwakernaak. *Modern Signal and Systems*. Prentice-Hall, 1991.
- [7] S. Mallat. *A wavelet tour of signal processing*. Number 2nd Ed. Academic Press, 1999.
- [8] A. V. Oppenheim and R. W. Schaffer. *Digital Signal Processing*. Prentice-Hall, 1975.
- [9] A. Papoulis and M. Bertrán. *Sistemas y Circuitos Digitales y Analógicos*. Marcombo, 1989.
- [10] L. Rabiner and R. Schaffer. *Digital processing of speech signals*. Prentice-Hall, 1978.
- [11] R. Roberts and R. Gabel. *Señales y sistemas lineales*. Limusa, 1994.
- [12] N. K. Sinha. *Linear Systems*. Wiley, 1991.
- [13] H. Skilling. *Circuitos en Ingeniería Eléctrica*. Cia. Ed. Continental, México, 1987.
- [14] G. Strang and T. Nguyen. *Wavelets and Filter Banks*. Wellesley-Cambridge Press, 1996.

Capítulo 4

Introducción a sistemas

Rubén Acevedo

Temas a tratar

- Definición de sistema.
- Propiedades de sistemas.
- Sistemas lineales e invariantes en el tiempo (LTI).
- Ecuaciones en diferencias.
- Diagramas de bloques.

Objetivos

- Comprender el concepto de sistema.
- Interpretar correctamente las propiedades de un sistema.
- Comprender la importancia de los sistemas LTI.
- Manejar el concepto de ecuaciones en diferencias.

4.1. Introducción

Un sistema se puede ver como cualquier proceso que produce una transformación de señales, por lo tanto tiene una señal de entrada (*entrada* de ahora en adelante) y una señal de salida (*salida* de ahora en adelante) que están relacionadas entre sí a través de la función de transferencia del sistema.

Los sistemas se pueden dividir en sistemas de tiempo continuo y sistemas de tiempo discreto. Un *sistema de tiempo continuo* es aquél en que las entradas de tiempo continuo son transformadas en salidas de tiempo continuo. La relación de entrada y salida se representa con la siguiente notación: $x(t) \rightarrow y(t)$, donde $x(t)$ es la entrada e $y(t)$ la salida.

Un *sistema de tiempo discreto* es aquél que transforma las entradas de tiempo discreto en salidas de tiempo discreto, la representación de éstos es de la siguiente forma: $x[n] \rightarrow y[n]$.

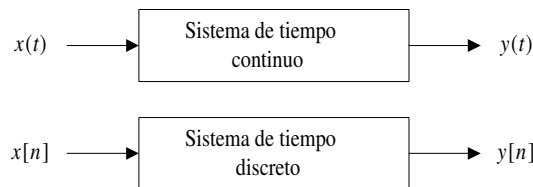


Figura 4.1. Representación de sistemas de tiempo continuo y discreto.

4.2. Interconexión de sistemas

Una idea muy importante es la de interconectar sistemas, ya sea en *serie* (también llamada en cascada) o en *paralelo*. En la Figura 4.2 se representan los dos tipos de interconexiones, a este tipo de diagramas se los denomina *diagramas de bloques*.

Las interconexiones se utilizan para construir sistemas a partir de otros ya existentes; por ejemplo se pueden diseñar sistemas para resolver expresiones matemáticas complicadas como el caso de $y[n] = (2x[n] - x[n]^2)^2$. El diagrama mostrado en la Figura 4.3 corresponde a dicho sistema, aquí los signos + y - indican que la señal $x[n]^2$ será restada de la señal $2x[n]$; si estos signos no están presentes se asumirá que las correspondientes señales serán sumadas.

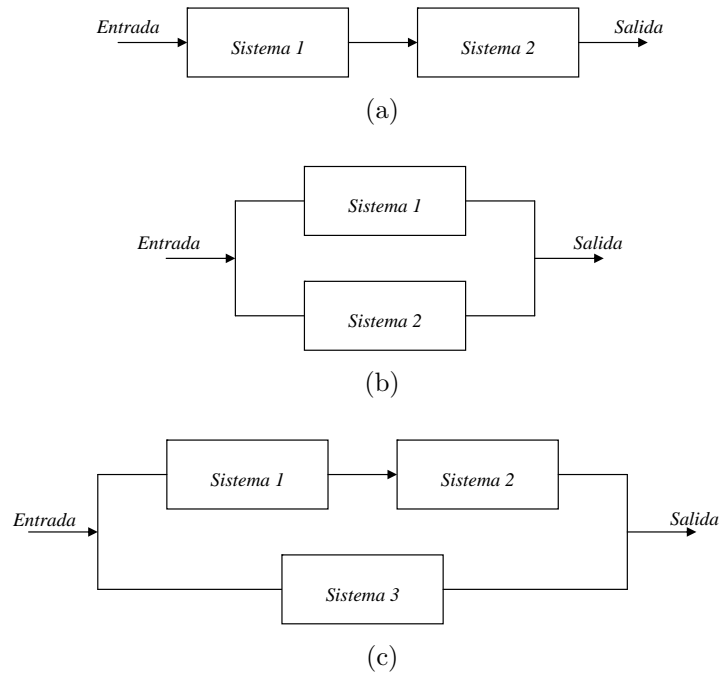


Figura 4.2. Interconexión de sistemas: (a) en serie, (b) en paralelo, (c) en serie/paralelo.

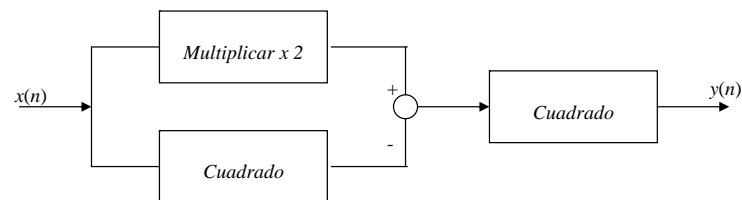


Figura 4.3. Sistema para el cálculo de $y[n] = (2x[n] - x[n]^2)^2$

Otro tipo importante de interconexión de sistemas es la interconexión de realimentación que se muestra en la Figura 4.4. La salida del sistema 1 es la entrada al sistema 2, mientras que la salida de éste se realimenta y se suma a la entrada externa del sistema 1 para producir la señal de entrada al mismo.

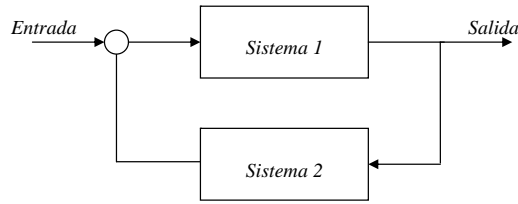


Figura 4.4. Interconexión de realimentación.

4.3. Propiedades de los sistemas

Los sistemas presentan propiedades que pueden ser tomadas como criterio para realizar una clasificación. Estas son válidas tanto para sistemas de tiempo continuo como de tiempo discreto, sin embargo se enfatizará el caso de estos últimos.

Sistemas con y sin memoria: un sistema es *sin memoria* si su salida en cada instante de tiempo depende sólo de la entrada en ese mismo instante. En el caso que la salida dependa de valores anteriores de la entrada el sistema es *con memoria*. Un ejemplo de sistema sin memoria es el definido por la siguiente ecuación $y(t) = Rx(t)$ (o su equivalente discreto $y[n] = Rx[n]$), donde $y(t)$ representa la diferencia de potencial en los extremos de una resistencia R y $x(t)$ la corriente a través de esta. El sistema $y[n] = \sum_{k=0}^3 x[n-k] + y[n-2]$ en cambio es un ejemplo de sistema con memoria.

Sistemas invertibles: un sistema es *invertible* si distintas entradas producen distintas salidas, es decir, dada la salida del sistema es posible determinar la entrada.

Ejemplos:

$$y[n] = 2x[n] \text{ sistema invertible}$$

$$y[n] = 2x[n]^2 \text{ sistema no invertible}$$

Sistemas causales: un sistema es *causal* si su salida en cualquier instante de tiempo depende sólo de los valores de la entrada en el instante actual y en instantes anteriores. A menudo este tipo de sistemas son llamados *no anticipativos*, ya que para calcular la salida no se necesita conocer con anticipación valores futuros de la entrada.

El sistema $y[n] = \sum_{k=0}^3 x[n-k]$ es un ejemplo de sistema causal, a diferencia del sistema $y[n] = x[n] - x[n+1]$.

Sistemas estables: un sistema *estable* es aquel en el que entradas pequeñas o perturbaciones conducen a respuestas que no divergen.

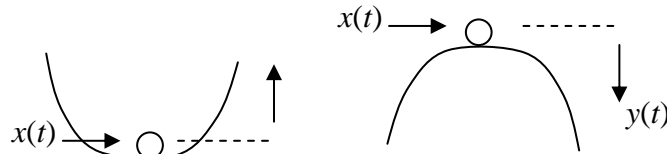


Figura 4.5. Ejemplos de sistema estable e inestable.

La definición precedente es intuitiva, estrictamente un sistema es *estable* si para una entrada limitada (su magnitud está acotada) su salida también lo es, y por lo tanto no puede divergir.

Los sistemas de la Figura 4.5 representan ambos tipos de sistemas, donde $x(t)$ es una fuerza horizontal aplicada a la pelota e $y(t)$ es la posición vertical de la misma.

Sistemas invariantes en el tiempo: un sistema es *invariante en el tiempo* si un desplazamiento en la entrada produce el mismo desplazamiento en la salida. Esto es equivalente a decir que si los coeficientes de la ecuación que define la dinámica del sistema (parámetros del sistema) son constantes. En caso contrario el sistema es variante en el tiempo.

Ejemplos:

$y[n] = 5x[n] + 2 \cdot y[n-1]$ sistema invariante en el tiempo.

$y[n] = \sin(\omega n)x[n-1]$ sistema variante en el tiempo.

Sistemas lineales: un sistema es *lineal* si posee la propiedad de superposición. Es decir, si una entrada consiste en la suma ponderada de varias señales entonces la salida del sistema es la superposición (la suma ponderada) de las respuestas del sistema para cada una de estas señales.

Matemáticamente, un sistema lineal se define a través de las siguientes propiedades: sean $y_1(t)$ y $y_2(t)$ las salidas del sistema a las entradas $x_1(t)$ y $x_2(t)$ respectivamente, entonces:

- la respuesta a $x_1(t) + x_2(t)$ es $y_1(t) + y_2(t)$ (aditividad)
- la respuesta a $ax_1(t)$ es $ay_1(t)$, donde a es un escalar cualquiera (escalamiento u homogeneidad)

Estas propiedades, que definen un sistema lineal, se pueden combinar en un solo enunciado, el cual se describe a continuación:

$$ax_1(t) + bx_2(t) \rightarrow ay_1(t) + by_2(t)$$

donde a y b son escalares complejos cualquiera.

Generalizando se puede demostrar que si:

$$x[n] = \sum_k a_k x_k[n] = a_1 x_1[n] + a_2 x_2[n] + \dots$$

es la entrada del sistema entonces:

$$y[n] = \sum_k a_k y_k[n] = a_1 y_1[n] + a_2 y_2[n] + \dots$$

es la salida del sistema. Este hecho se conoce como el *principio de superposición*, el cual se cumple en sistemas lineales de tiempo continuo y discreto.

4.4. Ecuaciones en diferencias

La dinámica de los sistemas de tiempo continuo se representa mediante una ecuación diferencial, en el caso que ésta sea *lineal y de coeficientes constantes* el sistema se denomina *lineal e invariante en el tiempo* (LTI). Estos sistemas representan una amplia variedad de sistemas y fenómenos físicos; ejemplo de éstos son los circuitos pasivos RLC, sistemas mecánicos masa-resorte y cinética de reacciones químicas, entre otros.

La representación de los sistemas LTI en tiempo discreto se realiza a través de *ecuaciones en diferencias lineales de coeficientes constantes*; estas ecuaciones describen procesos secuenciales como por ejemplo la obtención de la respuesta que produce el tracto vocal (señal de voz) a la excitación de las cuerdas vocales.

Una ecuación diferencial de orden N y coeficientes constantes está dada por:

$$\sum_{k=0}^N \alpha_k \frac{d^k y(t)}{dt^k} = \sum_{k=0}^M \beta_k \frac{d^k x(t)}{dt^k}$$

la contraparte discreta es la ecuación en diferencias lineal de coeficientes constantes:

$$\sum_{k=0}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k].$$

Esta última ecuación puede acomodarse de la siguiente forma:

$$y[n] = \frac{1}{a_0} \left[\sum_{k=0}^M b_k x[n-k] - \sum_{k=1}^N a_k y[n-k] \right]$$

y expresar la salida en el instante n en función de los valores previos de la entrada y la salida. Esta ecuación es llamada *ecuación recursiva*, ya que implica un procedimiento de esta naturaleza para determinar la salida. La *ecuación no recursiva* está dada por la expresión:

$$y[n] = \sum_{k=0}^M \frac{b_k}{a_0} x[n-k]$$

donde la salida es función de los valores previos de la entrada. Los sistemas representados por esta última ecuación se denominan de *respuesta finita al impulso* (FIR, del inglés *finite impulse response*) debido a que esta respuesta tiene una duración finita. En el caso de los sistemas representados por ecuaciones recursivas se denominan de *respuesta infinita al impulso* (IIR, del inglés *infinite impulse response*).

Los sistemas FIR son llamados de *promedios móviles o moving average* (MA), ya que realizan un promedio de la entrada en sucesivos instantes de tiempo. Los sistemas IIR se dividen en 2 tipos:

- los sistemas *autoregresivos* (AR) son aquellos en que la salida en el instante n depende sólo de los valores anteriores de ésta y del valor actual de la entrada, y la ecuación que rige sus dinámicas tiene la siguiente forma:

$$y[n] = - \sum_{k=1}^N \frac{a_k}{a_0} y[n-k] + x[n],$$

- los sistemas ARMA son aquellos donde la salida depende de valores anteriores de la salida y de la entrada, la ecuación que rige su dinámica se presentó anteriormente¹.

4.5. Representación de sistemas LTI discretos

Una forma de representar sistemas LTI discretos es mediante diagramas de bloques; para esto es necesario definir 3 operaciones básicas: suma, multiplicación por un escalar y retardo. La representación de cada una de estas operaciones se muestra en la Figura 4.6.

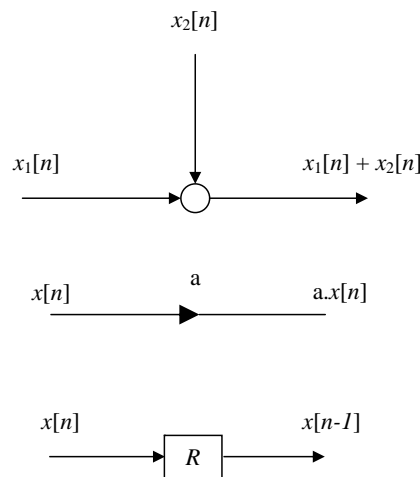


Figura 4.6. Representación de operaciones de un sistema LTI discreto. De arriba hacia abajo: sumador, multiplicación por un escalar, retraso unitario.

Por ejemplo, considere el sistema dado por la siguiente ecuación en diferencias $y[n] = 3x[n] + 2x[n-1]$, la representación utilizando diagrama de bloques se muestra en la Figura IV.7. Aquí los valores previos de la salida no se utilizan.

Esta situación no se observa en el diagrama del sistema $y[n] = 3x[n] - 2y[n-1]$ que se muestra en la Figura 4.8. En el diagrama de la Figura 4.8 se observa una realimentación de la salida, esto es consecuencia directa de la naturaleza recursiva de la ecuación.

¹Existen casos especiales en que un sistema ARMA puede comportarse como FIR ¿puede encontrar un ejemplo de este tipo?

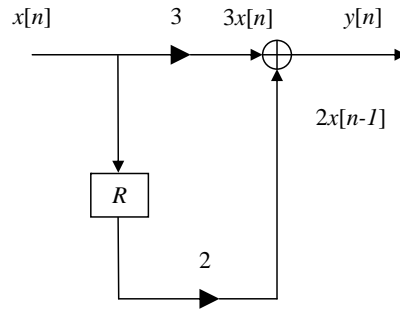


Figura 4.7. Diagrama de bloques del sistema $y[n] = 3x[n] + 2x[n - 1]$

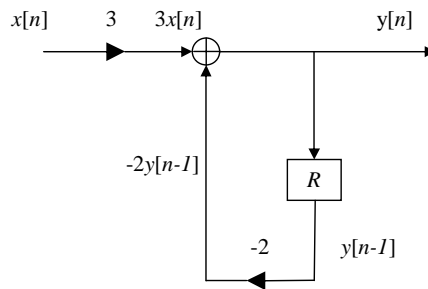


Figura 4.8. Diagrama de bloques del sistema $y[n] = 3x[n] - 2y[n - 1]$

Considere ahora el sistema $y[n] = 9x[n] + 6x[n-1] - 2y[n-1]$, éste se puede representar como la conexión en cascada de los 2 sistemas anteriores. En la Figura 4.9 se muestra el diagrama de bloques de dicho sistema, donde $w[n] = 3x[n] + 2x[n-1]$.

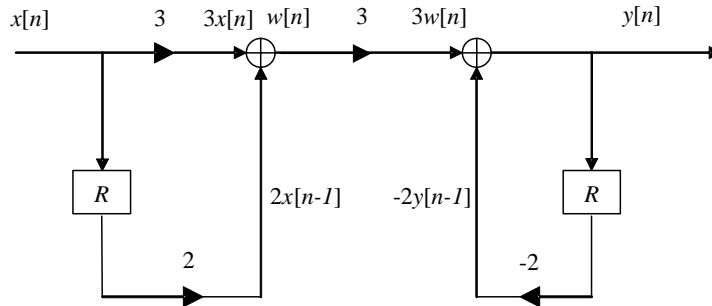


Figura 4.9. Diagrama de bloques del sistema $y[n] = 9x[n] + 6x[n-1] - 2y[n-1]$

4.6. Preguntas

1. ¿Qué formas pueden tomar las reglas de comportamiento de un sistema? Ejemplifique.
2. ¿Qué relación conceptual existe entre las ecuaciones diferenciales y las ecuaciones en diferencias?
3. ¿Todos los sistemas MA son FIR? Justifique.
4. ¿Todos los sistemas AR son IIR? Justifique.
5. ¿Todos los sistemas ARMA son IIR? Justifique.

4.7. Trabajos prácticos

Ejercicio 1: Para cada uno de los siguientes sistemas determine si son causales, lineales, invariantes en el tiempo y si poseen memoria. En cada caso grafique la salida del sistema $y[n]$ para una entrada dada.

1. $y[n] = g[n]x[n]$ donde $g[n] = A \sin(\omega nT)$ siendo A constante, $\omega = 2\pi f$ y T el período de muestreo.
2. $y[n] = \sum_{k=no}^n x[k]$
3. $y[n] = \sum_{k=n-no}^{n+no} x[k]$
4. $y[n] = x[n - no]$
5. $y[n] = e^{x[n]}$
6. $y[n] = x[n] + 2$
7. $y[n] = nx[n]$

Ejercicio 2: Los sistemas con la forma $y[n] = ax[n] + b$ son un caso particular de sistemas NO lineales que ha sido denominados *sistemas incrementalmente lineales*. Estos sistemas poseen la propiedad de responder en forma lineal a cambios en la entrada, es decir, la diferencia entre las respuestas de un sistema incremental lineal a dos entradas cualquiera es una función lineal de la diferencia de las dos entradas. Verifique matemáticamente la propiedad mencionada para el sistema $y[n] = 2x[n] + 3$.

Ejercicio 3: Considere un sistema representado por la ecuación en diferencias $y[n] - ay[n-1] = x[n]$ y con condición inicial $y[0] = 1$. Responda y justifique:

- ¿Es el sistema invariante en el tiempo?
- ¿Es el sistema lineal?
- Suponga que la condición inicial cambia a $y[0] = 0$, ¿modifica esto las respuestas de los puntos anteriores?

Ejercicio 4: Los sistemas LTI poseen dos interesantes propiedades:

- la salida es cero cuando la entrada es cero (y el sistema está inicialmente en reposo)
- no agregan componentes armónicas al espectro de frecuencias de la señal de entrada.

Proponga dos sistemas, uno lineal y otro no lineal, y verifique estas propiedades.

Ejercicio 5: Considere el diagrama en bloques de la Figura 4.10 y encuentre la expresión explícita de la señal de salida $y[n]$ en función de la señal de entrada $x[n]$.

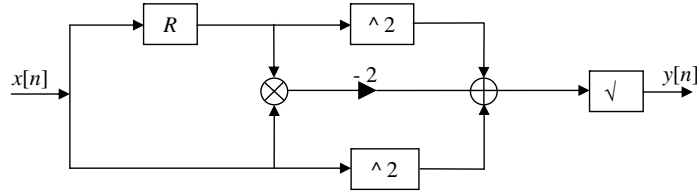


Figura 4.10. Diagrama en bloques para el Ejercicio 5.

Ejercicio 6: Considere el sistema LTI $y[n] - 0,5y[n-1] + 0,25y[n-2] = x[n]$ inicialmente en reposo. Encuentre el diagrama en bloques que lo representa y encuentre la salida del mismo para la entrada mostrada en la Figura 4.11.

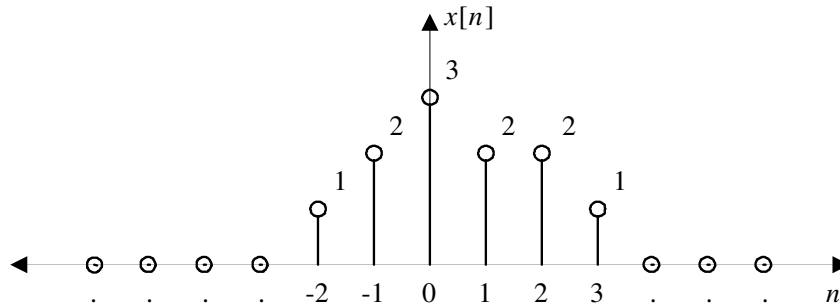


Figura 4.11. Entrada para Ejercicio 6.

Ejercicio 7: (*) Encuentre la respuesta al impulso de los sistemas LTI causales descritos por las siguientes ecuaciones en diferencias y clasifíquelos en función de ésta. Utilice condiciones iniciales nulas.

1. $y[n] - y[n-2] = x[n]$
2. $y[n] = x[n] + 0,5x[n-1]$

3. $y[n] - y[n - 1] = x[n] + 2x[n - 1]$

4. $y[n] - 0,5y[n - 1] + 0,25y[n - 2] = x[n]$

5. $y[n] = x[n] + x[n - 1] - y[n - 1]$

Ejercicio 8: ^(*) Represente los sistemas del ejercicio anterior mediante diagramas en bloque y encuentre sus respuestas al escalón unitario.

Bibliografía

- [1] L.E. Franks. *Teoría de la señal*. Reverté, Barcelona, 1975.
- [2] R.A. Gabel and R.A. Roberts. *Señales y sistemas lineales*. Ed. Limusa S.A., México, 1975.
- [3] H. Kwakernaak, R. Sivan, and R.C.W. Strijbos. *Modern Signals and Systems*. Prentice Hall, New Jersey, 1991.
- [4] A.V. Oppenheim, A.S. Willsky, S. H. Nawab, and G. Mata Hernández. *Señales y sistemas*. Prentice-Hall Hispanoamericana, México, 2a. (español) edition, 1998.
- [5] A. Papoulis. *Sistemas y Circuitos Digitales y Analógicos*. Marcombo, 1978.
- [6] N.K. Sinha. *Linear systems*. John Wiley, New York, 1991.
- [7] H. Skilling. *Circuitos en Ingeniería Eléctrica*. Cia. Ed. Continental, México, 1987.

Capítulo 5

Convolución discreta

Rubén Acevedo

Temas a tratar

- Convolución lineal: sumatoria de convolución.
- Propiedades de la sumatoria de convolución.
- Convolución circular.
- Deconvolución.

Objetivos

- Entender el concepto de la convolución lineal en tiempo discreto.
- Comprender y aplicar la relación entre la convolución circular y la Transformada Discreta de Fourier.

5.1. Introducción

La superposición es una de las propiedades más importantes de los *sistemas lineales e invariantes en el tiempo* (LTI). Esto significa que se puede representar la señal de entrada a un sistema LTI en términos de un conjunto de señales básicas y utilizar este principio para determinar la salida del sistema en términos de sus respuestas a estas señales básicas.

El impulso unitario se puede utilizar como bloque elemental para construir muchas señales de tipo general. Este hecho, junto con las propiedades de invarianza temporal y superposición permiten desarrollar una caracterización completa de sistemas LTI en términos de su respuesta al impulso. Esta representación, conocida como la *integral de convolución* en tiempo continuo, y la *sumatoria de convolución* en tiempo discreto, proporciona una considerable ventaja en el análisis de sistemas LTI. Las propiedades de esta última son de interés debido a su aplicación en procesamiento digital de señales.

La convolución es uno de los procesos más importantes y eficaces en el análisis de sistemas LTI, ya que permite establecer una relación entre la entrada y la salida en el dominio del tiempo y el de la frecuencia. Una multiplicación en el dominio del tiempo implica una convolución en la frecuencia o a la inversa, una multiplicación en el dominio de la frecuencia implica una convolución en el tiempo.

Para el caso bidimensional, que es una extensión directa del unidimensional, existen numerosas aplicaciones relacionadas con el procesamiento y filtrado de imágenes donde la variable independiente pasa a ser el espacio y no el tiempo.

5.2. Convolución lineal

Para entender el concepto de la convolución lineal es apropiado comenzar con la representación de una señal de tiempo discreto en términos de funciones impulso unitarios $\delta[n]$, equivalentes a las tipo *Delta de Dirac* para el caso continuo. En la Figura 5.1 se representan cinco secuencias, la primera corresponde a la señal completa y las cuatro restantes corresponden a impulsos escalados y desplazados en tiempo. El escalamiento de cada impulso es igual al valor de $x[n]$ en el instante de tiempo en el que se encuentra dicha muestra unitaria.

De acuerdo a esto, la suma de las cuatro secuencias es igual a $x[n]$ para

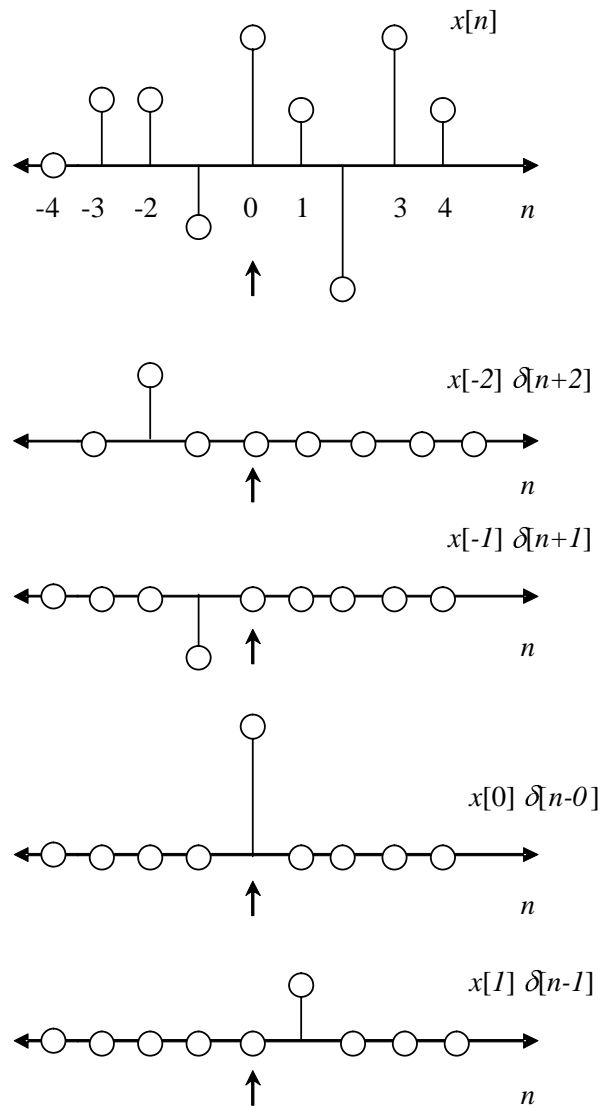


Figura 5.1. Descomposición de una señal de tiempo discreto en una suma de secuencia de impulsos ponderados y desplazados.

$-2 \leq n \leq 1$ y tiene la siguiente expresión:

$$x[-2]\delta[n+2] + x[-1]\delta[n+1] + x[0]\delta[n] + x[1]\delta[n-1]$$

Es decir, al incluir impulsos adicionales escalados y desplazados, se puede escribir:

$$x[n] = \cdots + x[-2]\delta[n+2] + x[-1]\delta[n+1] + x[0]\delta[n] + x[1]\delta[n-1] + \cdots$$

Es importante notar que para un valor dado de n sólo uno de los términos del lado derecho de la ecuación anterior es diferente de cero, y el escalamiento de ese término es $x[n]$.

En forma compacta se puede escribir la ecuación anterior de la siguiente manera:

$$x[n] = \sum_{k=-\infty}^{+\infty} x[k]\delta[n-k]$$

Ésta corresponde a la representación de una secuencia arbitraria como una combinación lineal de secuencias de impulsos unitarios desplazados $\delta[n-k]$, donde los pesos de esta combinación son $x[k]$.

Considere un sistema LTI, con una entrada arbitraria dada por la ecuación anterior, y $h[n]$ como la respuesta al impulso del sistema. Debido a la característica de invarianza temporal del sistema, si $h[n]$ es la respuesta a $\delta[n]$ entonces $h[n-m]$ es la respuesta a $\delta[n-m]$. Además, considerando que el sistema es lineal se puede aplicar el principio de superposición, de forma tal de obtener la salida $y[n]$ como una combinación lineal de las respuestas del sistema a impulsos desplazados. Entonces la respuesta del sistema a una entrada arbitraria $x[n]$ se puede expresar como:

$$y[n] = \sum_{k=-\infty}^{+\infty} x[k]h[n-k]$$

Este resultado se conoce como *sumatoria de convolución*, y la operación del lado derecho de la ecuación se conoce como la *convolución lineal* de las secuencias $x[n]$ y $h[n]$ y se representa con la notación $y[n] = x[n] * h[n]$.

En la Figura 5.2 se muestra el proceso de obtención de la salida $y[n]$, de un sistema LTI, a la entrada representada por la secuencia $x[n] = [1, 2, 2]$ y respuesta impulsional dada por la secuencia $h[n] = [2, 1, 0, 5]$. La entrada puede expresarse como una sumatoria de secuencias impulsos desplazados y escalados, de forma tal que $x[n] = x_0[n] + x_1[n] + x_2[n]$. En la columna derecha de la Figura 5.2 se muestra la respuesta del sistema a cada una

de las $x_i[n]$, las cuales consisten en respuestas al impulso desplazadas y escaladas adecuadamente; finalmente la salida del sistema se expresa como $y[n] = \sum_{i=0}^2 x[i]h[n-i]$. Este procedimiento se puede implementar como una multiplicación término a término similar a la que se utiliza para operar con polinomios. Considere las secuencias $x[n]$ y $h[n]$ definidas en el párrafo anterior, la multiplicación término a término resulta:

$$\begin{array}{rcccccl}
 h[n] & 2 & 1 & 0,5 & & \\
 x[n] & 1 & 2 & 2 & & \\
 \hline
 y_0[n] & 2 & 1 & 0,5 & \rightarrow x[0]h[n] & \\
 y_1[n] & & 4 & 2 & 1 & \rightarrow x[1]h[n-1] \\
 y_2[n] & & & 4 & 2 & 1 \rightarrow x[2]h[n-2] \\
 \hline
 y[n] & 2 & 5 & 6,5 & 3 & 1 \rightarrow y_0[n] + y_1[n] + y_2[n]
 \end{array}$$

Desarrollando la sumatoria de convolución se observa que la salida para cada instante de tiempo tiene la forma:

$$\begin{aligned}
 y[0] &= h[0]x[0] \\
 y[1] &= h[1]x[0] + h[0]x[1] \\
 y[2] &= h[2]x[0] + h[1]x[1] + h[0]x[2] \\
 &\vdots \\
 y[N] &= h[N]x[0] + \sum_{i=0}^{N-1} h[i]x[N-i]
 \end{aligned}$$

Este proceso se muestra en la Figura 5.3, y será utilizado para describir más adelante el procedimiento para calcular la convolución circular entre dos secuencias.

Finalmente, la operación de convolución se puede representar de forma matricial $\mathbf{y} = \mathbf{H}\mathbf{x}$, donde \mathbf{x} e \mathbf{y} son vectores de N muestras y \mathbf{H} una matriz de $N \times N$. A modo de ejemplo suponga $N = 4$, entonces:

$$\begin{bmatrix} y[0] \\ y[1] \\ y[2] \\ y[3] \end{bmatrix} = \begin{bmatrix} h[0] & 0 & 0 & 0 \\ h[1] & h[0] & 0 & 0 \\ h[2] & h[1] & h[0] & 0 \\ h[3] & h[2] & h[1] & h[0] \end{bmatrix} \cdot \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix}$$

Como se observa, de la expresión sólo se obtienen las primeras N muestras correspondientes a la operación $y = x * h$. De todas maneras esta forma de cálculo es útil debido a que se puede operar fácilmente para realizar

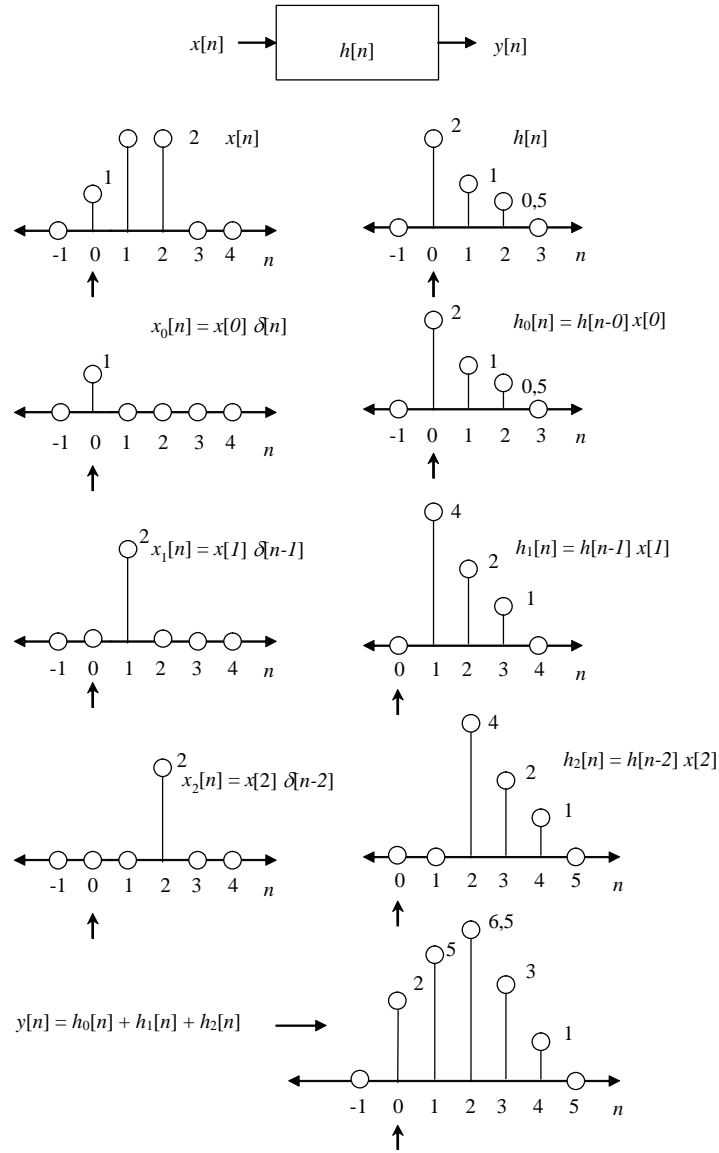


Figura 5.2. Obtención de la salida de un sistema LTI mediante la convolución lineal.

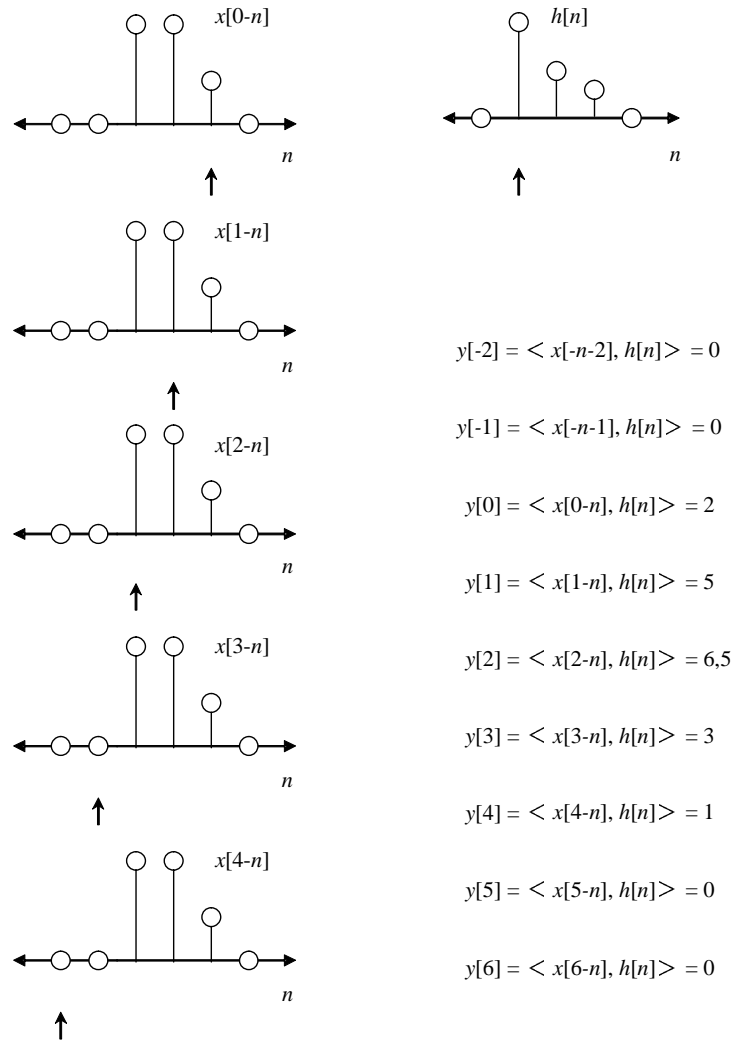


Figura 5.3. Convolución lineal de dos secuencias por el método de espejado.

la deconvolución, como se verá más adelante, y obtener la entrada $x[n]$ o la respuesta al impulso $h[n]$.

5.3. Convolución circular

Considere 2 secuencias $x_1[n]$ y $x_2[n]$, con una longitud de N muestras cada una, y $X_1[k]$ y $X_2[k]$ sus respectivas Transformadas Discretas de Fourier (TDF), multiplicando estas últimas entre sí se obtiene una secuencia de N muestras.

En tiempo continuo se cumple que $x(t) * y(t) \xleftrightarrow{TF} X(f)Y(f)$, donde TF significa Transformada de Fourier. Sin embargo en tiempo discreto la relación equivalente $x[n] * y[n] \xleftrightarrow{TDF} X[k]Y[k]$ no se cumple. En un primer análisis se puede observar que la cantidad de muestras no coincide, pero ni siquiera los valores del resultado son iguales para algún intervalo de N muestras.

Las ecuaciones correctas para el caso de tiempo discreto son:

$$x_1[n] \otimes x_2[n] \xleftrightarrow{TDF} X_1[k]X_2[k]$$

y

$$x_1[n]x_2[n] \xleftrightarrow{TDF} X_1[k] \otimes X_2[k]$$

porque es posible interpretar la operación \otimes como una convolución de dos secuencias, en la cual en vez de desplazar una secuencia en forma lineal sobre la otra, ambas se disponen en círculos concéntricos y los desplazamientos relativos se convierten en rotaciones (ver Figura 5.4).

5.4. Relación entre convolución lineal y circular

Es posible obtener la convolución lineal de dos secuencias utilizando la convolución circular. A continuación se resume la secuencia de pasos necesarios para llevar a cabo esta operación:

$$\begin{array}{lcl} x_1[n] \rightarrow x_{1m}[n] \rightarrow & X_{1m}[k] & \\ & \downarrow & \\ & X_{1m}[k]X_{2m}[k] \rightarrow x_{1m}[n] \otimes x_{2m}[n] \rightarrow x_1[n] * x_2[n] & \\ & \nearrow & \\ x_2[n] \rightarrow x_{2m}[n] \rightarrow & X_{2m}[k] & \end{array}$$

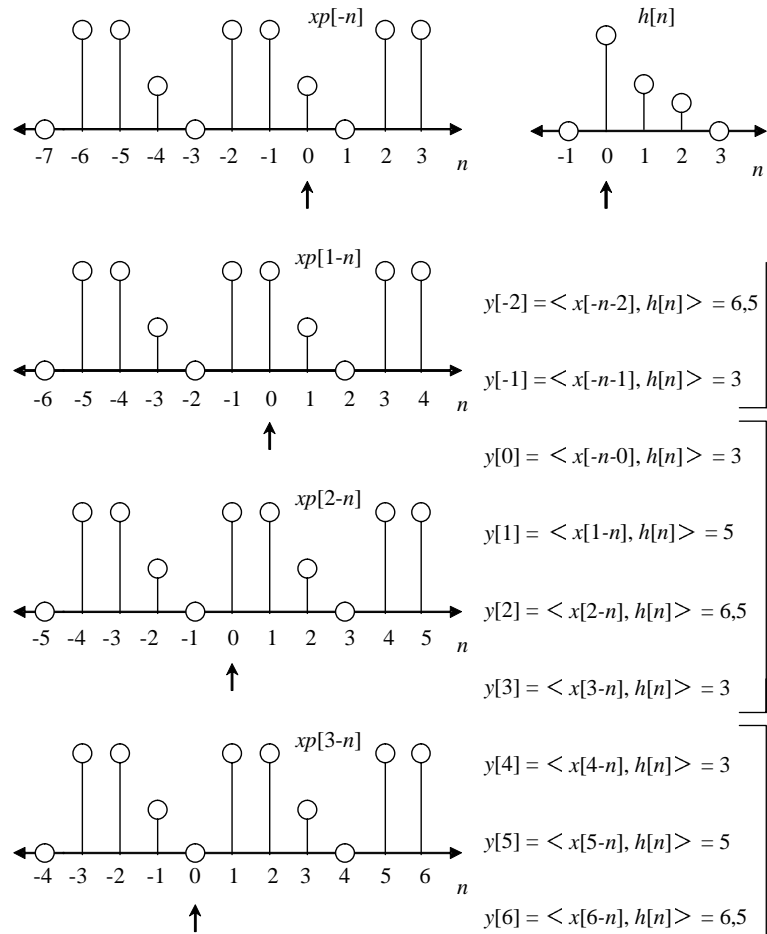


Figura 5.4. Convolución circular (o periódica) de dos secuencias.

Se supone que tanto $x_1[n]$ como $x_2[n]$ poseen N muestras, entonces el primer paso consiste en modificar cada una de las secuencias agregándoles $N - 1$ ceros, de forma tal de formar las secuencias $x_{1m}[n]$ y $x_{2m}[n]$ de $2N - 1$ muestras de longitud. Luego se calcula la TDF de cada secuencia, se las multiplica entre sí y se calcula la TDFI. El resultado es la convolución lineal entre $x_1[n]$ y $x_2[n]$.

5.5. Deconvolución

La deconvolución es la operación inversa de la convolución. Esto es, si un sistema LTI produjo una salida correspondiente a una señal de entrada o excitación a la cual no tenemos acceso, podremos recuperarla mediante la deconvolución y de ahí su importancia.

Una de las formas para obtener la deconvolución de dos secuencias es la división término a término, que es la operación recíproca de la multiplicación término a término. Considere las secuencias de la Figura 5.2 y suponga que se quiere determinar la secuencia $h[n]$ a partir de la deconvolución de $y[n]$ y $x[n]$. La forma de operar es análoga a la división entre polinomios. Por ejemplo, la división término a término de izquierda a derecha es:

$$\begin{array}{r}
 y[n] \rightarrow 2 \ 5 \ 6,5 \ 3 \ 1 \quad \left| \begin{array}{r} 1 \ 2 \ 2 \\ 2 \ 1 \ 0,5 \end{array} \right. \begin{array}{l} \leftarrow x[n] \\ \leftarrow h[n] \end{array} \\
 \underline{2 \ 4 \ 4} \\
 0 \ 1 \ 2,5 \ 3 \\
 \underline{1 \ 2 \ 2} \\
 0 \ 0,5 \ 1 \ 1 \\
 \underline{0,5 \ 1 \ 1} \\
 0 \ 0 \ 0
 \end{array}$$

y de derecha a izquierda:

$$\begin{array}{r}
 y[n] \rightarrow 2 \ 5 \ 6,5 \ 3 \ 1 \quad \left| \begin{array}{r} 1 \ 2 \ 2 \\ 2 \ 1 \ 0,5 \end{array} \right. \begin{array}{l} \leftarrow x[n] \\ \leftarrow h[n] \end{array} \\
 \underline{0,5 \ 1 \ 1} \\
 5 \ 6 \ 2 \ 0 \\
 \underline{1 \ 2 \ 2} \\
 2 \ 4 \ 4 \ 0 \\
 \underline{2 \ 4 \ 4} \\
 0 \ 0 \ 0
 \end{array}$$

En caso de que la convolución se hubiera hecho de la forma $\mathbf{y} = \mathbf{H} \cdot \mathbf{x}$, entonces se puede hacer la deconvolución de la siguiente manera: $\mathbf{H}^{-1} \cdot \mathbf{y} = \mathbf{x}$, donde \mathbf{H}^{-1} es la matriz inversa de \mathbf{H} , lo que suele llamarse el caso de *control*.

De forma análoga si planteamos ahora $\mathbf{y} = \mathbf{X} \cdot \mathbf{h}$ entonces $\mathbf{X}^{-1} \cdot \mathbf{y} = \mathbf{h}$, lo que suele llamarse el caso de *identificación*.

Como la deconvolución corresponde a lo que suele denominarse como un *problema inverso*, presenta algunas particularidades comunes a este tipo de problemas. Un aspecto a considerar al momento de realizar la deconvolución es la presencia de una señal de ruido $r[n]$. Ésta puede aparecer en la entrada del sistema (Figura 5.5) o en la salida del mismo (Figura 5.6) y dependiendo de dónde aparezca afectará más o menos a la deconvolución.

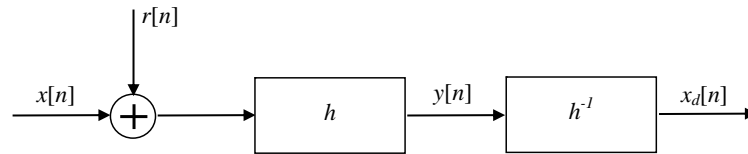


Figura 5.5. Deconvolución con ruido aditivo en la entrada.

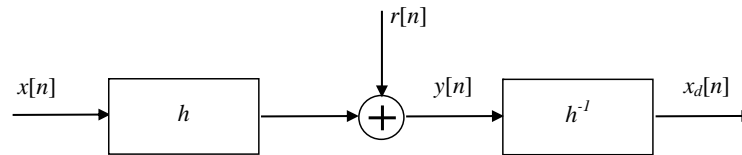
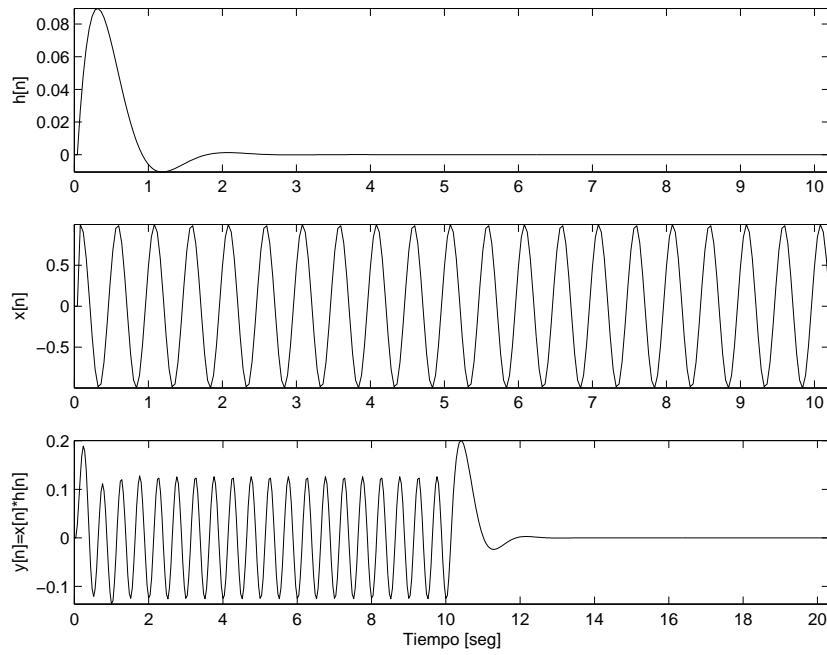


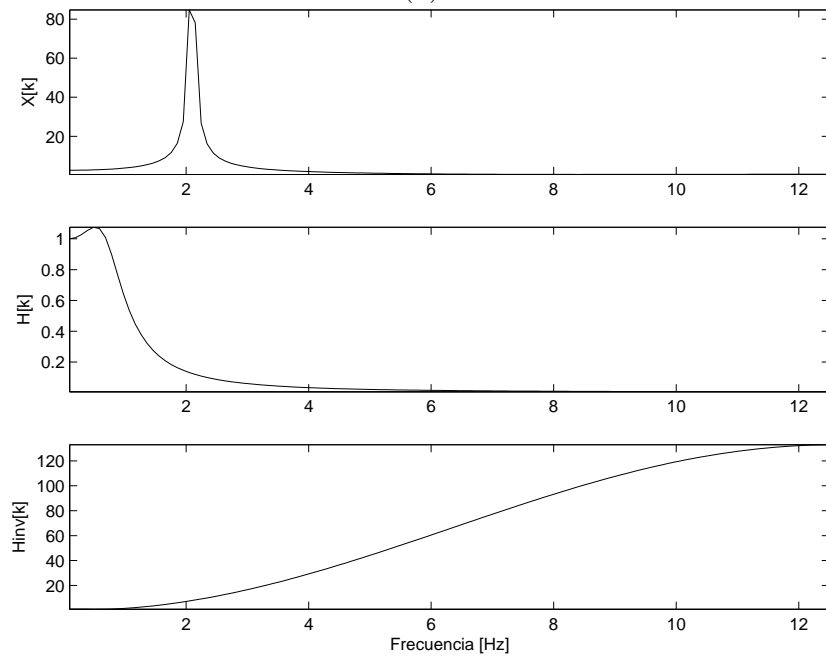
Figura 5.6. Deconvolución con ruido aditivo en la salida.

En la Figura 5.7 (a) se muestra la respuesta al impulso $h[n]$ de un sistema, la entrada $x[n]$ y la salida $y[n]$ obtenida de la forma $y = x * h$; en la Figura 5.7 (b) se muestra el espectro de frecuencia de la entrada $X[k]$, la respuesta en frecuencia del sistema $H[k]$ y la del sistema inverso $H^{-1}[k]$.

En la Figura 5.8 se muestra cómo influye el ruido en la salida del sistema al realizar la deconvolución de ésta con la respuesta al impulso del mismo. En la Figura 5.8 (a) se muestra la señal resultante de la deconvolución sin ruido y cuando en la salida del sistema se adiciona ruido de 5 y 10 Hz. En la Figura 5.8 (b) se muestran los espectros de frecuencia de las señales resultantes.

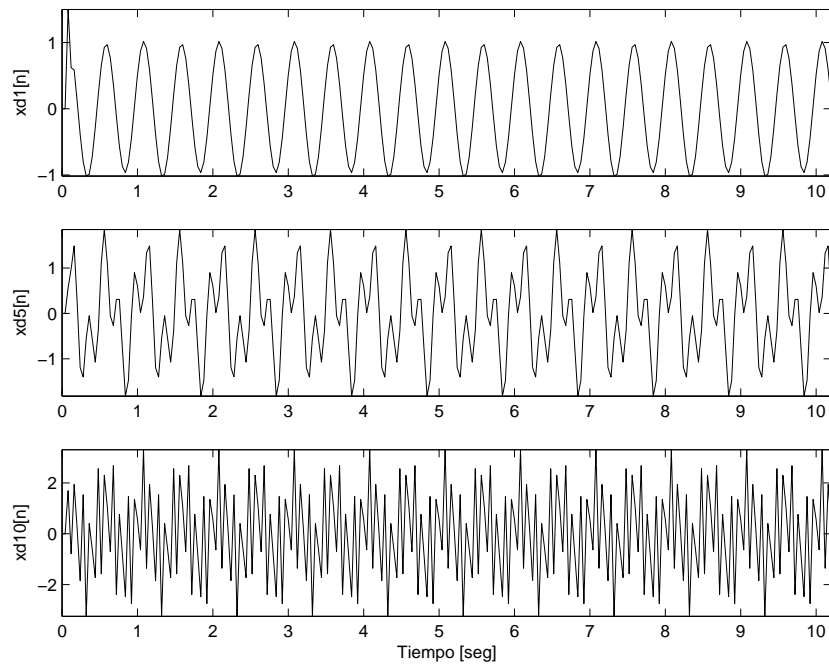


(a)

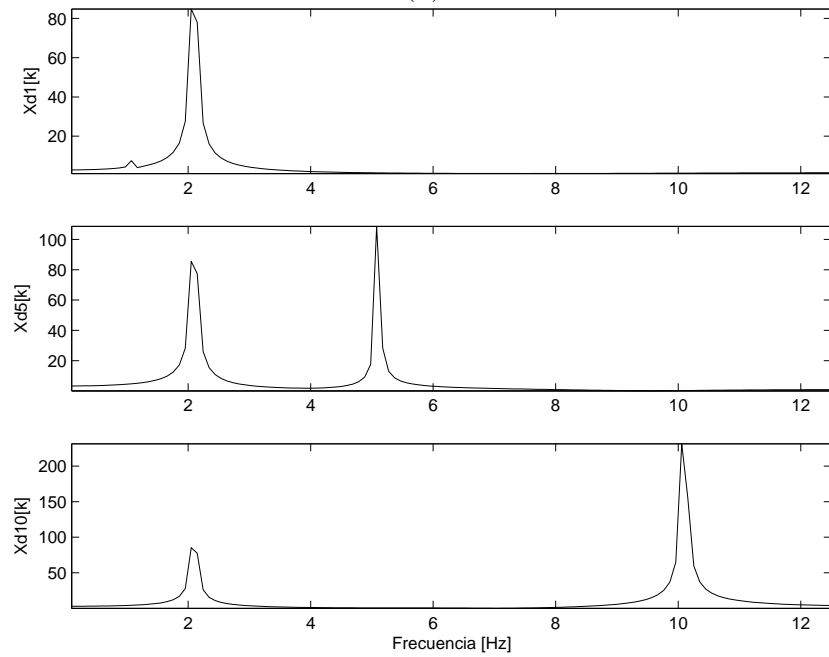


(b)

Figura 5.7. Señales de entrada, respuesta al impulso y salida de un sistema LTI.



(a)



(b)

Figura 5.8. Influencia del espectro de frecuencias del ruido en la deconvolución.

5.6. Preguntas

1. Demuestre las siguientes propiedades de la convolución lineal de dos señales continuas:

- Si $y(t) = x(t) * h(t)$ entonces $Y(w) = X(w)H(w)$.
- Si $y(t) = x(t)h(t)$ entonces $Y(w) = X(w) * H(w)$.

donde $Y(w)$, $X(t)$ y $H(w)$ son las transformadas de Fourier de $y(t)$, $x(t)$ y $h(t)$ respectivamente.

2. Suponga que un sistema posee una respuesta impulsional $h[n]$ que varía en el tiempo de acuerdo a una ley conocida. Considerando esto, ¿es posible encontrar la salida del sistema mediante la convolución lineal de la entrada $x[n]$ con $h[n]$? Proponga un ejemplo para responder la pregunta.

5.7. Trabajos prácticos

Ejercicio 1: Realice las tres operaciones siguientes y comente los resultados.

1. multiplique 121 por 311.
2. multiplique los polinomios $1 + 2x + x^2$ y $3 + x + x^2$
3. calcule la convolución de las señales $[1, 2, 1]$ y $[3, 1, 1]$.

Ejercicio 2: Dado el sistema $6y[n] - 4y[n - 1] + 5y[n - 2] = x[n] - 2x[n - 1] + x[n - 2]$, inicialmente en reposo, obtenga la respuesta al escalón unitario mediante la ecuación en diferencias y luego compárela con la calculada mediante la sumatoria de convolución, para lo que deberá encontrar previamente su respuesta al impulso.

Ejercicio 3: Considere un sistema LTI con respuesta al impulso $h[n]$ y muestre que cuando la entrada $x[n]$ es una secuencia periódica con período N , la salida $y[n]$ también es periódica con el mismo período.

Ejercicio 4: (*) Defina tres señales cualquiera y muestre numéricamente las siguientes propiedades de la convolución:

1. conmutativa: $\mathbf{y} * \mathbf{x} = \mathbf{x} * \mathbf{y}$
2. asociativa: $\mathbf{x} * (\mathbf{y} * \mathbf{w}) = (\mathbf{x} * \mathbf{y}) * \mathbf{w}$
3. distributiva con respecto a la suma: $\mathbf{x} * (\mathbf{y} + \mathbf{w}) = \mathbf{x} * \mathbf{y} + \mathbf{x} * \mathbf{w}$

Ejercicio 5: Verifique las condiciones de aplicabilidad para la propiedad:

$$\mathbf{x} * \mathbf{y} = \mathcal{F}^{-1}\{\mathcal{F}\{\mathbf{x}\}\mathcal{F}\{\mathbf{y}\}\}$$

utilizando señales de N muestras y comparando los resultados de la convolución calculada mediante:

1. la sumatoria de convolución con ciclos `for`,
2. la función `conv`,
3. la función `filter`,
4. las funciones `fft` y `ifft` utilizadas directamente como lo indica la propiedad,
5. las funciones `fft` e `ifft` pero agregando $N - 1$ ceros tanto a \mathbf{x} como a \mathbf{y} .

Ejercicio 6: Considere dos sistemas LTI conectados en cascada (Figura 5.9), con respuestas al impulso dadas por $h_A[n] = \sin(8n)$ y $h_B[n] = a^n u[n]$, donde $a \in \mathbb{R}$, $|a| < 1$ y $u[n]$ es la función escalón unitario. Determine la salida $y[n]$ para una entrada $x[n] = \delta[n] - a\delta[n - 1]$.

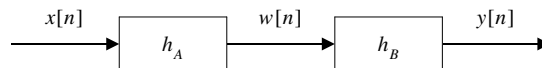


Figura 5.9. Sistemas en cascada.

Ejercicio 7: (*) La relación entrada-salida de dos sistemas LTI conectados en cascada no depende del orden en el que están conectados. Este hecho, conocido como propiedad conmutativa, depende tanto de la linealidad como de la invariancia en el tiempo de ambos sistemas. Considere dos sistemas A y B conectados como muestra la Figura 5.9. El sistema A es LTI con una respuesta impulsional dada por $h_A[n] = \left(\frac{1}{2}\right)^n u[n]$. El sistema B es lineal pero variante en el tiempo y da como salida $y[n] = nw[n]$ para una entrada $w[n]$.

1. Muestre que la propiedad conmutativa no se cumple si los dos sistemas no son LTI (suponga una entrada sencilla).
2. Muestre que la propiedad conmutativa se cumple redefiniendo el sistema B como LTI.
3. Redefina el sistema B según $y[n] = w[n] + 2$ y vuelva a verificar la propiedad.

Ejercicio 8: ^(*) En el archivo `pb100.txt` se encuentra la respuesta al impulso $h[n]$ de un sistema con frecuencia de muestreo 1000 Hz. Este sistema se comporta como un filtro pasa bajos con frecuencia de corte 100 Hz.

1. Utilice la convolución para filtrar la señal $x[n]$ consistente en la suma de cinco señales senoidales con frecuencias entre 50 y 150 Hz.
2. Utilice la deconvolución para obtener nuevamente la entrada $x[n]$ a partir de la salida $y[n]$ y la respuesta al impulso $h[n]$.
3. Evalúe el efecto del ruido en el proceso de deconvolución para las dos situaciones que se muestran en la Figura 5.10. En ambos casos analice lo que ocurre cuando:
 - el ancho de banda de $r[n]$ es inferior a 100 Hz.
 - $r[n]$ tiene componentes frecuenciales entre 100 y 300 Hz.

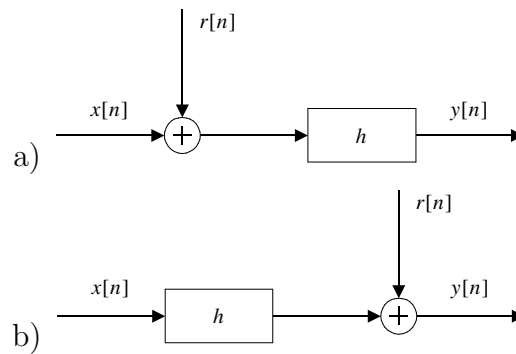


Figura 5.10. Sistema con ruido aditivo a) en la entrada y b) en la salida.

Bibliografía

- [1] H. Kwakernaak, R. Sivan, and R.C.W. Strijbos. *Modern Signals and Systems*. Prentice Hall, New Jersey, 1991.
- [2] A.V. Oppenheim and R. Schaffer. *Digital Signal Processing*. Pearson Higher Education, 1986.
- [3] A.V. Oppenheim, A.S. Willsky, S. H. Nawab, and G. Mata Hernández. *Señales y sistemas*. Prentice-Hall Hispanoamericana, México, 2 edition, 1998.
- [4] J.G. Proakis and D.G. Manolakis. *Tratamiento digital de señales*. Prentice Hall, 1 edition, 1998.
- [5] N.K. Sinha. *Linear systems*. John Wiley, New York, 1991.

Capítulo 6

Transformada Z

Humberto Torres, Rubén Acevedo

Temas a tratar

- Representación de sistemas de tiempo discreto.
- Mapeos $s - z$.
- Función de transferencia en z .
- Respuesta en frecuencia de sistemas de tiempo discreto.

Objetivos

- Utilizar la Transformada Z como herramienta para obtener la expresión de tiempo discreto de un sistema a partir de la ecuación diferencial que rige su dinámica.
- Obtener la respuesta en frecuencia de sistemas de tiempo discreto.
- Analizar las limitaciones de las transformaciones conformes utilizadas.

6.1. Introducción

En muchos casos se desea transformar un sistema de tiempo continuo en un sistema de tiempo discreto. Este tipo de transformaciones se ha visto favorecida por el advenimiento de la tecnología digital de alta densidad y alta velocidad, especialmente en sistemas de filtrado y procesamiento de señales que antes eran, esencialmente, de tiempo continuo. La flexibilidad que proveen estos dispositivos es una ventaja importante con respecto a la implementación tradicional. Este es el caso de los sistemas basados en *los Procesadores de Señales Digitales* o DSP. Sin embargo, en aplicaciones de muy alta frecuencia los sistemas analógicos son todavía irremplazables.

Mediante el empleo de transformaciones conformes, como las que serán mencionadas en este capítulo, los diseños de sistemas de tiempo continuo pueden ser transformados en sistemas de tiempo discreto con las especificaciones correspondientes.

6.2. Definición de Transformada Z

La Transformada Z de una secuencia de tiempo discreto $x[n]$ se define como:

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

donde z es una variable compleja de la forma $z = re^{j\Omega}$, r es la magnitud de z y Ω es el ángulo. La relación entre $x[n]$ y su Transformada Z se indica como:

$$x[n] \xleftrightarrow{Z} X(z).$$

Podría decirse que la transformada $X(z)$ es la Transformada de Laplace de la secuencia discreta $x[n]$. Para demostrarlo es posible aproximar $x[n]$ mediante $x_{\delta_T}(t)$, la versión muestreada de $x(t)$, obtenida mediante la modulación de la señal continua con un tren de pulsos tipo *delta de Dirac* equi-espaciados:

$$x[n] \approx x_{\delta_T}(t) = x(t) \sum_{n=-\infty}^{+\infty} \delta(t - nT) = \sum_{n=-\infty}^{+\infty} x(nT) \delta(t - nT)$$

entonces la Transformada de Laplace de $x_{\delta_T}(t)$ está dada por:

$$X_{\delta_T}(s) = \int_{-\infty}^{+\infty} \left[\sum_{n=-\infty}^{+\infty} x(nT) \delta(t - nT) \right] e^{-st} dt$$

Intercambiando el orden de la sumatoria y la integral:

$$X_{\delta_T}(s) = \sum_{n=-\infty}^{+\infty} x(nT) \left[\int_{-\infty}^{+\infty} \delta(t - nT) e^{-st} dt \right]$$

integrando se obtiene:

$$X_{\delta_T}(s) = \sum_{n=-\infty}^{+\infty} x(nT) e^{-snT}$$

Si comparamos esta ecuación con la definición de la Transformada Z, reemplazando previamente $x(nT)$ por $x[n]$, encontramos que basta la siguiente relación para igualarlas:

$$z = e^{sT}$$

Esta igualdad resulta clave para encontrar la relación entre la Transformada Z con otras transformaciones lineales de interés.

Si ahora sustituimos z por su expresión en forma polar se puede interpretar $X(z)$ en términos de la Transformada de Fourier:

$$X(z)|_{z=re^{jw}} = \sum_{n=-\infty}^{+\infty} x[n] (re^{jw})^{-n} = \sum_{n=-\infty}^{+\infty} x[n] r^{-n} e^{-jwn}$$

cuando $r = 1$ se obtiene

$$X(z)|_{z=re^{jw}} = \sum_{n=-\infty}^{+\infty} x[n] e^{-jwn}$$

De esta última ecuación se observa que la Transformada de Fourier de una secuencia discreta es en realidad la Transformada Z de la secuencia evaluada sobre el círculo unitario. Si esta evaluación se realiza sobre una única vuelta del círculo unitario y a intervalos discretos determinados, entonces estamos en el caso de la Transformada Discreta de Fourier.

6.2.1. Convergencia de la Transformada Z

La Transformada Z no converge para todas las secuencias $x[n]$, ni para todos los valores de z . Para una determinada secuencia, el conjunto de valores de z para los cuales la Transformada Z converge se denomina *región de convergencia*. Para que la Transformada Z de una secuencia sea convergente es necesario que la serie sea absolutamente sumable, es decir:

$$\begin{aligned} \sum_{n=-\infty}^{+\infty} |x[n] z^{-n}| &< M \\ \sum_{n=-\infty}^{+\infty} |x[n] r^{-n} e^{-j\theta n}| &< M \\ \sum_{n=-\infty}^{+\infty} |x[n]| r^{-n} &< M \end{aligned}$$

Luego, la convergencia o no de la Transformada Z viene determinada por los coeficientes $x[n]$. Por ejemplo, sea la secuencia $x[n] = a^n u[n]$; entonces:

$$X(z) = \sum_{n=-\infty}^{+\infty} a^n u(n) z^{-n} = \sum_{n=0}^{+\infty} a^n z^{-n}$$

y entonces:

$$\begin{aligned} \sum_{n=0}^{+\infty} |a^n z^{-n}| &= \sum_{n=0}^{+\infty} |a^n r^{-n} e^{-j\theta n}| \\ &= \sum_{n=0}^{+\infty} |a r^{-1} e^{-j\theta}|^n \\ &= \sum_{n=0}^{+\infty} \left| \frac{a}{r} \right|^n < M \end{aligned}$$

La Transformada Z es convergente solo si $r > a$, entonces $|z| > a$ y converge a:

$$\begin{aligned}
 X(z) &= \sum_{n=-\infty}^{+\infty} a^n u(n) z^{-n} \\
 &= \sum_{n=0}^{+\infty} a^n z^{-n} \\
 &= \sum_{n=0}^{+\infty} (az^{-1})^n \\
 &= 1 + az^{-1} + a^2 z^{-2} + \dots \\
 &= \frac{1}{1 - az^{-1}} \\
 &= \frac{z}{z - a}
 \end{aligned}$$

Ya que la Transformada Z es función de una variable compleja, es conveniente describirla e interpretarla usando el plano complejo, tal como se indica en la Figura 6.1.

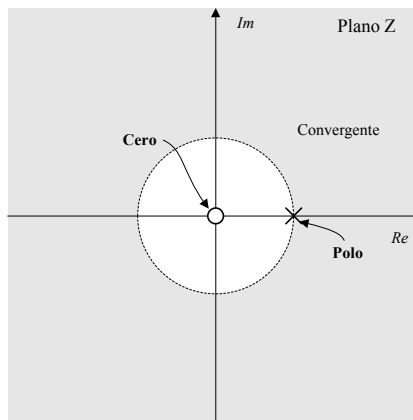


Figura 6.1. Representación de $X(z)$ en el plano complejo

Un grupo importante de Transformadas Z está constituido por aquellas funciones $X(z)$ que son racionales, es decir un cociente de polinomios en z . En ellas, las raíces del numerador (valores de z tales que $X(z) = 0$), se denominan ceros de $X(z)$. Análogamente, a las raíces del denominador (valores de z que hacen que $X(z) \rightarrow \infty$) se les denomina polos de $X(z)$. No

puede haber polos en la región de convergencia, estos se encuentran en el límite de la región de convergencia, como se ve en el ejemplo de la Figura 6.1.

6.2.2. La Transformada Z inversa

Una de las aplicaciones más importantes de la Transformada Z es el análisis de *sistemas discretos lineales e invariantes en el tiempo* (LTI). Este análisis suele requerir calcular la Transformada Z inversa.

La Transformada Z inversa está formalmente definida como:

$$x[n] = \frac{1}{2\pi j} \oint_C X(z) z^{n-1} dz$$

donde la integral es una integral de línea sobre el camino cerrado C que encierra al origen y se encuentra en la región de convergencia de $X(z)$ en el plano z .

Existen tres métodos, frecuentemente utilizados en la práctica, para recuperar la secuencia original a partir de su Transformada Z:

- Cálculo directo de la integral.
- Expansión en serie de términos z y z^{-1}
- Expansión en fracciones simples y búsqueda en tabla.

El método del cálculo de la integral rara vez se utiliza, e implica la utilización del teorema del residuo de Cauchy.

Si la Transformada Z es una función racional, la expresión en forma de serie de potencias puede obtenerse fácilmente mediante división de polinomios. Podremos observar como precisamente los coeficientes asociados a cada uno de los términos z^{-n} de la serie son los valores de la secuencia, ya que por definición la Transformada Z es:

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

Un procedimiento más general consiste en realizar una Descomposición en Fracciones Simples e identificar las transformadas simples de los términos así obtenidos.

Sea

$$X(z) = \frac{P(z)}{Q(z)}$$

siendo M el orden de $P(z)$ y N el orden de $Q(z)$.

Si $M < N$ y sólo existen polos de primer orden, se puede escribir:

$$\begin{aligned} X(z) &= \frac{P(z)}{Q(z)} \\ &= \sum_{k=1}^N \frac{A_k}{z - z_k} \quad \text{siendo } A_k = \left. \frac{(z - p_k) X(z)}{z} \right|_{z=p_k} \end{aligned}$$

Si $M \geq N$ entonces:

$$\begin{aligned} X(z) &= B_{M-N} z^{M-N} + B_{M-N-1} z^{M-N-1} + \dots \\ &\quad \dots + B_1 z + B_0 + \sum_{k=1}^N \frac{A_k}{z - z_k} \end{aligned}$$

siendo los B_i los coeficientes obtenidos mediante división hasta que el resto sea de un orden igual al del denominador menos 1. Con este resto se procede a descomponer en fracciones simples y el resultado se añade al de la división.

6.3. Propiedades de la Transformada Z

Al igual que otras transformadas, la Transformada Z posee propiedades que la hacen una herramienta muy útil para el estudio de señales y sistemas de tiempo discreto.

1. *Linealidad*

Si $x_1[n] \xrightarrow{Z} X_1(z)$ y $x_2[n] \xrightarrow{Z} X_2(z)$ entonces

$$ax_1[n] + bx_2[n] \xrightarrow{Z} aX_1(z) + bX_2(z).$$

2. *Desplazamiento en el tiempo*

Si $x[n] \xrightarrow{Z} X(z)$ entonces $x[n - n_o] \xrightarrow{Z} X(z)z^{-n_o}$.

3. *Desplazamiento en la frecuencia*

Si $x[n] \xrightarrow{Z} X(z)$ entonces $e^{j\Omega_0 n} x[n] \xrightarrow{Z} X(e^{-j\Omega_0} z)$.

4. *Inversión en el tiempo*

Si $x[n] \xrightarrow{Z} X(z)$ entonces $x[-n] \xrightarrow{Z} X(\frac{1}{z})$.

5. *Propiedad de convolución*

Si $x_1[n] \xrightarrow{Z} X_1(z)$ y $x_2[n] \xrightarrow{Z} X_2(z)$ entonces

$$x_1[n] * x_2[n] \xrightarrow{Z} X_1(z)X_2(z).$$

6. *Diferenciación en el dominio Z*

Si $x[n] \xrightarrow{Z} X(z)$ entonces $nx[n] \xrightarrow{Z} -z \frac{dX(z)}{dz}$

6.4. Representación de sistemas discretos mediante la Transformada Z

En el caso de los sistemas LTI definidos por ecuaciones en diferencias, las propiedades de la Transformada Z proveen un procedimiento muy conveniente para obtener la función de transferencia, la respuesta en frecuencia o la respuesta temporal del sistema.

Si se considera el sistema definido por la ecuación

$$y[n] - \frac{1}{2}y[n-1] = x[n] + \frac{1}{3}x[n-1]$$

aplicando la definición de Transformada Z a ambos lados de la ecuación y utilizando las propiedades de linealidad y desplazamiento se obtiene

$$Y(z) - \frac{1}{2}Y(z)z^{-1} = X(z) + \frac{1}{3}X(z)z^{-1}$$

Realizando algunos pasos algebraicos se obtiene la función de transferencia del sistema de la forma

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + \frac{1}{3}z^{-1}}{1 - \frac{1}{2}z^{-1}}.$$

Este proceso se puede generalizar para cualquier ecuación en diferencias lineal y con coeficientes constantes.

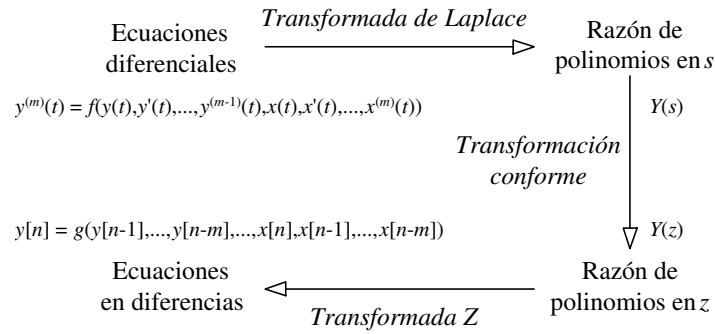


Figura 6.2. Proceso de obtención de la ecuación en diferencias mediante transformación conforme.

En el caso expuesto se dispone de la ecuación en diferencias y a partir de ésta se obtiene la función de transferencia del sistema; pero el caso de mayor interés es obtener la ecuación en diferencias a partir de la ecuación diferencial que define la dinámica del sistema en tiempo continuo.

Un método para obtener la ecuación en diferencias del sistema de tiempo discreto es utilizar transformaciones conformes que mapean el plano S en el plano Z como lo resume la Figura 6.2. Los pasos involucrados en el proceso son los siguientes :

- Obtener la función de transferencia del sistema en cuestión.
- Aplicar de una transformación conforme para mapear el plano s en el plano z y obtener de esta manera la expresión de la función de transferencia del sistema en el dominio de Z .
- Finalmente, aplicar la propiedad de desplazamiento para obtener la ecuación en diferencias del sistema.

Las transformaciones conformes deben cumplir ciertas condiciones impuestas por la relación $z = e^{sT}$, éstas se mencionan a continuación y se representan en la Figura 6.3:

Condición 1: el eje imaginario del plano s debe ser mapeado en el círculo unitario del plano z , esta condición es necesaria para preservar las características de respuesta en frecuencia del sistema continuo.

Condición 2: el semiplano izquierdo del plano s ($Re(s) < 0$) debe ser mapeado en el interior del círculo unitario del plano z .

Las condiciones son impuestas a toda transformación que intente mapear sistemas de tiempo continuo estables (polos en el semiplano izquierdo del plano s) en sistemas de tiempo discreto estables

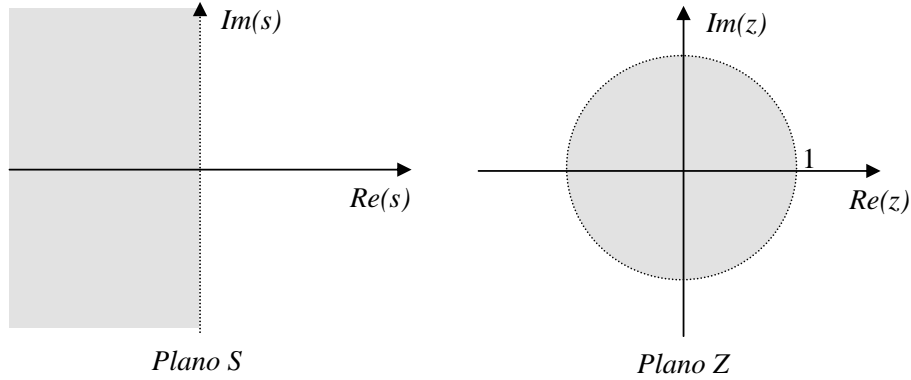


Figura 6.3. Condiciones de mapeo mediante transformación conforme.

(polos en el interior del círculo unitario con centro en el origen del plano z).

Hay varias transformaciones conformes que permiten realizar el mapeo mencionado anteriormente, dos de ellas son la transformación de Euler y la transformación bilineal, las cuales se describen a continuación.

6.4.1. Transformación de Euler

La transformación de Euler aproxima el cociente diferencial que define la derivada por un cociente incremental de la forma

$$\frac{dy}{dt} \approx \frac{\Delta y}{\Delta t}$$

A la derivada $\frac{dy}{dt}$, en el tiempo $t = nT$, la sustituimos por la *diferencia hacia atrás* (Figura 6.4)

$$\begin{aligned} \left. \frac{dy(t)}{dt} \right|_{t=nT} &= \frac{y(nT) - y(nT - T)}{T} \\ \left. \frac{dy(t)}{dt} \right|_{t=nT} &= \frac{y[n] - y[n - 1]}{T} \end{aligned}$$

Donde T representa el período de muestreo, e $y[n] = y(nT)$. El diferenciador analógico con salida $\frac{dy}{dt}$ tiene la función de transferencia $H(s) = s$, mientras el sistema digital que produce la salida $\frac{y(nT) - y(nT - T)}{T}$ tiene la

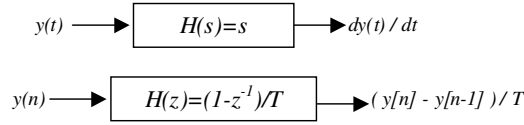


Figura 6.4. La sustitución de la diferencia hacia atrás por la derivada.

función de transferencia $H(z) = \frac{1-z^{-1}}{T}$. Consecuentemente, el equivalente en el dominio de la frecuencia es:

$$s = \frac{1 - z^{-1}}{T}$$

La segunda derivada $\frac{d^2y}{dt^2}$ se reemplaza por la segunda diferencia, que se deriva como sigue:

$$\begin{aligned} \left. \frac{d^2y(t)}{dt^2} \right|_{t=nT} &= \frac{d}{dt} \left[\frac{dy(t)}{dt} \right]_{t=nT} \\ \left. \frac{d^2y(t)}{dt^2} \right|_{t=nT} &= \frac{\frac{y(nT) - y(nT-T)}{T} - \frac{y(nT-T) - y(nT-2T)}{T}}{T} \\ \left. \frac{d^2y(t)}{dt^2} \right|_{t=nT} &= \frac{y[n] - 2y[n-1] + y[n-2]}{T^2} \end{aligned}$$

En el dominio de la frecuencia se tiene:

$$s^2 = \frac{1 - 2z^{-1} + z^{-2}}{T^2} = \left(\frac{1 - z^{-1}}{T} \right)^2$$

De este razonamiento se induce fácilmente que la k -ésima derivada de $y(t)$ resulta en la relación equivalente en el dominio de la frecuencia:

$$s^k = \left(\frac{1 - z^{-1}}{T} \right)^k$$

Consecuentemente, la función de transferencia obtenida como el resultado de la aproximación de las derivadas mediante diferencias finitas es:

$$H(z) = Ha(s)|_{s=(1-z^{-1})/T}$$

donde $Ha(s)$ es la función de transferencia analógica.

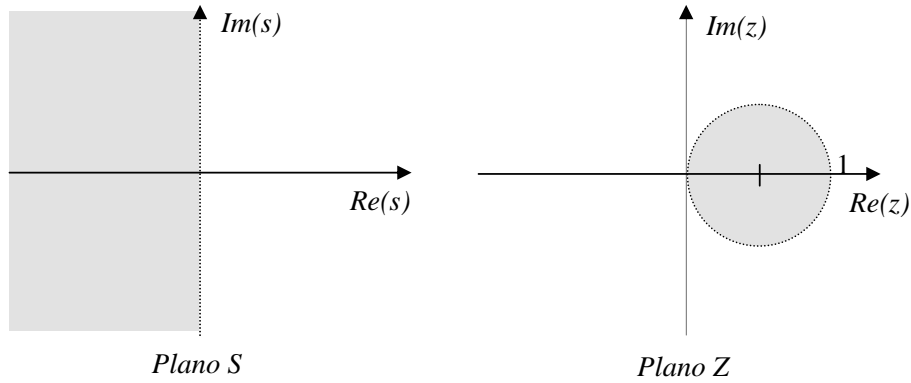


Figura 6.5. Transformación de Euler.

A continuación se analizan las implicaciones de la correspondencia del plano s al plano z . Haciendo $s = j\Omega$ se tiene que:

$$\begin{aligned} z &= \frac{1}{1 - sT} \\ z &= \frac{1}{1 - j\Omega T} \\ z &= \frac{1}{1 + \Omega^2 T^2} + j \frac{\Omega T}{1 + \Omega^2 T^2} \end{aligned}$$

Como $-\infty < \Omega < +\infty$, el correspondiente lugar de puntos en el plano z es un círculo de radio $z = 1/2$ y con centro en $z = 1/2$, como se muestra en la Figura 6.5.

Esta transformación puede ser utilizada sin problemas únicamente en el mapeo de sistemas del tipo pasabajos y pasabanda, ya que no cumple perfectamente con las dos condiciones de mapeo mencionadas. La Figura 6.6 muestra que la condición de que el eje imaginario del plano s se mapee en el círculo unitario, se aproxima aceptablemente hasta $\theta < \pi/6$ radianes.

Supongamos que se dispone de una secuencia $x[n]$, con un período de muestreo $T = 1/f_m$. Por el teorema del muestreo, se tiene que $f_m > 2f_M$, con f_M la frecuencia máxima presente en $x[n]$. Además, como $\theta = 2\pi f$, y teniendo en cuenta la restricción $\theta < \pi/6$ rad., se tendrá que $f_M < f_m/12$ para así cumplir con la condición que el eje imaginario del plano s se mapee aproximadamente sobre la circunferencia unidad en el plano z .

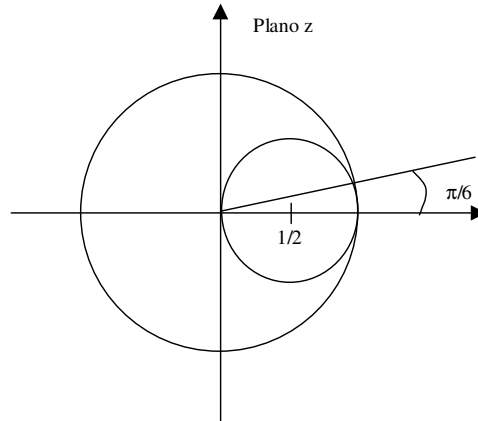


Figura 6.6. Restricción en θ para que el plano s se mapee en el círculo unitario.

6.4.2. Transformación bilineal

La transformación bilineal transforma el eje $j\Omega$ en la circunferencia unidad en el plano z sólo una vez, evitando el solapamiento de componentes de frecuencia. Además, todos los puntos del semiplano izquierdo de s se corresponden con el interior de la circunferencia unidad en el plano z y todos los puntos del semi plano derecho de s se corresponden con puntos fuera de la circunferencia unidad del plano z .

La transformación bilineal se puede ligar a la forma trapezoidal para integración numérica. Por ejemplo consideremos la función:

$$H(s) = \frac{b}{a + s}$$

Este sistema también está caracterizado por la ecuación diferencial

$$\frac{dy(t)}{dt} + a.y(t) = b.x(t)$$

que evaluada en $t = nT$ nos da:

$$y'(nT) = -a.y(nT) + b.x(nT)$$

En lugar de sustituir una diferencia finita por la derivada, supongamos que integramos la derivada y aproximamos la integral mediante la fórmula trapezoidal. Así

$$y(t) = \int_{t_0}^t y'(\tau) d\tau + y(t_0)$$

Para $t = nT$ y $t_0 = nT - T$ produce:

$$y(nT) = \frac{T}{2} (y'(nT) + y'(nT - T)) + y(nT - T)$$

sustituyendo la derivada

$$\begin{aligned} y(nT) = & \frac{T}{2} ((-ay(nT) + bx(nT) + \\ & + (-ay(nT - T) + bx(nT - T))) + y(nT - T) \end{aligned}$$

haciendo $n = nT$, y agrupando términos:

$$\left(1 + \frac{aT}{2}\right) y[n] - \left(1 - \frac{aT}{2}\right) y[n-1] = \frac{bT}{2} (x[n] + x[n-1])$$

la Transformada Z de esta ecuación en diferencias es:

$$\left(1 + \frac{aT}{2}\right) Y(z) - \left(1 - \frac{aT}{2}\right) z^{-1}Y(z) = \frac{bT}{2} [1 + z^{-1}] X(z)$$

y la función de transferencia queda:

$$H(z) = \frac{b}{\frac{2}{T} \left[\frac{1-z^{-1}}{1+z^{-1}}\right] + a}.$$

Claramente, la correspondencia del plano s al plano z es:

$$s = \frac{2}{T} \left[\frac{1-z^{-1}}{1+z^{-1}}\right]$$

y es la ecuación que define la transformación bilineal.

Aunque la anterior derivación fue para una ecuación diferencial de primer orden, se mantiene en general para una ecuación diferencial de orden N .

De la misma manera que para la transformación de Euler, se pueden analizar las características de la transformación bilineal. Sean

$$\begin{aligned} z &= re^{jw} \\ s &= \sigma + j\Omega \end{aligned}$$

entonces

$$\begin{aligned} s &= \frac{2}{T} \left[\frac{1-z}{z+1} \right] \\ s &= \frac{2}{T} \left[\frac{re^{jw} - 1}{re^{jw} + 1} \right] \\ s &= \frac{2}{T} \left[\frac{r^2 - 1}{1 + r^2 + 2r \cos(w)} + j \frac{2r \sin(w)}{1 + r^2 + 2r \cos(w)} \right] \\ \sigma &= \frac{2}{T} \frac{r^2 - 1}{1 + r^2 + 2r \cos(w)} \\ \Omega &= \frac{2}{T} \frac{2r \sin(w)}{1 + r^2 + 2r \cos(w)} \end{aligned}$$

Primero, hay que notar que si $r < 1$, entonces $\sigma < 0$, y si $r > 1$, entonces $\sigma > 0$. De aquí que el semiplano izquierdo en s se corresponda con el interior de la circunferencia unidad en el plano z y el semiplano derecho de s se corresponde con el exterior de la circunferencia unidad. Cuando $r = 1$, entonces $\sigma = 0$ y

$$\begin{aligned} \Omega &= \frac{2}{T} \frac{2 \sin(w)}{1 + 2 \cos(w)} \\ \Omega &= \frac{2}{T} \tan\left(\frac{w}{2}\right) \end{aligned}$$

De aquí que el rango completo en Ω se corresponde unívocamente con el rango $-\pi \leq w \leq \pi$. Sin embargo, la correspondencia es altamente no lineal (Ver Figura 6.7).

6.5. Trabajos prácticos

Ejercicio 1: Determine la función de transferencia de los siguientes sistemas LTI causales :

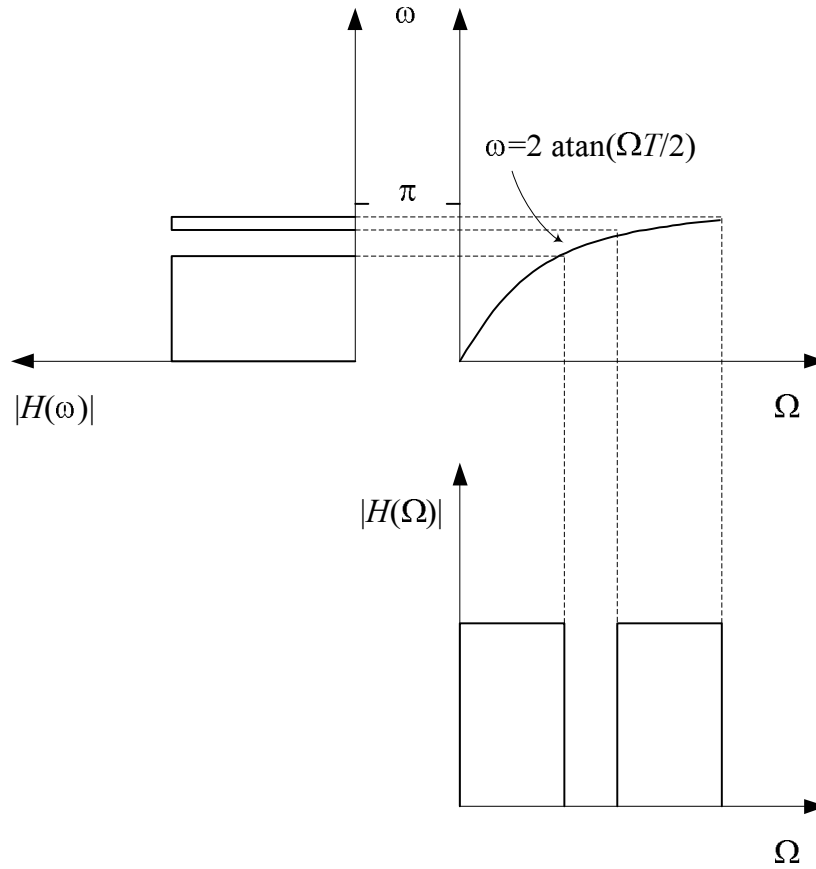


Figura 6.7. Mapeo de w en Ω por medio de la transformación bilineal para la respuesta en frecuencia de un filtro rechaza-banda

1. $y[n] - \frac{1}{2}y[n-1] + \frac{1}{4}y[n-2] = x[n]$
2. $y[n] = y[n-1] + y[n-2] + x[n-1]$
3. $y[n] = 7x[n] + 2y[n-1] - 6y[n-2]$
4. $y[n] = \sum_{k=0}^7 2^{-k}x[n-k]$

Ejercicio 2: Encuentre la respuesta en frecuencia de los sistemas anteriores suponiendo una frecuencia de muestreo de 10KHz.

Ejercicio 3: Considere el sistema $H(z) = \frac{1-2z^{-1}+2z^{-2}-z^{-3}}{(1-z^{-1})(1-0,5z^{-1})(1-0,2z^{-1})}$

1. Dibuje el diagrama de polos y ceros. ¿Es estable el sistema?
2. Determine la respuesta al impulso del sistema.

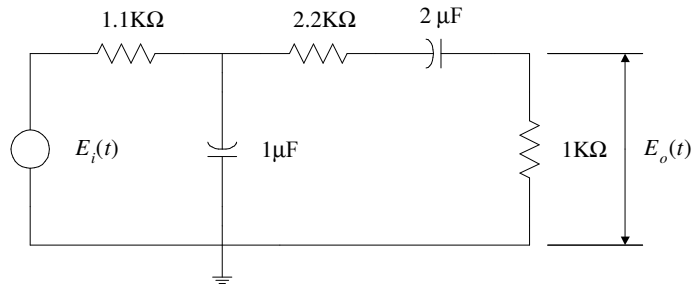


Figura 6.8. Circuito RC.

Ejercicio 4: (*) El circuito de la Figura 6.8 representa el modelo físico de un sistema en observación.

1. Encuentre la ecuación de predicción de la tensión en la resistencia $E_o[n]$ en función de la tensión de entrada $E_i[n]$ utilizando las transformaciones de Euler y bilineal.
2. Utilice distintas señales que representen la entrada $E_i[n]$ y obtenga la salida $E_o[n]$ para cada una de ellas.
3. Analice la respuesta en frecuencia de los sistemas discretos obtenidos con las dos transformaciones conformes y compárelas con la del sistema continuo. Utilice para ambas transformaciones conformes una frecuencia de muestreo cuatro veces superior a la frecuencia de corte del sistema continuo.

Bibliografía

- [1] T. Aljama, M. Cadena, S. Charleston, and O. Yáñez. *Procesamiento digital de señales*. Universidad Autónoma Metropolitana. Unidad Iztapalapa, México, 1992.
- [2] L.E. Franks. *Teoría de la señal*. Reverté, Barcelona, 1975.
- [3] R.A. Gabel and R.A. Roberts. *Señales y sistemas lineales*. Ed. Limusa S.A., México, 1975.
- [4] H. Kwakernaak, R. Sivan, and R.C.W. Strijbos. *Modern Signals and Systems*. Prentice Hall, New Jersey, 1991.
- [5] A.V. Oppenheim and R. Shaffer. *Digital Signal Processing*. Pearson Higher Education, 1986.
- [6] A.V. Oppenheim, A.S. Willsky, S. H. Nawab, and G. Mata Hernández. *Señales y sistemas*. Prentice-Hall Hispanoamericana, México, 2a. (español) edition, 1998.
- [7] A. Papoulis. *Sistemas y Circuitos Digitales y Analógicos*. Marcombo, 1978.
- [8] J.G. Proakis and D.G. Manolakis. *Tratamiento digital de señales*. Prentice Hall, 1a. (español) edition, 1998.
- [9] N.K. Sinha. *Linear systems*. John Wiley, New York, 1991.
- [10] H. Skilling. *Circuitos en Ingeniería Eléctrica*. Cia. Ed. Continental, México, 1987.

Capítulo 7

Identificación de sistemas mediante predicción lineal

Diego Milone

Temas a tratar

- Identificación de sistemas por métodos estáticos y adaptativos.
- Sistema de Wiener-Hopf y algoritmo de Levinson-Durbin.
- Determinación del número óptimo de parámetros.

Objetivos

- Obtener una visión general sobre los problemas relacionados con la identificación de sistemas. Optimización, filtrado adaptativo y otros.
- Implementar algunas técnicas sencillas para identificación de sistemas lineales.
- Reconocer las ventajas, desventajas y posibilidades de aplicación de los distintos métodos.

7.1. Introducción

Muchas veces nos encontramos frente a un sistema del que podemos obtener datos en forma de señales analógicas como movimientos, ondas de presión, ondas eléctricas o magnéticas, concentraciones, etc. En realidad desconocemos los mecanismos que dentro del sistema generan estas señales. Podemos ver al sistema como una *caja negra*, cuyas características intrínsecas desconocemos y queremos averiguar.

Identificar un sistema consiste en encontrar un conjunto de *reglas* y *parámetros* asociados que describan un modelo aceptable para el proceso que en él se está llevando a cabo.

Los datos obtenibles del sistema son su señal de salida y, en algunos casos, su señal de entrada. Por ejemplo, en el aparato fonador sabemos que las ondas de presión que entran a la tráquea tienen forma de pulsos similares a deltas de Dirac en el caso de las vocales. La salida del sistema son las ondas de presión del sonido que sale por la boca. Estas ondas están caracterizadas por ser periódicas y con ancho de banda limitado.

Los resultados de la identificación son la estructura y el conjunto de parámetros característicos del modelo. En algunos casos la estructura del sistema puede ser conocida o supuesta *a priori* y la identificación se reduce a la búsqueda de los parámetros. En el caso más general, tanto la estructura como los parámetros deben ser encontrados minimizando algún criterio de error. Para el ejemplo del aparato fonador podemos suponer que el sistema es lineal y que la salida del sistema sólo depende de la entrada actual y de un número p de valores anteriores de la salida. De esta forma hemos fijado la estructura del sistema (sistema tipo AR de orden p). Sólo resta encontrar los coeficientes de la ecuación que describe este tipo de sistemas.

La identificación de sistemas se basa fundamentalmente en la teoría de sistemas, técnicas de procesamiento de señales y algoritmos de búsqueda y optimización. Estas técnicas son aplicadas en una gran variedad de campos como en el procesamiento de señales biomédicas, sismografía, comunicaciones, inteligencia artificial, etc. Estos procedimientos son atractivos, por ejemplo, para la comprensión de datos. En vez de manipular todos los valores del vector de muestras, se usa solamente un reducido número de parámetros con lo que se simplifica el almacenamiento o transmisión de la señal. La compresión de la señal mediante un modelo paramétrico es también atractiva desde el punto de vista de la clasificación. Hay disponibles algoritmos efectivos para la clasificación automática de señales de electrocardiografía, que representan varios estados patológicos y son muy útiles para el diagnóstico.

También existen algoritmos para el reconocimiento de patrones de voz y características del aparato fonador.

Podemos hacer una clasificación de las técnicas para la identificación de sistemas para ayudarnos en el aprendizaje. Esta clasificación separa dos grandes grupos: las técnicas convencionales y las no convencionales. Veamos las características generales de cada uno.

7.1.1. Técnicas convencionales

Llamaremos técnicas convencionales a las que se basan en la teoría de sistemas *lineales* y señales *estacionarias* o cuasi estacionarias. En el caso de señales cuasi estacionarias podemos dividir la señal en trozos localmente estacionarios y aplicar las técnicas a cada uno por separado. Mediante la extracción de información de un conjunto de señales estacionarias obtendremos un modelo del sistema. En este caso el proceso de modelado se realiza en base a un modelo paramétrico, lineal y causal, descrito por una ecuación generalmente en el dominio de la frecuencia compleja (s o z). Esta ecuación es la denominada función de transferencia del sistema y el modelo más general es el de tipo ARMA.

Dadas estas hipótesis, hemos fijado una parte importante de la estructura del sistema y la identificación consistirá entonces en encontrar el *orden* y los *parámetros* de la función de transferencia.

Existen muchas formas de encarar la identificación bajo éstas hipótesis. Analizaremos en esta revisión teórica tres métodos para la determinación de parámetros: el análisis de la respuesta; el método de predicción lineal y el método adaptativo de Widrow. Estudiaremos también dos métodos para la estimación de orden: predicción del error final (FPE) y criterio de Akaike.

7.1.2. Técnicas no convencionales

Estas técnicas no se encuentran enmarcadas en la teoría de sistemas lineales y así eliminan una de las más importantes restricciones de aplicabilidad que poseen las anteriores. Su aplicabilidad se ve restringida sólo por los costos computacionales de sus implementaciones.

En general estas técnicas pertenecen al campo de la *búsqueda y optimización de soluciones*. Como la identificación de sistemas es realmente un problema de búsqueda y optimización de soluciones, podemos aplicarlas con mucho éxito y pocas restricciones sobre las características de los sistemas y señales intervinientes.

Entre las técnicas no convencionales podemos citar a los algoritmos genéticos, las redes neuronales, la programación dinámica, las búsquedas aleatorias y muchas otras.

7.2. Análisis de la respuesta para sistemas continuos

Como introducción a la identificación de sistemas revisaremos algunos métodos utilizados en sistemas continuos, principalmente en el ámbito del control de sistemas.

Podemos distinguir entre métodos de análisis de la respuesta transitoria del sistema y métodos de análisis de la respuesta en frecuencia. Todos se basan en la hipótesis de que trabajamos con sistemas continuos, lineales e invariantes en el tiempo, y que son conocidos y elementales los estímulos con que se los prueba.

Los métodos de análisis de la respuesta transitoria consisten en estimular al sistema con alguna forma de onda conocida y estudiar su respuesta temporal. Las funciones de estimulación más empleadas son el *delta de Dirac* y el *escalón*. Debemos conocer de antemano la velocidad de respuesta del sistema para determinar qué tan agudo debe ser el impulso de excitación o la pendiente del escalón en la experimentación práctica.

Mediante la Transformada de Laplace podemos escribir:

$$Y(s) = H(s)X(s)$$

siendo Y la respuesta, X la excitación o entrada y H la transferencia.

Sabemos que la Transformada de Laplace de un delta de Dirac es 1. Si estimulamos al sistema con un delta de Dirac tendremos $X(s) = 1$ y así:

$$Y(s) = H(s)$$

donde haciendo la transformada inversa de la salida obtenemos por simple inspección la transferencia del sistema.

Para sistemas de primer o segundo orden es simple obtener sus parámetros estimulándolos con un escalón y analizando gráficamente ciertos patrones identificables en su respuesta. Consideremos las respuestas al escalón de la Figura 7.1.

En el caso de un sistema de primer orden sabemos que su constante de tiempo T es igual al tiempo que tarda la salida del sistema en alcanzar el

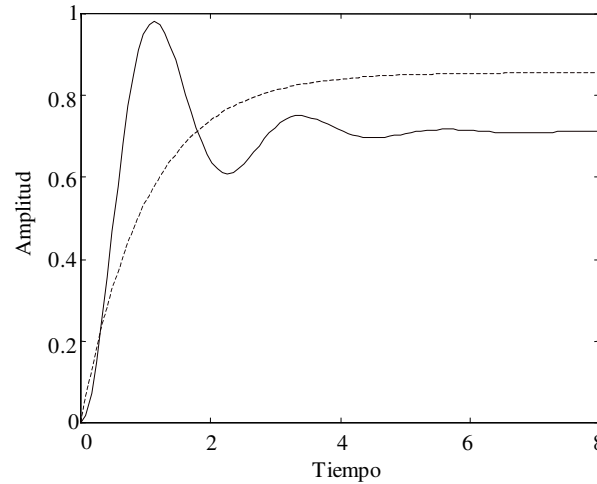


Figura 7.1. Respuesta al escalón de dos sistemas lineales. En línea de trazos para un sistema de primer orden y en línea continua para un sistema de segundo orden.

63 % de su valor final. De manera similar, para el sistema de segundo orden podemos encontrar la frecuencia natural no amortiguada ω_n y la relación de amortiguamiento ζ . En ambos casos resta determinar la constante de amplificación a partir de la relación entre el valor final que alcanza la salida del sistema y la amplitud del escalón. Así, los sistemas han sido identificados.

En caso de no conocer el orden del sistema podemos utilizar métodos más completos para determinarlo junto a los parámetros característicos. Entre éstos encontramos los métodos de Strejc y Sten.

El análisis gráfico de la *respuesta en frecuencia* del sistema también permite estimar fácilmente los parámetros característicos de sistemas lineales de orden 3 ó menor. Consiste en estimular al sistema con señales senoidales de frecuencias variables en el rango de interés y registrar la atenuación que éste produce en cada caso. De esta manera se puede aproximar una gráfica de respuesta en frecuencia para el sistema y de su análisis obtener los parámetros característicos.

De todas formas, aunque existe una batería de métodos de análisis de la respuesta para sistema continuos, resulta muy dificultosa la aplicación de los métodos antes descriptos a sistemas de orden 20, 50 o mayor.

En las siguientes secciones detallaremos la identificación de sistemas discretos de orden superior.

7.3. Métodos de predicción lineal

La denominación *predicción lineal* proviene del modelo que se utiliza en la aproximación del sistema real a identificar. Decimos que la salida es *predecible* a partir de una combinación *lineal* de las entradas y salidas pasadas.

7.3.1. El modelo ARMA

Consideremos la salida s_n de un sistema puede ser determinada a partir de su entrada u_n mediante la ecuación de recurrencia:

$$s_n = - \sum_{k=1}^p a_k s_{n-k} + G \sum_{\ell=0}^q b_\ell u_{n-\ell} \quad (7.1)$$

donde $1 \leq k \leq p$, $1 \leq \ell \leq q$ con $b_0 = 1$ y G una constante real. Mediante la utilización de la Transformada Z podemos describir el sistema de la ecuación (7.1) en el dominio de la frecuencia compleja:

$$H(z) = G \frac{1 + \sum_{\ell=1}^q b_\ell z^{-\ell}}{1 + \sum_{k=1}^p a_k z^{-k}} \quad (7.2)$$

7.3.2. El modelo AR

Este modelo de sistemas digitales es muy usado debido a su simplicidad. Debemos notar que cualquier sistema ARMA puede ser aproximado mediante un modelo AR. Si el sistema ARMA esta dado por la función de transferencia (7.2) entonces mediante división larga podemos obtener la siguiente igualdad:

$$H(z) = G \frac{B(z)}{A(z)} \approx \frac{G}{C(z)}$$

Sin embargo, el polinomio $C(z)$ será de orden infinito. Podemos aproximar con bastante exactitud el sistema ARMA (7.2) haciendo que $C(z)$ sea

del mismo orden que el polinomio $A(z)$. Utilizamos la siguiente expresión para la transferencia del modelo AR:

$$H(z) = \frac{G}{1 + \sum_{k=1}^p a_k z^{-k}} \quad (7.3)$$

que podemos expresar mediante ecuaciones de recurrencia como:

$$s_n = - \sum_{k=1}^p a_k s_{n-k} + G u_n \quad (7.4)$$

Supondremos por ahora que conocemos el orden p óptimo para nuestra aproximación y utilizaremos la siguiente expresión:

$$\hat{s}_n = - \sum_{k=1}^p a_k s_{n-k} \quad (7.5)$$

donde estamos eliminando intencionadamente la entrada del sistema (el circunflejo $\hat{\cdot}$ indica que se trata de una variable estimada). Este modelo es el más utilizado debido a que en la mayoría de las aplicaciones la entrada del sistema es totalmente desconocida. De esta manera, decimos que la salida puede ser predecida solamente como una combinación lineal de las salidas anteriores. Más adelante resolveremos tanto el problema del orden p como el de la ganancia G . Ahora nos encargaremos de los coeficientes del modelo.

7.3.3. Cuadrados mínimos

Supongamos ahora que utilizamos el sistema ARMA de las ecuaciones (7.1) y (7.2) para modelar a otro sistema desconocido con salida s_n y entrada u_n . El error cometido en la aproximación del valor de la n -ésima muestra de la salida será:

$$e_n = s_n - \hat{s}_n$$

Una mejor estimación del error es utilizar la medida del error cuadrático:

$$e_n^2 = (s_n - \hat{s}_n)^2 = (s_n + \mathbf{s}_n^T \mathbf{a} - G \mathbf{u}_n^T \mathbf{b})^2 \quad (7.6)$$

donde:

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \\ \vdots \\ a_p \end{bmatrix} \quad \mathbf{s}_n = \begin{bmatrix} s_{n-1} \\ s_{n-2} \\ \vdots \\ s_{n-k} \\ \vdots \\ s_{n-p} \end{bmatrix}$$

$$\mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_\ell \\ \vdots \\ b_q \end{bmatrix} \quad \mathbf{u}_n = \begin{bmatrix} u_n \\ u_{n-1} \\ \vdots \\ u_{n-\ell} \\ \vdots \\ u_{n-q+1} \end{bmatrix}$$

con $b_0 = 1$ y $1 \leq k \leq p$ y $0 \leq \ell \leq q$.

El *objetivo* del método es minimizar el error cuadrático a partir de la optimización de los vectores \mathbf{a} y \mathbf{b} . Para un problema en particular tanto la entrada como la salida del sistema estarán dadas y tendremos que considerar que debemos minimizar la función:

$$e_n^2(\mathbf{a}, \mathbf{b}).$$

Para cada instante de tiempo n nos encontramos buscando el mínimo en una superficie de error en el hiperespacio de soluciones con dimensión $p + q$. Otra forma de resolver el problema de minimización del error es considerar el error cuadrático total (ECT) sobre el proceso y optimizar los vectores \mathbf{a} y \mathbf{b} para todo el proceso. En cualquier caso el mínimo constituye la solución de nuestro problema.

La particular utilización de esta medida del error da origen a un amplio grupo de técnicas para encontrar el mínimo global de esta función cuadrática. De aquí que se ha denominado a éste como el *método de cuadrados mínimos*.

Supongamos que $\varepsilon^2(\mathbf{a}, \mathbf{b})$ es una expresión válida para la medida del error. Deberemos encontrar:

$$\nabla_{\mathbf{a}, \mathbf{b}} \varepsilon^2 = 0 \tag{7.7}$$

Esta condición puede cumplirse en cualquier mínimo local de la superficie de error. Por lo tanto agregaremos que los vectores óptimos \mathbf{a}^* y \mathbf{b}^* son los que dan el valor menor para $\varepsilon^2(\mathbf{a}^*, \mathbf{b}^*)$ entre todos los que satisfacen (7.7). Los vectores \mathbf{a}^* y \mathbf{b}^* son el *mínimo global* de la superficie de error.

La ecuación (7.7) aplicada a un modelo en particular y con las hipótesis necesarias sobre las señales que se tratan, nos lleva a sistemas de ecuaciones lineales simultáneas de cuya resolución se obtiene el conjunto de parámetros para el modelo lineal. Este sistema de ecuaciones es conocido como *sistema de Wiener-Hopf* y a continuación vamos a ver como llegar hasta él y algunas de las formas posibles de encontrar sus soluciones

7.3.4. Sistema de Wiener-Hopf para señales determinísticas

En el caso de las señales *determinísticas* podemos trabajar con la siguiente definición para el error cuadrático total (ECT):

$$\xi^2 = \sum_n e_n^2 \quad (7.8)$$

Supongamos que se utiliza un modelo AR. (Existe una considerable simplificación de las expresiones para modelos AR. Dado el alcance previsto para esta guía de estudio en adelante utilizaremos sólo este tipo de modelos). De la combinación de las ecuaciones (7.6), (7.7) y (7.8) obtenemos:

$$\left(\sum_n \mathbf{s}_n \mathbf{s}_n^T \right) \mathbf{a} = - \sum_n \mathbf{s}_n s_n \quad (7.9)$$

Debemos aún especificar el rango de sumación para la variable n . Esto da origen a dos métodos: autocorrelación y covariancia.

Autocorrelación

Una visión general del proceso nos llevaría a considerar $(-\infty, \infty)$ como intervalo para la sumación de n en (7.9). En este caso tenemos en el miembro derecho la *autocorrelación* \mathbf{r} para la señal s_n . Ahora la ecuación (7.9) se convierte en:

$$\mathbf{R}\mathbf{a} = -\mathbf{r} \quad (7.10)$$

donde la matriz $R_{i,j} = r_{i-j}$, con $1 \leq i, j \leq p$, es la matriz de autocorrelación para s_n . La matriz \mathbf{R} es simétrica y posee todas sus diagonales con elementos iguales. Estas matrices se denominan *matrices de Toeplitz*.

Este sistema de ecuaciones al que hemos arribado se denomina *sistema de Wiener-Hopf* y su solución constituye la solución de nuestro problema de identificación.

En las aplicaciones prácticas nunca podemos realizar la autocorrelación en el intervalo $(-\infty, \infty)$. Por esta razón deberemos limitar mediante ventanas el soporte de la señal. El ancho de las ventanas es muy dependiente de la aplicación. En el caso de señales de voz se pueden utilizar ventanas de 10 a 30 ms. Las ventanas deben tener variaciones suaves para no malcondicionar el sistema de ecuaciones. Se recomienda la utilización de ventanas como las de Hamming, Hanning o Blackman.

Covariancia

Como veíamos en el caso anterior, no es posible prácticamente utilizar un rango infinito para la sumatoria en el ECT. Podemos recurrir a la covariancia para acotar el intervalo a un número finito de muestras N . En este caso la ecuación (7.9) se convierte en:

$$\Phi\mathbf{a} = -\varphi$$

donde $\varphi_i = \Phi_{0,i}$, con $0 \leq i \leq N-1$ y Φ es la matriz de covariancia para s_n en el intervalo finito $0 \leq n \leq N-1$. En este caso también obtenemos una matriz simétrica pero los elementos en la diagonal ya no son necesariamente iguales. A pesar de que la matriz configura un sistema bien condicionado el error que se obtiene para distintas selecciones de N es dependiente de la señal de entrada. Nuevamente, en el ejemplo del aparato fonador, lo ideal sería que N coincidiera con el período de ocurrencia para el impulso glótico. Sin embargo, incluir un N variable no sólo implicaría un cálculo constante de la frecuencia glótica sino una adaptación en el algoritmo de resolución del sistema de ecuaciones.

7.3.5. Sistema de Wiener-Hopf para señales aleatorias

En el caso de señales *aleatorias* sólo podemos hablar de valores esperados para las medidas de error. De esta forma escribimos para un sistema AR:

$$\xi^2 = \mathcal{E} [e_n^2] = \mathcal{E} [s_n^2] + \mathbf{a}^T 2\mathcal{E} [s_n \mathbf{s}_n] + \mathbf{a}^T \mathcal{E} [\mathbf{s}_n \mathbf{s}_n^T] \mathbf{a} \quad (7.11)$$

De igual forma que en la ecuación (7.7) estamos buscando:

$$\nabla \xi^2 = 0$$

en el mínimo global en la superficie de error. Este gradiente es:

$$\nabla \xi^2 = \left\{ \begin{array}{c} \frac{\partial \mathcal{E} [e_n^2]}{\partial a_1} \\ \frac{\partial \mathcal{E} [e_n^2]}{\partial a_2} \\ \vdots \\ \frac{\partial \mathcal{E} [e_n^2]}{\partial a_p} \end{array} \right\} = 2\mathcal{E} [s_n \mathbf{s}_n] + 2\mathcal{E} [\mathbf{s}_n \mathbf{s}_n^T] \mathbf{a}$$

Así, obtenemos nuevamente el sistema de ecuaciones de *Wiener-Hopf*:

$$\mathcal{E} [\mathbf{s}_n \mathbf{s}_n^T] \mathbf{a} = -\mathcal{E} [s_n \mathbf{s}_n] \quad (7.12)$$

Sin embargo aún no hemos definido qué sucede con el operador esperanza que debemos resolver en función de las características de la señal aleatoria. Vamos a diferenciar entre señales estacionarias y señales no estacionarias.

Señales estacionarias

Para señales estacionarias se cumple que:

$$\mathcal{E} [\mathbf{s}_n \mathbf{s}_n^T] = \mathbf{R}$$

De esta forma el sistema de la ecuación (7.12) se reduce al de la ecuación (7.10).

Para el caso en que nos encontremos ante un proceso de esta naturaleza, podemos observar que la ecuación (7.11) se convierte en:

$$\xi^2 = \mathcal{E} [s_n^2] + 2\mathbf{r}\mathbf{a} + \mathbf{a}^T \mathbf{R} \mathbf{a} \quad (7.13)$$

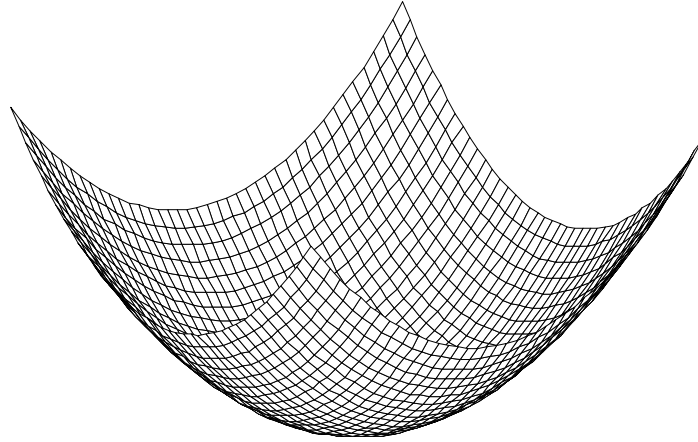


Figura 7.2. Superficie de error cuadrático para un sistema AR de orden 2.

Si suponemos que estamos ante un sistema de orden 2, donde las variables independientes del modelo AR son a_1 y a_2 , podemos ver al problema de identificación como la búsqueda del mínimo de la superficie $\xi^2(a_1, a_2)$. En esta ecuación el error cuadrático (7.13) es un polinomio de segundo orden en la variable \mathbf{a} . Por lo tanto la superficie de búsqueda se verá como la siguiente Figura 7.2.

Sin embargo, en las aplicaciones prácticas las hipótesis hechas sobre las señales no se cumplen y el problema no es tan sencillo como deslizarse sobre una superficie tan suave como ésta y con un solo mínimo, que por lo tanto es global.

Señales no estacionarias

En el caso de procesos no estacionarios debemos contar con la autocorrelación no estacionaria para la señal. Sin embargo, existen algunos procesos no estacionarios conocidos como procesos *localmente estacionarios* que pueden ser segmentados en procesos estacionarios. Ejemplos de este tipo de procesos son las señales de voz y electrocardiografía.

7.3.6. Resolución del sistema de Wiener-Hopf

Encontrar los coeficientes del modelo AR para la identificación de nuestro sistema incógnita requiere que resolvamos el sistema de ecuaciones

simultáneas de Wiener-Hopf en cualquiera de sus versiones. Para esto vamos a describir dos grandes grupos de métodos: los métodos estáticos y métodos adaptativos. En el primer caso la función objetivo es una estimación del ECT y en el segundo se optimiza instante a instante según el error cuadrático instantáneo (ECI) o una estimación de éste.

Métodos estáticos

Los métodos estáticos atacan directamente la resolución de un sistema de ecuaciones lineales mediante métodos numéricos.

En primer lugar debemos tener en cuenta los métodos tradicionales de resolución de sistemas de ecuaciones lineales. Entre ellos podemos mencionar: inversión de matrices, método de eliminación de Gauss, descomposición LU, pivoteo, métodos de Jacobi y Gauss-Seidel, etc.

Sin embargo, en algunos casos es posible aprovechar mejor las propiedades de la matriz \mathbf{R} . Dijimos que en el caso de señales determinísticas con el método de autocorrelación y para señales aleatorias estacionarias la matriz \mathbf{R} era una matriz de Toeplitz. Para estos casos existe un método iterativo que soluciona eficientemente el problema. Éste es el método de *Levinson-Durbin*, que a partir de $E_0 = r_0$ sigue la recurrencia:

$$\text{Para } 1 \leq i \leq p \rightarrow \begin{cases} k_i = -\frac{1}{E_{i-1}} \left[r_i + \sum_{j=1}^{i-1} a_{j,i-1} r_{i-j} \right] \\ a_{i,i} = k_i \\ a_{j,i} = a_{j,i-1} + k_i a_{i-j,i-1}, \quad 1 \leq j \leq i-1 \\ E_i = E_{i-1} (1 - k_i^2) \end{cases}$$

donde E_p es el ECT para la estimación de orden p y la solución final son los $a_{j,p}$ con $1 \leq j \leq p$.

Este método posee la ventaja de que arroja resultados parciales que, como veremos más adelante, pueden ser utilizados para estimar el orden apropiado del modelo mientras se lleva a cabo la iteración.

Otra característica particular es que cuando el método es utilizado para calcular el modelo AR del ejemplo del aparato fonador, se encuentra una vinculación directa entre las constantes k_i y la relación de diámetros en el tubo resonante. Se puede demostrar que si modelamos el tracto vocal mediante un tubo con p secciones S_i diferentes, se cumple la relación:

$$\frac{k_i + 1}{1 - k_i} = \frac{S_i}{S_{i+1}}$$

Métodos adaptativos

En los *métodos adaptativos* se busca minimizar el error cuadrático instantáneo mediante una adaptación permanente de los vectores que en este caso podemos denominar \mathbf{a}_n y \mathbf{b}_n .

Estos métodos se basan en la fórmula de Newton para la búsqueda de raíces en ecuaciones no lineales. Básicamente, consisten en realizar sucesivas aproximaciones a la raíz en el sentido del gradiente negativo de la función. Para nuestro caso, la función es el error cuadrático o la esperanza de éste para un tiempo n dado. Volviendo a la ecuación de error para sistemas AR, se propone la adaptación de los coeficientes mediante:

$$\mathbf{a}_{n+1} = \mathbf{a}_n + \mu(-\nabla \xi^2) \quad (7.14)$$

donde μ es una constante positiva que más adelante acotaremos. Este denominado *método de Newton* da origen a varios otros métodos que utilizan una diferente estimación para el gradiente del error cuadrático esperado. Veremos de entre ellos al método de Widrow.

Método adaptativo de Widrow: en este método se utiliza el error cuadrático instantáneo como una aproximación válida para el error cuadrático esperado. Por lo tanto tendremos:

$$\hat{\nabla} \xi_n^2 = \frac{\partial e_n^2}{\partial \mathbf{a}_n} = \left\{ \begin{array}{c} \frac{\partial e_n^2}{\partial a_{1,n}} \\ \frac{\partial e_n^2}{\partial a_{2,n}} \\ \vdots \\ \frac{\partial e_n^2}{\partial a_{p,n}} \end{array} \right\} = 2e_n \frac{\partial (s_n + \mathbf{a}_n^T \mathbf{s}_n)}{\partial \mathbf{a}_n} \quad (7.15)$$

Ahora la ecuación para la adaptación de los coeficientes la obtenemos reemplazando en (7.14) la aproximación de Widrow (7.15):

$$\mathbf{a}_{n+1} = \mathbf{a}_n - 2\mu e_n \mathbf{s}_n$$

donde el vector de coeficientes \mathbf{a}_n es inicializado con valores aleatorios en el rango $[-0,5, 0,5]$. El parámetro μ determina la velocidad y estabilidad de la convergencia del método. Cuando menor es μ mayor es el número de iteraciones pero nos asegura la estabilidad. En cambio un μ más grande hace

que el método converja más rápidamente pero pueden existir problemas de inestabilidad y divergencia. Puede demostrarse que en este caso aseguramos la convergencia en la media a la solución óptima del sistema de Wiener-Hopf con:

$$0 < \mu < \frac{1}{T(\mathbf{R})}$$

siendo $T(\mathbf{R})$ la traza de la matriz de autocorrelación \mathbf{R} .

7.3.7. Determinación de la constante de ganancia G

Consideremos nuevamente las ecuaciones (7.4) y (7.5) y supongamos que el error entre la salida real del sistema y nuestra estimación AR está dado por:

$$e_n = s_n - \hat{s}_n = Gu_n$$

y de esta forma tenemos un método sencillo para determinar la ganancia G en la entrada. Lo que debemos asegurar es que la energía total de la señal de entrada sea igual a la energía total de la señal de error.

Hay dos tipos particulares de entradas que nos interesan en nuestro análisis: la entrada impulsiva y la de ruido blanco. Para el caso de la entrada impulsiva:

$$h_n = - \sum_{k=1}^p a_k h_{n-k} + G \delta_n$$

A partir de (7.10) podemos escribir:

$$\mathbf{r}^T \mathbf{a} = -r_0 + G^2$$

de lo que concluimos:

$$G^2 = \mathbf{r}^T \mathbf{a} + r_0 = E_p$$

Puede demostrarse que para una entrada de ruido blanco la expresión es la misma.

7.4. Estimación del orden

En las secciones anteriores hemos asumido como conocida una de las características más importantes de los modelos que utilizamos para la identificación: su orden.

La estimación del orden usualmente se realiza mediante técnicas de optimización. Es decir, se busca el orden óptimo según algún criterio. En este criterio deberán incorporarse todas las variables que puedan influenciar la decisión acerca del orden del sistema. En primer lugar está la correcta representación del proceso que estamos identificando. Es decir, alguna medida del error total ocurrido en la optimización. Sin embargo, en la mayor parte de las aplicaciones el modelo siempre mejora cuando se incrementa el orden. Así, el problema que se plantea es cuándo detener el aumento del orden, ya que es claro que cuanto menor sea el orden que satisfaga los requerimientos de la aplicación en cuanto a error, mejor será el modelo resultante. Con menores órdenes es posible reducir la cantidad de cálculos, volumen de información a almacenar y otros problemas relacionados con la implementación.

Actualmente existen muchos métodos aplicables para elegir el orden de los modelos AR. Sin embargo, en algunas aplicaciones, ninguno de ellos es lo suficientemente preciso por lo que siempre debe hacerse hincapié en el ingenio del modelizador y los conocimientos previos del sistema. Vamos a ver a continuación dos de los métodos más utilizados para la determinación del orden apropiado para nuestro modelo: el *error de predicción final* (FPE) y el *criterio de Akaike*.

7.4.1. Error de predicción final

Este método utiliza la medición de los errores residuales en la resolución de las ecuaciones de Wiener-Hopf. Como vimos, estos errores son una estimación de la secuencia de entrada que asumimos no correlacionada y con espectro plano. Consideremos el modelo AR de orden p y repasemos la forma que toma este error. A partir de la ecuación (7.10) podemos obtener el mínimo error promedio para el sistema de orden p según:

$$E_p = r_0 + \mathbf{r}^T \mathbf{a}$$

A partir de esta expresión definimos el error normalizado como:

$$V_p = \frac{E_p}{r_0}$$

En base a las técnicas de estimación de parámetros que vimos es posible demostrar que V_p es monotónicamente decreciente con p y está acotado. En el caso que la señal esté generada por un sistema AR de orden p_0 también se demuestra que $V_p = V_{p_0}$ cuando $p \geq p_0$. Sin embargo, debido a que

no utilizamos secuencias infinitas en las aplicaciones prácticas, la pendiente de V_p en función del orden no se hace exactamente cero. Consideremos la utilización de un umbral para estimar el orden óptimo:

$$1 - \frac{V_{p+1}}{V_p} < \gamma \quad (7.16)$$

Así detendremos el proceso de incremento del orden cuando se cumpla la inecuación (7.16).

7.4.2. Criterio de Akaike

Akaike criticó el método anterior y propuso un nuevo criterio basado en la teoría de la información. Para el problema de todos polos y asumiendo una distribución gaussiana en la señal medimos el error según:

$$I_p = \log E_p + \frac{2p}{N_e}$$

donde N_e es el número *efectivo* de muestras en la señal. Con *efectivo* nos referimos a los posibles efectos del ventaneo en la señal. El ancho efectivo de una ventana puede ser calculado mediante la relación entre la energía de la ventana utilizada y la energía de una ventana cuadrada. Para el caso de una ventana de Hamming $N_e = 0,4N$, siendo N el ancho real de la ventana en muestras.

Si graficamos V_p e I_p en función del orden p , podemos encontrarnos con una curva como la que se muestra en la Figura 7.3. Se puede observar que, para este sistema de orden $p_0 = 10$ el V_p se mantiene constante a partir de este orden. En contraste vemos que I_p muestra un mínimo en $p = 10$. Este mínimo permite detectar más fácilmente el orden óptimo.

7.5. Preguntas

1. Enumere las simplificaciones más importantes que se realizan para llegar a la ecuación de Wiener-Hopf.
2. Detalle todos los pasos algebraicos necesarios para obtener la ecuación 7.9 a partir de (7.6), (7.7) y (7.8).
3. Muestre que se cumple $R_{i,j} = r_{i-j}$ en (7.10) para un $p = 2$ y $n = 1 \dots 3$.

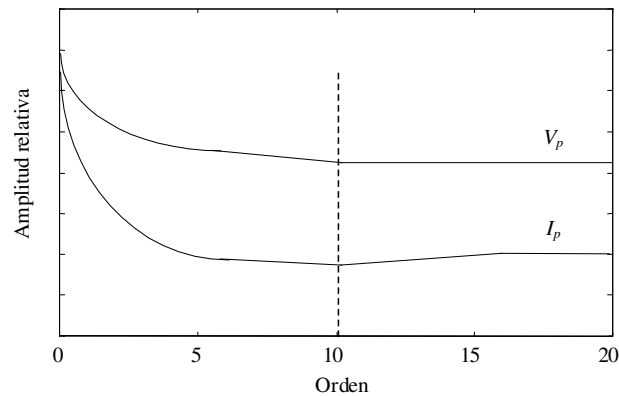


Figura 7.3. Comportamiento típico de los criterios de estimación de orden.

4. ¿Por qué ciertos sistemas requieren la aplicación de métodos adaptativos de identificación?
5. Explique la diferencia conceptual entre la utilización del error cuadrático medio y la del error cuadrático instantáneo.
6. Demuestre que en el caso de señales aleatorias estacionarias existe un único mínimo en la superficie del error cuadrático esperado.
7. ¿Cómo puede aplicarse el método de Levinson-Durbin a señales no estacionarias?
8. Detalle todos los pasos algebraicos necesarios para obtener la ecuación de la ganancia G , para una entrada de ruido blanco y para un tren de pulsos.
9. Explique conceptualmente por qué el método de Akaike generalmente ofrece un mínimo en el orden correcto.

7.6. Trabajos prácticos

Ejercicio 1: Considere el sistema $y[n] = 0,3y[n-1] - 0,4y[n-2] + 0,2y[n-3] + x[n]$ y genere una secuencia de salida a una entrada de tipo aleatoria con distribución uniforme y valor medio cero. Utilizando esta señal de salida implemente el método de

predicción lineal y verifique el comportamiento de los criterios para estimación del orden.

Ejercicio 2: (*) La señal de electroencefalograma se puede modelar mediante un sistema AR de orden cuatro a ocho. Identifique el sistema que generó la señal almacenada en el archivo `eeg.txt` y compare la respuesta en frecuencia de este sistema con el espectro de la señal.

Ejercicio 3: (**) En la Figura 7.4 se puede apreciar un esquema simplificado del aparato fonador humano. La zona comprendida entre la laringe (glotis) y los labios constituye el tracto vocal. Los sonidos del habla son el resultado de la excitación acústica del tracto vocal por la acción de una o más fuentes. En este proceso los órganos fonatorios desarrollan distintos tipos de actividades, tales como: movimiento de pistón que inician una corriente de aire; movimiento o posiciones de válvula que regulan el flujo de aire, y al hacerlo generan sonidos o en algunos casos simplemente modulan las ondas generadas por otros movimientos. El sistema respiratorio constituye la principal fuente de energía para producir sonidos en el aparato fonador humano. La energía es proporcionada en forma de flujo o corriente de aire y presiones que, a partir de las distintas perturbaciones, generan los diferentes sonidos.

El tracto vocal se puede modelar con relativa exactitud mediante un sistema AR. En la Figura 7.5 se aprecia el diagrama del modelo correspondiente. Este modelo permite la construcción de sintetizadores digitales de voz.

Para el presente ejercicio se quiere sintetizar una emisión de voz y se cuenta con las muestras digitalizadas de la palabra inglesa “hello” que se encuentra muestreada a 16 KHz en el archivo `hello.txt`. Se provee también una lista de etiquetas `hello.phn` que permiten la separación de la emisión en fonemas (ver Figura 7.6).

Acerca de la implementación: para encontrar los parámetros de cada fonema se sugiere utilizar el método de autocorrelación con ventaneo. Encuentre la solución al sistema

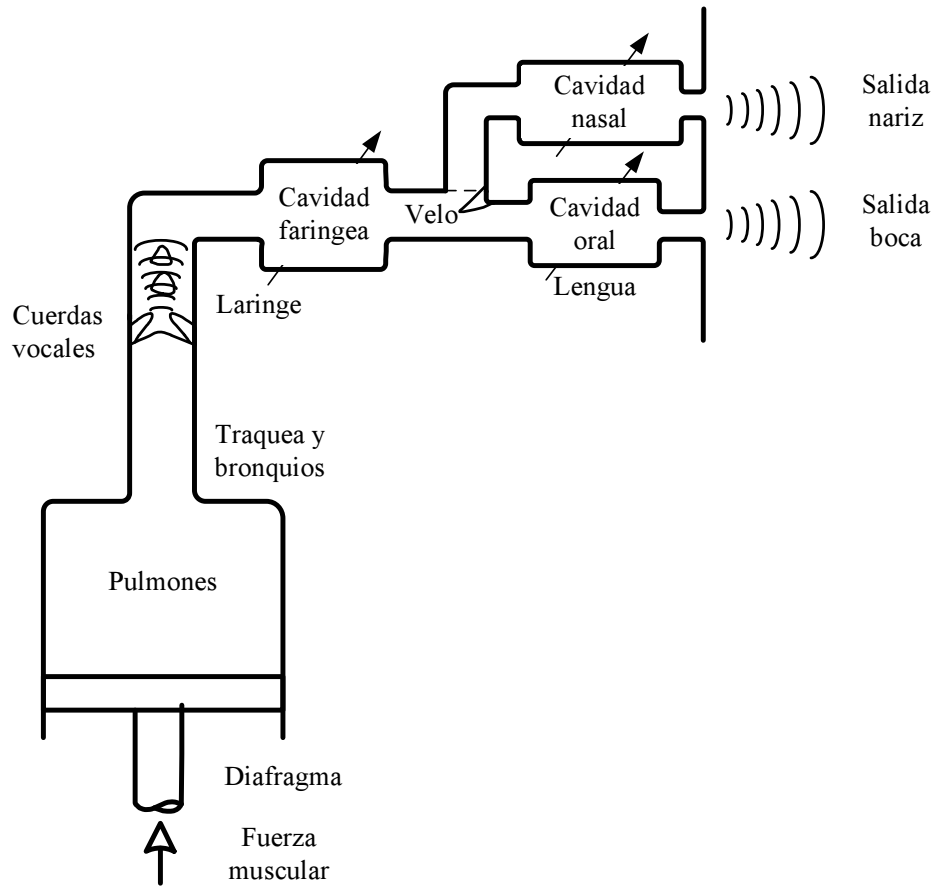


Figura 7.4. Esquema del aparato fonador.

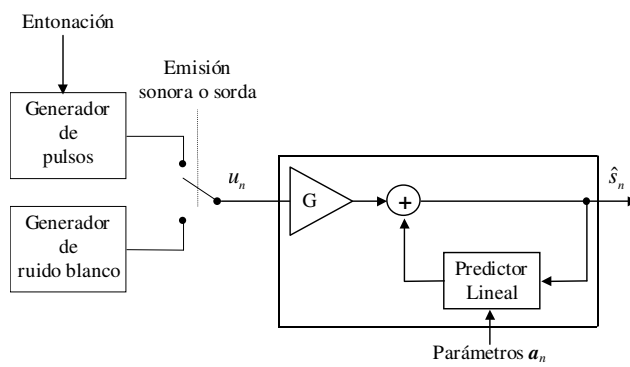


Figura 7.5. Diagrama para el modelo AR del aparato fonador.

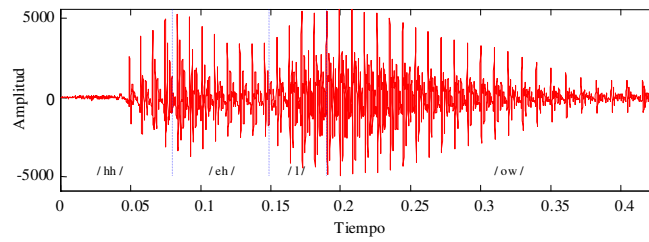


Figura 7.6. Palabra “hello” separada en fonemas.

de Wiener-Hopf mediante el algoritmo iterativo de Levinson-Durbin. Suponga que el sistema es siempre de orden 14, con condiciones iniciales nulas y una frecuencia de entonación constante de 100 Hz. Calcule el valor de G para que se igualen las energías de los fonemas sonoros y en el caso de los sordos considere la energía como 1/10 de la de los sonoros.

Bibliografía

- [1] A. Cohen. Time and frequency domains analysis. In *Biomedical Signal Processing*, volume I, chapter 7. CRC Press, 1986.
- [2] J. R. Deller, J. G. Proakis, and J. H. Hansen. *Discrete-Time Processing of Speech Signals*. Prentice Hall, 1987. Capítulo 5.
- [3] S. M. Kay and S. L. Marple. Spectrum analysis. In *Proc. IEEE*, volume 69, pages 1380–1419, 9 1981.
- [4] J. Makhoul. Linear prediction: A tutorial review. In *Proc. IEEE*, volume 63, pages 561–580, 4 1975.
- [5] H. Morikawa. Adaptative estimation of time-varying model order in the arma speech analysis. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38:1073–1083, 7 1990.
- [6] K. Ogata. *Ingeniería moderna de control*. Pearson Educación México, 2 edition, 1993.
- [7] A. Papoulis. Maximun entropy and spectral estimation: A review. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP 29:1176–1186, 12 1981.
- [8] R. A. Pepe and G. G. Pantaleo. Método de determinación de parámetros de sistemas lineales. *Revista Telegráfica Electrónica*, 12 1983.
- [9] L. R. Rabiner and B. Gold. *Theory and Application of Digital Signal Processing*. Prentice-Hall, 1975. Capítulo 12.
- [10] V. U. Reddy, B. Egardt, and T. Kailath. Least squares algorithm for adaptative implementation of pisarenko's harmonic retrieval method. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP 30:399–405, 6 1982.
- [11] L. F. Rocha. Predicción lineal aplicada a señales de voz. *Revista Telegráfica Electrónica*, 1979.
- [12] M. R. Schroeder. Linear prediction, entropy and signal analysis. *IEEE ASSP Magazine*, 7 1984.

- [13] B. Widrow and A. Lehr. 30 years of adaptative neural networks: Perceptron, madaline, and backpropagation. In *Proceedings of the IEEE*.

Capítulo 8

Identificación de sistemas no lineales mediante algoritmos genéticos

Diego Milone

Temas a tratar

- Generalidades de los algoritmos de computación evolutiva.
- Diseño de la solución de un problema mediante algoritmos genéticos.
- Aplicación de la técnica de algoritmos genéticos para la identificación de sistemas no lineales.

Objetivos

- Aprender los fundamentos de la técnica.
- Aplicar los algoritmos genéticos a la búsqueda de soluciones.
- Comprender la potencialidad de la técnica y sus limitaciones más importantes.
- Implementar algunas técnicas sencillas para identificación de sistemas no lineales.

8.1. Introducción

Los algoritmos genéticos (AGs) junto con las estrategias evolutivas y la programación evolutiva constituyen un nuevo enfoque a la solución de numerosos problemas de optimización y búsqueda. Esta técnica está en el contexto de una nueva ciencia denominada *computación evolutiva*. El campo de las aplicaciones de la computación evolutiva no reconoce fronteras. Es posible resolver problemas de las características más diversas y entre éstos vamos a ver la aplicación muy particular de la identificación de sistemas no lineales mediante algoritmos genéticos. Es importante dejar en claro que los AGs *no* son una técnica de identificación de sistemas sino más bien la identificación de sistemas es una posible *aplicación* de los AGs.

La analogía en que se basan los AGs estriba en reconocer el mecanismo esencial del proceso evolutivo en la naturaleza. Los componentes fundamentales de éste mecanismo son los cromosomas, el material genético de un individuo biológico, que determinan sus características únicas. Los cambios en el material genético de las especies permiten el proceso de adaptación.

Las fuerzas que subyacen al proceso evolutivo son: la selección natural, la recombinación de material genético y la mutación, fenómenos que se presentan durante la reproducción de las especies. La competencia entre los individuos por los recursos naturales limitados y por la posibilidad de procreación o reproducción permite que sólo los más fuertes o más adaptados sobrevivan. Esto significa que el material genético de los mejores individuos sobrevive y se reproduce, mientras que los genes de los individuos más débiles o menos adaptados, mueren o se extinguen. Y de esta forma la naturaleza optimiza la solución al problema de la vida.

Se propone a los AGs como una técnica computacional que intenta *imitar* el proceso evolutivo de la naturaleza, para el diseño de sistemas artificiales adaptativos.

8.2. Estructura de un AG

Los AGs clásicos, manipulan una población de soluciones potenciales codificadas en cadenas binarias que las representan. Este conjunto de cadenas representan el material genético de una población de individuos. Los operadores artificiales de selección, cruce y mutación son aplicados para buscar los mejores individuos –mejores soluciones– a través de la simulación del proceso evolutivo natural. Cada solución potencial se asocia con un valor de *fitness* (suele utilizarse la traducción *aptitud* pero no es totalmente

apropiada en el campo de la teoría darwiniana. Por lo tanto usaremos este término desde su inglés original), el cual mide qué tan buena es comparada con las otras soluciones de la población. Este valor de fitness es la simulación del papel que juega el ambiente en la evolución natural darwiniana.

El paradigma de los algoritmos genéticos puede esquematizarse como sigue:

```
Inicializar(Población)
MejorFitness=Evaluar(Población)
Mientras (MejorFitness<FitnessRequerido)
    Selección=Seleccionar(Población)
    Población=CruzarYMutar(Selección)
    MejorFitness=Evaluar(Población)
FinMientras
```

Para comenzar, se inicializa la población completamente al azar. En la inicialización hay que tener en cuenta que la distribución de valores debe ser uniforme para cada rango representado por los cromosomas. A continuación se decodifica el genotipo en fenotipo de esta población inicial y evalúa el fitness para cada individuo. Es decir, se le asigna un valor numérico a su *capacidad de supervivencia*; en el espacio de soluciones de nuestro problema medimos qué tan bien resuelve el problema cada individuo.

Luego entramos en el bucle de optimización o búsqueda. Este ciclo termina cuando nuestro AG ha encontrado una solución adecuada para el problema. Es decir, deberemos valernos del fitness para el mejor individuo y de un umbral para evaluar esta condición de finalización.

Durante el proceso evolutivo artificial, se aplican varios operadores. Mediante un proceso fuertemente estocástico se genera una nueva población de individuos tomando en cuenta su fitness. Básicamente durante la selección se decide cuáles individuos serán padres de la nueva generación. La nueva población puede remplazar completamente a la población anterior o solamente a los peores individuos, las peores soluciones.

Con los cromosomas candidatos a ser padres de la nueva población se efectúan cruza y mutaciones. Las cruza son intercambios de genes: el proceso consiste en intercambiar segmentos de los cromosomas de las parejas seleccionadas en forma aleatoria. Cuando un cromosoma sufre una mutación, el alelo de uno de sus genes cambia en forma aleatoria. Las mutaciones ocurren según una probabilidad de mutación, esta probabilidad es uno de los parámetros que gobierna la evolución.

Finalmente, la población nace y se decodifica el genotipo en fenotipo para evaluar su fitness. Al volver al principio del ciclo evolutivo verificamos nuevamente si nuestro mejor individuo supera los requisitos de la búsqueda, en caso contrario volvemos a repetir todo el proceso para obtener una nueva generación.

8.3. Diseño de la solución de un problema mediante AGs

Cuando queremos resolver un problema mediante AGs debemos determinar un conjunto de especificaciones clave:

Representación de los individuos: ¿cómo representamos una solución de nuestro problema mediante cromosomas?, ¿a partir de un conjunto de cromosomas dado, cómo obtenemos una solución? Deberemos determinar de qué forma se traduce el fenotipo en genotipo y viceversa.

Función de fitness: ¿cómo medimos la capacidad de supervivencia de un individuo, sus posibilidades de procrear y transferir la información de sus genes a la próxima generación? En el dominio de las soluciones debemos poder medir qué tan buena es cada solución con relación a las demás.

Mecanismo de selección: tenemos toda una población evaluada según el fitness y debemos elegir a los individuos que serán padres de la próxima generación. No es tan sencillo como elegir los M mejores. Veremos algunas formas de realizar una selección que, si bien premia a los mejores, no deja de dar la posibilidad azarosa de que uno de los peores individuos sea padre, como sucede en la naturaleza.

Operadores de variación y reproducción: los operadores básicos son las cruas y mutaciones. Sin embargo, a pesar de que limitemos el estudio sólo a estos operadores, veremos varias formas de aplicarlos. A partir de los operadores podemos reproducir y obtener la nueva población. Durante la reproducción también tenemos que considerar algunas opciones.

8.4. Representación de los individuos

El primer aspecto a resolver es la codificación del problema en un alfabeto finito. Tradicionalmente se han empleado cadenas binarias, pero recientemente se están empleando esquemas más flexibles. Debemos encontrar la forma de pasar desde el espacio del genotipo al espacio del fenotipo como muestra la Figura 8.1.

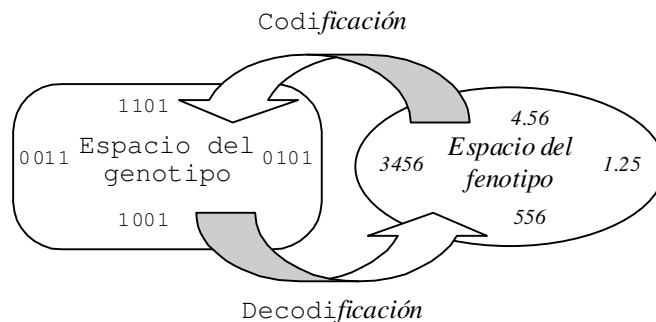


Figura 8.1. Representación de los individuos. Buscamos la forma de codificar soluciones en cromosomas y viceversa.

Para los AGs utilizamos la base canónica $\{0,1\}$ en cadenas de longitud fija y finita para representar soluciones de nuestro problema. Lo que representamos en estas cadenas depende de la aplicación. Podemos representar las conexiones, componentes y valores de éstos para un circuito electrónico; las líneas del código de un programa; los pesos y conexiones estructurales de una red neuronal y una infinidad de ejemplos más. Entre estos también encontramos a los coeficientes de un filtro ARMA o de un sistema no lineal.

Vamos a utilizar el caso más sencillo que consiste en representar una serie de coeficientes que describen algún sistema. Podemos utilizar un cromosoma para cada coeficiente. El cromosoma codificará en forma binaria el valor del coeficiente. Existen muchos métodos para codificar en forma binaria un valor real o entero, sólo veremos el caso más sencillo.

Debemos decidir con qué resolución queremos representar a cada coeficiente. Cuando más bits tenga nuestra cadena más amplio será el espacio de búsqueda. Es decir, deberemos establecer un compromiso entre la resolución de la codificación y la cantidad de dimensiones del espacio de búsqueda.

Suponga que necesitamos codificar un coeficiente x en el rango $[a, b]$ mediante un cromosoma de n bits (genes). Debemos seguir dos pasos:

1. Aplicar un factor de escala y truncamiento para convertir al real x en un entero X perteneciente al rango $[0, 2^n - 1]$
2. Convertir el entero X en un número binario.

Para decodificar y transformar el genotipo en fenotipo aplicamos los pasos inversos:

1. Convertimos el número binario de la cadena del cromosoma en entero X .
2. Aplicar un factor de escala para convertir el entero X perteneciente al rango $[0, 2^n - 1]$ en el real x en $[a, b]$.

8.5. Función de fitness

Debemos obtener ahora una medida para conocer qué tan buena es la solución de cada individuo. Esta función trabaja en el dominio del problema, sobre el fenotipo de cada individuo. Por lo tanto para obtener el valor de fitness para un individuo deberemos previamente hacerlo *nacer*, y en algunos casos *crecer*, a partir de su genotipo para luego evaluar objetivamente qué tan bueno es en relación con los otros.

Hay un aspecto fundamental a resolver, el valor de fitness debe ser monotónicamente creciente con la bondad de la solución que un individuo representa. Si ésta dependiera de varios factores, de la forma en que éstos sean pesados en la función de fitness dependerá la optimización que realice el AG.

El “cómo” nace y crece un individuo es completamente dependiente de la aplicación. En el caso de que estemos optimizando la trayectoria de un móvil robot deberemos tener un robot real para cada fenotipo necesario o, como generalmente se resuelve, utilizar una buena simulación de su comportamiento. En el otro extremo, la sencillez está representada por la simple evaluación de una ecuación que es directamente una medida del fitness. En el caso de la identificación de un sistema deberemos utilizar las ecuaciones en recurrencia que lo caracterizan para obtener su respuesta y compararla con la deseada. En este caso el error cuadrático medio puede considerarse como una medida inversa del fitness del individuo.

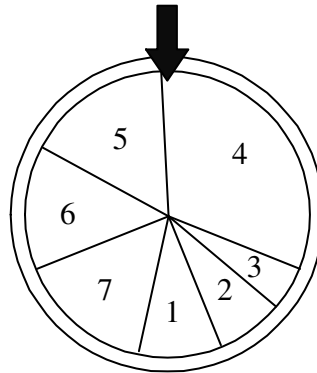


Figura 8.2. En este ejemplo de la rueda de ruleta el individuo número 4 es el que más probabilidades tiene de ser seleccionado. Sin embargo cualquiera puede ser padre.

8.6. Selección

Existen varias formas de realizar la selección de los progenitores. Recordemos que, al igual que en la naturaleza, no se selecciona simplemente los mejores. La selección no está relacionada directamente con el fitness de un individuo sino a través de operadores probabilísticos. Veamos algunos métodos sencillos.

8.6.1. Rueda de ruleta

También denominado *selección basada en rangos*, consiste en asignar a cada individuo una porción de la ruleta que es proporcional a su fitness. Luego, hacemos rodar la ruleta y el favorecido es seleccionado para ser padre en la próxima generación. A continuación mostramos el algoritmo del método de rueda de ruleta y en la Figura 8.2 un ejemplo gráfico.

$sumF$ = suma de todos los fitness

$sumR = rand(0, 1)sumF$

$sumP = 0, j = 0$

Repetir

$j = j + 1$

$sumP = sumP + fitness_j$

Hasta ($sumP \geq sumR$)

Seleccionado = j

En poblaciones grandes es común que unos pocos individuos extraordinarios se encuentren sumergidos en un mar de colegas mediocres. Es por esta

razón que la selección de la rueda de ruleta suele generar una convergencia lenta para el AG. Las áreas asignadas para los individuos buenos son proporcionalmente mayores pero hay tantos individuos mediocres que su área total es muy significativa. La selección no favorece suficientemente a los buenos individuos. Existe una técnica denominada *escalamiento de fitness* que ataca este problema justamente redefiniendo las distancias en fitness entre los mejores y los mediocres y peores. Sin embargo vamos a ver otras técnicas de selección en la que este problema no es tan marcado y aseguran una convergencia más rápida para el algoritmo.

8.6.2. Ventanas

En este método es necesario que primero ordenemos por fitness descendiente a los individuos. Hacemos N ventanas o rangos de selección $[0, q_i]$. El límite superior q_i va creciendo hasta llegar al total de la población. De cada ventana se elige al azar un individuo que queda seleccionado para ser padre.

En este método la mayor probabilidad de ser padre se asigna a los primeros individuos en la ordenación ya que están en todas las ventanas de selección. Los últimos individuos (menor fitness) sólo se encontrarán en condiciones de ser elegidos cuando se seleccione en la ventana $[0, q_N]$.

8.6.3. Competencias

En este caso se eligen, completamente al azar, $k > 1$ individuos; se los hace competir por fitness y queda seleccionado el ganador. Generalmente se utilizan valores de k entre 2 y 5 dependiendo del tamaño de la población. Este método es uno de los más utilizados debido a lo simple de su implementación.

8.7. Reproducción y operadores de variación

La reproducción es el proceso mediante el cual se obtiene la nueva población a partir de los individuos seleccionados y los operadores de variación. Existen varias alternativas para realizar la reproducción, en el caso más sencillo se obtienen *todos* los individuos de la nueva población a partir de variaciones (cruzas y mutaciones) de los progenitores.

Es posible también transferir directamente a la población nueva los padres seleccionados en la población anterior y completar los individuos faltantes mediante variaciones. En este caso interviene un parámetro denominado *brecha generacional* que determina cuántos padres son copiados

directamente desde la población anterior a la nueva. Este parámetro es un número G real que está entre 0 y 1. Multiplicándolo por la cantidad total de individuos de la población permite encontrar la cantidad de padres que serán copiados.

Otra alternativa algo menos biológica pero que es utilizada con muy buenos resultados es el *elitismo*. En esta estrategia se busca el mejor individuo de la población anterior e *independientemente de la selección y variación* se lo copia exactamente en la nueva población. De esta manera nos aseguramos de no perder la mejor solución generación tras generación. Como veremos esta estrategia permite elevar el índice de mutaciones y aumentar así la dispersión de las soluciones en el espacio de búsqueda. Vamos a ver a continuación los dos operadores de variación más utilizados.

8.7.1. Mutaciones

La mutación trabaja invirtiendo alelos de genes con una probabilidad p_m muy baja, por ejemplo $p_m = 0,001$ (ver Figura 8.3). Las mutaciones son típicamente realizadas con una probabilidad uniforme en toda la población y el número de mutaciones por individuo puede ser fijado de acuerdo a esta probabilidad y la cantidad de individuos. En los casos más simples se da la posibilidad de mutar sólo un alelo por individuo o se distribuye uniformemente sobre todo el cromosoma.

El papel fundamental de las mutaciones es no dejar que todos los individuos de la población se conviertan en mediocres (caigan en mínimos locales) permitiendo que constantemente se redistribuya la población sobre el espacio de búsqueda. Sin embargo una probabilidad de mutaciones demasiado elevada podría hacer que perdamos buenas soluciones impidiendo o retardando la convergencia. Cuando utilizamos elitismo nos aseguramos de no perder la mejor solución en probabilidades altas de mutación.

8.7.2. Cruzas

La cruce es un operador que, en el caso más sencillo, actúa sobre dos cromosomas para obtener otros dos. Existen dos tipos de cruces: cruces simples y cruces múltiples. En las cruces simples se elige un punto de cruce al azar y se intercambia el material genético correspondiente (ver Figura 8.4). En la cruce múltiple puede cortarse el cromosoma en más de dos partes para realizar el intercambio. En ambos casos el o los puntos de cruce son elegidos al azar. Veamos a continuación el algoritmo para la cruce simple.

```

cruza = rand(1, LongCromo - 1)
Para j de 1 a cruza
    Hijo1j = Mutar(Padre1j)
    Hijo2j = Mutar(Padre2j)
FinPara
Para j de cruza a LongCromo
    Hijo1j = Mutar(Padre2j)
    Hijo2j = Mutar(Padre1j)
FinPara

```

8.8. Características principales

Concebidos como una arquitectura genérica, como una técnica de búsqueda fuerte; cuando se aplican a problemas particulares los algoritmos genéticos deben sufrir modificaciones, para que presenten un buen desempeño y eficiencia. Así, a pesar de que tradicionalmente han empleado cromosomas binarios para codificar las soluciones, recientemente se está generalizando el empleo de estructuras más complejas de representación, como vectores de números reales, árboles, etc. Los operadores genéticos también sufren adecuaciones cuando se modifica la representación de las soluciones, en general, los nuevos operadores están en función del problema particular. Sin embargo, a diferencia de otras técnicas de computación evolutiva, el aspecto central de los algoritmos genéticos es la importancia que cumple el operador de cruzamiento y su relación con el operador de mutación. Mientras el operador *cruza* tiene una tasa o probabilidad elevada en todo algoritmo genético, el parámetro probabilidad de mutación tiene generalmente valores menos significativos.

Consideremos algunos ejemplos prácticos para hacer una comparación entre los algoritmos genéticos y los métodos de búsqueda por gradientes que vimos anteriormente.

Tomemos el ejemplo de la optimización para el sistema AR de segundo orden cuya superficie de error (fitness invertido) está dada por la suma de cuatro funciones gaussianas invertidas, como muestra la Figura 8.5. Agregue además la particularidad de que uno de los valles del error está solo 10^{-5} veces más cerca del cero que los otros tres. Encontrar el mínimo global implica encontrar el mínimo del valle que llega más abajo. Mientras un AG distribuye la población uniformemente sobre la superficie y analiza en paralelo todo el espacio de soluciones, el método de gradientes se encuentra limitado a la baja probabilidad de que, dadas las condiciones iniciales,

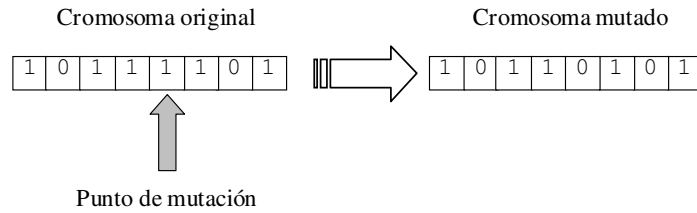


Figura 8.3. Mutación en un cromosoma de 8 genes.

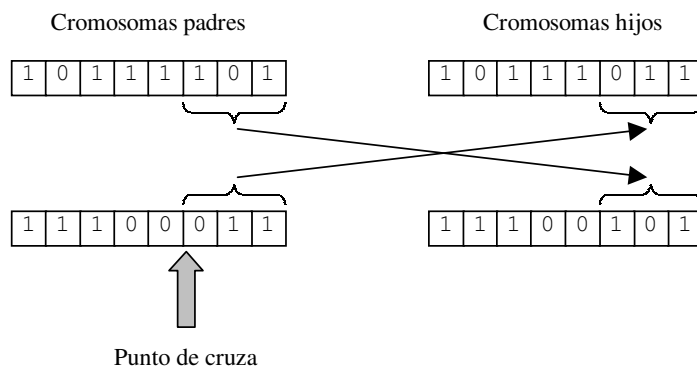


Figura 8.4. Cruza simple a partir de dos cromosomas de 8 genes.

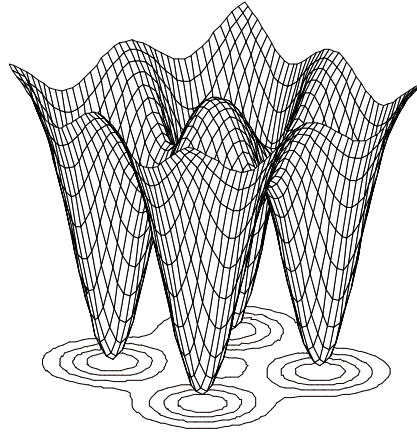


Figura 8.5. Curva de error o fitness invertido para un espacio de soluciones de dimensión 2. Uno de los picos llega 10^{-5} veces más abajo que los otros tres.

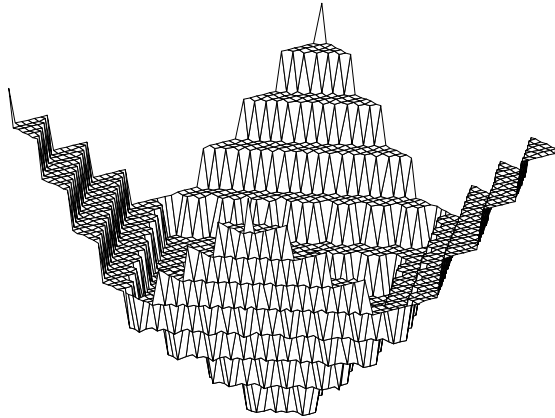


Figura 8.6. Curva de error o fitness invertido para un espacio de soluciones de dimensión 2. Los métodos basados en el gradiente pueden quedarse fijos en cualquiera de los escalones de la superficie.

el vector de coeficientes se encuentre en el valle apropiado. El AG busca simultáneamente es todo el espacio de soluciones, analiza *toda* la superficie antes de tomar una decisión. En contraste, el método de gradiente analiza sólo un punto de la superficie y un entorno pequeño de éste.

Otro caso donde el AG se ve favorecido es cuando la curva de error posee mesetas o escalones como se muestra en la Figura 8.6. ¿Qué pasa ahora con el método del gradiente del error en los escalones? Como el gradiente da cero si estamos en un escalón, el método no puede determinar la dirección que hay que seguir para caer en la superficie. Nuevamente el AG se ve favorecido

por analizar simultáneamente muchos puntos uniformemente distribuidos en la superficie.

A pesar de estas grandes ventajas del método no podemos dejar de mencionar que generalmente los algoritmos genéticos requieren un tiempo de cálculo y capacidad de almacenamiento mucho mayor que

otros métodos más convencionales. Este es el costo de la versatilidad y amplio espectro de aplicaciones que ofrecen los algoritmos genéticos.

Resumiremos en la siguiente lista las principales diferencias entre los algoritmos genéticos y otros métodos más tradicionales en la optimización y búsqueda:

1. Los AGs trabajan con una codificación del conjunto de parámetros de nuestro problema y no con los parámetros en sí mismos.
2. Los AGs buscan en una población de puntos en el espacio de soluciones, no en un punto en particular.
3. Los AGs utilizan la información de la función objetivo solamente y no sus derivadas o conocimiento auxiliar.
4. Los AGs utilizan reglas probabilísticas para las transiciones, no reglas determinísticas.

8.9. Introducción a los fundamentos matemáticos

Las características descritas de un algoritmo genético, responden a un cuerpo teórico sólido que los soporta. Este se puede sintetizar en la *teoría de esquemas*. Un esquema es una cadena de caracteres definida sobre un alfabeto finito.

La teoría de esquemas predice el comportamiento de un esquema describiendo un subconjunto de cadenas con similitudes en ciertas posiciones. De acuerdo a esto, la solución óptima de un problema, se encuentra –debido a que el *teorema fundamental de los algoritmos genéticos* garantiza que el crecimiento de una instancia de un esquema determinado es aproximadamente proporcional al fitness relativo observado de tal esquema, dado que los esquemas con longitud definida pequeña, bajo orden y fitness superior al promedio de la población, son incrementados exponencialmente

en la población—, a través de las generaciones, mientras que los esquemas con valores de fitness inferiores al promedio, longitud definida grande y alto orden, desaparecen.

La *hipótesis de los bloques constructores* plantea que los esquemas con alto valor de fitness (mayor que el promedio de la población), y longitud definida pequeña, donde longitud es la distancia entre la primera y la última posición fija en toda la cadena, constituyen los bloques constructores, los cuales permiten localizar las cadenas de mayor valor de fitness (las mejores soluciones o soluciones óptimas), mediante muestras de estos bloques con valor de fitness alto (selección) y también por combinación de bloques constructores (cruzamiento). La búsqueda altamente eficiente de los algoritmos genéticos, es explicada por lo que se ha denominado *paralelismo implícito*, un aspecto fundamental de los mismos. Esto significa que a pesar de que un algoritmo genético procesa solamente N cromosomas o soluciones en forma directa, se ha demostrado, que implícitamente se evalúan o exploran N^3 esquemas, dado que en el caso de una representación binaria un cromosoma representa 2^d esquemas diferentes, donde d es la longitud del cromosoma.

8.10. Trabajos prácticos

Ejercicio 1: Implemente las estructuras de datos y algoritmos básicos para la solución de un problema mediante algoritmos genéticos. Pruebe estas rutinas buscando el mínimo global de las siguientes funciones:

- $f(x) = x^2$
con $x \in [-5, 12 \dots 5, 12]$,
- $f(x) = -x \sin(\sqrt{|x|})$
con $x \in [-512 \dots 512]$,
- $f(x, y) = (x^2 + y^2)^{0,25} [\sin^2(50(x^2 + y^2)^{0,1}) + 1]$
con $x, y \in [-100 \dots 100]$.

Ejercicio 2: ^(**) Población de escarabajos *Tribolium* Se ha utilizado un modelo demográfico para predecir la dinámica de una población de escarabajos de harina *Tribolium* bajo condiciones de laboratorio. Este modelo consiste en tres ecuaciones diferenciales acopladas, no lineales y estocásticas que bajo ciertas condiciones muestran un comportamiento caótico.

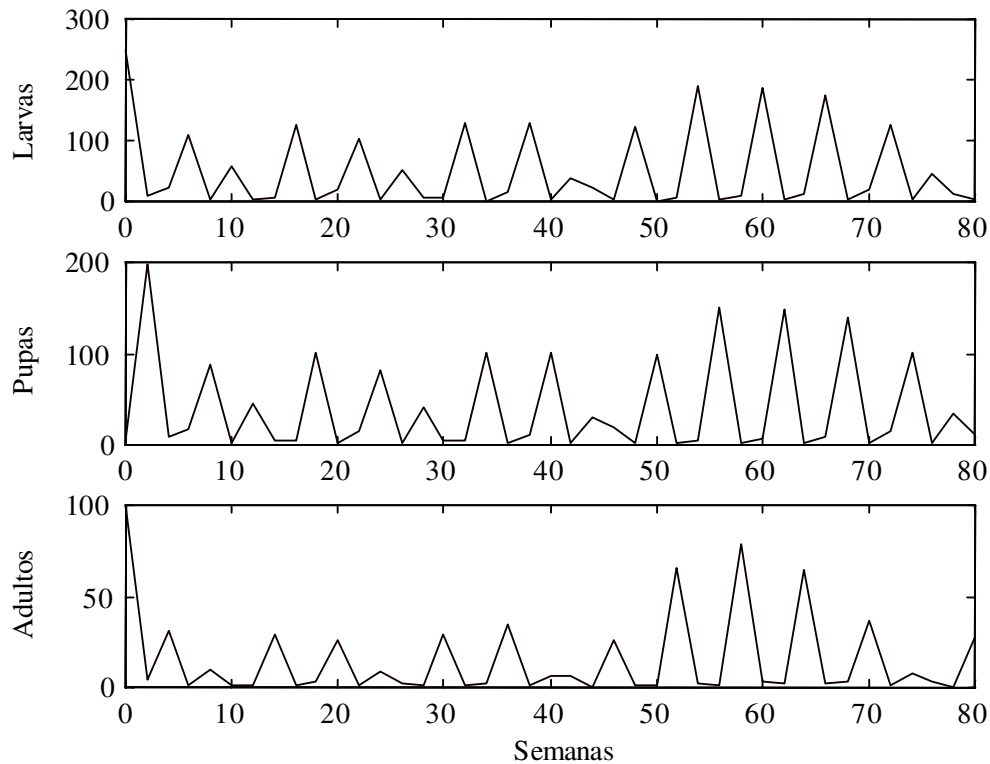


Figura 8.7. Evolución temporal de la población en sus tres estadios.

El modelo no ha sido completamente ajustado a los resultados obtenidos en experiencias de laboratorio y, si bien se conocen los rangos en los que pueden estar sus parámetros, no se conoce una buena aproximación de sus valores.

El objetivo principal de este trabajo es encontrar el conjunto de parámetros para los que el modelo se comporta de acuerdo a las mediciones experimentales realizadas en el laboratorio. Para lograr esto contamos con las mediciones experimentales, las ecuaciones en diferencia del modelo y los rangos biológicamente válidos en los que pueden encontrarse los valores de sus parámetros. Deberemos minimizar el error entre las mediciones de laboratorio y la salida del modelo mediante alguna técnica de identificación de sistemas.

Las características no lineales, el caos y las variables aleatorias presentes en el modelo dificultan la utilización de técnicas convencionales para la identificación de parámetros, y se propone en este ejercicio aplicar la técnica de algoritmos genéticos al problema de identificación de parámetros.

El modelo. El modelo describe la relación entre las poblaciones de escarabajos *Tribolium* de tres distintos estados de madurez. Las ecuaciones en diferencias de este modelo estocásticos son las siguientes:

$$\begin{aligned} L_{n+1} &= bA_n e^{-c_{el}L_n - c_{ea}A_n + E_{1n}} \\ P_{n+1} &= (1 - \mu_l)L_n e^{E_{2n}} \\ A_{n+1} &= [P_n e^{-c_{pa}A_n} + (1 - \mu_a)A_n] e^{E_{3n}} \end{aligned}$$

donde notamos como:

- n : tiempo discreto cuya unidad en el modelo es de 2 semanas.
- L_n : cantidad de larvas que sirven de alimento, para el tiempo n .
- P_n : cantidad de larvas que no sirven de alimento, pupas y adultos jóvenes, para el tiempo n .
- A_n : cantidad de adultos sexualmente maduros, para el tiempo n .
- b : número de larvas reclutadas por adulto y por unidad de tiempo en ausencia de canibalismo.
- μ_l : tasa de mortalidad de larvas en una unidad de tiempo.
- μ_a : tasa de mortalidad de adultos en una unidad de tiempo.
- $e^{-c_{el}L_n}$: probabilidad de que un huevo no sea comido en la presencia de L_n larvas, por unidad de tiempo (canibalismo).
- $e^{-c_{ea}A_n}$ probabilidad de que un huevo no sea comido en la presencia de A_n adultos, por unidad de tiempo (canibalismo).
- $e^{-c_{pa}A_n}$ probabilidad de supervivencia de pupas en presencia de A_n adultos, por unidad de tiempo.
- E_{Nn} variables de ruido aleatorio de una distribución normal multivariable acoplada con un vector de valor medio nulo y una matriz de varianza-covarianza Σ .

Las no linealidades exponenciales presentes en el modelo representan el canibalismo sobre los huevos que realizan larvas y adultos y sobre las larvas que realizan pupas y adultos. Las variables aleatorias representan desviaciones impredecibles de las observaciones realizadas sobre la estructura determinística ($\Sigma = \mathbf{0}$) del modelo, provocadas por cambios ambientales entre otras causas.

Datos experimentales. A partir de una población inicial de 250 larvas pequeñas, 5 pupas y 100 adultos, se realizaron los recuentos de la cantidad de escarabajos en cada etapa durante 80 semanas con una periodicidad de 2 semanas. Los resultados se guardaron en un archivo de texto que contiene 3 columnas y 40 filas. En cada fila se encuentran los recuentos correspondientes a un período de 2 semanas que según las columnas corresponden a L_n , P_n y A_n .

De acuerdo a las condiciones de laboratorio en la que evolucionó la población podemos estimar cierto rango para los valores de los parámetros. Además con relación al papel que desempeñan en el modelo sabemos que todos deben ser mayores que cero. Todas estas consideraciones nos permitirán reducir el espacio de búsqueda:

$$\begin{aligned} b &< 10^1 \\ \mu_l &< 10^0 \\ \mu_a &< 10^0 \\ c_{el} &< 10^{-1} \\ c_{ea} &< 10^{-1} \\ c_{pa} &< 10^0 \end{aligned}$$

En la Figura 8.7 vemos tres gráficas que muestran la evolución de la población bajo condiciones de laboratorio en un período de 80 semanas.

Acerca de la implementación. Se recomienda utilizar las siguientes pautas para la implementación:

- Codificación binaria de 16 bits de resolución ajustada a rango de variación de cada parámetro.
- Cruzas simples.

- Bajas probabilidades de mutación ($< 0,1$ no elitista, $< 0,4$ elitista).
- Una estructura determinística del modelo para simplificar este trabajo práctico.
- No menos de 100 individuos por generación.

Bibliografía

- [1] T. Bäck, U. Hammel, and H-F. Schewfel. Evolutionary computation: Comments on history and current state. *IEEE Trans. on Evolutionary Computation*, 1(1), 1997.
- [2] R. F. Constantino, R. A. Desharnais, J. M. Cushing, and B. Dennis. Chaotic dynamics in an insect population. *Science*, 275(5298):389–391, 1 1997.
- [3] D. E. Goldberg. *Genetic Algorithms in Search, Optimization & Machine Learning*. Addison Wesley, 3 1997. Capítulos 1, 3 y 4.
- [4] V. Maniezzo. Genetic evolution of the topology and weight distribution of neural networks. *IEEE Trans. on Neural Networks*, 5(1), 1 1994.
- [5] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag, 1992.
- [6] D. J. Montana. Neural networks weight selection using genetic algorithms. In S. Goonatilake and K. Sukhdev, editors, *Intelligent Hybrid Systems*, chapter 5. J. Wiley & Sons, 1995.
- [7] L. J. Park, C. H. Park, C. Park, and T. Lee. Application of genetic algorithms to parameter estimation of bioprocesses. *Medical & Biological Engineering & Computing*, 1 1997.

Apéndice A

Octave (v2.1.36)

A.1. Generales

- `octave` - Inicia una sesión de octave.
- `octave --traditional` - Inicia una sesión de octave en formato compatible con MatLab.
- `quit` o `exit` - Termina la sesión de Octave.
- `INTERRUPT` (*o sea* `C-C`) Termina el comando actual y devuelve el prompt.
- `help` - Lista todos los comandos y variables integradas.
- `help comando` - Describe brevemente el *comando*.
- `help -i` - Usa Info para navegar el manual de ayuda de Octave.
- `help -i comando` - Busca *comando* en el manual de Octave usando la interface de Info.

A.2. Comandos del sistema

- `cd dir` - Cambia el directorio de trabajo a *dir*.
- `pwd` - Muestra el directorio de trabajo actual.
- `ls [opciones]` - Muestra un listado de archivos y directorios del directorio actual.
- `getenv cadena` - Devuelve el valor de una variable de entorno.
- `system cmd` - Ejecuta un comando del sistema operativo.

A.3. Matrices y rangos

Las matrices se delimitan con corchetes. Los elementos de una fila se separan con comas. Las filas se separan con punto y coma (;). Las comas se pueden reemplazar por espacios y los punto y coma por uno o más enter. Los elementos de una matriz pueden ser expresiones arbitrarias, siempre que las dimensiones concuerden.

- `[x, y, ...]` - Ingresar un vector fila
- `[x; y; ...]` - Ingresar un vector columnar
- `[w, x; y, z]` - Ingresar una matrix de 2×2 .
- `base : limite`
- `base : incremento : limite`

Especifican un rango de valores empezando desde *base* con elementos no mayores de *límite*. Si no se pone un *incremento*, el valor por defecto es 1. Se pueden usar incrementos negativos.

A.4. Algunas variables predefinidas

- EDITOR - Especifica el editor a usar con el comando `edit`.
- Inf, NaN - Infinito, indeterminación (Not-a-Number).
- LOADPATH - Define el camino de búsqueda para los archivos de funciones.
- ans - Devuelve el último resultado que no haya sido explícitamente asignado.
- eps - Presición de la maquina.
- pi - π
- realmax - Valor máximo representable.
- realmin - Valor mínimo representable.

A.5. Operaciones aritméticas y operadores de incremento

- $x + y$ - Suma de matrices.
- $x - y$ - Resta de matrices.
- $x * y$ - Multiplicación Matricial.
- $x .* y$ - Multiplicación elemento a elemento.
- x / y - División matricial por la derecha, equivalente conceptualmente a $(\text{inv}(y') * x')'$.
- $x ./ y$ - División por la derecha elemento a elemento.
- $x \setminus y$ - División matricial por la derecha, equivalente conceptualmente a $\text{inv}(x) * y$
- $x .\setminus y$ - División elemento a elemento por la izquierda.
- $x ^ y$ - Operador de potencias matricial.
- $x .^ y$ - Operador de potencias elemento a elemento.
- $- x$ - Negación.
- $x '$ - Transpuesta compleja conjugada.
- $x .'$ - Transpuesta.
- $++ x$ ($-- x$) - Incrementa (decrementa) x , devuelve el *nuevo* valor.
- $x ++$ ($x --$) - Incrementa (decrementa) x , devuelve el valor *anterior*.

A.6. Operadores booleanos y de comparación

- $x < y$ - Verdadero si x es menor que y .
- $x \leq y$ - Verdadero si x es menor o igual que y .
- $x > y$ - Verdadero si x es mayor que y .
- $x \geq y$ - Verdadero si x es mayor o igual que y .
- $x == y$ - Verdadero si x es igual a y .
- $x != y$ - Verdadero si x no es igual a y .
- $x \& y$ - Verdadero si ambos, x e y son verdaderos.
- $x | y$ - Verdadero si al menos uno de x o y es verdadero.
- $! bool$ - Verdadero si *bool* es falso.

A.7. Sentencias

- `for identificador = expr`
`lista`
`endfor`
 Ejecutar la lista de comandos *lista* una vez por cada columna de *expr*. La variable *identificador* se setea al valor de la columna actual para cada iteración.
- `while (condición)`
`lista`
`endwhile`
 Ejecuta la lista de comandos *lista* mientras la *condición* sea verdadera.
- `break` - Sale del bucle más interno.
- `continue` - Vuelve al inicio del bucle mas interno.
- `return` - Sale de una función y devuelve el control al proceso que la había llamado.
- `if (condición)`
`lista-if`
`[else`
`lista-else]`
`endif`
 Ejecutar la lista de comandos *lista-if* si la *condición* es verdadera, de otra forma ejecuta la lista *lista-else*.
- `if (condición)`
`lista-if`
`[elseif (condición2)`
`lista-elseif]`
`[else`
`lista-else]`
`endif`
 Ejecutar la lista de comandos *lista-if* si la *condición* es verdadera, de otra forma ejecutar la lista *lista-elseif* correspondiente a la primera condición **elseif** que sea verdadera, de otra forma ejecutar *lista-else*.
 Cualquier número de cláusulas **elseif** puede aparecer en una sentencia **if**.

A.8. Manipulaciones básicas de matrices

- `rows(a)` - Devuelve el número de filas de *a*
- `columns(a)` - Devuelve el número de columnas de *a*
- `all(a)` - chequea si todos los elementos de *a* son distintos de cero.
- `any(a)` - chequea si cualquier elemento de *a* es distinto de.
- `find(a)` - Devuelve los índices de los elementos distintos de cero.
- `sort(a)` - Ordena los elementos de cada columna de *a*.
- `sum(a)` - Suma los elementos en cada columna de *a*.
- `prod(a)` - Producto de los elementos en cada columna de *a*.
- `min(args)` - Encuentra el mínimo.
- `max(args)` - Encuentra el máximo.
- `rem(x, y)` - Calcula el resto de x/y .
- `reshape(a, m, n)` - Reformatea la matriz *a* para que sea de *m* por *n*.
- `diag(v, k)` - Crea una matriz diagonal.
- `linspace(b, l, n)` - Crea un vector de elementos espaciados linealmente.
- `logspace(b, l, n)` - Crea un vector con espaciamiento logarítmico.

- `eye(n, m)` - Crea una matriz identidad de n por m .
- `ones(n, m)` - Crea una matriz de unos de n por m .
- `zeros(n, m)` - Crea una matriz de ceros de n por m .
- `rand(n, m)` - Crea una matriz de n por m con valores aleatorios con distribución uniforme entre 0 y 1.
- `randn(n, m)` - Crea una matriz de n por m con valores aleatorios con distribución normal de varianza 1 y media 0.

A.9. Funciones trigonométricas

- `sin(a)` `cos(a)` `tan(a)` - Funciones trigonométricas.
- `asin(a)` `acos(a)` `atan(a)` - Funciones trigonométricas inversas.
- `sinh(a)` `cosh(a)` `tanh(a)` - Funciones trigonométricas hiperbólicas.
- `asinh(a)` `acosh(a)` `atanh(a)` - Funciones trigonométricas hiperbólicas inversas.

A.10. Algebra Lineal

- `det(a)` - Calcula el determinante de una matriz.
- `eig(a)` - Autovalores y autovectores de una matriz.
- `expm(a)` - Calcula la exponencial de una matriz.
- `inverse(a)` - Inversa de una matriz cuadrada.
- `norm(a, p)` - Calcula la norma p de una matriz.
- `pinv(a)` - Calcula la pseudoinversa de la matriz a .
- `rank(a)` - Rango de una matriz.

A.11. Procesamiento de Señales

- `fft(a)` - Transformada Rápida de Fourier.
- `ifft(a)` - Transformada Rápida de Fourier Inversa.
- `freqz(args)` - Respuesta en frecuencia de filtros digitales.
- `sinc(x)` - Función Sinc ($\sin(\pi x)/(\pi x)$)

A.12. Procesamiento de Imágenes

- `colormap(map)` - Fija el mapa de colores actual.
- `image(img, zoom)` - Muestra una matriz de imagen de Octave.
- `imagesc(img, zoom)` - Muestra una matriz escalada como una imagen.

A.13. Funciones de entrada/salida

- `fopen(nombre, modo)` - Abre el archivo *name* en modo *modo*.
- `fclose(arch)` - Cierra el archivo *arch*
- `fprintf(arch, fmt, ...)` - Salida formateada a *arch*.
- `sprintf(fmt, ...)` - Salida formateada a una cadena.
- `fscanf(arch, fmt)` - Entrada formateada desde *arch*
- `sscanf(str, fmt)` - Entrada formateada desde la cadena *str*

- `fgets(arch, len)` - Lee *len* caracteres del archivo *arch*
- `ftell(arch)` - Fija la posición del puntero de lectura en un archivo binario.
- `frewind(arch)` - Mueve el puntero de un archivo al principio.
- `fread(arch, size, prec)` - Lee datos binarios de un archivo.
- `fwrite(arch, size, prec)` - Escribe datos binarios en un archivo.
- `feof(arch)` - Determina si el puntero de un archivo está al final del archivo.
- `save arch var ...` - Grabar variables en *arch*
- `load arch` - Cargar variables desde *arch*
- `disp(var)` - Mostrar el valor de *var* en la pantalla.

A.14. Misceláneas

- `eval(str)` - Evaluar la cadena *str* como un comando.
- `feval(str, ...)` - Evaluar la función de nombre *str*, pasando los demás argumentos a la función.
- `error(message)` - Imprimir un mensaje y retornar al nivel de ejecución superior.
- `clear pattern` - Borrar las variables concordantes con el patrón *pattern*.
- `exist(str)` - Chequea la existencia de una variable o función.
- `who` - Muestra las variables actuales (formato corto).
- `whos` - Muestra las variables actuales (formato largo).

A.15. Polinomios

- `conv(a, b)` - Convolución y producto de polinomios.
- `deconv(a, b)` - Deconvolución y división de polinomios.
- `poly(a)` - Crear un polinomio a partir de una matriz.
- `polyval(p, x)` - Valor de un polinomio *p* en *x*
- `roots(p)` - Encontrar las raíces de un polinomio.

A.16. Estadística

- `corrcoef(x, y)` - Coeficiente de correlación.
- `cov(x, y)` - Covarianza.
- `mean(a)` - Valor medio.
- `median(a)` - Mediana.
- `std(a)` - Desvío estándar.
- `var(a)` - Varianza.

A.17. Gráficos básicos

- `gplot [ranges] expr [using] [title] [style]` - Gráficos en 2D.
- `gsplot [ranges] expr [using] [title] [style]` - Gráficos en 3D.

ranges - Especifican el rango de los datos.

expr - Expresión a graficar.

using - Especifica las columnas a graficar.

title - Especifica un título para la gráfica.

style - Especifica el estilo de línea para la gráfica

Si los rangos *ranges* se especifican, deben estar antes de la expresión a graficar. Las opciones *using*, *title*, y *style* pueden aparecer en cualquier orden después de la expresión *expr*. Se pueden graficar múltiples expresiones en un solo gráfico, separándolas por comas.

- *setoptions* - Fijar las opciones de graficación.
- *showoptions* - Mostrar las opciones de graficación.
- *replot* - Redibujar el gráfico actual.
- *closeplot* - Cerrar la corriente a los procesos de *gnuplot*.
- *purge_tmp_files* - Borrar los archivos de dibujo temporales.

A.18. Otras funciones de graficación

- *plot(args)* - Gráfico en 2D con ejes lineales.
- *semilogx(args)* - Gráfico en 2D con eje x logarítmico.
- *semilogy(args)* - Gráfico en 2D con eje y logarítmico.
- *loglog(args)* - Gráfico en 2D logarítmico en ambos ejes.
- *bar(args)* - Gráfico de barras.
- *stairs(x, y)* - Gráfico de escaleras.
- *hist(y, x)* - Histograma.
- *title(string)* - Fija el título de un gráfico.
- *axis(limits)* - Fija los rangos de un gráfico.
- *xlabel(string)* - Fija una etiqueta para el eje x.
- *ylabel(string)* - Fija una etiqueta para el eje y.
- *grid [on|off]* - Setea el estado de la grilla.
- *hold [on|off]* - Setea el estado de retención de un gráfico.
- *mesh(x, y, z)* - Gráfico de una superficie en 3D.
- *meshgrid(x, y)* - Crea matrices de coordenadas para el mesh.

Apéndice B

Comandos de SciLab (v2.6)

B.1. Señales

- `Signal` - Descripción manual de una señal.
- `analpf` - Crear un filtro analógico pasabajos.
- `buttmag` - Respuesta de un filtro Butterworth.
- `casc` - Realización de filtros en cascada a partir de los coeficientes.
- `cepstrum` - Cálculo del Cepstro.
- `cheb1mag` - Respuesta de un filtro de Chebyshev tipo 1.
- `cheb2mag` - Respuesta de un filtro de Chebyshev tipo 2.
- `chepol` - Polinomio de Chebyshev.
- `convol` - Convolución.
- `corr` - Correlación, Covarianza.
- `cspect` - Estimación espectral por el método de correlación.
- `dft` - Transformada Discreta de Fourier.
- `ell1mag` - Magnitud de un filtro Elíptico.
- `eqfir` - Aproximación minimax para un filtro FIR.
- `eqiir` - Diseño de filtros IIR.
- `ffilt` - Coeficientes de un filtro FIR pasabajos.
- `fft` - Transformada Rápida de Fourier.
- `filter` - Aplicación de un filtro.
- `frmag` - Magnitud de Filtros FIR e IIR.
- `fsfirlin` - Diseño de filtros FIR de fase lineal por la técnica de muestreo en frecuencias.
- `group` - Retardo de grupo de un filtro digital.
- `hilb` - Transformada de Hilbert.
- `iir` - Filtro Digital IIR.
- `iirgroup` - Optimización del retardo de grupo de filtros pasabajos IIR.
- `iirlp` - Optimización de filtros IIR pasabajos.
- `intdec` - Interpolación/decimación de una señal (cambio de frecuencia de muestreo).
- `lev` - Ecuaciones de Yule-Walker (algoritmo de Levinson)
- `levin` - Resolución de sistemas de Toeplitz por el algoritmo de Levinson (multidimensional)
- `mese` - Estimación espectral por maximización de entropía.
- `mfft` - Transformada rápida de Fourier multidimensional
- `pspect` - Estimación espectral cruzada entre dos series.
- `remez` - Algoritmo de Remez.
- `remezb` - Aproximación Minimax de la respuesta de magnitud.
- `sinc` - Función Sinc ($\text{Sen}(x)/x$).

- `trans` - Transformación de filtros pasabajos a otros.
- `wfir` - Filtros FIR de fase lineal.
- `wigner` - Distribución de Wigner-Ville.
- `window` - Ventana simétrica.
- `yulewalk` - Diseño de Filtros por mínimos cuadrados.
- `zpbutt` - Prototipo analógico de filtro Butterworth.
- `zpch1` - Prototipo analógico de filtro Chebyshev tipo 1.
- `zpch2` - Prototipo analógico de filtro Chebyshev tipo 2.
- `zpell` - Prototipo analógico de filtro Elíptico.

B.2. Sonido

- `analyze` - Gráfico de frecuencias de una señal sonora.
- `auread` - Cargar un archivo de sonido .au.
- `auwrite` - Escribir un archivo de sonido .au.
- `lin2mu` - Codificación mu-law de una señal lineal.
- `loadwave` - Cargar un archivo de sonido .wav.
- `mapsound` - Graficar un mapa de sonido.
- `mu2lin` - Codificación lineal de un sonido en mu-law.
- `playsnd` - Reproducir un sonido.
- `savewave` - Grabar datos en un archivo .wav.
- `sound` - Reproducir un sonido.
- `wavread` - Leer un archivo .wav.
- `wavwrite` - Grabar un archivo .wav.

B.3. Control

- `bilin` - Transformación bilineal general.
- `calfrq` - Discretización de la respuesta en frecuencia.
- `cls2dls` - Transformación bilineal.
- `dbphi` - Transformación de la respuesta en frecuencia a la representación de magnitud y fase.
- `dscr` - Discretización de sistemas lineales.
- `flts` - Respuesta temporal de sistemas discretos.
- `frep2tf` - Transformación de la respuesta en frecuencia a Función de Transferencia.
- `freq` - Respuesta en frecuencia.

B.4. Funciones elementales

- `abs` - Valor absoluto, magnitud.
- `acos` - Arco Coseno.
- `acosh` - Arco Coseno Hiperbólico.
- `and (&)` - And (Y) lógico.
- `asin` - Arco Seno.
- `asinh` - Arco Seno Hiperbólico.
- `atan` - Arco Tangente.
- `atanh` - Arco Tangente Hiperbólica.
- `ceil` - Redondear hacia $+\infty$.
- `conj` - Complejo Conjugado.
- `cos` - Función Coseno.
- `cosh` - Coseno Hiperbólico.

- `cotg` - Cotangente
- `coth` - Cotangente Hiperbólica.
- `diag` - Extracción de matriz diagonal.
- `erf` - Función de Error.
- `eye` - Matriz Identidad.
- `fix` - Redondear hacia cero.
- `floor` - Redondear hacia $-\infty$.
- `imag` - Parte Imaginaria.
- `int` - Parte Entera.
- `interp` - Interpolación
- `interp1n` - Interpolación Lineal.
- `isdef` - Verdadero si la variable existe.
- `isinf` - Verdadero para los valores infinitos.
- `isnan` - Verdadero para las indeterminaciones (Not-a-Number).
- `isreal` - Verdadero para variables reales.
- `kron` - Producto de Kronecker (externo).
- `linspace` - Vector espaciado linealmente.
- `log` - Logaritmo Natural.
- `log10` - Logaritmo base 10.
- `log2` - Logaritmo base 2.
- `logspace` - Vector espaciado logarítmicamente.
- `max` - Máximo.
- `mean` - Valor medio de un vector o matriz.
- `median` - Mediana de un vector o matriz.
- `min` - Mínimo.
- `modulo` - Resto de la división entera.
- `not (~)` - Not (NO) Lógico.
- `ones` - Matriz de unos.
- `or —` - Or (O) lógico.
- `prod` - Productoria.
- `rand` - Generador de números aleatorios.
- `real` - Parte Real.
- `round` - Redondeo al entero más cercano.
- `sign` - Función Signo.
- `sin` - Función Seno.
- `sinh` - Seno Hiperbólico.
- `size` - Tamaño de una matriz.
- `smooth` - Suavizado (interpolado) por splines.
- `sort` - Ordenar en orden decreciente.
- `sqrt` - Raíz Cuadrada.
- `squarewave` - Forma de onda Cuadrada de período $2 * \pi$
- `st_deviation` - Desvío Estándar.
- `sum` - Sumatoria.
- `tan` - Tangente.
- `tanh` - Tangente Hiperbólica.
- `trfmod` - Gráfico de ceros y polos.
- `tril` - Submatriz triangular inferior.
- `triu` - Submatriz Triangular Superior.
- `zeros` - Matriz de Ceros.

B.5. E/S a archivos

- `diary` - Grabar la sesión de Scilab en un archivo.
- `disp` - Mostrar variables o mensajes.
- `dispfile` - Mostrar las propiedades de los archivos abiertos.
- `file` - Manejo de archivos.
- `fileinfo` - Información sobre un archivo.
- `fprintf` - Emulación de la función de C `fprintf` (escritura en archivos de datos con formato).
- `fprintfMat` - Grabar una matriz en un archivo.
- `fscanf` - Lectura de datos con formato desde un archivo.
- `fscanfMat` - Lee una matriz desde un archivo de texto.
- `input` - Espera una entrada por teclado del usuario.
- `lines` - Define las filas y columnas usadas para el display.
- `load` - Cargar variables grabadas.
- `mclose` - Cerrar un archivo abierto.
- `meof` - Chequea el fin de un archivo.
- `mfprintf` - Convierte, formatea y escribe datos a un archivo.
- `mget` - Lee un byte o word en un formato binario dado y lo convierte a double.
- `mgeti` - Lee un byte o word en un formato binario dado y lo convierte a entero.
- `mgetl` - Lee líneas de un archivo de texto.
- `mgetstr` - Lee una cadena de caracteres de un archivo.
- `mopen` - Abre un archivo.
- `mprintf` - Convierte, formatea e imprime datos en la pantalla de SciLab.
- `mput` - Escribe un byte o word en un formato binario dado.
- `mputl` - Escribe cadenas en un archivo ASCII.
- `mseek` - Fija la posición actual en un archivo binario.
- `msprintf` - Convierte, formatea y escribe datos a una cadena.
- `read` - Lee matrices.
- `save` - Graba variables en archivos binarios.

B.6. Creación de funciones

- `argn` - Número de argumentos de una función.
- `comp` - Precompilación de una función.
- `edit` - Edición de funciones.
- `endfunction` - Cierra la definición de una función.
- `function` - Abre la definición de una función.
- `genlib` - Compila una librería a partir de todas las funciones de un directorio.
- `getd` - Carga todas las funciones definidas en un directorio.
- `getf` - Carga una función definida en un archivo.
- `newfun` - Agrega un nombre en la tabla de funciones.
- `varargin` - Número variable de argumentos en una lista de argumentos de entrada.
- `varargout` - Número variable de argumentos en una lista de argumentos de salida.

B.7. Gráficos

- `Matplot` - Gráfico en 2D de una matriz usando colores.
- `Matplot1` - Gráfico en 2D de una matriz usando colores.
- `bode` - Gráfico de Bode.
- `champ` - Campo vectorial 2D.
- `champ1` - Campo vectorial 2D con flechas de colores.
- `colormap` - Selecciona un mapa de colores.
- `contour` - Curvas de nivel de una superficie en 3D.
- `contour2d` - Curvas de nivel de una superficie en un gráfico de 2D.
- `contourf` - Curvas de nivel rellenas de una superficie en un gráfico de 2D.
- `drawaxis` - Dibujar un eje.
- `errbar` - Agregar barras de error verticales a un gráfico 2D.
- `eval3d` - Valores de una función sobre una grilla.
- `evans` - Lugar de raíces de Evans.
- `gainplot` - Gráfico de magnitud.
- `getlinestyle` - Diálogo para seleccionar el estilo de línea.
- `getmark` - Diálogo para seleccionar las marcas del gráfico.
- `getsymbol` - Diálogo para seleccionar los símbolos y sus tamaños para gráficos.
- `graduate` - Graduación de los ejes.
- `graycolormap` - Mapa de colores gris lineal.
- `hist3d` - Histograma 3D.
- `histplot` - Histograma.
- `hotcolormap` - Mapa de colores rojo a amarillo.
- `legends` - Leyendas del gráfico.
- `nyquist` - Gráfico de Nyquist.
- `param3d` - Gráfico 3D de una curva (paramétrica).
- `plot` - Gráfico simple.
- `plot2d` - Gráfico 2D.
- `plot2d1` - Gráfico 2D (con ejes logarítmicos)
- `plot2d2` - Gráfico 2D (de escalera)
- `plot2d3` - Gráfico 2D (con barras verticales)
- `plot2d4` - Gráfico 2D (Con flechas verticales)
- `plot3d` - Gráfico 3D de una superficie.
- `plzr` - Gráfico de polos y ceros.
- `polarplot` - Gráfico en coordenadas polares.
- `replot` - Redibuja la ventana de dibujo actual con nuevos límites.
- `sgrid` - Grilla de líneas en el plano s.
- `subplot` - Divide una ventana gráfica en una matriz de sub ventanas.
- `titlepage` - Agrega un título a la ventana de dibujo.
- `winsid` - Devuelve una lista de las ventanas de dibujo.
- `xaxis` - Dibuja un eje.
- `xbasr` - Limpia una ventana gráfica y borra los gráficos asociados.
- `xbasimp` - Envía gráficos a una impresora Postscript o a un archivo.
- `xbasr` - Redibuja una ventana gráfica
- `xclear` - Borra una ventana gráfica.
- `xgrid` - Agrega una grilla a un gráfico 2D.
- `xload` - Carga un gráfico grabado.
- `xsave` - Graba gráficos a un archivo.

B.8. Operaciones lineales de matrices

- `det` - Determinante.
- `exp` - Exponencial elemento a elemento.
- `expm` - Exponencial de una matriz cuadrada.
- `inv` - Inversa de una matriz.
- `linsolve` - Resolución de ecuaciones lineales.
- `orth` - Base ortogonal.
- `pinv` - Pseudoinversa.
- `rank` - Rango de una matriz.
- `svd` - Descomposición en valores singulares.

B.9. No lineal

- `derivative` - Derivada aproximada.
- `intc` - Integral de Cauchy.
- `intg` - Integral definida.
- `linpro` - Resolución de problemas de programación lineal.
- `ode` - Resolución de ecuaciones diferenciales ordinarias.
- `ode_discrete` - Resolución de ecuaciones diferenciales ordinarias, simulación discreta.
- `ode_root` - Resolución de ecuaciones diferenciales ordinarias con detección de raíces.
- `optim` - Rutina de optimización no lineal.
- `quapro` - Resolución de problemas de programación lineal cuadrática.

B.10. Polinomios

- `clean` - Limpiar matrices (redondea a cero los valores pequeños).
- `coeff` - Coeficientes del polinomio característico de una matriz.
- `denom` - Denominador.
- `determ` - Determinante de una matriz polinómica.
- `numer` - Numerador.
- `pdiv` - División de polinomios.
- `roots` - Raíces de un polinomio.

B.11. Programación

- `abort` - Interrumpir la ejecución.
- `ans` - Última respuesta.
- `apropos` - Busca palabras clave en la ayuda de SciLab.
- `backslash` (`\`) - División por la izquierda de matrices (premultiplicación por inversa).
- `break` - Comando para interrumpir bucles.
- `call` - Llamada a subrutinas.
- `case` - Comando para seleccionar entre varias opciones.
- `clear` - Elimina variables.
- `clearglobal` - Elimina variables globales.
- `colon` (`:`) - Operador de Rango.
- `comments` - Comentarios.
- `else` - Palabra clave de las sentencia if-then-else.
- `elseif` - Palabra clave de la sentencia if-then-else.
- `end` - Palabra clave de fin.
- `exec` - Ejecución de un archivo de script.

- `exit` - Termina la sesión de SciLab.
- `for` - Palabra clave para bucles.
- `global` - Definir una variable global.
- `hat (^)` - Exponenciación
- `if then else` - Ejecución condicional.
- `mtlb_mode` - Cambia a operaciones compatibles con MatLab.
- `pause` - Detiene la ejecución hasta apretar una tecla.
- `percent (%)` - Comentarios.
- `pwd` - Muestra el directorio actual de trabajo.
- `quote (')` - Transpuesta o delimitador de cadenas.
- `select` - Comando select (similar a switch).
- `then` - Palabra clave de la sentencia if-then-else
- `while` - Bucles condicionales.
- `who` - Listado de variables.
- `whos` - Listado de variables en formato largo.

B.12. Cadenas de caracteres

- `convstr` - Conversión de mayúsculas/minúsculas.
- `emptystr` - Cadena de longitud cero.
- `length` - Tamaño de un objeto.
- `part` - Extracción de subcadenas.
- `strcat` - Concatenar cadenas de caracteres.
- `strindex` - Buscar la posición de una cadena dentro de otra.
- `stripblanks` - Eliminar los espacios en blanco al inicio y al final de una cadena.
- `strsubst` - Sustituir una cadena de caracteres por otra dentro de una cadena.

B.13. Utilidades

- `chdir` - Cambiar el directorio actual.
- `dec2hex` - Conversión de decimal a hexadecimal.
- `help` - Ayuda en línea.
- `hex2dec` - Conversión de hexadecimal a decimal.
- `man` - Ayuda en línea en formato manpages.
- `timer` - Tiempo del CPU.

Apéndice C

Comandos de MatLab (v4.2)

C.1. Comandos de propósito general

C.1.1. Comandos básicos

- `help` - Documentación On-line.
- `doc` - Carga la documentación Hipertextual.
- `what` - Listado de directorios de los archivos .M, .MAT y .MEX.
- `type` - Mostrar un archivo .M.
- `lookfor` - Búsqueda por palabra clave en la base de datos de ayuda.
- `which` - Localiza funciones y archivos.
- `demo` - Ejecuta demos.
- `path` - Control del árbol de directorios de búsqueda de MatLab.

C.1.2. Manejo de variables y del espacio de trabajo

- `who` - Listado de variables en uso.
- `whos` - Listado de variables en uso, formato largo.
- `load` - Recuperar variables desde disco.
- `save` - Guardar variables del espacio de trabajo a disco.
- `clear` - Eliminar variables y funciones de la memoria.
- `pack` - Compactar la memoria del espacio de trabajo.
- `size` - Tamaño de una matriz.
- `length` - Longitud de un vector.
- `disp` - Mostrar un texto por pantalla.

C.1.3. Trabajo con archivos y con el sistema operativo

- `cd` - Cambiar el directorio de trabajo actual.
- `dir` - Listado de directorios.
- `delete` - Borrar un archivo.
- `getenv` - Obtener el valor de una variable de entorno.
- `!` - Ejecutar un comando del sistema operativo.
- `unix` - Ejecutar comando del sistema operativo y devolver el resultado.
- `diary` - Guardar el texto de una sesión de MatLab.

C.1.4. Control de la ventana de comandos

- `cedit` - Cambiar la configuración de la edición/historial de la línea de comandos.
- `clc` - Borrar la ventana de comandos.
- `home` - Enviar el cursor al inicio.
- `format` - Setear el formato numérico de salida.
- `echo` - Activar/desactivar el eco de comandos dentro de Scripts.
- `more` - Control de salida por páginas en la ventana de comandos.

C.1.5. Iniciando y terminando MatLab

- `quit` - Terminar la sesión de MATLAB.
- `startup` - Archivo .M que se ejecuta al inicio de MatLab.
- `matlabrc` - Archivo .M maestro de configuración inicial del MatLab.

C.2. Gráficos bidimensionales

C.2.1. Gráficos X-Y elementales

- `plot` - Gráfico Lineal.
- `loglog` - Gráfico logarítmico en ambos ejes.
- `semilogx` - Gráfico semilogarítmico en el eje X.
- `semilogy` - Gráfico semilogarítmico en el eje Y.

C.2.2. Gráficos X-Y especializados

- `polar` - Gráfico en Coordenadas Polares.
- `bar` - Gráfico de Barras.
- `stem` - Gráfico de secuencias discretas.
- `stairs` - Gráfico de Escalera.
- `errorbar` - Gráfico con barras de error.
- `hist` - Histograma.
- `fplot` - Graficación de Funciones implícitas.

C.2.3. Referenciación de gráficos

- `title` - Título del gráfico.
- `xlabel` - Etiqueta de eje X.
- `ylabel` - Etiqueta del eje Y.
- `text` - Anotación de Texto dentro del gráfico.
- `gtext` - Anotación de texto en posición seleccionada por el mouse.
- `grid` - Rejilla.

C.3. Análisis de señales

C.3.1. Operaciones básicas

- `max` - Mayor componente.
- `min` - Menor componente.
- `mean` - Valor medio o promedio.
- `median` - Mediana.
- `std` - Desvío Estándar.
- `sort` - Ordenar en orden ascendente.
- `sum` - Suma de elementos.
- `prod` - Producto de elementos.

C.3.2. Operaciones con vectores

- `cross` - Producto cruz de vectores.
- `dot` - Producto punto de vectores.

C.3.3. Correlación

- `corrcoef` - Coeficientes de Correlación.
- `cov` - Matriz de Covarianza.
- `subspace` - Ángulo entre subespacios.

C.3.4. Filtrado y convolución

- `filter` - Filtrado digital unidimensional.
- `filter2` - Filtrado digital bidimensional.
- `conv` - Convolución y multiplicación de polinomios.
- `conv2` - Convolución bidimensional.
- `deconv` - Deconvolución y división de polinomios.

C.3.5. Transformadas de Fourier

- `fft` - Transformada discreta de Fourier unidimensional.
- `fft2` - Transformada discreta de Fourier bidimensional.
- `ifft` - Transformada discreta de Fourier Inversa unidimensional.
- `ifft2` - Transformada discreta de Fourier Inversa bidimensional.
- `abs` - Magnitud (módulo).
- `angle` - Ángulo de Fase.
- `unwrap` - Remover los saltos de ángulo de fase a través de los límites de 360 grados.
- `fftshift` - Mover la frecuencia cero al centro del espectro.

C.4. Funciones matemáticas elementales

C.4.1. Trigonométricas

- `sin` - Seno.
- `sinh` - Seno Hiperbólico.
- `asin` - Arco Seno.
- `asinh` - Arco Seno Hiperbólico.
- `cos` - Coseno.
- `cosh` - Coseno Hiperbólico.
- `acos` - Arco Coseno.
- `acosh` - Arco Coseno Hiperbólico.
- `tan` - Tangente.
- `tanh` - Tangente Hiperbólica.
- `atan` - Arco Tangente.
- `atan2` - Arco tangente corregida para los 4 cuadrantes.
- `atanh` - Arco Tangente Hiperbólica.
- `sec` - Secante.
- `sech` - Secante Hiperbólica.
- `asec` - Arco Secante.
- `asech` - Arco Secante Hiperbólica.
- `csc` - Cosecante.
- `csch` - Cosecante Hiperbólica.
- `acsc` - Cosecante Inversa.
- `acsch` - Cosecante Hiperbólica Inversa.
- `cot` - Cotangente.
- `coth` - Cotangente Hiperbólica.
- `acot` - Cotangente Inversa.
- `acoth` - Cotangente Inversa Hiperbólica.

C.4.2. Exponenciales

- `exp` - Exponencial.
- `log` - Logaritmo Natural.
- `log10` - Logaritmo base 10.
- `sqrt` - Raíz Cuadrada.

C.4.3. Complejos

- `abs` - Módulo (valor absoluto para reales).
- `angle` - Ángulo de Fase.
- `conj` - Complejo conjugado.
- `imag` - Parte Imaginaria.
- `real` - Parte Real.

C.4.4. Numéricas

- `fix` - Redondear hacia cero.
- `floor` - Redondear hacia $-\infty$.
- `ceil` - Redondear hacia $+\infty$.
- `round` - Redondear hacia el entero más cercano.
- `rem` - Residuo luego de la división entera.
- `sign` - Función signo.

C.5. Matrices elementales y manipulación de matrices

C.5.1. Matrices elementales

- `zeros` - Matriz de ceros.
- `ones` - Matriz de unos.
- `eye` - Matriz Identidad.
- `rand` - Matriz de números aleatorios uniformemente distribuidos.
- `randn` - Matriz de números aleatorios con distribución Normal.
- `linspace` - Vector linealmente espaciado.
- `logspace` - Vector espaciado logarítmicamente.
- `meshgrid` - Genera matrices X e Y para gráficos 3-D.
- `:` - Vector con espaciado regular.

C.5.2. Constantes y variables especiales

- `ans` - Última respuesta.
- `eps` - Precisión de punto flotante.
- `realmax` - Mayor número de punto flotante.
- `realmin` - Menor número positivo de punto flotante.
- `pi` - 3.1415926535897....
- `i`, `j` - Unidad imaginaria.
- `inf` - Infinito.
- `NaN` - Constante utilizada cuando se encuentra una indeterminación (Not-a-Number).
- `flops` - Número de operaciones de punto flotante.
- `nargin` - Número de argumentos de entrada de una función.
- `nargout` - Número de argumentos de salida de una función.
- `version` - Número de Versión de MatLab.

C.5.3. Horas y fechas

- `clock` - Hora.
- `cputime` - Tiempo de CPU pasado.
- `date` - Fecha.
- `etime` - Tiempo pasado desde el inicio.
- `tic`, `toc` - Iniciar o parar las funciones de timer.

C.5.4. Manipulación de matrices

- `diag` - Crear o extraer diagonales.
- `fliplr` - Invertir el orden de los elementos de una matriz de izquierda a derecha.
- `flipud` - Invertir el orden de los elementos de una matriz de arriba hacia abajo.
- `reshape` - Cambiar el tamaño de una matriz.
- `rot90` - Rotar una matriz 90 grados.
- `tril` - Submatriz triangular inferior.
- `triu` - Submatriz triangular superior.
- `:` - Índice genérico de rango de matrices.

C.6. Toolbox de procesamiento de señales

C.6.1. Generación de formas de onda

- `diric` - Función de Dirichlet (Sinc Periódico).
- `sawtooth` - Función diente de sierra.
- `sinc` - Función Sinc ($\sin(\pi*x)/(\pi*x)$).
- `square` - Función rectangular periódica.

C.6.2. Análisis e implementación de filtros

- `abs` - Magnitud.
- `angle` - Ángulo de Fase.
- `conv` - Convolución.
- `fftfilt` - Filtrado en dominio frecuencial.
- `filter` - Filtrado unidimensional.
- `filtfilt` - Filtrado con un filtro de fase cero.
- `filtic` - Determinar Condiciones Iniciales de un filtro.
- `freqs` - Respuesta en frecuencia a partir de la transformada de Laplace.
- `freqspace` - Espaciado en frecuencias para la respuesta en frecuencias.
- `freqz` - Respuesta en frecuencia a partir de la transformada Z.
- `grpdelay` - Retardo de Grupo.
- `impz` - Respuesta al impulso discreta.
- `zplane` - Gráfico de polos y ceros discreto.

C.6.3. Diseño de filtros digitales IIR

- `besself` - Diseño de filtro analógico de Bessel.
- `butter` - Diseño de filtro de Butterworth.
- `cheby1` - Diseño de filtro de Chebyshev tipo I.
- `cheby2` - Diseño de filtro de Chebyshev tipo II.
- `ellip` - Diseño de filtro Elíptico.
- `yulewalk` - Diseño de filtro por el método de Yule-Walker.

C.6.4. Selección del orden de filtros IIR

- `buttord` - Selección del orden de filtros de Butterworth.
- `cheb1ord` - Selección del orden de filtros de Chebyshev tipo I.
- `cheb2ord` - Selección del orden de filtros de Chebyshev tipo II.
- `ellipord` - Selección del orden de filtros Elípticos.

C.6.5. Diseño de filtros FIR

- `fir1` - Diseño de filtros FIR por ventanas — - pasabajos, pasaaltos, pasabanda, rechazabanda.
- `fir2` - Diseño de filtros FIR por ventanas — - Respuesta Arbitraria.
- `firls` - Diseño de filtros FIR — - Respuesta arbitraria con bandas de transición.
- `intfilt` - Diseño de filtros interpoladores.
- `remez` - Diseño de filtros FIR óptimos por método de Parks-McClellan.
- `remezord` - Selección del orden de filtros FIR de Parks-McClellan.

C.6.6. Transformadas

- `dct` - Transformada coseno discreta.
- `dftmtx` - Matriz de la transformada discreta de Fourier.
- `fft` - Transformada rápida de Fourier.
- `fftshift` - Centrar la frecuencia cero.
- `hilbert` - Transformada de Hilbert.
- `idct` - Transformada coseno discreta inversa.
- `ifft` - Transformada rápida de Fourier inversa.

C.6.7. Procesamiento estadístico de señales

- `corrcoef` - Coeficientes de Correlación.
- `cov` - Matriz de Covarianza.
- `csd` - Densidad Espectral Cruzada.
- `psd` - Densidad Espectral de Potencia.
- `xcorr` - Función de Correlación Cruzada.
- `xcov` - Función Covarianza.

C.6.8. Ventanas

- `bartlett` - Ventana de Bartlett.
- `blackman` - Ventana de Blackman.
- `boxcar` - Ventana Rectangular.
- `chebwin` - Ventana de Chebyshev.
- `hamming` - Ventana de Hamming.
- `hanning` - Ventana de Hanning.
- `kaiser` - Ventana de Kaiser.
- `triang` - Ventana Triangular.

C.6.9. Modelado paramétrico

- `levinson` - Método recursivo de Levinson-Durbin.
- `lpc` - Coeficientes de Predicción Lineal usando el método de autocorrelación.

C.6.10. Operaciones especializadas

- `cceps` - Cepstro Complejo.
- `decimate` - Remuestreo a una frecuencia menor.
- `deconv` - Deconvolución.
- `demod` - Demodulación para simulación de comunicaciones.
- `interp` - Remuestreo a una frecuencia mayor.
- `interp1` - Interpolación general unidimensional.
- `medfilt1` - Filtro de Mediana.
- `modulate` - Modulación para simulación de comunicaciones.
- `rceps` - Cepstro Real y reconstrucción de fase mínima.
- `resample` - Remuestreo con una nueva frecuencia de muestreo.
- `specgram` - Espectrograma.
- `spline` - Interpolación cúbica con Splines.

C.6.11. Prototipos de filtros analógicos pasabajo

- `besselap` - Prototipo de filtro de Bessel.
- `buttap` - Prototipo de filtro de Butterworth.
- `cheb1ap` - Prototipo de filtro de Chebyshev tipo I.
- `cheb2ap` - Prototipo de filtro de Chebyshev tipo II.
- `ellipap` - Prototipo de filtro Elíptico.

C.6.12. Traslaciones en frecuencia

- `lp2bp` - Pasabajos a Pasabanda.
- `lp2bs` - Pasabajos a Rechaza-Banda.
- `lp2hp` - Pasabajos a Pasaaltos.
- `lp2lp` - Pasabajos a Pasabajos.

C.6.13. Discretización de filtros

- `bilinear` - Transformación lineal con compensación de fase opcional.
- `impinvar` - Transformación invariante al impulso.

Este documento fue escrito en \LaTeX , compilado con \MiKTeX y editado en \TeXnicCenter . La nueva versión 2008 fue compilada con \teTeX y editada en Kile.
