

UNIVERSIDAD DE OVIEDO

**MASTER EN TECNOLOGÍAS DE LA INFORMACIÓN Y
COMUNICACIONES EN REDES MÓVILES - TICRM**

TESIS DE MASTER

**DESARROLLO DE LA HERRAMIENTA SOFTWARE
“SIGNALS & SYSTEMS THEORY 5.0” PARA EL
ANÁLISIS REAL Y ESPECTRAL DE SISTEMAS**

SARA MENÉNDEZ GARCÍA

JULIO 2012

Índice general

1. INTRODUCCIÓN	1
1.1. Antecedentes del proyecto.	1
1.1.1. El enfoque de la teoría de señales y sistemas.	1
1.1.2. El programa inicial.	2
1.2. Motivación y objetivos.	3
1.3. Fases y metodología de trabajo.	5
1.4. Estructura del documento.	6
1.5. Estructura general del programa.	7
1.5.1. Estructura a nivel de usuario.	7
1.5.2. Estructura interna.	7
1.6. Resumen de las nuevas aportaciones al programa.	9
2. RESUMEN DE LOS PRINCIPALES CONCEPTOS MATEMÁTICOS	11
2.1. Introducción.	11
2.2. Esquema general.	11
2.3. Espacios de señales.	12
2.3.1. Estructura de espacio vectorial para los espacios de señales.	12
2.3.2. Definición del álgebra de los espacios vectoriales.	14
2.3.3. Definición de bases.	15
2.3.4. Representación general en términos de bases/pseudobases.	16
2.3.5. Casos particulares.	17
2.3.6. Bases/Pseudobases ideales.	19
2.3.7. Bases/pseudobases y transformaciones.	22
2.4. Espacios de operadores.	25
2.4.1. Clasificación inicial importante.	25
2.4.2. Caracterización general.	26
2.4.3. Caracterización funcional de los operadores lineales.	27
2.5. Análisis espectral generalizado.	29
2.6. Casos particulares del esquema general.	31
2.6.1. Espacio de funciones complejas de variable real definidas para todo x . . .	32

2.6.2.	Espacio de funciones complejas de variable real periódicas de periodo X_0 .	33
2.6.3.	Espacio de funciones complejas de variable entera definidas para todo n .	34
2.6.4.	Espacio de funciones complejas de variable entera periódicas de periodo N_0 .	35
3.	ESTRUCTURA GENERAL Y FUNCIONALIDADES DEL PROGRAMA	36
3.1.	Introducción a la estructura del programa.	36
3.1.1.	Espacios de partida.	37
3.1.2.	Espacios de operadores.	37
3.2.	Espacios de señales.	38
3.2.1.	Definición del álgebra del espacio.	38
3.2.2.	Definición de las señales del espacio.	39
3.2.3.	Análisis en el dominio espectral.	39
3.3.	Operadores.	43
3.3.1.	Operadores suma y resta.	44
3.3.2.	Operador Multiplicación.	46
3.3.3.	Operador convolución.	51
3.3.4.	Operador correlación.	55
3.3.5.	Operador escalado.	56
3.3.6.	Operador identidad.	59
3.3.7.	Operador desplazamiento.	61
3.3.8.	Operadores transformada de Fourier y su inversa.	64
3.3.9.	Operador transformada de Hilbert y su inversa.	69
3.3.10.	Operador modulador en amplitud.	71
3.4.	Funcionalidades disponibles para la visualización de señales.	74
3.4.1.	Visualización principal	74
3.4.2.	Visualización del conjunto de respuestas al impulso.	76
3.4.3.	Visualización para el conjunto de funciones de transferencia del operador.	77
3.4.4.	Ventana de animación.	79
3.5.	Métodos numéricos existentes.	79
4.	FUNCIONAMIENTO INTERNO DEL SOFTWARE	81
4.1.	Funcionamiento general.	81
4.2.	Herramientas utilizadas para el almacenamiento de datos.	82
4.2.1.	Variables y estructuras de datos globales del espacio de señales continuas aperiódicas definidas para todo x .	82

4.2.2.	Variables y estructuras de datos globales del espacio continuo periódico. . .	87
4.2.3.	Variables y estructuras de datos globales del espacio discreto aperiódico. . .	87
4.2.4.	Variables y estructuras de datos globales del espacio discreto periódico. . .	88
4.2.5.	Variables y estructuras de datos globales del espacio de señales continuas aperiódicas definidas para valores positivos de la variable $x : S(0, \infty)$. . .	88
4.3.	Funcionamiento interno de cada espacio de señales: interfaces.	88
4.3.1.	Interfaces del módulo de señales continuas aperiódicas definidas para todo x	88
4.3.2.	Interfaces del resto de módulos del programa.	106
4.4.	Rutinas principales del software.	107
4.5.	Métodos numéricos.	109
4.6.	Bases de datos generadas.	109
4.6.1.	Base de datos para la representación del conjunto de funciones núcleo que caracterizan a los distintos operadores en el dominio transformado: <code>buildSpectralMatrix</code>	110
5.	MANUAL DE USUARIO	111
5.1.	Presentación y utilidades de programa Signals & Systems Theory v5.0.	111
5.2.	Descripción de las interfaces.	111
5.2.1.	Ventana de inicio.	112
5.2.2.	Menú principal.	112
5.2.3.	Interface de visualización.	121
5.2.4.	Ventana de visualización del conjunto de respuestas al impulso.	125
5.2.5.	Ventana de visualización del conjunto de funciones de transferencia.	128
5.2.6.	Selector de señales.	131
5.2.7.	Selector de los parámetros de una señal.	131
5.2.8.	Selector del rango de definición.	133
5.2.9.	Ventana de animación.	134
5.2.10.	Ventana para la representación compleja.	135
5.2.11.	Ventana de ayuda.	135
5.3.	Ficheros de datos.	137
5.3.1.	Escritura manual de datos en los ficheros.	137
5.4.	Ejemplos de Uso.	139
5.4.1.	Análisis del Operador Multiplicación para el Espacio Continuo Aperiódico.	140
5.4.2.	Análisis del Operador Convolución para el Espacio Continuo Aperiódico.	147
5.4.3.	Análisis del Operador Transformada de Fourier para el Espacio Continuo Aperiódico.	154

5.4.4.	Análisis del Operador Modulador en Amplitud para el Espacio Continuo Periódico.	156
5.4.5.	Análisis de la Operación Suma en el espacio Discreto Aperiódico.	160
6.	CONCLUSIONES Y LÍNEAS FUTURAS	166
6.1.	Conclusiones.	166
6.2.	Líneas futuras.	167
A.	PRESUPUESTO ECONÓMICO	170
A.1.	Medición de recursos.	170
A.2.	Medición de costes.	170
A.3.	Presupuestos parciales.	171
A.4.	Presupuesto de ejecución material.	171
A.5.	Presupuesto de ejecución por contrata.	172
B.	SEÑALES DISPONIBLES EN EL PROGRAMA	173
B.1.	Señales del espacio continuo aperiódico: $S(-\infty, \infty)$	173
B.2.	Señales del espacio continuo periódico: $P(X_0)$	174
B.3.	Señales del espacio discreto aperiódico: $D(-\infty, \infty)$	175
B.4.	Señales del espacio discreto periódico: $D(N_0)$	176
B.5.	Señales del espacio continuo aperiódico definidas para valores positivos: $S(0, \infty)$	177

1. INTRODUCCIÓN

1.1. Antecedentes del proyecto.

La Tesis de Máster que se presenta a continuación, surge con el objetivo principal de estudiar la Teoría Generalizada de Señales y Sistemas, y más en concreto el análisis espectral generalizado de dicha teoría, y desarrollar una herramienta software que permita su análisis de forma didáctica e intuitiva. Este trabajo forma parte de las actividades docente del Máster "Tecnologías de la Información y Comunicaciones Redes Móviles (TICRM)".

Este proyecto surge como una continuación de varios trabajos previos entre los que se encuentra el Proyecto Fin de Carrera del mismo autor. Se centra principalmente en el estudio del análisis espectral de los sistemas lineales no invariantes y, a partir de estos resultados, en la generalización de dicho análisis espectral para todo sistema lineal. El análisis de este tipo de sistemas resulta de gran importancia en el modelado de problemas físicos en el espacio (por ejemplo, problemas de radiación y dispersión electromagnética) que, si bien los sistemas son habitualmente invariantes en el tiempo, no suelen serlo en el espacio.

A continuación se presenta una serie de breves explicaciones para facilitar al lector la comprensión del trabajo presentado en este documento y, más en concreto, de los contenidos de esta introducción.

1.1.1. El enfoque de la teoría de señales y sistemas.

El proyecto expuesto en el presente texto ha consistido, en líneas generales, en el desarrollo de una nueva versión del software **Signals and Systems Theory**. Dicho programa tiene como cometido acercar al usuario, de una forma didáctica e intuitiva, a la teoría de señales y sistemas. Como ya ocurría en versiones previas de este software, la base teórica sobre la que se fundamenta no se corresponde con la forma tradicional de entender dicha teoría sino que hace referencia a un enfoque generalizado de la teoría de señales y sistemas, [1], [2].

Este nuevo enfoque teórico consiste en la generalización de la teoría basada en los conceptos algebraicos de espacios vectoriales y espacios de operadores aplicado al caso de los espacios de funciones (señales en su concepto más general) y de los espacios de operadores operando sobre los espacios de funciones. En este sentido, cabe destacar que dichos espacios, en el caso más general, son de dimensión infinita, con las complicaciones matemáticas que esto conlleva. En relación a este enfoque, se aborda el análisis de los sistemas lineales no invariantes mediante el estudio del conjunto de respuestas al impulso que caracteriza a los operadores que los definen así como su representación gráfica. También se debe resaltar que, fruto de esa generalización, se aborda el concepto de transformación (transformada) desde un punto de vista general donde las transformadas habituales (desarrollos en serie de Fourier, transformadas de Fourier, etc.) se ven como un caso particular de dicha teoría general. Asimismo, y como resultado de lo anterior, surge el concepto de análisis espectral del sistema, gracias al cual se permite estudiar el sistema analizado bajo la transformada seleccionada mediante el estudio del conjunto de funciones de transferencia. Véase el Cap. 2 para una explicación más detallada de este enfoque.

Todos estos aspectos generales se han tenido en cuenta a la hora de desarrollar el software y, muy especialmente, a la hora de pensar en posibles futuras ampliaciones del mismo.

1.1.2. El programa inicial.

El software que presentamos en este proyecto ha sido el resultado de varias ampliaciones y modificaciones aplicadas a la versión anterior del software denominada **Signals & Systems Theory v4.0**. A continuación se presenta de forma resumida la evolución que ha ido sufriendo el software a lo largo de las distintas versiones desarrolladas:

- **Signals & Systems Theory v1.0**, [3]. La versión inicial fue concebida y desarrollada para que su aplicación fuese docente. De esta manera pretendía alcanzar dos propósitos: ayudar al profesor en sus clases, tanto prácticas como teóricas, y mejorar la comprensión por parte de los alumnos de los conceptos explicados en el aula. Esta primera versión de la herramienta ya intentaba incorporar la teoría de señales y sistemas con el enfoque original que se presentó en 1.1.1 estableciendo un nexo entre dicha teoría, en su forma más abstracta, y sus aplicaciones prácticas. Permitiendo así emplear la teoría de señales y sistemas, como la potente herramienta matemática que es, en diversas áreas del conocimiento. A pesar de ello, no lograba dar la idea de la unificación de la teoría general conseguida en versiones posteriores. Su estructura interna se organizaba en torno a los espacios de señales de los que disponía. A cada uno de ellos le correspondía un módulo independiente para cada una de las funcionalidades disponibles en el programa (visor de señales, operaciones básicas, etc.). De este modo para cada espacio de señales se disponía de seis módulos diferentes. Esta distribución modular resultaba práctica a la hora de realizar nuevas ampliaciones (añadiendo un módulo independiente) aunque muy redundante ya que, para su correcto funcionamiento, el programa debía disponer de una copia idéntica de la mayoría de las rutinas dentro de cada módulo.
- **Signals & Systems Theory v2.0**, [4]. La segunda versión del software, desarrollada por el mismo autor que el presente documento, supuso un gran cambio para el programa y fue la que dio el salto definitivo a la implementación de una herramienta fiel a la teoría generalizada de señales y sistemas, aunque implementando exclusivamente sistemas lineales e invariantes. El software evolucionó tanto en apariencia como en estructura interna. Las interfaces se modificaron para atender y exponer de forma clara e intuitiva la teoría de señales y sistemas. A su vez, el programa sufrió una reorganización total agrupando todos los archivos en cuatro módulos independientes, correspondientes a cada espacio de señales implementado. De este modo todas las funcionalidades de cada espacio compartían recursos mejorando la eficiencia del programa. También se añadieron nuevas funcionalidades, como el análisis espectral de sistemas, y se incrementó el número de señales disponibles con las que trabajar, consiguiendo así una herramienta más útil y completa. Adicionalmente, se pretendió aumentar el abanico de usuarios, abarcando desde el público docente hasta investigadores y/o especialistas en la materia.
- **Signals & Systems Theory v3.0**, [5]. La tercera versión se mantuvo fiel a la anterior en cuanto al enfoque teórico y estructura general. Esta versión completó a las anteriores introduciendo un nuevo espacio de señales, con todas las nuevas funcionalidades que ello implica. Este nuevo espacio suponía un nuevo módulo en el software y seguía una estructura similar al resto de espacios. Cabe destacar que una de las principales novedades que permitió incluir fue la **transformada de Bessel**, añadida como operador y como transformada bajo la cual analizar espectralmente los sistemas. Adicionalmente se completó uno de los módulos existentes, el **continuo aperiódico**, implementando el operador **transformada de Hilbert** y el análisis espectral bajo dicha transformada.
- **Signals & Systems Theory v4.0**, [6]. La cuarta y última versión del software, desarrollada por el mismo autor que el presente documento, supuso de nuevo un cambio significativo para el programa incluyendo un análisis más generalizado de la teoría de señales y sistemas e incluyendo la teoría de operadores en el software. De este modo el programa modificó

tanto el interfaz gráfico como su funcionamiento interno incluyendo nuevas funcionalidades muy interesantes como la clasificación de los operadores en función de sus propiedades o la caracterización generalizada de los operadores lineales no invariantes. Hasta la tercera versión, aunque existían operadores lineales no invariantes disponibles en el software, jugaban un papel muy secundario y no permitían el análisis completo del sistema. Es en esta nueva versión en la que trabaja para proveer de esa funcionalidad y permitir al usuario un estudio exhaustivo del sistema en el dominio real.

1.2. Motivación y objetivos.

El presente proyecto ha consistido en el desarrollo y puesta en funcionamiento de una nueva versión del software SST denominada: **Signals & Systems Theory v5.0**, así como en el estudio de la teoría de señales y sistemas sobre la que se construye el programa. Esta nueva actualización incluye como novedad principal el análisis espectral generalizado, incluyendo así en análisis espectral de todos los sistemas lineales, sean éstos invariantes o no. De este modo, y tal y como ocurría en la versión previa del programa, éste ya no se concibe solamente como una herramienta de apoyo en el estudio de la teoría de señales y sistemas, sino también como una herramienta de comprobación de nuevos resultados teóricos desarrollados en las últimas versiones de la teoría generalizada de señales y sistemas.

Desde este punto de vista, el programa cumple cuatro funciones principales:

1. Analiza en el dominio real las transformaciones que sufre una señal cuando atraviesa un sistema. Este sistema estará caracterizado por un operador definido por el usuario. En este caso, el programa clasificará el sistema analizado según sus propiedades, basándose principalmente en las propiedades de linealidad e invarianza. De este modo, el software será capaz de analizar el sistema en el dominio real cuando el operador sea lineal, en caso contrario el operador no se puede caracterizar por un conjunto de respuestas al impulso.
2. Analiza en el dominio espectral las transformaciones que sufre una señal cuando atraviesa un sistema. En este caso, el programa analizará el sistema estudiado en función del operador definido por el usuario y de la transformada sobre la que se realice el análisis en el dominio transformado. Cabe destacar que dicho análisis se realiza de forma generalizada para todos los tipos de sistemas, siempre y cuando éstos sean lineales, representando el conjunto de funciones de transferencia que caracterizan al operador.
3. Realiza operaciones básicas con dos señales, estas son la suma, la resta, el producto y la división. Se debe aclarar que, aunque algunas de estas operaciones también están definidas como operadores, esta funcionalidad del software trata a las dos señales definidas por el usuario como señales de entrada y en ningún caso como señales que son sometidas a las transformaciones impuestas por un operador. Por lo tanto, en este caso no se analizará un operador ni se estudiarán su caracterización y sus propiedades.
4. Visualiza señales (tanto en variable continua como en variable discreta) y determina los parámetros básicos de éstas asociadas a una cierta definición de álgebra de señales específica, expuestos en detalle en la Secc. 3.2.

El lenguaje de programación que se ha utilizado, al igual que en las versiones previas del software, Secc. 1.1, es **MATLAB**, por estar especialmente diseñado para trabajar con expresiones matemáticas, y por la multitud de herramientas sencillas que pone a disposición del programador para diseñar e implementar entornos gráficos interactivos.

Tras analizar la base teórica mencionada en la Secc. 1.1 y realizar una revisión del estado inicial del software, se ha detectado un vacío conceptual que, siguiendo con las líneas futuras

del programa planteadas en las versiones anteriores, ha dado como fruto los siguientes objetivos para este proyecto:

1. **Permitir el estudio de los sistemas lineales no invariantes en el dominio espectral.** A pesar de que ya existiesen operadores lineales no invariantes en las versiones anteriores del software, éste no permitía realizar un análisis espectral del operador, y por tanto un estudio completo del sistema en el dominio transformado. Esta nueva actualización se centra en proveer al usuario de una herramienta que permita realizar dicho estudio a un nivel gráfico, permitiendo **visualizar el conjunto de funciones de transferencia** que caracterizan al operador en el dominio transformado.
2. **Proveer de una herramienta para el estudio generalizado de los sistemas lineales en el dominio espectral.** Si bien es cierto que el análisis espectral de los sistemas lineales invariantes ya estaba disponible en versiones anteriores del programa, mediante la representación de la función de transferencia, se ha decidido modificar el enfoque de esta funcionalidad para adaptarlo al caso más general de los sistemas lineales, sean estos invariantes o no. De este modo, siempre y cuando el operador con el que se esté trabajando sea lineal, el usuario podrá analizar su comportamiento en el dominio transformado de una forma generalizada.
3. Paralelamente a los objetivos anteriores, y para poder cumplirlos, surge la necesidad de **estudiar y comprender la teoría sobre la que se sustenta el programa**, un resumen de dicha teoría se expone en la Secc. 2. Un estudio en profundidad de dicha base teórica se puede encontrar en [2].

A estos objetivos principales se suman otros no menos importantes:

4. Continuar la línea de las versiones anteriores para compartir la metodología, organización y notación con los contenidos en [1] - [2], [5] - [7], permitiendo así facilitar su utilización, así como su posible distribución conjunta. Este aspecto es especialmente relevante ya que la metodología que se presenta en las referencias citadas se aborda desde un punto de vista diferente al que se puede encontrar en la bibliografía habitual, [8] - [9], por ejemplo.
5. Dotarlo de rigor teórico. Dado que en muchas de las señales utilizadas las aproximaciones numéricas no serían nunca suficientes y que algunos de los alumnos que utilicen el programa, en un principio, no tienen un conocimiento profundo de la asignatura, es necesario que los resultados presentados sean los propios resultados teóricos, allí donde esto sea posible. Entendiendo que estos resultados teóricos tendrán siempre como limitación la propia precisión de la máquina utilizada y de los algoritmos de más bajo nivel.
6. Dotarlo de una gran versatilidad de cara a la experimentación. Para ello el programa ha de contar con una serie de métodos numéricos que aumenten así las posibilidades de las que se dispondría en caso de utilizar sólo resultados teóricos.
7. Continuar mejorando la interfaz de usuario de modo que, además de ser intuitiva, sencilla de utilizar y fiel a la que se ha utilizado en las últimas versiones del software, se adapte cada vez más a los conceptos teóricos sobre los que se fundamenta el programa, [1], [2] y [7].
8. Prepararlo para poder añadir futuras ampliaciones de la forma más sencilla posible, Secc. 6.2.
9. Reparar pequeños errores existentes en las versiones anteriores del software.

1.3. Fases y metodología de trabajo.

La metodología de trabajo que seguida para el desarrollo de este proyecto ha seguido una serie de etapas que se repiten de manera cíclica. A continuación se explica cada una de las fases que se han llevado a cabo.

- **Definición de los objetivos a través del estudio de las posibles mejoras y ampliaciones del programa a nivel funcional.** La primera fase del trabajo ha sido el diseño de las ampliaciones que se van a implementar en la nueva versión del software. Como se parte de una herramienta existente, en esta fase no sólo se incluyen nuevas incorporaciones sino que también se plantean las modificaciones que se quieren realizar sobre la versión anterior.
- **Documentación y estudio teórico.** Una vez se han definido los objetivos a los que se aspira, se hace necesaria una segunda fase para estudiar y comprender toda la teoría sobre la que se fundamentan las nuevas ampliaciones a desarrollar y, de este modo, poder implementar el resto de etapas correctamente. En esta fase se estudiarán los documentos y trabajos previos necesarios para el desarrollo de cada ampliación planteada.
- **Diseño del interfaz de usuario.** El diseño del entorno de visualización del programa constituye la tercera etapa de trabajo. Esta fase se aborda una vez que se conocen tanto el objetivo a desarrollar como la base teórica sobre la que se fundamenta. En este punto del proceso se plantean los interfaces necesarios para cubrir las nuevas necesidades del programa. Dichos interfaces se diseñan siempre de la forma más intuitiva y manejable posible incluyendo los gráficos y ayudas que se consideren necesarios para lograr la mayor comprensión y facilidad de uso posibles por parte de los usuarios.
- **Implementación y codificación.** Cuando se termina la fase de diseño, se materializan los bocetos obtenidos en las etapas anteriores. Para desarrollar el interfaz gráfico se utiliza la herramienta GUIDE de MATLAB. A continuación se codifican las rutinas necesarias para implementar las funcionalidades planteadas en la primera fase. El código a su vez, se escribe también en lenguaje MATLAB.
- **Implementación de los métodos de cálculo.** Durante la quinta etapa se obtienen los resultados teóricos que se necesitan sirviéndose del análisis teórico realizado en la segunda fase expuesta en este apartado. Adicionalmente se implementan los métodos numéricos necesarios para aumentar la versatilidad del programa; dichos métodos se explican en detalle en las Seccs. 3.5 y 4.5. Como es lógico, se debe comprobar que estos métodos funcionen correctamente.
- **Detección y corrección de errores.** Esta supone la última fase de trabajo cíclica. Cuando se ha generado todo el código se realiza una batería de pruebas con el objetivo de detectar los posibles fallos que pudieran existir y volver a la fase necesaria para corregirlos.

Las seis fases descritas previamente se repiten cíclicamente hasta que se obtiene un código robusto y que cumple los objetivos marcados. Sin embargo, las tareas a realizar no terminan aquí. Una vez finalizado dicho proceso, aún faltan dos tareas que realizar:

- **Generación del fichero ejecutable.** Tras obtener una versión final del software, habiendo pasado por las fases descritas previamente de forma cíclica las veces necesarias, se compila el código para obtener un fichero ejecutable apto para su distribución.
- **Redacción de la memoria.** Cuando se ha finalizado el desarrollo del software se procede a realizar la última tarea: redactar el presente documento. En la memoria se expone el proceso seguido durante el desarrollo del software y se presenta la base teórica sobre la

que se fundamenta. Asimismo, en este documento se incluye el manual de usuario que acompañará al ejecutable en su distribución.

1.4. Estructura del documento.

El presente documento se estructura en dos partes: el bloque principal, en el que se tratan todos aspectos importantes del proyecto, y un bloque con apéndices.

El bloque principal se divide a su vez en los seis capítulos que se describen a continuación:

1. **Introducción.** En el primer capítulo, en el que está contenido este apartado, se resumen los puntos principales de este proyecto con el objetivo de que el lector pueda hacerse una idea general del trabajo desarrollado y pueda leer el resto del texto con una visión global del proyecto presentado. Entre otras cosas, se resumen los antecedentes, los objetivos y los resultados obtenidos tras la realización de esta Tesis Fin de Máster.
2. **Resumen de los conceptos matemáticos más importantes.** En este capítulo se pretende resumir la teoría generalizada de señales y sistemas sobre la que se fundamenta el software, poniendo énfasis en la teoría matemática dedicada al análisis espectral de los sistemas lineales no invariantes, de modo que el lector pueda comprender las modificaciones y novedades incluidas en esta nueva versión del programa.
3. **Estructura general y funcionalidades del programa.** Una vez planteadas las necesidades detectadas y la base teórica de la que se parte, se presenta la estructura general del programa, en la que se pretende plasmar la teoría estudiada. En este capítulo se tratarán en profundidad las distintas funcionalidades que el software pone a disposición del usuario en esta nueva actualización.
4. **Funcionamiento interno del software.** En el cuarto capítulo del texto se describen las distintas herramientas utilizadas para el desarrollo del software, desde las estructuras para el almacenamiento de la información hasta las bases de datos y métodos numéricos implementados. Este capítulo es muy importante ya que en él se presentan los interfaces desarrollados. Cuando el lector estudie este capítulo deberá tener en cuenta que la eficiencia computacional nunca ha sido uno de los objetivos principales de este proyecto, sino que el software se ha desarrollado como herramienta sobre la que estudiar la Teoría Generalizada de Señales y Sistemas.
5. **Manual de usuario.** Este capítulo se ha diseñado para poder ser distribuido de manera independiente al resto del documento cuando sea necesario, de modo que pueda servir como guía para el usuario. En él se describen todas las interfaces del software y los pasos a seguir para implementar todas las funcionalidades. Además, incluye una serie de ejemplos prácticos muy interesantes en los que se muestran no sólo los pasos a seguir sino también una serie de observaciones teóricas con las que el usuario puede aprender y extraer sus propias conclusiones.
6. **Conclusiones y líneas futuras.** Como capítulo final se incluye un apartado dedicado a las conclusiones obtenidas tras la realización del proyecto y una serie de líneas futuras a desarrollar en versiones futuras del programa.

Finalmente, en los apéndices se incluye el presupuesto económico del proyecto y la librería de señales disponibles en el programa para cada espacio de señales.

1.5. Estructura general del programa.

1.5.1. Estructura a nivel de usuario.

El programa está diseñado atendiendo a la Teoría Generalizada de Señales y Sistemas, de modo que sigue el enfoque descrito en [2], [5], [7], véase la Secc. 1.1. De este modo el software simula, gráficamente, un sistema en el que se diferencian tres partes principales:

1. **Espacio vectorial de entrada.** En este apartado se establece la definición del álgebra del espacio que se utilizará a la hora de medir magnitudes sobre dicho espacio. Son tres las magnitudes utilizadas: el producto escalar, la norma (energía) y la métrica (distancia entre dos señales). En esta zona de la ventana también se encuentran los controles necesarios para la definición de la señal de entrada al sistema. Esta señal será escogida por el usuario de entre las disponibles en las bases de datos del software. Las señales disponibles variarán según el espacio en el que se esté trabajando (véase [2] y [7] para una descripción detallada de los espacios de partida incluidos en el programa).
2. **Sistema definido en forma de operador.** La segunda parte de la ventana está concebida para contener la información del sistema analizado. Será en esta zona donde el usuario escogerá el operador que caracterizará al sistema de entre los disponibles en el programa, la disponibilidad de los distintos operadores puede variar de un espacio de señales a otro. También en este área de la ventana se mostrará la información sobre el operador seleccionado (propiedades, esquema, texto explicativo, etc.). Los operadores disponibles en esta versión del programa se dividen en tres grupos: familias de operadores que necesitan la definición de una señal para ser definidos, familias de operadores que necesitan ajustar ciertos parámetros para definir al operador y operadores concretos. Según el tipo de operador escogido el usuario podrá tener que definir la función o parámetros que lo caractericen. Hay que mencionar que, en los casos en el que el sistema necesite más de una señal de entrada, la señal definida en este apartado se ceñirá al espacio de partida y las definiciones algebraicas de éste. Adicionalmente el usuario podrá escoger, también en esta zona de la ventana, analizar operaciones básicas en lugar de operadores.
3. **Espacio vectorial de salida.** Este espacio no se podrá seleccionar manualmente por el usuario ya que dependerá de las transformaciones que sufra la señal en su paso por el sistema. Cuenta también con su propia definición de álgebra del espacio que será independiente a la del espacio de entrada.

En la fig. 1.1 se muestra un ejemplo de la ventana principal del software cuando el espacio de partida es el continuo aperiódico, $S(-\infty, \infty)$, y se ha seleccionado la familia de operadores convolución. Si se observa el interfaz, en él se pueden diferenciar claramente las distintas partes de un sistema: el espacio vectorial de partida, el sistema definido por un operador y el espacio vectorial de salida. En este ejemplo el sistema está definido por una familia de operadores y, por lo tanto, hasta que no se haya definido el operador concreto que se va a utilizar el sistema no estará totalmente definido.

1.5.2. Estructura interna.

Si bien la estructura a nivel de usuario es la descrita previamente, la estructura interna es totalmente diferente. Ésta se divide en cinco secciones principales que se refieren a cada uno de los espacios de partida disponibles:

1. Espacio vectorial de señales complejas de variable real (variable continua) definidas para todo x (dimensión no numerable): $S(-\infty, \infty)$ en [7].

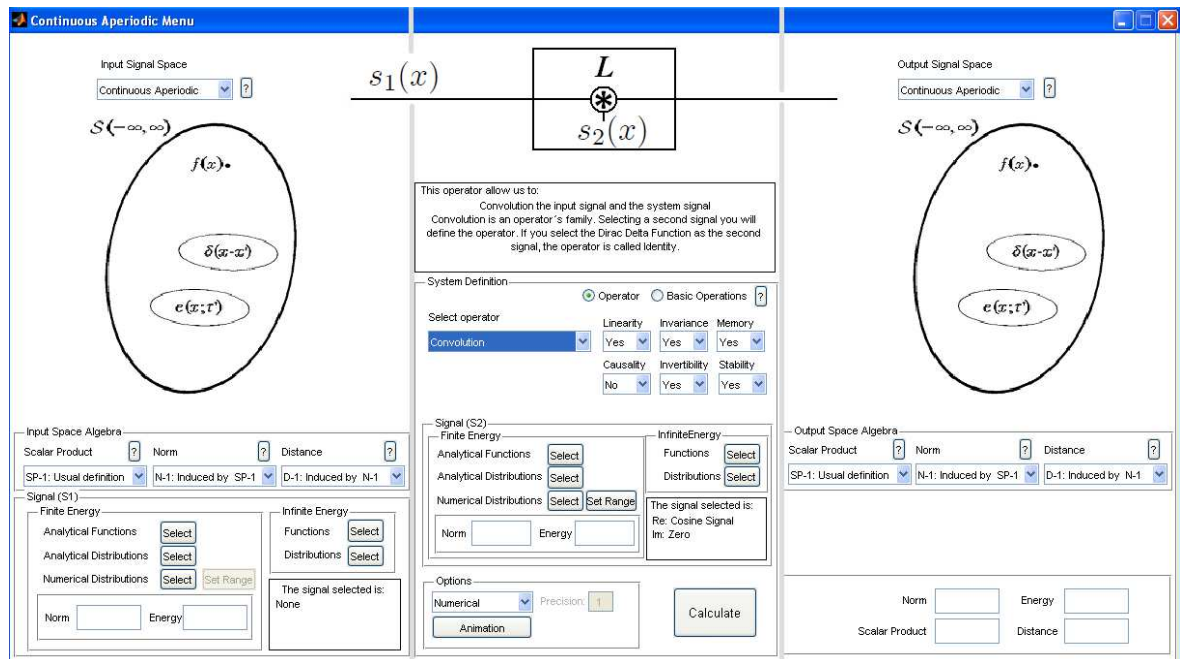


Figura 1.1: Ventana principal del programa cuando el espacio de partida seleccionado es el de señales continuas aperiódicas, $S(-\infty, \infty)$, y se ha escogido la familia de operadores convolución. En este ejemplo se pueden apreciar las tres zonas principales de un sistema: espacio de entrada, espacio de operadores y espacio de salida.

2. Espacio vectorial de señales complejas de variable real (variable continua) periódicas de periodo X_0 (dimensión numerable): $P(X_0)$ en [7].
3. Espacio vectorial de señales complejas de variable entera (variable discreta) definidas para todo n (dimensión numerable): $D(-\infty, \infty)$ en [7].
4. Espacio vectorial de señales complejas de variable entera (variable discreta) periódicas de periodo N_0 (dimensión nfinite): $D(N_0)$ en [7].
5. Espacio vectorial de señales complejas de variable real (variable continua) definidas para valores positivos de x (dimensión no numerable): $S(0, \infty)$ en [5].

Para cada uno de estos apartados subyace la misma base matemática general descrita, resumida en la Secc.1.1.1, con aspectos, definiciones, análisis e interpretaciones diferentes para cada caso. Por lo tanto, las definiciones de álgebra del espacio y de las señales son diferentes así como el modo de realizar las distintas operaciones que se proponen en el programa. Es decir, todas las rutinas que se implementan en lenguaje MATLAB variarán sensiblemente según la sección en que nos encontremos.

Dentro de cada sección (según el espacio vectorial seleccionado) el programa ofrece las siguientes funcionalidades:

- Análisis de operaciones básicas entre dos señales: suma, resta, multiplicación y división.
- Análisis de operadores generales: suma, resta, multiplicación, escalado, convolución, correlación, identidad y desplazamiento.
- Análisis de operadores moduladores: modulador en amplitud.
- Análisis de operadores transformadas y su transformadas inversas, en concreto: la transformada de Hilbert de variable continua en $S(-\infty, \infty)$, la transformada de Bessel de variable

continua en $S(0, \infty)$, la transformada de Fourier de variable continua en $S(-\infty, \infty)$, el desarrollo de Fourier de variable continua en $P(X_0)$, la transformada de Fourier de variable discreta en $D(-\infty, \infty)$ y el desarrollo en serie de Fourier de variable discreta en $D(N_0)$ según se parta del espacio $S(-\infty, \infty)$, $P(X_0)$, $D(-\infty, \infty)$, $D(N_0)$ o $S(0, \infty)$.

- Selección de la definición de los parámetros del álgebra: producto escalar, norma y distancia dentro del espacio seleccionado de las distintas señales así como su cálculo. El cálculo de la energía se realiza a partir de dichas definiciones.
- Visualización de señales.
- Visualización del conjunto de respuestas al impulso para sistemas lineales.
- Análisis espectral del sistema: se utilizarán los análisis espectrales que caracterizan al sistema en función de las transformadas descritas en el punto anterior. Destaca como novedad principal de esta versión del programa el análisis espectral del sistema cuando éste sea lineal no invariante, hasta ahora esta funcionalidad sólo estaba disponible para los sistemas lineales invariantes. Se debe resaltar asimismo, que esta funcionalidad permite el análisis generalizado de los sistemas lineales, independientemente a su invarianza, cambiando sensiblemente la forma de realizar este análisis para los casos ya implementados en versiones anteriores. Hay que mencionar que esta actualización sólo permite el análisis espectral generalizado bajo la Transformada de Fourier o Desarrollo en Serie de Fourier, según el espacio de partida seleccionado, y que se deja como una línea futura el desarrollo de esta funcionalidad bajo otras transformadas, como Hilbert o Bessel, véase la Secc. 6.2.
- Animación del sistema para los operadores: multiplicación, convolución y correlación.

Nótese cómo, dependiendo del espacio de partida seleccionado, la implementación de dichas funcionalidades será diferente, tal y como se mencionó previamente.

Como se puede observar en las explicaciones anteriores este programa pone gran énfasis en la base teórica y algebraica que subyace. Por ello cabe destacar que en su desarrollo se ha dado más importancia al hecho de mostrar dichos conceptos que a la propia programación y eficacia numérica del software.

1.6. Resumen de las nuevas aportaciones al programa.

En este apartado se resumen las ampliaciones que se han añadido a la versión anterior y que unidas a ella integran la herramienta software **Signals & Systems Theory v5.0**. Puesto que el objetivo principal de esta tesis en cuanto al software ha sido desarrollar la funcionalidad del análisis espectral del sistema para los sistemas lineales no invariantes, las nuevas aportaciones están íntimamente relacionadas con dicho objetivo.

- **Nueva funcionalidad para el análisis espectral de sistemas lineales no invariantes bajo la transformada de Fourier.** Esta funcionalidad se ha desarrollado para los módulos del programa correspondientes a los espacios de señales de partida Continuo Aperiódico y Discreto Aperiódico, se entiende que se ha realizado el estudio bajo la transformada de Fourier para el primer espacio y bajo la transformada discreta de Fourier en el segundo. Se han creado y adaptado los métodos numéricos y bases de datos necesarias para el cálculo del conjunto de funciones de transferencia de los operadores lineales invariantes de estos espacios.
- **Nueva funcionalidad para el análisis espectral de sistemas lineales no invariantes bajo el desarrollo en serie de Fourier.** Esta funcionalidad se ha desarrollado para los módulos del programa correspondientes a los espacios de partida Continuo Periódico y Discreto Periódico. Al igual que en el caso anterior, se han modificado y creado las

rutinas y métodos necesarios para el cálculo del conjunto de funciones de transferencia que representan a los distintos operadores lineales no invariantes en el dominio transformado.

- **Ventana de visualización del conjunto de funciones de transferencia.** El análisis espectral del sistema se presenta en el software como una funcionalidad principalmente gráfica. De este modo, se ha visto necesario el desarrollo de un nuevo interfaz gráfico que permita la correcta visualización de este conjunto de señales de transferencia. Esta nueva ventana muestra en tres dimensiones la parte real y la parte imaginaria (o módulo y fase) del conjunto de funciones de transferencia y en dos dimensiones los cortes deseados de la superficie 3D representada.
- **Generalización del análisis espectral del sistema de modo que incluya en el método de análisis general al caso particular de los sistemas lineales invariantes.** Aprovechando el desarrollo del interfaz descrito anteriormente y cumpliendo con el objetivo de plasmar la teoría generalizada de sistemas y señales en el software se ha decidido ampliar el alcance de los operadores que puedan aprovecharse de dicha funcionalidad a todos los que sean lineales, independientemente a su invarianza.

2. RESUMEN DE LOS PRINCIPALES CONCEPTOS MATEMÁTICOS

2.1. Introducción.

Como ya se ha comentado en la Secc. 1.1.1, la teoría de señales y sistemas a partir de la cual se desarrolla el programa está enfocada tomando como base los conceptos matemáticos asociados al álgebra de los espacios vectoriales de señales así como a la teoría de operadores.

Aunque se puede obtener una explicación detallada en [2] y [7], en este capítulo se intentan concentrar y resumir los conceptos principales sobre los que se sustenta la teoría de señales y sistemas. Estos conceptos se consideran fundamentales para que el lector pueda comprender la estructura y diseño del software objeto del presente proyecto.

2.2. Esquema general.

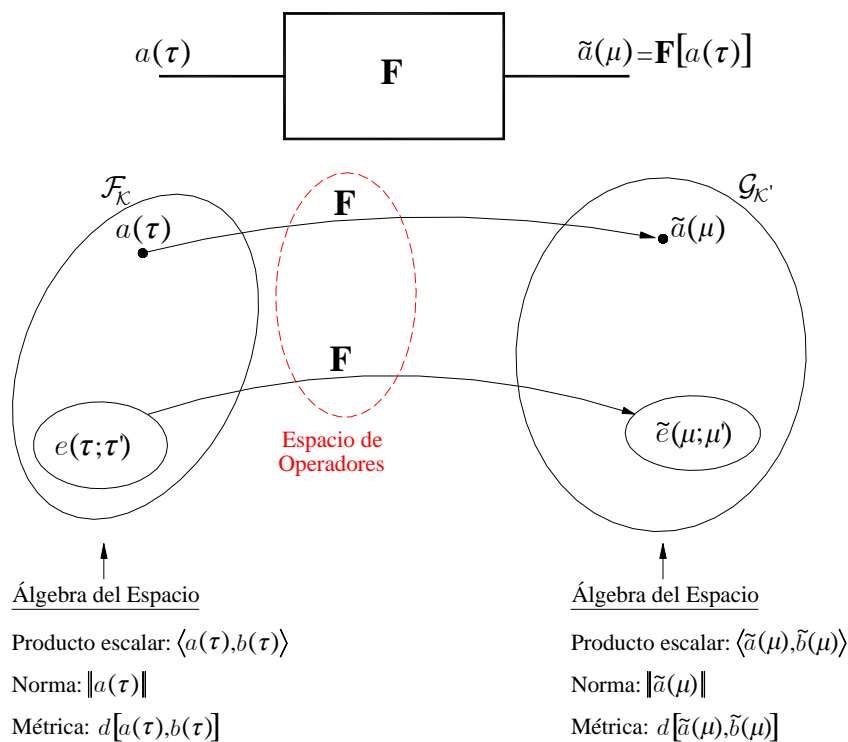


Figura 2.1: Representación genérica de un sistema en el que la transformación de las señales viene descrita por un operador \mathbf{F} entre dos espacios de señales $\mathcal{F}_{\mathcal{K}}$ y $\mathcal{G}_{\mathcal{K}'}$. En el esquema se muestra cómo las variables de las señales pertenecientes a los espacios de partida y de llegada podrían ser, en principio, diferentes, así como el álgebra asociada a cada espacio. Se muestra también la notación que se usará (Secc. 2.3) para describir un subespacio o conjunto de señales dentro de los espacios de señales (variables τ' y μ' en esta descripción general).

El programa ha sido diseñado en torno a la teoría de señales y sistemas. En la Fig. 2.1, se presenta un esquema que pretende exponer esta teoría en la forma más general posible. En ella

se representa la transformación que sufre una señal cuando atraviesa un sistema. De este modo partiendo de una señal de entrada $a(\tau)$ a la salida se obtiene otra $\tilde{a}(\mu)$ (de igual o distinta naturaleza que la señal de entrada) que no es más que la señal original transformada según el operador \mathbf{F} . Así se define el concepto de un sistema como cualquier proceso que intervenga en la transformación de señales.

Del mismo modo, se puede observar que la teoría de señales está presente ya que tanto las señales de entrada al sistema como las que se obtienen a su salida pueden tener una naturaleza independiente y estar dotadas cada una de ellas de la estructura y propiedades del espacio vectorial al que pertenecen.

A lo largo del presente capítulo se expondrán y resumirán los conceptos mostrados en el esquema general, con el objetivo de que el lector se haga una buena idea de la base matemática que sustenta el software *Signals & Systems Theory v4.0*.

2.3. Espacios de señales.

2.3.1. Estructura de espacio vectorial para los espacios de señales.

Para poder establecer y describir la estructura de un espacio vectorial, primero se debe definir el conjunto de elementos que componen dicho espacio, esto es, el espacio de señales sobre los que se operará. Dicho espacio de señales estará compuesto, como es lógico, por señales o funciones:

$$\mathcal{F} = \{a(\tau), b(\tau), c(\tau)\dots\},$$

donde τ representará la variable sobre las que están descritas las funciones de \mathcal{F} , es decir, el dominio de dichas funciones. Por lo tanto, existen distintos espacios de señales dependiendo de la naturaleza de la variable τ . A continuación se describen algunos ejemplos:

- Si $\tau \in \mathbb{Z}$, entonces \mathcal{F} representará un espacio de funciones de variable entera, es decir, el conjunto de señales que compone el espacio sólo estará definido para valores enteros. Un ejemplo de este tipo de funciones podría ser $x(n)$, donde $n \in \mathbb{Z}$ y representa los instantes temporales del muestreo de una señal.
- Cuando $\tau \in \mathbb{R}$, el espacio de señales estaría compuesto de funciones de variable real. Existen numerosos ejemplos para este espacio de funciones. Uno de los ejemplos típico podría ser $x(t)$, donde $t \in \mathbb{R}$ representaría el tiempo. Otro ejemplo podrían ser las señales $f(x)$, donde $x \in \mathbb{R}$ representaría la distancia.
- Si $\tau \in \mathbb{R}^+$, \mathcal{F} contendrá señales de variable real pero definidas únicamente para valores positivos. Este tipo de espacio no podría contener señales periódicas. En un sentido estricto, este sería el espacio que contendría los ejemplos descritos en el punto anterior, pues no existen físicamente distancias o instantes temporales negativos, sin embargo, la teoría de señales puede tomar como instante inicial un instante de referencia que haría las veces de $t_0 = 0$ o una posición de referencia que se denotaría como $x_0 = 0$. En el supuesto descrito sí se podrían dar distancias y tiempos negativos.
- En el caso de que $\tau \in [a, b]$ o (a, b) , donde $a, b \in \mathbb{R}$, entonces \mathcal{F} representaría un espacio de señales de variable real definido para un dominio acotado, es decir, entre a y b . Al igual que en el caso anterior este espacio no podría contener señales periódicas.
- Si $\tau \in \mathbb{R}^3$, el espacio de señales \mathcal{F} contendrá funciones de variable real tridimensional, es decir, el dominio de las señales sería un espacio de tres dimensiones. En este tipo de espacios de señales se podría dar el caso de que las señales sean periódicas o aperiódicas en alguna

o todas las variables. Un ejemplo claro es del espacio $f(\vec{r})$, donde $\vec{r} = (x, y, z) \in \mathbb{R}^3$ representaría un punto del espacio euclídeo tridimensional.

- Cuando $\tau \in \mathbb{C}$, \mathcal{F} representaría al espacio de funciones de variable compleja, es decir, dichas señales estarían definidas sobre los número complejos. Este tipo de espacios se utilizan para el análisis de problemas de radiación y dispersión electromagnética.

Una vez que se ha definido un cierto espacio de señales, $\mathcal{F} = \{a(\tau), b(\tau), c(\tau)\dots\}$, con el que se quiere trabajar se le debe dotar de una estructura de espacio vectorial respecto a un cierto cuerpo de escalares \mathcal{K} . De este modo, los elementos del espacio inicial $\{a(\tau), b(\tau)\dots\}$ podrán ser analizados como vectores $\{\vec{a}, \vec{b}\dots\}$ que serán funciones del conjunto \mathcal{F} definido inicialmente. El espacio resultante será un espacio vectorial sobre un cuerpo \mathcal{K} de escalares, que denotaremos como $\mathcal{F}_{\mathcal{K}}$ a lo largo de todo el proyecto.

Las dos propiedades asociadas a un espacio vectorial que se deberán definir adecuadamente serán las leyes de operación (i) interna y (ii) externa respecto a \mathcal{K} , esto es:

- (i): La suma de vectores: $\vec{a} + \vec{b} \equiv a(\tau) + b(\tau) = h(\tau)$.
(ii): El producto por un escalar: $\alpha \vec{a} \equiv \alpha a(\tau) = p(\tau)$, con $\alpha \in \mathcal{K}$.

En relación con la estructura de espacio vectorial, cabe destacar los siguientes aspectos a modo de resumen:

1. Con la adecuada definición de estas operaciones el resultado de ambas da lugar a nuevos elementos $h(\tau)$ y $p(\tau)$ pertenecientes al espacio vectorial $\mathcal{F}_{\mathcal{K}}$ de origen.
2. Se define el concepto de independencia lineal entre dos vectores. Se establecerá entonces que dos vectores son linealmente independientes cuando se cumpla que:¹

$$\alpha a(\tau) + \beta b(\tau) = 0(\tau) \Leftrightarrow \alpha = \beta = 0.$$

3. Definidas estas operaciones, ya es posible construir combinaciones lineales de cualesquiera M vectores del espacio, esto es,

$$\{a_m(\tau)\}_M \longrightarrow \sum_{m=1}^M \alpha_m a_m(\tau) = a(\tau) \in \mathcal{F}_{\mathcal{K}} \quad (2.1)$$

y generalizar a esas M funciones el concepto de independencia lineal,

$$\{a_m(\tau)\}_M \text{ linealmente independientes} \longrightarrow \sum_{m=1}^M \alpha_m a_m(\tau) = 0(\tau) \Leftrightarrow \alpha_m = 0 \forall m. \quad (2.2)$$

4. Aparece también el concepto de base del espacio vectorial de funciones que se abordará con más detalle en la Secc. 2.3.3.

Teniendo en cuenta todo lo anterior, se puede concluir que cualquier problema que tenga como base un espacio de señales se puede geometrizar del mismo modo que si fuese un espacio vectorial.

¹En este caso, el elemento $0(\tau)$ a la derecha de la igualdad representaría a cualquier función que, desde el punto de vista de la métrica (Secc. 2.3.2) represente el elemento neutro del espacio de funciones.

2.3.2. Definición del álgebra de los espacios vectoriales.

Una vez se haya establecido el espacio vectorial de funciones $\mathcal{F}_{\mathcal{K}}$ se puede definir una serie de parámetros algebraicos con los que estudiar la relación entre los distintos elementos del espacio: la métrica, la norma y el producto escalar.

Métrica. Se utiliza para diferenciar dos elementos del conjunto. Permite medir las distancias existentes entre los distintos elementos del conjunto. La distancia considerada será una aplicación que asocie a dos elementos del espacio un valor real y positivo, esto es,

$$\left\{ \begin{array}{l} d : \mathcal{F}_{\mathcal{K}} \times \mathcal{F}_{\mathcal{K}} \rightarrow \mathbb{R}^+, \\ \alpha = d[a(\tau), b(\tau)], \end{array} \right\}, a(\tau), b(\tau) \in \mathcal{F}_{\mathcal{K}} \text{ y } \alpha \in \mathbb{R}^+. \quad (2.3)$$

La métrica determinará el criterio establecido para discernir si dos funciones cualesquiera del espacio son iguales o diferentes. Lo más habitual será seleccionar como métrica particular del espacio la inducida por una norma definida en el espacio. La métrica determinará, por tanto, la definición del elemento neutro $0(\tau)$ comentada previamente.

Norma. Es la aplicación que permite medir el tamaño de los elementos del espacio. Se definirá una norma válida dando lugar a un espacio vectorial de señales normado,

$$\left\{ \begin{array}{l} \|\cdot\| : \mathcal{F}_{\mathcal{K}} \rightarrow \mathbb{R}^+, \\ \alpha = \|a(\tau)\| \end{array} \right\}, a(\tau) \in \mathcal{F}_{\mathcal{K}} \text{ y } \alpha \in \mathbb{R}^+. \quad (2.4)$$

Una vez definida, será posible seleccionar como métrica particular del espacio la inducida por dicha norma,

$$d[a(\tau), b(\tau)] = \|a(\tau) - b(\tau)\|.$$

Producto escalar. Por último se definirá la aplicación del producto escalar de funciones, que permita introducir las ventajas asociadas al concepto de proyección de un elemento sobre otro del espacio, de la forma,

$$\left\{ \begin{array}{l} \langle \cdot, \cdot \rangle : \mathcal{F}_{\mathcal{K}} \times \mathcal{F}_{\mathcal{K}} \rightarrow \mathbb{C}, \\ \alpha = \langle a(\tau), b(\tau) \rangle \end{array} \right\}, a(\tau), b(\tau) \in \mathcal{F}_{\mathcal{K}} \text{ y } \alpha \in \mathbb{C}. \quad (2.5)$$

Definido un producto escalar se puede asegurar que la norma inducida por el producto escalar es también válida,

$$\|a(\tau)\| = \langle a(\tau), a(\tau) \rangle^{\frac{1}{2}}. \quad (2.6)$$

De las igualdades (2.3) y (2.6) se obtiene la siguiente métrica a partir del producto escalar,

$$d[a(\tau), b(\tau)] = \|a(\tau) - b(\tau)\| = \langle a(\tau) - b(\tau), a(\tau) - b(\tau) \rangle^{\frac{1}{2}}. \quad (2.7)$$

Considerando los resultados anteriores, cabe destacar las siguientes conclusiones importantes:

1. El valor complejo asignado a dos funciones cualesquiera por dicha el producto escalar mide la proyección de la primera sobre la segunda.
2. Si su valor es nulo se dice que las funciones son ortogonales: $\langle a(\tau), b(\tau) \rangle = 0 \rightarrow a(\tau) \perp b(\tau)$.
3. Se podrán considerar como bases ortogonales aquellas en las que el producto escalar entre cada dos elementos entre sí sea nulo. Si además la norma de los elementos de la base es la unidad se denotará como base ortonormal.

2.3.3. Definición de bases.

Como se ha mencionado previamente, una vez se dota al espacio de señales de una estructura vectorial aparecen conceptos como el de combinación lineal que lleva directamente al de base del espacio. Se denota así al conjunto de N de elementos $\{e_m(\tau)\}_N$ del espacio $\mathcal{F}_{\mathcal{K}}$ linealmente independientes tales que todas las combinaciones lineales posibles entre ellos dan lugar a todo el espacio $\mathcal{F}_{\mathcal{K}}$. El número de elementos N de dicho conjunto será la dimensión del espacio vectorial,

$$\{e_m(\tau)\}_N \longrightarrow a(\tau) = \sum_{m=1}^M \alpha_m e_m(\tau), \alpha_m \in \mathcal{F}_{\mathcal{K}}. \quad (2.8)$$

Nótese que cuando la dimensión del espacio sea **infinita** el número de elementos de cualquier base necesarios para representar dicho espacio será infinito, esto es, $\{e_m(\tau)\}_N \rightarrow \{e_m(\tau)\}$. En estas condiciones pueden desaparecer conceptos como el de generación de todo el espacio o incluso puede que los elementos de la base no pertenezcan al espacio original. Así podemos hablar de una aplicación similar a una base de dimensión finita pero de características especiales, dando lugar al concepto de **pseudobase**.

De este modo podemos dividir las bases en dos grupos:

- Bases (en dimensión finita).
- Pseudobases (en dimensión infinita).

Con estos conceptos en mente, denotaremos por $e(\tau; \mu)$ a un conjunto de funciones base o pseudobases donde τ' será la variable que describe al conjunto (finito o infinito) de funciones base.

Hay que mencionar que el nuevo parámetro introducido μ no es una nueva variable, ya que las funciones base siguen estando definidas en una única variable, sino que se trata de un parámetro que se utiliza para identificar cada uno de los elementos de la base. De esta forma, el parámetro τ' puede pertenecer a cualquier dominio, sea éste discreto, finito o infinito, o continuo. Por lo tanto, la expresión $e(\tau; \mu_0)$ identifica a la función $\mu = \mu_0$ del conjunto $e(\tau; \mu)^2$.

Por otra parte, existen dos conceptos relacionados con las bases que se deben resaltar:

1. Para un mismo espacio de señales, existen infinidad de conjuntos que pueden ejercer de base, sin embargo son importantes las bases cuyos elementos sean ortogonales entre sí.
2. Aunque pueda ser evidente, la elección de una base u otra hará que cambien los coeficientes (coordenadas para un espacio vectorial) de un mismo elemento del espacio de señal $\mathcal{F}_{\mathcal{K}}$.

²En dimensión finita, la notación habitual sería $e_n(\tau)$, equivalente en nuestro caso a $e(\tau; n)$.

2.3.4. Representación general en términos de bases/pseudobases.

Un elemento cualquiera del espacio $a(\tau) \in \mathcal{F}_K$ se podrá representar como una combinación lineal de los elementos de un cierto conjunto de funciones (bien en términos de bases o pseudobases, según el espacio). Una representación general de este concepto se podría escribir de la forma siguiente,

$$a(\tau) = \mathbf{CL}_\mu[\alpha(\mu)e(\tau; \mu)]. \quad (2.9)$$

Nótese que con \mathbf{CL} estamos denotando un operador genérico que representa la combinación lineal de elementos $e(\tau; \mu)$ con los coeficientes $\alpha(\mu)$. Su expresión final dependerá del espacio \mathcal{F}_K considerado, así como de las definiciones de las operaciones interna y externa establecidas sobre dicho espacio.

El cálculo de los coeficientes $\alpha(\mu)$ vendrá determinado por el análisis de la distancia entre el elemento a representar y el resultado de la combinación lineal expresada en (2.9) y que deberá ser cero o tender a cero (situaciones habituales en dimensión finita e infinita, respectivamente),

$$d\{a(\tau), \mathbf{CL}_\mu[\alpha(\mu)e(\tau; \mu)]\} \rightarrow 0. \quad (2.10)$$

Como ya se ha expuesto, el desarrollo y estudio de la expresión (2.10) nos permitirá obtener la expresión de los coeficientes $\alpha(\mu)$. Su expresión en dimensión finita vendrá dada por la proyección del elemento original a representar sobre cada uno de los elementos de la base. Este resultado se extenderá, habitualmente, al caso de dimensión infinita, esto es,

$$\alpha(\mu) = \langle a(\tau), e(\tau; \mu) \rangle. \quad (2.11)$$

Las ecuaciones (2.9), (2.10) y (2.11) llevan a la generalización del concepto de transformada que puede ser particularizado a casos concretos. Se presenta a continuación un ejemplo que permita entender mejor estos conceptos.

Ejemplo. Si $\tau \equiv x \in \mathbb{R}$ (variable continua) y $\mu \equiv \xi \in \mathbb{R}$ (variable espectral) se puede considerar la pseudobase $e(\tau; \mu) = e(x; \xi) = e^{j\xi x}$. Así un elemento arbitrario del espacio será $a(\tau) \equiv f(x)$ y el producto escalar habitual vendrá definido por,

$$\langle a(\tau), b(\tau) \rangle = \langle f(x), g(x) \rangle = \int_{-\infty}^{\infty} f(x)g^*(x)dx. \quad (2.12)$$

Esto nos llevaría a la descripción particular de la transformada de Fourier mediante el cálculo de los coeficientes,

$$F(\xi) = \langle f(x), e^{j\xi x} \rangle = \int_{-\infty}^{\infty} f(x)e^{-j\xi x}dx, \quad (2.13)$$

y también a la descripción de la transformada inversa a través de la combinación lineal,

$$f(x) = \mathbf{CL}_\xi[F(\xi)e^{j\xi x}] = \int_{-\infty}^{\infty} F(\xi)e^{j\xi x}d\xi, \quad (2.14)$$

donde $F(\xi) = \frac{1}{2\pi}\alpha(-x)$.

2.3.5. Casos particulares.

Funciones aperiódicas de variable continua.

- Definiciones del espacio

Variable: $\tau \equiv x \in \mathbb{R}$: Variable real.

Espacio vectorial: $\mathcal{F}_{\mathcal{K}} \equiv S(-\infty, \infty)$ Espacio vectorial de funciones complejas de var. continua.

Elemento arbitrario: $a(\tau) \equiv f(x)$.

- Definiciones algebraicas:

Producto escalar: $\langle f(x)g(x) \rangle = \int_{-\infty}^{\infty} f(x)g^*(x)dx$

Norma: $\|f(x)\|^2 = \int_{-\infty}^{\infty} |f(x)|^2 dx$

Métrica: $d^2[f(x), g(x)] = \int_{-\infty}^{\infty} |f(x) - g(x)|^2 dx$

- Bases/Pseudobases: $e(\tau; \mu) \equiv e(x; \mu)$ donde μ es la variable descriptiva del conjunto según la cual:

$$\begin{aligned} \mu \equiv m \in \mathbb{Z} : \text{ discreta} &\rightarrow e(x; m) : \text{ conjunto discreto} \\ \mu \equiv x' \in \mathbb{R} : \text{ continua} &\rightarrow \left\{ \begin{array}{l} e(x; x') : \text{ conjunto continuo} \\ (x \text{ y } x' \text{ son de idéntica naturaleza}) \end{array} \right\} \\ \mu \equiv \xi \in \mathbb{R} : \text{ continua} &\rightarrow \left\{ \begin{array}{l} e(x; \xi) : \text{ conjunto continuo} \\ (\xi \text{ y } x \text{ son de distinta naturaleza}) \end{array} \right\} \end{aligned}$$

A partir de la ecuación (2.9) y las bases definidas según la variable descriptiva del conjunto μ se pueden obtener las siguientes expresiones para el operador **CL**:

$$\mu \equiv m \rightarrow f(x) = \sum_{m=-\infty}^{\infty} \alpha(m)e(x; m) \quad (2.15)$$

$$\mu \equiv x' \rightarrow f(x) = \int_{-\infty}^{\infty} \alpha(x')e(x; x')dx' \quad (2.16)$$

$$\mu \equiv \xi \rightarrow f(x) = \int_{-\infty}^{\infty} \alpha(\xi)e(x; \xi)d\xi \quad (2.17)$$

Funciones periódicas de variable continua.

- Definiciones del espacio:

Variable: $\tau \equiv x \in \mathbb{R}$: Variable real.

Espacio vectorial: $\mathcal{F}_{\mathcal{K}} \equiv P(X_0)$ Espacio vectorial de funciones complejas de var. continua.

Elemento arbitrario: $a(\tau) \equiv f_0(x)$ de periodo X_0 .

- Definiciones algebraicas:

Producto escalar: $\langle f_0(x), g_0(x) \rangle = \int_{\langle X_0 \rangle} f_0(x)g_0^*(x)dx$

Norma: $\|f_0(x)\|^2 = \int_{\langle X_0 \rangle} |f_0(x)|^2 dx$

Métrica: $d^2[f_0(x), g_0(x)] = \int_{\langle X_0 \rangle} |f_0(x) - g_0(x)|^2 dx$

- Bases/Pseudobases: $e(\tau; \mu) \equiv e(x; \mu)$ donde μ es la variable descriptiva del conjunto según la cual:

$\mu \equiv m \in \mathbb{Z} : \text{discreta} \rightarrow e(x; m) : \text{conjunto discreto}$

A partir de la ecuación (2.9) y la base definida se puede obtener la siguiente expresión para el operador **CL**:

$$\mu \equiv m \rightarrow f_0(x) = \sum_{m=-\infty}^{\infty} \alpha(m)e(x; m) \quad (2.18)$$

Funciones aperiódicas de variable discreta.

- Definiciones del espacio:

Variable: $\tau \equiv n \in \mathbb{Z} : \text{Variable entera.}$

Espacio vectorial: $\mathcal{F}_{\mathcal{K}} \equiv D(-\infty, \infty)$ Espacio vectorial de funciones complejas de var. discreta.

Elemento arbitrario: $a(\tau) \equiv x(n).$

- Definiciones algebraicas:

Producto escalar: $\langle x(n), y(n) \rangle = \sum_{n=-\infty}^{\infty} x(n)y^*(n)$

Norma: $\|x(n)\|^2 = \sum_{n=-\infty}^{\infty} |x(n)|^2$

Métrica: $d^2[x(n), y(n)] = \sum_{n=-\infty}^{\infty} |x(n) - y(n)|^2$

- Bases/Pseudobases: $e(\tau; \mu) \equiv e(n; \mu)$ donde μ es la variable descriptiva del conjunto según la cual:

$$\mu \equiv m \in \mathbb{Z} : \text{discreta} \rightarrow \left\{ \begin{array}{l} e(n; m) : \text{conjunto discreto} \\ (n \text{ y } m \text{ de la misma naturaleza)} \end{array} \right\}$$

$\mu \equiv \xi \in \mathbb{R} : \text{continua} \rightarrow e(n; \xi) : \text{conjunto continuo}$

A partir de la ecuación (2.9) y las bases definidas según la variable descriptiva del conjunto μ se obtienen las siguientes expresiones para el operador **CL**:

$$\mu \equiv m \rightarrow x(n) = \sum_{m=-\infty}^{\infty} \alpha(m)e(n; m) \quad (2.19)$$

$$\mu \equiv \Omega \rightarrow x(n) = \int_{\langle 2\pi \rangle} \alpha(\Omega)e(n; \Omega)d\Omega \quad (2.20)$$

Funciones periódicas de variable discreta.

- Definiciones del espacio:

Variable: $\tau \equiv n \in \mathbb{Z}$: Variable entera.

Espacio vectorial: $\mathcal{F}_{\mathcal{K}} \equiv D(N_0)$ Espacio vectorial de funciones complejas de var. discreta.

Elemento arbitrario: $a(\tau) \equiv x_0(n)$ de periodo N_0 .

- Definiciones algebraicas:

Producto escalar: $\langle x_0(n), y_0(n) \rangle = \sum_{n=\langle N_0 \rangle} x_0(n)y_0^*(n)$

Norma: $\|x_0(n)\|^2 = \sum_{n=\langle N_0 \rangle} |x_0(n)|^2$

Métrica: $d^2[x_0(n), y_0(n)] = \sum_{n=\langle N_0 \rangle} |x_0(n) - y_0(n)|^2$

- Bases/Pseudobases: $e(\tau; \mu) \equiv e(n; \mu)$ donde μ es la variable descriptiva del conjunto según la cual:

$$\mu \equiv m \in \mathbb{Z} : \text{discreta} \rightarrow \left\{ \begin{array}{l} e(n; m) : \text{conjunto discreto} \\ (n \text{ y } m \text{ de la misma naturaleza)} \end{array} \right\}$$

A partir de la expresión (2.9) y la base definidas según la variable descriptiva del conjunto μ se puede obtener la siguiente expresión para el operador **CL**:

$$\mu \equiv m \rightarrow x_0(n) = \sum_{m=\langle N_0 \rangle} \alpha(m)e(n; m) \quad (2.21)$$

2.3.6. Bases/Pseudobases ideales.

En variable continua: Delta de Dirac. El objeto matemático denominado Delta de Dirac $\delta(x)$, no es propiamente una función sino un funcional que se aplica sobre las funciones de variable continua. Su utilidad radica en que es un objeto capaz de seleccionar un valor concreto de un continuo de valores. La delta de Dirac puede ser definida como una distribución a partir de una sucesión de funciones de buen comportamiento como, por ejemplo, gaussianas, funciones sinc o derivadas de arcotangentes. De esta manera, la delta de Dirac quedaría definida como una distribución que toma el valor 0 en todo su dominio excepto en el 0 donde no estaría definida:

$$\delta(x) = \left\{ \begin{array}{l} ? : \text{si } x = 0 \\ 0 : \text{resto} \end{array} \right\}.$$

Otra definición de este objeto matemático es viéndolo como un operador que se aplica sobre una función y obtiene como resultado el valor de la función considerada en el origen:

$$\begin{aligned}\delta(x) : S(-\infty, \infty) &\rightarrow \mathbb{C} \\ f(x) &\mapsto f(0)\end{aligned}$$

La obtención del valor deseado se obtiene de la siguiente forma:

$$\delta(x) : f(x) \mapsto \int_{x=-\infty}^{\infty} f(x)\delta(x)dx = f(0) \in \mathbb{C}. \quad (2.22)$$

Hay que tener en cuenta, que la integral que aparece en 2.22 no es una integral de Riemman, ya que no existe ningún área bajo la curva, sino como una notación que asociada a la Delta de Dirac significa que se obtiene el valor de la función $f(x)$ en el origen, es decir, se obtiene $f(0)$. En este caso, la integral se podrá interpretar como el producto escalar de la función $f(x)$ con la Delta de Dirac $\delta(x)$, obteniéndose la siguiente expresión:

$$\langle f(x), \delta(x) \rangle = \int_{x=-\infty}^{\infty} f(x)\delta(x)dx.$$

De nuevo en esta expresión hay que tener en cuenta que la integral es una notación y no se entiende como una integral de Riemman. En cualquier caso, podría entenderse que el valor complejo $f(0)$ es la proyección de la función $f(x)$ sobre la Delta de Dirac $\delta(x)$.

Una de las conclusiones más interesantes que se puede obtener de los conceptos expuestos previamente, es que la Delta de Dirac realiza un muestreo sobre la función con la que interactúa en el punto origen.

Una vez se ha estudiado como funciona la Delta de Dirac, basta con desplazar su localización para conseguir el valor de la función sobre la que se aplica para obtener el valor de dicha función en cualquier punto. Esto es, si en lugar de trabajar con una Delta de Dirac localizada en el origen se trabaja con una Delta de Dirac desplazada y localizada en el punto x_0 , se obtiene el valor de la función en dicho punto, $f(x_0)$, tal y como se muestra a continuación:

$$\delta(x - x_0) : f(x) \mapsto \int_{x=-\infty}^{\infty} f(x)\delta(x - x_0)dx = f(x_0) \in \mathbb{C},$$

$$\langle f(x), \delta(x - x_0) \rangle = \int_{x=-\infty}^{\infty} f(x)\delta(x - x_0)dx.$$

Con todo lo anterior, se concluye que, barriendo todo el dominio de la función $f(x)$ y situando la Delta de Dirac en cada punto $\delta(x - x_0)$, se puede obtener cada uno de los valores particulares $f(x_0) \in \mathbb{C}$ de la función $f(x)$.

Se puede utilizar ahora, el conjunto de deltas desplazadas como base ideal. Se parte del espacio vectorial $\mathcal{F}_{\mathcal{K}}$ de funciones de variable continua ($\tau \equiv x \in \mathbb{R}$) que tiene como elemento genérico del espacio $a(\tau) \equiv f(x)$. Si dada la base del espacio $e(x; \mu)$ se define la variable descriptiva del conjunto como $\tau' \equiv x'$ se puede considerar como caso particular al conjunto de funciones Delta de Dirac,

$$e(x; \mu) \equiv e(x; x') = \delta(x - x'), \quad (2.23)$$

donde $\tau' \equiv x'$ denota, en este caso específico, todos los posibles desplazamientos del objeto matemático $\delta(x)$. En este caso y teniendo en cuenta las expresiones (2.9) y (2.16) obtenemos que un elemento cualquiera del espacio se puede definir, en términos de la combinación lineal y el producto escalar, como,

$$f(x) = \mathbf{CL}_{(x')}[\alpha(x')\delta(x - x')] = \int_{-\infty}^{\infty} \alpha(x')\delta(x - x')dx'. \quad (2.24)$$

La definición rigurosa del objeto $\delta(x)$ como una distribución o función generalizada nos lleva a determinar que los coeficientes son $\alpha(x') = f(x')$, que junto a la expresión anterior (2.24) nos lleva al resultado bien conocido,

$$f(x) = \int_{-\infty}^{\infty} f(x')\delta(x - x')dx' = f(x) * \delta(x). \quad (2.25)$$

Cabe destacar, como conclusión importante en este caso, que *cualquier elemento del espacio $f(x)$ se puede expresar a partir de una pseudobase ideal $e(x; x') = \delta(x - x')$ en términos de sí misma, hecho representado porque los coeficientes $\alpha(x') = f(x')$.*

Hay que mencionar que, el análisis riguroso del objeto $\delta(x)$, que garantiza la validez de los desarrollos y expresiones anteriores, lleva directamente al estudio de la Teoría de Distribuciones, cuyo análisis detallado se puede encontrar, por ejemplo, en [10] y [11].

En variable discreta: Delta de Kronecker. Para el caso de variable discreta también existe un conjunto de funciones que hace el papel de base ideal. Se trata de las Deltas de Kronecker. Pese a que su notación es prácticamente idéntica a las Deltas de Dirac, las Deltas de Kronecker son funciones perfectamente definidas como:

$$\delta(n) = \left\{ \begin{array}{l} 1 : \text{si } n = 0 \\ 0 : \text{resto} \end{array} \right\}. \quad (2.26)$$

Ahora, dado el espacio vectorial $\mathcal{F}_{\mathcal{K}}$ de funciones de variable discreta ($\tau \equiv n \in \mathbb{Z}$) sus elementos serán del tipo $a(\tau) = x(n)$.

Si dada la base del espacio $e(x; \mu)$ definimos la variable descriptiva del conjunto como $\mu \equiv m$. Utilizando las Deltas de Kronecker definidas en (2.26), se define la base ideal como:

$$e(n; \mu) \equiv e(n; m) = \delta(n - m), \quad (2.27)$$

donde $\mu \equiv m$ denota todos los posibles desplazamientos de la función $\delta(n)$. De este modo, ahora cualquier función $x(n) \in D(-\infty, \infty)$ se puede expresar como:

$$x(n) = \mathbf{CL}_{(m)}[\alpha(m)\delta(n - m)] = \sum_{m=-\infty}^{\infty} \alpha(m)\delta(n - m). \quad (2.28)$$

Con la expresión anterior (2.28) los coeficientes resultan ser $\alpha(m) = x(m)$ dando lugar a la expresión bien conocida,

$$x(n) = \sum_{m=-\infty}^{\infty} x(m)\delta(n - m) = x(n) * \delta(x). \quad (2.29)$$

Finalmente, se llega a la conclusión de que *cualquier elemento del espacio $x(n)$ se puede expresar en términos de la base $e(n; m) = \delta(n - m)$ a partir de ella misma, de coeficientes $\alpha(m) = x(m)$.*

En este caso basta con la definición de la función $\delta(n)$ en (2.26) para que eso sea cierto. Nótese cómo el papel de la $\delta(x)$, y su definición como distribución, está directamente relacionada con la *generación de un objeto matemático que realice el mismo papel, en variable continua, que realiza $\delta(n)$ en variable discreta.*

2.3.7. Bases/pseudobases y transformaciones.

Aunque el concepto de transformada pueda ser visto desde el punto de vista de un operador entre dos espacios de señales, en este apartado se analizará el sentido algebraico de su significado, introduciendo, por tanto, el concepto de *transformada generalizada*. Dicho análisis está estrechamente relacionado con el estudio de la representación de una señal $a(\tau)$ de un espacio $\mathcal{F}_{\mathcal{K}}$ en términos de una base $e(\tau; \mu)$.

Como ya se ha visto antes, la señal se puede expresar en términos de la base del espacio como:

$$a(\tau) = \mathbf{CL}_{\mu}[\alpha(\mu)e(\tau; \mu)], \quad (2.30)$$

siendo, tanto para dimensión finita como infinita numerable y muchos casos de dimensión infinita no numerable,

$$\alpha(\mu) \propto \langle a(\tau), e(\tau; \mu) \rangle. \quad (2.31)$$

Al par de ecuaciones anteriores, 2.30 y 2.31, se les denomina transformación o transformada y a la combinación lineal mediante la que se obtiene la función original, $a(\tau)$, se le denomina transformada inversa.

Algunas razones principales por las que son importantes las transformaciones son:

- Permite identificar a una señal a través de sus coeficientes (coordenadas del espacio vectorial), de modo que se puede geometrizar y, por tanto, simplificar.
- En general, simplifica mucho los cálculos: en muchos casos, el análisis de las señales en el dominio espectral es más sencillo que en el dominio real. Algunos parámetros característicos de las señales en el dominio real son mucho más fáciles de obtener en el dominio espectral y, en general, las posibles operaciones suelen ser más sencillas, sobre todo cuando se trata del análisis de una señal a través de un sistema lineal.
- Dependiendo de la aplicación (entiéndase como tal el espacio de señales de partida, el sistema y las señales concretas o mayoritarias con las que se va a trabajar), se escogerá una transformada, y por tanto una base, adecuada al problema concreto que se esté analizando.

A continuación se exponen algunos casos de transformaciones concretas que se han tratado en el software.

En variable continua y señales periódicas: Desarrollo en Serie de Fourier. Se parte del espacio vectorial $\mathcal{F}_{\mathcal{K}}$ de funciones de variable continua ($\tau \equiv x \in \mathbb{R}$) que tiene como elemento genérico del espacio $a(\tau) \equiv f_0(x)$ de periodo X_0 .

Si dada la base del espacio $e(x; \mu)$ definimos la variable descriptiva del conjunto como $\mu \equiv m$ se puede considerar como caso particular al siguiente conjunto de exponenciales imaginarias,

$$e(x; \mu) \equiv e(x; m) = e^{jm\xi_0 x}, \quad (2.32)$$

donde la constante $\xi_0 = \frac{2\pi}{X_0}$. En este caso y teniendo en cuenta las expresiones (2.9) y (2.18) se obtiene que un elemento cualquiera del espacio se puede definir, en términos de la combinación lineal, como,

$$f_0(x) = \mathbf{CL}_{(m)}[\alpha(m)e^{jm\xi_0 x}] = \sum_{m=-\infty}^{\infty} \alpha(m)e^{jm\xi_0 x}. \quad (2.33)$$

El estudio particular de la ecuación (2.10) aplicado a este caso lleva a que la distancia tiende a cero cuando los coeficientes $\alpha(\mu) = \alpha(m)$ son iguales a,

$$\alpha(m) = \frac{1}{X_0} \int_{\langle X_0 \rangle} f_0(x) e^{-jm\xi_0 x} dx = \frac{1}{X_0} \langle f_0(x), e(x; m) \rangle. \quad (2.34)$$

De este modo *los coeficientes de la combinación lineal $\alpha(m)$ se pueden obtener en términos del producto escalar dando como resultado los coeficientes del desarrollo en serie de Fourier. La ecuación (2.33) junto con el valor de los coeficientes en (2.34) da lugar al Desarrollo en Serie de Fourier de señales complejas en variable continua.*

En variable continua y señales aperiódicas: Transformada de Fourier. Dado del espacio vectorial $\mathcal{F}_{\mathcal{K}}$ de funciones de variable continua ($\tau \equiv x \in \mathbb{R}$) que tiene como elemento genérico del espacio $a(\tau) \equiv f(x)$.

Si se considera la base del espacio $e(x; \mu)$, se define la variable descriptiva del conjunto como $\mu \equiv \xi$ y se considera como caso particular al siguiente conjunto de exponenciales imaginarias,

$$e(x; \mu) \equiv e(x; \xi) = e^{j\xi x}. \quad (2.35)$$

La expresión de la combinación lineal en (2.9) se escribirá ahora como,

$$f(x) = \mathbf{CL}_{(\xi)}[F(\xi)e^{j\xi x}] = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\xi)e^{j\xi x} d\xi. \quad (2.36)$$

En este caso, analizando la expresión (2.10) se tiene que la distancia tiende a cero cuando los coeficientes $\alpha(\mu) = F(\xi)$ son iguales a,

$$F(\xi) = \mathbf{TF}[f(x)] = \int_{-\infty}^{\infty} f(x)e^{-j\xi x} dx = \langle f(x), e(x; \xi) \rangle, \quad (2.37)$$

donde la integral no es más que la definición del producto escalar en el espacio de señales aperiódicas de variable continua. De este modo *los coeficientes correspondientes a la combinación lineal $F(\xi)$ se pueden escribir como el producto escalar del elemento original $f(x)$ por el continuo de elementos de la base, es decir, como la proyección del elemento $f(x)$ sobre cada uno de los elementos de la base.* Las ecuaciones (2.36) y (2.37) definen *la transformada de Fourier de una señal $f(x)$.*

En variable discreta y señales periódicas: Desarrollo en Serie de Fourier. Considerando el espacio vectorial $\mathcal{F}_{\mathcal{K}}$ de funciones de variable discreta ($\tau \equiv n \in \mathbb{Z}$) que tiene como elemento genérico del espacio $a(\tau) \equiv x_0(n)$ de periodo N_0 .

Si se parte la base del espacio $e(n; \mu)$, se define la variable descriptiva del conjunto como $\mu \equiv m$ y se puede considerar como caso particular al siguiente conjunto de exponenciales imaginarias,

$$e(n; \mu) \equiv e(n; m) = e^{jm\Omega_0 n}, \quad (2.38)$$

donde la constante $\Omega_0 = \frac{2\pi}{N_0}$. En este caso y teniendo en cuenta las expresiones (2.9) y (2.21) se obtiene que un elemento cualquiera del espacio se puede definir en términos de la combinación lineal y el producto escalar como,

$$x_0(n) = \mathbf{CL}_{(m)}[\alpha(m)e^{jm\Omega_0 n}] = \sum_{m=\langle N_0 \rangle} \alpha(m)e^{jm\Omega_0 n}. \quad (2.39)$$

Teniendo en cuenta que el conjunto de funciones definidas en (2.38) es finito (N_0 funciones), el análisis de la distancia en (2.10) se reduce al análisis de un sistema de N_0 ecuaciones con N_0 incógnitas cuya solución exacta lleva a la expresión de los coeficientes,

$$\alpha(m) = \frac{1}{N_0} \sum_{m=\langle N_0 \rangle} x_0(n)e^{-jt\Omega_0 n} = \frac{1}{N_0} \langle x_0(n), e(n; m) \rangle. \quad (2.40)$$

De este modo *los coeficientes de la combinación lineal $\alpha(m)$ se pueden obtener en términos del producto escalar dando como resultado los coeficientes del desarrollo en serie de Fourier.* Las ecuaciones (2.39) y (2.40) definen *el desarrollo en serie de Fourier de una señal arbitraria $x_0(n)$.*

En variable discreta y señales aperiódicas: Transformada de Fourier en Variable Discreta. Se parte del espacio vectorial $\mathcal{F}_{\mathcal{K}}$ de funciones de variable discreta ($\tau \equiv n \in \mathbb{Z}$) que tiene como elemento genérico del espacio $a(\tau) \equiv x(n)$.

Si dada la base del espacio $e(n; \mu)$ se define la variable descriptiva del conjunto como $\mu \equiv \Omega$, se puede considerar como caso particular al siguiente conjunto de exponenciales imaginarias,

$$e(n; \mu) \equiv e(n; \Omega) = e^{j\Omega n}. \quad (2.41)$$

Teniendo en cuenta la propiedad de periodicidad en Ω (2π) de las funciones en (2.41), la expresión de la combinación lineal en (2.9) se escribirá ahora como,

$$x(n) = \mathbf{CL}_{(\Omega)}[X(\Omega)e^{j\Omega n}] = \frac{1}{2\pi} \int_{\Omega=\langle 2\pi \rangle} X(\Omega)e^{j\Omega n} d\Omega. \quad (2.42)$$

El análisis de la distancia en (2.10) particularizada a este caso llevará al valor correcto de los coeficientes para que dicha distancia tienda a cero, obteniéndose,

$$X(\Omega) = \mathbf{TF}[x(n)] = \sum_{n=-\infty}^{\infty} x(n)e^{-j\Omega n} = \langle x(n), e(n; \Omega) \rangle, \quad (2.43)$$

donde el sumatorio no es más que la definición del producto escalar en el espacio de señales aperiódicas de variable continua. De este modo *los coeficientes correspondientes a la combinación lineal $X(\Omega)$ se pueden escribir como el producto escalar del elemento original $x(n)$ por el continuo de elementos de la base, es decir, como la proyección del elemento $x(n)$ sobre cada uno de los elementos de la base*. Las ecuaciones (2.42) y (2.43) definen así *la transformada de Fourier de una señal $x(n)$* .

2.4. Espacios de operadores.

Se define a un operador \mathbf{F} como una aplicación entre dos espacios vectoriales $\mathcal{V}_{\mathcal{K}_1}$ y $\mathcal{W}_{\mathcal{K}_2}$:

$$\left\{ \begin{array}{l} \mathbf{F} : \mathcal{V}_{\mathcal{K}_1} \rightarrow \mathcal{W}_{\mathcal{K}_2} \\ \vec{a} \mapsto \mathbf{F}[\vec{a}] = \vec{b} \end{array} \right\}. \quad (2.44)$$

Cuando ambos espacios vectoriales son considerados como espacios de funciones $\mathcal{F}_{\mathcal{K}_1}$ y $\mathcal{G}_{\mathcal{K}_2}$, la definición se mantiene de la forma:

$$\left\{ \begin{array}{l} \mathbf{F} : \mathcal{F}_{\mathcal{K}_1} \rightarrow \mathcal{G}_{\mathcal{K}_2} \\ a(\tau) \mapsto \mathbf{F}[a(\tau)] = b(\tau') \end{array} \right\}. \quad (2.45)$$

Como se indica en (2.45), las variables de las funciones origen y las funciones resultado no tienen por qué ser las mismas. Esto quiere decir entonces, que los espacios de partida y llegada pueden ser diferentes, así como los cuerpos sobre los que están definidos.

2.4.1. Clasificación inicial importante.

Se habla de espacios de operadores como aquellos conjuntos de operadores reunidos en función de alguna característica común que los describa. Las propiedades más importantes a considerar para generar espacios de operadores son:

- **Linealidad.** Se dice que un sistema descrito por un operador será lineal cuando una combinación lineal de señales a la entrada produce una combinación lineal a la salida con los mismos coeficientes, es decir,

$$\left\{ \begin{array}{l} \mathbf{F} : a_1(\tau) \rightarrow b_1(\tau) = \mathbf{F}[a_1(\tau)] \\ \mathbf{F} : a_2(\tau) \rightarrow b_2(\tau) = \mathbf{F}[a_2(\tau)] \end{array} \right\} \Rightarrow \quad .^3 \quad (2.46)$$

$$\mathbf{F} : \alpha a_1(\tau) + \beta a_2(\tau) \rightarrow \alpha b_1(\tau) + \beta b_2(\tau) = \alpha \mathbf{F}[a_1(\tau)] + \beta \mathbf{F}[a_2(\tau)]$$

- **Invarianza.** Un sistema se considera invariante cuando un desplazamiento sobre la variable independiente de la señal de entrada produce el mismo desplazamiento sobre la señal a la salida. Matemáticamente,

$$a(\tau) \rightarrow b(\tau) = \mathbf{F}[a(\tau)] \Rightarrow a(\tau - \tau_0) \rightarrow b(\tau - \tau_0). \quad (2.47)$$

Se clasifican entonces los espacios de operadores en tres grupos principales:

1. Sistemas no lineales, descritos por \mathbf{F} no lineal.
2. Sistemas lineales, descritos por \mathbf{F} lineal.
3. Sistemas lineales invariantes, descritos por \mathbf{F} lineal e invariante.

Donde los operadores del segundo tipo suponen el concepto central alrededor del cual se desarrolla este proyecto y los del tercer tipo son un caso particular de los segundos.

Siendo éstas las características más importantes, existen otras propiedades y subconjuntos por las que se pueden agrupar los operadores como subespacios. Algunas de estas características son:

- Memoria.
- Causalidad.
- Invertibilidad.
- Estabilidad.
- Inyectividad.
- Suprayectividad.
- Biyectividad.
- Subespacio rango del operador $Rang\{\mathbf{F}\}$.
- Subespacio imagen del operador $Imag\{\mathbf{F}\}$.
- Subespacio núcleo del operador $Ker\{\mathbf{F}\}$.
- Subespacio de funciones propias $Aut\{\mathbf{F}\}$.

Este subespacio se caracteriza porque el conjunto de funciones propias o autofunciones del operador se define como aquellas funciones que, tras ser influidas por el operador, se transforman en sí mismas multiplicadas por un escalar. Matemáticamente se traduce en:

$$Aut\{\mathbf{F}\} = \{\alpha(\tau) \in \mathcal{F}_{\mathcal{K}} \text{ tq. } \mathbf{F}[\alpha(\tau)] = \lambda\alpha(\tau) \text{ y } \lambda \in \mathcal{K}\} \quad (2.48)$$

Al escalar que multiplica la señal se le llama valor propio o autovalor. Hay que mencionar que sólo es posible encontrar autofunciones cuando los espacios de partida y salida son el mismo.

2.4.2. Caracterización general.

Si se elige una base/pseudobase $e(\tau; \tau')$ para el espacio considerado, se puede representar a un elemento del espacio siguiendo la expresión (2.9) como

$$\mathbf{F}[a(\tau)] = \mathbf{F}\{\mathbf{CL}_{(\tau')}[\alpha(\mu)e(\tau; \mu)]\}, \quad (2.49)$$

y si, además, el operador es lineal,

$$\mathbf{F}[a(\tau)] = \mathbf{CL}_{(\tau')}\{\alpha(\mu)\mathbf{F}[e(\tau; \mu)]\}. \quad (2.50)$$

Lo que quiere decir que la transformación realizada por el operador queda caracterizada por la transformación de la base/pseudobase elegida.

Si el operador es lineal y la base/pseudobase pertenece al subespacio de funciones propias o autofunciones ($e(\tau; \mu) \in \text{Aut}\{\mathbf{F}\}$),

$$\mathbf{F}[e(\tau; \mu)] = \lambda(\mu)e(\tau; \mu),$$

siendo $\lambda(\tau')$ el conjunto de autovalores asociados a la base/pseudobase. Así, por ejemplo, $\lambda(\mu_0)$ será el autovalor de la función $e(\tau; \mu_0)$ respecto del operador \mathbf{F} .

2.4.3. Caracterización funcional de los operadores lineales.

Espacios de funciones de variable continua. Se parte de un espacio vectorial $\mathcal{F}_{\mathcal{K}}$ de funciones de variable continua ($\tau \equiv x \in \mathbb{R}$) con la pseudobase ideal de variable continua $e(x; x') = \delta(x - x')$. Trasladando la expresión (2.25) al espacio de operadores se obtiene que:

$$g(x) = \mathbf{F}[f(x)] = \mathbf{F}\left[\int_{-\infty}^{\infty} f(x')\delta(x - x')dx'\right]. \quad (2.51)$$

Si el operador es **lineal** la expresión anterior se transforma en la siguiente:

$$g(x) = \mathbf{F}[f(x)] = \int_{-\infty}^{\infty} f(x')\mathbf{F}[\delta(x - x')]dx' = \int_{-\infty}^{\infty} f(x')h(x; x')dx'. \quad (2.52)$$

De este modo queda demostrado que *todo operador lineal en variable continua tiene una caracterización integral en términos del conjunto de funciones $h(x; x')$,*

$$h(x; x') = \mathbf{F}[\delta(x - x')]. \quad (2.53)$$

Así se puede trabajar en el espacio de funciones en lugar de en el de operadores, siempre y cuando se considere al conjunto de funciones $h(x; x')$ adecuado. *A este conjunto de funciones se le denomina conjunto de respuestas al impulso.*

Operador (\mathbf{F}) lineal	Conjunto de funciones	
\mathbf{F}_1	$h_1(x; x')$	Operador integral en (2.52)
\mathbf{F}_2	$h_2(x; x')$	
\mathbf{F}_3	$h_3(x; x')$	
...	...	

Si además de lineal el operador es **invariante** el conjunto de funciones $h(x; x')$ pasa a ser

$$\mathbf{F}[\delta(x - x')] = h(x; x') = h(x - x'), \quad (2.54)$$

de modo que la expresión (2.52) se reescribe como

$$g(x) = \mathbf{F}[f(x)] = \int_{x'} f(x')h(x - x')dx' = f(x) * h(x). \quad (2.55)$$

En este caso *todo el conjunto de respuestas al impulso* $h(x; x') = h(x - x')$ *se describe en términos de una única función,*

$$h(x) = \mathbf{F}[\delta(x)]. \quad (2.56)$$

Finalmente se concluye que *todo operador lineal e invariante tiene una representación integral en términos de una única función* $h(x)$ *a la que denominamos respuesta al impulso del sistema.* El esquema anterior se reduce ahora al siguiente:

Operador (\mathbf{F}) lineal e invariante	Conjunto de funciones	
\mathbf{F}_1	$h_1(x) \rightarrow h_1(x - x')$	Operador integral en (2.55) (convolución continua)
\mathbf{F}_2	$h_2(x) \rightarrow h_1(x - x')$	
\mathbf{F}_3	$h_3(x) \rightarrow h_3(x - x')$	
...	...	

Espacios de funciones de variable discreta. Se parte del el espacio vectorial $\mathcal{F}_{\mathcal{X}}$ de funciones de variable discreta ($\tau \equiv n \in \mathbb{Z}$) con la pseudobase ideal de variable discreta $e(n; m) = \delta(n - m)$. Si se traslada la expresión (2.29) al espacio de operadores se tiene que,

$$y(n) = \mathbf{F}[x(n)] = \mathbf{F}\left[\sum_{m=-\infty}^{\infty} x(m)\delta(n - m)\right]. \quad (2.57)$$

Si el operador es **lineal**, la expresión (2.57) se transforma en la siguiente:

$$y(n) = \mathbf{F}[x(n)] = \sum_{m=-\infty}^{\infty} x(m)\mathbf{F}[\delta(n - m)] = \sum_{m=-\infty}^{\infty} x(m)h(n; m). \quad (2.58)$$

De este modo queda demostrado que *todo operador lineal en variable discreta tiene una caracterización sumatoria en términos del conjunto de funciones* $h(n; m)$,

$$h(n; m) = \mathbf{F}[\delta(n - m)]. \quad (2.59)$$

De este modo *se trabaja en el espacio de funciones en lugar de en el de operadores, siempre y cuando se considere al conjunto de funciones* $h(n; m)$ *adecuado. Dicho conjunto de funciones se denomina conjunto de respuestas al impulso del operador.*

Operador (\mathbf{F}) lineal	Conjunto de funciones	
\mathbf{F}_1	$h_1(n; m)$	Operador suma en (2.58)
\mathbf{F}_2	$h_2(n; m)$	
\mathbf{F}_3	$h_3(n; m)$	
...	...	

Si además el operador es **invariante** el conjunto de respuestas al impulso $h(n; m)$ pasa a ser,

$$\mathbf{F}[\delta(n - m)] = h(n; m) = h(n - m), \quad (2.60)$$

reescribiendo la expresión (2.58) como,

$$y(n) = \mathbf{F}[x(n)] = \sum_{m=-\infty}^{\infty} x(m)h(n - m) = x(n) * h(n). \quad (2.61)$$

En este caso *todo el conjunto de respuestas al impulso* ($h(n; m) = h(n - m)$) *se describe en términos de una única función,*

$$h(n) = \mathbf{F}[\delta(n)]. \quad (2.62)$$

Finalmente concluimos que *todo operador lineal e invariante tiene una representación sumatoria en términos de una única función $h(n)$, denominada respuesta al impulso del sistema.* El esquema anterior se reduce ahora al siguiente:

Operador (\mathbf{F}) lineal e invariante	Conjunto de funciones	
\mathbf{F}_1	$h_1(n) \rightarrow h_1(n - m)$	Operador suma en (2.61) (convolución discreta)
\mathbf{F}_2	$h_2(n) \rightarrow h_1(n - m)$	
\mathbf{F}_3	$h_3(n) \rightarrow h_3(n - m)$	
...	...	

2.5. Análisis espectral generalizado.

A lo largo del presente apartado se mostrará, de forma resumida, el desarrollo matemático que se debe seguir para interpretar un sistema lineal, en general no invariante, en el dominio espectral. De forma general, este análisis ya no se corresponderá con el estudio de los autovalores que producen las autofunciones del sistema, sino que consistirá en estudiar, por tanto, la señal de salida en el dominio espectral, independientemente de si las funciones que describen la transformada bajo la cual se está realizando el análisis espectral son autofunciones del sistema en cuestión. De esta forma se extiende y generaliza el concepto de análisis espectral.

Como ya se ha expuesto previamente, se puede expresar la señal de salida de un sistema como la combinación lineal de la señal de entrada al sistema y el conjunto de respuestas al impulso que lo caracterizan, esto es,

$$b(\tau) = \mathbf{CL}_{\tau'}[a(\tau')h(\tau; \tau')]. \quad (2.63)$$

Asimismo, las señales se pueden expresar como una combinación lineal de una serie de coeficientes y la base escogida,

$$\begin{aligned} a(\tau) &= \mathbf{CL}_\mu[\alpha(\mu)e(\tau; \mu)] \\ b(\tau) &= \mathbf{CL}_\mu[\beta(\mu)e(\tau; \mu)] \end{aligned} \quad (2.64)$$

donde los coeficientes $\alpha(\mu)$ y $\beta(\mu)$ son la transformada de las señales originales en el dominio espectral. El objetivo será, por tanto, conseguir una expresión que permita obtener la transformada de la señal de salida en función de la transformada de la señal de entrada, esto es, la relación entre ambos conjuntos de coeficientes.

Como ya se ha expuesto, el conjunto de coeficientes no es más que la señal original en el dominio transformado, se obtiene entonces que:

$$\beta(\mu) = \mathbf{T}_\tau[b(\tau)] = \mathbf{T}_\tau[\mathbf{CL}_{\tau'}[a(\tau')h(\tau; \tau')]] = \mathbf{CL}_{\tau'}[a(\tau')\mathbf{T}_\tau[h(\tau; \tau')]]. \quad (2.65)$$

Por otro lado, se obtiene que la transformada del conjunto de respuestas al impulso respecto a la variable τ se expresa como:

$$\mathbf{T}_\tau[h(\tau; \tau')] = \mathbf{T}_\tau[\mathbf{CL}_\mu[H(\mu; \tau')e(\tau; \mu)]] = \mathbf{CL}_{\tau'}[H(\mu; \tau')\mathbf{T}_\tau[e(\tau; \mu)]]. \quad (2.66)$$

Y finalmente, la transformada del conjunto de funciones base sobre esa misma base siempre será una delta:

$$\mathbf{T}_\tau[e(\tau; \mu_0)] = A\delta(\mu' - \mu'_0). \quad (2.67)$$

De modo que la expresión de la transformada del conjunto de respuestas al impulso se reduce a:

$$\mathbf{T}_\tau[h(\tau; \tau')] = \mathbf{CL}_\mu[H(\mu; \tau')A\delta(\mu' - \mu'_0)] = AH(\mu; \tau') \quad (2.68)$$

y la transformada de la señal de salida se puede expresar como:

$$\beta(\mu) = A\mathbf{CL}_{\tau'}[a(\tau')H(\mu; \tau')]. \quad (2.69)$$

Si se considera el parámetro τ' como una nueva variable, cuya función original era la de enumerar las funciones, el conjunto de funciones $H(\mu; \tau')$ pueden expresarse como una combinación lineal si se redefine el conjunto de funciones base sobre la variable τ' descritas por el parámetro μ' :

$$H(\mu; \mu) = \mathbf{CL}_{\mu'}[H(\mu; \mu')e(\mu; \mu')]. \quad (2.70)$$

De este modo la transformada de la señal de salida se puede expresar como:

$$\beta(\mu) = A\mathbf{CL}_{\tau'}[a(\tau')\mathbf{CL}_{\mu'}[H(\mu; \mu')e(\tau'; \mu')]] = A\mathbf{CL}_{\mu'}[H(\mu; \mu')\mathbf{CL}_\mu[a(\tau')e(\tau'; \mu')]]. \quad (2.71)$$

Por otro lado, la señal de entrada se puede expresar, en otras variables, como:

$$a(\tau') = \mathbf{CL}_\varpi[\alpha(\varpi)e(\tau'; \varpi)]. \quad (2.72)$$

Introduciendo esta nueva variable en la expresión (2.71) se obtiene el siguiente resultado:

$$\begin{aligned}\beta(\mu) &= A\mathbf{CL}_{\mu'}[H(\mu; \mu')\mathbf{CL}_{\tau'}[\mathbf{CL}_{\varpi}[\alpha(\varpi)e(\tau'; \varpi)]e(\tau'; \mu')]] = \\ &= A\mathbf{CL}_{\mu'}[H(\mu; \mu')\mathbf{CL}_{\varpi}[\alpha(\varpi)\mathbf{CL}_{\tau'}[e(\tau'; \varpi)e(\tau'; \mu')]]].\end{aligned}\quad (2.73)$$

La combinación lineal sobre la variable τ' no es más que la combinación lineal de una base ponderada por una función que hace el papel de coeficientes y esto se traduce en la transformada inversa de una función de la base que define dicha transformada, esto es, una delta⁴:

$$\mathbf{CL}_{\tau'}[e(\tau'; \varpi)e(\tau'; \mu')] = B\delta_*(\mu'; \varpi). \quad (2.74)$$

Ahora la combinación lineal en ϖ se resuelve como las modificaciones de la función $\alpha(\varpi)$ evaluada en los distintos valores de μ' :

$$\mathbf{CL}_{\varpi}[\alpha(\varpi)B\delta_*(\mu'; \varpi)] = B\alpha_*(\mu'). \quad (2.75)$$

Con este resultado en mente se obtiene que la transformada de la señal de salida se puede expresar como:

$$\beta(\mu) = K\mathbf{CL}_{\mu'}[\alpha_*(\mu')H(\mu; \mu')], \quad (2.76)$$

donde $K = AB$, resultado que representa la relación entre la transformada de las señales de entrada y salida y que se puede comparar con la relación en el dominio real.

De esta manera, se puede expresar la representación general de un sistema lineal en el dominio espectral como:

$$\mathbf{F}_{\mathbf{L}_\mu}[o] = K\mathbf{CL}_{\mu'}[(o)_{\mu'}H(\mu; \mu')]. \quad (2.77)$$

En el caso particular de que las funciones base sean autofunciones del sistema la expresión se simplifica a:

$$\beta(\mu) = \alpha(\mu)\lambda(\mu), \quad (2.78)$$

véase para una explicación rigurosa de la simplificación anterior [2].

2.6. Casos particulares del esquema general.

A continuación se presentan una serie de casos en los que se particulariza, a modo de resumen, el esquema general de la Fig. 2.1 a los casos que se han expuesto a lo largo del capítulo. Con los siguientes esquemas se pretende predisponer al lector para que comprenda mejor las siguientes explicaciones del presente proyecto, relativas al funcionamiento del software desarrollado.

⁴Donde $\delta_*(\mu'; \varpi)$ es una delta genérica de parámetros μ' y ϖ , que dependiendo de la transformada puede representar, por ejemplo, las siguientes funciones:

- $\delta_*(\mu'; \varpi) = \delta(\mu' - \varpi)$
- $\delta_*(\mu'; \varpi) = \delta(\mu' + \varpi)$

2.6.1. Espacio de funciones complejas de variable real definidas para todo x .

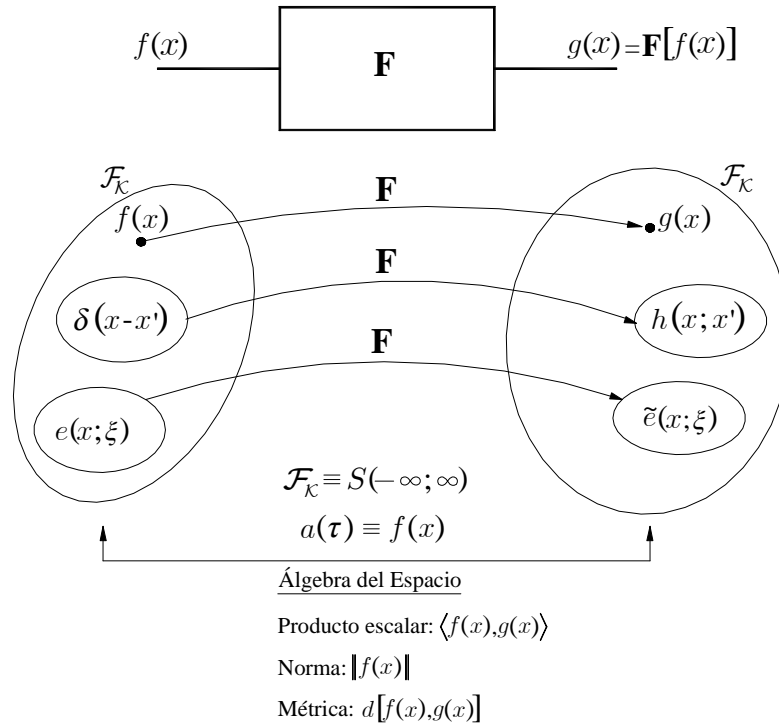


Figura 2.2: Si las funciones base elegidas son $e(x; \xi) = e^{j\xi x}$, se habla de la Transformada de Fourier. Si, además, el sistema \mathbf{F} es lineal e invariante, dichas funciones pertenecerán a $\text{Aut}\{\mathbf{F}\}$, de forma que $\tilde{e}(x; \xi) = H(\xi)e(x; \xi)$.

- A continuación se muestra la definición de los parámetros algebraicos del espacio:

Producto escalar:	$\langle f(x)g(x) \rangle = \int_{-\infty}^{\infty} f(x)g^*(x)dx$
Norma:	$\ f(x)\ ^2 = \int_{-\infty}^{\infty} f(x) ^2 dx$
Métrica:	$d^2[f(x), g(x)] = \int_{-\infty}^{\infty} f(x) - g(x) ^2 dx$

- A continuación se muestra la representación de un sistema seleccionable en este espacio en los dominios real y espectral, en este caso bajo la Transformada de Fourier:

	Sistema Lineal	Sistema Lineal e Invariante
Dominio Real	$\mathbf{F}_{\mathbf{L}_x}[o] = \int_{x'} (o)_{x'} h(x; x') dx'$	$\mathbf{F}_{\mathbf{L}_x}[o] = (o) * h(x)$
Dominio Espectral	$\mathbf{F}_{\mathbf{L}_\xi}[o] = \frac{1}{2\pi} \int_{\xi'} (o)_{\xi'} H(\xi; -\xi') d\xi'$	$\mathbf{F}_{\mathbf{L}_\xi}[o] = (o)H(\xi)$

2.6.2. Espacio de funciones complejas de variable real periódicas de periodo X_0 .

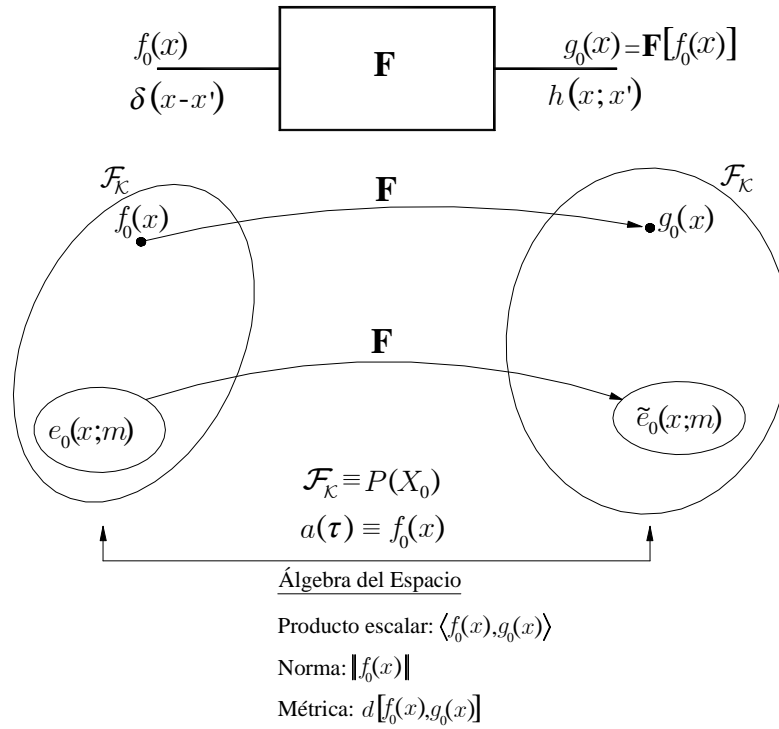


Figura 2.3: Si las funciones base elegidas son $e_0(x; m) = e^{jm\frac{2\pi}{X_0}x}$, se está hablando del Desarrollo en Serie de Fourier. Si, además, el sistema \mathbf{F} es lineal e invariante, dichas funciones pertenecerán a $\text{Aut}\{\mathbf{F}\}$, de forma que $\tilde{e}_0(x; m) = H(m)e_0(x; m)$.

- Los parámetros algebraicos definidos para el presente espacio son:

Producto escalar:	$\langle f_0(x), g_0(x) \rangle = \int_{\langle X_0 \rangle} f_0(x) g_0^*(x) dx$
Norma:	$\ f_0(x)\ ^2 = \int_{\langle X_0 \rangle} f_0(x) ^2 dx$
Métrica:	$d^2[f_0(x), g_0(x)] = \int_{\langle X_0 \rangle} f_0(x) - g_0(x) ^2 dx$

- A continuación se muestra la representación de un sistema seleccionable en este espacio en los dominios real y espectral, en este caso bajo el Desarrollo en Serie de Fourier:

	Sistema Lineal	Sistema Lineal e Invariante
Dominio Real	$\mathbf{F}_{L_x}[o] = \int_{x'} (o)_{x'} h(x; x') dx'$	$\mathbf{F}_{L_x}[o] = (o) * h(x)$
Dominio Espectral	$\mathbf{F}_{L_m}[o] = X_0 \sum_{m'} (o)_{\xi'} H(m; -m')$	$\mathbf{F}_{L_m}[o] = (o) H(m)$

2.6.3. Espacio de funciones complejas de variable entera definidas para todo n .

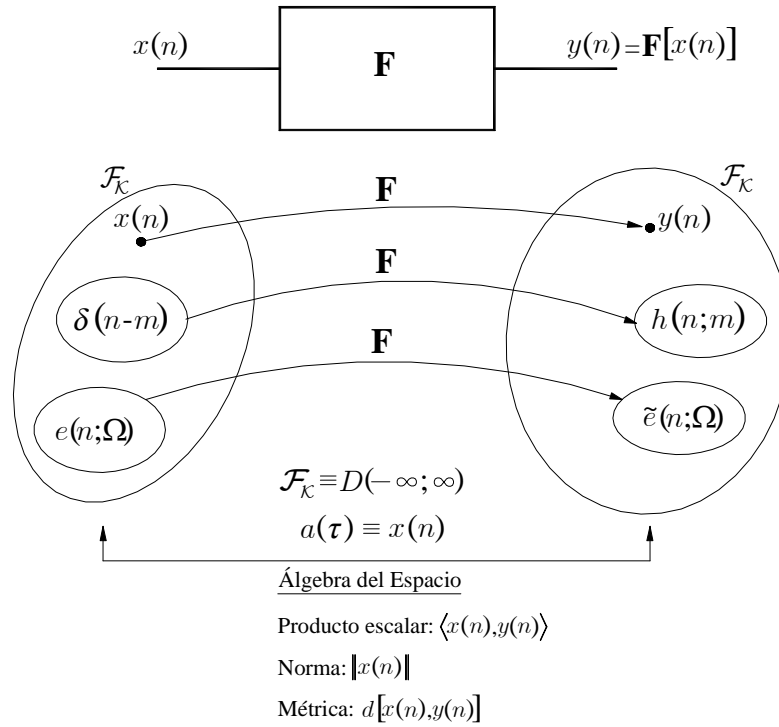


Figura 2.4: Si las funciones base elegidas son $e(n; \Omega) = e^{j\Omega n}$, se habla de la Transformada de Fourier. Si, además, el sistema \mathbf{F} es lineal e invariante, dichas funciones pertenecerán a $\text{Aut}\{\mathbf{F}\}$, de forma que $\tilde{e}(n; \Omega) = H(\Omega)e(n; \Omega)$.

- En la siguiente tabla se definen los parámetros algebraicos del espacio:

Producto escalar:	$\langle x(n), y(x) \rangle = \sum_{n=-\infty}^{\infty} x(n)y^*(n)$
Norma:	$\ x(n)\ ^2 = \sum_{n=-\infty}^{\infty} x(n) ^2$
Métrica:	$d^2[x(n), y(n)] = \sum_{n=-\infty}^{\infty} x(n) - y(n) ^2$

- A continuación se muestra la representación de un sistema seleccionable en este espacio en los dominios real y espectral, en este caso bajo la Transformada Discreta de Fourier:

	Sistema Lineal	Sistema Lineal e Invariante
Dominio Real	$\mathbf{F}_{\mathbf{L}_n}[o] = \sum_{n'} (o)_{n'} h(n; n')$	$\mathbf{F}_{\mathbf{L}_n}[o] = (o) * h(n)$
Dominio Espectral	$\mathbf{F}_{\mathbf{L}_\Omega}[o] = \frac{1}{2\pi} \int_{\Omega'} (o)_{\Omega'} H(\Omega; -\Omega') d\Omega'$	$\mathbf{F}_{\mathbf{L}_\Omega}[o] = (o)H(\Omega)$

2.6.4. Espacio de funciones complejas de variable entera periódicas de periodo N_0 .

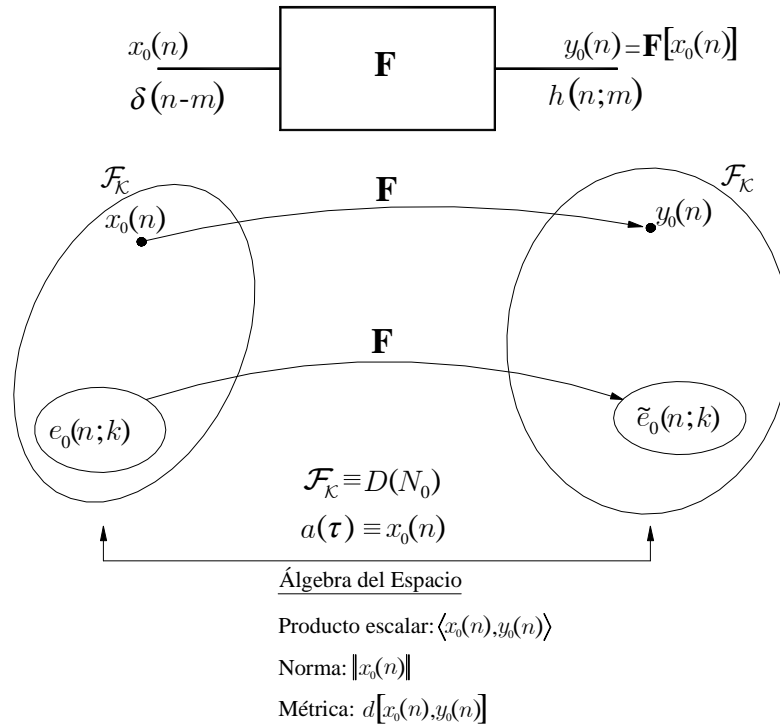


Figura 2.5: Si las funciones base elegidas son $e_0(n; k) = e^{jk\frac{2\pi}{N_0}n}$, se estará hablando del Desarrollo en Serie de Fourier de Variable Discreta. Si, además, el sistema \mathbf{F} es lineal e invariante, dichas funciones pertenecerán a $\text{Aut}\{\mathbf{F}\}$, de forma que $\tilde{e}_0(n; k) = H(k)e_0(n; k)$.

- Los parámetros algebraicos definidos para este espacio son los siguientes:

Producto escalar:	$\langle x_0(n), y_0(n) \rangle = \sum_{n=0}^{N_0} x_0(n)y_0^*(n)$
Norma:	$\ x_0(n)\ ^2 = \sum_{n=0}^{N_0} x_0(n) ^2$
Métrica:	$d^2[x_0(n), y_0(n)] = \sum_{n=0}^{N_0} x_0(n) - y_0(n) ^2$

- A continuación se muestra la representación de un sistema seleccionable en este espacio en los dominios real y espectral, en este caso bajo el Desarrollo en Serie de Fourier:

	Sistema Lineal	Sistema Lineal e Invariante
Dominio Real	$\mathbf{F}_{\mathbf{L}_n}[o] = \sum_{n'} (o)_{n'} h(n; n')$	$\mathbf{F}_{\mathbf{L}_n}[o] = (o) * h(n)$
Dominio Espectral	$\mathbf{F}_{\mathbf{L}_k}[o] = \sum_{k'} (o)_{k'} H(k; -k')$	$\mathbf{F}_{\mathbf{L}_k}[o] = (o)H(k)$

3. ESTRUCTURA GENERAL Y FUNCIONALIDADES DEL PROGRAMA

3.1. Introducción a la estructura del programa.

En la fig. 3.1 se muestra un esquema muy básico de la teoría de señales y sistemas. Como se puede apreciar en él, se pueden realizar dos tipos de divisiones. Por un lado, la división principal del esquema es aquella que distingue entre los siguientes elementos:

- El espacio de señales de entrada. Se debe recordar como ya se expuso en el capítulo 2 que este espacio de señales se construye a partir de un espacio vectorial que dispone de sus propias definiciones algebraicas.
- El espacio de operadores. El operador seleccionado será el que defina el verdadero comportamiento del sistema, esto es, las transformaciones que producirá sobre la señal de entrada y que generarán una señal de salida.
- El espacio de señales de salida. De nuevo, al igual que el espacio de entrada, estará construido a partir de un espacio vectorial que es independiente al espacio de entrada.

Por otro lado, se puede distinguir entre los dos tipos de análisis que se pueden realizar del sistema:

- El análisis en el dominio real. En terminos generales, cuando el sistema sea lineal, la señal de salida se obtendrá como una combinación lineal de la señal entrante y el conjunto de respuestas al impulso que caracterizan al operador en el dominio real.
- El análisis en el dominio espectral. Este análisis utiliza el dominio transformado para estudiar la señal de salida, en dicho dominio, como una combinación lineal de la transformada de la señal de entrada y el conjunto de funciones núcleo que caracterizan al operador en el dominio espectral.

Puesto que el principal objetivo de este proyecto ha sido el estudio y desarrollo de la funcionalidad para el análisis espectral generalizado, a lo largo del presente capítulo se describirán en profundidad los conceptos y desarrollos realizados para tal fin describiendo brevemente el resto de funcionalidades. Para más información se puede consultar [4] - [6].

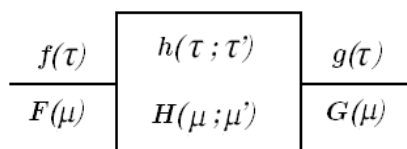


Figura 3.1: En este esquema se muestran las tres partes principales del sistema, de izquierda a derecha: espacio de señales de entrada, espacio de operadores y espacio de salida. También se observa una división horizontal, en la parte de arriba se analiza el sistema en el dominio real y en la parte inferior en el dominio espectral.

3.1.1. Espacios de partida.

Cuando se ejecuta el software se puede apreciar su analogía con un sistema de señales: en su interfaz principal se pueden distinguir los espacios de señales de entrada y salida y el espacio de operadores. Se cumple así con uno de los objetivos del programador: abstraer al usuario de la base conceptual subyacente y complicada ofreciéndole como única funcionalidad un sistema de señales de gran versatilidad.

Sin embargo, como se explicará en el capítulo 1.5.2, a nivel interno el programa tiene una división muy diferente. Está estructurado en cinco módulos principales que son totalmente independientes entre sí, ya que tienen una base conceptual totalmente distinta. Dichos módulos se generan en torno al espacio de señales del que parte el programa para trabajar. De esta forma serán:

- Continuo aperiódico.
- Continuo periódico.
- Discreto aperiódico.
- Discreto periódico.
- Continuo aperiódico definido para valores positivos.

Para cada uno de ellos, el programa pondrá a disposición del usuario una serie de señales y operadores con los que poder trabajar. A lo largo de este capítulo se expondrán y explicarán tanto dichos objetos como la base conceptual subyacente a ellos.

3.1.2. Espacios de operadores.

El objetivo principal del software es analizar un sistema de señales en profundidad, que como ya se ha expuesto estará mostrado por los espacios de señales de entrada y salida y el espacio de operadores. Para conseguir una herramienta realmente funcional y que cumpla ese objetivo se ha desarrollado un programa que, a grandes rasgos, desempeña tres funciones principales: opera con señales dando como resultado una nueva señal, calcula sus parámetros algebraicos y las representa.

En cuanto a la operación con las señales se debe distinguir entre dos casuísticas muy diferenciadas:

- Operadores. Los operadores caracterizan y definen el sistema. Los sistemas, como ya se ha indicado, cuentan con una señal de entrada, una señal de salida y una serie de propiedades. La señal de salida se calcula como una combinación lineal de la señal de entrada y el conjunto de respuestas al impulso que caracterizan al operador, y que variarán para cada operador. Los operadores cuentan con varias propiedades de las que cabe destacar dos principales: linealidad e invarianza. Sólo cuando el operador sea lineal se podrá caracterizar en términos de un conjunto de funciones.
- Operaciones básicas. Las operaciones básicas son aquellas que calculan, punto a punto, una señal a partir de dos señales iniciales. Estas operaciones no se utilizan para analizar o caracterizar ningún sistema y, por lo tanto, carecen de las propiedades que tienen los operadores. La presente versión del software dispone de las siguientes operaciones básicas: suma, resta, multiplicación y división.

Tal y como se puede intuir por la descripción de estas dos funcionalidades, serán los operadores aquellos que se utilizarán para definir y poder analizar un sistema, siendo las operaciones básicas una funcionalidad de menos importancia dentro del software. Ya que a lo largo de este capítulo

se expondrán las novedades estudiadas e implementadas en esta actualización del programa, centrándose en la funcionalidad para el análisis espectral generalizado, sólo se tratarán los operadores. Para más información a cerca de las operaciones básicas puede consultar [6].

3.2. Espacios de señales.

Los espacios de señales no han sido objeto de modificación u ampliación a lo largo del desarrollo del proyecto, éstos se han mantenido con la misma estructura y definiciones que en las versiones previas del software, versiones 2, 3 y 4. Por lo tanto, no se profundizará sobre estos aspectos sino que simplemente se presentarán los conceptos fundamentales relacionados con los espacios de señales de modo que el lector pueda entender el funcionamiento general del software. Para una explicación detallada puede consultar [4] - [6].

A continuación se describen brevemente los dos tipos de definiciones que presenta el programa para cada espacio de partida, definiciones algebraicas y señales con las que trabajar.

3.2.1. Definición del álgebra del espacio.

Dado que los espacios de señales están basados en la teoría generalizada de señales y sistemas, disponen de una estructura de espacio vectorial. Para cada uno de estos espacios, se definirá un elemento de ellos como un vector cualquiera definido por la señal escogida de modo que se cumplan las leyes de composición del espacio, es decir, de modo que la suma y el producto por un escalar se definan punto a punto.

Una vez se ha definido el conjunto de partida se definen el resto de parámetros algebraicos que se presentan a continuación.

Producto escalar. Esta operación permite introducir en el espacio las ventajas de la operación del producto escalar de vectores asociadas a la proyección de un elemento sobre otro.

Norma. Se utiliza para cuantificar los elementos del espacio. En todos los casos analizados en el software estará inducida por el producto escalar. Además, la norma definida estará directamente relacionada con la energía siendo ésta última el cuadrado de la norma.

Distancia. Es el elemento que nos permite medir la distancia entre dos elementos del espacio.

A modo de resumen, se presentan en la siguiente tabla las definiciones algebraicas disponibles para cada espacio.

Espacio de señales complejas de variable real definidas para todo valor $S(-\infty, \infty)$ ($\ni S(0, \infty)$)	P. Escalar	$\langle f(x), g(x) \rangle = \int_{-\infty}^{\infty} f(x)g^*(x)dx$
	Norma	$\ f(x)\ ^2 = \int_{-\infty}^{\infty} f(x) ^2 dx$
	Distancia	$d^2[f(x), g(x)] = \int_{-\infty}^{\infty} f(x) - g(x) ^2 dx$
Espacio de señales complejas de variable real con periodo $X_0 : P(X_0)$	P. Escalar	$\langle f_0(x), g_0(x) \rangle = \int_{\langle x_0 \rangle} f_0(x)g_0^*(x)dx$
	Norma	$\ f_0(x)\ ^2 = \int_{\langle x_0 \rangle} f_0(x) ^2 dx$
	Distancia	$d^2[f_0(x), g_0(x)] = \int_{\langle x_0 \rangle} f_0(x) - g_0(x) ^2 dx$
Espacio de señales complejas de variable entera definidas para todo valor $D(-\infty, \infty)$	P. Escalar	$\langle x(n), y(n) \rangle = \sum_{n=-\infty}^{\infty} x(n)y^*(n)$
	Norma	$\ x(n)\ ^2 = \sum_{n=-\infty}^{\infty} x(n) ^2$
	Distancia	$d^2[x(n), y(n)] = \sum_{n=-\infty}^{\infty} x(n) - y(n) ^2$
Espacio de señales complejas de variable entera con periodo $N_0 : D(N_0)$	P. Escalar	$\langle x_0(n), y_0(n) \rangle = \sum_{n=\langle N_0 \rangle} x_0(n)y_0^*(n)$
	Norma	$\ x_0(n)\ ^2 = \sum_{n=\langle N_0 \rangle} x_0(n) ^2$
	Distancia	$d^2[x_0(n), y_0(n)] = \sum_{n=\langle N_0 \rangle} x_0(n) - y_0(n) ^2$

(3.1)

3.2.2. Definición de las señales del espacio.

La base de datos de señales del programa ha permanecido sin modificaciones respecto a versiones anteriores del software, por lo que no se explicarán las señales disponibles en el software. Si desea consultar sus definiciones puede consultar el apéndice B o [4] - [5]. Sin embargo, y para facilitar su comprensión, se explicará en este apartado su clasificación general.

La clasificación de las señales se establece en cuanto a dos motivos: su naturaleza y su tratamiento. Respecto al primer motivo, la naturaleza de las señales, la clasificación se lleva a cabo en dos pasos:

1. Inicialmente, se debe diferenciar entre los dos tipos de señales de las disponibles: funciones y distribuciones. Las primeras se refieren a las funciones de buen comportamiento, es decir, que toman valores para todo el rango de definición. Las señales del segundo tipo, distribuciones o funciones generalizadas, serán aquellas que presenten ciertas particularidades que impidan clasificarlas en el grupo anterior, como discontinuidades o inderivabilidad. Hay que mencionar que esta primera clasificación no tiene sentido en los espacios de variable discreta, dada su naturaleza.
2. La siguiente distinción se realiza en función de la energía de las señales en estudiadas. Éstas pueden ser de energía finita o infinita.

Respecto al tratamiento de las señales, estas se dividen en dos tipos: analíticas y numéricas. Las primeras tomarán valores en el rango $(-\infty, \infty)$. Mientras que las segundas sólo tomarán valores para un rango concreto y definido $[a, b] \in \tau$. Esta distinción adquirirá sentido a la hora de operar con las señales.

3.2.3. Análisis en el dominio espectral.

Tal y como se ha introducido en capítulos anteriores, el principal objetivo de este proyecto ha sido el estudio del análisis espectral generalizado de los sistemas lineales, sean éstos invari-

antes o no. A continuación, y a modo de resumen, se presentan los casos de análisis espectral generalizado implementados en el software.

Se debe resaltar que esta funcionalidad sólo estará disponible siempre y cuando sea realizable, esto es, cuando el operador analizado sea lineal y por tanto exista un conjunto de respuestas al impulso que caractericen al operador.

Análisis espectral generalizado bajo la transformada de Fourier para el espacio de señales complejas de variable real definidas para todo valor: $S(-\infty, \infty)$. En este caso el análisis se realiza bajo la transformada de Fourier. Su definición en este espacio es la siguiente:

$$F(\xi) = \int_{-\infty}^{\infty} f(x)e^{-j\xi x} dx. \quad (3.2)$$

De este modo la señal de entrada en el dominio transformado se obtendrá a partir de la señal definida en el dominio real bajo la expresión mostrada en 3.2.

Por otra parte, el conjunto de funciones núcleo que definen al operador en el dominio transformado se obtiene a partir el conjunto de respuestas que le caracterizan en el dominio real $h(x; x')$. De modo que:

$$H(\xi; \xi') = \mathbf{TF}_{x'}[\mathbf{TF}_x[h(x; x')]] = \int_{x'=-\infty}^{\infty} \int_{x=-\infty}^{\infty} h(x; x')e^{-j\xi x} e^{-j\xi' x'} dx dx'. \quad (3.3)$$

Finalmente, la señal resultante se obtiene a partir de las dos expresiones anteriores:

$$G(\xi) = \frac{1}{2\pi} \int_{\xi'=-\infty}^{\infty} F(\xi')H(\xi; -\xi')d\xi'. \quad (3.4)$$

Cuando el sistema es lineal e invariante, los cálculos se simplifican. En este caso, el conjunto de respuestas al impulso $h(x; x')$ se transforma en un conjunto de señales de una sola variable desplazadas, esto es, $h(x - x')$. Por lo tanto, el cálculo del conjunto de funciones núcleo que representan al operador en el dominio espectral se reduce a:

$$H(\xi; \xi') = \mathbf{TF}_{x'}[\mathbf{TF}_x[h(x - x')]] = \int_{x'=-\infty}^{\infty} \int_{x=-\infty}^{\infty} h(x - x')e^{-j\xi x} e^{-j\xi' x'} dx dx'. \quad (3.5)$$

$$\begin{aligned} \text{Si } u &= x - x' \\ H(\xi; \xi') &= \int_{x'=-\infty}^{\infty} \int_{u=-\infty}^{\infty} h(u)e^{-j\xi(u+x')} e^{-j\xi' x'} du dx' = \\ &= \int_{x'=-\infty}^{\infty} e^{-j(\xi+\xi')x'} dx' \int_{u=-\infty}^{\infty} h(u)e^{-j\xi u} du = 2\pi\delta(\xi + \xi')H(\xi). \end{aligned} \quad (3.6)$$

Ahora se puede desarrollar la expresión 3.4 con los resultados obtenidos de modo que:

$$G(\xi) = \frac{1}{2\pi} \int_{\xi'=-\infty}^{\infty} F(\xi')2\pi\delta(\xi - \xi')H(\xi)d\xi' = F(\xi)H(\xi). \quad (3.7)$$

Análisis espectral generalizado bajo el desarrollo en serie de Fourier para el espacio de señales complejas de variable real con periodo X_0 : $P(X_0)$. En este caso el análisis se realiza bajo el desarrollo en serie de Fourier. Su definición en este espacio es la siguiente:

$$a(m) = \frac{1}{X_0} \int_{\langle X_0 \rangle} f_0(x) e^{-jm\xi_0 x} dx, \quad (3.8)$$

donde $\xi_0 = \frac{2\pi}{X_0}$.

De este modo la señal de entrada en el dominio transformado se obtendrá a partir de la señal definida en el dominio real bajo la expresión mostrada en 3.8.

Por otra parte, el conjunto de funciones núcleo que definen al operador en el dominio transformado se obtiene a partir el conjunto de respuestas que le caracterizan en el dominio real $h(x; x')$. De modo que:

$$H(m; m') = \mathbf{DSF}_{x'}[\mathbf{DSF}_x[h(x; x')]] = \frac{1}{X_0^2} \int_{x'} \int_x h(x; x') e^{-jm\xi_0 x} e^{-jm'\xi_0 x'} dx dx'. \quad (3.9)$$

Finalmente, la señal resultante se obtiene a partir de las dos expresiones anteriores:

$$b(m) = X_0 \sum_{m'} a(m') H(m; -m'). \quad (3.10)$$

Cuando el sistema es lineal e invariante, los cálculos se simplifican. En este caso, el conjunto de respuestas al impulso $h(x; x')$ se transforma en un conjunto de señales de una sola variable desplazadas, esto es, $h(x - x')$. Por lo tanto, el cálculo del conjunto de funciones núcleo que representan al operador en el dominio espectral se reduce a:

$$H(m; m') = \mathbf{DSF}_{x'}[\mathbf{DSF}_x[h(x - x')]] = \frac{1}{X_0^2} \int_{x'} \int_x h(x - x') e^{-jm\xi_0 x} e^{-jm'\xi_0 x'} dx dx' \quad (3.11)$$

$$\begin{aligned} \text{Si } u &= x - x' \\ H(m; m') &= \int_{x'} \int_u h(u) e^{-jm\xi_0(u+x')} e^{-jm'\xi_0 x'} du dx' = \\ &= \int_{x'} e^{-j(m+m')\xi_0 x'} dx' \int_u h(u) e^{-jm\xi_0 u} du = \frac{1}{X_0} \delta(m + m') H(m). \end{aligned} \quad (3.12)$$

Ahora se puede desarrollar la expresión 3.10 con los resultados obtenidos de modo que:

$$b(m) = X_0 \sum_{m'} a(m') \frac{1}{X_0} \delta(m - m') H(m) = a(m) H(m). \quad (3.13)$$

Análisis espectral generalizado bajo la transformada discreta de Fourier para el espacio de señales complejas de variable entera definidas para todo valor: $D(-\infty, \infty)$. En este caso el análisis se realiza bajo la transformada discreta de Fourier. Su definición en este espacio es la siguiente:

$$X(\Omega) = \sum_n x(n) e^{-j\Omega n} dx. \quad (3.14)$$

De este modo la señal de entrada en el dominio transformado se obtendrá a partir de la señal definida en el dominio real bajo la expresión mostrada en 3.14.

Por otra parte, el conjunto de funciones núcleo que definen al operador en el dominio transformado se obtiene a partir el conjunto de respuestas que le caracterizan en el dominio real $h(n; n')$. De modo que:

$$H(\Omega; \Omega') = \mathbf{DTF}_{n'}[\mathbf{DTF}_n[h(n; n')]] = \sum_{n'} \sum_n h(n; n') e^{-j\Omega' n'} e^{-j\Omega n}. \quad (3.15)$$

Finalmente, la señal resultante se obtiene a partir de las dos expresiones anteriores:

$$Y(\Omega) = \frac{1}{2\pi} \int_{\Omega'} X(\Omega') H(\Omega; -\Omega') d\Omega'. \quad (3.16)$$

Cuando el sistema es lineal e invariante, los cálculos se simplifican. En este caso, el conjunto de respuestas al impulso $h(n; n')$ se transforma en un conjunto de señales de una sola variable desplazadas, esto es, $h(n - n')$. Por lo tanto, el cálculo del conjunto de funciones núcleo que representan al operador en el dominio espectral se reduce a:

$$H(\Omega; \Omega') = \mathbf{DTF}_{n'}[\mathbf{DTF}_n[h(n - n')]] = \sum_{n'} \sum_n h(n - n') e^{-j\Omega' n'} e^{-j\Omega n}. \quad (3.17)$$

$$\begin{aligned} \text{Si } l &= n - n' \\ H(\Omega; \Omega') &= \sum_{n'} \sum_l h(l) e^{-j\Omega(l+n')} e^{-j\Omega' n'} = \sum_{n'} e^{-j(\Omega+\Omega')n'} \sum_l h(l) e^{-j\Omega l} = 2\pi\delta(\Omega + \Omega') H(\Omega). \end{aligned}$$

Ahora se puede desarrollar la expresión 3.16 con los resultados obtenidos de modo que:

$$Y(\Omega) = \frac{1}{2\pi} \int_{\Omega'} X(\Omega') 2\pi\delta(\Omega + \Omega') H(\Omega) d\Omega' = X(\Omega) H(\Omega). \quad (3.18)$$

Análisis espectral generalizado bajo el desarrollo en serie de Fourier para el espacio de señales complejas de variable entera con periodo N_0 : $D(N_0)$. En este caso el análisis se realiza bajo el desarrollo en serie de Fourier. Su definición en este espacio es la siguiente:

$$a(k) = \frac{1}{N_0} \sum_n x_0(n) e^{-jk\Omega_0 n} dx, \quad (3.19)$$

donde $\Omega_0 = \frac{2\pi}{N_0}$.

De este modo la señal de entrada en el dominio transformado se obtendrá a partir de la señal definida en el dominio real bajo la expresión mostrada en 3.19.

Por otra parte, el conjunto de funciones núcleo que definen al operador en el dominio transformado se obtiene a partir el conjunto de respuestas que le caracterizan en el dominio real $h(n; n')$. De modo que:

$$H(k; k') = \mathbf{DSF}_{n'}[\mathbf{DSF}_n[h(n; n')]] = \frac{1}{N_0^2} \sum_{n'} \sum_n h(n; n') e^{-jk'\Omega_0 n'} e^{-jk\Omega_0 n}. \quad (3.20)$$

Finalmente, la señal resultante se obtiene a partir de las dos expresiones anteriores:

$$b(k) = N_0 \sum_{k'} a(k') H(k; -k'). \quad (3.21)$$

Cuando el sistema es lineal e invariante, los cálculos se simplifican. En este caso, el conjunto de respuestas al impulso $h(n; n')$ se transforma en un conjunto de señales de una sola variable desplazadas, esto es, $h(n - n')$. Por lo tanto, el cálculo del conjunto de funciones núcleo que representan al operador en el dominio espectral se reduce a:

$$H(k; k') = \mathbf{DSF}_{n'}[\mathbf{DSF}_n[h(n - n')]] = \frac{1}{N_0^2} \sum_{n'} \sum_n h(n - n') e^{-jk'\Omega_0 n'} e^{-jk\Omega_0 n}. \quad (3.22)$$

$$\begin{aligned} \text{Si } l &= n - n' \\ H(k; k') &= \frac{1}{N_0^2} \sum_{n'} \sum_l h(l) e^{-jk'\Omega_0 n'} e^{-jk\Omega_0 (l+n')} = \\ &= \frac{1}{N_0^2} \sum_{n'} e^{-j(k+k')\Omega_0 n'} \sum_l h(l) e^{-jk\Omega_0 l} = \frac{1}{N_0} \delta(k + k') H(k). \end{aligned} \quad (3.23)$$

Ahora se puede desarrollar la expresión 3.21 con los resultados obtenidos de modo que:

$$b(k) = N_0 \sum_{k'} a(k') \frac{1}{N_0} \delta(k - k') H(k) = a(k) H(k). \quad (3.24)$$

3.3. Operadores.

Con el objetivo de desarrollar una aplicación de gran versatilidad, se pone a disposición del usuario una amplia gama de operadores con los que experimentar. Los operadores disponibles se encuentran en los cinco espacios de señales explicados anteriormente, aunque algunos de ellos no estarán disponibles en todos los espacios. Son utilizados para definir el comportamiento de un sistema de comunicaciones. Aunque es cierto que existen operadores que necesitan dos señales definidas por el usuario siguen constituyendo un operador, ya que la segunda señal es la que caracteriza el sistema. Cuando se trabaja con operadores el usuario podrá estudiar las características y propiedades del sistema.

Los operadores disponibles son:

- Suma.
- Resta.
- Multiplicación.
- Desplazamiento.
- Convolución.
- Correlación.
- Identidad.
- Escalado.
- Transformada de Fourier.
- Transformada inversa de Fourier.

- Transformada de Hilbert.
- Transformada inversa de Hilbert.
- Transformada de Bessel.
- Transformada inversa de Bessel.
- Modulador en Amplitud.

Los distintos operadores transformada no estarán disponibles en todos los espacios de definición. Por ejemplo, la transformada de Hilbert sólo estará disponible en el espacio de señales continuas aperiódicas y la de Bessel en el de señales continuas aperiódicas definidas para valores positivos.

A continuación se explicará en detalle cada uno de los operadores disponibles en el programa.

3.3.1. Operadores suma y resta.

Funcionamiento interno. Tanto el operador suma como el operador resta realizan la misma operación: suman dos señales. La única diferencia entre ellos es que la resta cambia el signo de la segunda señal de signo de realizar la operación.

Estos operadores están disponibles para todos los espacios de partida del software y tienen un modo de resolución en función del tipo de señales con las que vaya a operar. A continuación se presenta un cuadro donde se muestra el modo de resolución de la operación (numérico o teórico) según el tipo de señal de entrada (analítica o numérica):

Señal 1	Señal 2	Modo de resolución
Analítica	Analítica	Teórico
	Numérica	
Numérica	Analítica	
	Numérica	Numérico

Como se observa, la señal de salida sólo se obtendrá de modo numérico cuando las dos señales que intervienen sean numéricas, es decir, ambas estén definidas para un soporte finito determinado por el usuario. En este caso, se obtendrá el rango del resultado (aquel que contenga los dos rangos iniciales) y sumarán punto a punto las señales de entrada para dicho soporte. Esto se puede realizar teniendo en cuenta que las señales numéricas toman el valor cero fuera del soporte finito que se haya definido.

La resolución del sistema será teórica siempre que una o las dos señales que se van a sumar sean analíticas.

Cuando el espacio de partida sea aperiódico, lo primero que se deberá hacer una distinción entre las operaciones con señales analíticas de energía finita y las señales analíticas de energía infinita:

- En el primer caso, señales analíticas de energía finita, se definirá un soporte finito para la señal de modo que fuera de él la señal no tome valores significativos. Una vez definido dicho soporte se operaría como con las señales numéricas: se calcula el rango de definición de la señal resultado y se opera punto a punto.
- En el segundo caso, cuando las señales son analíticas de energía infinita, el modo de operar cambia. Como no se puede definir un soporte finito para ellas, el programa simplemente

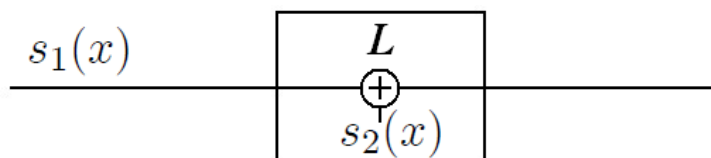


Figura 3.2: Esquema del operador suma para el espacio de partida de señales complejas de variable real definidas para todo valor $S(-\infty, \infty)$. Como se observa, la segunda señal, $s_2(x)$, es una señal interna al sistema y no una señal de entrada (como ocurriría en la operación básica suma). Por lo tanto, la segunda señal define al operador suma. Una vez que dicha señal esté totalmente definida se pasará de trabajar con la familia de operadores suma a trabajar con un operador concreto.

opera punto a punto con las señales en el rango de visualización en que nos encontremos. Es decir, la operación se llevará a cabo acotándola según los límites establecidos por el usuario para la gráfica correspondiente del interfaz de Visualización.

Por otro lado, cuando la operación se realice en espacios de partida periódicos, el modo teórico de resolver la operación cambia. Al estar en un espacio periódico las operaciones se reducen y simplemente se opera en un periodo completo del resultado. Esto se ve además muy simplificado ya que el programa sólo permite introducir un valor de periodo para todas las señales de entrada por lo que el periodo del resultado será el mismo y no hará falta calcularlo.

Análisis del operador en el dominio real. Al escoger un operador, el usuario analiza el comportamiento de un sistema de comunicaciones y podrá estudiar y aprender muchas de sus características gracias a controles gráficos, carteles informativos y visualizaciones. Las características principales del operador suma / resta son las siguientes:

- Este operador **suma o resta una señal a la señal de entrada**, véase la fig. 3.2. Tanto la señal de entrada como la señal que caracteriza el sistema, segunda señal, son definidas por el usuario.
- Este operador es, en realidad, una **familia de operadores**. Se debe recordar que un operador devuelve, para la misma señal de entrada, siempre la misma salida. Sin embargo, el operador suma puede devolver una salida distinta para la misma señal de entrada cuando la señal que caracteriza al operador, la que se suma a la señal de entrada, sea diferente. Por tanto, el operador sólo estará completamente caracterizado cuando la segunda señal esté definida.

El usuario podrá analizar también las propiedades del sistema estudiado:

- Como se demuestra a continuación, este operador **no es lineal**.

Dada una señal de entrada su paso por el sistema se obtiene:

$$f_1(x) \rightarrow g_1(x) = f_1(x) + s_2(x). \quad (3.25)$$

Lo mismo ocurre si se introduce otra señal de entrada diferente:

$$f_2(x) \rightarrow g_2(x) = f_2(x) + s_2(x). \quad (3.26)$$

Si como entrada al sistema se introduce una combinación lineal de las dos señales de entrada mencionadas previamente, se obtiene:

$$\alpha f_1(x) + \beta f_2(x) \rightarrow g_3(x) = \alpha f_1(x) + \beta f_2(x) + s_2(x) \neq \alpha g_1(x) + \beta g_2(x). \quad (3.27)$$

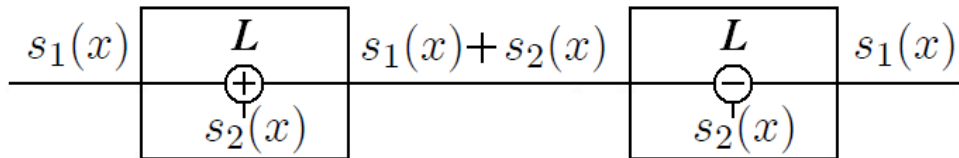


Figura 3.3: Esquema en el que se representa que el operador suma es invertible. Simplemente hay que restar la señal del sistema a la señal resultado para obtener de nuevo la señal de entrada, es decir, volver a sumar la señal del sistema cambiada de signo.

Dado que el operador no es lineal, no existe ningún conjunto de señales que lo caracterice, es decir, no dispone de un conjunto de respuestas al impulso.

- Este operador **no es invariante**. Esto se demuestra de la siguiente forma:

Dada una señal de entrada, definida como se muestra en (3.25). Si se introduce en el sistema esta señal desplazada o retardada se obtiene que:

$$f_1(x - x_0) \rightarrow g_4(x) = f_1(x - x_0) + s_2(x) \neq g_1(x - x_0). \quad (3.28)$$

- La operación suma se realiza punto a punto, es decir, la salida para cada instante no depende de valores de la señal de entrada en instantes diferentes de tiempo. Esto quiere decir que el operador **no tiene memoria**. Poner desarrollo matemático.
- Por el mismo motivo que se demuestra que el operador no tiene memoria se sabe que **es causal**, ya que la salida del sistema no depende de los valores futuros que tome la entrada.
- Otra propiedad fácil de comprobar es la **invertibilidad** del sistema, ya que recuperar la señal de entrada es tan simple como sumarle a la señal de salida la señal que caracteriza el sistema cambiada de signo. Esto se representa en el esquema 3.3.
- Finalmente, **la propiedad de estabilidad dependerá de la señal que caracterice al sistema**, si es acotada, la señal de salida también lo será, siempre y cuando la señal de entrada lo sea, y el sistema será estable.

Análisis espectral del operador. Dado que el operador suma no es lineal no dispone de un conjunto de respuestas al impulso que le caracterice en el dominio real y, por tanto, no se puede obtener un conjunto de funciones núcleo que definan al operador en el dominio espectral.

3.3.2. Operador Multiplicación.

Funcionamiento interno. El operador multiplicación sólo se puede implementar de forma numérica. Para los cinco espacios de señales disponibles esto se resume en la siguiente tabla:

Señal 1	Señal 2	Modo de resolución
Analítica	Analítica	No disponible
	Numérica	
Numérica	Analítica	
	Numérica	Numérico

Como se observa, el sistema sólo permite calcular la señal de salida cuando las dos señales que intervienen, la señal de entrada y la señal que define el sistema, son de tratamiento numérico.

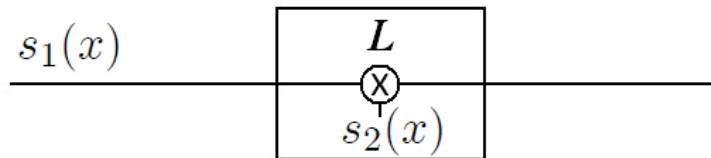


Figura 3.4: Esquema que representa al operador multiplicación. Como se observa la señal de entrada siempre es multiplicada por una misma señal, $s_2(x)$, que define al operador. Mientras dicha señal esté sin definir, se hablará de la familia de operadores multiplicación. Una vez $s_2(x)$ esté totalmente definida, se estará trabajando con un operador concreto. Hay ejemplos típicos de operadores que pertenecen a esta familia como puede ser el modulador en amplitud, que no es más que un operador multiplicación donde $s_2(x) = A \cos(\omega x + \phi)$.

Una vez establecido esto, el modo obtener la señal de salida es muy similar al de los operadores suma y resta en modo numérico: se define el rango de la señal resultado y se opera punto a punto.

Análisis del operador en el dominio real. Las características principales del operador multiplicación son las siguientes:

- Este operador **multiplica una señal por la señal de entrada**, véase la fig. 3.4. Tanto la señal de entrada como la señal que caracteriza el sistema, segunda señal, son definidas por el usuario.
- Este operador es, en realidad, una **familia de operadores**. Se debe recordar que un operador devuelve, para la misma señal de entrada, siempre la misma salida. Sin embargo, el operador multiplicación puede devolver una salida distinta para la misma señal de entrada cuando la señal que caracteriza al operador, la que multiplica a la señal de entrada, sea diferente. Por tanto, el operador sólo estará completamente caracterizado cuando la segunda señal esté definida. Un ejemplo de operador de esta familia es el modulador en amplitud, que multiplica la señal de entrada por un coseno, y que también está disponible en el programa. El usuario podrá comprobar que con la misma señal de entrada obtendrá la misma salida utilizando este operador y definiendo como segunda señal un coseno y utilizando el operador modulador en amplitud (siempre teniendo en cuenta que los parámetros del coseno sean los mismos).

El usuario podrá estudiar, también, las propiedades del sistema analizado:

- Como se demuestra a continuación, este operador **es lineal**:

Dada una señal de entrada su paso por el sistema se obtiene:

$$f_1(x) \rightarrow g_1(x) = f_1(x)s_2(x). \quad (3.29)$$

Lo mismo ocurre si se introduce otra señal de entrada diferente:

$$f_2(x) \rightarrow g_2(x) = f_2(x)s_2(x). \quad (3.30)$$

Si como entrada al sistema se introduce una combinación lineal de las dos señales de entrada mencionadas previamente, se obtiene:

$$\begin{aligned} \alpha f_1(x) + \beta f_2(x) &\rightarrow g_3(x) = [\alpha f_1(x) + \beta f_2(x)]s_2(x) = \\ &= \alpha f_1(x)s_2(x) + \beta f_2(x)s_2(x) = \alpha g_1(x) + \beta g_2(x). \end{aligned} \quad (3.31)$$

Como el operador es lineal, se puede representar su conjunto de respuestas al impulso, véase la fig. 3.5.

- Este operador **no es invariante**. Esto se demuestra de la siguiente forma:

Dada una señal de entrada, definida como se muestra en (3.29). Si se introduce en el sistema esta señal desplazada o retardada se obtiene que:

$$f_1(x - x_0) \rightarrow g_4(x) = f_1(x - x_0)s_2(x) \neq g_1(x - x_0) = f_1(x - x_0)s_2(x - x_0). \quad (3.32)$$

Como el operador no es invariante el conjunto de respuestas al impulso es necesario para caracterizar su comportamiento, no está compuesto de una misma señal desplazada. Esto se puede observar la representación del conjunto de respuestas al impulso, fig. 3.5.

- La operación multiplicación se realiza punto a punto, es decir, la salida para cada instante no depende de valores de la señal de entrada en instantes diferentes de tiempo. Esto quiere decir que el operador **no tiene memoria**.

$$f_1(x_1) \rightarrow g_1(x_1) = f_1(x_1)s_2(x_1). \quad (3.33)$$

De nuevo esta propiedad se puede observar en la fig. 3.5. Como se observa el conjunto de respuestas al impulso está compuesto de un tren de deltas en la recta $\tau = \tau'$. Para una explicación matemática que le permita comprender mejor esta propiedad, puede consultar [2].

- Por el mismo motivo que se demuestra que el operador no tiene memoria se sabe que **es causal**, ya que la salida del sistema no depende de los valores futuros que tome la entrada.
- En general, se considerará que el sistema **no es invertible**. Esto se debe a que no siempre será posible multiplicar la señal de salida por la inversa de la señal del sistema, siempre que la señal tenga discontinuidades o nulos.
- Finalmente, la propiedad de **estabilidad dependerá de la señal que caracterice al sistema**, si ésta es acotada, la señal de salida también lo será, siempre y cuando la señal de entrada lo sea, y el sistema será estable.

En la siguiente tabla se muestra el conjunto de respuestas al impulso del operador multiplicación para los distintos espacios de partida del software:

Espacio de partida	Conjunto de respuestas al impulso del op. multiplicación
$S(-\infty; \infty)$	$h(x; x') = s_2(x)\delta(x - x') = s_2(x')\delta(x - x')$
$P(X_0)$	$h(x; x') = s_2(x)\delta_0(x - x') = s_2(x')\delta_0(x - x')$
$D(-\infty; \infty)$	$h(n; n') = s_2(n)\delta(n - n') = s_2(n')\delta(n - n')$
$D(N_0)$	$h(n; n') = s_2(n)\delta_0(n - n') = s_2(n')\delta_0(n - n')$

(3.34)

Análisis espectral del operador. Al tratarse de un operador lineal existe un conjunto de respuestas al impulso que lo caracteriza en el dominio real y existe, por tanto, un conjunto de funciones núcleo o funciones de transferencia que lo caracterizan en el dominio espectral.

En la siguiente tabla se muestra el conjunto de funciones de transferencia del operador multiplicación para cada uno de los espacios de partida en los que está implementado, téngase en cuenta que.

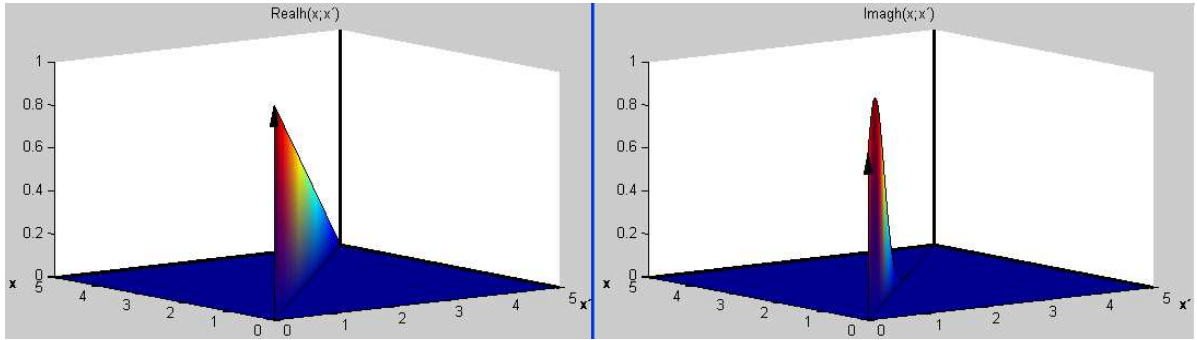


Figura 3.5: Conjunto de respuestas al impulso del operador multiplicación en el espacio de señales continuas aperiódicas cuando la señal que le caracteriza está compuesta por un pulso triangular en la parte real y un pulso gaussiano en la parte imaginaria. Visualizando el conjunto de respuestas al impulso se pueden apreciar las distintas propiedades del operador: su linealidad (en caso contrario no existiría el conjunto de respuestas al impulso representado), su no invarianza (ya que no está compuesta por la misma señal desplazada y centrada en la recta $\tau = \tau'$), su falta de memoria y su causalidad (como se ve se compone por un tren de deltas en el eje $\tau = \tau'$, es decir, para cada instante de la señal de salida sólo se tiene en cuenta la señal de entrada en dicho valor).

Espacio de partida	Conjunto de funciones de transferencia del op. multiplicación
$S(-\infty; \infty)$ bajo la TF	$H(\xi; \xi') = S_2(\xi + \xi')$
$P(X_0)$ bajo el DSF	$H(m; m') = \frac{1}{X_0} S_2(m + m')$
$D(-\infty; \infty)$ bajo la DTF	$H(\Omega; \Omega') = \sum_m S_2(\Omega + \Omega' - 2\pi m)$
$D(N_0)$ bajo el DSF	$H(k; k') = \frac{1}{N_0} S_2(k + k')$

(3.35)

A continuación se expone, en detalle, el cálculo desarrollado para obtener las expresiones anteriores:

- **Espacio de señales de variable continua definidas para todo valor** $S(-\infty; \infty)$.

$$H(\xi; \xi') = \mathbf{TF}_{x'}[\mathbf{TF}_x[s_2(x')\delta(x - x')]] = \int_{x'=-\infty}^{\infty} \int_{x=-\infty}^{\infty} s_2(x')\delta(x - x')e^{-j\xi x} e^{-j\xi' x'} dx dx'. \quad (3.36)$$

Se calcula primero la transformada respecto a x :

$$\mathbf{TF}_x[s_2(x')\delta(x - x')] = s_2(x')\mathbf{TF}_x[\delta(x - x')] = s_2(x')e^{-j\xi x'}. \quad (3.37)$$

Se calcula ahora la transformada respecto a x' , haciendo uso de la propiedad que cumple que un producto de señales en el dominio real equivale a una convolución escalada de las señales en el dominio espectral:

$$\mathbf{TF}_{x'}[s_2(x')e^{-j\xi x'}] = \frac{1}{2\pi} \mathbf{TF}_{x'}[s_2(x')] * \mathbf{TF}_{x'}[e^{-j\xi x'}], \quad (3.38)$$

donde:

$$\mathbf{TF}_{x'}[e^{-j\xi x'}] = 2\pi\delta(\xi + \xi'). \quad (3.39)$$

De modo que se obtiene el conjunto de funciones de transferencia como:

$$H(\xi; \xi') = S_2(\xi) * \delta(\xi + \xi') = S_2(\xi + \xi'). \quad (3.40)$$

- **Espacio de señales de variable continua periódicas** $P(X_0)$.

$$H(m; m') = \mathbf{DSF}_{m'}[\mathbf{DSF}_m[s_2(x')\delta_0(x - x')]] = \quad (3.41)$$

$$= \frac{1}{X_0^2} \int_{x'} \int_x s_2(x')\delta_0(x - x')e^{-jm\xi_0x}e^{-jm'\xi_0x'} dx dx'. \quad (3.42)$$

Se calcula primero la transformada respecto a x :

$$\mathbf{DSF}_m[s_2(x')\delta_0(x - x')] = s_2(x')\mathbf{DSF}_m[\delta_0(x - x')] = s_2(x')\frac{1}{X_0}e^{-jm\xi_0x'}. \quad (3.43)$$

Se calcula ahora la transformada respecto a x' , haciendo uso de la propiedad que cumple que un producto de señales en el dominio real equivale a una convolución escalada de las señales en el dominio espectral:

$$\mathbf{DSF}_{x'}[s_2(x')\frac{1}{X_0}e^{-jm\xi_0x'}] = \frac{1}{X_0} \sum_k S_2(m' - k)\mathbf{DSF}_{m'}[e^{-jm\xi_0x'}], \quad (3.44)$$

donde:

$$S_2(k) = \mathbf{DSF}_{m'}[s_2(x')], \quad (3.45)$$

$$\mathbf{DSF}_{m'}[e^{-jm\xi_0x'}] = \begin{cases} 1, & k = -m \\ 0, & k \neq -m \end{cases}.$$

De modo que se obtiene el conjunto de funciones de transferencia como:

$$H(m; m') = \frac{1}{X_0} S_2(m + m'). \quad (3.46)$$

- **Espacio de señales de variable entera definidas para todo valor** $D(-\infty; \infty)$.

$$H(\Omega; \Omega') = \mathbf{DTF}_{n'}[\mathbf{DTF}_n[s_2(n')\delta(n - n')]] = \sum_{n'} \sum_n s_2(n')\delta(n - n')e^{-j\Omega'n'}e^{-j\Omega n}. \quad (3.47)$$

Se calcula primero la transformada respecto a n :

$$\mathbf{DTF}_n[s_2(n')\delta(n - n')] = s_2(n')\mathbf{DTF}_n[\delta(n - n')] = s_2(n')e^{-j\Omega n'}. \quad (3.48)$$

Se calcula ahora la transformada respecto a n' , haciendo uso de la propiedad que cumple que un producto de señales en el dominio real equivale a una convolución escalada de las señales en el dominio espectral:

$$\mathbf{DTF}_{n'}[s_2(n')e^{-j\Omega n'}] = \frac{1}{2\pi} \mathbf{DTF}_{n'}[s_2(n')] * \mathbf{DTF}_{n'}[e^{-j\Omega n'}], \quad (3.49)$$

donde:

$$\mathbf{DTF}_{n'}[e^{-j\Omega n'}] = 2\pi \sum_m \delta(\Omega + \Omega' - 2\pi m). \quad (3.50)$$

De modo que se obtiene el conjunto de funciones de transferencia como:

$$H(\Omega; \Omega') = \sum_m S_2(\Omega + \Omega' - 2\pi m). \quad (3.51)$$

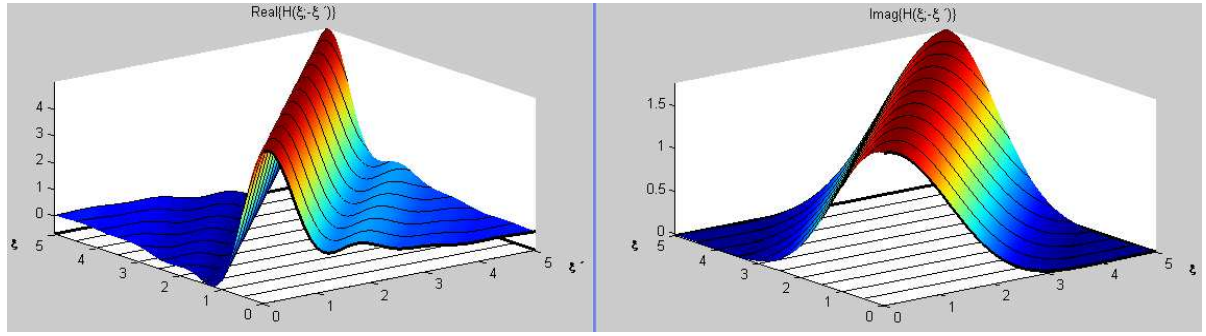


Figura 3.6: Conjunto de funciones núcleo que caracterizan a uno de los operadores de la familia multiplicación en el dominio transformado, en este caso bajo la transformada de Fourier.

- **Espacio de señales de variable entera periódicas** $D(N_0)$.

$$H(k; k') = \mathbf{DSF}_{n'}[\mathbf{DSF}_n[s_2(n')\delta_0(n - n')]] = \frac{1}{N_0^2} \sum_{n'} \sum_n s_2(n')\delta_0(n - n')e^{-jk'\Omega_0n'} e^{-jk\Omega_0n}. \quad (3.52)$$

Se calcula primero la transformada respecto a n :

$$\mathbf{DSF}_n[s_2(n')\delta_0(n - n')] = s_2(n')\mathbf{DSF}_n[\delta_0(n - n')] = s_2(n')\frac{1}{N_0}e^{-jk\Omega_0n'}. \quad (3.53)$$

Se calcula ahora la transformada respecto a n' , haciendo uso de la propiedad que cumple que un producto de señales en el dominio real equivale a una convolución escalada de las señales en el dominio espectral:

$$\mathbf{DSF}_{n'}[s_2(n')\frac{1}{N_0}e^{-jk\Omega_0n'}] = \frac{1}{N_0} \sum_m S_2(k' - m)\mathbf{DSF}_{k'}[e^{-jk\Omega_0n'}], \quad (3.54)$$

donde:

$$\begin{aligned} S_2(k') &= \mathbf{DSF}_{k'}[s_2(n')], \\ \mathbf{DSF}_{k'}[e^{-jk\Omega_0n'}] &= \begin{cases} 1, m = -k + nN_0m \\ 0, m \neq -k + nN_0m \end{cases}. \end{aligned} \quad (3.55)$$

De modo que se obtiene el conjunto de funciones de transferencia como:

$$H(k; k') = \frac{1}{N_0} S_2(k + k'). \quad (3.56)$$

En la fig. 3.6 se muestra un ejemplo del conjunto de funciones de transferencia que caracterizan al operador en el dominio espectral.

3.3.3. Operador convolución.

Funcionamiento interno. Este operador se utiliza para obtener el resultado de la operación convolución entre dos señales: la señal de entrada al sistema y señal que caracteriza al operador (que en este caso es la respuesta al impulso del sistema).

El operador convolución esta disponible en los cinco espacios de partida y la metodología seguida para realizar el cálculo, en función de la naturaleza de las señales involucradas para obtener la señal resultado, es la misma en todos ellos, véase la siguiente tabla:

Señal 1	Señal 2	Modo de resolución
Analítica	Analítica	Teórico
	Numérica	
Numérica	Analítica	
	Numérica	Numérico

Cuando las dos señales con las que se opera (señal de entrada y respuesta al impulso del sistema) son de carácter numérico, la operación se resuelve numéricamente. Esto se traduce en que el programa implementa una subrutina que contiene el método numérico para resolver la operación.

En caso contrario, alguna de las dos señales consideradas sea analítica, la operación se resuelve de forma teórica. En este supuesto surgen dos casos de análisis: las señales son analíticas de energía finita o las señales son analíticas de energía infinita.

En el primer caso, señales analíticas de energía finita, el programa definirá un rango para la señal que este utilizando. Este rango se definirá de modo que no se pierda información de la señal tomando ésta valores no significativos fuera de él. Una vez definido el rango de la señal pasará a tratarse numéricamente implementando el método numérico correspondiente. En el caso de que se trabaje con señales periódicas, se acotarán las señales para un período y se operará en él, facilitando los cálculos necesarios.

Si se trata de señales analíticas de energía infinita, el programa no podrá resolverlas por sí mismo. En este caso, los resultados de las distintas convoluciones estarán almacenados en una base de datos. Como es lógico, estos resultados habrán sido calculados previamente por los programadores siguiendo la expresión matemática mostrada en la siguiente tabla, 3.57, correspondiente a cada espacio de entrada. Cuando el programa tenga que calcular una de éstas convoluciones accederá a la base de datos y, si ésta contiene el resultado deseado, lo mostrará. En caso contrario se avisará de que el resultado no está disponible. Para conocer cuales son los resultados contenidos en las bases de datos del software puede consultar [4].

Espacio de señales	Expresión de la convolución
$S(-\infty, \infty)$ y $S(0, \infty)$	$c(x) = f(x) * g(x) = \int_{-\infty}^{\infty} f(x')g(x - x')dx'$
$P(X_0)$	$c_0(x) = f_0(x) * g_0(x) = \int_{\langle X_0 \rangle} f_0(x')g_0(x - x')dx'$
$D(-\infty, \infty)$	$x(n) * y(n) = \sum_{m=-\infty}^{\infty} x(m)y(n - m)$
$D(N_0)$	$c_0(x) = x_0(n) * y_0(n) = \sum_{m=\langle N_0 \rangle} x_0(m)y_0(n - m)$

(3.57)

Análisis del operador en el dominio real. Gracias al hecho de que la convolución está definida en el software como un operador, el usuario analiza el comportamiento del sistema de comunicaciones definido por él y podrá estudiar y aprender muchas de sus características gracias a controles gráficos, carteles informativos y visualizaciones. Las características principales del operador multiplicación son las siguientes:

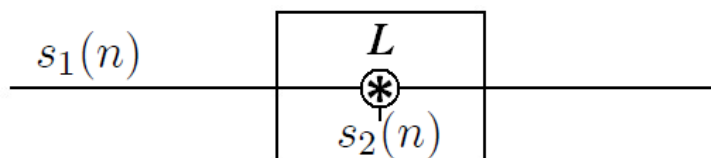


Figura 3.7: Esquema que representa al operador convolución. Como se observa la señal de entrada siempre es convolucionada por una misma señal, $s_2(x)$, que define al operador y que es la respuesta al impulso del sistema. Mientras dicha señal esté sin definir, se hablará de la familia de operadores convolución. Una vez $s_2(x)$ esté totalmente definida, se estará trabajando con un operador concreto caracterizado por un conjunto de respuestas al impulso. Por las propiedades de este operador, hablar de respuesta al impulso y conjunto de respuestas al impulso es indiferente, ya que el conjunto está formado por réplicas de la respuesta al impulso desplazadas y centradas en la recta $\tau = \tau'$. Hay ejemplos típicos de operadores que pertenecen a esta familia como puede ser el operador identidad donde $s_2(x) = h(x) = \delta(x)$.

- Este operador **convoluciona una señal, que es la respuesta al impulso del sistema, por la señal de entrada**, véase la fig. 3.7. Tanto la señal de entrada como la respuesta al impulso del sistema son definidas por el usuario.
- Este operador es, en realidad, una **familia de operadores**. Se debe recordar que un operador devuelve, para la misma señal de entrada, siempre la misma salida. Sin embargo, el operador convolución puede devolver una salida distinta para la misma señal de entrada cuando la señal que caracteriza al operador, la que convoluciona a la señal de entrada, sea diferente. Por tanto, el operador sólo estará completamente caracterizado cuando la respuesta al impulso esté definida. Un ejemplo de operador de esta familia es el escalado, que multiplica la señal por un valor constante, es decir, convoluciona la señal de entrada con una delta de la amplitud deseada. Otro ejemplo es el operador desplazamiento, que convoluciona la señal de entrada con una delta de Dirac desplazada. También se implementa en el software el operador identidad, que convoluciona la señal de entrada por una delta de amplitud unidad y desplazamiento cero. El usuario podrá comprobar que con la misma señal de entrada obtendrá la misma salida utilizando este operador y definiendo como segunda señal una delta con los parámetros adecuados (según quiera simular cada uno de los tres operadores definidos).

El usuario podrá estudiar, también, las propiedades del sistema estudiado.

- Como se demuestra a continuación, este operador **es lineal**.

Dada una señal de entrada su paso por el sistema se obtiene:

$$f_1(x) \rightarrow g_1(x) = \int_{-\infty}^{\infty} f_1(x')s_2(x-x')dx'. \quad (3.58)$$

Lo mismo ocurre si se introduce otra señal de entrada diferente:

$$f_2(x) \rightarrow g_2(x) = \int_{-\infty}^{\infty} f_2(x')s_2(x-x')dx'. \quad (3.59)$$

Si como entrada al sistema se introduce una combinación lineal de las dos señales de entrada mencionadas previamente, se obtiene:

$$\begin{aligned} \alpha f_1(x) + \beta f_2(x) &\rightarrow g_3(x) = \int_{-\infty}^{\infty} [\alpha f_1(x') + \beta f_2(x')]s_2(x-x')dx' = \\ &= \alpha \int_{-\infty}^{\infty} f_1(x')s_2(x-x')dx' + \beta \int_{-\infty}^{\infty} f_2(x')s_2(x-x')dx' = \alpha g_1(x) + \beta g_2(x). \end{aligned} \quad (3.60)$$

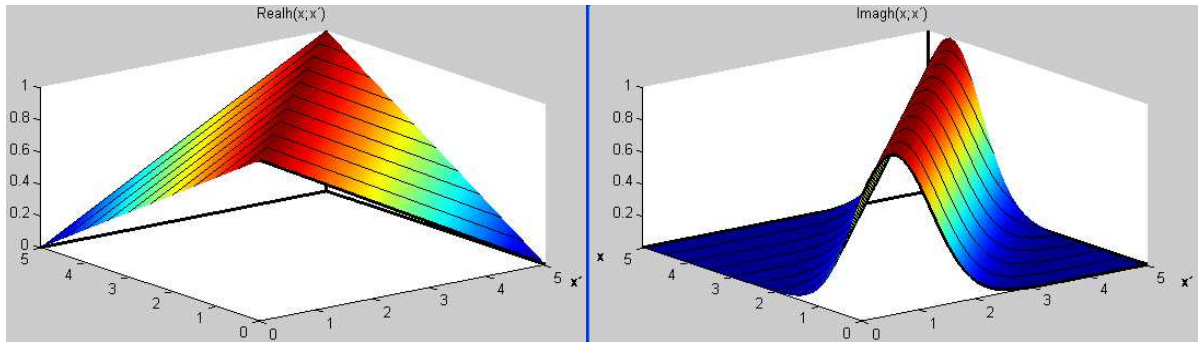


Figura 3.8: Conjunto de respuestas al impulso del operador convolución en el espacio de señales continuas aperiódicas cuando la señal que le caracteriza está compuesta por un pulso triangular en la parte real y un pulso gaussiano en la parte imaginaria. Visualizando el conjunto de respuestas al impulso se pueden apreciar las distintas propiedades del operador: su linealidad (en caso contrario no existiría el conjunto de respuestas al impulso representado), su invarianza (ya que el conjunto de respuestas al impulso está compuesto por la misma señal desplazada y centrada en la recta $\tau = \tau'$), su memoria (ya que se tienen en cuenta instantes anteriores y posteriores al considerado) y su causalidad (ya que se tienen en cuenta instantes posteriores al considerado para cada valor).

Como el operador es lineal, se puede representar su conjunto de respuestas al impulso, véase la fig. 3.8.

- Este operador **es invariante**. Esto se demuestra de la siguiente forma:

Dada una señal de entrada, definida como se muestra en (3.58). Si se introduce en el sistema esta señal desplazada o retardada se obtiene que:

$$f_1(x - x_0) \rightarrow g_4(x) = \int_{-\infty}^{\infty} f_1(x' - x_0) s_2(x - (x' - x_0)) dx' = g_1(x - x_0). \quad (3.61)$$

Como el operador es invariante el conjunto de respuestas al impulso se puede representar como una única función que se va desplazando y siempre se centra en la recta $\tau = \tau'$. Esta propiedad se puede observar la representación del conjunto de respuestas al impulso, fig. 3.8.

- Como se observa en la Tabla 3.57, para obtener el resultado de la convolución entre dos señales, para cada instante se tiene en cuenta toda la señal de entrada, incluyendo los instantes anteriores y posteriores al considerado. Por lo tanto, el operador **tiene memoria**.
- Por el mismo motivo que se demuestra que el operador tiene memoria se sabe que **no es causal**, ya que la salida del sistema depende de los valores futuros que tome la entrada.
- Además, el sistema **es invertible**, ya que es posible definir un operador que recupere la señal original.
- Finalmente, la propiedad de **estabilidad dependerá de la señal que caracterice al sistema**. El sistema será estable cuando la integral del módulo de dicha señal sea finita.

Análisis espectral del operador. Como ya se ha indicado, el operador convolución es un operador lineal e invariante, lo que quiere decir, tal y como se ha expuesto en el apartado 3.57, que el conjunto de funciones núcleo que definen al operador está compuesto por un tren de deltas en $\mu = \mu'$ escalado por la transformada de la respuesta al impulso del sistema $H(\mu)$.

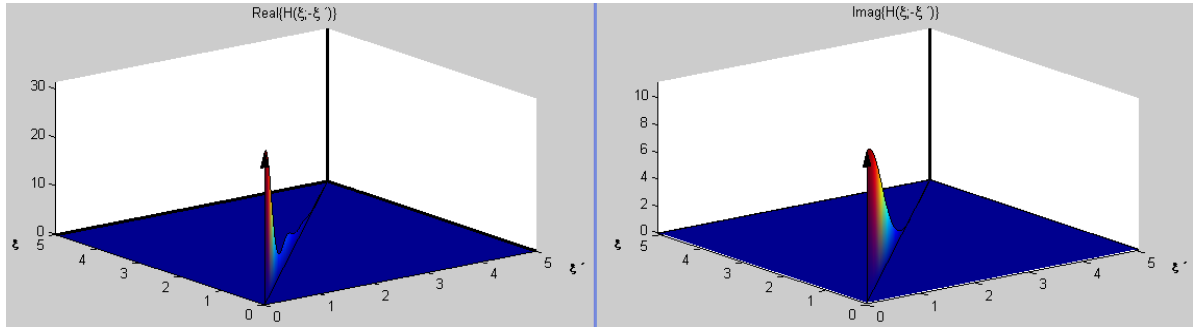


Figura 3.9: Conjunto de funciones núcleo que caracterizan a uno de los operadores de la familia convolución en el dominio transformado, en este caso bajo la transformada de Fourier.

Además, en este caso particular, la respuesta al impulso del sistema es la propia señal definida por el usuario para caracterizar el operador, $s_2(\tau)$, por lo que la función de transferencia del sistema no será más que la transformada de dicha señal, $S_2(\mu)$. En la siguiente tabla se muestra, a modo de resumen, la expresión genérica del conjunto de funciones que caracterizan al operador en el dominio espectral para los espacios de partida del software:

Espacio de partida	Conjunto de funciones de transferencia del op. convolución
$S(-\infty; \infty)$ bajo la TF	$H(\xi; \xi') = 2\pi\delta(\xi + \xi')S_2(\xi)$
$P(X_0)$ bajo el DSF	$H(m; m') = \frac{1}{X_0}\delta(m + m')S_2(m)$
$D(-\infty; \infty)$ bajo la DTF	$H(\Omega; \Omega') = 2\pi\delta(\Omega + \Omega')S_2(\Omega)$
$D(N_0)$ bajo el DSF	$H(k; k') = \frac{1}{N_0}\delta(k + k')S_2(k)$

(3.62)

En la fig. 3.9 se muestra un ejemplo del conjunto de funciones de transferencia que caracterizan al operador en el dominio espectral.

3.3.4. Operador correlación.

Funcionamiento interno. Otro de los operadores disponibles en el programa es el operador correlación. Este operador realiza una operación de convolución característica por lo que puede verse como un caso particular del operador convolución.

Este operador, al igual que la convolución, está disponible en los cinco espacios de partida del software y resuelve la operación del mismo modo que la convolución, comportamiento resumido en la siguiente tabla:

Señal 1	Señal 2	Modo de resolución
Analítica	Analítica	Teórico
	Numérica	
Numérica	Analítica	Numérico
	Numérica	

Cuando las señales de entrada al sistema sean de carácter numérico, la operación se resolverá

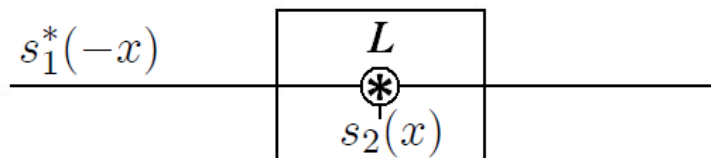


Figura 3.10: Esquema que representa al operador correlación. Como se observa la señal de entrada, que se conjuga y abate, siempre es convolucionada por una misma señal, $s_2(x)$, que define al operador. Mientras dicha señal esté sin definir, se hablará de la familia de operadores correlación. Una vez $s_2(x)$ esté totalmente definida, se estará trabajando con un operador concreto caracterizado por un conjunto de respuestas al impulso.

de modo numérico. Es decir, el programa implementará la subrutina que contenga el método numérico correspondiente. Dicho método variará en función del espacio de señales en el que se esté trabajando.

Cuando alguna de las señales que intervengan sea analítica, el programa resolverá la operación de forma teórica. En este supuesto, si el programa puede realizar una buena aproximación podrá utilizar el método numérico apoyado en dichas aproximaciones para resolver la operación. En caso contrario, la única forma de resolver la operación es mediante la ecuación matemática correspondiente y en papel. Después se almacenarían dichos valores en una base de datos para su posterior utilización. Sin embargo, esta versión del programa no cuenta con dicha base de datos para ninguno de los espacios de señales existentes. Es decir, en este caso todas las combinaciones posibles se resolverán numéricamente y las demás no tendrán solución.

Análisis del operador en el dominio real. Al igual que ocurre con todos los operadores del software, el usuario podrá aprender las características del operador correlación cuando trabaje con él. Ya que el operador estudiado es un caso particular del operador convolución sólo se mostrarán a continuación las características que no sean idénticas a las del operador mencionado:

- Este operador **correlaciona una señal con la señal de entrada**, véase la fig.3.10 . Tanto la señal de entrada como la señal que caracteriza al operador son definidas por el usuario.
- Este operador es, en realidad, una **familia de operadores**, ya que el operador sólo estará completamente caracterizado cuando la señal que le caracteriza esté totalmente definida.

Análisis espectral del operador. Como ya se ha indicado, este operador se puede ver como un caso particular de la convolución y, por tanto, su análisis espectral no difiere con el del operador convolución.

3.3.5. Operador escalado.

Funcionamiento interno. El operador escalado permite modificar la amplitud de la señal de entrada. Aunque esta operación puede verse como la multiplicación de la señal de entrada por una constante, en realidad se trata como la convolución de la señal de entrada por una delta de cuya amplitud sea el factor de escala. Por lo tanto, este operador es un caso particular del operador convolución.

Dadas las características del operador, ya que la respuesta al impulso del sistema siempre va a estar compuesta por una delta, la forma de realizar los cálculos se realiza siempre de la misma forma: de forma teórica. Sin embargo, para maximizar la versatilidad del software y considerando las características del operador, no se han almacenado todos los resultados en

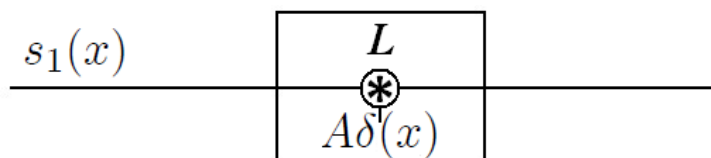


Figura 3.11: Esquema que representa al operador escalado. Como se observa la señal de entrada siempre es convolucionada con una delta centrada en el origen cuya amplitud representa el factor de escala que se quiera introducir. Este operador está contenido en la familia de operadores de la convolución y compone, a su vez, otra familia de operadores. Mientras el factor de escala esté sin definir, se hablará de la familia de operadores escalado. Una vez que la amplitud de la delta esté definida, se estará trabajando con un operador concreto. El operador identidad podría verse como un caso particular de esta familia, cuando el factor de escala es 1.

una base de datos sino que se ha implementado un método numérico que permite al software recalcular los resultados numéricos y modificar las representaciones necesarias.

Análisis del operador en el dominio real. Al igual que ocurre con todos los operadores del software, el usuario podrá aprender las características del operador escalado cuando trabaje con él. Ya que el operador estudiado es un caso particular del operador convolución sólo se mostrarán a continuación las características que no sean idénticas a las del operador mencionado:

- Este operador **convolucionada una delta, cuya amplitud es el factor de escala, con la señal de entrada**, véase la fig. 3.11. La señal de es definida por el usuario. La respuesta al impulso del sistema es de una forma fija (ya que siempre es una delta) pero el usuario puede modificar su amplitud de modo que pueda manipular el factor de escala a su albedrío.
- Aunque este operador no es una familia grande de operadores, como puede ser el operador convolución, es también una **familia de operadores** ya que el operador sólo estará completamente caracterizado cuando se fije la amplitud de la delta (que es el factor de escala del operador).

Además, algunas de las propiedades de este operador difieren con las citadas en la Secc.3.3.3 del operador convolución, ya que la respuesta al impulso del sistema es de una determinada forma:

- Al igual que la familia a la que pertenece, la convolución, el operador escalado **es lineal e invariante**, véase la fig.3.12 en la que se muestra su conjunto de respuestas al impulso.
- Como se observa en la fig. 3.12, el conjunto de respuestas al impulso está compuesto por un tren de deltas en la recta $\tau = \tau'$, esto significa que para cada instante la señal de salida sólo tiene en cuenta la señal de entrada en dicho instante. Esto quiere decir que el sistema **no tiene memoria**, ya que no tienen en cuenta instantes anteriores o posteriores al considerado para obtener la señal resultado.
- Por el mismo motivo que se demuestra que el operador no tiene memoria se sabe que, este caso concreto, **sí es causal**, ya que la salida del sistema no depende de los valores futuros que tome la entrada.
- Además, el sistema **es invertible**, ya que es posible definir un operador que recupere la señal original, no hay más que utilizar de nuevo un operador escalado cuyo factor de escala sea el inverso del factor de escala anterior. Esto es válido suponiendo que los factores de escala cero e infinito no sean admisibles.
- Finalmente, en el operador escalado la propiedad de **estabilidad está asegurada**, al igual que ocurría antes cuando el factor de escala sea distinto de cero o infinito.

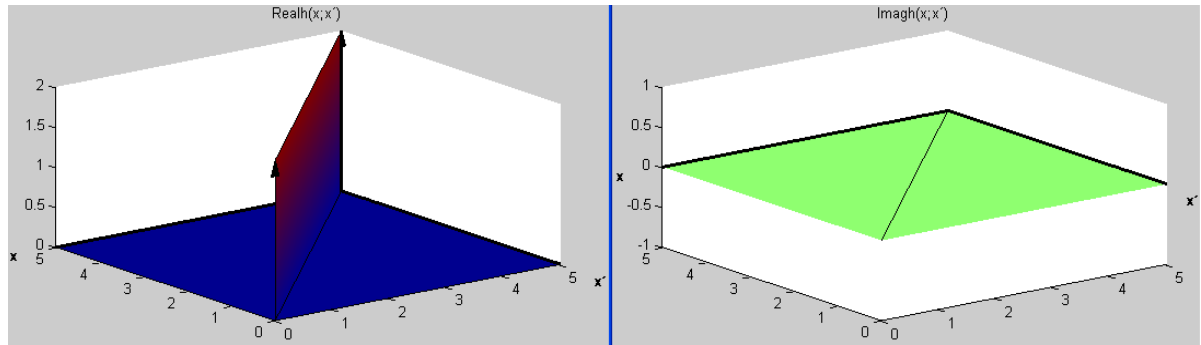


Figura 3.12: Conjunto de respuestas al impulso del operador escalado en el espacio de señales continuas aperiódicas cuando el factor de escala toma el valor 2. Visualizando el conjunto de respuestas al impulso se pueden apreciar las distintas propiedades del operador: su linealidad (en caso contrario no existiría el conjunto de respuestas al impulso representado), su invarianza (ya que el conjunto de respuestas al impulso está compuesto por la misma señal desplazada y centrada en la recta $\tau = \tau'$), su falta de memoria y su causalidad (como se ve se compone por un tren de deltas en el eje $\tau = \tau'$, es decir, para cada instante de la señal de salida sólo se tiene en cuenta la señal de entrada en dicho valor). Este operador forma parte de la familia de operadores convolución y compone, al mismo tiempo, una familia de operadores más pequeña denominada operador escalado. Una vez se haya definido el factor de escala se podrá hablar de un operador concreto.

En la siguiente tabla se muestra la expresión de la respuesta al impulso en los distintos espacios de partida:

Espacio de señales	Respuesta al impulso del operador escalado	
$S(-\infty, \infty)$ y $S(0, \infty)$	$h(x) = A\delta(x)$	
$P(X_0)$	$h(x) = A\delta(x)$	(3.63)
$D(-\infty, \infty)$	$h(n) = A\delta(n)$	
$D(N_0)$	$h(n) = A\delta(n)$	

Análisis espectral del operador. El operador escalado es un caso particular de la convolución, de modo que el conjunto de funciones núcleo que definen al operador está compuesto por un tren de deltas en $\mu = \mu'$ escalado por la transformada de la respuesta al impulso del sistema. Para este operador, la expresión de dicho conjunto de funciones es:

Espacio de partida	Conjunto de funciones de transferencia del operador escalado
$S(-\infty; \infty)$ bajo la TF	$H(\xi; \xi') = 2\pi A\delta(\xi + \xi')$
$P(X_0)$ bajo el DSF	$H(m; m') = \frac{A}{X_0}\delta(m + m')$
$D(-\infty; \infty)$ bajo la DTF	$H(\Omega; \Omega') = 2\pi A\delta(\Omega + \Omega')$
$D(N_0)$ bajo el DSF	$H(k; k') = \frac{A}{N_0}\delta(k + k')$

(3.64)

siendo A el factor de escala definido por el usuario.

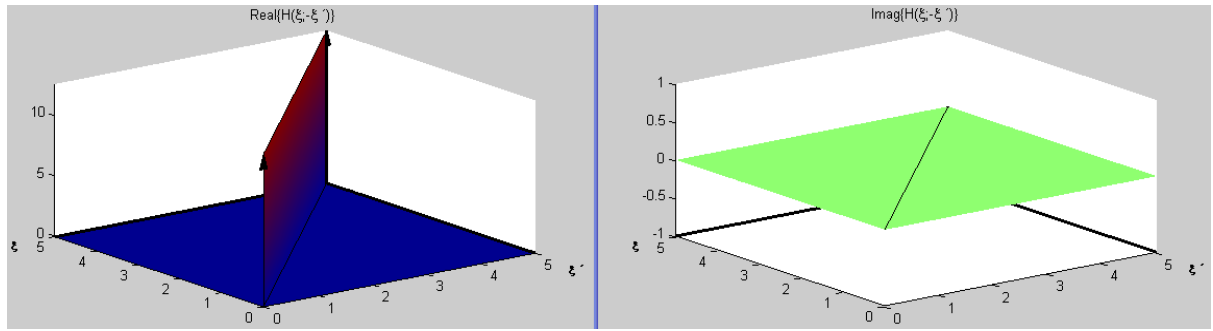


Figura 3.13: Conjunto de funciones núcleo que caracterizan al operador escalado de factor de escala 2 en el dominio transformado, en este caso bajo la transformada de Fourier.

En la fig. 3.13 se muestra un ejemplo del conjunto de funciones de transferencia que caracterizan al operador en el dominio espectral.

3.3.6. Operador identidad.

Funcionamiento interno. El operador identidad es un operador muy interesante ya que permite simular un sistema de comunicaciones que no afecte a la señal de entrada, de modo que la señal resultado es la misma que la señal de partida. Aunque esta operación no hace nada, no afecta a la señal, se modela como la convolución de la señal de entrada con una delta pura, es decir, una delta centrada en el origen y cuya amplitud es la unidad. Por lo tanto, este operador es un caso muy particular del operador convolución.

Debido al comportamiento particular del operador, el programa no realiza ningún cálculo con la señal de entrada y la trata del mismo modo independientemente de cual sea la naturaleza de las señales involucradas. El software simplemente ejecutará un método numérico que recalculé la señal en los límites deseados y copie los resultados obtenidos a las variables necesarias.

Análisis del operador en el dominio real. Del mismo modo que ocurre con el resto de operadores del software, el usuario podrá aprender las características del operador identidad cuando trabaje con él. Ya que el operador estudiado es un caso particular del operador convolución sólo se mostrarán a continuación las características que no sean idénticas a las del operador mencionado:

- Este operador **convoluciona una delta con la señal de entrada**, véase la fig. 3.14. La señal de entrada es definida por el usuario. La respuesta al impulso del sistema es fija por lo que el usuario no podrá modificar ninguno de sus parámetros, la delta siempre estará centrada en el origen y de amplitud unitaria.
- El operador identidad es un **operador concreto y totalmente definido** que pertenece a la familia de operadores convolución. Esto quiere decir que, una vez definida la señal de entrada, siempre se obtendrá la misma señal de salida, en este caso se devolverá la señal de entrada y, como es lógico, una vez definida ésta no se podrá modificar la salida del sistema.

Además, algunas de las propiedades de este operador difieren con las citadas en la Secc.3.3.3 del operador convolución, ya que la respuesta al impulso del sistema es de una determinada forma:

- Al igual que la familia a la que pertenece, la convolución, el operador identidad es **lineal e invariante**, véase la fig. 3.15 en la que se muestra su conjunto de respuestas al impulso.
- Como se observa en la fig. 3.15, el conjunto de respuestas al impulso está compuesto por un tren de deltas en la recta $\tau = \tau'$, esto quiere decir, que para cada instante la señal de

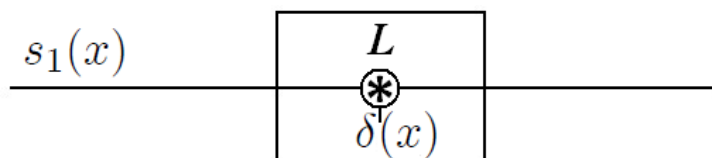


Figura 3.14: Esquema que representa al operador identidad. Como se observa la señal de entrada siempre es convolucionada con una delta centrada en el origen y de amplitud 1. Este operador está contenido en la familia de operadores de la convolución.

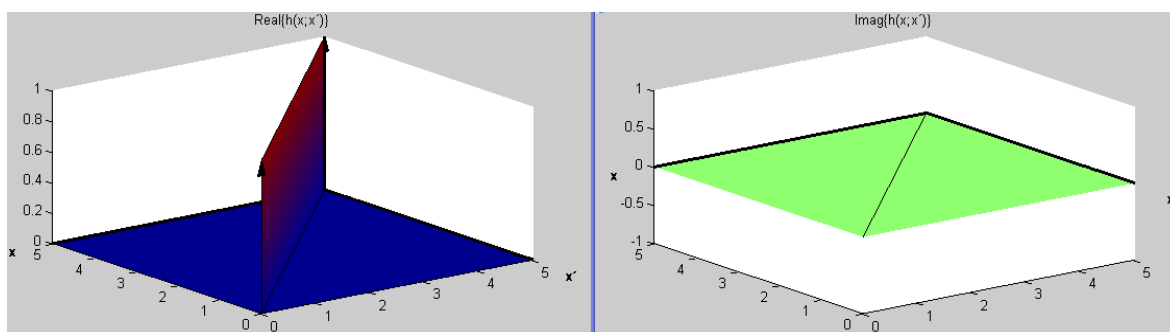


Figura 3.15: Conjunto de respuestas al impulso del operador identidad en el espacio de señales continuas periódicas. Visualizando el conjunto de respuestas al impulso se pueden apreciar las distintas propiedades del operador: su linealidad (en caso contrario no existiría el conjunto de respuestas al impulso representado), su invarianza (ya que el conjunto de respuestas al impulso está compuesto por la misma señal desplazada y centrada en la recta $\tau = \tau'$), su falta de memoria y su causalidad (como se ve se compone por un tren de deltas en el eje $\tau = \tau'$, es decir, para cada instante de la señal de salida sólo se tiene en cuenta la señal de entrada en dicho valor). Este operador forma parte de la familia de operadores convolución y es un operador concreto y definido.

salida sólo tiene en cuenta la señal de entrada en dicho instante. Esto quiere decir que el sistema **no tiene memoria**, ya que no tienen en cuenta instantes anteriores o posteriores al considerado para obtener la señal resultado.

- Por el mismo motivo que se demuestra que el operador no tiene memoria se sabe que, este caso concreto, **sí es causal**, ya que la salida del sistema no depende de los valores futuros que tome la entrada.
- Además, el sistema **es invertible**, ya que es posible definir un operador que recupere la señal original, de hecho en este caso ni siquiera sería necesario ya que la señal de salida es la misma que la señal de entrada. En cualquier caso, utilizando de nuevo el operador identidad se obtendría la señal de entrada.
- Finalmente, en el operador escalado la propiedad de **estabilidad está asegurada**, al obtener a la salida la misma señal que se tiene a la entrada, si la entrada está acotada la salida lo estará también.

En la siguiente tabla se muestra la expresión de la respuesta al impulso en los distintos espacios de partida:

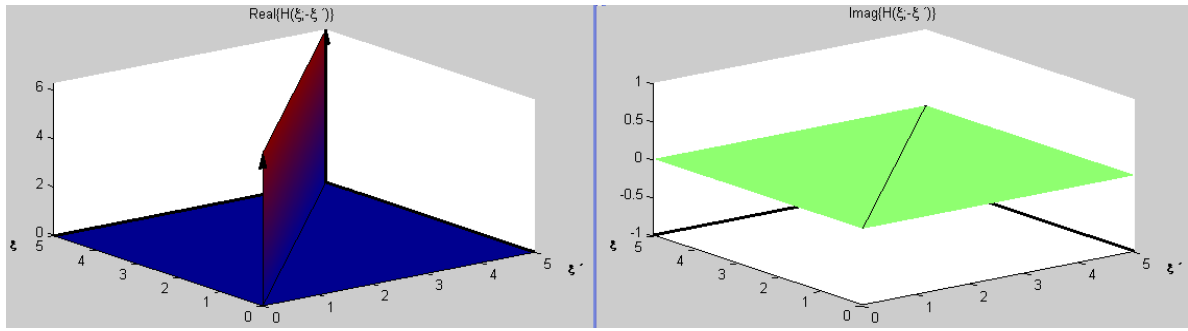


Figura 3.16: Conjunto de funciones núcleo que caracterizan al operador identidad en el dominio transformado, en este caso bajo la transformada de Fourier.

Espacio de señales	Respuesta al impulso del operador identidad	
$S(-\infty, \infty)$ y $S(0, \infty)$	$h(x) = \delta(x)$	
$P(X_0)$	$h(x) = \delta(x)$	(3.65)
$D(-\infty, \infty)$	$h(n) = \delta(n)$	
$D(N_0)$	$h(n) = \delta(n)$	

Análisis espectral del operador. El operador identidad es un caso particular de la convolución, de modo que el conjunto de funciones núcleo que definen al operador está compuesto por un tren de deltas en $\mu = \mu'$ escalado por la transformada de la respuesta al impulso del sistema. Para este operador, la expresión de dicho conjunto de funciones es:

Espacio de partida	Conjunto de funciones de transferencia del operador identidad
$S(-\infty; \infty)$ bajo la TF	$H(\xi; \xi') = 2\pi\delta(\xi + \xi')$
$P(X_0)$ bajo el DSF	$H(m; m') = \frac{1}{X_0}\delta(m + m')$
$D(-\infty; \infty)$ bajo la DTF	$H(\Omega; \Omega') = 2\pi\delta(\Omega + \Omega')$
$D(N_0)$ bajo el DSF	$H(k; k') = \frac{1}{N_0}\delta(k + k')$

(3.66)

En la fig. 3.16 se muestra un ejemplo del conjunto de funciones de transferencia que caracterizan al operador en el dominio espectral.

3.3.7. Operador desplazamiento.

Funcionamiento interno. El operador desplazamiento permite al usuario realizar un cambio en el eje de representación de modo que se desplaza la señal de la misma forma a otra zona del eje de abscisas. Este operador es muy interesante a la hora de simular retardos en una comunicación. La forma de modelar este operador matemáticamente es como una delta de Dirac desplazada de modo que este centrada en el punto donde se desee centrar la señal original. Es decir, una delta con el mismo desplazamiento que se desea aplicar a la señal de entrada. Por lo tanto, este operador es un caso particular del operador convolución.

Debido al comportamiento concreto de este operador, ya que la respuesta al impulso del sistema siempre va a estar compuesta por una delta, la forma de realizar los cálculos se realiza siempre

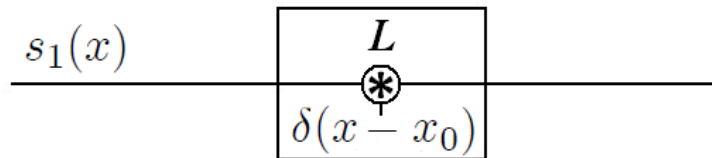


Figura 3.17: Esquema que representa al operador desplazamiento. Como se observa la señal de entrada siempre es convolucionada con una delta de amplitud 1 y con un desplazamiento fijado por el usuario. Este operador está contenido en la familia de operadores de la convolución y forma, a su vez, una familia pequeña de operadores, ya que mientras no se haya definido el desplazamiento el operador no estará caracterizado.

de la misma forma: de forma teórica. Sin embargo, para maximizar la versatilidad del software y considerando las características del operador, no se han almacenado todos los resultados en una base de datos sino que se ha implementado un método numérico que permite al software recalcular los resultados numéricos y modificar las representaciones necesarias.

Análisis del operador en el dominio real. Del mismo modo que ocurre con el resto de operadores del software, el usuario podrá aprender las características del operador desplazamiento cuando trabaje con él. Ya que el operador estudiado es un caso particular del operador convolución sólo se mostrarán a continuación las características que no sean idénticas a las del operador mencionado:

- Este operador **convoluciona una delta desplazada con la señal de entrada**, véase la fig.3.17. La señal de entrada es definida por el usuario. La respuesta al impulso del sistema es de una forma fija por lo que el usuario sólo podrá modificar uno de sus parámetros: su desplazamiento. De este modo, el usuario podrá desplazar la señal a donde quiera sin modificarla en otro aspecto, ya que la delta siempre tendrá amplitud unitaria.
- El operador desplazamiento compone una **familia pequeña de operadores** que, a su vez, pertenece a la familia de operadores convolución. Esto se debe a que, según el desplazamiento seleccionado por el usuario, para una misma señal de entrada, se obtendrán distintos resultados.

Este operador, comparte las mismas propiedades que la familia a la que pertenece, véase la Secc. 3.3.3 para conocer las propiedades del operador. Tan sólo hay que aclarar que, aunque el conjunto de respuestas al impulso esté compuesto por un tren de deltas, al igual que ocurre en los operadores de identidad y escalado, este operador **sí tiene memoria y no tiene porque ser causal**. Esto es debido a que lo que se introduce es un retardo o adelanto y, por lo tanto, se deben tener en cuenta valores de la señal de entrada en instantes distintos al considerado en el cálculo del resultado. Esta propiedad también puede observarse en la representación de su conjunto de respuestas al impulso, fig. 3.18, donde se observa que el tren de deltas no está en la recta $\tau = \tau'$.

En la siguiente tabla se muestra la expresión de la respuesta al impulso en los distintos espacios de partida:

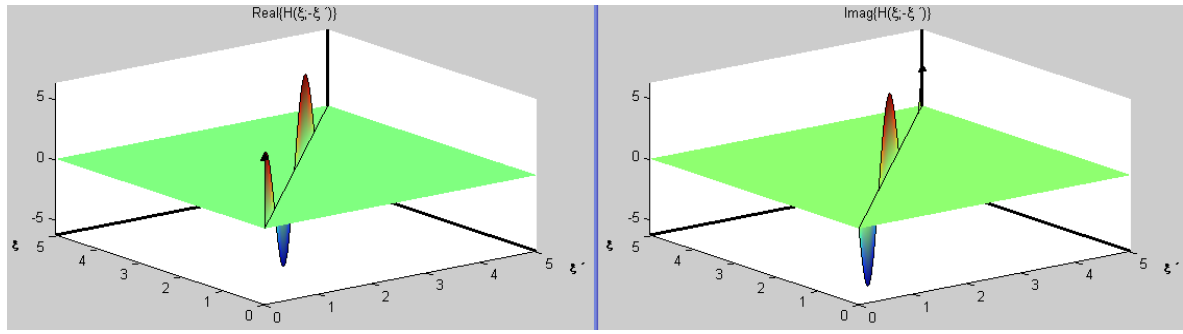


Figura 3.18: Conjunto de respuestas al impulso del operador desplazamiento en el espacio de señales continuas periódicas. Visualizando el conjunto de respuestas al impulso se pueden apreciar las distintas propiedades del operador: su linealidad (en caso contrario no existiría el conjunto de respuestas al impulso representado), su invarianza (ya que el conjunto de respuestas al impulso está compuesto por la misma señal desplazada y centrada en la recta $\tau = \tau'$), su falta de memoria y su causalidad (como se ve se compone por un tren de deltas en el eje $\tau = \tau'$, es decir, para cada instante de la señal de salida sólo se tiene en cuenta la señal de entrada en dicho valor). Este operador forma parte de la familia de operadores convolución y forma una familia de operadores más pequeña, ya que hasta que no se concrete el valor del desplazamiento el operador no estará definido.

Espacio de señales	Respuesta al impulso del operador desplazamiento
$S(-\infty, \infty)$ y $S(0, \infty)$	$h(x) = \delta(x - x_0)$
$P(X_0)$	$h(x) = \delta(x - x_0)$ (3.67)
$D(-\infty, \infty)$	$h(n) = \delta(n - n_0)$
$D(N_0)$	$h(n) = \delta(n - n_0)$

Análisis espectral del operador. El operador desplazamiento es un caso particular de la convolución, de modo que el conjunto de funciones núcleo que definen al operador está compuesto por un tren de deltas en $\mu = \mu'$ escalado por la transformada de la respuesta al impulso del sistema. Para este operador, la expresión de dicho conjunto de funciones es:

Espacio de partida	Conjunto de funciones de transferencia del op. desplazamiento
$S(-\infty; \infty)$ bajo la TF	$H(\xi; \xi') = 2\pi\delta(\xi + \xi')e^{-j\xi x_0}$
$P(X_0)$ bajo el DSF	$H(m; m') = \frac{1}{X_0}\delta(m + m')e^{-jm\xi_0 x_0}$
$D(-\infty; \infty)$ bajo la DTF	$H(\Omega; \Omega') = 2\pi\delta(\Omega + \Omega')e^{-j\Omega n_0}$
$D(N_0)$ bajo el DSF	$H(k; k') = \frac{1}{N_0}\delta(k + k')e^{-jk\Omega_0 n_0}$

(3.68)

En la fig. 3.19 se muestra un ejemplo del conjunto de funciones de transferencia que caracterizan al operador en el dominio espectral.

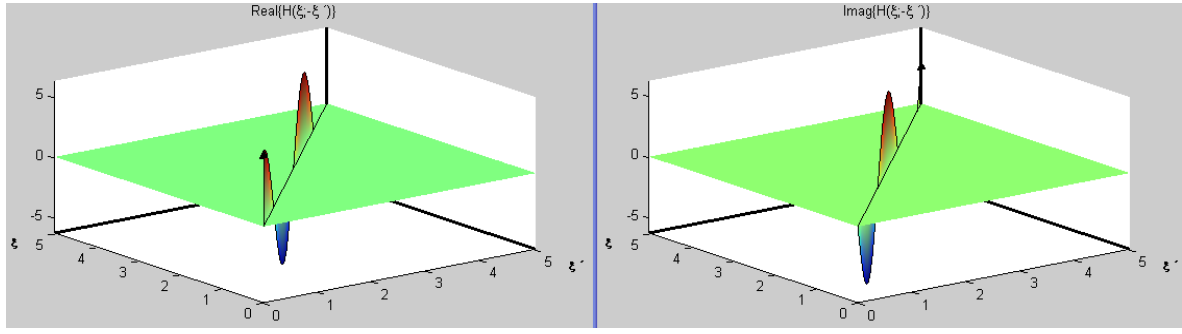


Figura 3.19: Conjunto de funciones núcleo que caracterizan al operador desplazamiento con factor de desplazamiento 2 en el dominio transformado, en este caso bajo la transformada de Fourier.

3.3.8. Operadores transformada de Fourier y su inversa.

Funcionamiento interno. Este operador se utiliza para calcular la transformada de Fourier o la transformada inversa de Fourier de la señal seleccionada como señal de entrada.

Como ocurre en otros operadores del software, existen dos modos para obtener una señal resultado: un modo teórico y un modo numérico. El modo escogido para resolver la operación dependerá exclusivamente de la naturaleza de la señal de entrada. De esta forma si la señal de entrada es de carácter analítico, toma valores en un rango infinito, la operación se resolverá de forma teórica. En el otro caso, señales numéricas definidas para un soporte finito, el sistema obtiene la señal de salida apoyándose en un método numérico, explicado en detalle en [4]. Esto se resume en el siguiente cuadro:

Señal de entrada	Modo de resolución
Analítica	Teórico
Numérica	Numérico

Cuando el modo de análisis es teórico, los resultados para cada señal de entrada son calculados previamente, siguiendo las expresiones descritas en las Tablas 3.69 y 3.70, y son almacenados en una base de datos. Como se puede observar en ambas tablas se puede obtener el resultado de la transformada inversa a partir de la directa y viceversa, por lo que sólo será necesario almacenar uno de los dos cálculos. Para conocer cuales son los resultados contenidos en las bases de datos del software puede consultar [4].

<p>Espacio de señales de partida</p> <p>$S(-\infty, \infty)$</p> <p>$P(X_0)$</p> <p>$D(-\infty, \infty)$</p> <p>$D(N_0)$</p>	<p>Cálculo de la transformada de Fourier</p> $F(\xi) = \mathbf{TF}[f(x)] = \int_{-\infty}^{\infty} f(x)e^{-j\xi x} dx$ $a(m) = \mathbf{DSF}[f_0(x)] = \frac{1}{X_0} \int_{\langle X_0 \rangle} f_0(x)e^{-jm\xi_0 x} dx$ $f_0(x) = \mathbf{DTF}[a(m)] = \sum_{m=-\infty}^{\infty} a(m)e^{jm\xi_0 x}, \xi_0 = \frac{2\pi}{X_0} = 1$ $a(m) = \mathbf{DSF}[x_0(n)] = \frac{1}{N_0} \sum_{n=\langle N_0 \rangle} x_0(n)e^{-jm\frac{2\pi}{N_0} n}$
--	--

(3.69)

Espacio de señales de partida	Cálculo de la transformada inversa de Fourier
$S(-\infty, \infty)$	$f(x) = \mathbf{TF}^{-1}[F(\xi)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\xi) e^{j\xi x} dx$
$P(X_0)$	$a(m) = \mathbf{TDF}^{-1}[f_0(n)] = \frac{1}{X_0} \int_{\langle X_0 \rangle}^{\infty} f_0(n) e^{jm\xi_0 n} dn$
$D(-\infty, \infty)$	$f_0(x) = \mathbf{DSF}^{-1}[a(m)] = \sum_{m=-\infty}^{\infty} a(m) e^{-jm\xi_0 x}, \xi_0 = \frac{2\pi}{X_0} = 1$
$D(N_0)$	$x_0(n) = \mathbf{DSF}^{-1}[x_0(n)] = \sum_{n=\langle N_0 \rangle}^{\infty} a(m) e^{jm\frac{2\pi}{N_0} n}$

(3.70)

Análisis del operador en el dominio real. Como la transformada de Fourier se puede analizar en el programa como un operador, el usuario analiza el comportamiento del sistema de comunicaciones definido por él y puede estudiar y aprender muchas de sus características gracias a controles gráficos, carteles informativos y visualizaciones. Las características principales del operador multiplicación son las siguientes:

- Este operador **calcula y representa la señal original en el dominio transformado cuando a ésta se le ha aplicado la Transformada de Fourier**. La señal de entrada será definida por el usuario.
- Este operador **es un operador concreto**, no constituye una familia de operadores ya que siempre devuelve, para la misma señal de entrada, la misma señal de salida. Esto quiere decir, a su vez, que su conjunto de respuestas al impulso esta totalmente definido y siempre es el mismo.

El usuario podrá estudiar, también, las propiedades del sistema analizado.

- Como se demuestra a continuación, este operador **es lineal**.

Dada una señal de entrada su paso por el sistema se obtiene:

$$f_1(x) \rightarrow g_1(x) = \int_{-\infty}^{\infty} f_1(x) e^{-j\xi x} dx. \quad (3.71)$$

Lo mismo ocurre si se introduce otra señal de entrada diferente:

$$f_2(x) \rightarrow g_2(x) = \int_{-\infty}^{\infty} f_2(x) e^{-j\xi x} dx. \quad (3.72)$$

Si como entrada al sistema se introduce una combinación lineal de las dos señales de entrada mencionadas previamente, se obtiene:

$$\begin{aligned} \alpha f_1(x) + \beta f_2(x) \rightarrow g_3(x) &= \int_{-\infty}^{\infty} [\alpha f_1(x) + \beta f_2(x)] e^{-j\xi x} dx = \\ &= \alpha \int_{-\infty}^{\infty} f_1(x) e^{-j\xi x} dx + \beta \int_{-\infty}^{\infty} f_2(x) e^{-j\xi x} dx = \alpha g_1(x) + \beta g_2(x). \end{aligned} \quad (3.73)$$

Como el operador es lineal, se puede representar su conjunto de respuestas al impulso, véase la fig. 3.20.

- Este operador **no es invariante**. Esto se demuestra de la siguiente forma:

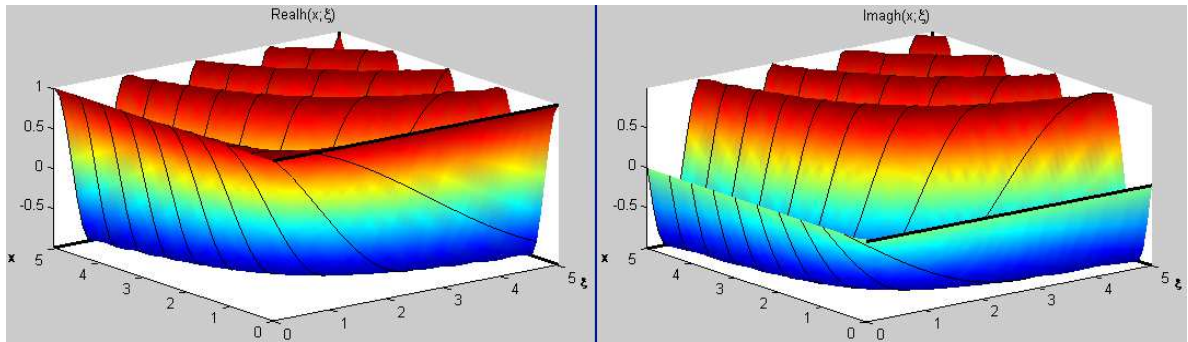


Figura 3.20: Conjunto de respuestas al impulso del operador transformada de Fourier en el espacio de señales continuas aperiódicas. Visualizando el conjunto de respuestas al impulso se pueden apreciar las distintas propiedades del operador: su linealidad (en caso contrario no existiría el conjunto de respuestas al impulso representado), su no invarianza (ya que el conjunto de respuestas al impulso no está compuesto por la misma señal desplazada y centrada en la recta $\tau = \tau'$ si no que se necesita la representación de toda la curva para conocer su caracterización), su memoria y su no causalidad (ya que el operador tiene en cuenta la señal de entrada en instantes anteriores y posteriores al considerado para el cálculo de la señal resultado).

Dada una señal de entrada, definida como se muestra en (3.71). Si se introduce en el sistema esta señal desplazada o retardada se obtiene que:

$$f_1(x - x_0) \rightarrow g_4(x) = \int_{-\infty}^{\infty} f_1(x - x_0) e^{-j\xi x} dx \neq \int_{-\infty}^{\infty} f_1(x - x_0) e^{-j\xi(x-x_0)} dx = g_1(x - x_0). \quad (3.74)$$

Como el operador no es invariante el conjunto de respuestas al impulso no se puede representar como una única función sino que es necesario representar dicho conjunto de señales para caracterizar adecuadamente al operador. Esta propiedad se puede observar en la representación del conjunto de respuestas al impulso, fig. 3.20.

- Como se observa en las fórmulas de la tabla 3.69, para obtener el resultado de la transformada de Fourier de una señal, para cada instante se tiene en cuenta toda la señal de entrada, incluyendo los instantes anteriores y posteriores al considerado. Por lo tanto, el sistema **tiene memoria**.
- Por el mismo motivo que se demuestra que el operador tiene memoria se sabe que **no es causal**, ya que la salida del sistema depende de los valores futuros que tome la entrada.
- Además, el sistema **es invertible**, ya que es posible definir un operador que recupere la señal original. El operador Transformada Inversa de Fourier consigue recuperar la señal original después de que ésta se haya transformado mediante el operador Transformada de Fourier.
- Finalmente, el operador **no es estable**, ya que aunque se introduzca una señal acotada la señal de salida no lo es. De hecho, cuanto más estrechas sean las señales en el dominio real, más planas serán en el dominio transformado. Un ejemplo clásico de este concepto es la transformada de Fourier de un pulso rectangular, que es una sinc.

En la siguiente tabla se muestra la expresión de la respuesta al impulso en los distintos espacios de partida:

Espacio de señales	Expresión de la respuesta al impulso del operador	
	Transformada de Fourier	Transf. Inversa de Fourier
$S(-\infty, \infty)$	$h(x; x') = e^{-jxx'}$	$h(x; x') = e^{jxx'}$
$P(X_0)$	$h(m, x') = \frac{1}{X_0} e^{-jm\xi_0 x'}$	$h(m, x') = \frac{1}{X_0} e^{jm\xi_0 x'}$
$D(-\infty, \infty)$	$h(\Omega; n') = e^{-j\Omega n'}$	$h(\Omega; n') = e^{j\Omega n'}$
$D(N_0)$	$h(m; n') = \frac{1}{N_0} e^{-jm\Omega_0 n'}$	$h(m; n') = \frac{1}{N_0} e^{jm\Omega_0 n'}$

(3.75)

Análisis espectral del operador. El operador escalado es un caso particular de la convolución, de modo que el conjunto de funciones núcleo que definen al operador está compuesto por un tren de deltas en $\mu = \mu'$ escalado por la transformada de la respuesta al impulso del sistema. Para este operador, la expresión de dicho conjunto de funciones es:

Espacio de partida	Conjunto de funciones de transferencia del operador	
	Transformada de Fourier	Transf. Inversa de Fourier
$S(-\infty; \infty)$ bajo la TF	$H(\xi; \xi') = e^{j\xi\xi'}$	$H(\xi; \xi') = e^{-j\xi\xi'}$
$P(X_0)$ bajo el DSF	$H(\Omega, m') = \frac{1}{X_0} e^{jm'\Omega}$	$H(\Omega, m') = \frac{1}{X_0} e^{-jm'\Omega}$
$D(-\infty; \infty)$ bajo la DTF	$H(m; \Omega') = e^{jm\Omega'}$	$H(m; \Omega') = e^{-jm\Omega'}$
$D(N_0)$ bajo el DSF	$H(k; k') = \frac{1}{N_0^2} e^{jkk'\Omega_0}$	$H(k; k') = \frac{1}{N_0^2} e^{-jkk'\Omega_0}$

(3.76)

A continuación se expone, para cada uno de los casos, el método de cálculo empleado para obtener la expresión final del conjunto de funciones de transferencia. Dada la similitud de ambos operadores, transformada y transformada inversa de Fourier, sólo se expondrá el caso de la transformada directa:

- **Espacio de señales de variable continua definidas para todo valor** $S(-\infty; \infty)$.

$$H(\xi; \xi') = \mathbf{TF}_{x'}[\mathbf{TF}_x[e^{-jxx'}]] = \int_{x'} \int_x e^{-jxx'} e^{-j\xi x} e^{-j\xi' x'} dx dx'. \quad (3.77)$$

Se calcula primero la transformada respecto a x :

$$\mathbf{TF}_x[e^{-jxx'}] = 2\pi\delta(-\xi - x'). \quad (3.78)$$

Se calcula ahora la transformada respecto a x' :

$$\mathbf{TF}_{x'}[2\pi\delta(-\xi - x')] = 2\pi\mathbf{TF}_{x'}[\delta(-\xi - x')] = 2\pi e^{j\xi\xi'}. \quad (3.79)$$

De modo que se obtiene el conjunto de funciones de transferencia como:

$$H(\xi; \xi') = 2\pi e^{j\xi\xi'}. \quad (3.80)$$

- **Espacio de señales de variable continua periódicas** $P(X_0)$.

$$H(\Omega, m') = \mathbf{DSF}_{x'}[\mathbf{TF}_m[\frac{1}{X_0}e^{-jm\xi_0x'}]]. \quad (3.81)$$

Se calcula primero la transformada respecto a m :

$$\mathbf{TF}_m[\frac{1}{X_0}e^{-jm\xi_0x'}] = \frac{2\pi}{X_0} \sum_n \delta(\Omega + \xi_0x' - 2\pi n). \quad (3.82)$$

Se calcula ahora el desarrollo en serie respecto a x' :

$$\begin{aligned} \mathbf{DSF}_{x'}[\frac{2\pi}{X_0} \sum_n \delta(\Omega + \xi_0x' - 2\pi n)] &= \frac{1}{X_0} \int_{\langle X_0 \rangle} \frac{2\pi}{X_0} \sum_n \delta(\Omega + \xi_0x' - 2\pi n) e^{-jm'\xi_0x'} dx' \\ &= \frac{1}{X_0} \frac{2\pi}{X_0} \frac{1}{\xi_0} \int_{\langle X_0 \rangle} \delta(x' + \frac{\Omega}{\xi_0}) e^{-jm'\xi_0x'} dx' = \frac{1}{X_0} e^{jm'\Omega}. \end{aligned} \quad (3.83)$$

De modo que se obtiene el conjunto de funciones de transferencia como:

$$H(\Omega, m') = \frac{1}{X_0} S_2(m + m'). \quad (3.84)$$

- **Espacio de señales de variable entera definidas para todo valor** $D(-\infty; \infty)$.

$$H(m; \Omega') = \mathbf{TF}_{n'}[\mathbf{DSF}_\Omega[e^{-j\Omega n'}]]. \quad (3.85)$$

Se calcula primero el desarrollo en serie respecto a Ω , tomando como periodo $X_0 = 2\pi$:

$$\begin{aligned} \mathbf{DSF}_\Omega[e^{-j\Omega n'}] &= \frac{1}{X_0} \int_{\langle X_0 \rangle} e^{-j\Omega n'} e^{-j\Omega m \xi_0} d\Omega = \frac{1}{2\pi} \int_{\langle X_0 \rangle} e^{-j\Omega(n'+m)} d\Omega \\ \mathbf{DSF}_\Omega[e^{-j\Omega n'}] &= \begin{cases} 1, & \text{si } n' = -m \\ 0, & \text{si } n' \neq -m \end{cases}. \end{aligned} \quad (3.86)$$

Se calcula ahora la transformada respecto a n' :

$$\mathbf{DTF}_{n'}[\delta(m + n')] = e^{jm\Omega'}. \quad (3.87)$$

De modo que se obtiene el conjunto de funciones de transferencia como:

$$H(m; \Omega') = e^{jm\Omega'}. \quad (3.88)$$

- **Espacio de señales de variable entera periódicas** $D(N_0)$.

$$H(k; k') = \mathbf{DSF}_{n'}[\mathbf{DSF}_m[\frac{1}{N_0}e^{-jm\Omega_0 n'}]] = \frac{1}{N_0^2} \sum_{n'} \sum_m \frac{1}{N_0} e^{-jm\Omega_0 n'} e^{-jk'\Omega_0 n'} e^{-jk\Omega_0 m}. \quad (3.89)$$

Se calcula primero la transformada respecto a n :

$$\mathbf{DSF}_m\left[\frac{1}{N_0}e^{-jm\Omega_0n'}\right] = \frac{1}{N_0} \sum_p \delta(k + n' - pN_0). \quad (3.90)$$

Se calcula ahora la transformada respecto a n' , haciendo uso de la propiedad que cumple que un producto de señales en el dominio real equivale a una convolución escalada de las señales en el dominio espectral:

$$\mathbf{DSF}_{n'}\left[\frac{1}{N_0} \sum_p \delta(k + n' - pN_0)\right] = \frac{1}{N_0^2} e^{jkk'\Omega_0}. \quad (3.91)$$

De modo que se obtiene el conjunto de funciones de transferencia como:

$$H(k; k') = \frac{1}{N_0^2} e^{jkk'\Omega_0}. \quad (3.92)$$

3.3.9. Operador transformada de Hilbert y su inversa.

Funcionamiento interno. Este operador se utiliza para calcular la transformada de Hilbert o la transformada inversa de Hilbert de la señal seleccionada como señal de entrada. Este operador sólo está disponible en el espacio de señales continuas aperiódicas.

Como ocurre en otros operadores del software, existen dos modos para obtener una señal resultado: un modo teórico y un modo numérico. El modo escogido para resolver la operación dependerá exclusivamente de la naturaleza de la señal de entrada. De esta forma si la señal de entrada es de carácter analítico, toma valores en un rango infinito, la operación se resolverá de forma teórica. En el otro caso, señales numéricas definidas para un soporte finito, el sistema obtiene la señal de salida apoyándose en un método numérico, explicado en detalle en [5]. Esto se resume en el siguiente cuadro:

Señal de entrada	Modo de resolución
Analítica	Teórico
Numérica	Numérico

Cuando el modo de análisis es teórico, los resultados para cada señal de entrada son calculados previamente, siguiendo la expresiones descrita en (3.93), y son almacenados en una base de datos. Como se puede observar en (tf y ift) se puede obtener el resultado de la transformada inversa a partir de la directa y viceversa, por lo que sólo será necesario almacenar uno de los dos cálculos. Para conocer cuales son los resultados contenidos en las bases de datos del software puede consultar [5].

$$g(x) = TH[f(x)] = \int_{-\infty}^{\infty} -\frac{f(x')}{\pi(x-x')} dx' \quad (3.93)$$

Análisis del operador en el dominio real. Como la transformada de Hilbert se puede analizar en el programa como un operador, el usuario analiza el comportamiento del sistema de comunicaciones definido por él y puede estudiar y aprender muchas de sus características gracias a controles gráficos, carteles informativos y visualizaciones. Las características principales del operador multiplicación son las siguientes:

- Este operador **calcula y representa la señal original en el dominio transformado cuando a ésta se le ha aplicado la Transformada de Hilbert**. La señal de entrada será definida por el usuario.
- Este operador **es un operador concreto**, no constituye una familia de operadores ya que siempre devuelve, para la misma señal de entrada, la misma señal de salida. Esto quiere decir, a su vez, que su conjunto de respuestas al impulso esta totalmente definido y siempre es el mismo.

El usuario podrá estudiar, también, las propiedades del sistema analizado.

- Como se demuestra a continuación, este operador **es lineal**.

Dada una señal de entrada su paso por el sistema se obtiene:

$$f_1(x) \rightarrow g_1(x) = \int_{-\infty}^{\infty} -\frac{f_1(x')}{\pi(x-x')} dx' \quad (3.94)$$

Lo mismo ocurre si se introduce otra señal de entrada diferente:

$$f_2(x) \rightarrow g_2(x) = \int_{-\infty}^{\infty} -\frac{f_2(x')}{\pi(x-x')} dx' \quad (3.95)$$

Si como entrada al sistema se introduce una combinación lineal de las dos señales de entrada mencionadas previamente, se obtiene:

$$\begin{aligned} \alpha f_1(x) + \beta f_2(x) &\rightarrow g_3(x) = \int_{-\infty}^{\infty} -\frac{[\alpha f_1(x') + \beta f_2(x')]}{\pi(x-x')} dx' = \\ &= \alpha \int_{-\infty}^{\infty} -\frac{f_1(x')}{\pi(x-x')} dx' + \beta \int_{-\infty}^{\infty} -\frac{f_2(x')}{\pi(x-x')} dx' = \alpha g_1(x) + \beta g_2(x). \end{aligned} \quad (3.96)$$

Como el operador es lineal, se puede representar su conjunto de respuestas al impulso, véase la fig. 3.21.

- Este operador **es invariante**. Esto se demuestra de la siguiente forma:

Dada una señal de entrada, definida como se muestra en (3.71). Si se introduce en el sistema esta señal desplazada o retardada se obtiene que:

$$f_1(x-x_0) \rightarrow g_4(x) = \int_{-\infty}^{\infty} -\frac{f_1(x'-x_0)}{\pi(x-(x'-x_0))} dx' = g_1(x-x_0). \quad (3.97)$$

Como el operador es invariante el conjunto de respuestas al impulso se puede representar como una única función que se va desplazando y siempre está centrada en la recta $\tau = \tau'$. Esta propiedad se puede observar en la representación del conjunto de respuestas al impulso, fig. 3.21.

- Como se observa en las fórmulas de la tabla 3.93, para obtener el resultado de la transformada de Hilbert de una señal, para cada instante se tiene en cuenta toda la señal de entrada, incluyendo los instantes anteriores y posteriores al considerado. Por lo tanto, el sistema **tiene memoria**.
- Por el mismo motivo que se demuestra que el operador tiene memoria se sabe que **no es causal**, ya que la salida del sistema depende de los valores futuros que tome la entrada.
- Además, el sistema **es invertible**, ya que es posible definir un operador que recupere la señal original. El operador Transformada Inversa de Hilbert consigue recuperar la señal original después de que ésta se haya transformado mediante el operador Transformada de Hilbert.

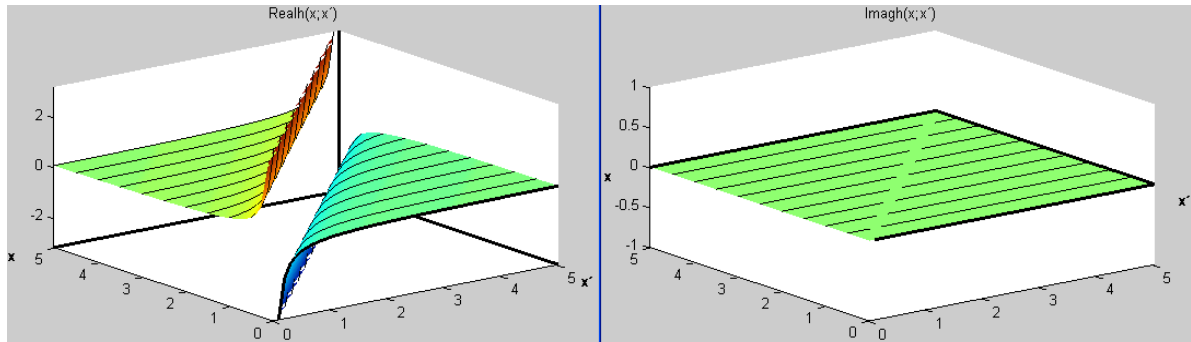


Figura 3.21: Conjunto de respuestas al impulso del operador transformada de Hilbert en el espacio de señales continuas aperiódicas. Visualizando el conjunto de respuestas al impulso se pueden apreciar las distintas propiedades del operador: su linealidad (en caso contrario no existiría el conjunto de respuestas al impulso representado), su invarianza (ya que el conjunto de respuestas al impulso está compuesto por la misma señal desplazada y centrada en la recta $\tau = \tau'$), su memoria y su nocausalidad (ya que el operador tiene en cuenta la señal de entrada en instantes anteriores y posteriores al considerado para el cálculo de la señal resultado).

- Finalmente, el operador **no es estable**, ya que aunque se introduzca una señal acotada la señal de salida no lo es.

La respuesta al impulso de este operador, para el espacio $S(-\infty, \infty)$, único espacio en el que está definido el operador, tiene la siguiente expresión:

Espacio de señales	Expresión de la respuesta al impulso del operador		
	Transformada de Hilbert	Transf. Inversa de Hilbert	
$S(-\infty, \infty)$	$h(x; x') = -\frac{1}{\pi(x-x')}$	$h(x; x') = \frac{1}{\pi(x-x')}$	(3.98)

Análisis espectral del operador. En la siguiente tabla se muestra la expresión del conjunto de funciones núcleo que caracterizan a los operadores transformada de Hilbert y su inversa en el dominio espectral:

Espacio de partida	Conjunto de funciones de transferencia del operador		
	Transformada de Hilbert	Transf. Inversa de Hilbert	
$S(-\infty; \infty)$ bajo la TF	$H(\xi; \xi') = 2\pi j \operatorname{sgn}(\xi) \delta(\xi - \xi')$	$H(\xi; \xi') = -2\pi j \operatorname{sgn}(\xi) \delta(\xi - \xi')$	(3.99)

En la fig. 3.22 se muestra el conjunto de funciones de transferencia que caracterizan al operador en el dominio espectral.

3.3.10. Operador modulador en amplitud.

Funcionamiento interno. El operador modulador en amplitud modula la señal de entrada con un coseno. Esto se traduce en la multiplicación de la señal de entrada por un coseno cuyos parámetros define el usuario. Al tratarse de un caso particular de una multiplicación su modo de tratamiento es el mismo que el de dicho operador.

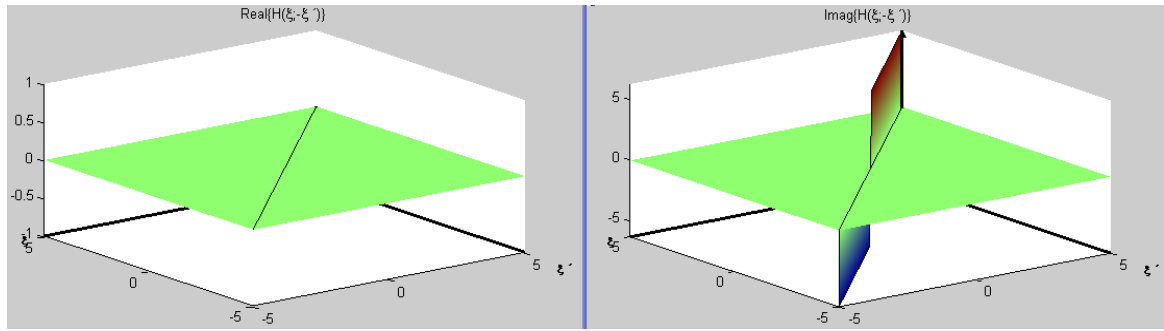


Figura 3.22: Conjunto de funciones núcleo que caracterizan a uno del operador transformada de Hilbert en el dominio transformado, en este caso bajo la transformada de Fourier.

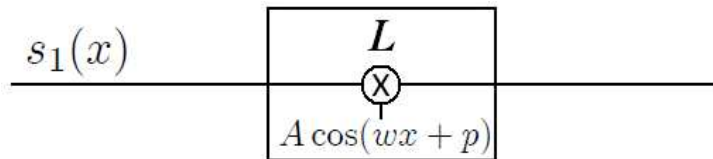


Figura 3.23: Esquema que representa al operador modulador en amplitud. Como se observa la señal de entrada siempre es multiplicada con un coseno. Según se modifiquen los parámetros de dicho coseno (amplitud, pulsación y fase) el coseno variará y por tanto el resultado de la operación. Esto quiere decir que el operador constituye una familia de operadores. Al mismo tiempo, este operador está contenido en la familia de operadores de la multiplicación.

El modulador en amplitud se ha implementado en el software como un operador y, por lo tanto, cuando se seleccione, el usuario podrá analizar el comportamiento del sistema de comunicaciones y podrá estudiar y aprender muchas de sus características gracias a controles gráficos, carteles informativos y visualizaciones. Las características principales del operador modulación en amplitud son las siguientes:

Este operador **multiplica un coseno por la señal de entrada**, véase la fig. 3.23. La señal de entrada está definida por el usuario. La señal que caracteriza al sistema, el coseno, es siempre de una forma fija, sin embargo, sus parámetros (amplitud, pulsación y desfase) pueden ser modificados por el usuario.

Este operador es, en realidad, una **familia de operadores**. Aunque no constituya una familia grande de operadores, como es el caso de la multiplicación, al poder modificar los parámetros del coseno, este operador compone una familia pequeña de operadores. Se debe recordar que un operador devuelve, para la misma señal de entrada, siempre la misma salida. Sin embargo, el operador considerado puede devolver una salida distinta para la misma señal de entrada cuando el coseno tenga parámetros distintos. Por tanto, el operador sólo estará completamente caracterizado cuando el coseno esté totalmente definido.

Análisis del operador en el dominio real. El usuario podrá estudiar, también, las propiedades del sistema analizado. Como el operador modulador en amplitud es un caso concreto del operador multiplicación sólo se expondrán aquí las diferencias que tenga con la familia de operadores a la que pertenece:

- Como ocurre con la multiplicación, este operador es lineal, no invariante, sin memoria y causal, véase la fig.3.24 para observar dichas propiedades.
- El modulador en amplitud es invertible, ya que existe la posibilidad de utilizar un demo-

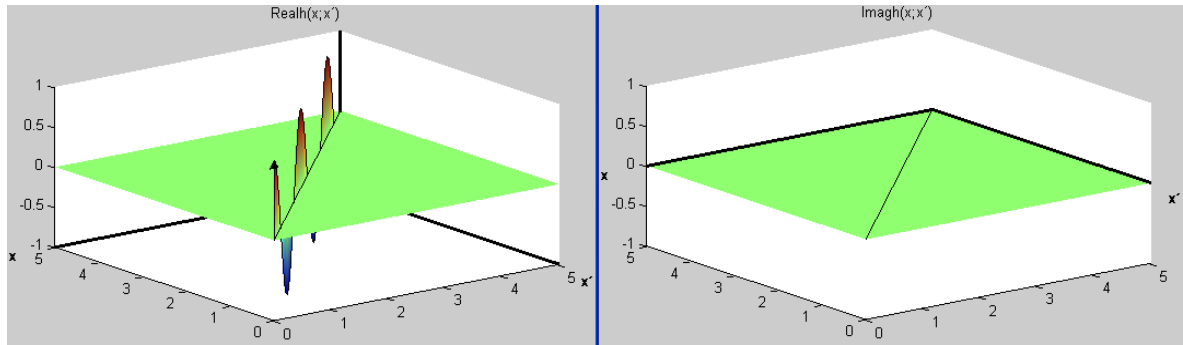


Figura 3.24: Conjunto de respuestas al impulso del operador modulador en amplitud en el espacio de señales continuas aperiódicas. Visualizando el conjunto de respuestas al impulso se pueden apreciar las distintas propiedades del operador: su linealidad (en caso contrario no existiría el conjunto de respuestas al impulso representado), su no invarianza (ya que el conjunto de respuestas al impulso no está compuesto por la misma señal desplazada y centrada en la recta $\tau = \tau'$, se observa que a lo largo de dicha recta hay un tren de deltas modulado por el coseno), su falta de memoria y su causalidad (ya que el operador sólo tiene en cuenta la señal de entrada en el instante considerado para el cálculo de la señal resultado).

dulador para recuperar la señal inicial.

- Finalmente, la propiedad de estabilidad está asegurada para este operador.

El operador modulador en amplitud es un caso particular del operador multiplicación donde la señal moduladora está compuesta por un coseno en la parte real. Por tanto, las expresiones del conjunto de respuestas al impulso de este operador serán iguales a las del operador multiplicación para una señal específica, $s_2(\tau) = A \cos(B\tau + C)$. En la siguiente tabla se muestran las expresiones para los distintos espacios de partida a modo de resumen: e

Espacio de partida	Conjunto de respuestas al impulso del op. modulador en amplitud
$S(-\infty; \infty)$	$h(x; x') = A \cos(Bx + C)\delta(x - x') = A \cos(Bx' + C)\delta(x - x')$
$P(X_0)$	$h(x; x') = A \cos(Bx + C)\delta_0(x - x') = A \cos(Bx' + C)\delta_0(x - x')$
$D(-\infty; \infty)$	$h(n; n') = A \cos(Bn + C)\delta(n - n') = A \cos(Bn' + C)\delta(n - n')$
$D(N_0)$	$h(n; n') = A \cos(B\tau n + C)\delta_0(n - n') = A \cos(Bn' + C)\delta_0(n - n')$

(3.100)

Análisis espectral del operador. De nuevo, al igual que ocurre en el dominio real, el análisis espectral de este operador equivale al análisis espectral del operador multiplicación teniendo en cuenta el hecho de que la señal moduladora es de la forma $A \cos(B\tau + C)$.

En la siguiente tabla se muestra la transformada de la señal moduladora para los distintos espacios de partida, de modo que sólo hay que sustituir esta expresión en los conjuntos de funciones de transferencia del operador multiplicación.

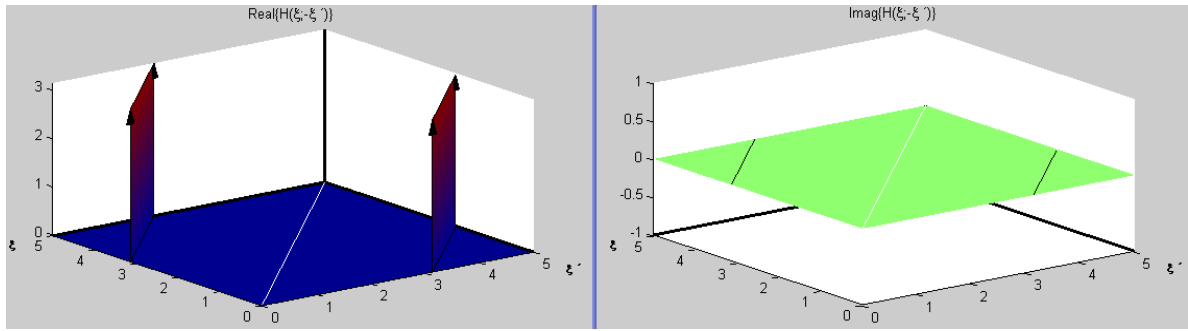


Figura 3.25: Conjunto de funciones núcleo que caracterizan al operador modulador en amplitud en el dominio transformado, en este caso bajo la transformada de Fourier.

Espacio de partida	Transformada de la señal moduladora: $A \cos(B\tau + C)$	(3.101)
$S(-\infty; \infty)$ bajo la TF	$\pi\delta(\xi - \xi_0) + \pi\delta(\xi + \xi_0)$	
$P(X_0)$ bajo el DSF	$a(m = \pm 1) = \frac{1}{2}, a(m \neq \pm 1) = 0$	
$D(-\infty; \infty)$ bajo la DTF	$\pi \sum_k \delta(\Omega - k\Omega_0 - 2\pi k) + \pi \sum_k \delta(\Omega + k\Omega_0 - 2\pi k)$	
$D(N_0)$ bajo el DSF	$a(k) = \frac{1}{2}, k = \pm p \pm mN_0$	

En la fig. 3.25 se muestra el conjunto de funciones de transferencia que caracterizan al operador en el dominio espectral.

3.4. Funcionalidades disponibles para la visualización de señales.

A lo largo del presente capítulo se ha ido presentando la estructura disponible en el software y que se corresponde con el esquema general de la teoría de señales y sistemas: los espacios de señales y los operadores disponibles. En esta sección se presentan las funcionalidades de visualización disponibles en el software que, sin formar parte del esquema general presentado en 2, son de gran importancia para que el usuario pueda analizar el comportamiento y funcionamiento de los sistemas de comunicaciones. A continuación se describen cada una de las 4 funcionalidades de visualización principales.

3.4.1. Visualización principal

Esta funcionalidad representa las señales de entrada y salida del sistema, de modo que el usuario pueda observar las transformaciones que sufre la señal cuando está influida por un operador. Puede visualizar un ejemplo de esta ventana en la fig. 3.26. Como se observa, la ventana se divide en seis zonas claramente diferenciadas. Las tres zonas superiores corresponden al análisis del sistema de comunicaciones en el dominio real mientras que en las tres inferiores el usuario podrá analizar el sistema en el dominio transformado.

Por otra parte, si se analiza la ventana de izquierda a derecha, la interfaz está dividida en tres partes. En la primera de ellas se representa la señal de entrada al sistema en el modo seleccionado (real / imaginario o módulo / fase). En la segunda zona el usuario dispone de dos botones para visualizar el conjunto de respuestas al impulso que caracteriza al operador, que se visualizará en una ventana diferente explicada a continuación. Finalmente, la tercera parte, situada a la

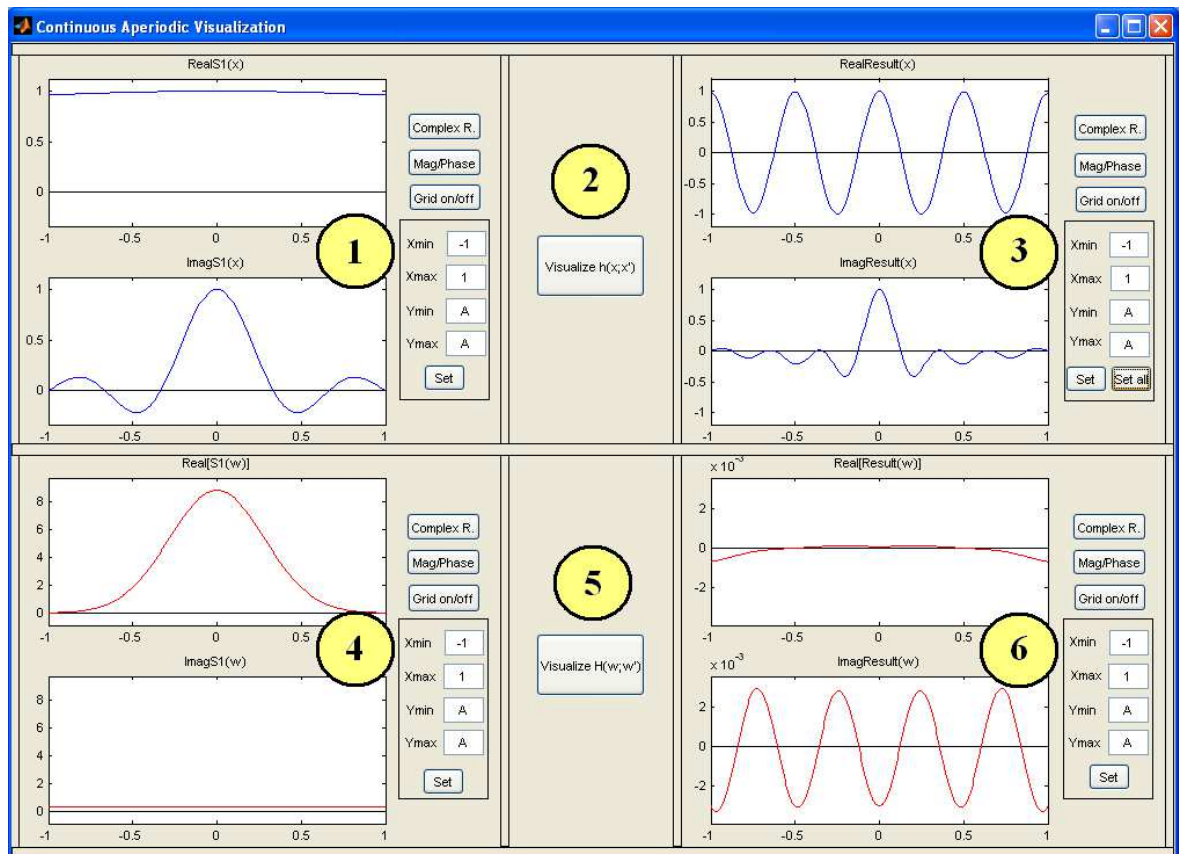


Figura 3.26: Ventana de visualización principal. En la ventana se diferencian seis zonas, las tres superiores corresponden al análisis en el dominio real y las tres inferiores al análisis en el dominio transformado. Al mismo tiempo, las partes 1 y 4 (a la izquierda de la ventana) representan la entrada al sistema, las 2 y 5 (centrales) permiten acceder al análisis gráfico del operador y las 3 y 6 (a la derecha) representan la señal de salida.

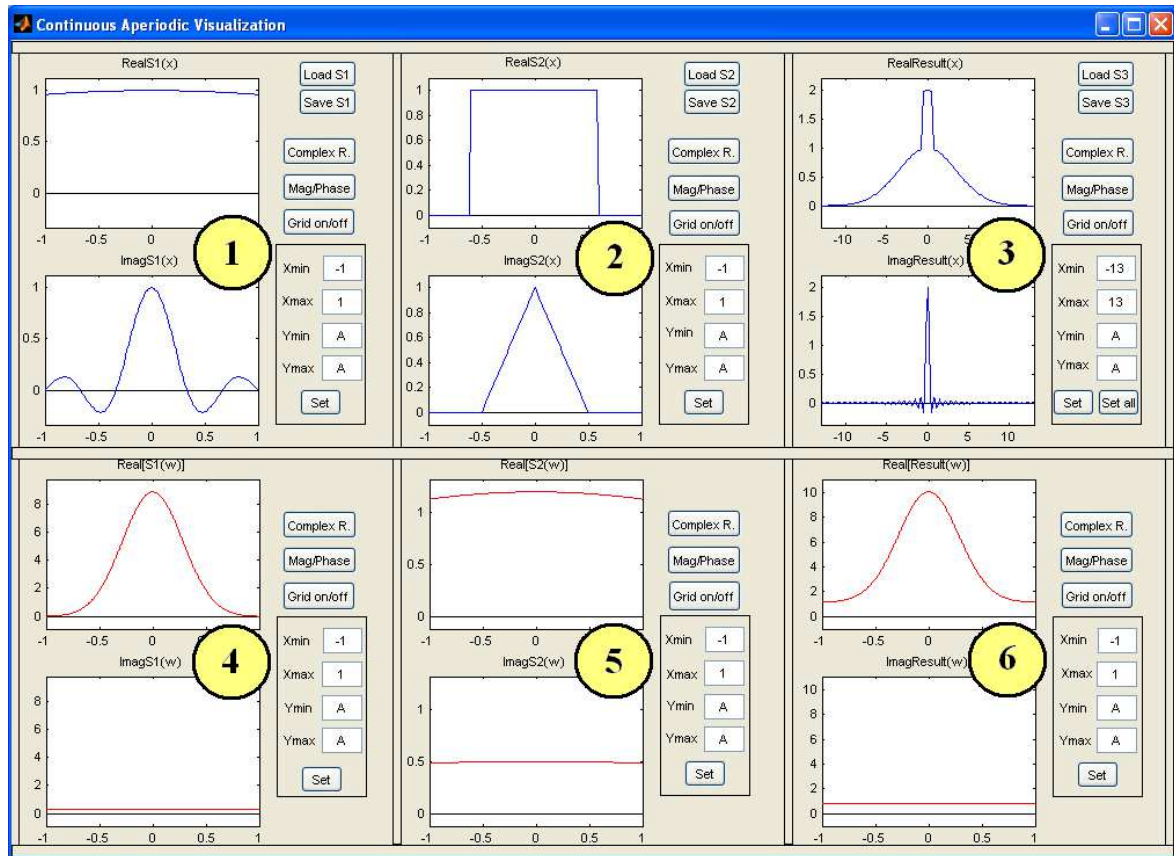


Figura 3.27: Ventana de visualización principal cuando se trabaja con operaciones básicas. Como se observa, la ventana se modifica para representar en la zona central la segunda señal con la que se opera.

derecha de la interfaz, se utiliza para visualizar la señal resultante. De este modo, comparando la primera y tercera zona de la ventana, el usuario podrá visualizar las modificaciones sufridas por la señal a su paso por el sistema.

Como se puede observar, también en la fig. 3.26, el usuario dispone de varios controles para modificar la visualización y sacar el máximo rendimiento de esta ventana. Si quiere conocer su funcionamiento puede consultar el manual de usuario, Cap.5 .

Finalmente hay que comentar que la ventana representada, fig. 3.26, corresponde a la ventana mostrada cuando se trabaja con operadores. Cuando el usuario haya seleccionado una operación básica se mostrará esta ventana ligeramente modificada, fig. 3.27, para mostrar la segunda señal de entrada al sistema simulado en lugar de acceder a la representación del conjunto de respuestas al impulso.

3.4.2. Visualización del conjunto de respuestas al impulso.

Esta funcionalidad representa el conjunto de respuestas al impulso que caracteriza al operador seleccionado. Sólo estará disponible cuando se trabaje con operadores, no para operaciones básicas, y cuando éstas sean lineales. La ventana de visualización, véase la fig. 3.28, está subdividida en tres ventanas pequeñas: dos para la representación en tres dimensiones del conjunto de respuestas al impulso en el modo seleccionado (real/imaginario o módulo/fase) y otra para las representaciones de los cortes de la curva en dos dimensiones y que alberga los controles que puede manipular el usuario. Esta ventana resulta muy interesante para el análisis del espacio

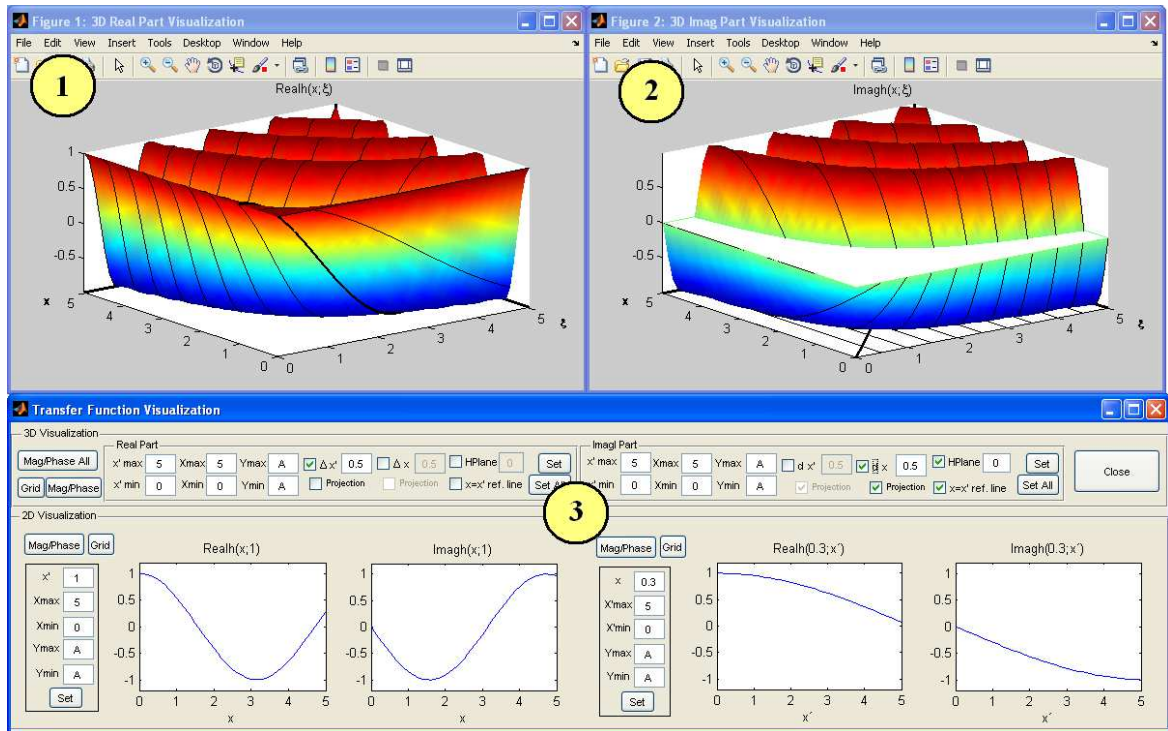


Figura 3.28: Ventana de visualización del conjunto de respuestas al impulso. Se accede a esta ventana desde la visualización principal y en ella se representa el conjunto de respuestas al impulso que caracteriza al operador seleccionado. Como se puede ver se divide en tres zonas principales: las dos primeras para la representación en tres dimensiones del conjunto de señales y la tercera para la visualización de los cortes deseados.

de operadores ya que en ella se pueden analizar las propiedades del sistema seleccionado.

De nuevo, si desea conocer el funcionamiento de cada uno de los controles de estas ventanas puede consultar el manual de usuario, Cap.5.

3.4.3. Visualización para el conjunto de funciones de transferencia del operador.

Esta funcionalidad representa el conjunto de funciones núcleo que caracteriza al operador seleccionado en el dominio espectral. Sólo estará disponible cuando se trabaje con operadores, no para operaciones básicas, y cuando éstas sean lineales. Como puede observarse en la fig. 3.29, esta ventana tiene una estructura muy similar a la del interfaz utilizado para el análisis del conjunto de respuestas al impulso en el dominio real. Está subdividida en tres ventanas pequeñas: dos para la representación en tres dimensiones del conjunto de funciones núcleo que caracteriza al operador en el dominio espectral en el modo seleccionado (real/imaginario o módulo/fase) y otra para las representaciones de los cortes de la curva en dos dimensiones y que alberga los controles que puede manipular el usuario.

El análisis generalizado de los sistemas lineales, sean éstos invariantes o no, en el dominio espectral sólo está disponible bajo la transformada de Fourier o el desarrollo en serie de Fourier. Se deja pendiente, como una de las líneas futuras más interesantes, el estudio y desarrollo de esta funcionalidad bajo otras transformadas como la de Hilbert o Bessel.

Para estos casos el programa dispone de una ventana de representación en dos dimensiones, véase la fig. 3.30, que permite visualizar la función de transferencia del sistema siempre y cuando éste sea lineal e invariante.

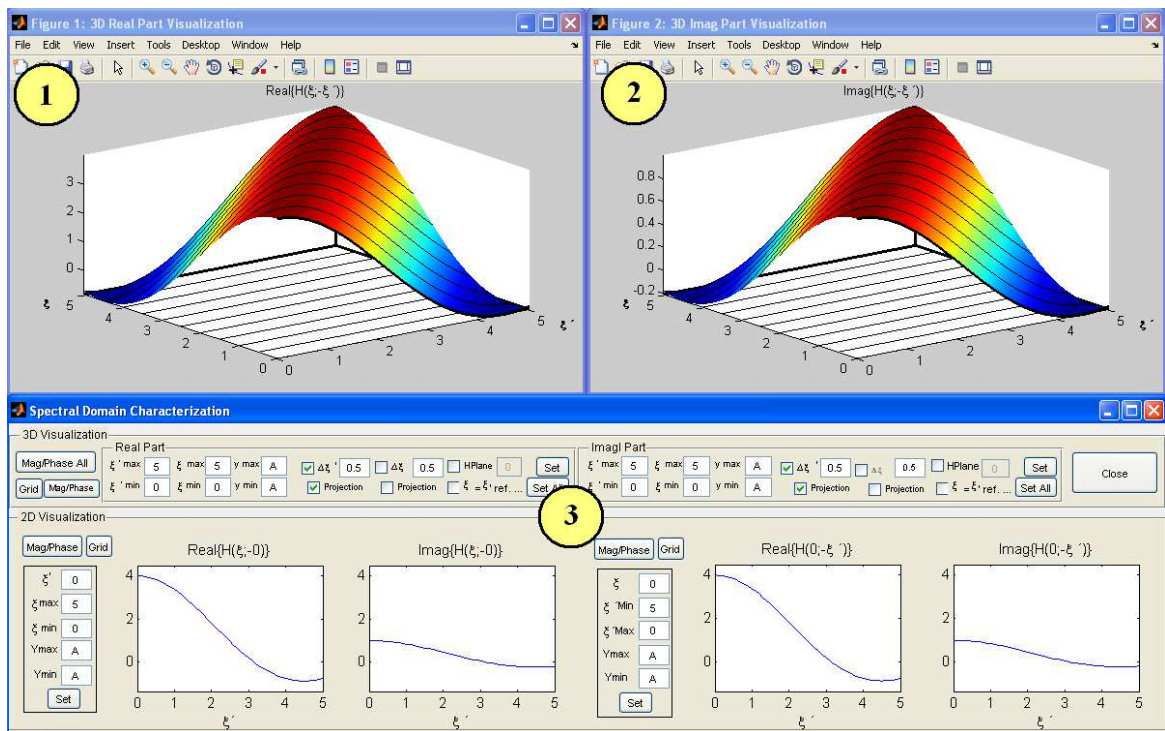


Figura 3.29: Ventana de visualización del conjunto de funciones núcleo que caracterizan al operador en el dominio espectral. Se accede a esta ventana desde la visualización principal y en ella se representa el conjunto de funciones de transferencia del al operador seleccionado. Como se puede ver se divide en tres zonas principales: las dos pimeras para la representación en tres dimensiones del conjunto de señales y la tercera para la visualización de los cortes deseados.

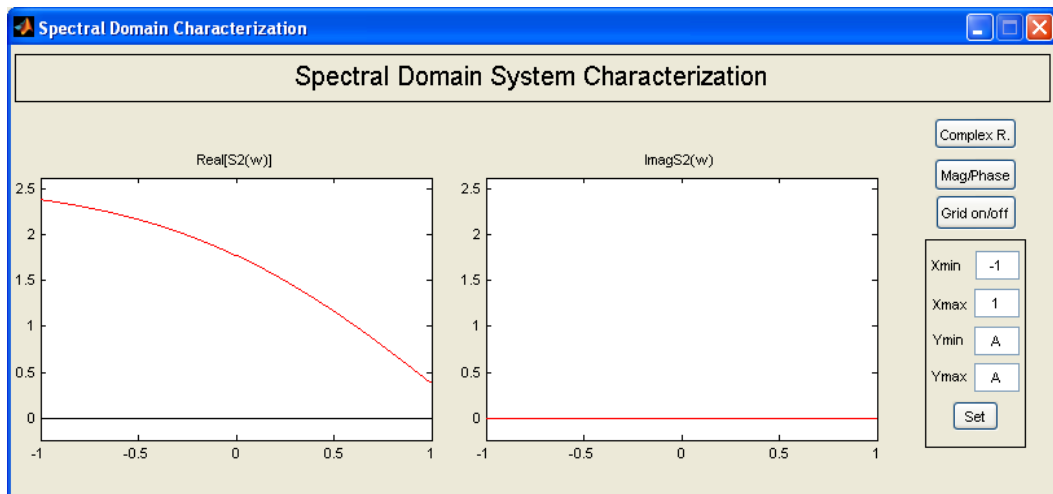


Figura 3.30: Ventana auxiliar para la visualización de la respuesta al impulso en el dominio transformado. Como se observa, sólo está disponible la representación en dos dimensiones por lo que sólo se podrá utilizar cuando se estudien operadores lineales e invariantes.

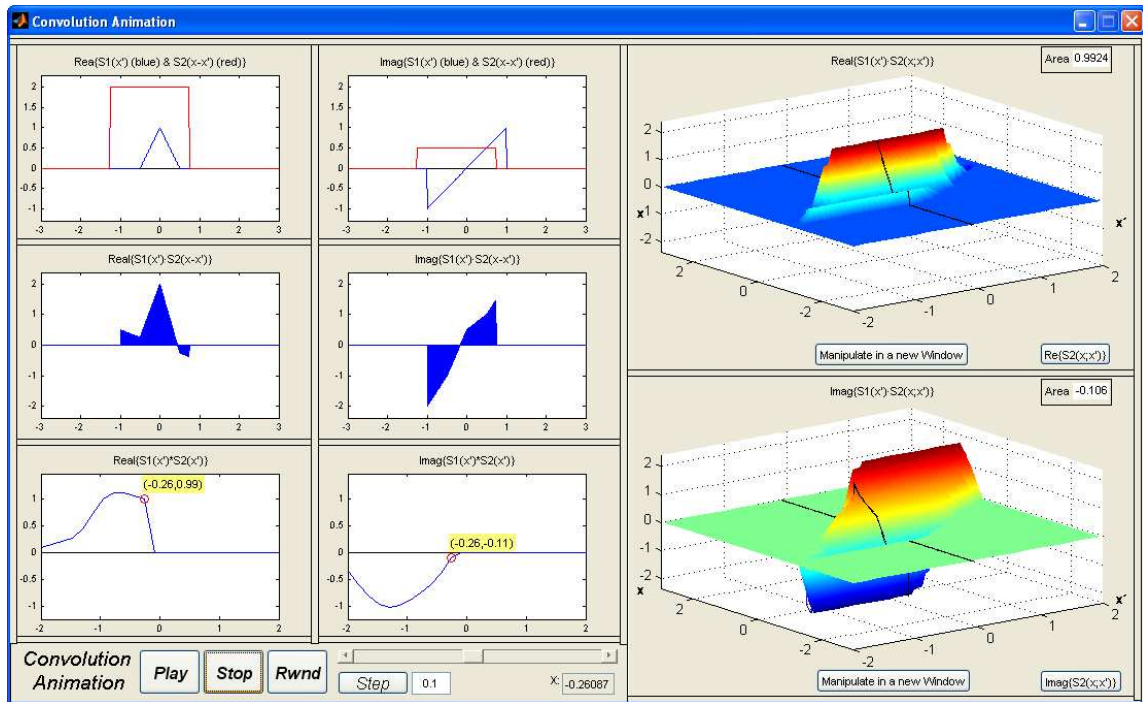


Figura 3.31: Ventana de animación. En esta ventana se visualiza paso a paso el funcionamiento del sistema seleccionado. Está disponible para los operadores de convolución y correlación (lineales e invariantes) y para el operador de multiplicación (lineal no invariante).

3.4.4. Ventana de animación.

Con esta ventana el usuario podrá visualizar paso a paso el funcionamiento del sistema de comunicaciones. Está disponible para los operadores de convolución y correlación, lineales e invariantes, y para el operador de multiplicación, lineal no invariante. La ventana de animación, fig.3.31, se divide en dos zonas principales: las representaciones en dos dimensiones, a la izquierda, y las representaciones en tres dimensiones a la derecha. En las representaciones en tres dimensiones se visualiza el producto del conjunto de respuestas al impulso por la señal de entrada y, para cada instante de la animación, el área o punto considerado. En las representaciones en dos dimensiones se muestra, para cada instante, la proyección de la señal de entrada sobre la señal del conjunto de respuestas al impulso correspondiente, el producto entre ambas y la señal que se va obteniendo como resultado. Para conocer en profundidad el modo de utilización puede consultar el manual de usuario, Cap. 5.

3.5. Métodos numéricos existentes.

Con el objetivo de maximizar la versatilidad del programa y permitir la experimentación por parte del usuario el software dispone de una serie de métodos numéricos que se utilizan en la resolución numérica de las distintas operaciones. Estos métodos se han heredado de versiones previas del software por lo que en este apartado sólo se citarán los existentes. Para conocer en detalle el funcionamiento de estos métodos puede consultar [3] - [6]. Como es lógico, los métodos que se han creado para esta versión del programa de expondrán en profundidad.

A continuación se listan los métodos numéricos disponibles en el software:

- Operaciones básicas (suma, resta, multiplicación y división), operador suma, operador resta, operador multiplicación y operador modulador en amplitud.

- Operador escalado
- Operador identidad
- Operador desplazamiento
- Operador convolución
- Operador correlación
- Operador transformada de Fourier y su inversa
- Operador transformada de Hilbert y su inversa
- Cálculo de la norma, la energía y el producto escalar

4. FUNCIONAMIENTO INTERNO DEL SOFTWARE

A lo largo de este capítulo se presentarán de forma resumida los puntos más importantes relacionados con el funcionamiento interno del programa: se tratarán las estructuras y variables para el almacenamiento de los datos, las interfaces utilizadas, los métodos numéricos implementados y las bases de datos del software. Dado que el presente proyecto ha consistido en el desarrollo de una ampliación del software, muchos de estos aspectos a penas se han visto modificaos. Por tanto, sólo se pretende con este capítulo dar una idea general de la estructura interna del programa y exponer las novedades respecto a las versiones anteriores. Para conocer la información omitida en profundidad puede consultar [4] - [6].

4.1. Funcionamiento general.

El software está dividido en cinco módulos independientes, correspondientes a los cinco espacios de señales de partida, como ya se ha expuesto en la Secc.1.5.2. Las subrutinas utilizadas por cada uno de ellos será diferente pero, a su vez, las estructuras internas de los módulos serán muy similares entre sí.

En el organigrama de la fig 4.1, se ilustra esquemáticamente el funcionamiento interno del programa, donde los cuadros y flechas azules representan las acciones llevadas a cabo por el programa y las flechas rojas las efectuadas por el usuario.

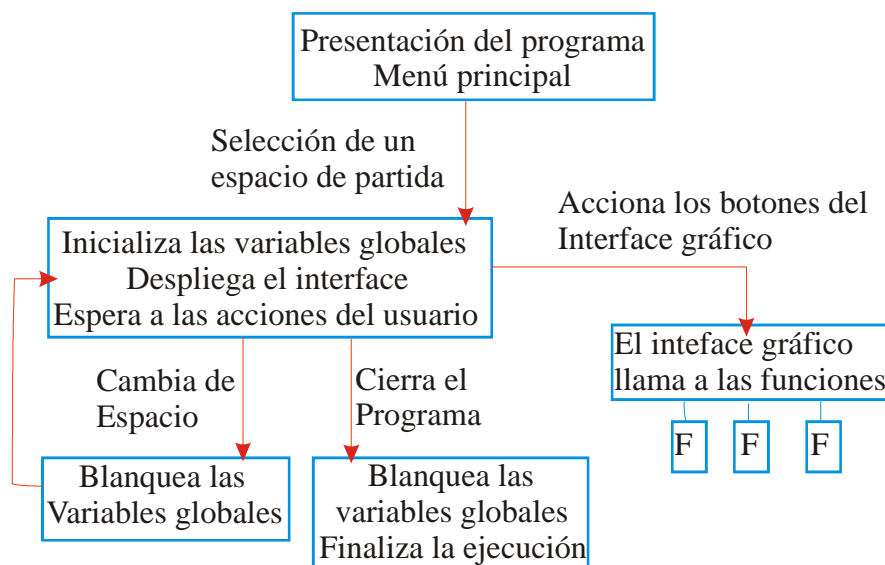


Figura 4.1: Organigrama que representa el funcionamiento interno del software. Representa las acciones que lleva a cabo el programa al ser implementado: los cuadros azules son las acciones realizadas por el software y las flechas rojas las ejecutadas por el usuario.

Al comenzar la ejecución del software, se implementan las rutinas de presentación e inicio y se despliega el menú principal. A partir de él, el usuario selecciona uno de los cinco espacios de partida disponibles haciendo que el programa acceda a uno de sus módulos internos. Después y hasta que el programa se cierre, éste seguirá la dinámica cíclica que se explica a continuación.

Como se observa en el organigrama, cada vez que el usuario seleccione un nuevo espacio de partida, el programa ejecutará dos operaciones: inicializará las variables globales necesarias para el correcto funcionamiento del programa y desplegará el interfaz gráfico para que el usuario pueda interactuar a través de él. Una vez realizadas, el programa se mantendrá a la espera de órdenes a ejecutar.

Cuando el usuario interactúa con el programa lo puede hacer con tres objetivos principales:

1. Trabajar en el espacio seleccionado. En este caso el programa llamará internamente a las subrutinas oportunas para satisfacer las exigencias del usuario. Para ello utilizará las variables globales necesarias.
2. Seleccionar otro espacio de partida. En este supuesto, lo primero que hará será blanquear las variables globales para evitar fallos posteriores. Después se cerrará el interfaz gráfico del espacio inicial y volverá a comenzar con su dinámica cíclica: inicializará las variables globales del espacio pertinente, desplegará su interfaz gráfico y esperará las órdenes del usuario.
3. Cerrar el programa. En este último caso, el programa blanqueará las variables globales para evitar posibles fallos y cerrará el interfaz gráfico terminando así con la ejecución del programa.

4.2. Herramientas utilizadas para el almacenamiento de datos.

El software *Signals & Systems Theory v5.0* dispone de varias herramientas para almacenar los datos que necesita para trabajar. En concreto, el programa utiliza una serie de variables y estructuras de datos globales, para facilitar el trabajo por parte del programa ya que para la implementación de la mayoría de las operaciones será necesaria la implementación de varias subrutinas. Este conjunto de herramientas se define e inicializa en un único archivo, que es el primer archivo ejecutado por cada uno de los cinco módulos del software. El mismo documento es el encargado de blanquearlas en el momento oportuno.

A continuación se listan y explican las variables y estructuras que utiliza el programa. Ya que las herramientas utilizadas por los distintos módulos son muy similares se expondrán a modo de ejemplo las del espacio de señales continuas aperiódicas definidas para todo x . Después se explicarán las pequeñas particularidades que existan en los otros espacios.

4.2.1. Variables y estructuras de datos globales del espacio de señales continuas aperiódicas definidas para todo x .

El programa cuenta con una gran cantidad de estructuras de datos. Sin embargo, se pueden agrupar según su propósito. Las distintas estructuras de cada grupo compartirán los campos internos permitiendo así explicar su funcionamiento de forma conjunta.

Como es lógico, muchas de estas variables ya estaban disponibles en versiones anteriores del software y no han sido modificadas. Sin embargo, se expondrán y explicarán muy brevemente con el objetivo de que el lector pueda comprender el funcionamiento del programa. Para más información sobre ellas puede consultar [4] - [6].

Tipo de Estructura	Nombre de las Estructuras
Definición teórica de señales	f11_conv_c, f12_conv_c, f21_conv_c, f22_conv_c, f31_conv_c, f32_conv_c, f1_dsf_a, f1_dsf_b, f2_dsf_a, f2_dsf_b, f3_dsf_a, f3_dsf_b, aux11global aux12global, aux21global, aux22global
Definición numérica de señales	f11v, f12v, f21v, f22v, conv1, conv2, dsf1_a, dsf1_b, dsf2_a, dsf2_b, dsf3_a, dsf3_b
Límites de definición	w1_def, w2_def, w3_def, w11_def, w12_def, w21_def, w22_def
Constantes Dom. Real	CONST_CONV_C
Constantes Dom. Transformado	CONST_DSF_C
Animación: Representación 3D	shortValues

Estructuras para la definición teórica de las señales. En estas estructuras se almacenan los datos y parámetros relativos a las diferentes partes de las señales cuando se definen de forma teórica. Entendiendo como primera parte la real o módulo, según el modo de definición de la señal, y la segunda la imaginaria o fase. Los campos que las componen son los siguientes:

- **Class:** almacena el código de la señal definida.
- **a, b, c, d, e, f:** recogen el valor de los parámetros teóricos para la definición de las distintas señales.
- **Mode:** contiene el modo de definición de la señal.

En las estructuras f11_conv_c, f12_conv_c, f21_conv_c, f22_conv_c, f31_conv_c, f32_conv_c, aux11global, aux12global, aux21global y aux22global el primer número representa la señal que está definiendo y el segundo la parte de ésta. De este modo f11_conv_c sería la estructura que define la parte real o módulo de la primera señal.

Por el contrario, en las estructuras f1_dsf_a, f1_dsf_b, f2_dsf_a, f2_dsf_b, f3_dsf_a y f3_dsf_b el número representa la señal y la letra la parte, refiriéndose a a la primera parte y b a la segunda. Estas últimas estructuras se utilizarán para almacenar los parámetros de las señales que representen el análisis espectral del sistema.

Además, en las estructuras aux11global, aux12global, aux21global y aux22global aparecen otros dos campos:

- **isA:** identifica si la señal es o no analítica.
- **isD:** identifica si la señal es una función o una distribución.

Estructuras para la definición numérica de las señales. Se utilizan para guardar los datos y parámetros de las distintas partes de las señales definidas de forma numérica. Los campos que contienen son:

- **Xvalue:** vector que contiene el valor de las abscisas de la señal.
- **Yvalue:** vector que almacena los valores de ordenadas de la señal correspondientes a los valores de abscisas del campo anterior.
- **Class:** valor numérico para distinguir si la función contiene deltas de Dirac o no.
- **Mode:** modo de definición de la señal.

En las estructuras `f11v`, `f12v`, `f21v` y `f22v` el primer número representa la señal que está definiendo y el segundo la parte de ésta. De este modo `f11v` sería la estructura que represente la parte real o módulo de la primera señal. El programa las utilizará para almacenar el resultado del muestreo de la señal definida en forma teórica.

Las estructuras `conv1` y `conv2` representarán la primera y segunda parte respectivamente de la señal resultado. Guardarán numéricamente los datos calculados de las distintas operaciones.

En las estructuras `dsf1_a`, `dsf1_b`, `dsf2_a`, `dsf2_b`, `dsf3_a` y `dsf3_b` el número identificará la señal y la letra la parte de la señal, refiriéndose `a` a la primera parte y `b` a la segunda. El programa las utilizará para almacenar el resultado del muestreo de las señales del análisis espectral del sistema.

Estructuras para la definición de los límites de las señales. Estas estructuras se utilizan para la definición de señales numéricas. Estas señales, que son distintas de las señales definidas numéricamente, se definen de forma teórica pero su soporte es finito. El valor de dicho soporte se almacenará mediante la opción `Set Range`. Sus campos son los siguientes:

- **Xmin:** valor mínimo que tomarán las abscisas.
- **Xmax:** valor máximo que tomarán las abscisas.
- **Ymin:** valor mínimo que tomarán las ordenadas. Si se le asigna como valor el caracter `A` el programa decidirá su valor para que no se pierda información.
- **Ymax:** valor máximo que tomarán las ordenadas. Como antes se le puede asignar el caracter `A` con el mismo fin.

En este caso las estructuras se inicializan con unos valores por defecto siendo `-1`, `1`, `A` y `A` para `xmin`, `xmax`, `ymin` e `ymax` respectivamente.

En las tres primeras, el número que contiene el nombre de las distintas estructuras, `w1_def`, `w2_def` y `w3_def`, representa a la señal. En el resto, el primer número representa la señal y el segundo la parte de ésta, así `w11_def` sería la estructura correspondiente a la primera señal y la parte real o módulo de ella.

Estructura para la definición de las constantes del dominio real. Este grupo está compuesto por una única estructura. Como su propio nombre indica, contiene el valor de los distintos parámetros utilizados en las señales definidas en el dominio real. Está compuesta por los siguientes campos:

- **RES:** valor de la resolución, es decir, el número de muestras de las señales que se representan en pantalla..
- **CONV_RES:** valor de la resolución para la señal resultado.
- **FIRST_DELTA:** se utiliza para identificar las señales que tienen deltas de Dirac. Si el valor del campo `class` es menor que `FIRST_DELTA` no contiene deltas, si es `FIRST_DELTA`

contiene deltas de Dirac, si es `FIRST_DELTA+1` contiene derivadas de la delta de Dirac y si es `FIRST_DELTA+2` contiene derivadas dobles de la delta de Dirac.

- **MENU1:** se utiliza para identificar a un tipo de señales que tiene como parámetros los campos a-f de las estructuras para la definición teórica de señales o expresiones matemáticas de MATLAB o vectores de ordenadas y abscisas.
- **MENU2:** identifica las señales obtenidas en forma de muestras por el propio programa.

Esta estructura se inicializa con los valores adecuados y no se modifica a lo largo de la ejecución del programa, ya que contiene valores de referencia.

Estructura para la definición de las constantes del dominio transformado. Este grupo está compuesto por una estructura muy similar a la explicada en el punto anterior. La diferencia radica en que esta estructura contiene el valor de los distintos parámetros utilizados en las señales definidas en el dominio transformado. Está compuesta por los siguientes campos:

- **RES_VIEW:** resolución de las señales del dominio de partida.
- **RES_FT:** resolución de las señales del dominio transformado.
- **MENU1** y **MENU2:** iguales que sus homónimos en la estructura anterior.
- **MENU1A** y **MENU2A:** iguales que **MENU1** y **MENU2** para las señales del dominio transformado.

Estructura para la ventana de animación. Este tipo de estructuras se utilizan para ejecutar permitir la representación en tres dimensiones de la ventana de animación. Está compuesto por una única estructura con los siguientes campos:

- **vector:** vector que contiene el rango de la variable de definición. En el espacio continuo aperiódico sería el rango de definición de la señal numérica, es decir, $[x_{\min}, x_{\max}]$.
- **vector1:** este vector contiene la misma información que el campo anterior pero diezmada por un factor 5, con el objetivo de maximizar la agilidad del sistema.
- **vector2:** vector que contiene el rango de trabajo de la variable descriptiva. En el espacio continuo aperiódico sería el rango en el que se opera en la variable x' .
- **newMatrix1:** matriz que contiene la parte real del resultado de la multiplicación de la señal de entrada y el conjunto de respuestas al impulso $h(x; x')$. Dicho resultado se almacena en otra variable global denominada **INT1**, sin embargo, es necesario almacenarla en esta estructura porque será diezmada tanto en x como en x' tal y como ocurre con los vectores de ambas variables (**vector1** y **vector2**), con el objetivo de mejorar la velocidad de la animación.
- **newMatrix2:** contiene la misma información que el campo anterior pero almacenando, en este caso, la información relativa a la parte imaginaria de la señal.
- **INT1_b:** este campo contiene la misma información que **newMatrix1** pero sólo está diezmada en x , no en x' . Se utiliza para tener una mayor resolución en las representaciones en dos dimensiones que se van realizando sobre la curva.
- **INT2_b:** como en el caso anterior contiene la misma información que **newMatrix2** pero diezmada, únicamente, en x .

Al igual que en el caso anterior, esta estructura está inicializada con los valores adecuados y no se modifica a lo largo de la ejecución del programa.

El número de variables globales es más reducido aunque también se pueden agrupar por funcionalidades.

Tipo de variable	Nombre de las variables
Identificadores del tipo de señal	is_distribution_11, is_distribution_21 is_analitica_1, is_analitica_2, is_FT
Identificación de operación	operacionSistema
Modo de definición	mode_def1, mode_def2, mode_def3
Constantes Dom. Transformado	RES_VIEW, RES_FT
Identificador del tipo de análisis	isOperator
Handles para ventanas de representación 3D	hand
	real3DWinHan
	imag3DWinHan
Identificador de estado de la animación	firstStep
Identificación del análisis espectral	aesSelected

Variables que identifican el tipo de señal. Estas variables son de tipo numérico. Funcionan como si fuesen booleanas tomando sólo dos valores según cumplan o no una determinada característica.

- is_distribution_11 e is_distribution_21: identificarán si la señal considerada es una distribución o una función. El primer número de su nombre identifica la señal.
- is_analitica_1 e is_analitica_2: identifican si la señal es analítica o no, en cuyo caso será de soporte finito o numérica. En ellas el número identifica la señal considerada.
- is_FT: identifica si la señal resultado procede de una transformación de la señal.

Variable que identifica el tipo de operación. Esta variable se utiliza para conocer el último operador u operación que se ha seleccionado.

Variables para la definición del modo. Estas variables toman dos valores, parece booleanas, según el modo en que esté definida la señal. Siendo el primer modo el Real/Imaginario y el segundo el Módulo/Fase. El número que contiene su nombre identifica a la señal que representa.

Variables para la definición de las constantes del dominio transformado. Estas variables contienen el valor original de los campos de su mismo nombre de la estructura CONST_DSFC. Se necesita guardar su valor original en otra variable ya que cuando el usuario introduce un valor de precisión el campo de la estructura almacena el valor original multiplicado por el valor de precisión.

Variable identificar el tipo de análisis empleado. La variable isOperator se utiliza para diferenciar entre operadores (sumas, multiplicaciones, moduladores, convoluciones, transformadas, etc.) y operaciones básicas (suma, resta, multiplicación, división o visor de señales). Funcionará como una variable booleana tomando el valor 1 – *cierto* cuando se trabaje con un

operador y el valor 0 – *falso* cuando se haya seleccionado una operación básica. El valor de esta variable se comprobará, entre otros casos, para saber que controles mostrar u ocultar en la interfaz principal.

Variabes para almacenar los handles de las ventanas de representación 3D. Las variables `real3DWinHan` e `imag3DWinHan` se utilizan para almacenar los handle de las ventanas de representación en tres dimensiones del conjunto de respuestas al impulso.

Por otro lado, la variable `hand` se utiliza para guardar el handle de la ventana auxiliar de representación en tres dimensiones de la animación.

Estas variables son necesarias para saber si las ventanas están abiertas o no y poder representar la señal en ellas.

Variable para identificar el estado de la animación. Esta variable es utilizada por la ventana de animación para el correcto funcionamiento de todos sus controles. En concreto, la variable `firstStep` se utiliza para identificar si se está ejecutando la primera iteración de la animación. Se necesita para saber si hay que borrar el dibujo de la iteración anterior en las representaciones en tres dimensiones.

Variable para identificar el tipo de análisis espectral seleccionado. Esta variable es utilizada por el programa para identificar el tipo de análisis espectral generalizado que se ha seleccionado, es decir, para distinguir la transformada bajo la cual se analizará el sistema.

4.2.2. Variables y estructuras de datos globales del espacio continuo periódico.

En este espacio sólo aparece una nueva variable respecto al anterior. Por otra parte, las estructuras son las mismas.

Tipo de variable	Nombre de las variables
Definición de la señal Dom. Real	X_o

Variabes para la definición de las señales del dominio real. Esta nueva variable, X_o , se utiliza para definir y almacenar el valor del periodo de las señales del espacio. Se inicializa con el valor 1.

4.2.3. Variables y estructuras de datos globales del espacio discreto aperiódico.

Como en el caso anterior este espacio aporta tan solo una variable nueva. En este caso también desaparecen algunas variables y se modifican algunas estructuras.

Tipo de variable	Nombre de las variables
Definición de la señal Dom. Transformado	X_o

Variabes para la definición de las señales del dominio real. Esta nueva variable, X_o , se utiliza para definir el valor del periodo que tomarán las señales del espacio en el dominio transformado. Se inicializa con el valor 2π y no se puede modificar.

Variables eliminadas. Por tratarse de un espacio discreto, el concepto de distribución carece de sentido, por lo que todas las señales que se tratan se consideran funciones. Debido a ello desaparecen las variables `is_distribution_11` e `is_distribution_21`.

Estructuras modificadas. En los casos anteriores las estructuras para la definición de los límites de las señales `w1_def`, `w2_def` y `w3_def` tomaban valores de abscisas entre -1 y 1 . En el caso de espacios discretos esos valores nos darían sólo tres muestras por lo que se ha modificado su valor y las abscisas se considerarán, inicialmente, entre -10 y 10 .

4.2.4. Variables y estructuras de datos globales del espacio discreto periódico.

Este espacio también cuenta con una variable nueva. De nuevo hacemos desaparecer algunas variables y se modificamos algunas estructuras.

Tipo de variable	Nombre de las variables
Definición de la señal Dom. Real	X_0

Variables para la definición de las señales del dominio real. Esta nueva variable, X_0 , se utiliza para definir y almacenar el valor del periodo de las señales del espacio. Se inicializa con el valor 10 .

Variables eliminadas y estructuras modificadas. Se trata de las mismas que en espacio anterior 4.2.3.

4.2.5. Variables y estructuras de datos globales del espacio de señales continuas aperiódicas definidas para valores positivos de la variable $x : S(0, \infty)$.

Este espacio tiene las mismas variables y estructuras globales que el espacio de señales continuas aperiódicas definidas para todo valor de x puesto de ejemplo en esta sección.

4.3. Funcionamiento interno de cada espacio de señales: interfaces.

Como ya se introdujo en la Secc. 4.1 el programa se sirve de dos herramientas principales para su ejecución: los interfaces gráficos y una librería de funciones. En este apartado se explicarán detalladamente los interfaces disponibles en el software así como su funcionamiento.

Se comenzará describiendo el funcionamiento de las ventanas para el espacio de señales continuas aperiódicas definidas para todo valor de x , $S(-\infty, \infty)$, y después se detallarán las diferencias que aporten los otros espacios con respecto a el primero.

4.3.1. Interfaces del módulo de señales continuas aperiódicas definidas para todo x .

Cuando el programa arranca su ejecución solicita al usuario que seleccione un espacio de partida. Una vez que el usuario seleccione uno de los cinco disponibles se iniciará, internamente, un módulo. En este apartado se tomará como ejemplo el espacio de partida de señales continuas aperiódicas definidas para todo x . La ejecución del módulo comienza con la declaración e inicialización de las variables y estructuras globales descritas en el apartado anterior, 4.2, mediante el archivo `continuousAperiodic.m`. Desde este mismo archivo se llama al interfaz gráfico principal del módulo y se blanquean las herramientas globales en el momento oportuno.

Interfaz principal. Es el interfaz que se abre cuando se selecciona como espacio de partida el continuo aperiódico. Se encuentra en los archivos `continuousAperiodicMenu.fig` y `continuousAperiodicMenu.m`. A partir de él se accede a todas las funcionalidades del módulo. En la figura 4.2 se muestra la pantalla con los controles etiquetados para facilitar su explicación.

1. Espacio de partida: es un menú desplegable que permite seleccionar el espacio de las señales de entrada del sistema. Pulsando sobre su flecha se verán los cinco espacios disponibles en el programa. En este caso estará seleccionando el espacio continuo aperiódico. Cuando se seleccione otro espacio el programa blanquea las variables y estructuras globales del módulo, cierra el interfaz y comienza a ejecutar otro módulo.
2. Esquema del operador seleccionado: en esta imagen se representará el esquema del operador u operación seleccionado. En este caso, al no estar seleccionado ninguno, se muestra la típica caja negra con un operador genérico. Esta imagen se considera de gran ayuda para que el usuario comprenda el funcionamiento de cada operador. También sirve para comprender la diferencia entre operaciones básicas con dos señales y los operadores homónimos disponibles en el software.
3. Espacio de llegada: como para el espacio de entrada, el control es un menú desplegable que muestra el espacio de salida del sistema. Este menú sólo cuenta con una opción que variará en función de las transformaciones a las que el sistema someta a la señal de entrada, es decir, el espacio de salida variará pero no es modificable por el usuario. En la figura se muestra vacío, ya que al no haber un operador seleccionado el espacio de salida es desconocido.
4. Ayuda de los espacios: al pinchar sobre este botón se abre una ventana de ayuda con la definición de los espacios. Este botón está replicado tanto para el espacio de entrada al sistema como para el de salida.
5. Explicación del operador u operación seleccionado: en esta caja de texto se mostrará información sobre el comportamiento del operador de modo que el usuario pueda conocer su funcionalidad antes de utilizarlo.
6. Esquema algebraico del espacio de las señales de entrada: en esta imagen se muestra el esquema algebraico del espacio de las señales de entrada, de modo que el usuario pueda visualizar de un modo más intuitivo la teoría generalizada de señales y sistemas.
7. Esquema algebraico del espacio de las señales de salida: en este espacio se muestra, al igual que en el control anterior, el esquema algebraico del espacio de las señales de salida. Al no estar definido el operador, el espacio de salida se desconoce y, por lo tanto, no se muestra su esquema.
8. Producto escalar del espacio de entrada: este menú desplegable permite seleccionar la definición de producto escalar que el usuario desee para el espacio de entrada. Actualmente el programa sólo cuenta con una por lo que será siempre la misma. Este menú está replicado para el espacio de salida de modo que las definiciones algebraicas pueden ser diferentes en ambos espacios.
9. Norma del espacio de entrada: menú desplegable para seleccionar la definición de norma deseada para el espacio de señales de entrada. Actualmente el programa sólo cuenta con una. Como en el caso anterior, está repetido para el espacio de salida.
10. Distancia del espacio de entrada: menú desplegable que permite al usuario escoger la definición de distancia que desee para el espacio de entrada. Como en los casos anteriores, la versión actual del programa sólo cuenta con una. De nuevo, este control está duplicado en el espacio de salida del sistema.

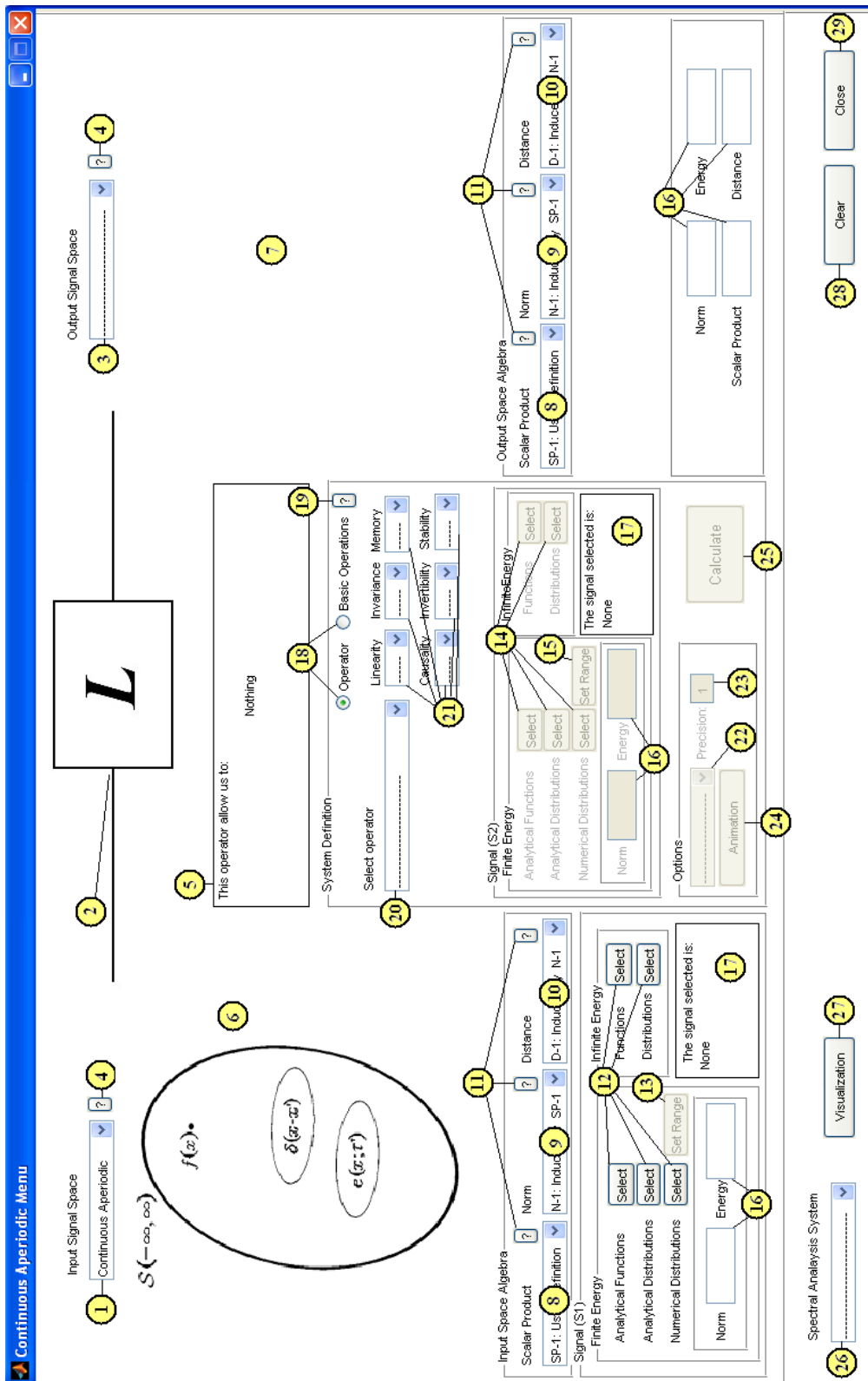


Figura 4.2: Interface principal con el espacio de entrada: Continuo Aperiódico. Esta ventana será la utilizada para acceder a todas las funcionalidades del programa. Además el usuario puede obtener en ella los resultados numéricos disponibles.

11. Ayudas para el álgebra del espacio: estos botones, disponibles tanto en las definiciones algebraicas del espacio de salida como en las del espacio de llegada, abren una ventana de ayuda con la definición matemática del parámetro algebraico correspondiente.
12. Selectores de S1: estos botones despliegan los distintos selectores de señales. La señal definida mediante estos selectores será la señal 1, es decir, S1.
13. Definición de rango de S1: pulsando este botón se abre la ventana que permite establecer el rango de definición en las señales de soporte finito. Esta nueva ventana que permite introducir los valores del rango y previsualizar la señal 1 en el espacio seleccionado. El valor de estos límites se guardará en la estructura `w1_def`.
14. Selectores de S2: botones que abren los distintos selectores de señales. La señal definida mediante estos selectores será la señal 2, es decir, S2.
15. Definición de rango de S2: como para la señal 1, pulsando este botón se accede a la ventana que permite establecer el rango de definición en las señales de soporte finito. Desde la ventana se pueden introducir los valores del rango y previsualizar la señal 2 en ese espacio. El valor de estos límites se guardará en la estructura `w2_def`.
16. Resultados algebraicos: son cajas de texto que muestran el resultado de cálculos de carácter algebraico. Muestran el valor de la norma y la energía de todas las señales. Este valor se calcula al definir las señales o calcular la señal resultado. También muestran los valores de producto escalar y distancia de las señales S1 y S2. Estos parámetros se calculan al realizar las distintas operaciones, dependiendo de cuál sea la definición seleccionada en los menús desplegables mediante las rutinas `pEscalar_Callback` y `calcularDistancia`, siempre y cuando el operador trabaje con dos señales.
17. Ayuda de las señales definidas: estos cuadros de texto muestran las definiciones de las señales seleccionadas para el usuario para cada parte de la señal. Sirven a modo de recuerdo al no estar su visualización en la misma pantalla.
18. Selector del modo de análisis: estos botones se utilizan para seleccionar si lo que se analiza es un operador o una operación básica con dos señales. Es necesario ya que en función del modo de análisis seleccionado se mostrarán u ocultarán varios controles y se modificarán las opciones seleccionables en el control 20.
19. Ayuda del tipo de análisis: al pulsar este botón se abrirá una ventana de ayuda en la que se explican las diferencias entre trabajar con un operador y con una operación.
20. Selector del operador: con este menú desplegable se selecciona la operación o el operador que se desee aplicar a la señal de entrada. Entre otros están disponibles operadores transformada, convoluciones, multiplicaciones, etc. Su contenido variará en función del tipo de análisis que se haya seleccionado en el control 18.
21. Menús de propiedades: estos menús desplegables sólo se mostrarán cuando se esté analizando un operador. En ellos el programa muestra si el operador es lineal, invariante, tiene memoria, es causal, invertible o estable.
22. Opciones de cálculo: este menú varía sus opciones dependiendo del operador seleccionado. Indica si las operaciones se calculan de forma teórica o numérica. En el último caso muestra los métodos numéricos disponibles para la operación.
23. Precisión: en esta caja de texto el usuario introduce un número que define la precisión de cálculo. Se utiliza para realizar las transformadas de Fourier y sus inversas implementando métodos numéricos. El programa parte de un valor de precisión contenido en la estructura `CONST_DSFC`, en concreto es el campo `RES_VIEW` o el `RES_FT`. Cuando el software

implementa el método numérico multiplica el valor introducido en esta caja de texto por el valor de precisión definido inicialmente.

24. Animación: este botón abre una nueva ventana que muestra la animación del cálculo realizado por el sistema. Se utiliza en los operadores multiplicación, convolución y correlación activándose sólo cuando se implementa un método numérico.
25. Calcular: este botón se utiliza para calcular la salida del sistema. Además calcula el producto escalar y la distancia de las señales S1 y S2 cuando ambas están definidas e intervienen en el sistema. Este botón ejecuta funciones distintas según el valor del menú selector del operador: `basic_operation_Callback`, `convolution_operation_Callback`, `correlation_operation_Callback`, `calcularFT_Callback` y `calcularIFT_Callback`. Sus resultados se almacenan en las estructuras `conv1` y `conv2`.
26. Análisis Espectral del Sistema: este menú desplegable se utiliza para seleccionar el modo espectral en que se desea analizar el sistema. El resultado de dicho análisis se muestra en la ventana de visualización. Actualmente está disponible el análisis bajo la transformada de Fourier, bajo la transformada de Hilbert y bajo la transformada de Bessel (la disponibilidad de estos análisis variará para cada espacio de partida seleccionado).
27. Visualización: es el botón encargado de abrir la ventana de visualización, que se explicará a continuación.
28. Botón Clear: este botón se utiliza para borrar la pantalla. Al pulsarlo el programa blanquea las variables y estructuras globales, cierra el interfaz gráfico y vuelve a iniciar la ejecución del mismo módulo.
29. Botón Close: se utiliza para cerrar el programa. También blanquea las variables y estructuras globales y cierra el interfaz gráfico terminando así con la ejecución del programa.

Interfaz de visualización. Este interfaz se abre desde la pantalla principal. Se utiliza para

visualizar las señales y resultados obtenidos en el módulo seleccionado. Está almacenado en los archivos `continuousAperiodicOperatorVisualization.fig` y `continuousAperiodicOperatorVisualization.m`. Al abrirse se ejecutan las distintas funciones para dibujar las señales: `printFunctions`, `printDistributions` y `draw_conv`. También se ejecutan las funciones `calcularFT_sistema_Callback`, `calcularHT_sistema_Callback` y `draw_dsf_system` para calcular y mostrar el análisis espectral del sistema si así se requiere. A continuación, el programa se mantiene a la espera de órdenes por parte del usuario.

En la figura 4.3 se muestra el interfaz gráfico con los controles etiquetados para facilitar su explicación.

Como se puede observar en la figura 4.3 la pantalla se divide en seis apartados. En los tres apartados superiores corresponden al análisis del sistema en el dominio real y los tres inferiores en el dominio transformado. Al mismo tiempo en las gráficas situadas a la izquierda se representará la entrada al sistema, a continuación se muestran los controles para acceder al estudio gráfico del operador seleccionado y, finalmente, en la zona derecha de la pantalla se representa la señal resultado.

A continuación se explica la funcionalidad de cada uno de los controles etiquetados en la fig.4.3.

1. Primera gráfica de representación: en las gráficas etiquetadas con el número 1 se representarán las primeras partes, partes reales o módulos, de las señales correspondientes, señal de entrada y resultado en los dominios reales y transformados.

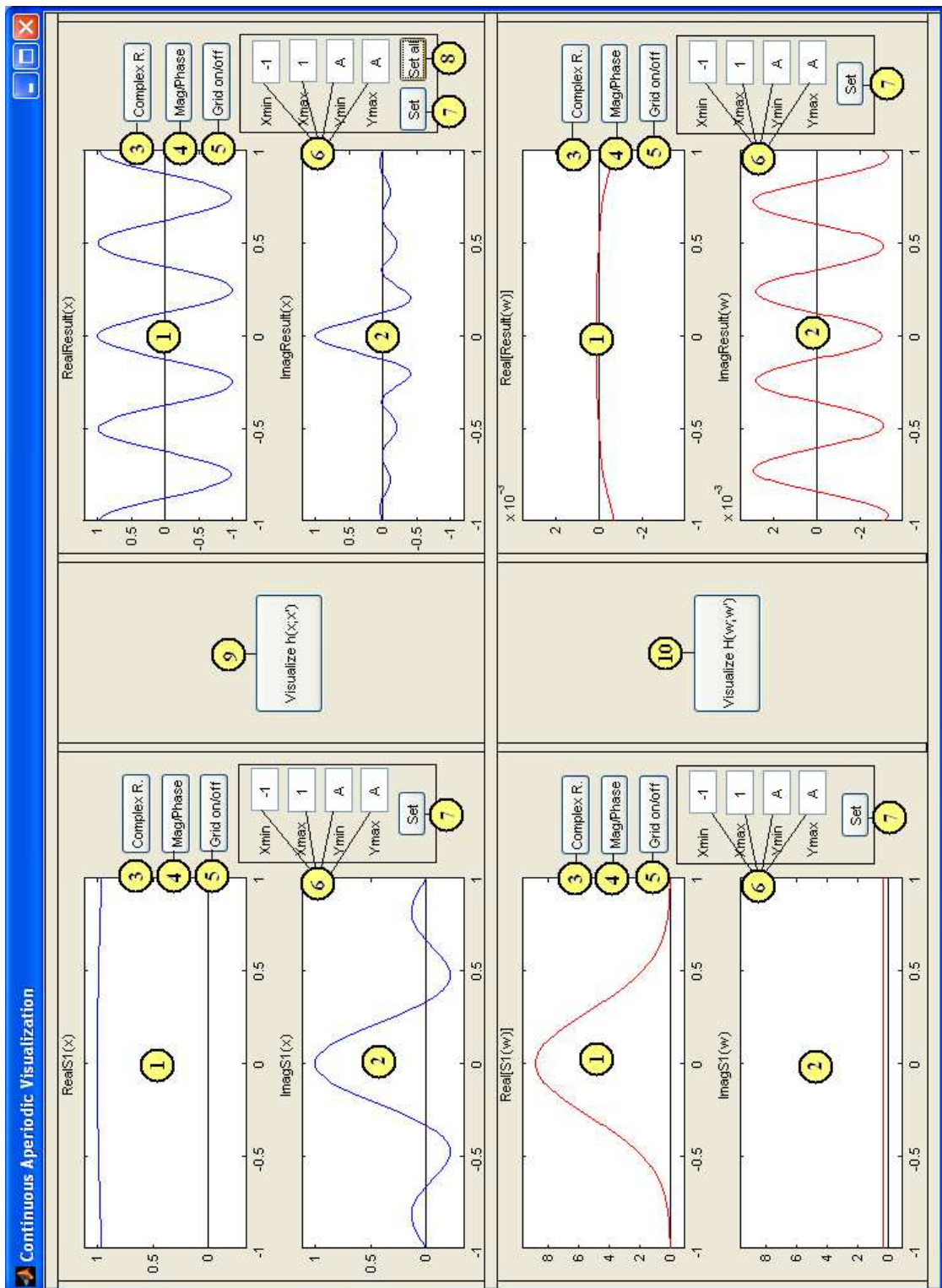


Figura 4.3: Interface de visualización. En esta ventana se muestran todos los resultados gráficos que ofrece el programa: la señal de entrada al sistema, la señal resultado y el análisis espectral del sistema completo. Además, desde esta ventana se puede acceder a las ventanas para el estudio del operador seleccionado, donde el usuario podrá visualizar el conjunto de respuestas al impulso $h(x; x')$ que lo caracteriza.

2. Segunda gráfica de representación: en las gráficas etiquetadas con el número 2 se representarán las segundas partes, partes imaginarias o fases, de las señales correspondientes, señal de entrada y resultado en los dominios reales y transformados.
3. Representación compleja: al pulsar este control se abre una ventana con la representación compleja de la señal correspondiente.
4. Modo de representación: este botón permite visualizar la función en los distintos modos, Real/Imaginario o Módulo/Fase. Este control no modifica el modo de definición de la señal, tan sólo varía la representación.
5. Rejilla para las gráficas: pulsando el control el programa pone o quita la rejilla en las gráficas de la señal. La rejilla ayuda al usuario a visualizar y obtener datos concretos de la información representada con facilidad.
6. Rango de visualización: estas cajas de texto permiten al usuario introducir el rango de visualización que desee para la señal. Distintos del soporte finito que se define en algunas señales, estos valores sólo toman sentido para la representación gráfica.
7. Botón Set: este botón recalcula las señales para los valores introducidos en las cajas de texto anteriores.
8. Botón Set all: se utiliza para dar a todas las gráficas del interfaz el mismo rango de visualización que el de la señal resultado. Con él se pretende ayudar a la comprensión global del sistema.
9. Estudio del operador en el dominio real: pulsando este botón se accede a una nueva ventana, que se explicará a continuación, en la que se puede analizar gráficamente el operador que se ha seleccionado. En esta nueva ventana se representará el conjunto de respuestas al impulso $h(x; x')$ siempre que el operador sea lineal.
10. Estudio del operador en el dominio espectral: este otro botón permite analizar el operador gráficamente en el dominio espectral. Al pulsarlo se abrirá una nueva ventana, explicada en apartados posteriores, en la que se representará el conjunto de funciones núcleo que caractericen al operador en el dominio transformado, siempre y cuando éste sea lineal.

La ventana mostrada en la fig. 4.3 es la que el programa utiliza cuando se ha seleccionado un operador. Cuando el usuario decide trabajar con operaciones básicas con dos señales se muestra una ventana diferente, almacenada en los archivos `continuousAperiodicVisualization.fig` y `continuousAperiodicVisualization.m`. Esta ventana, véase la fig. 4.4, es prácticamente idéntica. La mayor diferencia es que esta ventana no dispone de controles para acceder al estudio gráfico del operador sino que, en lugar de eso, muestra las gráficas correspondientes a la segunda señal con la que se opera. Además, la ventana incluye dos nuevos controles:

- 11: Cargar señal: este botón permite cargar una señal desde un fichero para visualizarla. De esta manera, el programa permite comparar unas señales con otras. Para ello utiliza la rutina `loadf_Callback`.
- 12: Salvar señal: este otro botón guarda una señal que se está visualizando en un fichero de datos mediante la función `savef_Callback`.

Interfaz para la visualización del conjunto de respuestas al impulso del operador seleccionado en el dominio real. Esta ventana se abre cuando se pulsa el botón correspondiente de la ventana de visualización y sólo estará disponible cuando se esté analizando un operador. En ella se muestra el conjunto de respuestas al impulso $h(x; x')$ siempre que

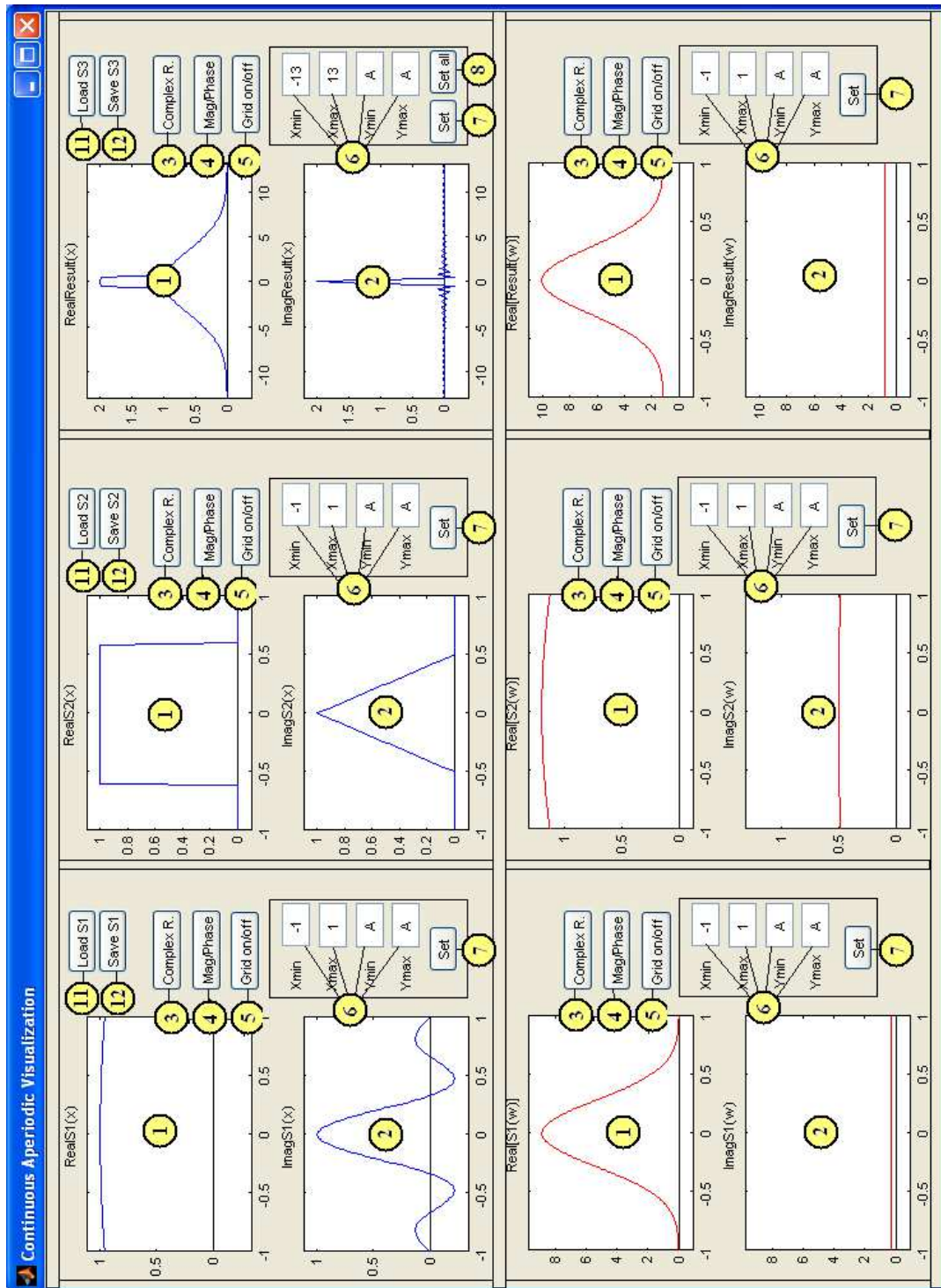


Figura 4.4: Interface de visualización cuando se analiza una operación básica en lugar de un operador. En esta ventana se muestran todos los resultados gráficos que ofrece el programa: las dos señales con las que se opera, la señal resultado y el análisis espectral del sistema completo.

exista, es decir, cuando el operador sea lineal. La ventana está almacenada en los ficheros `transferFunctionVisualization.m` y `transferFunctionVisualization.fig`.

Al abrirse se ejecutan las distintas funciones para dibujar las señales: `print3DSignal` y `print2DSignal`. También se ejecuta la función `buildMatrix` para calcular el conjunto de respuestas al impulso asociado al operador definido.

En la fig.4.5 se muestra el interfaz con los botones etiquetados para su explicación.

1. Representación de la primera parte del conjunto de respuestas al impulso: en esta gráfica se representa la parte real o el módulo del conjunto de funciones $h(x; x')$ que caracteriza al operador. Para maximizar las posibilidades de estudio de la curva en tres dimensiones se ha optado por representarla en una ventana independiente con todos los controles disponibles en MatLab.
2. Representación de la segunda parte del conjunto de respuestas al impulso: en esta gráfica se representa la parte imaginaria o fase del conjunto de funciones $h(x; x')$ que caracteriza al operador. Sus características son idénticas que las explicadas para la otra parte de la representación en tres dimensiones.
3. Representación de la parte real o módulo de los cortes del conjunto de señales $h(x; x')$: en estas gráficas se muestra la parte real o módulo de los distintos cortes en x o x' . Es decir, se representan las señales $\text{Re}\{h(k, x')\}$ o $\text{Mag}\{h(k, x')\}$, donde k es un escalar definido por el usuario.
4. Representación de la parte imaginaria o fase de los cortes del conjunto de señales $h(x; x')$: en estas gráficas se representan, como en el caso anterior, $\text{Im}\{h(x, k)\}$ o $\text{Phase}\{h(x, k)\}$, donde k es un escalar definido por el usuario.
5. Modo de representación global: este botón permite modificar el modo de representación de todas las gráficas de la ventana a la vez. Permite al usuario visualizar las señales en los distintos modos, Real/Imaginario o Módulo/Fase.
6. Modo de representación: este botón permite visualizar la función representada en cada conjunto de gráficas en los distintos modos, Real/Imaginario o Módulo/Fase.
7. Rejilla para las gráficas: pulsando el control el programa pone o quita la rejilla en las gráficas de la señal. La rejilla ayuda al usuario a visualizar y obtener datos concretos de la información representada con facilidad.
8. Rango de visualización: estas cajas de texto permiten al usuario introducir el rango de visualización que desee para la señal. Según se esté modificando una de las partes de la visualización 3D o uno de los cortes el número de parámetros introducidos y su significado variará.
9. Controles para dibujar líneas de x' constante sobre la representación 3D. Esta funcionalidad se compone de dos controles: un check para mostrar y ocultar las rectas y una caja de texto para introducir la separación entre las líneas representadas.
10. Proyección de las líneas de x' constante: activando este control, y siempre que la opción de mostrar las líneas de x' esté activada, se mostrará la proyección de las líneas dibujadas sobre la curva en el plano inferior de la gráfica.
11. Controles para dibujar líneas de x constante sobre la representación 3D. Esta funcionalidad se compone de dos controles: un check para mostrar y ocultar las rectas y una caja de texto para introducir la separación entre las líneas representadas.

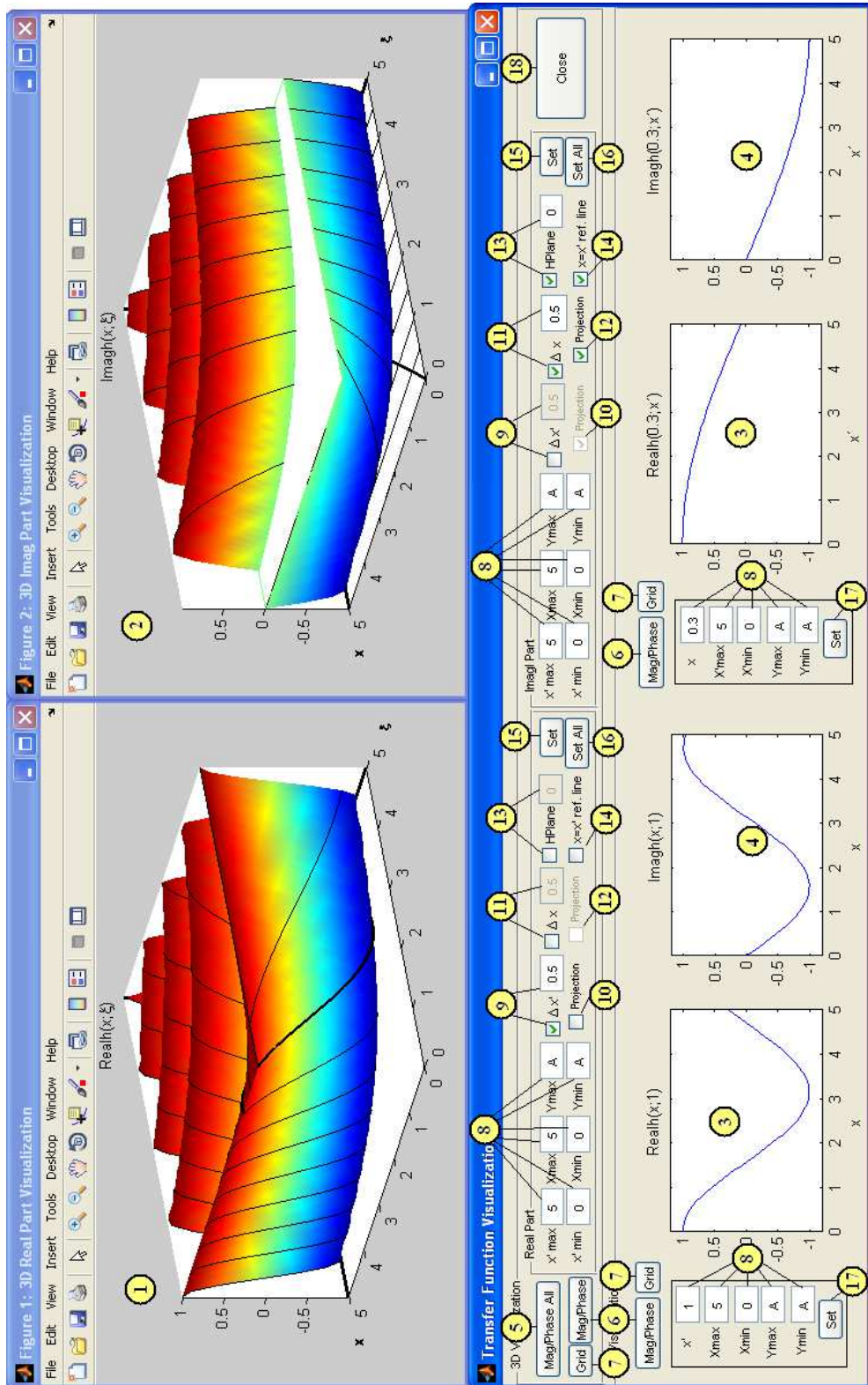


Figura 4.5: Interfaz para el estudio del operador en el que se representa su conjunto de respuestas al impulso $h(x; x')$ siempre que el operador sea lineal. En esta ventana se pueden estudiar las propiedades del operador y visualizar, además, cortes del conjunto de respuestas al impulso para facilitar el estudio de la curva.

12. Proyección de las líneas de x constante: activando este control, y siempre que la opción de mostrar las líneas de x esté activada, se mostrará la proyección de las líneas dibujadas sobre la curva en el plano inferior de la gráfica.
13. Representación de un plano horizontal: con estos controles se puede dibujar un plano horizontal sobre la gráfica en tres dimensiones. Se considera interesante ya que puede servir de ayuda para visualizar correctamente algunas partes de una representación. Esta funcionalidad está compuesta por dos controles: un check para dibujar o borrar el plano horizontal y una caja de texto para introducir la altura a la que se pretende dibujar.
14. Representación de la línea de referencia $x = x'$: este control se utiliza para dibujar la línea de referencia sobre la gráfica en tres dimensiones. Esta línea facilitará al usuario la comprobación de la propiedad de invarianza del operador estudiado.
15. Botón **Set**: este botón recalcula la parte real o imaginaria del conjunto de respuestas al impulso para los cambios introducidos en los controles explicados previamente.
16. Botón **Set all**: este botón se utiliza para representar tanto la parte real como la parte imaginaria del conjunto de respuestas al impulso con los cambios introducidos en los controles de la parte correspondiente sin tener que duplicar manualmente estos cambios.
17. Botón **Set** para las representaciones en dos dimensiones: este botón modifica la representación de los cortes del conjunto de señales $h(x; x')$ según los valores que haya introducidos en las cajas de texto que contienen la información sobre el rango de visualización.
18. Botón **Close**: como el interfaz se compone de tres ventanas independientes se ha incluido este botón que cierra las tres ventanas automáticamente.

Interfaz para la visualización del conjunto de funciones núcleo que representan a un operador en el dominio transformado. Se dispone de esta ventana para el estudio del análisis espectral generalizado. Esta funcionalidad permite representar el conjunto de funciones núcleo que caracterizan un operador en el dominio transformado para cualquier operador lineal siempre y cuando se analice bajo la transformada de Fourier. Este interfaz se encuentra en los ficheros `spectralDomainSystemCharacterization3D.m` y `spectralDomainSystemCharacterization3D.fig`. Al abrirse se ejecuta la rutina `draw_dsf_sistem` para representar la señal.

Al abrirse se ejecutan las distintas funciones para dibujar las señales: `print3DSpectralSignal` y `print2DSpectralSignal`. También se ejecuta la función `buildSpectralMatrix` para calcular el conjunto de respuestas al impulso asociado al operador definido.

En la fig.4.6 se muestra el interfaz. Como se puede observar sigue la misma estructura y posee los mismos controles que la ventana para la representación del conjunto de respuestas al impulso por lo que, para conocer sus funcionalidad, se invita al lector a estudiar la ventana expuesta anteriormente.

Cabe resaltar que, en el caso de los sistemas lineales e invariantes, la segunda representación en dos dimensiones no se utiliza para representar un corte del conjunto de funciones núcleo sino que se utiliza para representar la función de transferencia que caracteriza al operador, lineal e invariante, en el dominio transformado.

Interfaz para la visualización de la función de transferencia de un operador cuando no se analiza bajo la transformada de Fourier. El análisis espectral generalizado sólo se ha implementado en esta versión del software cuando se selecciona la transformada de Fourier. Cuando el análisis se realice bajo otras transformadas (Hilbert, Bessel), el programa sólo podrá representar la función de transferencia de los operadores lineales e invariantes y

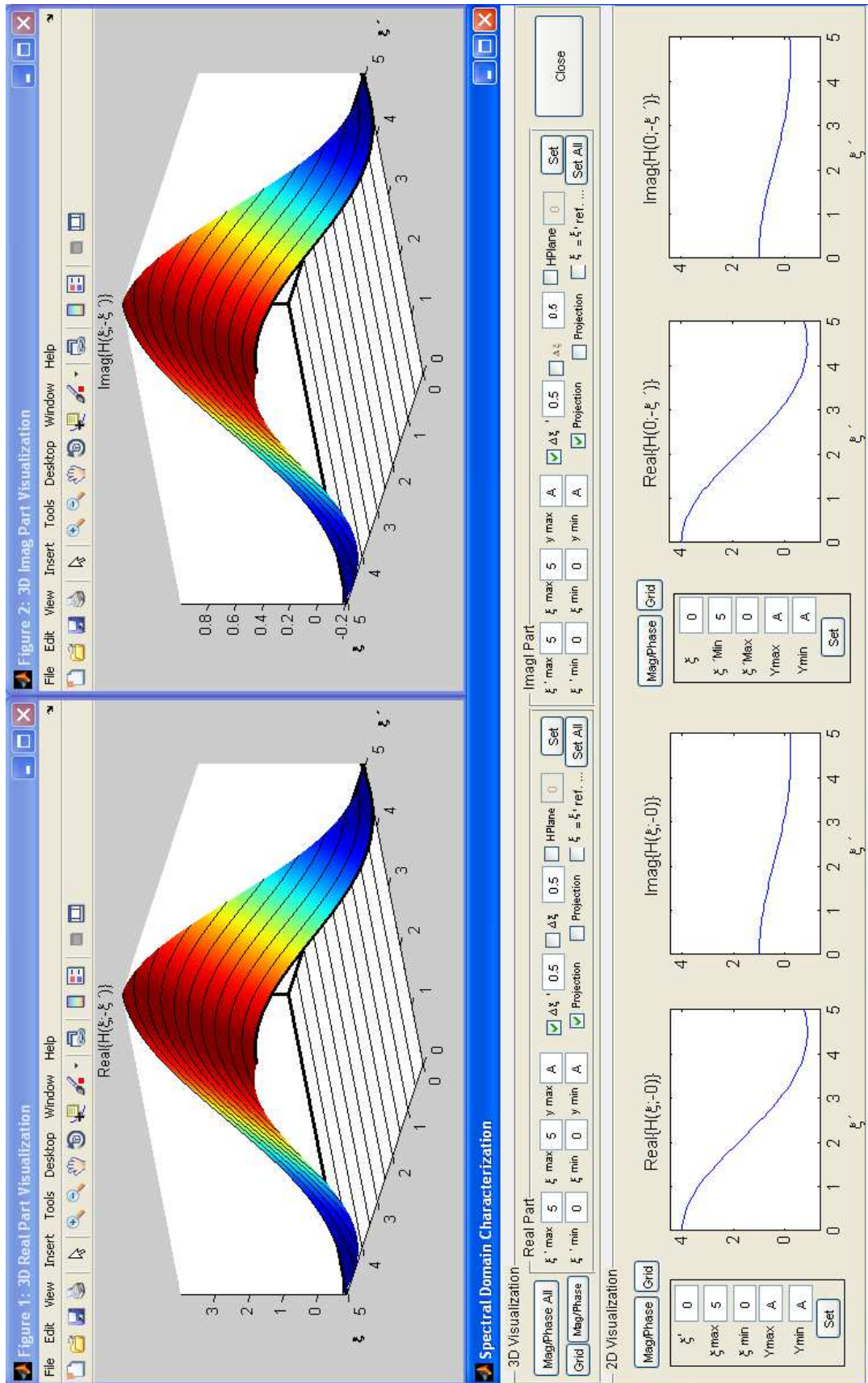


Figura 4.6: Ventana para la visualización del conjunto de funciones núcleo que representan a un operador en el dominio de la frecuencia. Como se observa sigue la misma estructura y posee los mismos controles que la ventana para la visualización del conjunto de respuestas al impulso del operador, por lo que se invita al lector a consultar la información relativa a dicha ventana para conocer el funcionamiento de ésta.

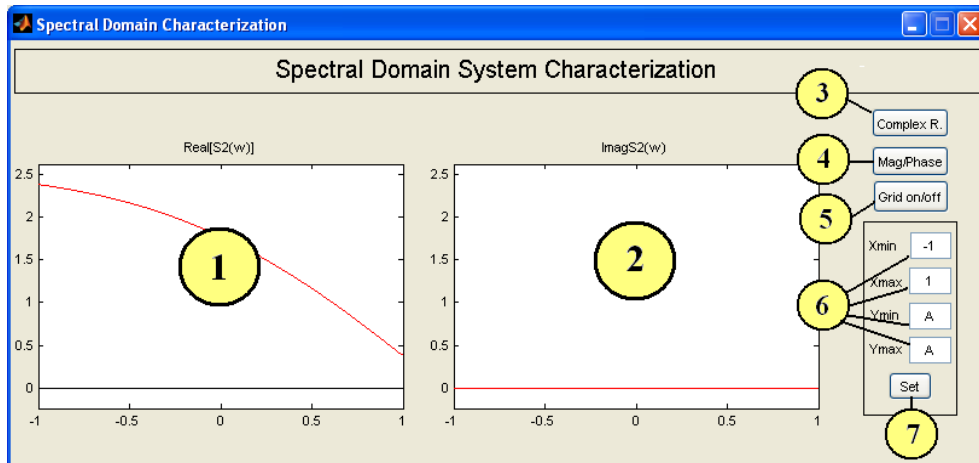


Figura 4.7: Ventana que muestra la caracterización del operador estudiado en el dominio espectral. Esta ventana sólo estará disponible para la visualización de la caracterización de operadores lineales e invariantes.

para ello utilizará esta ventana. Este interfaz se encuentra en los ficheros `spectralDomainSystemCharacterization.m` y `spectralDomainSystemCharacterization.fig`. Al abrirse se ejecuta la rutina `draw_dsf_sistem` para representar la señal.

En esta versión del software sólo está pensada para representar señales en dos dimensiones pudiendo sólo mostrar la caracterización espectral del operador cuando éste sea lineal e invariante. Sin embargo, se plantea como línea futura el desarrollo de este interfaz para el estudio de sistemas lineales no invariantes.

En la figura 4.7, se muestra el interfaz con sus controles etiquetados para su explicación:

1. Representación de la parte real o módulo de la señal: se muestra la primera parte de la función que caracteriza al operador representada en el dominio espectral.
2. Representación de la parte imaginaria o fase de la señal: se muestra la segunda parte de la función que caracteriza al operador representada en el dominio espectral.
3. Representación compleja: al pulsar este control se abre una ventana con la representación compleja de la señal.
4. Modo de representación: este botón permite visualizar la función en los distintos modos, Real/Imaginario o Módulo/Fase.
5. Rejilla: pulsando el control el programa pone o quita la rejilla en las gráficas de la señal. La rejilla ayuda al usuario a visualizar y obtener datos concretos de la información representada con facilidad.
6. Rango de visualización: estas cajas de texto permiten al usuario introducir el rango de visualización que desee para la señal.
7. Botón Set: este botón recalcula las señales para los valores introducidos en las cajas de texto anteriores.

Interfaz para la selección de señales. El espacio de señales continuas aperiódicas cuenta con cinco selectores de señales distintos. Dichos selectores están replicados para las dos señales definibles del programa S1 y S2. Se encuentran definidos en los ficheros de extensiones `.m` y `.fig` con los siguientes nombres: `continuousAperiodicFiniteDistributionSelector1/2`,

continuousAperiodicFiniteDistributionSelectorN1/2, continuousAperiodicSingularDistributionSelector1/2, continuousAperiodicFiniteFunctionSelector1/2 y continuousAperiodicInfiniteFunctionSelector1/2. Todos ellos comparten una estructura interna idéntica y un interfaz muy similar por lo que se explicará uno de ellos a modo de ejemplo.

En la figura 4.8 se expone el selector de funciones de energía finita con los controles etiquetados para su posterior explicación.

1. Selectores de modo: estos controles son de tipo check-box y sirven para seleccionar el modo de definición de la señal. Cuando se selecciona uno de ellos el otro deja de estar marcado para evitar errores. También se cambian los nombres de las partes de la señal. Cuando se pulsa el botón set, etiqueta 6, la función correspondiente evalúa el valor de estos controles para establecer el modo de la señal.
2. Definición de la señal: estos dos controles son menús desplegables y se utilizan para seleccionar la definición de la primera y la segunda parte de la señal, parte real o módulo y parte imaginaria o fase respectivamente.
3. Definición de parámetros: estas cajas de texto serán utilizadas por el usuario para introducir los valores necesarios según la señal seleccionada.
4. Botón Set: una vez el usuario ha seleccionado y definido los campos anteriores pulsa este botón para guardar la definición de la señal y cerrar la ventana. Al pulsarlo se ejecuta la rutina `set_Callback`.
5. Imágenes de ayuda: al seleccionar una función para cada parte de la señal, estas imágenes se actualizan y se muestran las definiciones de las señales seleccionadas de modo que el usuario pueda identificar a que parte de la definición matemática corresponde cada parámetro. Para mostrar las imágenes adecuadas se utiliza la función `gethelp`.

Inteface para la definición de los parámetros de la señal que caracteriza a un operador La ventana mostrada en el apartado anterior es la que se abre cuando se define una señal. En algunos operadores, estos selectores se activan para que el usuario defina a su gusto la señal que los caracterice, como por ejemplo el operador convolución o el operador multiplicación. Sin embargo, existen otros operadores que no necesitan la definición de una señal aunque sí admiten la modificación de algunos de sus parámetros, por ejemplo el operador modulador en amplitud o el operador desplazamiento. Cuando se selecciona uno de estos operadores el programa dará la posibilidad de modificar sus parámetros abriendo una ventana de apariencia y funcionamiento muy similar pero distinta.

Esta nueva ventana se denomina selector de parámetros y se almacena en los ficheros `signalParameterSelector.m` y `signalParameterSelector.fig`. La ventana se muestra en la figura 4.9, donde se han etiquetado sus controles para su explicación.

1. Cajas de texto que muestran la definición de la señal: estos dos controles no son manipulables por el usuario, simplemente muestran, a modo informativo, el nombre de la función que caracteriza al operador seleccionado.
2. Definición de parámetros: estas cajas de texto serán utilizadas por el usuario para introducir los valores necesarios según la señal seleccionada.
3. Botón Set: una vez el usuario ha seleccionado y definido los campos anteriores pulsa este botón para guardar la definición de la señal y cerrar la ventana. Al pulsarlo se ejecuta la rutina `set_Callback`.

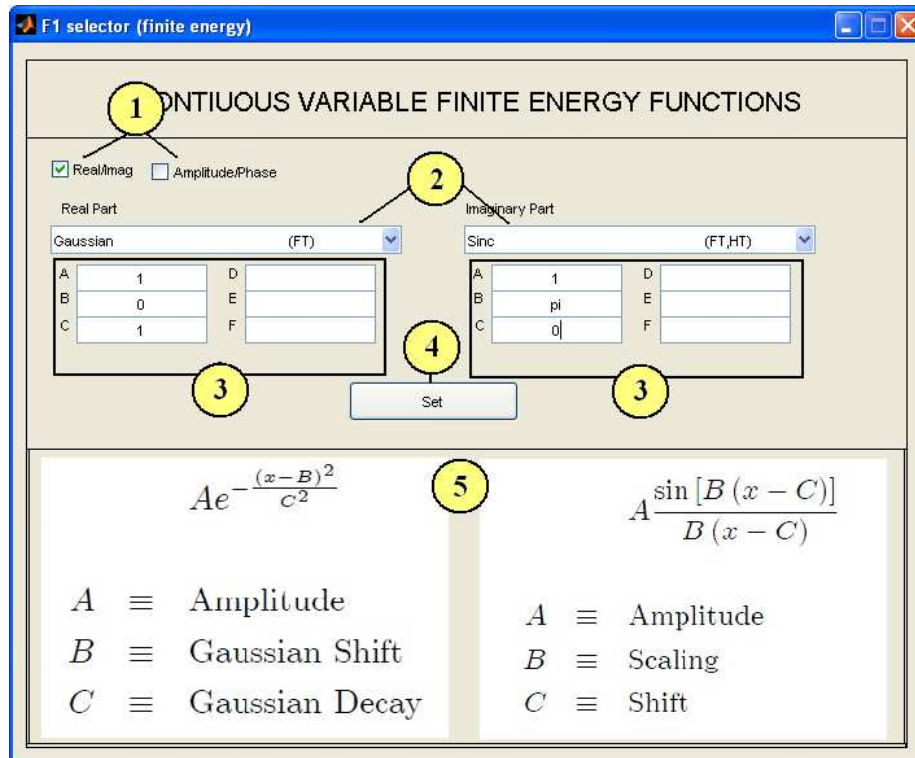


Figura 4.8: Selector de señales. A través de esta ventana se definen las señales con las que trabaja el programa: señales de entrada al sistema y señales que caracterizan al operador definido. En ella se definen los parámetros de la señal y su modo de definición.

4. Imágenes de ayuda: según el operador seleccionado, estas imágenes muestran las definiciones de las señales correspondientes de modo que el usuario pueda identificar a que parte de la definición matemática corresponde cada parámetro.

Inteface para la definición del soporte finito. Esta ventana se utiliza para definir el soporte finito de las señales que clasificadas como numéricas. Éste permite definir los valores del rango y previsualizar la señal par comprobar que se selecciona el intervalo deseado.

Existen dos réplicas de esta ventana para las dos señales del espacio, S1 y S2. Se almacenan en los ficheros: `continuousAperiodicf1DefinitionLimits` y `continuousAperiodicf2DefinitionLimits` de extensiones `.m` y `.fig`. En la figura 4.10 se muestra la pantalla con los controles numerados para facilitar su explicación.

1. Límites: el usuario introduce en estas cajas de texto los distintos valores para la definición del soporte finito.
2. Previsualización: al pulsar este botón el usuario puede visualizar la señal con los valores establecidos en las cajas de texto. Esta opción permite al usuario modificar el soporte finito sin tener que volver a definir la señal. Utiliza las funciones `preview_function_Callback` y `preview_distribution_Callback`.
3. Botón Set: este botón modifica definitivamente el valor del soporte finito de la señal. Vuelve a definir la señal para los nuevos valores, calcula de nuevo la norma y la energía y cierra la ventana. Ejecuta las rutinas `calculate_function_Callback` y `calculate_distribution_Callback`.

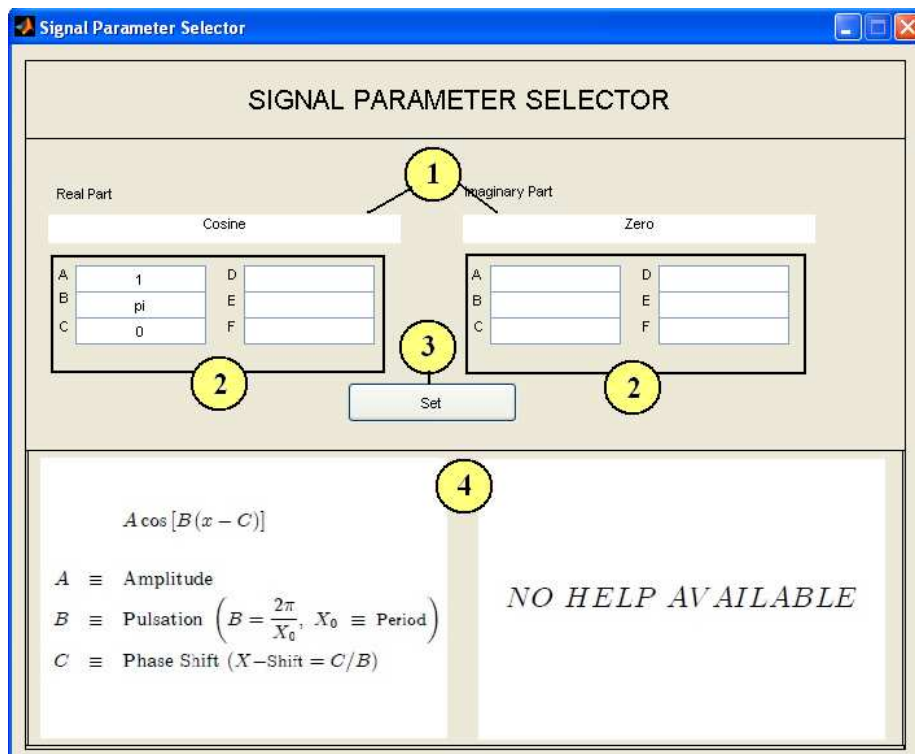


Figura 4.9: Ventana para la definición de los parámetros de la señal fija que caracteriza a un operador concreto. En esta ventana sólo se podrán modificar los parámetros de la señal y nunca su definición. Por ejemplo, para el operador modulador en amplitu la señal siempre será un coseno para la parte real y con parte imaginaria cero, sin embargo, se podrán modificar sus parámetros (amplitud, pulsación y fase).

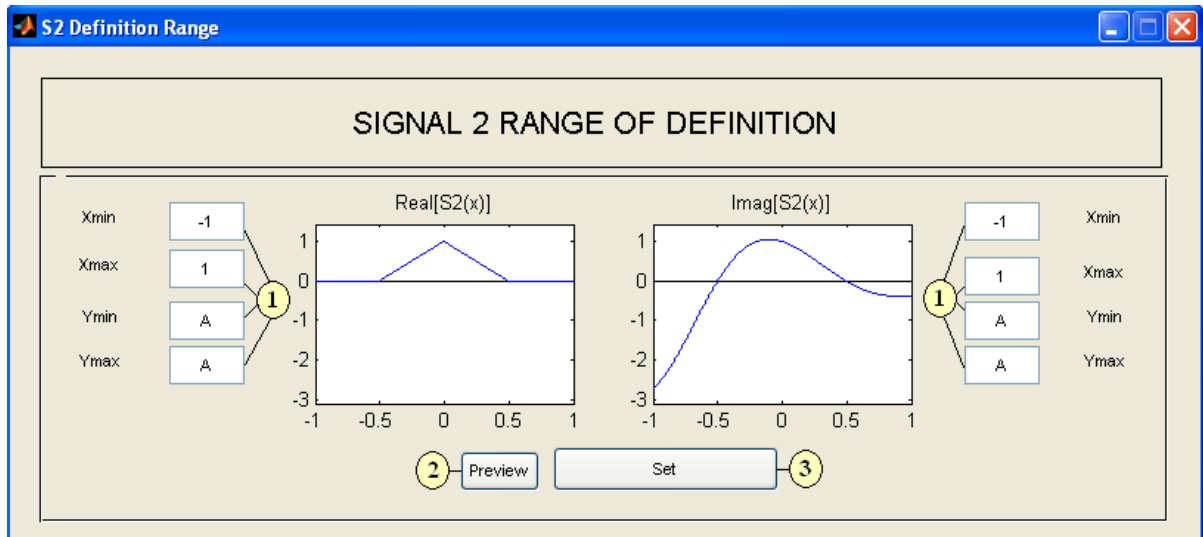


Figura 4.10: Ventana para definir el rango de una señal. Se utiliza en las señales numéricas para establecer el soporte finito en que están definidas. Fuera de dicho rango la señal toma el valor 0.

Interfaz de animación. Este interfaz muestra la animación de las operaciones de convolución y correlación. Permite observar el desarrollo de la operación y comprender como se realiza. Sólo se activa cuando la operación se lleva a cabo mediante un método numérico.

Está repetido para cada operación. Se almacena en los ficheros `animation_multiplication.fig`, `animation_convolution.fig` y `animation_correlation.fig`. Al ejecutarlo el programa utiliza las siguientes funciones: `animationMultiplication.m`, `multiplication_calcular_numerico.m` y `multiplication_launch_animation.m` para la operación de multiplicación, `animationConvolution.m`, `convolution_calcular_numerico.m` y `convolution_launch_animation.m` para la operación de convolución y las funciones: `animationCorrelation.m`, `correlation_calcular_numerico.m` y `correlation_launch_animation.m` para la correlación.

En la figura 4.11 se encuentra la ventana de animación con los controles etiquetados para su explicación.

1. Representación de la parte real de la señal de entrada y la parte real del conjunto de respuestas al impulso $h(x; x')$ para cada valor de x .
2. Representación de la parte imaginaria de la señal de entrada y la parte imaginaria del conjunto de respuestas al impulso $h(x; x')$ para cada valor de x .
3. Representación de la parte real de la señal de entrada multiplicada por la parte real del conjunto de respuestas al impulso $h(x; x')$ para cada valor de x .
4. Representación de la parte imaginaria de la señal de entrada multiplicada por la parte imaginaria del conjunto de respuestas al impulso $h(x; x')$ para cada valor de x .
5. Representación de la parte real de la señal de entrada multiplicada por el conjunto de respuestas al impulso $h(x; x')$ para todos los valores de x y x' . Sobre la curva se presenta en negro la función $h(t, x')$, donde t es el instante de x que se está simulando, dicha función es la misma que se representa en 3.
6. Representación de la parte imaginaria de la señal de entrada multiplicada por la parte imaginaria del conjunto de respuestas al impulso $h(x; x')$ para todos los valores de x y x' .

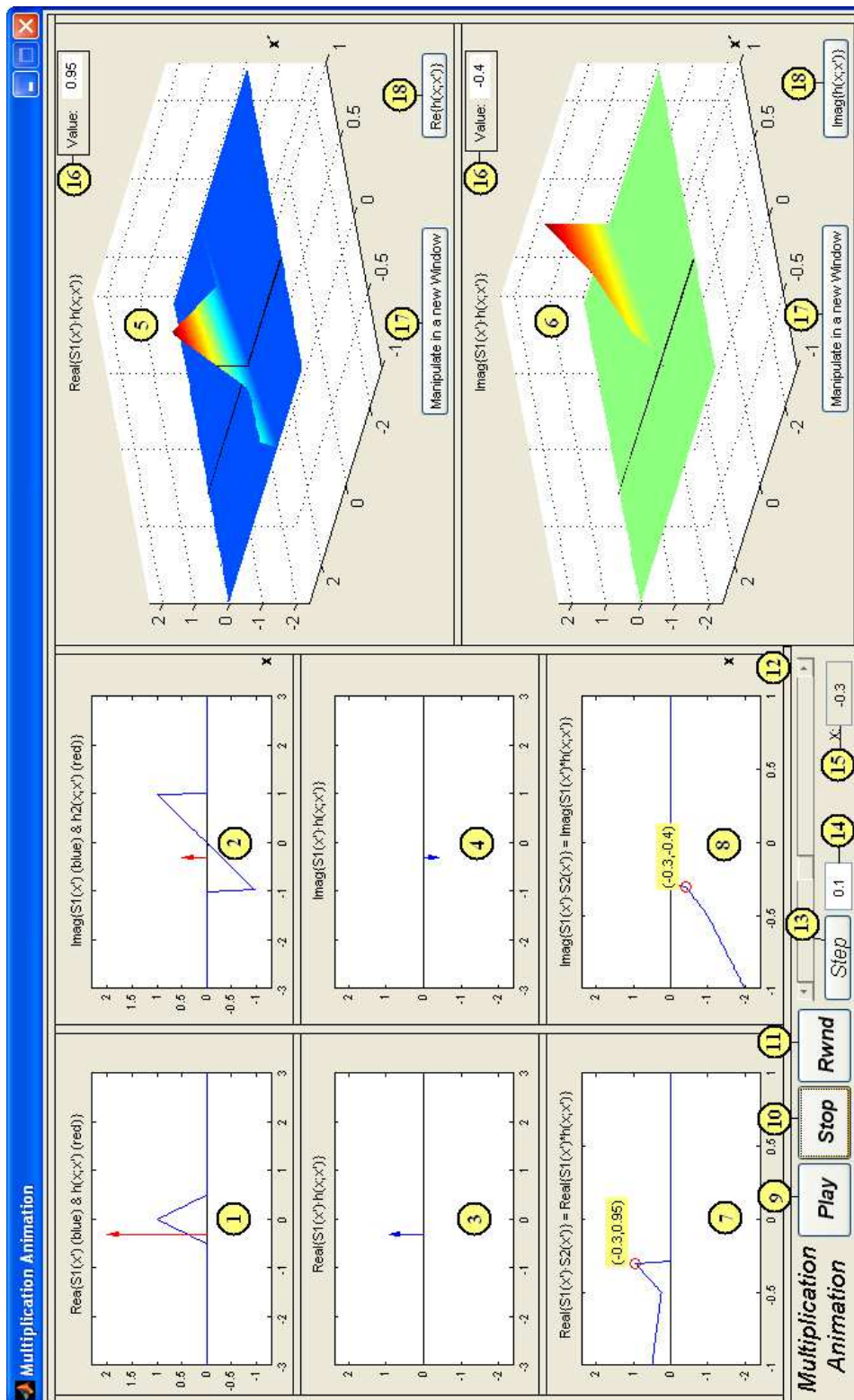


Figura 4.11: Pantalla de animación. Se puede acceder a ella para observar las distintas etapas para calcular una multiplicación, una convolución o una correlación. Sólo es posible utilizarla cuando la operación se realiza de forma numérica.

Sobre la curva se presenta en negro la función $h(t, x')$, donde t es el instante de x que se está simulando, dicha función es la misma que se representa en 4.

7. Representación de la parte real de la señal resultado, que no es más que la integral de la señal representada en 3.
8. Representación de la parte imaginaria de la señal resultado, que no es más que la integral de la señal representada en 4.
9. Botón Play: al pulsar este botón se inicia la animación. También se utiliza para reanudar la animación en caso de haberla parado.
10. Botón Stop: este botón para la animación.
11. Botón Rwnd: al pulsar éste botón se vuelve al punto inicial de la ejecución. La ejecución debe haber parado o finalizado antes para su correcto funcionamiento.
12. Barra de desplazamiento: al mover el valor de esta barra se accede a un punto concreto de la animación. El valor de las abscisas en ese instante se puede observar en la caja de texto, etiqueta 15. Como en el caso anterior la ejecución debe haber parado o finalizado para que funcione adecuadamente.
13. Botón Step: este botón se utiliza para moverse más rápido por la animación. Se puede realizar la simulación a saltos de la variable x del ancho deseado.
14. Caja de texto para introducir el ancho de salto / step: en esta caja de texto el usuario introduce el valor del salto que va a aplicar la próxima vez que pulse el botón Step.
15. Valor de las abscisas: en esta caja de texto se muestra el valor de abscisas que toma la animación en cada instante.
16. Valor de la señal resultado para cada instante: en estas cajas de texto se muestra el valor de la multiplicación, convolución o correlación en cada instante. Cuando el operador estudiado sea una convolución o correlación el conjunto de respuestas al impulso ya no estará compuesto por deltas y el texto Value se sustituirá por Area o Sum, según se esté en un espacio continuo o discreto.
17. Botones para visualizar la señal representada en una nueva ventana: al pulsar estos botones el usuario podrá visualizar la señal de tres dimensiones representada en la ventana de animación en un nuevo interfaz.
18. Botones para visualizar el conjunto de respuestas al impulso $h(x; x')$: al pulsar estos botones se visualizará en una nueva ventana el conjunto de respuestas al impulso $h(x; x')$, de este modo el usuario podrá comparar la multiplicación de la señal de entrada y el conjunto con el conjunto de señales.

Interfaz para la representación compleja. Esta ventana muestra la representación compleja de cada señal. Se accede a ella desde la ventana de visualización ejecutando la rutina `printComplex.m`. Está almacenada en el archivo `continuousAperiodicComplexRepresentation.fig`. En la figura 4.12 se muestra la ventana.

4.3.2. Interfaces del resto de módulos del programa.

Las interfaces del resto de módulos tienen una apariencia, funcionamiento y funcionalidades prácticamente idéntico al citado en los puntos anteriores por lo que no se explicarán. Debe tener en cuenta que en los espacios periódicos el interfaz principal presentará una caja de texto para que el usuario pueda seleccionar el período que desee y que en los espacios

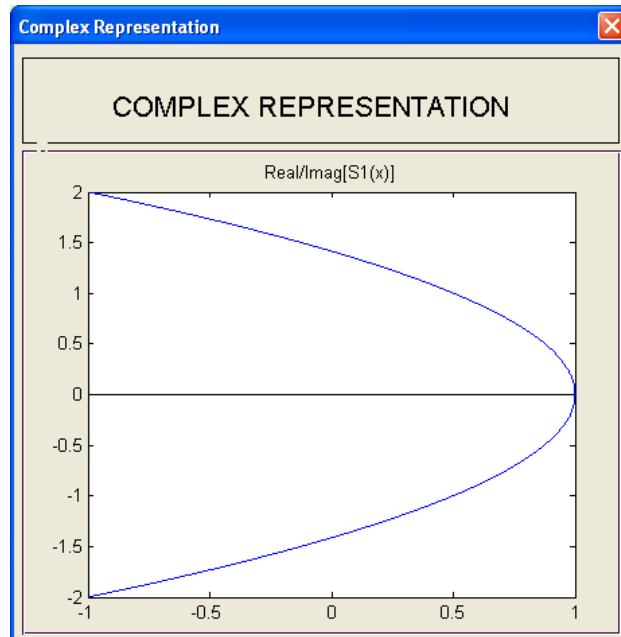


Figura 4.12: Ventana para la representación compleja. Se accede a ella desde el interface de visualización y permite visualizar la señal correspondiente de manera compleja.

discretos el concepto de distribución desaparece eliminándose por tanto los selectores de señales correspondientes.

Si desea conocer en detalle su funcionamiento y los archivos que las contienen puede consultar [4] - [6].

4.4. Rutinas principales del software.

En este apartado se pretenden exponer las funciones principales disponibles en el software, tanto su estructura general como el orden en que se llaman y el modo en que están relacionadas para proveer al usuario de cada funcionalidad. De este modo se podrían estructurar las rutinas, según su funcionalidad, en los siguientes tipos:

- Funciones para la definición de las señales de entrada al sistema y caracterización del operador.
- Rutinas para la obtención de las señales resultado y los cálculos numéricos asociadas a ella, cálculo de la norma, energía, distancia y producto escalar.
- Funciones para desplegar la visualización principal.
- Rutinas para la visualización de la caracterización del sistema en el dominio real.
- Rutinas para la visualización de la caracterización del sistema en el dominio espectral.

Aunque para el desarrollo de la presente versión del software se han modificado y creado rutinas de todos los tipos, la estructura y funcionamiento es similar a la presentada en [4] y, por lo tanto no se explicará en profundidad. Tan solo es preciso exponer las rutinas del último tipo, aquellas destinadas a representar el conjunto de funciones núcleo que caracterizan un operador en el dominio espectral, por ser una de las ampliaciones principales desarrolladas durante este proyecto. En la fig.4.13 se presenta un organigrama en el que se ven las principales funciones involucradas en la funcionalidad del análisis espectral generalizado y el orden en que son llamadas. A continuación se explican, brevemente, cada una de ellas.

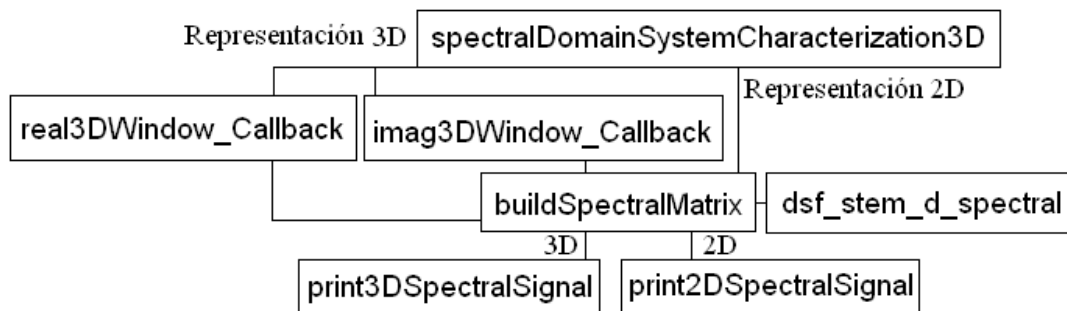


Figura 4.13: Organigrama que muestra las rutinas utilizadas para la representación del conjunto de funciones núcleo que representan a un operador lineal en el dominio transformado, así como el orden en que son invocadas.

spectralDomainSystemCharacterization3D. Es la función principal en la que se almacena toda la información relacionada con el interfaz en el que se representa el conjunto de funciones de transferencia. En él están todas las funciones a las que se llama cuando se pulsa uno de los botones gráficos y desde ella se llama al resto de rutinas necesarias para mostrar las señales. En este fichero, además de las funciones asociadas a cada control gráfico, existe un código de inicialización para mostrar las señales necesarias en cuanto la ventana sea desplegada.

real3DWindow_Callback. Esta rutina es la encargada de crear la ventana en la que se representará la parte real o módulo del conjunto de respuestas al impulso. Para comenzar comprueba los límites de visualización y el modo de representación. Una vez realizado esto, inicializa los vectores de representación e invoca a la función **buildSpectralMatrix** para obtener la matriz a dibujar. Cuando dispone de esta información crea la ventana de representación, si no está ya creada, y pinta la parte real o módulo del conjunto de respuestas al impulso.

imag3DWindow_Callback. Esta rutina es muy similar a **real3DWindow_Callback** con la diferencia de que crea la ventana en la que se representará la parte imaginaria o la fase del conjunto de respuestas al impulso. Para comenzar comprueba los límites de visualización y el modo de representación. Una vez realizado esto, inicializa los vectores de representación e invoca a la función **buildSpectralMatrix** para obtener la matriz a dibujar. Cuando dispone de esta información crea la ventana de representación, si no está ya creada, y pinta la parte real o módulo del conjunto de respuestas al impulso.

buildSpectralMatrix. Esta función es la encargada de definir la matriz de valores que represente el conjunto de funciones núcleo que caracterizan el operador seleccionado en el dominio espectral. Esta función es, en realidad, una base de datos que contiene la información asociada a cada operador. Como algunos operadores permiten que el usuario defina la señal que les caracteriza, por ejemplo el operador multiplicación o el convolución, puede necesitar llamar a la subrutina **dsf_stem_d_spectral** para calcular de forma numérica el conjunto de funciones de transferencia del operador.

print3DSpectralSignal. Esta rutina es la encargada de representar el conjunto de funciones de transferencia, tanto la parte real o módulo como la parte imaginaria o fase. Para realizarlo primero determinará si la señal a representar contiene deltas, como es el caso de los operadores lineales, o no y una vez hecho esto dibujará la superficie. Para finalizar, comprobará el estado de los distintos controles gráficos de modo que ajuste la representación a las exigencias del usuario.

print2DSpectralSignal. Esta rutina es muy similar a la explicada anteriormente, `print3DSpectralSignal`, y es la encargada de representar los cortes de la curva representada en las gráficas en tres dimensiones. Como en el caso anterior, la misma rutina se utiliza para dibujar ambos cortes tanto en parte real/imaginaria como en módulo/fase. Para realizarlo primero determinará si la señal a representar contiene deltas o no y una vez hecho esto dibujará la curva en los límites de representación y el modo indicado por el usuario en los controles de la pantalla. En el caso de los operadores lineales e invariantes, la segunda representación en dos dimensiones se utilizará para representar la función de transferencia del operador, que será representado por esta misma rutina.

dsf_stem_d_spectral. Este archivo contiene un método para el cálculo, mediante un método numérico (`dsf_c_numerical`) o invocando a una base de datos (`dsf_c_theoretical`), de la transformada de una señal. El resultado se utilizará después por la rutina `buildSpectralMatrix` para el cálculo de las funciones de transferencia del operador.

Aunque para este ejemplo se ha tomado el espacio de señales continuas aperiódicas el funcionamiento de estas rutinas es igual en el resto de espacios, teniendo en cuenta que en los espacios periódicos se tendrá en cuenta el periodo y que en los espacios discretos sólo se definen funciones y no distribuciones.

4.5. Métodos numéricos.

A continuación se presentan de forma resumida los distintos métodos numéricos disponibles en el software. Para obtener información detallada de estas rutinas se puede consultar la bibliografía [4] - [6]:

- Método numérico para el cálculo de la distancia, las operaciones básicas con dos señales (suma, resta, multiplicación y división) y los operadores suma, resta, multiplicación y modulador en amplitud: `conv_n`.
- Método numérico para el operador escalado: `scaling_operator_Callback`.
- Método numérico para el operador identidad: `identity_operator_Callback`.
- Método numérico para el operador desplazamiento: `shifting_operator_Callback`.
- Método numérico para el operador de convolución: `conv_n`.
- Método numérico para el operador correlación: `corr_n`.
- Método numérico para el operador transformada de Fourier y su inversa: `dsf_c_numerical` e `idsf_c_numerical`.
- Método numérico para el operador transformada de Hilbert y su inversa: `Hilbert_c_theoretical` e `idsf_c_numerical`.
- Método numérico para el cálculo de la norma, la energía y el producto escalar.

4.6. Bases de datos generadas.

Como ya se ha comentado al inicio de este capítulo, muchas de las bases de datos del software no se han modificado y, por lo tanto, no se explicarán en este documento. Sin embargo, con el objetivo de que el lector pueda conocer cuales existen se listan a continuación:

- Base de datos para la resolución del operador convolución: `conv_t_ri`.

- Base de datos para la resolución del operador transformada de Fourier y transformada inversa de Fourier: `dsf_c_theoretical`.
- Base de datos para la resolución del operador transformada de Hilbert y transformada inversa de Hilbert: `Hilbert_c_theoretical`.
- Base de datos para la resolución del operador transformada de Bessel y transformada inversa de Bessel: `Bessel_c_theoretical`.
- Bases de datos para la comprobación de las señales definidas por el usuario: `function_comprobatation` y `distribution_comprobatation`.
- Bases de datos para la resolución de la norma de las señales analíticas: `calculaNormaAnalitica`.
- Base de datos para generar las ayudas de señales en el interfaz principal: `setHelpName`.
- Bases de datos para la definición de señales: `determine_d_draw_conv_c_v`, `determine_d_draw_dsf_c_v`, `determine_f_draw_conv_c_v` y `determine_f_draw_dsf_c_v`.
- Base de datos para la representación del conjunto de respuestas al impulso de los distintos operadores: `buildMatrix`.
- Base de datos para la comprobación de las señales definidas por el usuario cuando éste sólo puede modificar sus parámetros: `parameter_comprobatation`.
- Base de datos para actualizar los controles del interfaz principal cuando se selecciona un operador: `setWindowValues`.

Para conocer en detalle su funcionamiento puede consultar [4] - [6].

Adicionalmente, se ha creado una nueva base de datos necesaria para implementar el análisis espectral generalizado.

4.6.1. Base de datos para la representación del conjunto de funciones núcleo que caracterizan a los distintos operadores en el dominio transformado: `buildSpectralMatrix`.

Esta base de datos contiene las distintas definiciones del conjunto de funciones de transferencia para los diferentes operadores del software. La base de datos dará valores a la matriz de representación en función del operador seleccionado. Por ejemplo, para el caso de los operadores lineales e invariantes dará valor $2\pi H(\xi)$ a los puntos de la matriz que se vayan a representar sobre la línea de referencia $\xi = \xi'$.

Esta base de datos también se utiliza para representar los distintos cortes del conjunto de funciones de transferencia.

Como parámetros recibe los dos vectores de representación, ξ y ξ' cuando se trabaje en el espacio de señales continuas aperiódicas, un parámetro denominado `origen`, que identificará si se debe definir la parte real o la parte imaginaria de la señal y un parámetro `ref` para conocer si se va a asignar valores a una matriz o a un vector que varíe a lo largo de ξ o de ξ' .

5. MANUAL DE USUARIO

5.1. Presentación y utilidades de programa Signals & Systems Theory v5.0.

Signals & Systems Theory v5.0 es una herramienta software concebida para el estudio y tratamiento de señales y operadores como un sistema. De este modo, el programa ha sido desarrollado con el objetivo de permitir al usuario la experimentación con las distintas señales cumpliendo éstas diversas funcionalidades.

Desde dicho punto de vista, el programa implementa las siguientes funcionalidades:

- Estudio de las transformaciones sufridas por una señal cuando ésta se ve afectada por distintos operadores y familias de operadores: el operador suma, resta, multiplicación, escalado, convolución, correlación, identidad, desplazamiento y modulador en amplitud.
- Implementación y cálculo de distintas transformadas: transformada de Fourier, desarrollo en serie de Fourier, transformada de Hilbert y transformada de Bessel. Todas ellas en su forma directa e inversa.

Además de estas operaciones como sistema, este programa también se utiliza para:

- Visualizar señales.
- Realizar operaciones básicas con dos señales: suma, resta, multiplicación y división.
- Realizar el análisis espectral del sistema gráficamente, bajo las distintas transformadas disponibles.
- Realizar otros cálculos: norma, energía, distancia entre dos señales y el producto escalar de dos señales.

Complementando dichas utilidades, dispone de cinco espacios de señales distintos con los que puede actuar: espacio de señales aperiódicas de variable continua, espacio de señales aperiódicas de variable continua definidas para valores positivos, espacio de señales periódicas de variable continua, espacio de señales aperiódicas de variable discreta y espacio de señales periódicas de variable discreta. Todos ellos están basados en la definición de álgebra que haya escogido.

Como principal novedad de esta versión del software destaca el **análisis espectral de los sistemas lineales no invariantes**, incluyendo la representación del conjunto de funciones de transferencia que caracterizan el operador en el dominio transformado.

5.2. Descripción de las interfaces.

A continuación se explica el modo de utilización del programa y sus distintas funcionalidades a través de las ventanas que puede utilizar.

El programa se estructura en tres interfaces principales a los que acompaña una serie de ventanas secundarias. Las ventanas principales y algunas de las secundarias sufrirán modificaciones según el espacio de partida seleccionado por el usuario.

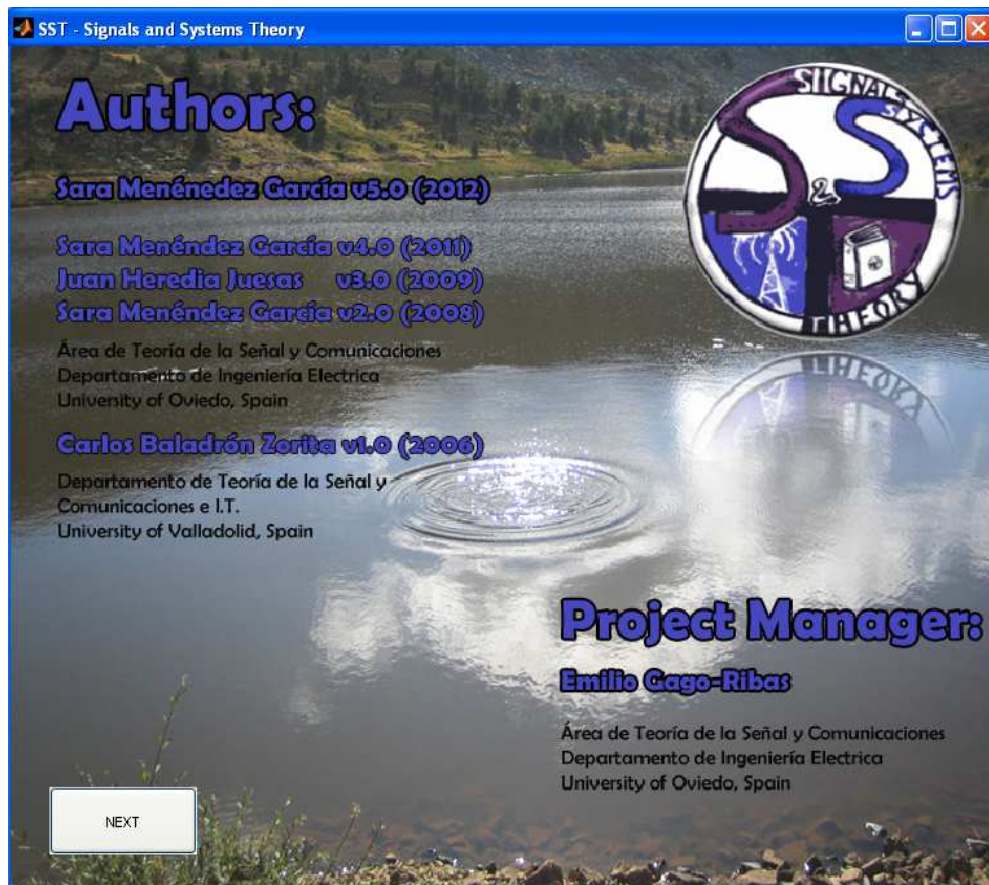


Figura 5.1: Pantalla Inicial. Es la primera ventana que aparece cuando se ejecuta el programa. Contiene información de la versión y los autores. Para continuar la ejecución del programa debe pulsar el botón "Next".

5.2.1. Ventana de inicio.

Esta ventana será la primera que vea cuando ejecute el programa, véase la Fig. 5.1. En ella aparece el nombre del programa y su versión. También se enumeran sus autores. Para avanzar basta con hacer clic sobre el botón Next de la pantalla.

5.2.2. Menú principal.

Se trata de la ventana principal del software. Desde ella podrá acceder a todas las funcionalidades del software y será la que utilice para dar todo tipo de órdenes, excepto las órdenes sobre la visualización. Esta ventana, apoyada por las demás, será la encargada de realizar prácticamente todas las funcionalidades del software.

Su estructura pretende ilustrar el esquema general la teoría de señales y sistemas sobre la que se fundamenta el programa: se muestran los espacios de entrada y salida al sistema y el espacio de operadores. Pese a que la estructura de la ventana represente siempre dicho esquema, su apariencia se verá modificada según el momento de ejecución en que nos encontremos y del espacio de partida que se haya seleccionado. Conforme a lo dicho, la pantalla que se muestra inicialmente, tras pinchar el botón Next del interfaz anterior, es la mostrada en la figura 5.2.

Una vez haya accedido a la ventana principal simplemente tendrá que seleccionar un espacio de partida para poder trabajar con él. Como se observa en la imagen el propio programa muestra un mensaje de ayuda para que el usuario pueda avanzar. Para seleccionar el espacio

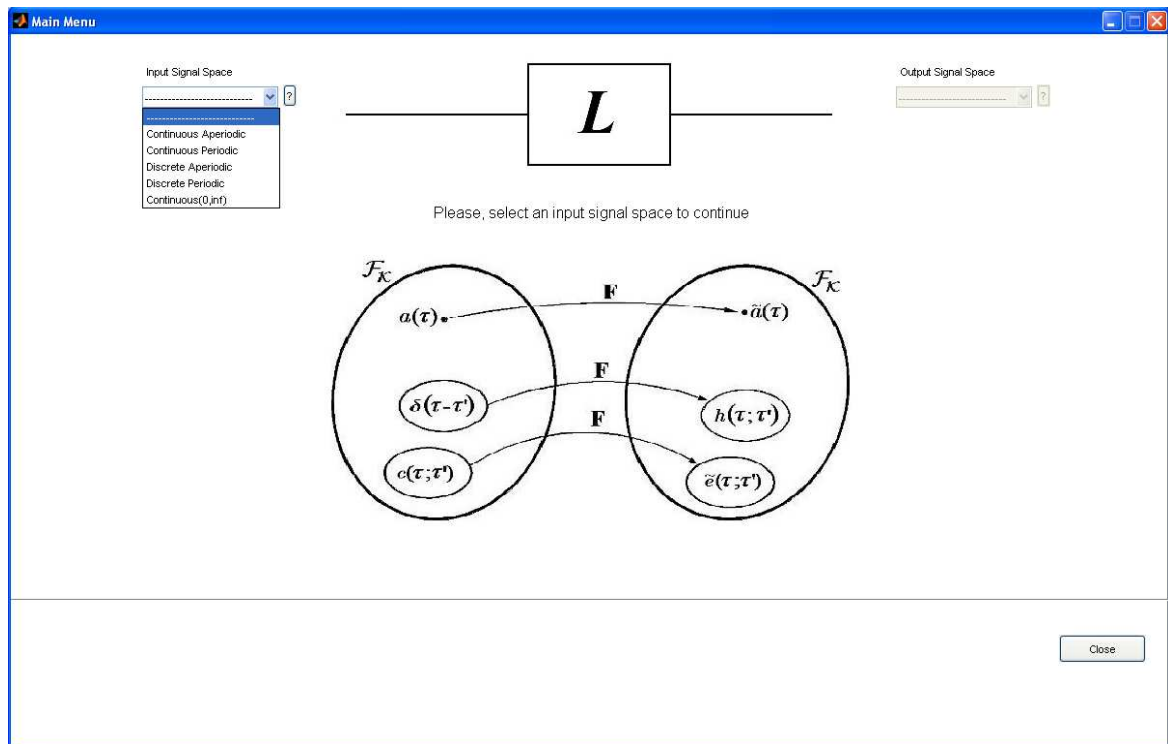


Figura 5.2: Menú principal en estado inicial. El menú principal sólo se mostrará así al inicio de su ejecución. Lo primero que debe hacer es seleccionar un espacio de entrada en el menú desplegable para poder avanzar.

de partida con el que desee trabajar utilice el menú desplegable. Si tiene alguna duda acerca de la definición de cualquiera de los espacios disponibles puede acceder a la ayuda pulsando el botón correspondiente. Este botón es fácilmente reconocible: está junto al menú desplegable y lleva escrita una interrogación.

Si lo que desea es terminar la ejecución del programa pulse en el botón Close y la ventana se cerrará automáticamente.

Una vez haya seleccionado un espacio de partida observará que el aspecto de la pantalla ha cambiado. Si selecciona, por ejemplo, el espacio aperiódico de variable continua su ventana tendrá un aspecto como el de la figura 5.3. En ella se pueden apreciar cuatro partes diferenciadas, etiquetadas con números para facilitar su explicación. La primera es la parte que corresponde al espacio de entrada, su definición y sus señales. La segunda parte contiene información del operador, en él pueden intervenir otras señales que serán de la naturaleza marcada por el espacio de entrada. La tercera parte corresponde al espacio de salida del sistema y su definición. La cuarta y última parte es ajena al sistema, se utiliza para otras funcionalidades como el análisis espectral del sistema o resetear o cerrar el programa.

Como se observa, a pesar de que hayan aparecido nuevos controles en la ventana y, gráficamente, se haya visto modificada, en ella sigue apareciendo el esquema general de la teoría de señales y sistemas y la estructura se mantiene.

A continuación se explicarán en detalle todos los botones y funcionalidades de cada una de las partes del menú.

Primera parte: entrada al sistema. Como su propio nombre indica es la parte de la ventana correspondiente a la entrada del sistema: en ella definirá todos los aspectos relativos a

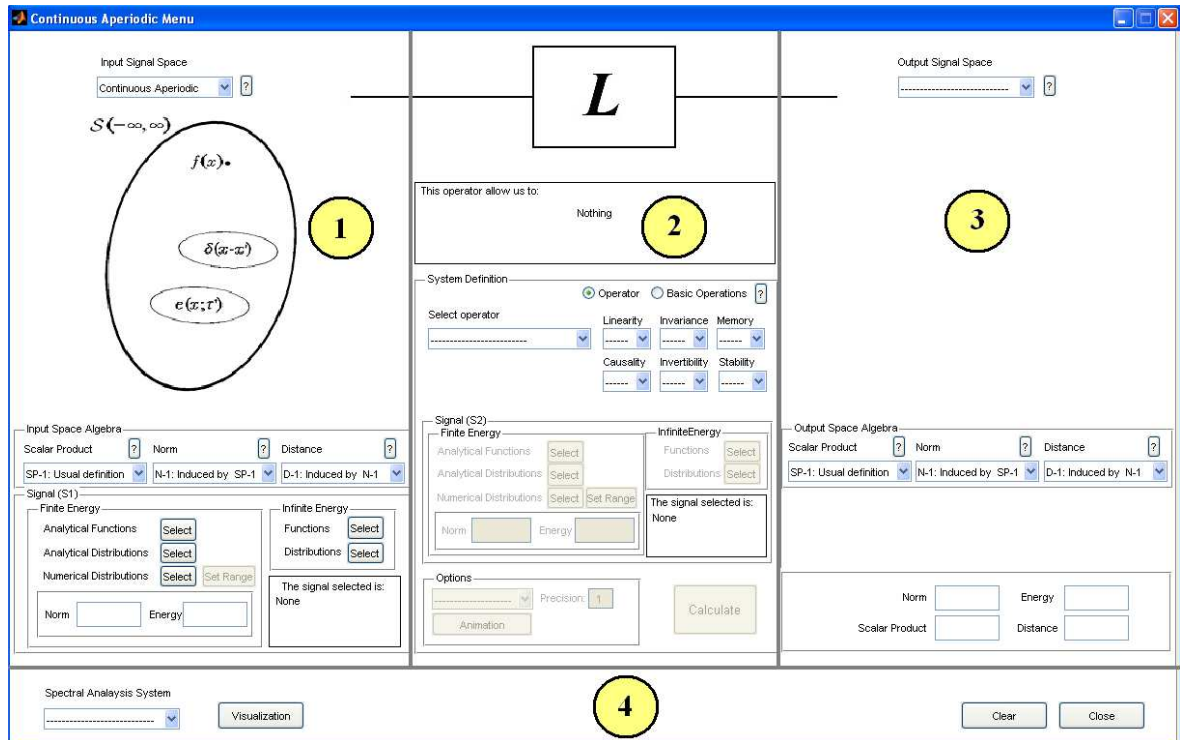


Figura 5.3: Menú principal cuando el espacio de entrada seleccionado es el continuo aperiódico. En esta ventana podrá definir señales, operar con ellas y obtener algunos resultados numéricos. Además le permitirá acceder al resto de ventanas como son las de visualización o animación.

la señal de entrada, aquella que sufrirá las transformaciones impuestas por el sistema.

En la figura 5.4 se presenta esta parte del interfaz. Como se observa en la imagen contiene los siguientes elementos:

1. Espacio de entrada: se trata de la primera selección necesaria para poder definir una señal de un tipo u otro. En este menú desplegable puede escoger el espacio de entrada en el que desee trabajar de los que haya disponibles. En la versión actual del software puede seleccionar cinco espacios diferentes: espacio de señales complejas de variable continua definidas para todos los valores (Continuous Aperiodic), espacio de señales complejas de variable continua con período X_0 (Continuous Periodic), espacio de señales complejas de variable discreta definidas para todos los valores (Discrete Aperiodic), espacio de señales complejas de variable continua con período N_0 (Discrete Periodic) y , espacio de señales complejas de variable continua definidas para valores positivos (Continuous Aperiodic(0,+inf)).
2. Ayuda del espacio de entrada: haciendo clic sobre él aparecerá una nueva ventana de ayuda con la definición de los espacios disponibles.
3. Esquema algebraico del espacio de entrada seleccionado: a modo ilustrativo se muestra un esquema algebraico del espacio seleccionado. Como es lógico dicho esquema cambiará según cual sea el espacio de entrada seleccionado por el usuario.
4. Ayudas del álgebra del espacio: puede pinchar sobre ellas para conocer la definición de cada uno de los parámetros algebraicos disponibles según la definición escogida para cada uno de ellos.
5. Definición de los parámetros algebraicos: puede seleccionar la definición que desee para cada uno de dichos parámetros. En la versión actual del software sólo dispone de una

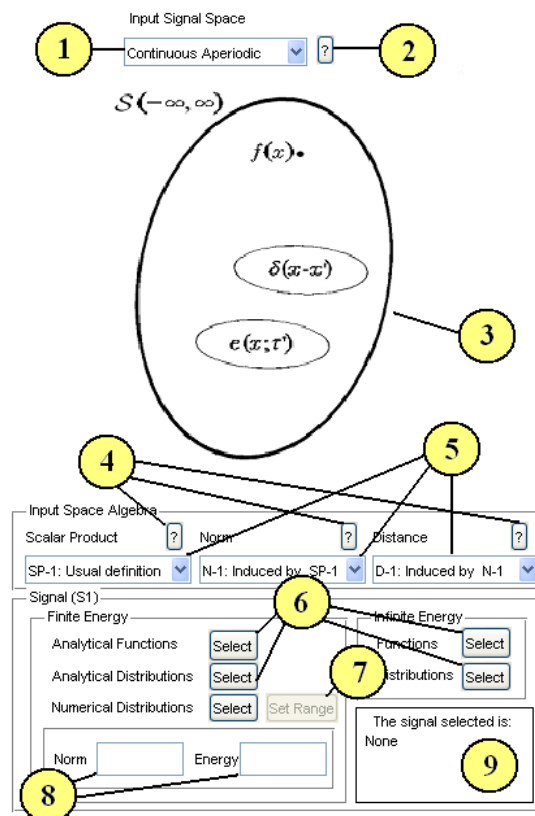


Figura 5.4: Entrada al sistema del menú principal. Esta parte del menú será utilizada para definir el álgebra del espacio de entrada y la señal de a estudiar (la que entra al sistema).

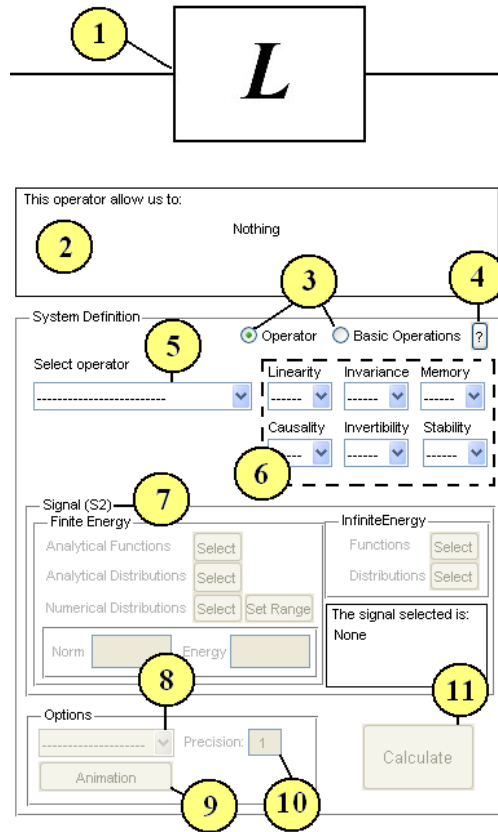


Figura 5.5: Espacio de operadores. En esta parte del menú principal se selecciona el operador que definirá el sistema. Podrá elegir uno de los operadores de los disponibles en la lista desplegable: suma, identidad, convolución, transformada de Fourier, etc. Una vez seleccionado el operador, según cual sea, podrá terminar de definirlo (definiendo la respuesta al impulso del sistema) y ajustar algunos parámetros de la simulación. A modo ilustrativo el programa mostrará al usuario las propiedades del operador, así como una breve descripción y un esquema que lo represente. Adicionalmente podrá estudiar algunas operaciones básicas: suma, resta, multiplicación y división, sin que éstas sean definidas como un operador.

definición para cada uno de ellos.

6. Selectores de señales: cuando desee definir una señal de entrada al sistema debe pinchar estos botones, accederá a cada uno dependiendo de la naturaleza que desee dar a la señal de entrada. Actualmente puede seleccionar señales analíticas (definidas para todo el espacio) y señales numéricas (tomarán valores no nulos en el rango que usted defina).
7. Selector del rango de definición: este botón sólo se activará en caso de que haya definido una señal de tipo numérica. En dicho caso podrá definir y modificar el rango de abscisas y ordenadas para el que desee que la función tome valores significativos pulsando este botón.
8. Cajas de resultados: en estas cajas de texto se mostrarán los valores de norma y energía de la señal que haya definido. Siempre y cuando no se trate de señales de energía infinita, en cuyo caso estas cajas se mostrarán vacías.
9. Ayuda de la señal seleccionada: en esta caja de texto se mostrará el nombre de la señal que haya definido como entrada al sistema. De este modo se evita que olvide la señal que está utilizando durante el transcurso de la ejecución del programa.

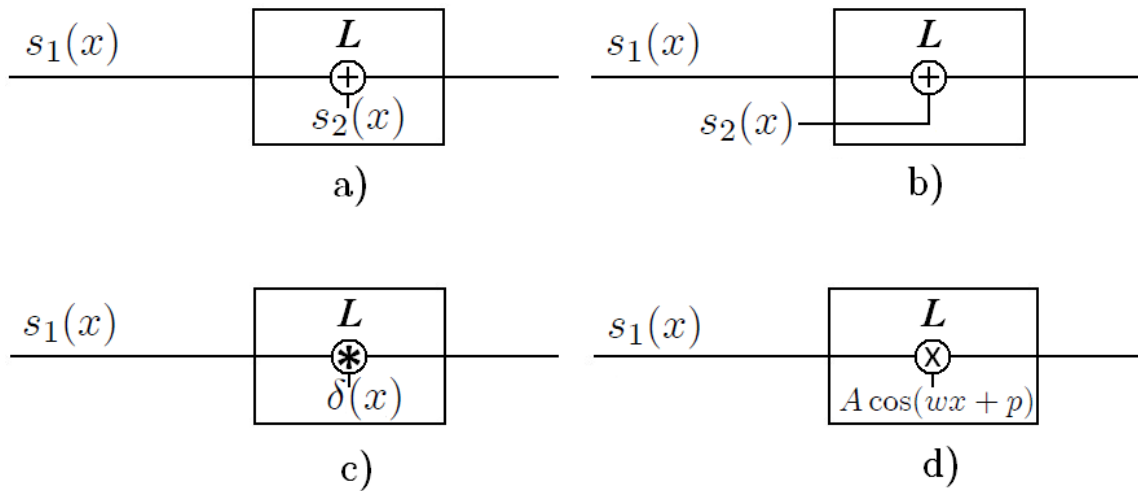


Figura 5.6: Esquema del operador. En esta imagen se muestran algunos ejemplos de los distintos esquemas del operador que se muestran, según el sistema que defina. El esquema se muestra tanto cuando usted selecciona un operador como cuando lo que selecciona es una de las operaciones básicas. Los ejemplos mostrados son: a) Operador suma: como se muestra este esquema corresponde a una familia de operadores y, por tanto, necesita que se defina una señal para definir el operador: $s_2(x)$. Cada vez que seleccione una $s_2(x)$ diferente estará definiendo un nuevo operador y por tanto un nuevo sistema. b) Operación básica suma: como se muestra en la imagen en este caso lo que usted definirá serán dos señales de entrada que se sumarán para dar una señal resultado. c) Operador identidad: en este caso no será necesario que defina una segunda señal puesto que lo que estamos seleccionando es un operador concreto. A modo ilustrativo el esquema muestra que la respuesta al impulso de este operador es $h(x; x') = \delta(x)$.

Segunda parte: interior del sistema. Esta parte de la ventana es la que utilizará para controlar el comportamiento interno del sistema. Podrá seleccionar sus controles de forma que cumplan todas y cada una de las funcionalidades del programa, excepto la visualización y el análisis espectral de sistema que veremos más adelante.

En la figura 5.5 se ilustra la parte de la ventana que contiene los controles para configurar el interior del sistema. A continuación se explica cada uno de ellos:

1. Esquema del operador. La famosa caja negra estudiada en la teoría de señales y sistemas irá cambiando en función del operador u operación seleccionado, para mostrar de forma clara e intuitiva el funcionamiento interno de éste. En la figura 5.6 se pueden ver algunos ejemplos.
2. Descripción del operador u operación seleccionado. En este cuadro de la ventana se mostrará un texto explicativo de cada operador u operación seleccionada.
3. Selector del tipo de análisis. Esta versión del software le permitirá realizar dos tipos de análisis: analizar un sistema de comunicaciones o analizar una operación básica. En el primer caso usted seleccionará un operador con el que definirá el sistema de comunicaciones que atravesará la señal de entrada. En el segundo caso analizará las operaciones básicas con dos señales de entrada (suma, resta, multiplicación y división). Para seleccionar un tipo de análisis u otro simplemente pulse el botón correspondiente.
4. Ayuda del tipo de análisis. Si pulsa este botón se abrirá una ventana en la que se le explicará las dos opciones de análisis que tiene y cómo puede seleccionarlas.
5. Selector del operador u operación. Con este menú desplegable podrá seleccionar distintos

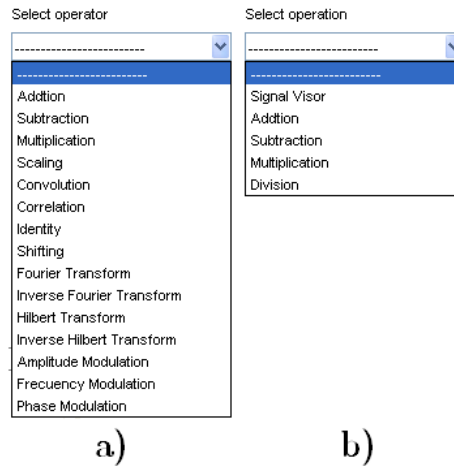


Figura 5.7: Selector de operador u operación. En a) se muestra la lista de operadores disponibles en el espacio de señales continuas aperiódicas. En b) se muestra la lista de operaciones disponible en el espacio de señales continuas aperiódicas. Como se puede, observar además de las operaciones básicas, se incluye en la lista una funcionalidad para visualizar señales.

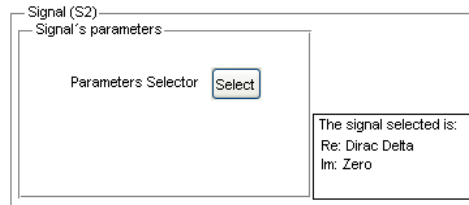


Figura 5.8: Controles de definición de la segunda señal definida cuando se trabaja con un operador concreto. Esta parte de la ventana cambiará y mostrará sólo un botón con el que accederá a la ventana del selector de parámetros. En dicha ventana podrá modificar la amplitud, desplazamiento, pulsación, etc. de la señal que define al operador pero no podrá seleccionar una señal. El selector de señales sólo se muestra cuando se selecciona un operador concreto y no una familia de operadores. Algunos ejemplos son el modulador en amplitud (en este caso la segunda señal definida siempre será un coseno de parámetros modificables) o el operador desplazamiento (la segunda señal y respuesta al impulso siempre será una delta de amplitud 1 y con un desplazamiento seleccionable).

operadores y definirá así el comportamiento del sistema. Según el espacio de partida y el tipo de análisis seleccionado se mostrarán distintas opciones, expuestas en la figura 5.7. Según la opción escogida, el resto de controles mostrados en esta parte de la interfaz cambiarán.

6. Propiedades de los operadores. Este conjunto de controles sólo se muestra cuando este definiendo un operador. Para el operador que usted defina muestran las distintas propiedades que posee: linealidad, invarianza, memoria, causalidad, invertibilidad y estabilidad.
7. Controles correspondientes a la segunda señal definida (S2). La explicación de estos controles es la misma que la de los controles 6, 7, 8 y 9 de la figura 5.4. Sólo en algunos casos, cuando usted seleccione operadores concretos y no familias de operadores, su apariencia cambiará y se mostrará como en la figura 5.8. En este caso, el único botón nuevo se utilizará para modificar los parámetros de la señal que define al operador. Es decir, usted podrá modificar su amplitud, desplazamiento, etc. pero no podrá decidir cual es la señal.

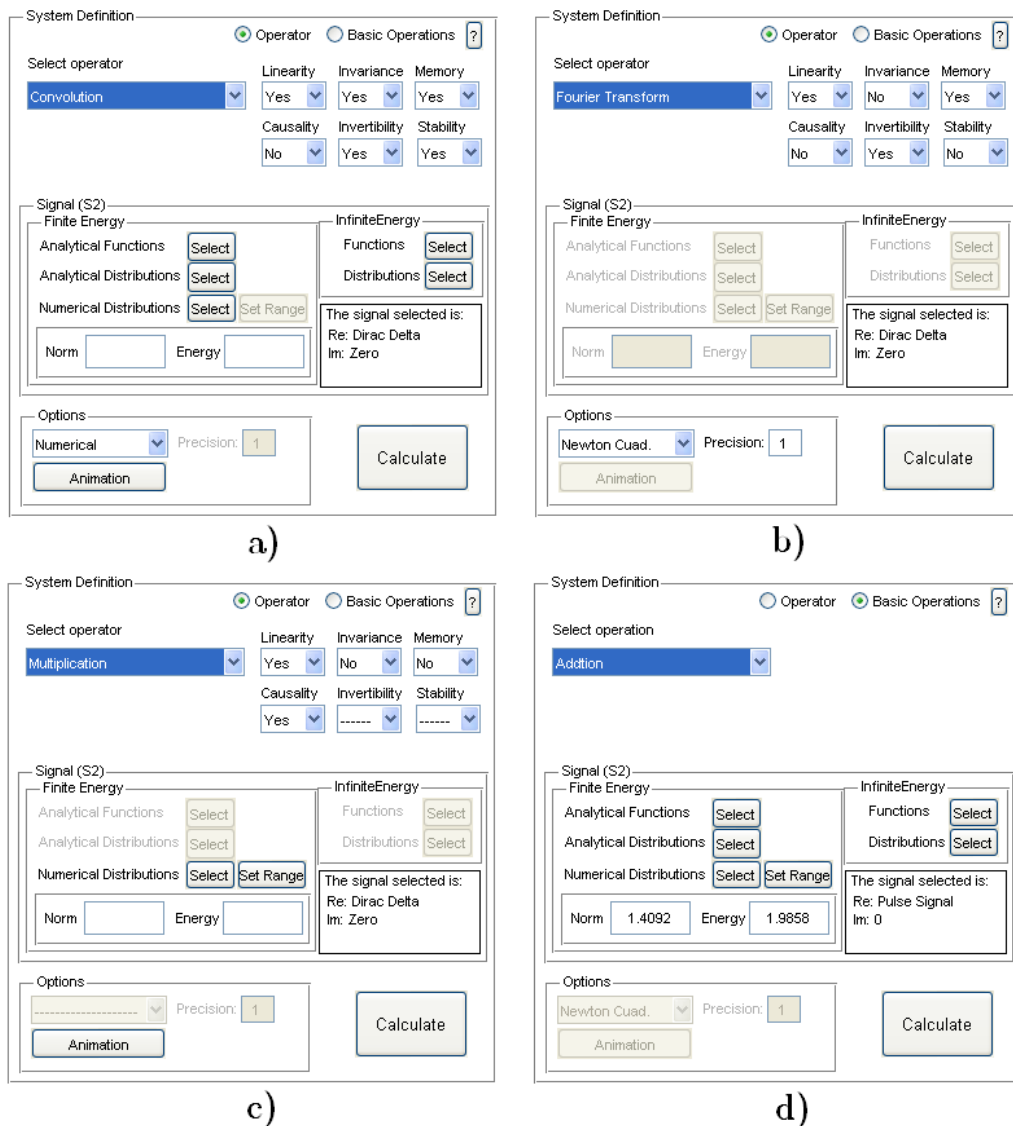


Figura 5.9: Definición del sistema estudiado. Como se observa, según el operador u operación que haya seleccionado, la apariencia de la ventana cambia activándose / desactivándose y mostrándose / ocultándose los controles que requiera cada operador u operación.

8. Opciones de cálculo. Este menú desplegable se puede manipular en algunos operadores para definir el modo en que se quieren realizar o el método numérico que se desea implementar para calcular la señal de salida.
9. Animación. Este botón sólo se activará cuando calcule multiplicaciones, convoluciones y correlaciones de forma numérica. Cuando lo pulse se abrirá una ventana mostrando la animación del desarrollo de la operación, véase la Fig. 5.25.
10. Precisión: esta caja de texto se activará cuando realice transformadas de Fourier o sus inversas. Al modificarlo, cambiará la precisión con que la máquina realiza estos cálculos.
11. Botón calcular. Una vez haya definido la señal de entrada y el operador adecuadamente, deberá pulsar este botón para obtener la señal de salida.

La Fig. 5.9, se muestran unos ejemplos de como se activarían o desactivarían y mostrarían u ocultarían los controles 6 al 10 según el operador que haya seleccionado.

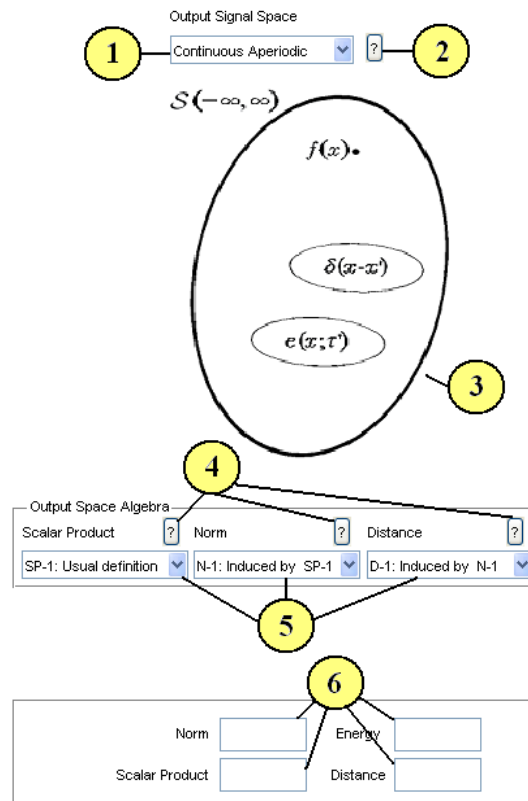


Figura 5.10: Salida del sistema. En esta parte del menú principal podrá observar los resultados obtenidos de la señal a su paso por el sistema: el espacio de salida y la norma y la energía de la señal resultado. Además podrá ver el resultado de las relaciones producto escalar y distancia de las dos señales de entrada o de la señal de entrada y señal que define al sistema, en aquellos casos en que la segunda señal sea definida por el usuario. En esta parte de la ventana, sólo podrá manipular las definiciones algebraicas del espacio de salida para que los cálculos se realicen en función de ellas.

Tercera parte: salida del sistema. Esta es la última parte de la interfaz que tiene relación con el sistema. En realidad, en esta parte de la ventana podrá modificar muy pocas cosas ya que se utilizará para mostrar los resultados obtenidos de la señal de entrada a su paso por el sistema.

Como se puede ver en la figura 5.10, la salida del sistema consta de los siguientes apartados:

1. Espacio de salida. En este menú visualizará el espacio al que pertenece la señal una vez ha sido transformada por el sistema. Evidentemente, no podrá seleccionar este espacio sino que lo observará como parte del resultado.
2. Ayuda del espacio de salida. Al igual que para el espacio de entrada, si pulsa este botón obtendrá la definición del espacio de salida.
3. Ayudas del álgebra del espacio. Pinchando sobre ellas conocerá la definición de cada uno de los parámetros algebraicos disponibles según la definición escogida para cada uno de ellos. En este caso se refieren al álgebra del espacio de salida.
4. Definición de los parámetros algebraicos. Como en el espacio de entrada podrá seleccionar la definición que desee para cada uno de dichos parámetros. Esta versión del software sólo dispone de una definición para cada uno de ellos. Al igual que ocurre en las ayudas, estas

definiciones se refieren al álgebra del espacio de salida.

5. Resultados numéricos. La pantalla dispone de cuatro tipos de resultados. En las dos superiores se mostrarán los valores de norma y energía de la señal resultado. Adicionalmente, en las dos inferiores, podrá ver el valor del producto escalar y de la distancia entre las dos señales de entrada (cuando trabaje con operaciones básicas) o entre la señal de entrada al sistema y la que usted haya escogido para definir el operador del sistema.

Cuarta parte: otras funcionalidades del software. La última parte de la ventana tiene se utiliza para acceder a las funcionalidades ajenas al sistema simulado.



Figura 5.11: Controles secundarios. Esta parte del menú es independiente al sistema. En ella podrá acceder a otras funcionalidades del software como la ventana de visualización, el análisis espectral del sistema o simplemente cerrar el programa.

Como se muestra en la figura 5.11 esta parte de la interfaz dispone de los siguientes controles:

1. Análisis espectral del sistema. El menú desplegable le mostrará las distintas opciones que existen para este análisis: análisis bajo la transformada de Fourier, bajo la transformada de Hilbert, bajo la transformada de Bessel... Las opciones variarán en función del espacio de partida que haya seleccionado. Una vez seleccionado un tipo de análisis podrá estudiarlo en la ventana de visualización.
2. Visualización. Pulsando este botón abrirá el interfaz de visualización, véase la Fig. 5.12, donde podrá visualizar las señales involucradas en el sistema.
3. Botón Clear. Utilice este botón para resetear la pantalla volviendo a ponerla en su estado inicial.
4. Botón Close. Puse este botón para terminar la ejecución del programa. Además de cerrar el menú principal, cerrará todas las ventanas del software que estén abiertas.

A lo largo del presente apartado se ha explicado el funcionamiento general de esta ventana sobre el ejemplo de señales definidas para cualquier valor de variable continua. Como es evidente, cuando en la pantalla se seleccione un espacio periódico aparecerá un nuevo control para introducir el valor del periodo con el que se quiera trabajar. Además, en el caso de espacios discretos desaparecerán los selectores de señales que se refieran a distribuciones.

5.2.3. Interface de visualización.

La interfaz de visualización es la más importante después del menú principal. Accederá a ella para visualizar todas las señales que componen el sistema y para visualizar el análisis espectral del sistema si lo desea.

Según el tipo de análisis que haya seleccionado la ventana de visualización tendrá un aspecto diferente. De este modo cuando trabaje con las operaciones básicas accederá a la ventana mostrada en la figura 5.12 y cuando trabaje con operadores será la mostrada en la figura 5.13. Como puede observar la ventana siempre está dividida en seis partes, la única diferencia es que cuando trabaje con operadores, en lugar de visualizar la segunda señal definida, accederá a una nueva ventana para poder visualizar el conjunto de respuestas al impulso del sistema en

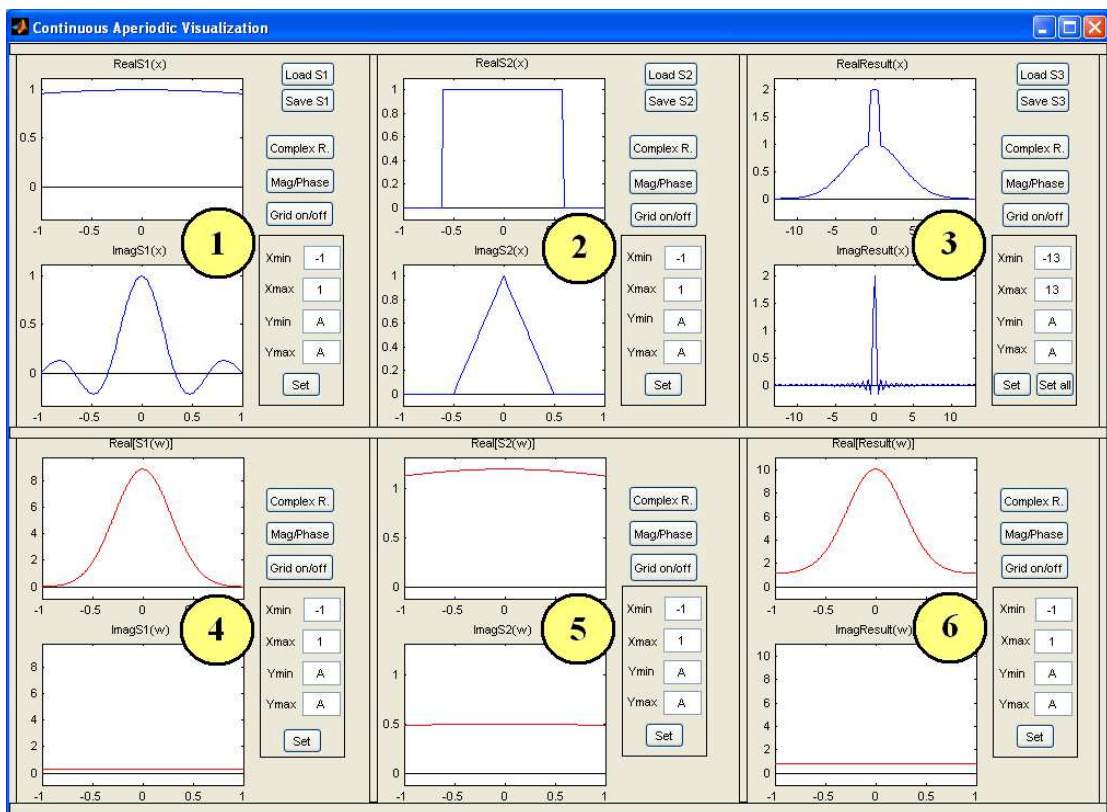


Figura 5.12: Ventana de visualización cuando analiza operaciones básicas. En ella podrá observar todas las señales representadas en modo real/imaginario o módulo/fase. Además podrá modificar los límites de visualización y acceder a la representación compleja. También aquí, podrá salvar señales en un fichero o cargarlas desde uno ya existente.

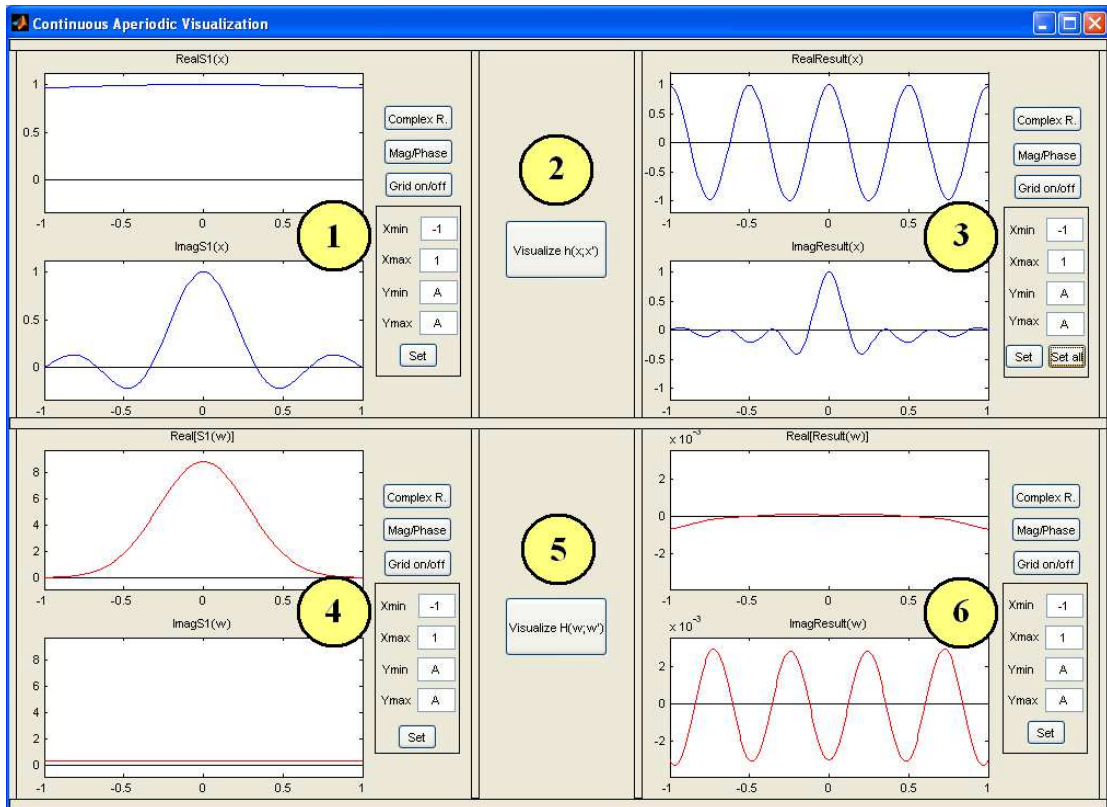


Figura 5.13: Ventana de visualización cuando analiza operadores. En ella podrá observar las señales de entrada y salida al sistema representadas en modo real/imaginario o módulo/fase. Además podrá modificar los límites de visualización y acceder a la representación compleja. Desde esta ventana podrá acceder a la visualización en tres dimensiones del conjunto de respuestas al impulso del sistema. También aquí, podrá salvar señales en un fichero o cargarlas desde uno ya existente.

estudio (hayan sido definidas por el usuario o estén ya definidas por el operador). Las tres partes superiores se refieren al sistema en el dominio real y las tres inferiores al sistema en el dominio transformad, serán las utilizadas para visualizar el análisis espectral del mismo. Las partes superiores en las que se representa señal son idénticas entre sí y las inferiores, en las que se representa señal, se les parecen mucho, la única diferencia es que desaparecen dos controles con respecto a las de arriba. Por ello sólo se explicarán los controles de una de las partes superiores, entendiendo que con ello comprenderá los de toda la ventana.

En la figura 5.14 , se ilustra la tercera parte superior. Sus controles se encuentran etiquetados para facilitar la explicación que sigue:

1. Primera gráfica: en ella se mostrará la primera parte de la señal (la parte real o el módulo).
2. Segunda gráfica: en ella se mostrará la segunda parte de la señal (parte imaginaria o fase).
3. Botón Load: con el podrá visualizar señales que haya almacenadas en un fichero de datos. Ver Secc. 5.3. Este botón aparecerá y desaparecerá dependiendo de la parte de la ventana y el análisis realizado.
4. Botón Save: al pulsarlo guardará la señal visualizada en las gráficas en un fichero de datos. Ver Secc. 5.3. Al igual que el botón anterior, éste se mostrará u ocultará en los distintos casos.

5. Representación compleja: al pulsar este botón abrirá una nueva ventana con la representación compleja de la señal.
6. Modo de representación: este botón le permite modificar el modo de visualización de la señal pulsando sobre él (de real/imaginario a módulo/fase y viceversa).
7. Botón Grid: si lo activa se mostrará una rejilla sobre las gráficas. Puede resultar muy útil a la hora de obtener resultados precisos de la gráfica.
8. Rango de representación: introduzca en estas cajas de texto el rango de representación de desee visualizar.
9. Botón Set: pulse este botón para visualizar la señal en el rango establecido por los controles anteriores.
10. Botón Set all: pulse este botón para visualizar todas las señales de la ventana en el rango establecido por los controles expuestos en 8. Este botón sólo está disponible para este apartado de la ventana. Es útil para obtener una visión conjunta de las señales como sistema.

Además de un valor numérico, en las cajas de texto correspondientes las ordenadas del control 8 puede introducir el caracter A. De esta manera, el programa decidirá el valor máximo y mínimo de las ordenadas de modo que no se pierda información de la señal en el rango de abscisas definido.

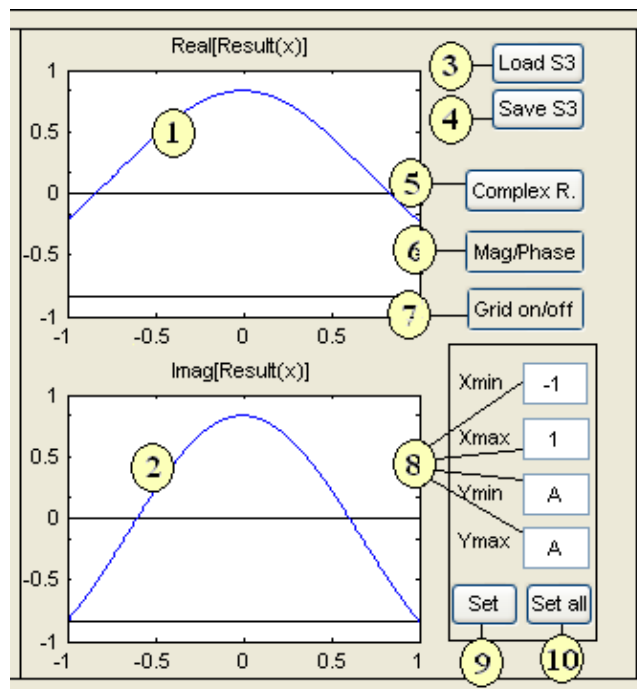


Figura 5.14: Controles de la ventana de visualización. En esta imagen puede observar los distintos controles que puede utilizar para estudiar la representación gráfica de las señales del sistema.

Como puede observar en las figuras 5.12 y 5.13, cuando esté analizando un sistema la visualización de la segunda señal desaparecerá. Esto es debido a que la segunda señal se corresponde con el conjunto de respuestas al impulso que consiste, como su nombre indica, en un conjunto de señales y, por lo tanto, deben representarse en tres dimensiones. Para acceder a su visualización deberá pulsar el botón correspondiente, etiquetado como 2 en la figura 5.13. Lo mismo ocurre

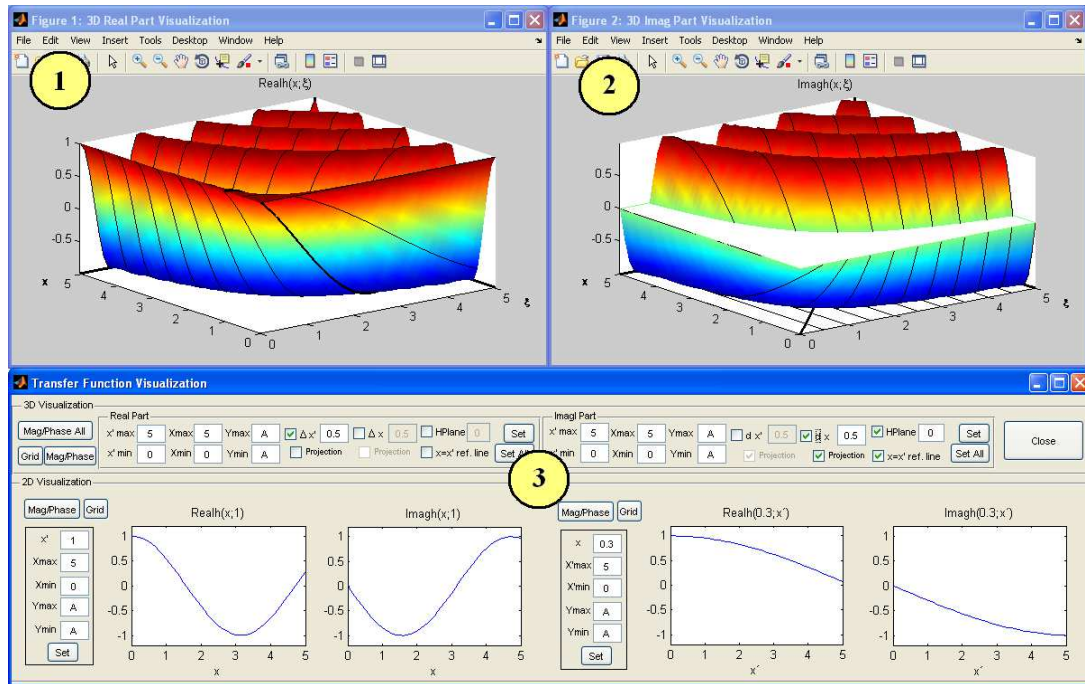


Figura 5.15: Ventana de representación del conjunto de respuestas al impulso. El caso mostrado es para el espacio de partida continuo aperiódico, donde ambas variables de representación son continuas y definidas para todo el rango de valores. En la parte superior de la interfaz, correspondiente a las subventanas 1 y 2, se representa el conjunto de respuestas al impulso. En la parte inferior de la pantalla, correspondiente a la subventana 3, se representan los cortes en ambas variables.

para el análisis espectral del sistema. Para visualizar la gráfica que falta deberá pulsar el botón etiquetado como 5 en la misma figura. En la versión actual del software todavía no se puede representar el conjunto de funciones de transferencia del sistema, sin embargo, se ha habilitado una nueva ventana con el objeto de incluir esta funcionalidad en futuras versiones.

5.2.4. Ventana de visualización del conjunto de respuestas al impulso.

Cuando analice un sistema de comunicaciones la ventana de visualización mostrará únicamente la señal de entrada y la de salida a dicho sistema. Para poder analizar el sistema en profundidad (conocer su comportamiento y sus propiedades) el software le permite visualizar el conjunto de respuestas al impulso. Dicho conjunto de respuestas sólo se podrá visualizar cuando el sistema analizado sea lineal, en caso contrario el software mostrará una ventana advirtiéndole que su representación no está disponible.

Dado que la respuesta al impulso es un conjunto de funciones el software dispone de una nueva ventana, fig 5.15, para su representación. Como puede observar, la interfaz de representación se estructura en tres subventanas. En las dos primeras ventanas se representa el conjunto de respuestas al impulso: en la primera ventana se representará la parte real o módulo y en la segunda la parte imaginaria y fase del conjunto de funciones. La tercera ventana se utiliza para representar cortes del conjunto de respuestas al impulso y ejecutar algunas ordenes sobre las dos primeras ventanas.

En la figura 5.16 se muestra con mayor detalle un ejemplo de las dos primeras subventanas de la interfaz de visualización del conjunto de respuestas al impulso. Como puede observar la ventana, además de contener la gráfica de representación, posee varios controles en la parte superior que

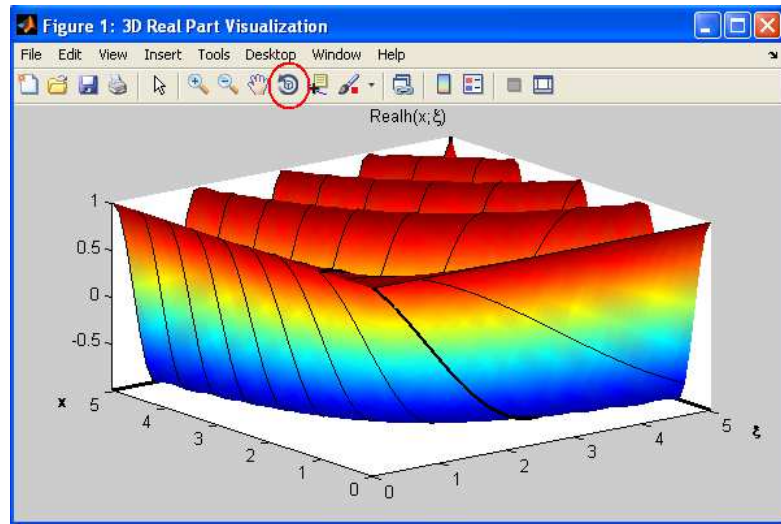


Figura 5.16: Representación en tres dimensiones del conjunto de respuestas al impulso. Cada una de las dos subventanas de la interfaz le permitirá manipular la representación para poder observarla de la mejor forma posible. Le permitirá: rotarla, cambiar la escala de representación, guardarla, etc.

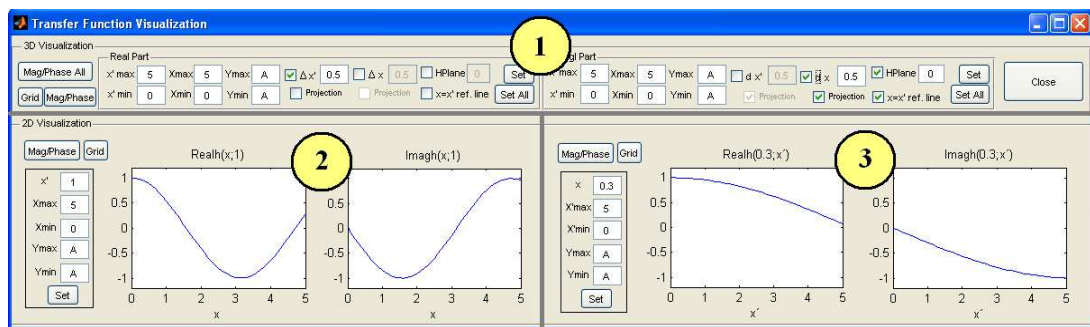


Figura 5.17: Representación en dos dimensiones de cada una de las funciones que componen el conjunto de respuestas al impulso y controles para modificar la visualización de las gráficas en tres dimensiones.

le permitirán, entre otras cosas, guardar la representación, manipularla, conocer el valor exacto de los puntos representados o hacer zoom sobre ella. De entre todos los controles cabe destacar el de rotación, señalado con un círculo rojo en la figura 5.16. Este control le permitirá rotar la figura para poder observarla desde cualquier punto de vista.

A su vez, la parte inferior de la ventana de representación se estructura en tres partes, véase la fig. 5.17:

1. Controles para la representación en tres dimensiones: en esta parte de la ventana podrá modificar los modos de representación, los límites de visualización de las representaciones y otras funcionalidades de representación para las dos gráficas en tres dimensiones.
2. Representación de un corte del conjunto de respuestas al impulso para un valor concreto de la variable descriptiva, definido por el usuario.
3. Representación de un corte del conjunto de respuestas al impulso para un valor concreto de la variable de definición del espacio, determinado por el usuario.

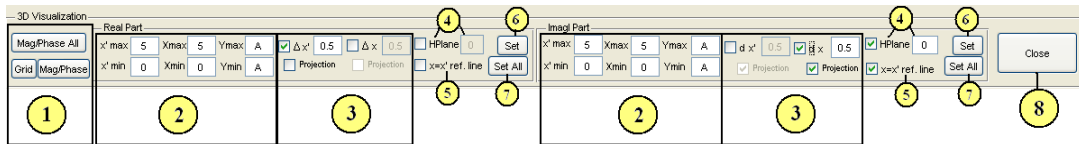


Figura 5.18: En esta parte de la ventana podrá modificar la representación del conjunto de respuestas al impulso: sus límites de representación o el modo de representación. También podrá mostrar u ocultar algunas líneas y planos de ayuda.

A continuación se explicarán los controles y funcionalidades de las tres zonas de la ventana mostrada en la figura 5.17.

La primera parte se muestra con más detalle en la figura 5.18, donde se han etiquetado los distintos controles que se explican a continuación:

1. Controles para el modo de representación y la rejilla. Este apartado cuenta con tres botones: **Mag/Phase All** (con el que podrá cambiar el modo de representación de todas las gráficas de la ventana), **Mag/Phase** (con el que podrá cambiar el modo de representación de las ventanas de visualización en tres dimensiones) y el botón **Grid** (con él podrá activar o desactivar la rejilla de las gráficas en tres dimensiones).
2. Límites de representación. Estos conjuntos de controles le permitirán modificar los límites de representación de ambas representaciones de forma independiente. Podrá introducir los valores que quiera que tomen tanto la variable de definición como la variable descriptiva. También podrá definir el rango de valores representados para el eje de ordenadas. Con en todas las representaciones del software dichos valores, los correspondientes al eje de ordenadas, podrán tomar el valor A (automático) adaptándose así a los valores que tome la función.
3. Representación de líneas sobre la curva. Estos controles le permitirán dibujar líneas sobre la curva representada a intervalos fijos, de este modo podrá representar líneas separadas entre sí un valor fijo. Puede visualizar un ejemplo de esta funcionalidad en la figura 5.16, donde se visualizan las líneas a lo largo de la variable de definición para valores de la variable descriptiva separados 0.5. Podrá visualizar estas líneas a lo largo de ambas variables (de definición y descriptiva) y para los intervalos que desee. Adicionalmente podrá dibujar su proyección sobre el plano inferior de la representación. Esta funcionalidad puede resultar muy interesante para evitar errores de percepción. Estos controles sólo están disponibles cuando tanto la variable de definición como la descriptiva son continuas.
4. Representación de un plano horizontal. Con este botón podrá representar un plano horizontal, a la altura que usted indique, sobre la curva representada. Esta opción puede resultar muy útil para visualizar con mayor claridad cuales son los valores y zonas de la curva que superan un determinado umbral.
5. Línea de referencia $\tau = \tau'$. Esta línea se representará sobre el plano inferior horizontal cuando active esta opción. Como es lógico, la recta sólo se mostrará cuando, según los límites de representación, se esté representando algún valor en el que coincidan ambas variables (de definición y descriptiva). Esta línea resulta de mucha utilidad a la hora de estudiar el conjunto de respuestas al impulso ya que, cuando se analizan sistemas lineales e invariantes, todas las funciones del conjunto de respuestas al impulso son la misma centradas en el valor $\tau = \tau'$.
6. Botón **Set**. Cuando haya realizado todos los cambios en los controles anteriores deberá pulsar este botón para que se apliquen todos los cambios, aunque algunos de los controles

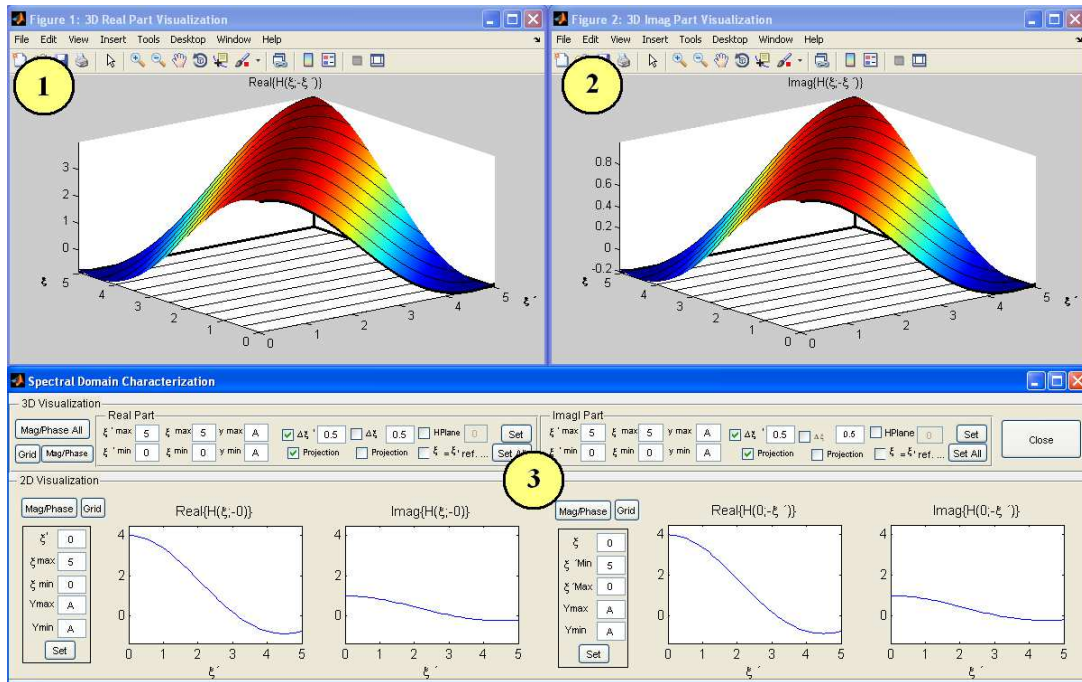


Figura 5.19: Ventana de representación del núcleo de funciones que caracterizan al operador seleccionado en el dominio transformado. El caso mostrado es para el espacio de partida continuo aperiódico, donde ambas variables de representación son continuas y definidas para todo el rango de valores. En la parte superior de la interfaz, correspondiente a las subventanas 1 y 2, se representa el conjunto de funciones de transferencia. En la parte inferior de la pantalla, correspondiente a la subventana 3, se representan los cortes en ambas variables.

anteriores se actualizan automáticamente, deberá pulsarlo para garantizar que los cambios se actualicen.

7. Botón **Set All**. Este botón, como el descrito en el punto anterior, guarda los cambios realizados en los controles anteriores. La diferencia que tiene con el anterior es que actualiza los cambios definidos en los controles que están en su panel (real o imaginario) en las dos representaciones (real e imaginaria).
8. Botón **Close**. Pulsando este botón cerrará las tres sub ventanas de la interfaz de representación del conjunto de respuestas al impulso.

5.2.5. Ventana de visualización del conjunto de funciones de transferencia.

Esta nueva actualización del software incluye como novedad principal la funcionalidad del análisis espectral generalizado. Cuando esta opción es seleccionada, se puede analizar por completo el sistema en el dominio transformado. Las señales de entrada y salida del sistema en el dominio espectral se visualizan en la ventana de visualización principal, mostrada en la fig. 5.13. El conjunto de funciones que caracterizan el operador en el dominio transformado se representarán en una nueva ventana, cuya estructura es muy similar a la ventana de representación del conjunto de respuestas al impulso.

Esta ventana, tal y como se muestra en la fig. 5.19, se divide a su vez en tres ventanas independientes. Las dos primeras se utilizan para representar las partes real / imaginaria o módulo / fase del conjunto de funciones de transferencia. La tercera ventana se utiliza para realizar representaciones en dos dimensiones de este conjunto de funciones y manipular ciertos controles que actúan sobre las representaciones en tres dimensiones.

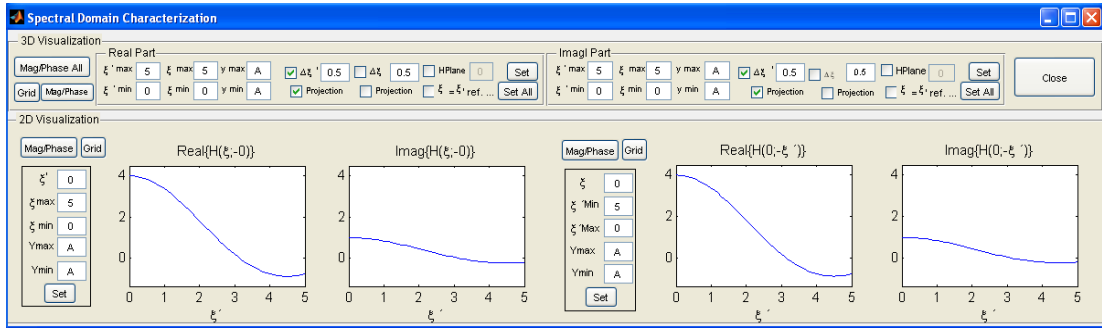


Figura 5.20: Cuando el operador seleccionado es lineal no invariante, en la tercera subventana de la representación del conjunto de funciones de transferencia $H(\tau; \tau')$ se representan los cortes de la superficie en ambas variables: τ y τ' . A la izquierda se representan los cortes para un valor de τ' concreto. A la derecha se muestran los cortes para un valor concreto de τ .

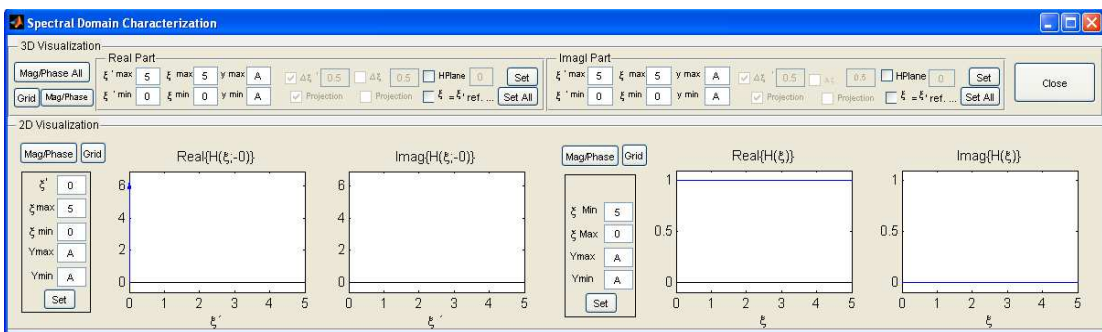


Figura 5.21: Cuando el operador seleccionado es lineal e invariante, en la tercera subventana de la representación del conjunto de funciones de transferencia $H(\tau; \tau')$ se realizan dos representaciones diferentes. Por un lado, en las representaciones de la izquierda se muestran los cortes para un valor de τ' concreto, que, por la naturaleza del conjunto de funciones de transferencia, toma los mismos valores que un corte en la otra variable de representación. Por otra parte, y he aquí la diferencia con el caso de los sistemas lineales no invariantes, a la derecha se representa la función de transferencia que caracteriza al operador seleccionado en el dominio espectral $H(\tau)$.

Dada la gran similitud de este interfaz con el descrito en el punto anterior no se expondrán en detalle sus utilidades y controles.

Cabe resaltar que las representaciones en dos dimensiones tendrán significados distintos en función de las características del sistema analizado. Cuando el sistema analizado sea lineal no invariante, caso más general, en estas gráficas se mostrarán cortes de la representación en tres dimensiones en ambas variables de representación, véase la fig. 5.20. Por otra parte, cuando el sistema sea lineal e invariante, el conjunto de funciones núcleo que describen al operador estará compuesto por un tren de deltas (en su significado más general) en la recta $\tau = \tau'$ (siendo estas las dos variables de representación). Dicho tren de deltas estará modulado por la función de transferencia que caracteriza el operador. De este modo, para un mismo valor fijo de una de las variables, el corte en la segunda variable de representación será siempre igual, independientemente de si se toma el corte en τ o en τ' . Por lo tanto, carece de especial interés disponer de ambas representaciones y se aprovecha el espacio de la segunda representación en dos dimensiones para representar la función de transferencia del operador, que no es más que la transformada de la respuesta al impulso que lo caracteriza en el dominio real. En la fig. 5.21 se puede ver un ejemplo de este caso particular, en el que el operador seleccionado es lineal e invariante.

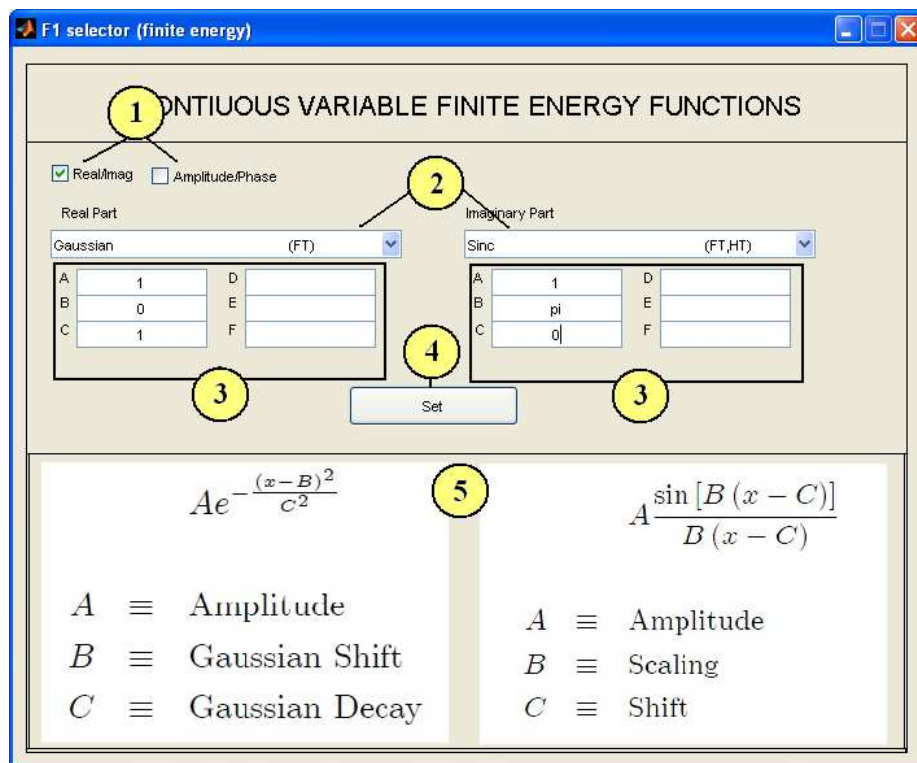


Figura 5.22: Selector de señales. Esta ventana será la que utilice para definir las señales que desee controlando sus parámetros y su modo de definición. Para que pueda saber a que parámetro corresponde cada casilla de definición se muestra la definición de la señal en la parte inferior de la ventana.

5.2.6. Selector de señales.

Utilizará esta ventana para definir las señales de entrada al sistema. Accederá a ella cada vez que pulse uno de los controles etiquetados como 5 de la figura 5.4 o sus equivalentes para la segunda señal de entrada, mostrados en la figura 5.5. Su nombre y las señales disponibles se verán modificados según pulse uno u otro botón. Sin embargo, su funcionamiento es siempre el mismo.

En la figura 5.22 se muestra la ventana en cuestión con sus controles etiquetados para su explicación:

1. Modo de definición. Con estos controles podrá escoger el modo de definición de la señal teniendo las dos opciones: modo Real/Imaginario o modo Módulo/Fase.
2. Selectores de la señal, uno para cada parte (real/módulo e imaginaria/fase). Ambos son menús desplegable. Según el menú con el que esté interactuando, usted deberá escoger la señal que quiera definir como parte real o módulo o como parte imaginaria o fase. Si no selecciona ninguna señal, el programa la interpretará como la señal cero.
3. Cajas de texto para los parámetros de definición de la señal. Dependiendo de la señal que haya escogido deberá establecer los parámetros necesarios para completar su definición. Estos parámetros suelen ser la amplitud, el desfase, la pendiente, etc.
4. Botón Close/Set. Cuando no haya ninguna señal definida se mostrará como Close y se utilizará para cerrar el selector sin realizar otra operación. Si usted ha definido alguna señal mostrará el texto Set y se utilizará para cerrar el selector quedando la señal definida y almacenada en el programa.
5. Ayuda de las señales seleccionadas. En estas casillas de la ventana el selector mostrará una imagen de ayuda con la definición de la señal. De este modo podrá saber en que caja de texto deberá introducir cada uno de los valores a definir.

En caso de que la señal definida tuviese algún error en su definición, se avisará en este punto de la ejecución y la señal quedará sin definir. Si no hay ningún error, una vez definida la señal obtendrá el valor de su norma y energía en el caso de que sean de energía finita y podrían visualizarla cuando quisiesen.

Para las señales numéricas, en este momento se abre también la ventana para la definición del rango, véase Fig. 5.24. En el caso de señales no numéricas, cuando esté disponible la funcionalidad, se abrirá también una ventana de la misma estructura que se utilizará para previsualizar la señal con los parámetros definidos.

5.2.7. Selector de los parámetros de una señal.

El selector de parámetros, mostrado en la figura 5.22, tiene una funcionalidad y una estructura prácticamente idéntica a la del selector de señales expuesto en el apartado anterior. La diferencia con respecto al selector de señales es que el usuario ya no puede seleccionar una señal sino que ésta vendrá definida por el operador y el usuario sólo podrá modificar algunos de sus parámetros (amplitud, desfase, etc.).

Este selector se activa cuando el operador seleccionado tiene una señal concreta que le caracteriza y si esta se cambia el operador ya no es el mismo. En estos casos, el usuario sólo podrá modificar algunos parámetros de la señal y por ello se ha definido esta ventana. Un ejemplo de esta interfaz se muestra en la figura 5.23. Los controles etiquetados servirán para:

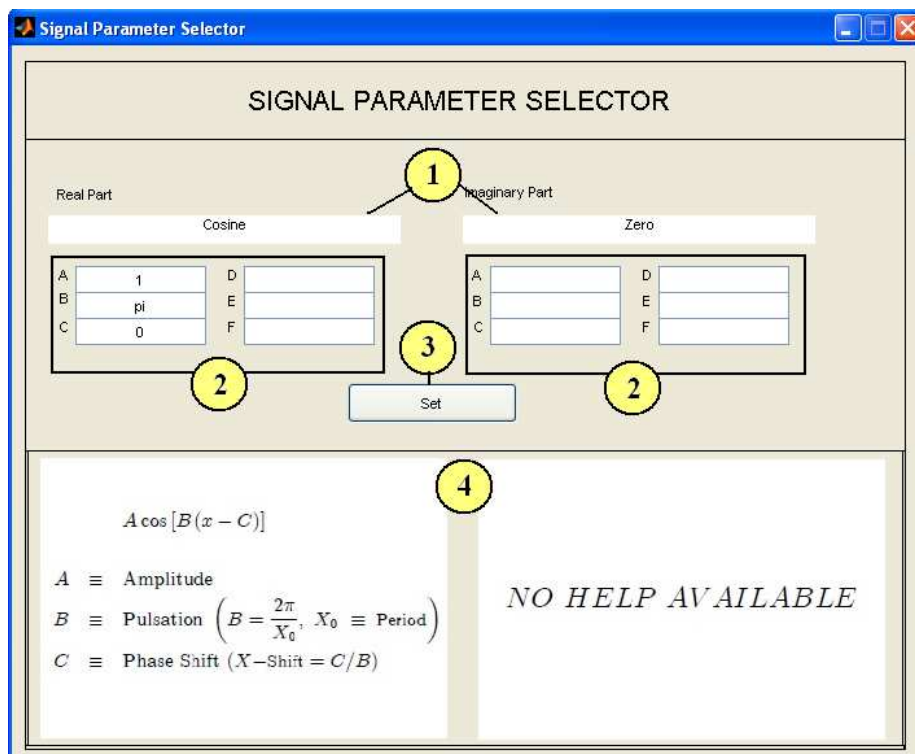


Figura 5.23: Selector de parámetros. El acceso a esta ventana se activará cuando el operador del sistema tenga una función característica concreta y el usuario sólo pueda modificar algunos parámetros de la misma. Su estructura es la misma que la del selector de señales pero los menús desplegables se sustituyen por un control de texto no manipulable con el nombre de la señal.

1. Definición de la señal. Estos controles son informativos y muestran el nombre de la señal para que pueda reconocerla.
2. Cajas de texto para los parámetros de definición de la señal. Como en el selector de señales utilizará estos controles para modificar los parámetros de la señal. Al abrir esta ventana las cajas de texto contienen unos valores predefinidos que el usuario podrá modificar a su gusto.
3. Botón Set. Este botón se utiliza para cerrar el selector de parámetros y guardar los cambios que haya realizado.
4. Ayuda de las señales. En estas casillas de la ventana el selector mostrará una imagen de ayuda con la definición de la señal. De este modo podrá saber en que caja de texto deberá introducir cada uno de los valores a definir.

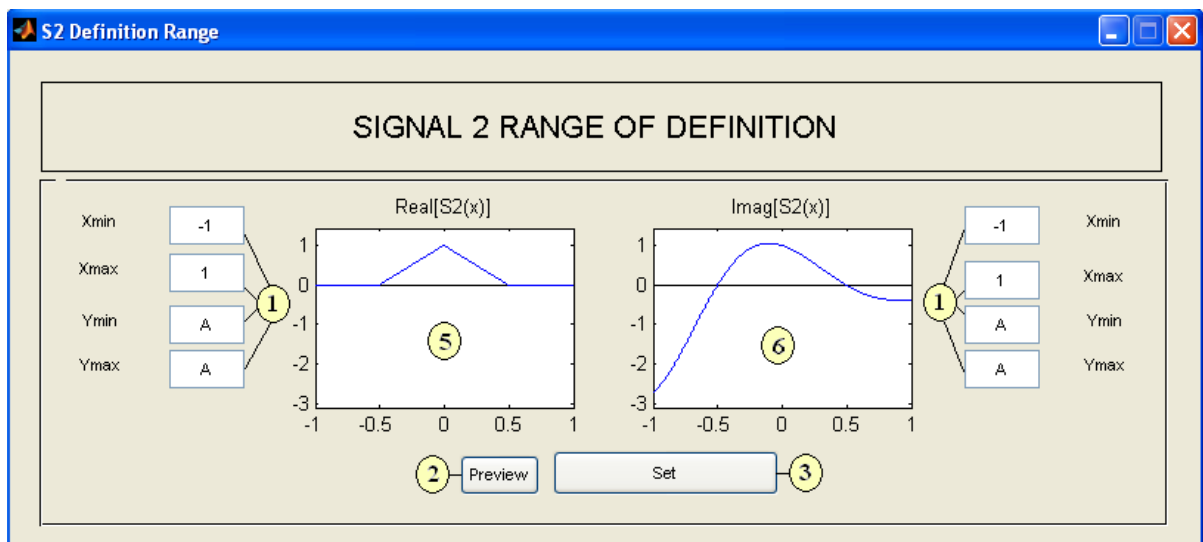


Figura 5.24: Ventana para definir el rango de las señales. Sólo lo utilizará para las señales de tipo numérico. Podrá definir los límites de las señales teniendo en cuenta que fuera de éstos la señal tomará el valor cero.

5.2.8. Selector del rango de definición.

Utilizará esta ventana para definir el soporte finito de una señal numérica. Este soporte finito será el rango de abscisas y ordenadas donde quiera que la señal tome valores significativos, fuera de dicho rango la señal tomará el valor cero. Sólo podrá acceder a esta ventana cuando la señal definida sea de carácter numérico, en otro caso la ventana carecerá de sentido.

En la figura se muestra el selector del rango que se explica a continuación:

1. Rango de definición de la primera parte de la señal: introduzca en estas cajas de texto el rango de definición para la parte real o módulo.
2. Rango de definición de la segunda parte de la señal: introduzca en estas cajas de texto el rango de definición para la parte imaginaria o fase.
3. Previsualización: pulse este botón para visualizar la señal con los valores de abscisas y ordenadas que haya escogido.
4. Botón Set: pulse este botón para redefinir la señal para los valores de rango escritos en las cajas de texto.

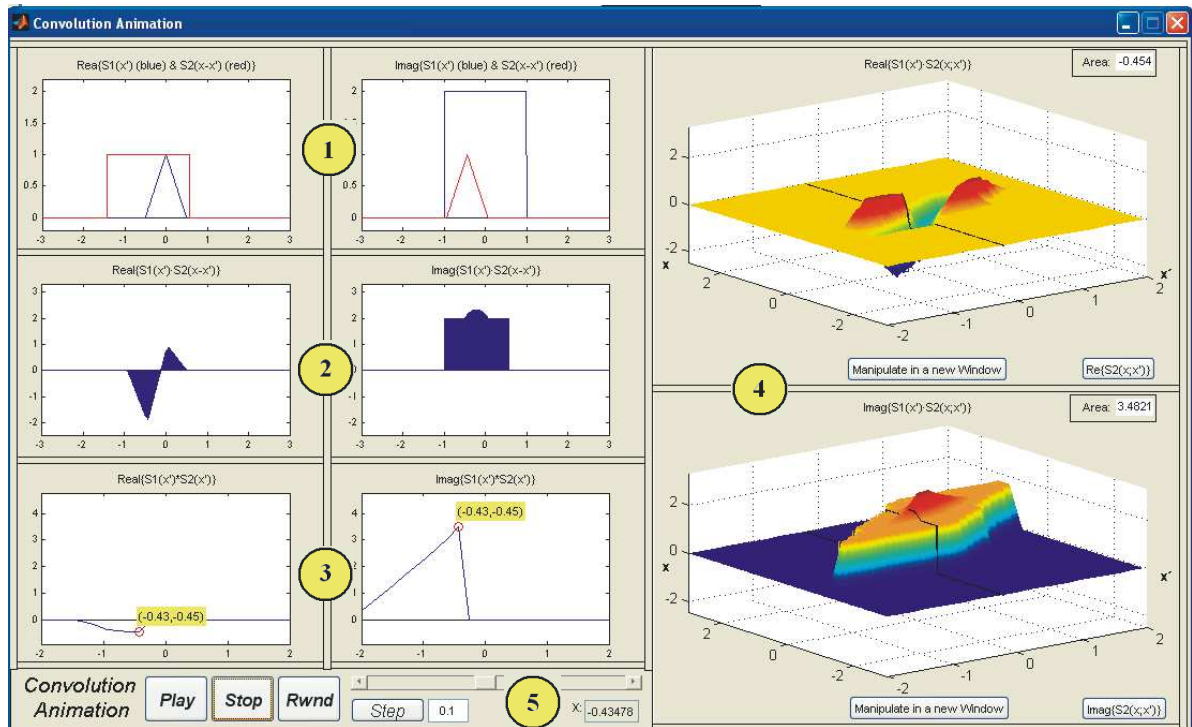


Figura 5.25: Ventana de animación. En esta ventana se muestra como se realizan paso a paso las operaciones de convolución, correlación y multiplicación.

5. Primera gráfica: en ella se mostrará la primera parte de la señal (la parte real o el módulo).
6. Segunda gráfica: en ella se mostrará la segunda parte de la señal (parte imaginaria o fase).

Como ya comentamos en la pantalla de visualización, en las cajas de texto correspondientes las ordenadas de los controles 1 y 2 puede introducir el carácter A. De esta manera, el programa decidirá el valor máximo y mínimo de las ordenadas de modo que no se pierda información de la señal en dicho rango de abscisas.

Debe notar que este rango no tiene nada en común con el establecido por los controles 8 y 9 de la figura 5.14. En dicho caso hablábamos del rango de visualización (fuera de dicho rango la señal podía seguir tomando valores significativos) y en este caso hablamos del rango de definición.

5.2.9. Ventana de animación.

La puede abrir cuando este activado y pulse el botón correspondiente, etiquetado como 9, de la ventana 5.5. Esta ventana, véase la Fig. 5.25, muestra la animación de las operaciones multiplicación, convolución y correlación cuando se obtienen de forma numérica. Para ello las dos señales que intervienen deben estar definidas como funciones numéricas. Según cual sea la operación seleccionada algunos aspectos visuales de la ventana, como pueden ser su nombre o el título inferior, se verán modificados.

En el caso de que alguna las dos señales que intervienen tengan definidos rangos distintos para la primera y la segunda parte, el rango se modificará temporalmente. La animación se realizará con las dos partes de la señal definidas en el mismo rango: el más amplio según los dos rangos definidos. Por ejemplo, si el rango de la primera parte es (-1,2) y el de la segunda (0,5), el rango de la señal para la animación será el (-1,5).

Dicho esto, la ventana, con ayuda de las etiquetas mostradas en la figura 5.25, se explica a continuación:

1. Las dos gráficas representan lo mismo, una para la parte real y otra para la parte imaginaria de la señal. En ellas se representa la primera señal tal y como se haya definido y superpuesta sobre ella irá desplazándose la segunda señal y sufriendo las transformaciones que definen cada operación.
2. En estas gráficas se representa el producto de la primera señal con la segunda para un valor concreto de desplazamiento (el desplazamiento dependerá del instante de simulación). Como en el caso anterior la primera gráfica corresponde al producto de las partes reales y la segunda al de las partes imaginarias.
3. Por último se representa la señal resultado. En estas gráficas se representa la parte real y la parte imaginaria de la señal resultado de la operación seleccionada.
4. En estas gráficas se representa el producto de la señal de entrada y la segunda señal en tres dimensiones, es decir, representa el conjunto de respuestas al impulso multiplicado por la señal de entrada. Esta parte de la ventana se subdivide a su vez en dos partes: la parte superior en la que se representa la parte real y la inferior en la que se representa la parte imaginaria. Para cada una de las partes la ventana dispone de tres controles: una caja de texto situada sobre la gráfica a la derecha que contiene el valor resultado de la operación para cada instante (será la integral o suma), el botón **Manipulate in a new Window** para abrir la gráfica en una nueva ventana y así poder rotarla, hacer zoom, etc. y el botón $\text{Re}\{h(\tau; \tau)\} / \text{Im}\{h(x; x)\}$ que muestra en una nueva ventana el conjunto de respuestas al impulso.
5. En la última parte de la ventana dispone de algunos controles para actuar sobre la animación. Deberá pulsar el botón **Play** para iniciar la animación o retomarla. Podrá pulsar el botón **Stop** para parar la animación y el botón **Rewind** para volver al inicio de la animación. Además hay una barra de desplazamiento para que pueda seleccionar el momento de la animación que desee visualizar. Adicionalmente dispone del botón **Step** para avanzar a lo largo de la animación cuando lo pulse, este botón viene acompañando de la caja de texto correspondiente en la que definirá el valor del salto. Finalmente incluye una caja de texto en la que se muestra el valor de abscisas actual.

5.2.10. Ventana para la representación compleja.

Esta ventana se abre desde el interfaz de visualización, pulsando el control 5 de la figura 5.14. En ella podrá ver la representación compleja de la señal: Real/Imaginaria o Módulo/Fase. Como se observa en la figura 5.26, la ventana no tiene ningún tipo de control y sólo se utiliza para visualizar.

5.2.11. Ventana de ayuda.

Esta ventana se abrirá cada vez que pulse un botón de ayuda. Hay varios en el programa y se identifican por tener escrita una interrogación.

Se trata de una ventana en blanco en la que se mostrará una imagen con la ayuda correspondiente: definición de parámetros algebraicos, definición de espacios de señales u otros tipos de ayuda relacionados con el funcionamiento del software.

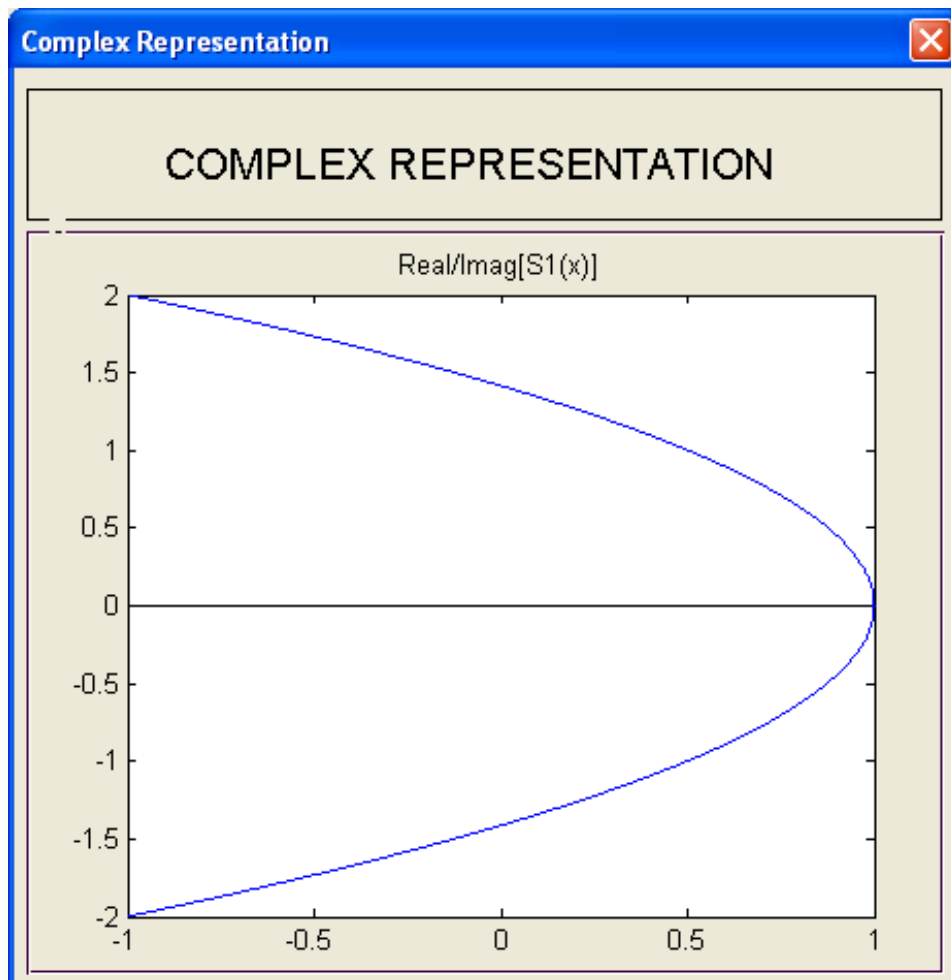


Figura 5.26: Ventana para la representación compleja. Podrá acceder a ella desde el interface de visualización. En ella podrá observar la representación compleja de las distintas señales bien real/imaginaria o módulo/fase según esté pulsado el botón de modo en la ventana de visualización.

5.3. Ficheros de datos.

Esta versión del programa le permite trabajar con ficheros de datos, aunque sólo a nivel de visualización. Se considera muy interesante sobretodo a la hora de comparar unas señales con otras. Por ejemplo para comparar señales que resultan de una misma operación pero se han obtenido por métodos diferentes.

Para trabajar con ellos simplemente hay que pulsar los botones correspondientes de la ventana de visualización, controles y de la figura . Con dichos controles podrá tanto guardar señales que haya obtenido como cargar al interfaz señales que tenga almacenadas.

Hay definidos cuatro formatos de archivo diferentes, dependiendo de si guardan información correspondiente a una función periódica o aperiódica, de variable continua o discreta. Éstos son:

- CPF. Continuous variable periodic function, o función periódica de variable continua.
- CAF. Continuous variable aperiodic function, o función aperiódica de variable continua.
- PDS. Periodic discrete sequence, o secuencia discreta periódica.
- ADS. Aperiodic discrete sequence, o secuencia discreta aperiódica.

En consecuencia, cuando desee cargar una señal desde un fichero de datos se abrirá una ventana para que busque y seleccione dicho fichero. En esta ventana sólo aparecerán los archivos del tipo adecuado.

Del mismo modo, cuando guarde una señal en un fichero de datos el programa la salvará automáticamente con la extensión correspondiente.

Cabe destacar que en los espacios continuo periódico y discreto aperiódico no se pueden guardar los resultados de una transformación, desarrollo en serie o sus inversos ya que el espacio de dichas señales no coincide con el de entrada al sistema.

5.3.1. Escritura manual de datos en los ficheros.

En este apartado se procede a explicar la construcción interna de los ficheros de datos para que, aunque no sea necesario, pueda modificarlos o crearlos si así lo desea.

El formato interno de los archivos es muy similar para los cuatro tipos definidos: tres columnas de valores numéricos. Según el tipo de fichero lo que varía es el significado de las primeras dos filas de valores:

```

// CPF file - Continuous variable periodic function data for C:\Documents and Se
// Created on 30-Mar-2008. //

// *** NOTE: For Period and Mode rows, 2nd and 3rd values are unused *** //

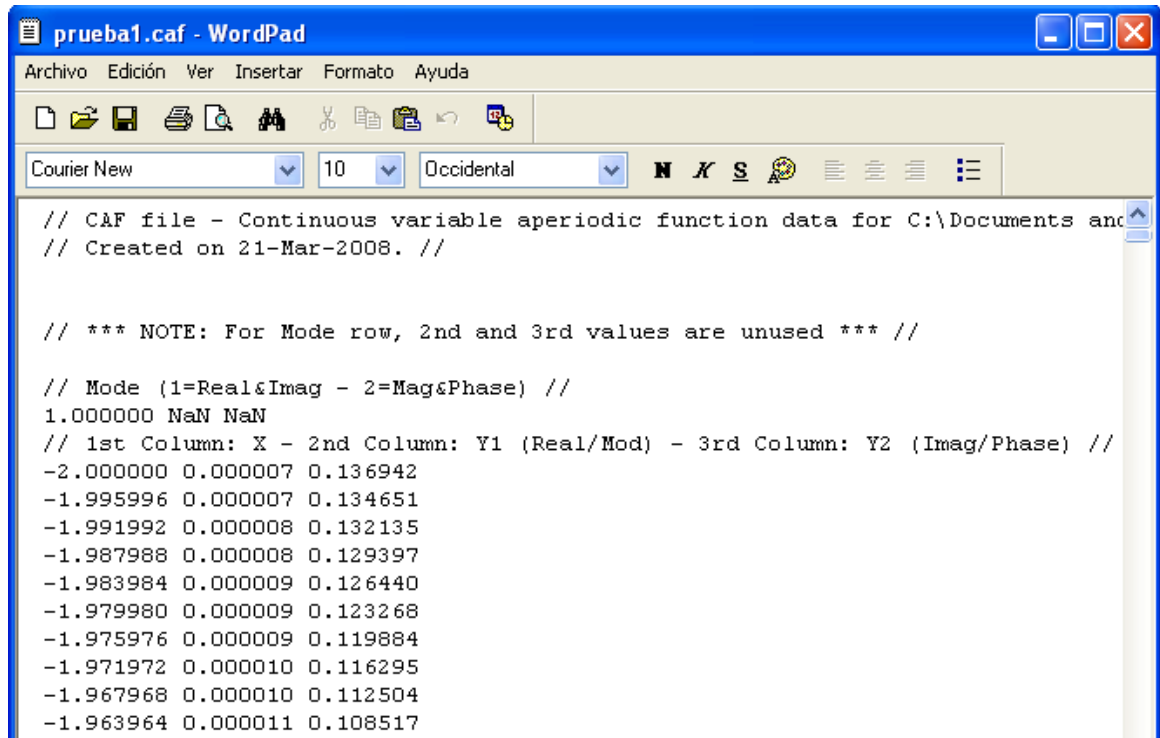
// Period Xo //
1.000000 NaN NaN
// Mode (1=Real&Imag - 2=Mag&Phase) //
1.000000 NaN NaN
// 1st Column: X - 2nd Column: Y1 (Real/Mod) - 3rd Column: Y2 (Imag/Phase) //
-2.000000 0.091957 -0.023123
-1.000000 0.494443 -0.097116
0.000000 0.636620 1.456742
1.000000 -0.070029 -0.097116
2.000000 -0.176839 -0.023123

```

Figura 5.27: Ejemplo de fichero CPF. Fichero de datos en el que se almacena una señal perteneciente al espacio continuo periódico.

- Formatos periódicos (CPF, PDS). El primer valor de la primera fila, es decir, el valor (1,1), es el periodo. El primer valor de la segunda fila, el valor (2,1), es el modo en que está definida la señal dentro del archivo. Puede tomar el valor 1 si está definida en modo Real/Imaginario, o el 2 si está definida en modo Módulo/Fase. El resto de valores de las filas 1 y 2, no tienen utilidad alguna, pero deben estar presentes para que el archivo sea leído correctamente. Cuando se salva alguna señal, el programa les asigna por defecto el valor NaN (not a number en lenguaje MATLAB). Podemos ver un ejemplo en la figura 5.27.
- Formatos aperiódicos (CAF, ADS). El primer valor de la primera fila, el (1,1), representa el modo de definición. Como ya hemos dicho, valdrá 1 para modo Real/Imaginario y 2 para modo Módulo/Fase. También en este caso el resto de valores de la primera fila tienen que estar presentes aunque no tengan utilidad, y si es el propio programa el que crea el archivo, se pondrán por defecto a NaN. La figura 5.28 muestra un ejemplo de este tipo de archivos.

Para el resto de filas del archivo el significado es independiente del formato que estemos utilizando. La primera columna representará los valores de las abscisas, y las columnas segunda y tercera valores de ordenadas de la señal. Si el modo de definición toma el valor 1, es decir, es el Real/Imaginario, la segunda columna representará la parte real de la señal y la tercera la parte imaginaria. Si por el contrario el modo de definición es 2, Módulo/Fase, la segunda columna será el modulo mientras que la tercera será la fase. De esta forma cada valor de abscisas en la primera columna tendrá asociado un valor de ordenadas, dado por los valores de las columnas 2 y 3. Es necesario hacer notar que los valores de dichas columnas tienen que estar presentes para que el archivo se pueda leer correctamente, aunque sean nulos. Si es preciso, introduzca el cero explícitamente.



```
// CAF file - Continuous variable aperiodic function data for C:\Documents and
// Created on 21-Mar-2008. //

// *** NOTE: For Mode row, 2nd and 3rd values are unused *** //

// Mode (1=Real&Imag - 2=Mag&Phase) //
1.000000 NaN NaN
// 1st Column: X - 2nd Column: Y1 (Real/Mod) - 3rd Column: Y2 (Imag/Phase) //
-2.000000 0.000007 0.136942
-1.995996 0.000007 0.134651
-1.991992 0.000008 0.132135
-1.987988 0.000008 0.129397
-1.983984 0.000009 0.126440
-1.979980 0.000009 0.123268
-1.975976 0.000009 0.119884
-1.971972 0.000010 0.116295
-1.967968 0.000010 0.112504
-1.963964 0.000011 0.108517
```

Figura 5.28: Ejemplo de fichero CAF. Fichero de datos en el que se almacena una señal perteneciente al espacio continuo aperiódico. Como se observa aquí desaparece la línea con la información del periodo.

También debe tener en cuenta que, para facilitar la edición de los archivos, el programa interpolará linealmente la señal en los intervalos existentes entre los nodos que aparecen especificados en el fichero de datos. Sin embargo y debido a esto debe tomar dos precauciones a la hora de editar los ficheros:

- Introducir datos de al menos dos nodos, para que el programa pueda llevar a cabo correctamente la interpolación.
- Si quiere introducir datos para puntos aislados, y que no se lleve a cabo la interpolación en los valores intermedios (caso especialmente interesante para trabajar con secuencias discretas), deberá introducir cada nodo rodeado de otros dos con las siguientes características: los valores de abscisas deben ser próximos a él, uno por debajo y otro por encima, y los valores de ordenadas correspondientes a ellos nulos. De esta forma el programa lleva a cabo la interpolación pero, al estar cada nodo de datos rodeado de ceros, siempre se interpolan valores nulos.

Por último cabe reseñar que puede introducir comentarios en los archivos de datos de forma simple: añadiendo // al comienzo de cada línea de comentario.

5.4. Ejemplos de Uso.

En este apartado del manual de usuario se presentan algunos ejemplos de los múltiples casos de análisis disponibles en el software SST v4.0. En general la metodología que deberá seguir para estudiar cualquiera de ellos es la misma y resulta muy fácil de seguir gracias al diseño de las interfaces. Adicionalmente a la metodología que deberá seguir se expondrán algunas observaciones teóricas que podrá realizar cuando estudie casos similares.

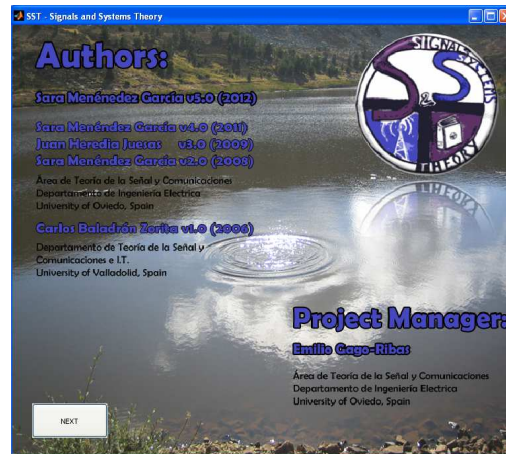


Figura 5.29: Ventana de inicio. Se muestra al arrancar el programa. Pulse el único botón que aparece, cuyo texto es Next, para continuar.

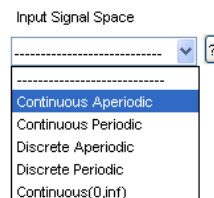


Figura 5.30: Seleccione el espacio de partida.

5.4.1. Análisis del Operador Multiplicación para el Espacio Continuo Aperiódico.

En este ejemplo se describirán los pasos que deberá seguir para estudiar el comportamiento del operador multiplicación cuando el espacio de partida es el de señales de variable continua aperiódicas.

1. Ejecute el software para comenzar a utilizarlo. Cuando el programa comience su ejecución visualizará la ventana de inicio, véase fig. 5.29. Para continuar pulse el botón Next de la ventana y se abrirá la nueva interfaz.
2. En el menú desplegable denominado Input Signal Space seleccione el espacio Continuous Aperiodic, fig. 5.30 y automáticamente se cargarán en la pantalla los controles para avanzar. Como puede observar, ahora la pantalla aparece, además del nombre del espacio de partida, su esquema algebraico, fig. 5.31.
3. La siguiente acción a realizar sería seleccionar un algebra para definir el espacio. En esta versión del software sólo hay una definición disponible por lo que se obviará este paso.
4. A continuación se deberá definir una señal de entrada al sistema. En el caso de la multiplicación, el software no dispone de un método de cálculo numérico por lo que deberá seleccionar una señal numérica. Por lo tanto pulse sobre el botón Select del tipo de señales Numerical Distributions.
5. Una vez haya pulsado el botón se abrirá la ventana para seleccionar la señal, véase fig. 5.32. Ahora deberá definir la señal de entrada siguiendo los pasos descritos a continuación:

: Lo primero que debe decidir es si va a definirla en modo real e imaginario o en módulo

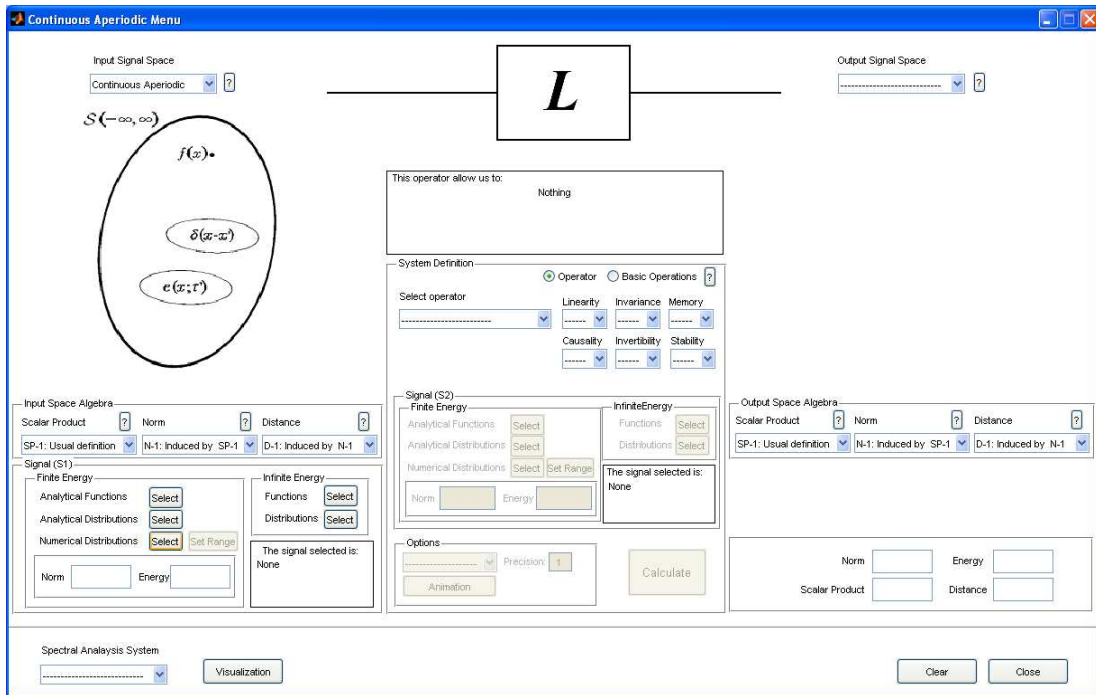


Figura 5.31: Apariencia inicial del menú cuando seleccione el espacio Continuo Aperiódico.

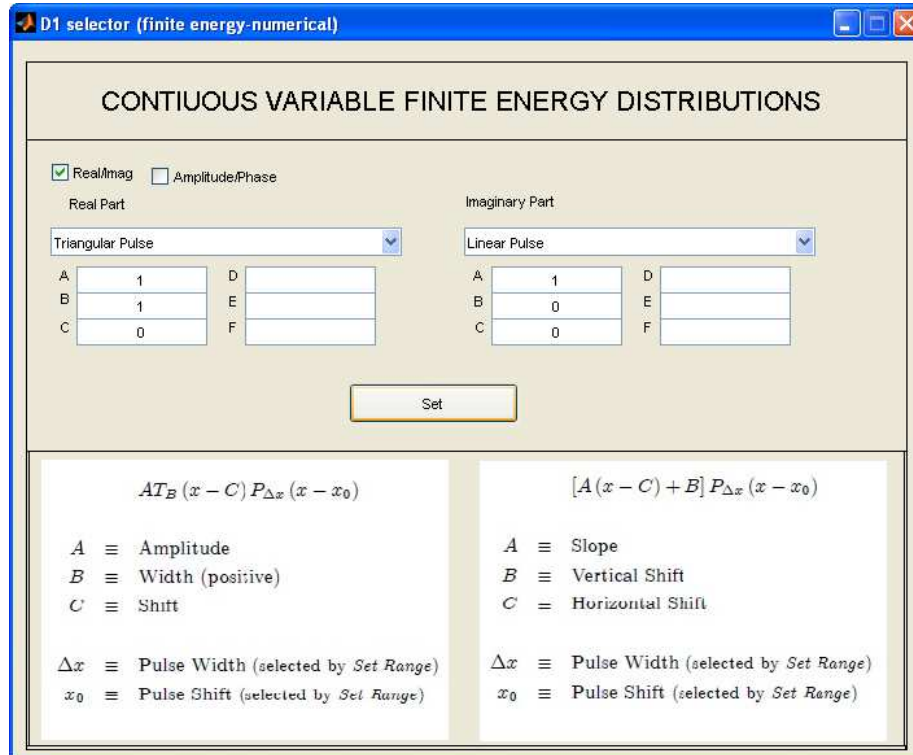


Figura 5.32: Ventana del selector de señales con las señales seleccionadas para definir la señal de entrada al sistema.

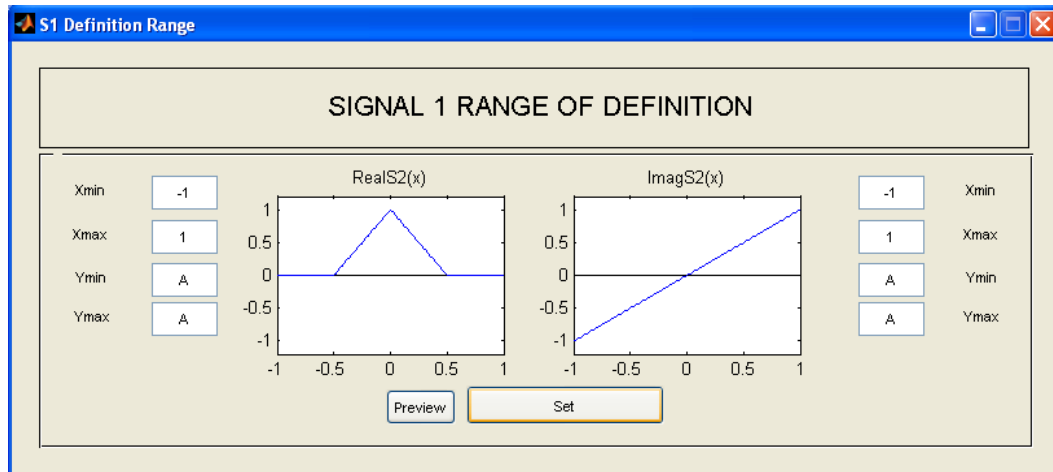


Figura 5.33: Ventana para seleccionar el rango de definición de la señal de entrada. En realidad, este rango define el ancho y desplazamiento del pulso rectangular que multiplica la señal de entrada.

y fase y comunicárselo al software utilizando los check box que aparecen en la parte superior de la ventana a al izquierda (bajo el título). Para este caso se va a definir la señal en modo real e imaginario por lo que deberá marcar la casilla Real/Imag.

- : A continuación se definirá cada una de las dos partes de la señal, primero la real y después, del mismo modo, la imaginaria. Para ello despliegue el menú que hay bajo el título **Real Part** y seleccione la señal que desee. A continuación observe la ayuda mostrada en la parte inferior de la ventana y asigne un valor a cada uno de los parámetros que aparecen en ella identificados por una letra de la A a la F (ignore los parámetros Δx y x_0 pues se definirán a continuación). En este ejemplo se define como parte real un pulso triangular de amplitud 1, ancho 1 y desplazamiento 0 y para la parte imaginaria un pulso lineal de pendiente 1 y desplazamientos, horizontal y vertical, 0.
 - : Una vez asignados los parámetros pulse el botón **Set** para cerrar la ventana.
 - : Una vez cerrada esta ventana se abrirá una nueva ventana, véase fig 5.33, para definir el rango. El rango sólo se definirá en las señales numéricas, pues las señales analíticas, aunque tiendan a cero, toman valores no nulos en toda la variable. En realidad, lo que se define como señales numéricas en este software son señales analíticas multiplicadas por una ventana rectangular o pulso. Los parámetros de dicha ventana serán claves para definir la señal y son los que deberá definir en esta ventana: Δx , ancho del pulso, y x_0 , desplazamiento del pulso. Estos parámetros se pueden definir de forma independiente para cada una de las partes de la señal. En este ejemplo se definirán ambas partes del mismo modo: $\Delta x = 2$ y $x_0 = 0$, lo que se traduce a $x_{\min} = -1$ y $x_{\max} = 1$. Una vez establecidos los valores deseados pulse el botón **Set** y la ventana se cerrará dejando la señal completamente definida.
 - : Ahora en el menú principal podrá ver el nombre de las funciones que definen cada parte de la señal así como su norma y su energía, véase fig 5.34.
6. A continuación se definirá el sistema a través del operador multiplicación. Para ello asegúrese de que los botones circulares de la caja **System Definition** tengan la opción **Operator** seleccionada y seleccione la opción **Multiplication** del menú desplegable denominado **Select Operator**. Como se muestra en la fig 5.35, automáticamente se modificará el esquema

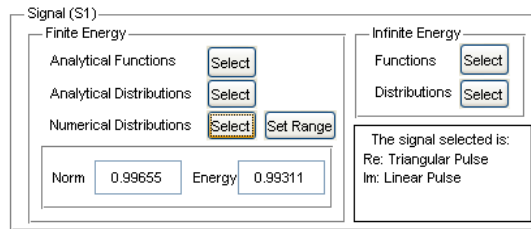


Figura 5.34: Panel de la señal de entrada en el que se muestra cual es la señal definida y sus valores de norma y energía.

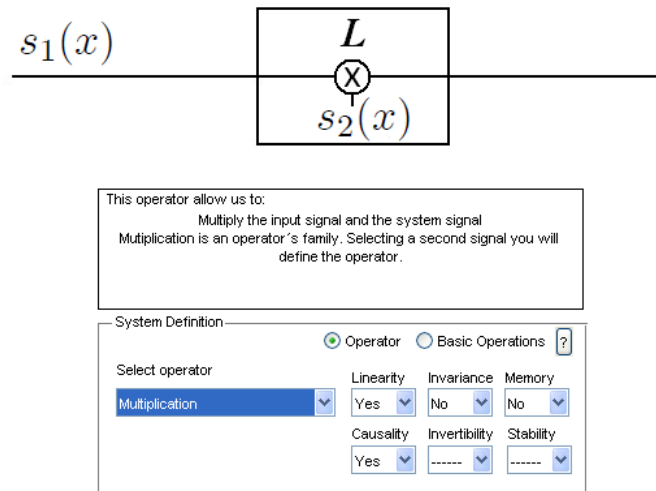


Figura 5.35: La caja que representa el sistema cambiará para mostrar un esquema que represente al operador seleccionado. Además se mostrará una breve descripción del operador y las propiedades que tiene.

del operador y aparecerá un texto explicativo bajo la imagen. En este caso el texto indica que:

- : El operador seleccionado multiplica la señal de entrada por la señal que define al sistema.
 - : El operador multiplicación es en realidad una familia de operadores.
 - : Una vez se haya definido la señal del sistema el operador quedará completamente definido, ya no será una familia de operadores sino uno en concreto.
7. Además de la información citada anteriormente, el programa le mostrará cuales son propiedades del operador. Resultaría muy conveniente y educativo que realizase una parada en este momento para asimilar cuales son las propiedades del operado y así después intentar reconocerlas cuando visualice el conjunto de respuestas al impulso que caracteriza al operador. El operador multiplicación, como se muestra en la fig 5.35, es un operador lineal, no invariante, sin memoria y causal. Las propiedades de invertibilidad y estabilidad dependerán de la señal que defina el operador.
 8. Dado que el operador multiplicación representa a una familia de operadores deberá definir la señal interna al sistema (aquella que multiplicará a cualquier señal de entrada). Dado que el operador multiplicación sólo trabaja con señales numéricas todos los selectores de operaciones excepto el denominado **Numerical Distributions** estará inhabilitados. Pulse sobre

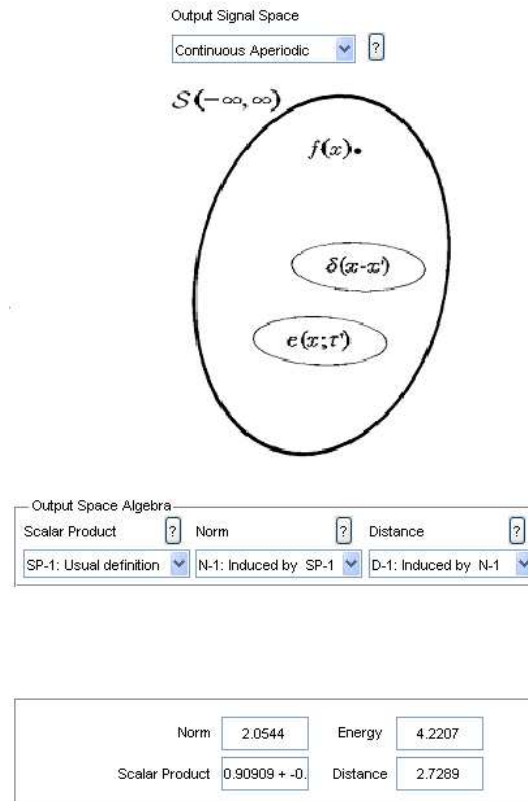


Figura 5.36: La salida del sistema se modificará cuando se haya calculado el resultado: se mostrará el espacio de salida y los valores numéricos asociados a ella.

el selector y defina la segunda señal del mismo modo que se ha explicado en el punto cinco de este ejemplo. En este caso se ha definido como parte real un pulso rectangular de amplitud 2 y como parte imaginaria otro pulso rectangular de amplitud 0,5, ambos con el mismo rango de definición que la señal de entrada.

9. Finalmente pulse el botón **Calculate** para obtener el resultado. Una vez realizado el cálculo y obtenida la señal de salida del sistema verá que en la tercera parte de la ventana se muestra información relacionada con ella, fig. 5.36: el espacio de salida (su nombre y su esquema algebraico, como en la entrada), la norma y la energía de la señal resultado y la distancia y producto escalar entre la señal de entrada y la señal que define al sistema. De nuevo en esta parte se podría escoger un álgebra, que podría ser distinto al del espacio de entrada, si hubiese más de una definición disponible.
10. A continuación abra la ventana de visualización, pulsando en el botón **Visualization** que aparece en la parte inferior izquierda de la ventana, para observar las señales de entrada y salida al sistema, fig 5.37. Como se observa la señal resultado es la multiplicación de la señal de entrada por la señal del sistema. Puede manipular los controles de la ventana para visualizar las señales en otro rango de representación, poner la rejilla, cambiar el modo de representación o visualizar la representación compleja de cada señal a su antojo.
11. Pulse el botón **Visualize $h(x; x')$** para acceder a la ventana de visualización del conjunto de respuestas al impulso, fig 5.38. En este caso $h(x; x') = s_2(x')\delta(x - x')$, es decir, es un tren de deltas sobre la línea de referencia $x = x'$. A partir de dicha representación se pueden estudiar las propiedades del sistema:

: Como se puede observar el sistema no es invariante, pues dicho tren de deltas no toma

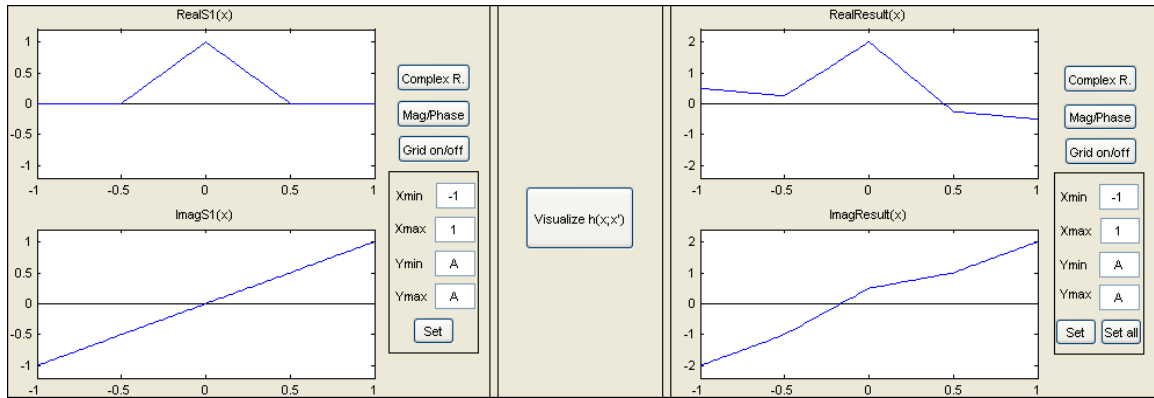


Figura 5.37: Pulsando el botón Visualization del menú principal podrá observar las señales de entrada y salida del operador. Así podrá visualizar las modificaciones que ha sufrido la señal a su paso por el sistema.

el mismo valor en toda la recta a lo largo del espacio. En este caso da la casualidad de que $s_2(x)$ es un pulso rectangular y si se observase sólo en los rangos de x y $x' = [-1 \ 1]$ se podría inducir a un error pensando que el sistema es invariante. Sin embargo, al visualizar en un rango más amplio se observa que no siempre toma el mismo valor, por lo que queda claro que se trata de un sistema no invariante.

- : También se puede observar que el sistema no tiene memoria. Esto significa que la señal de salida en cada valor sólo depende de la señal de entrada en ese valor. Esto se traduce en que el conjunto de respuestas al impulso sea un tren de deltas modulado por la señal que define al operador, como se observa en la gráfica.
 - : Por la misma razón que se observa que el sistema no tiene memoria se observa que no es causal, pues sólo depende del valor de la señal de entrada en cada instante y nunca de la señal de entrada para instantes posteriores.
 - : Por supuesto, también se puede observar que el sistema es lineal ya que existe un conjunto de funciones $h(x; x')$ que caracteriza el operador.
12. También puede estudiar los cortes del conjunto de funciones $h(x; x')$ y apreciar con más claridad que los valores no nulos de las gráficas en tres dimensiones son deltas de Dirac.
 13. Cierre la ventana de visualización del conjunto de respuestas al impulso con el botón Close para visualizar, a continuación, el conjunto de señales que le corresponde en el dominio transformado. Para ello haga clic sobre el botón denominado Visualize $H(w; wt)$ de la ventana de visualización principal. Como observa en la fig. 5.39, esta ventana se subdivide a su vez en tres ventanas. Las dos superiores se utilizan para representar la parte real/módulo y la parte imaginaria/fase del conjunto de funciones núcleo que caracterizan al operador multiplicación definido en el dominio transformado. A su vez, la parte inferior de la ventana se utiliza para visualizar los distintos cortes de estas superficies.
 14. Por último, cierre las ventanas de visualización y acceda de nuevo al menú principal. Si observa el cuadro de opciones del operador, Options, verá que el botón de animación está activado. Esto significa que puede ver paso a paso como se calcula la señal de salida. Pulse sobre el botón para abrir la ventana de animación y cuando esté abierta pulse el botón Play para comenzar su ejecución. En la ventana, fig 5.40, visualizará cada paso del proceso y verá como se relaciona el conjunto de respuestas al impulso con la señal de entrada para producir un resultado- En la parte derecha se visualizará de forma estática la curva resultante de la multiplicación de la señal de entrada y el conjunto de respuestas al impulso. Según se vaya

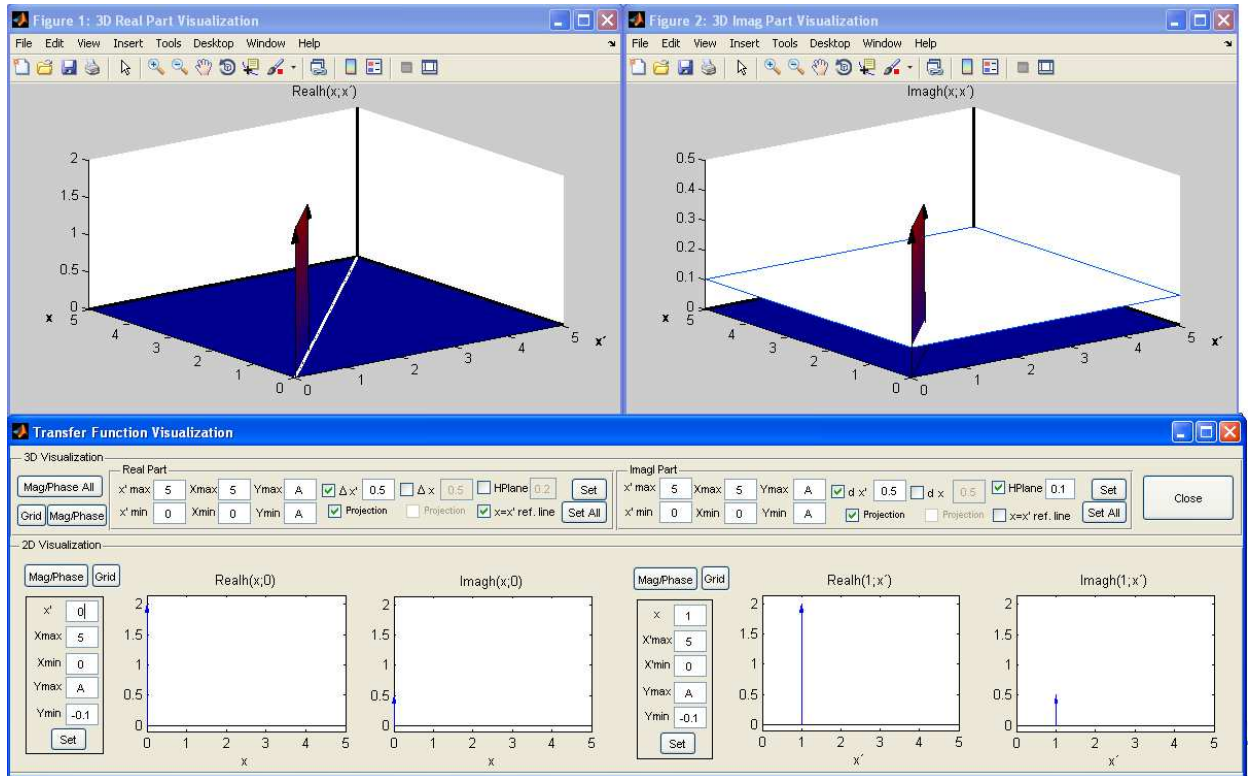


Figura 5.38: Cuando visualice el conjunto de respuestas al impulso podrá observar y analizar las propiedades que tiene el operador.

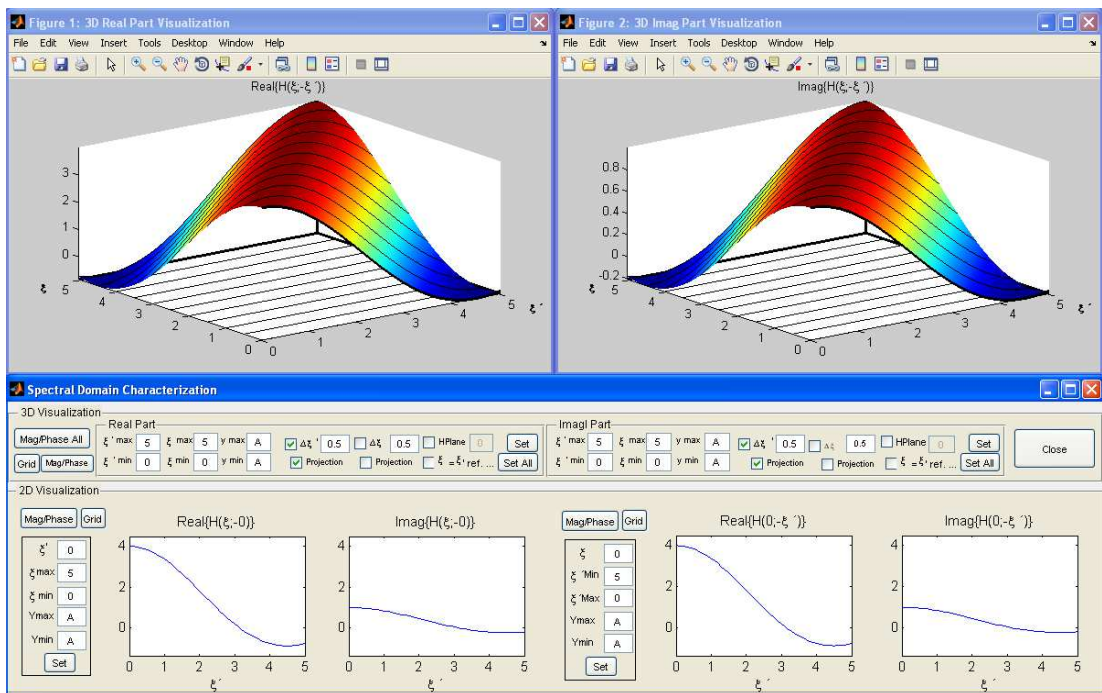


Figura 5.39: Ventana del conjunto de funciones núcleo que caracterizan el operador multiplicación definido en el eje p en el dominio espectral.

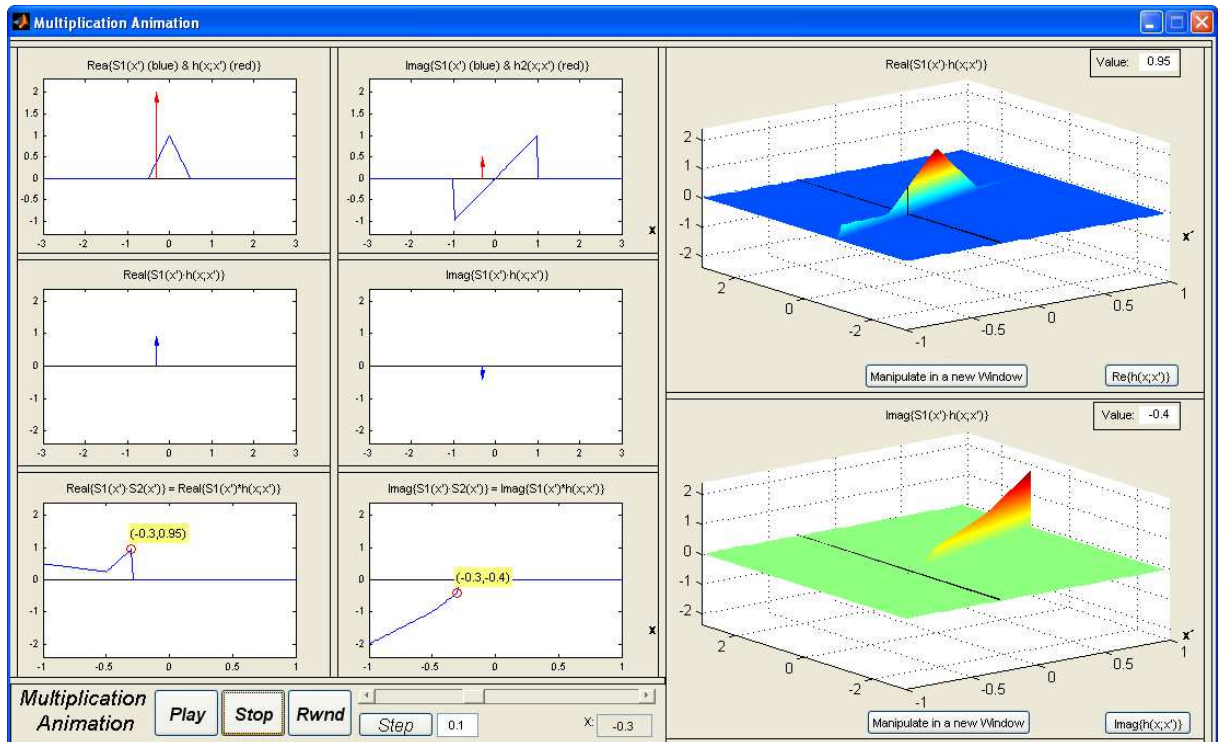


Figura 5.40: En la ventana de animación podrá visualizar como se calcula la señal resultado paso a paso, observando la relación que sufre la señal de entrada con el conjunto de respuestas al impulso.

avanzando la simulación, para cada valor de x , se verá la función $h(x; x') = k\delta(x - x')$, donde k representa la amplitud de la delta y será el valor que tome la señal resultado para el valor de x considerado.

5.4.2. Análisis del Operador Convolución para el Espacio Continuo Aperiódico.

En este ejemplo se describirán los pasos que deberá seguir para estudiar el comportamiento del operador convolución cuando el espacio de partida es el de señales de variable continua aperiódicas.

1. Arranque el programa y siga los pasos 1 a 5 detallados en 5.4.1 para definir una señal de entrada, para facilitar las explicaciones se definirá la misma señal de entrada.
2. Seleccione el operador convolución para definir el sistema. Para ello asegúrese de que los botones circulares de la caja System Definition tengan la opción Operator seleccionada y seleccione la opción Convolution del menú desplegable denominado Select Operator. Como se muestra en la fig 5.41, automáticamente se modificará el esquema del operador y aparecerá un texto explicativo bajo la imagen. En este caso el texto indica que:
 - : El operador seleccionado convolucionara la señal de entrada por la señal que define al sistema.
 - : El operador convolución es, en realidad, una familia de operadores.
 - : Una vez se haya definido la señal del sistema el operador quedará completamente definido, ya no será una familia de operadores sino uno en concreto.

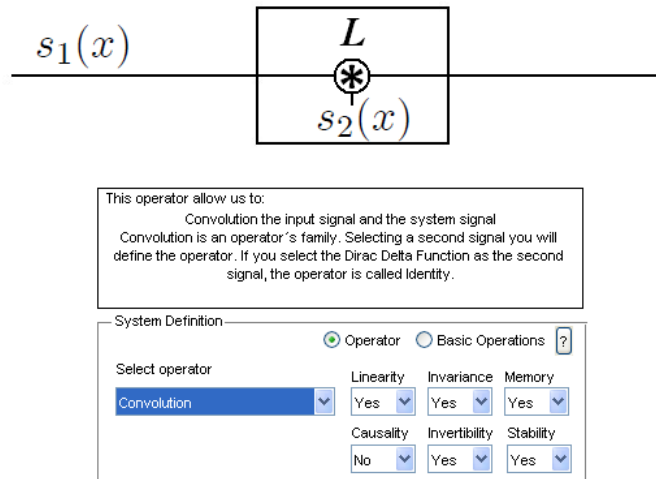


Figura 5.41: La caja que representa el sistema cambiará para mostrar un esquema que represente al operador seleccionado. Además se mostrará una breve descripción del operador y las propiedades que tiene.

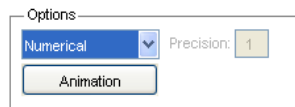


Figura 5.42: Panel de opciones del operador convolución.

3. Además de la información citada anteriormente, el programa le mostrará cuales son propiedades del operador. Resultaría muy conveniente y educativo que realizase una parada en este momento para asimilar cuales son las propiedades del operado y así después intentar reconocerlas cuando visualice el conjunto de respuestas al impulso que caracteriza al operador. El operador de convolución es lineal, invariante, con memoria, no causal, invertible y estable.
4. Dado que el operador convolución representa a una familia de operadores deberá definir la señal interna al sistema (aquella con la que se convolucionará a cualquier señal de entrada y que representa, a su vez, el conjunto de respuestas al impulso). De nuevo, para facilitar las explicaciones, se definirá la misma señal que en 5.4.1.
5. Observe el panel **Options** y vea las opciones disponibles, fig 5.42. Dado que las dos señales definidas son numéricas el software sólo dispone de un método de cálculo numérico. Cuando las señales sean teóricas podrá seleccionarse un modo de cálculo teórico.
6. Finalmente pulse el botón **Calculate** para obtener el resultado. Una vez realizado el cálculo y obtenida la señal de salida del sistema verá que en la tercera parte de la ventana se muestra información relacionada con ella, fig. 5.43: el espacio de salida (su nombre y su esquema algebraico, como en la entrada), la norma y la energía de la señal resultado y la distancia y producto escalar entre la señal de entrada y la señal que define al sistema. De nuevo en esta parte se podría escoger un álgebra, que podría ser distinto al del espacio de entrada, si hubiese más de una definición disponible.
7. En este ejemplo se va a estudiar la funcionalidad del análisis espectral del sistema. Para ello, seleccione en el menú desplegable que se encuentra en la esquina inferior izquierda de la ventana una de las opciones de análisis: Fourier Transform o Hilbert Transform. En este ejemplo, véase la fig 5.44, se va a realizar el análisis espectral bajo la transformada de

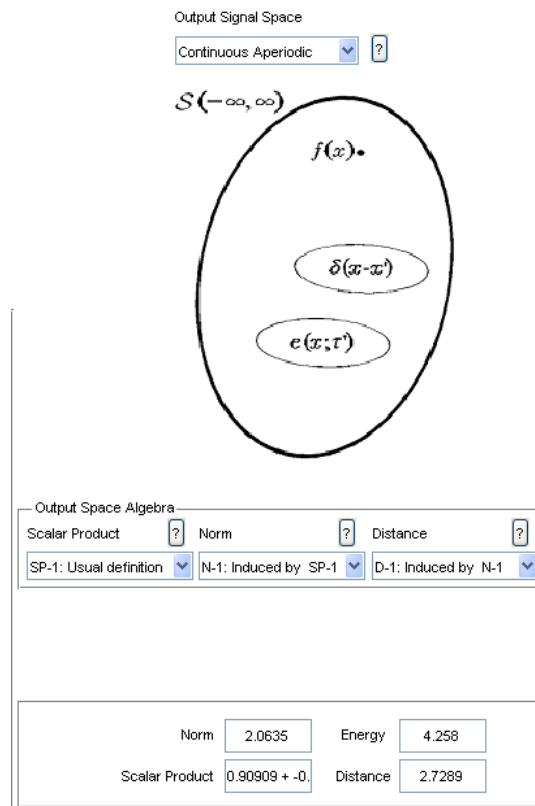


Figura 5.43: La salida del sistema se modificará cuando se haya calculado el resultado: se mostrará el espacio de salida y los valores numéricos asociados a ella.

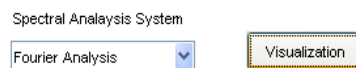


Figura 5.44: Seleccione la transformada bajo la que quiere analizar el sistema espectralmente utilizando el menú desplegable denominado Spectral Analysis System.

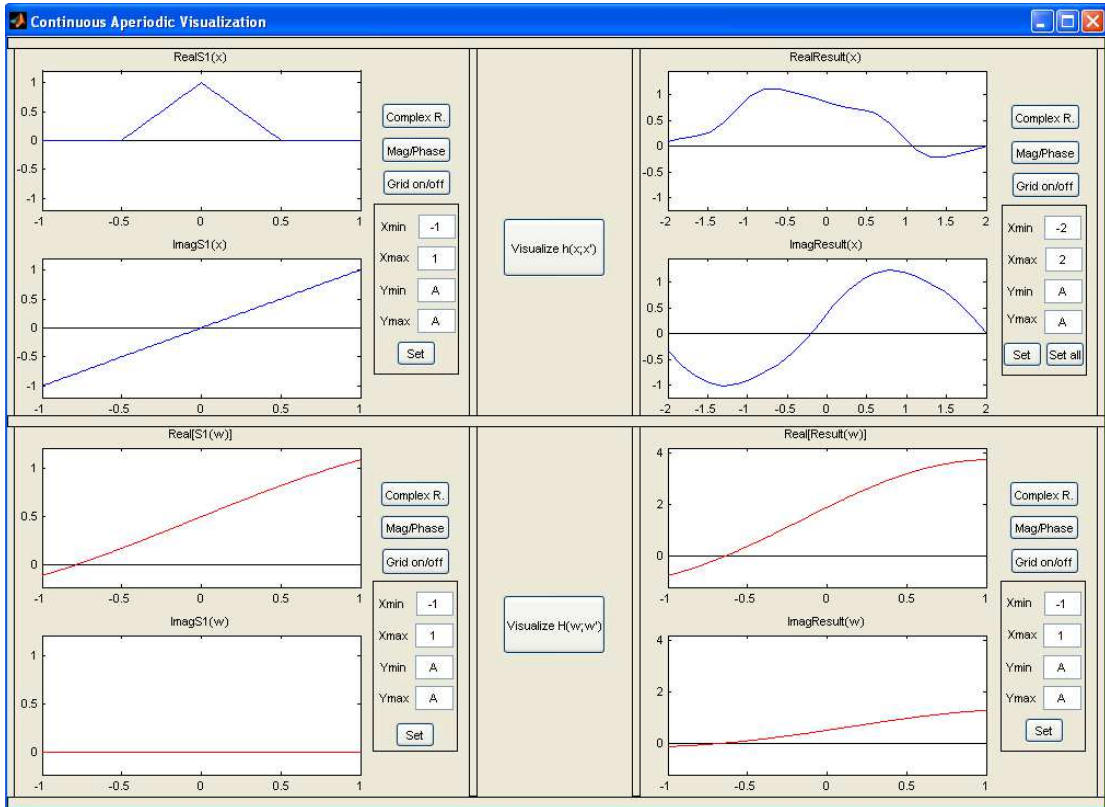


Figura 5.45: Pulsando el botón **Visualization** del menú principal podrá observar las señales de entrada y salida del operador. Así podrá visualizar las modificaciones que ha sufrido la señal a su paso por el sistema.

Fourier.

8. A continuación abra la ventana de visualización, pulsando en el botón **Visualization** que aparece en la junto al menú del análisis espectral, para observar las señales de entrada y salida al sistema, fig 5.45. En las gráficas superiores podrá visualizar la representación de ambas señales en el dominio real, dibujadas en azul, mientras que en las gráficas inferiores verá las señales representadas en el dominio transformado, en color rojo. La señal resultado es la convolución de la señal de entrada por la señal del sistema. Puede manipular los controles de la ventana para visualizar las señales en otro rango de representación, poner la rejilla, cambiar el modo de representación o visualizar la representación compleja de cada señal a su antojo.
9. Para analizar el sistema pulse el botón **Visualize $h(x; x')$** y accederá a la ventana de visualización del conjunto de respuestas al impulso, fig 5.46. En este caso $h(x; x') = h(x) = s_2(x)$, es decir, el conjunto de respuestas al impulso está compuesta por la misma señal a lo largo de x centrada en la recta $x = x'$. A partir de dicha representación se pueden estudiar las propiedades del sistema:

- : Como se puede observar el sistema es invariante, pues, como ya se ha dicho, el conjunto de respuestas al impulso está compuesto por una señal centrada en la recta $x = x'$ y, por tanto, se puede caracterizar como una sola señal. Además, al tratarse de la operación de convolución la respuesta al impulso es la propia señal que define el operador $h(x) = s_2(x)$.
- : También se puede observar que el sistema tiene memoria, esto es, la señal de salida

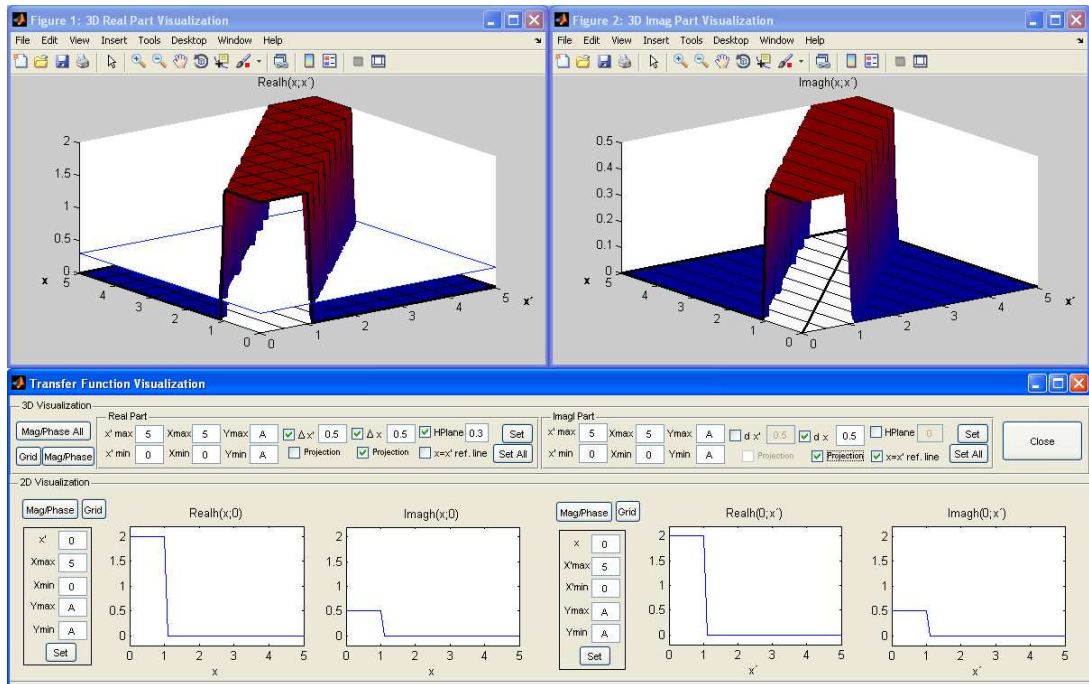


Figura 5.46: Cuando visualice el conjunto de respuestas al impulso podrá observar y analizar las propiedades que tiene el operador. En este caso se analiza el operador convolución que es lineal, invariante, con memoria y no causal.

para cada instante depende de valores de la señal de entrada en instantes posteriores. Resulta evidente observarlo ya que, en caso contrario, el conjunto de respuestas al impulso debería estar constituido por un tren de deltas de Dirac y en este caso se trata de una superficie.

- : En esta representación se observa que el sistema no es causal, esto quiere decir que la señal para un instante no depende únicamente de valores de la señal de entrada anteriores a dicho instante. Si lo fuese, el conjunto de respuestas al impulso $h(x; x')$ tomaría valores nulos para valores de x menores de un valor concreto x'_0 de la variable x' . Es decir, si visualizásemos la recta $x = x'_0$ hacia un lado de ella toda la superficie valdría cero. Es posible que el caso representado induzca a confusión por el hecho de haber escogido un pulso rectangular como conjunto de respuestas al impulso. Sin embargo, si define una señal de energía infinita podrá visualizarlo sin problemas.
 - : Por supuesto, también se puede observar que el sistema es lineal ya que existe un conjunto de funciones $h(x; x')$ que caracteriza el operador.
10. Cierre la ventana de visualización del conjunto de respuestas al impulso con el botón Close para visualizar, a continuación, la el conjunto de funciones que le corresponden al operador seleccionado en el dominio transformado. Para ello haga clic sobre el botón denominado Visualize $H(w; w)$ de la ventana de visualización principal. En la fig. 5.47 se puede observar el conjunto de funciones núcleo que representan al operador convolución definido en el dominio espectral. De esta representación cabe resaltar algunos aspectos:

- : El operador estudiado es lineal e invariante y, por tanto, la señal de salida del sistema en el dominio espectral se puede obtener como la multiplicación de la transformada de la señal de entrada por la transformada de la función de transferencia que caracteriza al operador en el dominio transformado, esto es, $S_{out}(\xi) = S_{in}(\xi)H(\xi)$. Por tanto, es lógico que el conjunto de funciones núcleo que caracterizan el operador en el dominio

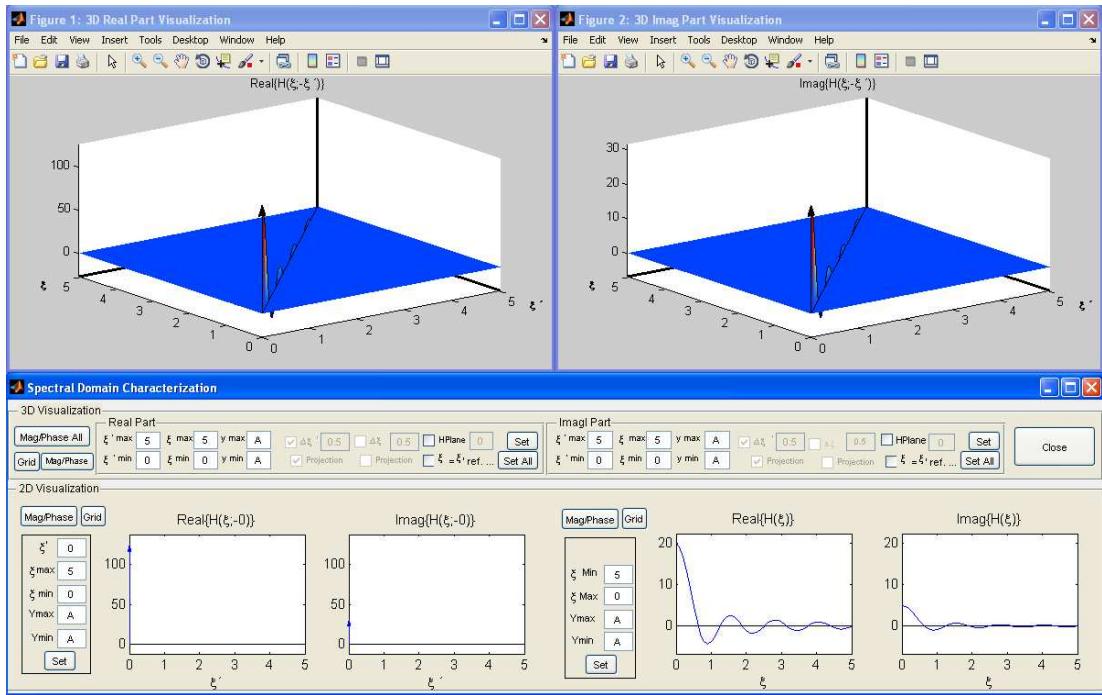


Figura 5.47: Conjunto de funciones núcleo que caracterizan al operador convolución definido por el usuario en el dominio espectral, bajo la transformada de Fourier.

de la frecuencia tengan la misma forma el conjunto de respuestas al impulso del operador multiplicación, es decir, un tren de deltas en $\xi = \xi'$ modulado por la función de transferencia del sistema.

- : Dadas las características de este conjunto de funciones núcleo, un corte en $\xi = \xi_0$ equivaldrá a un corte en $\xi' = \xi_0$. Aprovechando esta particularidad, se utilizará una de las dos representaciones en dos dimensiones para representar la función de transferencia del sistema $H(\xi)$, dejando sólo la primera representación en dos dimensiones para tomar un corte de la representación en tres dimensiones.

11. Dadas las características del sistema, se trata de un operador lineal e invariante, se podrá analizar éste en el dominio transformado mediante las representaciones en dos dimensiones. Si se estudian conjuntamente la señales de entrada y salida en el dominio transformado mostradas en la ventana de visualización y la función de transferencia de la ventana del análisis espectral generalizado se puede comprobar que la señal de salida se obtiene como la multiplicación punto a punto de las señales de entrada y del sistema, véase la fig. 5.48.
12. Por último se mostrará la funcionalidad de animación para esta operación. Cierre todas las ventanas de visualización que estén abiertas y acceda al menú principal. Localice el botón de animación que se encuentra en el panel de opciones del operador y pulse sobre él. Al hacerlo, accederá a la ventana de animación. Sus funcionalidades y su estructura es idéntica a la de la ventana de animación explicada en el ejemplo anterior (5.4.1). La diferencia principal entre los dos casos, véase la fig. 5.49, es que al tratarse de un sistema con memoria el producto entre la señal de entrada y el conjunto de respuestas al impulso ya no es un tren de deltas sino una superficie. Por lo tanto, el resultado ya no vendrá determinado por el valor de la delta en cada instante sino por la integral del área que encierre la función considerada en cada instante.

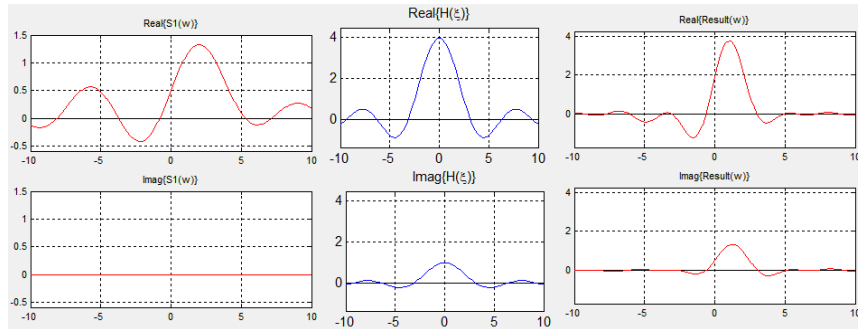


Figura 5.48: En esta figura se representan, en el dominio transformado, la señal de entrada al sistema, la función de transferencia y la señal de salida del sistema. Se puede comprobar que la señal de salida no es más que la multiplicación punto a punto de las dos primeras señales, esto es, $S_{out}(\xi) = S_{in}(\xi)H(\xi)$.

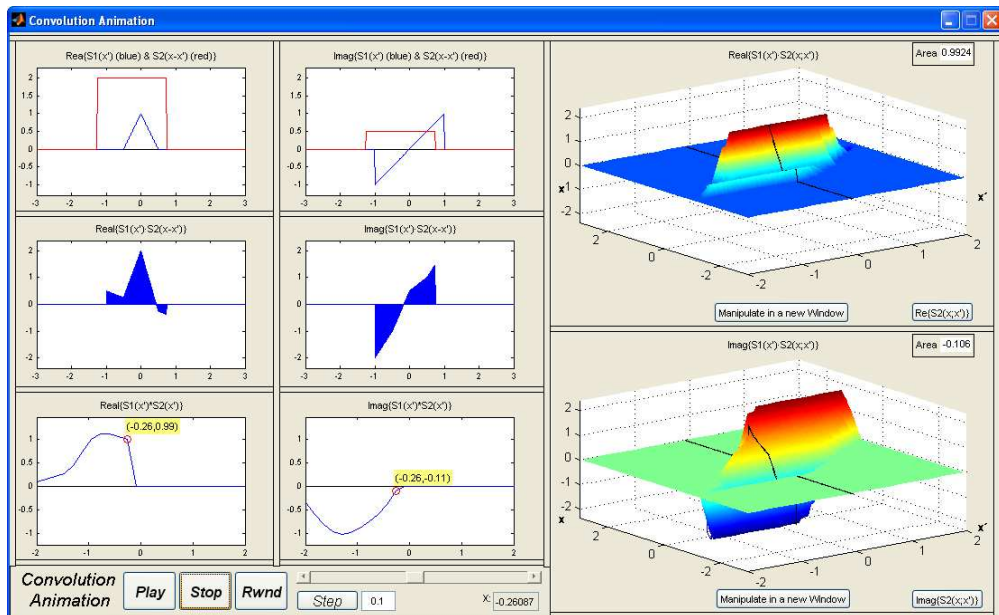


Figura 5.49: En la ventana de animación podrá visualizar como se calcula la señal resultado paso a paso, observando la relación que sufre la señal de entrada con el conjunto de respuestas al impulso.

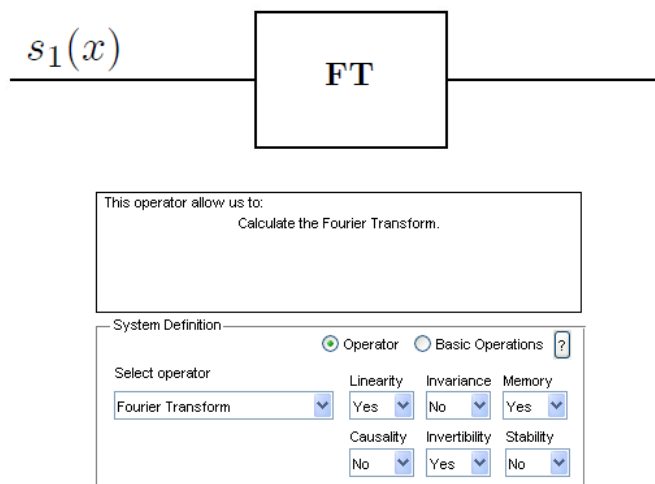


Figura 5.50: Operador Transformada de Fourier. La caja que representa el sistema cambiará para mostrar un esquema que represente al operador seleccionado. Además se mostrará una breve descripción del operador y las propiedades que tiene.

5.4.3. Análisis del Operador Transformada de Fourier para el Espacio Continuo Aperiódico.

En este ejemplo se describirán los pasos que deberá seguir para estudiar el comportamiento del operador transformada de Fourier cuando el espacio de partida es el de señales de variable continua aperiódicas.

1. Arranque el programa y siga los pasos 1 a 5 detallados en 5.4.1 para definir una señal de entrada, para facilitar las explicaciones se definirá la misma señal de entrada que en dicho ejemplo.
2. Seleccione el operador transformada de Fourier para definir el sistema. Para ello asegúrese de que los botones circulares de la caja System Definition tengan la opción Operator seleccionada y seleccione la opción Fourier Transform del menú desplegable denominado Select Operator. Como se muestra en la fig 5.50, automáticamente se modificará el esquema del operador y aparecerá un texto explicativo bajo la imagen. En este caso el texto indica que el operador escogido calculará la transformada de fourier sobre la señal de entrada.
3. Adicionalmente, el programa le mostrará cuales son propiedades del operador. Resultaría muy conveniente y educativo que realizase una parada en este momento para asimilar cuales son las propiedades del operador y así después intente reconocerlas cuando visualice el conjunto de respuestas al impulso que caracteriza al operador. El operador transformada de Fourier es lineal, no invariante, con memoria, no causal, invertible e inestable.
4. Al contrario que en los ejemplos anteriores, el operador transformada de Fourier no pertenece a una familia y está completamente definido por su nombre, por lo tanto no será necesario definir la una señal que le caracterice o asignar los parámetros a la señal característica que tenga. Por lo tanto el siguiente paso es escoger la opción más adecuada de las disponibles en el panel Options, mostrado junto al botón Calculate. En este caso, al haber definido como señal de entrada una señal numérica, el programa le ofrecerá dos métodos de cálculo, véase la fig. 5.51, el método basado en Cuadratura de Newton y el basado en la Fast Fourier Transform. Junto al menú desplegable que muestra estas opciones hay un cuadro de texto, denominado Precision, en el que podrá definir el nivel de precisión

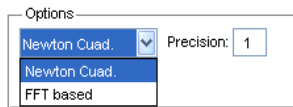


Figura 5.51: Apariencia del panel de opciones cuando el operador seleccionado es la transformada de Fourier. Este panel le da a elegir entre distintos métodos de cálculo y la precisión con la que quiera que se realicen.

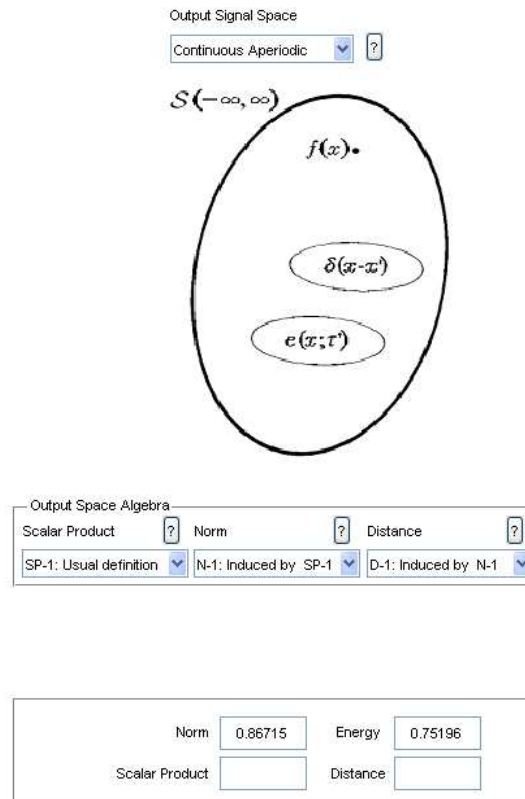


Figura 5.52: Salida del sistema para el operador Transformada de Fourier cuando el espacio de la señal de entrada es continuo Aperiódico. Dado que el operador no precisa de la definición de una señal que lo caracterice, el software no calculará ni la distancia ni el producto escalar.

con el que se realizarán los cálculos. Para este ejemplo se utilizará el método basado en Cuadratura de Newton con una precisión de valor 1.

5. Finalmente, una vez definidas todas las opciones, pulse el botón **Calculate** para obtener el resultado. Como en los ejemplos anteriores, la parte de la ventana correspondiente a la salida del sistema cambiará para mostrar el espacio de salida y el resultado de los cálculos numéricos realizados, fig. 5.52. En este caso, al no haber tenido que definir una segunda señal, no se calcularán ni la distancia ni el producto escalar.
6. A continuación abra la ventana de visualización, pulsando en el botón **Visualization** que aparece en la parte inferior izquierda de la ventana, fig 5.53. En las gráficas correspondientes podrá visualizar las señales de entrada y salida al sistema que, en este caso, son la señal en el dominio real y la señal en el dominio transformado. Puede manipular los controles de la ventana para visualizar las señales en otro rango de representación, poner la rejilla, cambiar el modo de representación o visualizar la representación compleja de cada señal a su antojo.

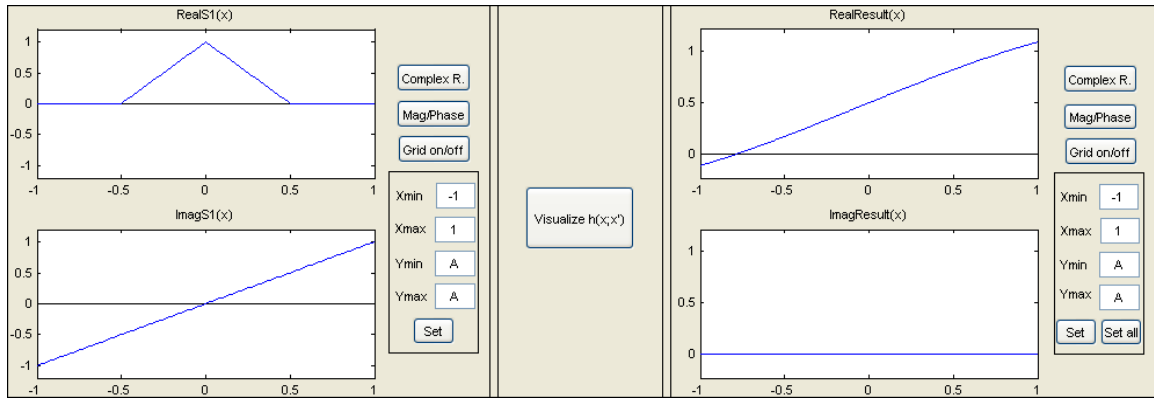


Figura 5.53: Pulsando el botón **Visualization** del menú principal podrá observar las señales de entrada y salida del operador. Así podrá visualizar las modificaciones que ha sufrido la señal a su paso por el sistema.

7. Por último se visualizará el conjunto de respuestas al impulso pulsando el botón **Visualize $h(x; x')$** , fig 5.54. Como el sistema es no invariante el conjunto de respuestas al impulso no puede reducirse a una sola señal. A partir de la representación observada podrá analizar las propiedades del sistema:

- : Como se puede observar el sistema no es invariante, pues cada una de las señales que forman el conjunto de respuestas al impulso es diferente. Además de en la gráfica en tres dimensiones puede observarlo visualizando distintos cortes de la superficie en las gráficas inferiores.
- : Siguiendo el mismo razonamiento expuesto en el ejemplo anterior se puede observar que el sistema tiene memoria y no es causal. En este ejemplo puede apreciarse con más claridad la propiedad de no causalidad.
- : Por supuesto, también se puede observar que el sistema es lineal ya que existe un conjunto de funciones $h(x; x')$ que caracteriza el operador.

5.4.4. Análisis del Operador Modulador en Amplitud para el Espacio Continuo Periódico.

1. Ejecute como se indica en 5.4.1 seleccionando esta vez como espacio de partida el **Continuous Periodic**. Automáticamente se cargarán en la pantalla los controles para avanzar.
2. A continuación se deberá definir una señal de entrada al sistema. En este caso se definirá una señal numérica, por lo tanto pulse sobre el botón **Select** del tipo de señales **Numerical Distributions**.
3. Una vez haya pulsado el botón se abrirá la ventana para seleccionar la señal, véase fig. 5.55. Ahora deberá definir la señal de entrada siguiendo los pasos descritos en 5.4.1. La señal definida para el ejemplo estará formada por un pulso triangular de amplitud 1, ancho 1 y desplazamiento 0 para la parte real y un pulso rectangular de amplitud 2 para la parte imaginaria. A continuación, en la ventana de definición del rango, fig. 5.56, se definirá un rango de $\Delta x = 1$ y $x_0 = 0$ para la parte real y de $\Delta x = 0,5$ y $x_0 = 0$ para la parte imaginaria.
4. A continuación se definirá el sistema a través del operador modulador en amplitud. Para ello asegúrese de que los botones circulares de la caja **System Definition** tengan la opción **Operator** seleccionada y seleccione la opción **Amplitude Modulator** del menú desplegable

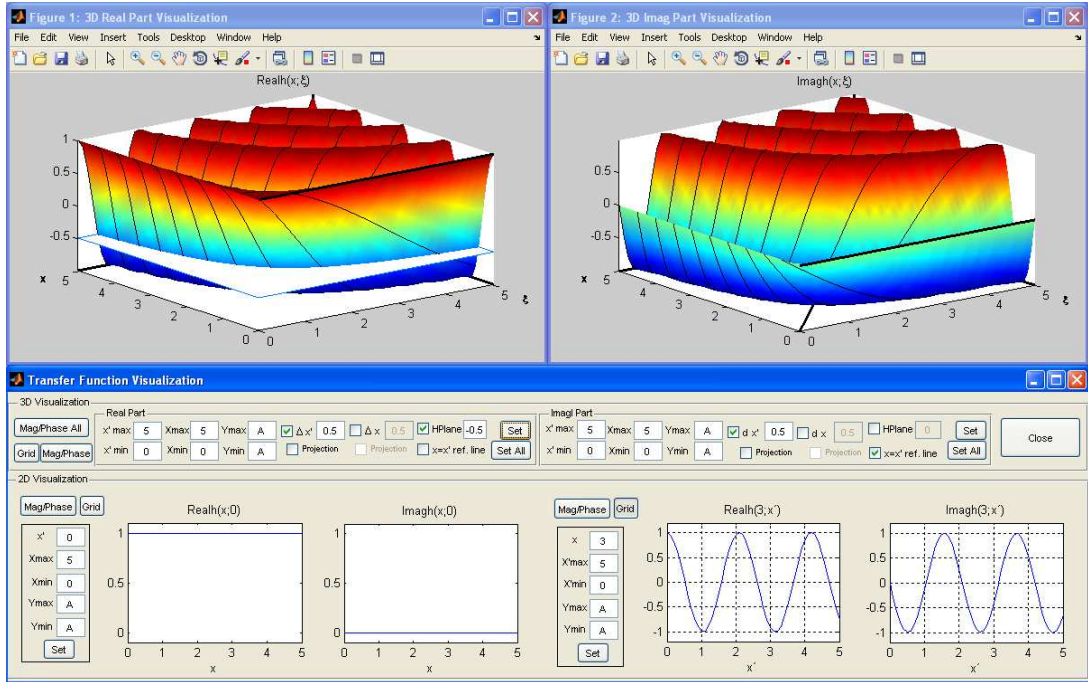


Figura 5.54: Cuando visualice el conjunto de respuestas al impulso podrá observar y analizar las propiedades que tiene el operador. En este caso se analiza el operador transformada de Fourier que es lineal, no invariante, con memoria y no causal.

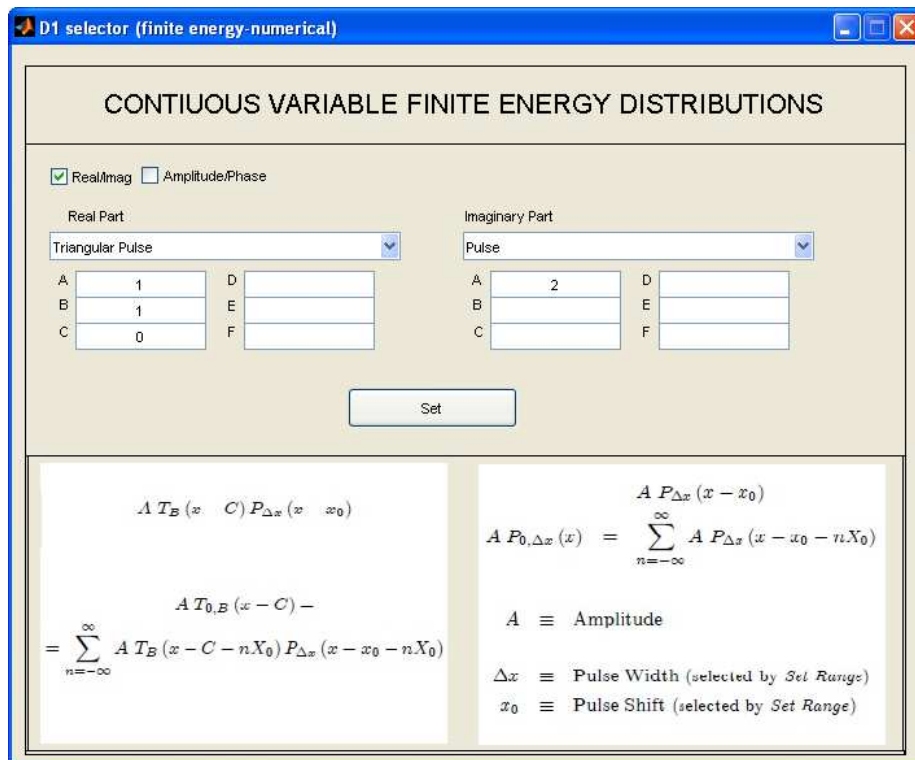


Figura 5.55: Ventana del selector de señales con las señales seleccionadas para definir la señal de entrada al sistema.

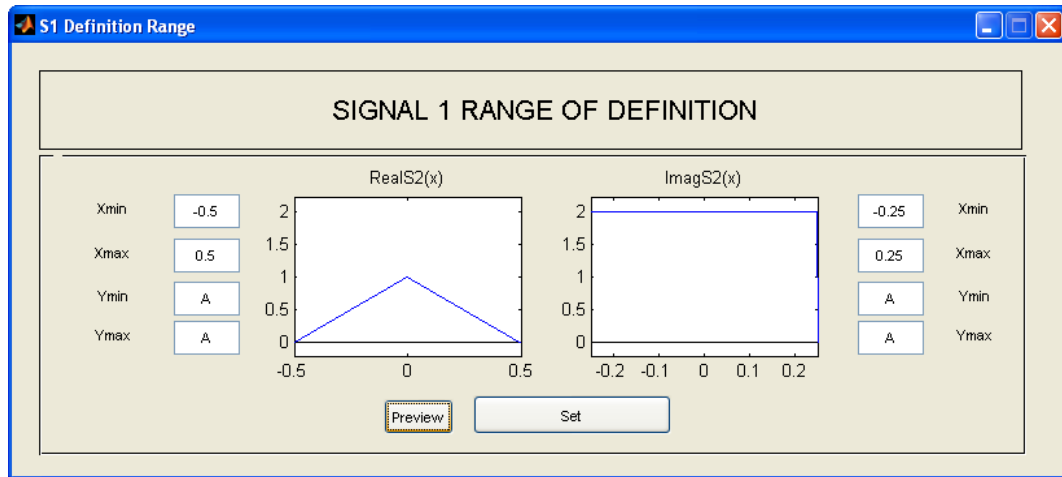


Figura 5.56: Ventana para seleccionar el rango de definición de la señal de entrada. En realidad, este rango define el ancho y desplazamiento de los pulsos rectangulares, que componen el tren de pulsos, que multiplican la señal de entrada.

denominado **Select Operator**. Como se muestra en la fig 5.57, automáticamente se modificará el esquema del operador y aparecerá un texto explicativo bajo la imagen. En este caso el texto indica que:

- : El operador seleccionado multiplicará la señal de entrada por un coseno.
 - : El operador modulador en amplitud pertenece a la familia de operadores de la multiplicación. Con esto se entiende que si, en lugar de seleccionar el operador modulador en amplitud, se seleccionase el operador multiplicación con un coseno como señal característica se obtendría el mismo resultado.
5. Además de la información citada anteriormente, el programa le mostrará cuales son propiedades del operador. Resultaría muy conveniente y educativo que realizase una parada en este momento para asimilar cuales son las propiedades del operado y así después intentar reconocerlas cuando visualice el conjunto de respuestas al impulso que caracteriza al operador. El operador modulador en amplitud, como se muestra en la fig 5.57, es un operador lineal, no invariante, sin memoria y causal. Si consulta las propiedades del operador multiplicación, descritas en 5.4.1, observará que son las mismas. Este hecho es lógico si se tiene en cuenta que el modulador en amplitud es un operador que pertenece a la familia de operadores multiplicación.
 6. Si bien es cierto que el nombre modulador en amplitud ya implica que la señal que multiplique a la señal de entrada sea un coseno, el software permite al usuario definir los parámetros de dicha función. De este modo, el operador modulador en amplitud, se puede considerar como una nueva familia de operadores contenida en la familia multiplicación. Para ajustar los parámetros del coseno pulse sobre el botón **Select** junto al texto **Parameters Selector**, fig. 5.58, para abrir el selector de parámetros. Cuando lo abra, véase la fig. 5.59, verá que ya contiene unos valores por defecto para los parámetros a definir. En este ejemplo se utilizarán dichos valores, por tanto, pulse el botón **Set** para cerrar el selector.
 7. Una vez haya realizado todos los pasos previos, pulse el botón **Calculate** para obtener el resultado. Una vez realizado el cálculo y obtenida la señal de salida del sistema verá que en la tercera parte de la ventana se muestra información relacionada con ella, fig. 5.60: el espacio de salida (su nombre y su esquema algebraico, como en la entrada), la norma y la energía de la señal resultado y la distancia y producto escalar entre la señal de entrada

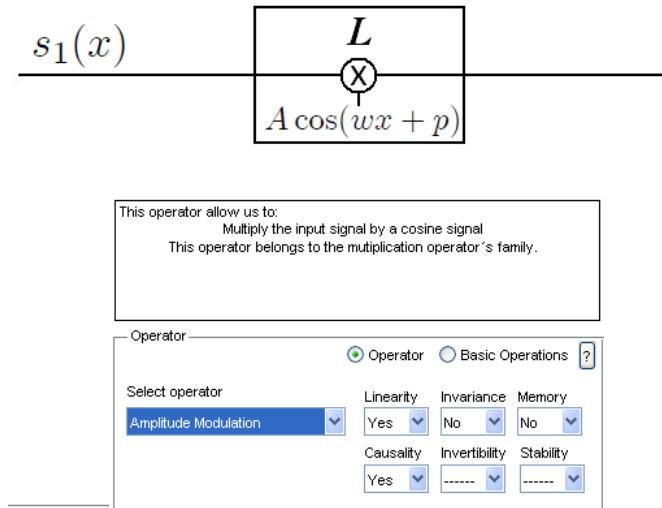


Figura 5.57: La caja que representa el sistema cambiará para mostrar un esquema que represente al operador seleccionado. Además se mostrará una breve descripción del operador y las propiedades que tiene.

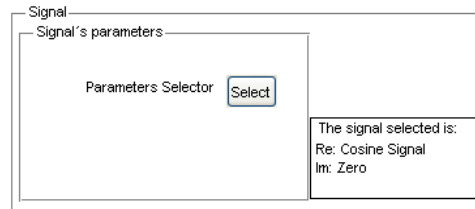


Figura 5.58: Pulse sobre el botón Select para definir los parámetros del coseno.

SIGNAL PARAMETER SELECTOR

Real Part: Cosine

Imaginary Part: Zero

A: 1	D: 0	A:	D:
B: 2*pi	E:	B:	E:
C: 0	F:	C:	F:

$$s(x) = \begin{cases} A \cos[B(x - C)], & \forall D - \frac{X_0}{2} \leq x < D + \frac{X_0}{2} \\ 0, & \text{otherwise} \end{cases}$$

$$s_0(x) = \sum_{n=-\infty}^{\infty} s(x - nX_0)$$

Where A, B, C and D can be vectors of the same length

NO HELP AVAILABLE

Figura 5.59: El selector de parámetros se utilizará para definir las características de la señal que describe al operador, en este caso un coseno.

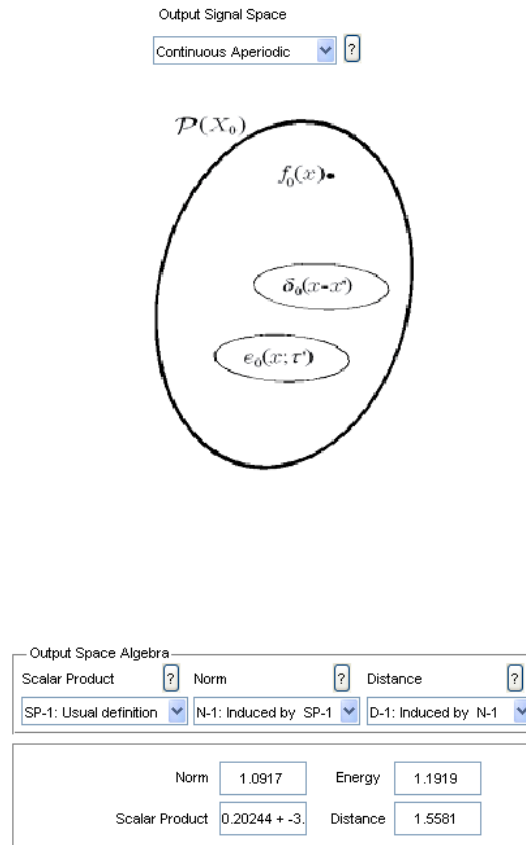


Figura 5.60: La salida del sistema se modificará cuando se haya calculado el resultado: se mostrará el espacio de salida y los valores numéricos asociados a ella.

y la señal que define al sistema (en este caso se trata de un coseno cuyos parámetros ha definido el usuario).

8. A continuación abra la ventana de visualización, pulsando en el botón **Visualization** que aparece en la parte inferior izquierda de la ventana, para observar las señales de entrada y salida al sistema, fig 5.61. Como se observa la señal resultado es la multiplicación de la señal de entrada por el coseno, lo que es conocido como una modulación en amplitud. Puede manipular los controles de la ventana para visualizar las señales en otro rango de representación, poner la rejilla, cambiar el modo de representación o visualizar la representación compleja de cada señal a su antojo.
9. Pulse el botón **Visualize $h(x; x')$** para acceder a la ventana de visualización del conjunto de respuestas al impulso, fig 5.62. En este caso $h(x; x') = A \cos(wx' + p)\delta(x - x')$, es decir, es un tren de deltas modulado por un coseno y descrito sobre la línea de referencia $x = x'$. Del mismo modo que se explica en 5.4.1, esta representación le permitirá observar las propiedades del sistema.

5.4.5. Análisis de la Operación Suma en el espacio Discreto Aperiódico.

En este ejemplo se analizará una de las operaciones básicas disponibles, la operación suma de dos señales, cuando el espacio de definición de las señales es el discreto aperiódico.

1. Ejecute como se indica en 5.4.1 seleccionando esta vez como espacio de partida el **Discrete Aperiodic**. Automáticamente se cargarán en la pantalla los controles para avanzar. Como

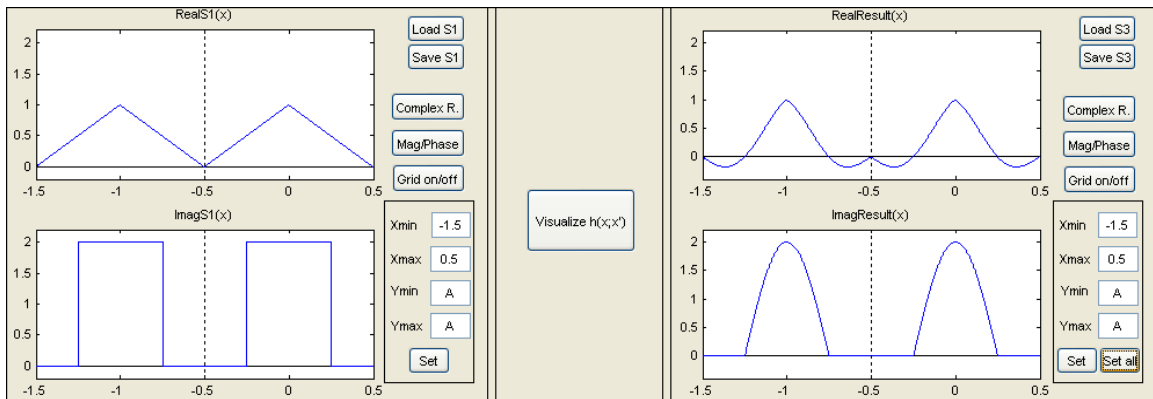


Figura 5.61: Pulsando el botón Visualization del menú principal podrá observar las señales de entrada y salida del operador. Así podrá visualizar las modificaciones que ha sufrido la señal a su paso por el sistema.

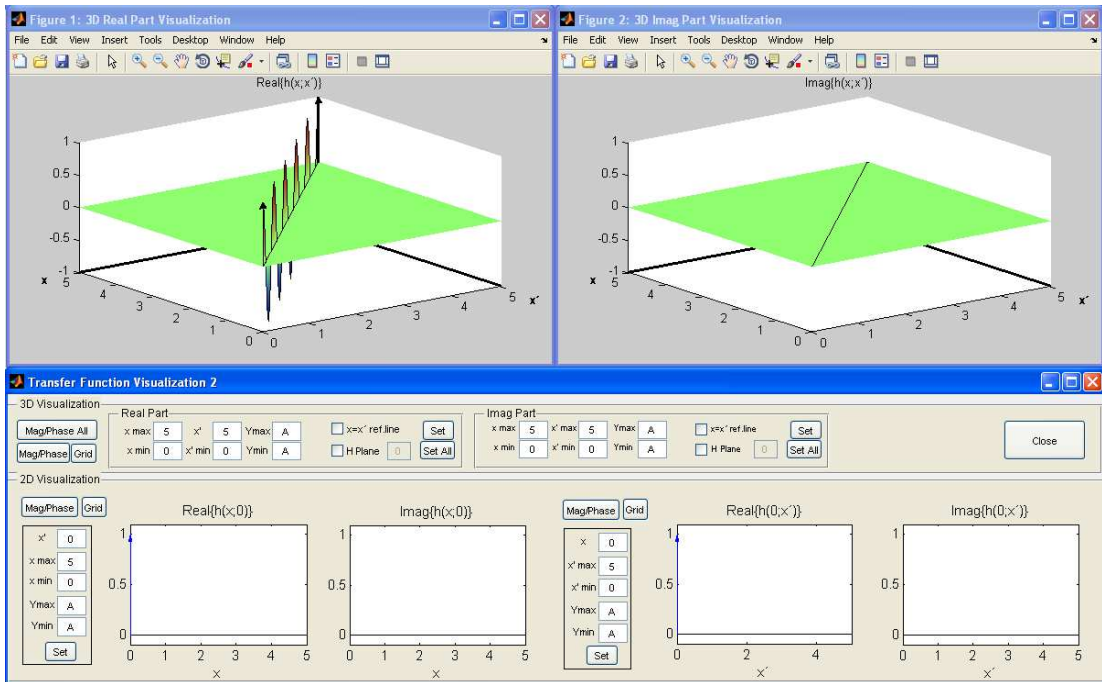


Figura 5.62: Cuando visualice el conjunto de respuestas al impulso podrá observar y analizar las propiedades que tiene el operador.

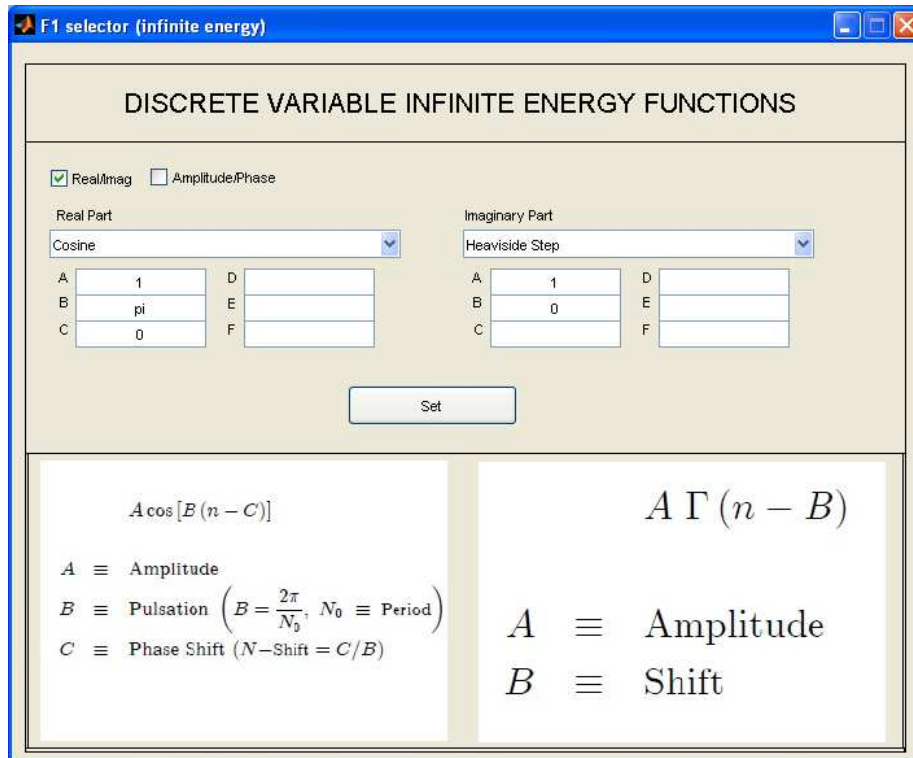


Figura 5.63: Ventana del selector de señales con las señales seleccionadas para definir la primera señal de entrada al sistema.

podrá observar, el aspecto de la pantalla es prácticamente el mismo que el mostrado en los casos anteriores.

2. A continuación se deberá definir una señal de entrada al sistema. En este caso se definirá una señal de energía infinita, por lo tanto pulse sobre el botón Select del tipo de señales Functions del panel Infinite Energy.
3. Una vez haya pulsado el botón se abrirá la ventana para seleccionar la señal, véase fig. 5.63. Ahora deberá definir la señal de entrada siguiendo los pasos descritos en 5.4.1. La señal definida para el ejemplo estará formada por un coseno de amplitud 1, pulsación π y desplazamiento 0 para la parte real y el escalón de Heaviside de amplitud 1 para la parte imaginaria.
4. A continuación se definirá el sistema a través de la operación básica suma. Para ello asegúrese de que los botones circulares de la caja System Definition tengan la opción Operation seleccionada y seleccione la opción Addition del menú desplegable denominado Select Operation. Como se muestra en la fig 5.64, automáticamente se modificará el esquema del sistema y aparecerá un texto explicativo bajo la imagen. En este caso el texto indica que la operación seleccionada sumará las dos señales definidas.
5. Dado que lo que se está estudiando es una operación básica se debe definir una segunda señal de entrada al sistema, que en este caso será sumada a la señal definida previamente. En este ejemplo se definirá como segunda señal una función de energía infinita definida como: la señal signo de amplitud 1 y desplazamiento 0 para la parte real y un coseno de amplitud 1, pulsación π y desfase 0. Si bien es cierto que el nombre modulador en amplitud ya implica que la señal que multiplique a la señal de entrada sea un coseno, el software permite al usuario definir los parámetros de dicha función, véase la fig 5.65.

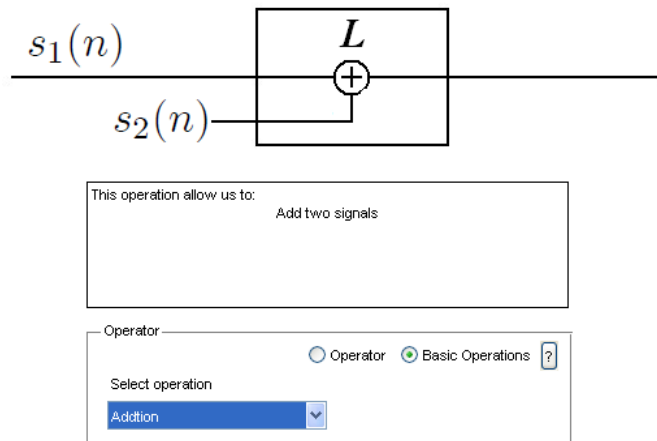


Figura 5.64: La caja que representa el sistema cambiará para mostrar un esquema que represente la operación seleccionada. Además, indicará textualmente lo que hace la operación seleccionada.

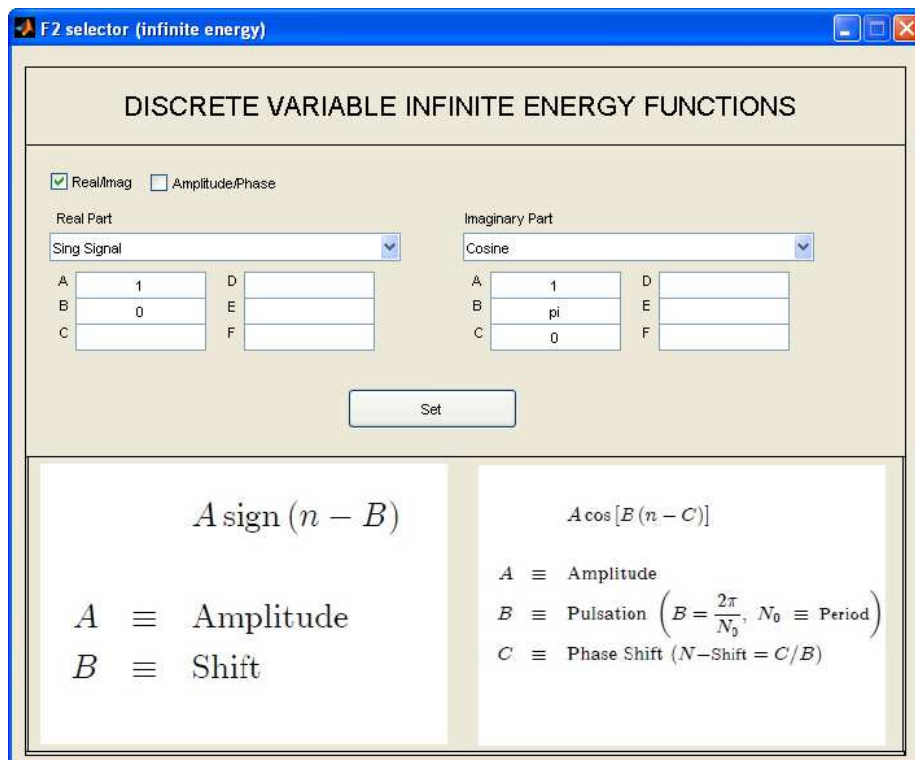


Figura 5.65: Ventana del selector de señales con las señales seleccionadas para definir la segunda señal de entrada al sistema.

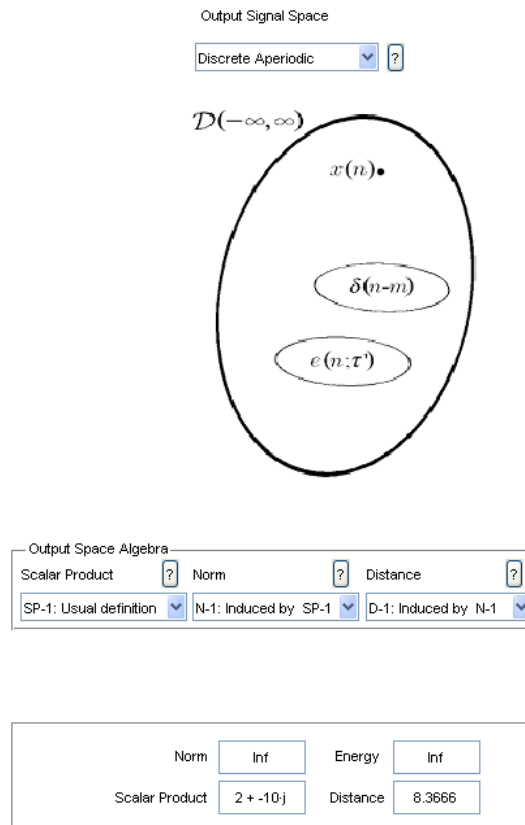


Figura 5.66: La salida del sistema se modificará cuando se haya calculado el resultado: se mostrará el espacio de salida y los valores numéricos asociados a ella.

6. Para obtener los resultados, pulse el botón **Calculate**. Una vez realizado el cálculo y obtenida la señal de salida del sistema verá que en la tercera parte de la ventana se muestra información relacionada con ella, fig. 5.66: el espacio de salida (su nombre y su esquema algebraico, como en la entrada), la norma y la energía de la señal resultado y la distancia y producto escalar entre las señales de entrada al sistema.
7. Finalmente, abra la ventana de visualización, pulsando en el botón **Visualization** que aparece en la parte inferior izquierda de la ventana, para observar las señales de entrada y la señal de del sistema, fig 5.67. Como se observa la señal resultado es la suma de las dos señales de entrada al sistema. Puede manipular los controles de la ventana para visualizar las señales en otro rango de representación, poner la rejilla, cambiar el modo de representación o visualizar la representación compleja de cada señal a su antojo.

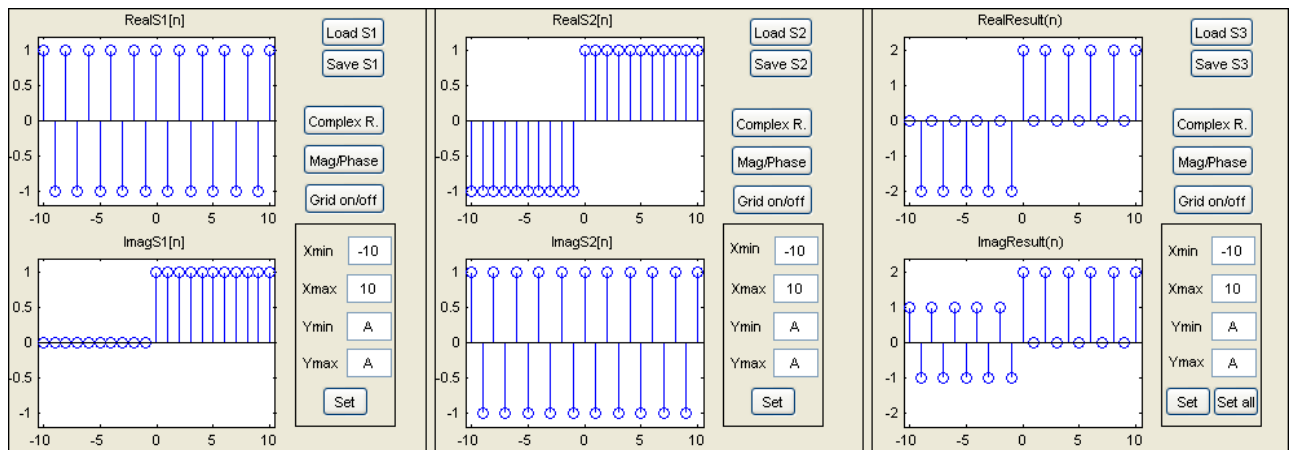


Figura 5.67: Pulsando el botón Visualization del menú principal podrá observar las señales de entrada y la señal salida que, en este caso, es la suma de las dos señales de entrada.

6. CONCLUSIONES Y LÍNEAS FUTURAS

Se presentan a continuación las principales conclusiones que se han obtenido del desarrollo del proyecto y el software resultante.

También se plantean varias líneas de investigación a desarrollar en futuros trabajos para obtener versiones posteriores del software.

6.1. Conclusiones.

Como se introdujo en la Secc. 1.2 de este documento, el desarrollo de este proyecto, y por tanto el desarrollo de la herramienta software resultante, partía de una serie de propuestas y objetivos bien definidos. Una vez finalizada la presente versión del programa, **Signals & Systems Theory v5.0**, éstos se han cumplido en mayor o menor medida. En relación con los principales objetivos planteados, se ha implementado una herramienta con las siguientes características:

- **Estudia en profundidad los sistemas lineales en el dominio espectral, incluyendo como novedad el análisis del caso de los sistemas lineales no invariantes.** Para cumplir este objetivo se ha realizado un estudio teórico previo. Durante esta fase de documentación se ha profundizado en los conceptos de transformada y análisis espectral del sistema. Se ha realizado un estudio particular para cada operador implementado en el software bajo las transformadas disponibles para desarrollar la funcionalidad. Posteriormente, durante una fase de diseño, se han ideado y desarrollado las interfaces gráficas necesarias para el estudio de estos operadores. En este punto hay que destacar el software permite la representación en tres dimensiones del conjunto de funciones núcleo que caracterizan los distintos operadores implementados en el software. Gracias a esta representación, el usuario podrá realizar un estudio profundo del sistema, ya que, de no disponer de esta funcionalidad, el usuario no podría realizar el análisis espectral de los sistemas no invariantes. En relación a este punto cabe destacar que el software es capaz de representar el conjunto de funciones aún cuando está compuesto por deltas de Dirac, con las dificultades que ello conlleva. Finalmente, se han desarrollado las rutinas y los métodos teóricos y numéricos necesarios para los distintos espacios de señales y operadores.
- **Fiel a la teoría generalizada de señales y sistemas.** Ya en versiones anteriores del programa se había conseguido plasmar la teoría de señales y sistemas en su forma más generalizada posible, intentando proveer de las mismas funcionalidades a todos los sistemas implementados, independientemente de sus características. Sin embargo, al incluir la funcionalidad de análisis espectral generalizado, se debe contemplar de nuevo la necesidad de generalizar el estudio de la teoría. Es por ello que, a pesar de ya disponer del análisis espectral del sistema para los operadores lineales e invariantes mediante la representación de la función de transferencia, se ha modificado esta funcionalidad desarrollando un software que **representa el conjunto de funciones núcleo que caracterizan a todos los operadores disponibles en el sistema**, siempre y cuando sean lineales.

Adicionalmente a estos objetivos alcanzados, que se consideran los de mayor importancia de esta tesis, se ha conseguido seguir el resto de pautas definidas para obtener un software:

- Fiel a la metodología, organización y notación que se encuentra en [1] - [2] y [5] - [7], y que venía siendo utilizada en las versiones anteriores del programa, de modo que pueda

apoyar las explicaciones seguidas en las citadas referencias bibliográficas y permita su distribución conjunta en caso requerido. Este aspecto es especialmente interesante de cara a las aplicaciones docentes del software.

- Dotado de rigor teórico. Esto se ha conseguido manteniendo las bases de datos que almacenan los resultados teóricos existentes y mejorándolas en los casos necesarios. También se han creado las bases de datos requeridas para la implementación de las nuevas funcionalidades del software. A pesar de que se considera que el programa ha alcanzado este objetivo, es bien sabido que podría contar con un mayor número de resultados teóricos para operaciones como, por ejemplo, la correlación, por lo que se invita a los futuros autores del programa a que aumenten dichas posibilidades.
- Muy versátil en cuanto a la experimentación con los distintos operadores disponibles por parte de los usuarios. Para ello dispone de una amplia gama de operadores, acompañados de los métodos numéricos y rutinas necesarias para su implementación. Además, se han mejorado y completado los métodos numéricos ya existentes cuando se ha considerado necesario para corregir errores y problemas de cálculo. Hay que mencionar que no todos los operadores están disponibles en todos los espacios de señales, ya que, en función de las características de cada espacio de partida, los distintos operadores pueden ser o no implementables y enriquecer o no el programa con su implementación.
- Preparado para nuevas ampliaciones. En la medida de lo posible, este programa se ha preparado para las futuras ampliaciones que se proponen en la Secc. 6.2.

6.2. Líneas futuras.

Finalmente se presenta a continuación una serie de ideas y líneas generales que sería interesante seguir para desarrollar futuras versiones del programa:

- **Aumentar las definiciones algebraicas** para permitir al usuario profundizar en el conocimiento del álgebra de funciones cuyo estudio se pretende con este programa. Nótese que este punto requeriría previamente de los estudios teóricos asociados al nuevo álgebra.
- **Añadir más apoyos para la aplicación docente.** Consistiría en incluir más funcionalidades como la ventana de animación que permitan al usuario comprender como se obtienen los resultados de forma gráfica e intuitiva.
- **Añadir nuevos operadores.** Es, quizá, la forma más obvia de añadir nuevas funcionalidades al programa. Se podrían añadir nuevos moduladores, el desarrollo del modulador en frecuencia y del modulador en fase se considera muy interesante y se ha dejado el software totalmente preparado para su inclusión. También resultaría interesante desarrollar filtros en frecuencia, como el filtro de Chebyshev o el de Butterworth, y ventaneados en el dominio real, como pueden ser el ventaneado de coseno alzado o el ventaneado de Hamming. También resulta de gran interés incluir nuevos operadores de transformadas, como la transformada de Laplace o la transformada Z, y el desarrollo de nuevos espacios para su posible análisis espectral, ya que sus transformadas pertenecen a un espacio de variable compleja. Contando con el inconveniente que presentan éstas últimas: el desarrollo de un algoritmo de inversión numérica de dichas transformadas.
- **Implementar el análisis espectral de sistemas lineales, en general no invariantes, bajo nuevas transformadas.** Esta es una de las ampliaciones más interesantes de realizar ya que, actualmente, sólo se puede realizar un análisis espectral generalizado en cuatro de los cinco espacios de señales disponibles y bajo la Transformada de Fourier o Desarrollo en Serie de Fourier. Sería muy interesante estudiar el análisis espectral de sistemas lineales no invariantes bajo otras transformadas como la de Hilbert o la de Bessel. La

ampliación consistiría en desarrollar los métodos numéricos e interfaces gráficas necesarias para la representación del conjunto de respuestas al impulso en el dominio espectral bajo la transformada estudiada basándose en los estudios teóricos publicados en [2].

- **Permitir la interacción del programa con señales del mundo real.** Es decir, incorporar en los interfaces formas sencillas de obtener datos del mundo real. Dicha capacidad está actualmente presente, aunque de una forma muy rudimentaria, a través de la posibilidad de cargar y salvar archivos. Pero lo interesante sería incorporar herramientas que permitieran dicha interacción de una forma intuitiva, sin tener que convertir datos de uno a otro formato, y que además permitiese operar con dichas señales. Un caso especialmente interesante de dicha interacción sería el de las señales de audio, pues es un medio especialmente intuitivo que puede resultar muy útil para realizar ciertos experimentos. Finalmente podemos nombrar otro módulo que podría ser muy interesante a la hora de estudiar la relación de la teoría con el mundo real: uno que nos permitiera comparar espectros teóricos con espectros reales obtenidos a partir de analizadores de espectros.
- **Completar el software ya existente** con nuevas señales, nuevos resultados teóricos, nuevos métodos numéricos, etc. con el fin de aumentar la versatilidad del programa.
- Finalmente sería interesante abordar en un futuro la **implementación del software** no desde el punto de vista matemático sino desde la perspectiva informática, con el objetivo de mejorar su apariencia, funcionalidad y, en especial, su eficiencia.

Apéndices

Apéndice A. PRESUPUESTO ECONÓMICO

A.1. Medición de recursos.

Recursos humanos.

Descripción	Unidad	Número de unidades
Analista y diseñador	Hora	200
Programador	Hora	350
Editor de la documentación	Hora	150

Recursos de hardware.

Descripción	Unidad	Número de unidades
Ordenador tipo PC: 2.20 GHz. 4GB de RAM 500 GB de disco duro	Unidad	1

Recursos de software.

Descripción	Unidad	Número de unidades
MATLAB 7.10.0 (R2010a)	Unidad	1
Scientific WorkPlace	Unidad	1

A.2. Medición de costes.

Recursos humanos.

Descripción	Unidad	Precio unitario (€)
Analista y diseñador	Hora	40
Programador	Hora	30
Editor de la documentación	Hora	20

Recursos de hardware.

Descripción	Unidad	Precio unitario (€)
Ordenador tipo PC: Duo P8400 GHz. 4 GB de RAM 250 GB de disco duro	Unidad	700

Recursos de software.

Descripción	Unidad	Precio unitario (€)
MATLAB 7.10.0	Unidad	5.000

A.3. Presupuestos parciales.

Recursos humanos.

Descripción	Unidad	Medición	Precio unitario (€)	Total (€)
Analista y diseñador	Hora	200	40	8.000
Programador	Hora	350	30	10.500
Editor de la documentación	Hora	150	20	3.000
TOTAL				21.500

Recursos de hardware.

Descripción	Unidad	Medición	Precio unitario (€)	Total (€)
Ordenador tipo PC: Duo P8400 GHz. 4 GB de RAM 250 GB de disco duro	Unidad	1	700	700
TOTAL				700

Recursos de software.

Descripción	Unidad	Medición	Precio unitario (€)	Total (€)
MATLAB 7.10.0 (R2010a)	Unidad	1	5.000	5.000
TOTAL				5.000

A.4. Presupuesto de ejecución material.

Descripción	Total (€)
Recursos humanos	21.500
Recursos de hardware	700
Recursos de software	5.000
Presupuesto de ejecución material	27.200

A.5. Presupuesto de ejecución por contrata.

Presupuesto de ejecución material	=	27.200'00 €
Beneficio industrial (6%)	=	1.632'00 €
Costes generales (15%)	=	4.080'00 €
Suma de gastos y beneficios	=	<u>32.912'00 €</u>
IVA (18%)	=	<u>5.924'16 €</u>
Presupuesto de ejecución por contrata	=	38.836'16 €

Asciende el presupuesto de ejecución por contrata a la expresada cantidad de **treinta y ocho mil ochocientos treinta y seis euros con dieciséis céntimos (38.836'16€)**.

Gijón a, 13 de Julio de 2012.

Firmado: Sara Menéndez García (DNI: 53.556.809 K).

Apéndice B. SEÑALES DISPONIBLES EN EL PROGRAMA

B.1. Señales del espacio continuo aperiódico: $S(-\infty, \infty)$.

Coseno	$A \cos[B(x - C)]$
Seno	$A \sin[B(x - C)]$
Gaussiana	$Ae^{-\frac{(x-B)^2}{C^2}}$
1ª Derivada de la Gaussiana	$\frac{d}{dx} Ae^{-\frac{(x-B)^2}{C^2}}$
2ª Derivada de la Gaussiana	$\frac{d^2}{dx^2} Ae^{-\frac{(x-B)^2}{C^2}}$
Pseudo-Gaussiana	$\frac{A}{(x-B)^2+C^2}$
1ª Derivada de la Pseudo-Gaussiana	$\frac{d}{dx} \frac{A}{(x-B)^2+C^2}$
2ª Derivada de la Pseudo-Gaussiana	$\frac{d^2}{dx^2} \frac{A}{(x-B)^2+C^2}$
Pseudo-Gaussiana multiplicada por x	$A \frac{(x-B)}{(x-B)^2+C^2}$
Sinc	$A \operatorname{sinc}[B(x - C)]$
1ª Derivada de la Sinc	$\frac{d}{dx} A \operatorname{sinc}[B(x - C)]$
2ª Derivada de la Sinc	$\frac{d^2}{dx^2} A \operatorname{sinc}[B(x - C)]$
3ª Derivada de la Sinc	$\frac{d^3}{dx^3} A \operatorname{sinc}[B(x - C)]$
Delta	$A\delta(x - B)$
1ª Derivada de la Delta	$A \frac{d}{dx} \delta(x - B)$
2ª Derivada de la Delta	$A \frac{d^2}{dx^2} \delta(x - B)$
Escalón (Heaviside)	$A \cup (x - B)$
Signo	$A \operatorname{sgn}(x - B)$
Escalón Coseno	$A \cos[B(x - C)] \cup (x - D)$
Escalón Seno	$A \sin[B(x - C)] \cup (x - D)$
Escalón Coseno Exponencial	$A \cos[B(x - C)]e^{-D(x-E)} \cup (x - F)$
Escalón Seno Exponencial	$A \sin[B(x - C)]e^{-D(x-E)} \cup (x - F)$
Escalón Exponencial	$Ae^{-B(x-C)} \cup (x - D)$
Rampa	$\frac{A}{B}(x - C)[\cup(x - C) - \cup(x - C - B)]$
Pulso Rectangular	$AP_{\Delta x}(x - x_0)$
Pulso Triangular	$AT_B(x - C)P_{\Delta x}(x - x_0)$
Pulso Exponencial	$Ae^{-B(x-C)}P_{\Delta x}(x - x_0)$

Pulso Gaussiano	$Ae^{-\frac{(x-C)^2}{B^2}} P_{\Delta x}(x-x_0)$
Pulso Lineal	$[A(x-C) + B] P_{\Delta x}(x-x_0)$
Pulso Parabólico	$[A(x-C)^2 + B] P_{\Delta x}(x-x_0)$
Pulso Cosenoidal	$A \cos[B(x-C)] P_{\Delta x}(x-x_0)$
Pulso Senoidal	$A \sin[B(x-C)] P_{\Delta x}(x-x_0)$
Pulso Coseno Exponencial	$A \cos[B(x-C)] e^{-D(x-E)} P_{\Delta x}(x-x_0)$
Pulso Seno Exponencial	$A \sin[B(x-C)] e^{-D(x-E)} P_{\Delta x}(x-x_0)$
Dos Pulsos	$[AP_B(x-C) + DP_E(x-F)] P_{\Delta x}(x-x_0)$
Dos Pulsos Exponenciales	$[\sum_{i=1}^2 A_i e^{-B_i(x-C_i)} P_{D_i}(x-E_i)] P_{\Delta x}(x-x_0)$
Expresión de MATLAB	
Representación vectorial	

B.2. Señales del espacio continuo periódico: $P(X_0)$.

A continuación se presenta una lista de señales $s(x)$. La señal periódica final será la siguiente:

$$s_0(x) = \sum_{n=-\infty}^{\infty} s(x - nX_0).$$

Coseno	$\left\{ \begin{array}{l} A \cos[B(x+C)], \forall D - \frac{X_0}{2} < x < D + \frac{X_0}{2} \\ 0, \text{ otro} \end{array} \right\}$
Senos	$\left\{ \begin{array}{l} A \sin[B(x+C)], \forall D < x < D + X_0 \\ 0, \text{ otro} \end{array} \right\}$
Coseno Exponencial	$\left\{ \begin{array}{l} A \cos\{B[(x-E)-C]\} e^{-D(x-E)}, \forall E < x < E + X_0 \\ 0, \text{ otro} \end{array} \right\}$
Senos Exponencial	$\left\{ \begin{array}{l} A \sin\{B[(x-E)-C]\} e^{-D(x-E)}, \forall E < x < E + X_0 \\ 0, \text{ otro} \end{array} \right\}$
Exponencial	$\left\{ \begin{array}{l} Ae^{-B(x-C)}, \forall D < x < D + X_0 \\ 0, \text{ otro} \end{array} \right\}$
Sinc	$\left\{ \begin{array}{l} A \text{sinc}[B(x-C)], \forall D - \frac{X_0}{2} < x < D + \frac{X_0}{2} \\ 0, \text{ otro} \end{array} \right\}$
Delta	$A\delta(x-B)$
1ª Derivada de la Delta	$A \frac{d}{dx} \delta(x-B)$
2ª Derivada de la Delta	$A \frac{d^2}{dx^2} \delta(x-B)$
Diente de Sierra	$\left\{ \begin{array}{l} \frac{A}{B}(x-C), \forall C < x < C + B \\ 0, \text{ otro} \end{array} \right\}$
Pulso Rectangular	$AP_{\Delta x}(x-x_0)$
Pulso Triangular	$AT_B(x-C) P_{\Delta x}(x-x_0)$
Dos Pulsos	$[AP_B(x-C) + DP_E(x-F)] P_{\Delta x}(x-x_0)$

Expresión MATLAB

Representación vectorial

B.3. Señales del espacio discreto aperiódico: $D(-\infty, \infty)$.

Escalón (Heaviside)	$A \cup (n - B)$
Signo	$A \operatorname{sgn}(n - B)$
Coseno	$A \cos[B(n - C)]$
Senos	$A \sin[B(n - C)]$
Escalón Coseno	$A \cos[B(n - C)] \cup (n - D)$
Escalón Seno	$A \sin[B(n - C)] \cup (n - D)$
Gaussiana	$Ae^{-\frac{(n-B)^2}{C^2}}$
1ª Derivada de la Gaussiana	$\frac{d}{dn} Ae^{-\frac{(n-B)^2}{C^2}}$
2ª Derivada de la Gaussiana	$\frac{d^2}{dn^2} Ae^{-\frac{(n-B)^2}{C^2}}$
Pseudo-Gaussiana	$\frac{A}{(n-B)^2 + C^2}$
1ª Derivada de la Pseudo-Gaussiana	$\frac{d}{dn} \frac{A}{(n-B)^2 + C^2}$
2ª Derivada de la Pseudo-Gaussiana	$\frac{d^2}{dn^2} \frac{A}{(n-B)^2 + C^2}$
Pseudo-Gaussiana multiplicada por n	$\frac{d^2}{dn^2} \frac{A}{(n-B)^2 + C^2}$
Sinc	$A \operatorname{sinc}[B(n - C)]$
1ª Derivada de la Sinc	$A \frac{d}{dn} \operatorname{sinc}[B(n - C)]$
2ª Derivada de la Sinc	$A \frac{d^2}{dn^2} \operatorname{sinc}[B(n - C)]$
3ª Derivada de la Sinc	$A \frac{d^3}{dn^3} \operatorname{sinc}[B(n - C)]$
Escalón Coseno Exponencial	$A \cos[B(n - C)]e^{-D(n-E)} \cup (n - F)$
Escalón Seno Exponencial	$A \sin[B(n - C)]e^{-D(n-E)} \cup (n - F)$
Escalón Exponencial	$Ae^{-B(n-C)} \cup (n - D)$
Rampa	$\frac{A}{B}(n - C)[\cup(n - C) - U(n - C - B)]$
Pulso Rectangular	$AP_{\Delta n}(n - n_0)$
Pulso Triangular	$AT_B(n - C)P_{\Delta n}(n - n_0)$
Pulso Exponencial	$Ae^{-B(n-C)}P_{\Delta n}(n - n_0)$
Pulso Gaussiano	$Ae^{-\frac{(n-B)^2}{C^2}}P_{\Delta n}(n - n_0)$
Pulso Lineal	$[A(n - C) + B]P_{\Delta n}(n - n_0)$
Pulso Parabólico	$[A(n - C)^2 + B]P_{\Delta n}(n - n_0)$
Pulso Cosenoidal	$A \cos[B(n - C)]P_{\Delta n}(n - n_0)$

Pulso Senoidal	$A \sin[B(n - C)]P_{\Delta n}(n - n_0)$
Pulso Coseno Exponencial	$A \cos[B(n - C)]e^{-D(n-E)}P_{\Delta n}(n - n_0)$
Pulso Seno Exponencial	$A \sin[B(n - C)]e^{-D(n-E)}P_{\Delta n}(n - n_0)$
Dos Pulsos	$[AP_B(n - C) + DP_E(n - F)]P_{\Delta n}(n - n_0)$
Dos Pulsos Exponenciales	$[\sum_{i=1}^2 A_i e^{-B_i(n-C_i)} P_{D_i}(n - E_i)]P_{\Delta n}(n - n_0)$
Expresión MATLAB	
Representación vectorial	

B.4. Señales del espacio discreto periódico: $D(N_0)$.

A continuación se presenta una lista de señales $s(n)$. La señal periódica final será la siguiente:

$$s_0(n) = \sum_{k=-\infty}^{\infty} s(n - kN_0).$$

Coseno	$\left\{ \begin{array}{l} A \cos[B(n - C)], \forall D - \frac{N_0}{2} < n < D + \frac{N_0}{2} \\ 0, \text{ otro} \end{array} \right\}$
Seno	$\left\{ \begin{array}{l} A \sin[B(n - C)], \forall D < n < D + N_0 \\ 0, \text{ otro} \end{array} \right\}$
Coseno Exponencial	$\left\{ \begin{array}{l} A \cos[B(n - E) - C]e^{-D(n-E)}, \forall E < n < E + N_0 \\ 0, \text{ otro} \end{array} \right\}$
Seno Exponencial	$\left\{ \begin{array}{l} A \sin[B(n - E) - C]e^{-D(n-E)}, \forall E < n < E + N_0 \\ 0, \text{ otro} \end{array} \right\}$
Exponencial	$\left\{ \begin{array}{l} Ae^{-B(n-C)}, \forall D < n < D + N_0 \\ 0, \text{ otro} \end{array} \right\}$
Sinc	$\left\{ \begin{array}{l} A \text{sinc}[B(x - C)], \forall D - \frac{N_0}{2} < x < D + \frac{N_0}{2} \\ 0, \text{ otro} \end{array} \right\}$
Diente de Sierra	$\left\{ \begin{array}{l} \frac{A}{B}(n - C), C < n < C + B \\ 0, \text{ otro} \end{array} \right\}$
Pulso Rectangular	$AP_{\Delta n}(n - n_0)$
Pulso Triangular	$[AT_B(n - C)]P_{\Delta n}(n - n_0)$
Dos Pulsos	$[AP_B(n - C) + DP_E(n - F)]P_{\Delta n}(n - n_0)$
Expresión MATLAB	
Representación vectorial	

B.5. Señales del espacio continuo aperiódico definidas para valores positivos:
 $S(0, \infty)$.

Raíz de x multiplicada por la Funcion de Bessel	$A\sqrt{Bx}J_C(Bx)$
Pseudo-Gaussiana modificada	$A\frac{x^{C+\frac{1}{2}}}{x^2+B^2}$
Gaussiana multiplicada por x	$Ax^{C+\frac{1}{2}}e^{-Bx^2}$
Raíz de x multiplicada por la Gaussiana y multiplicada por la Funcion de Bessel	$A\sqrt{x}e^{-Bx^2}J_D(Cx)$
Delta	$A\delta(x - B)$
1ª Derivada de la Delta	$A\frac{d}{dx}\delta(x - B)$
2ª Derivada de la Delta	$A\frac{d^2}{dx^2}\delta(x - B)$
Pulso en Potencia	$Ax^{B+\frac{1}{2}}P_1(x - \frac{1}{2})$
Gaussiana dividida por la raíz de x	$A\frac{e^{-Bx^2}}{\sqrt{x}}, C$
Exponencial dividida por la raíz de x	$A\frac{e^{-Bx}}{\sqrt{x}}, C$

Bibliografía

- [1] E. Gago-Ribas, J Heredia Juesas. *A general approach to Signals & Systems Theory*.
- [2] Juan Heredia Juesas. *Generalización de la Teoría de Señales y Sistemas para el análisis de problemas Lineales No Invariantes*. Tesis fin de master. Master en Tecnologías de la Información y Comunicaciones en Redes Móviles, Universidad de Oviedo, 2010.
- [3] Carlos Baladrón Zorita. *Signals & Systems Theory v1.0*. Proyecto Fin de Carrera *Desarrollo de una Herramienta Software para la Docencia de la Teoría de Señales y Sistemas*, E.T.S.I. Telecomunicación de Valladolid, 2006.
- [4] Sara Menéndez García. *Signals & Systems Theory v2.0*. Proyecto Fin de Carrera *Desarrollo de una Herramienta Software para el Estudio de la Teoría de Señales y Sistemas*, Escuela Universitaria de Ingeniería Técnica Informática de Gijón, 2008.
- [5] Juan Heredia Juesas. *Signals & Systems Theory v3.0*. Proyecto Fin de Carrera *Análisis General de la Teoría de Señales y Sistemas para su Aplicación al Modelado de Problemas Físicos*, Escuela Politécnica Superior de Ingeniería de Gijón, 2009.
- [6] Sara Menéndez García. *Signals & Systems Theory v4.0*. Proyecto Fin de Carrera *Desarrollo de la Herramienta Software "Signals & Systems Theory v4.0" para el Análisis Real y Espectral de Sistemas Lineales no Invariantes*, Escuela Politécnica Superior de Ingeniería de Gijón, 2011.
- [7] Emilio Gago-Ribas. *Señales y Sistemas Escalares Unidimensionales de Variable Real: Introducción y Análisis General de Señales y Sistemas de Variable Continua. Vol. STI de la serie Electromagnetics & Signal Theory Notebooks*. GR Editores, S.L., 2002.
- [8] A.V. Oppenheim, A.S. Willsky, I.T. Young. *Signals and Systems (2nd. Ed.)*. Prentice-Hall International, 1997.
- [9] A.V. Oppenheim, R.W. Schafer. *Discrete-Time Signal Processing*. Prentice-Hall International, 1989.
- [10] V.S. Vladimirov. *Generalized Functions in Mathematical Physics*. Ed. Mir, 1979.
- [11] J.I. Richards y H.K. Youn. *Theory of Distributions. A Non-Technical Introduction*. Cambridge University Press, 1990.
- [12] Emilio Gago-Ribas. *Señales y Sistemas Escalares Unidimensionales de Variable Real. Ejercicios Resueltos*. GR Editores, S.L., 2009.
- [13] Emilio Gago-Ribas, Sara Menéndez García, Juan Heredia Juesas. *A generalized signal & systems theory and an associated software tool*. International Journal for Knowledge, Science and Technology (IJKST), Bilbao, 2010.
- [14] E. Gago-Ribas, S. Menéndez García, M. Colunga Nachón. *A Generalized Signals & Systems Theory and an Associated Software Tool*, International Conference on Engineering and Mathematics, ENMA 2009 Proceedings, pp. 274-281, Bilbao, España, Junio 17-19, 2009 (ISBN 978-84-937328-0-6).
- [15] M.R. Spiegel, J. Liu, L. Abellanas. *Fórmulas y Tablas de Matemática Aplicada*. McGraw-Hill, 2000.

- [16] G. Petit Bois. *Tables of Indefinite Integrals*. Dover Publications, Inc. 1961.
- [17] L. Bayón, P. Suárez. *Análisis real y complejo (versión en apuntes)*, 2004.
- [18] The MathWorks Inc. *The Student Edition of MATLAB*. Prentice-Hall, 1992.
- [19] The MathWorks Inc. *MATLAB Creating Graphical User Interfaces*.
- [20] <http://www.mathworks.com/support>. The MathWorks MATLAB support web site.