

YONG, DAVID MARCELO
Análisis de perturba

2013

73210

73210



**ANALISIS DE PERTURBACIONES EN SISTEMAS
ELECTRICOS DE POTENCIA UTILIZANDO
TRANSFORMADA WAVELETS, REDES NEURONALES Y
MÁQUINAS DE SOPORTE DE VECTORES**

por

David Marcelo de Yong

**Tesis presentada para la obtención del grado de Magíster en
Ciencias de la Ingeniería**

Mención: Ingeniería Eléctrica

**FACULTAD DE INGENIERÍA UNIVERSIDAD
NACIONAL DE RÍO CUARTO**

01587

Diciembre de 2013

01227

73210

MFN:
Clasif:
T. 872

ANALISIS DE PERTURBACIONES EN SISTEMAS ELECTRICOS DE POTENCIA UTILIZANDO TRANSFORMADA WAVELETS, REDES NEURONALES Y MÁQUINAS DE SOPORTE DE VECTORES

por

David Marcelo de Yong



Dr. Fernando Magnago
Director



Dr. Claudio Reineri
Co-Director

Aprobada por:



Dr. Ing Enrique Mombello
Jurado



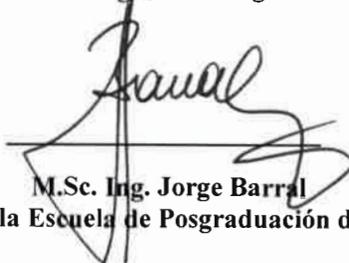
Dr. Ing. Guillermo Bossio
Jurado



Dr. Ing. Carlos Amatti
Jurado



Dr. Ing Leonardo Molisani
Secretario de Postgrado de Ingeniería UNRC



M.Sc. Ing. Jorge Barral
Director de la Escuela de Posgraduación de la UNRC

Diciembre de 2013

*A Paola, cuyo inagotable amor, comprensión y apoyo
me es, fue y será indispensable para vivir.*

A la memoria de mi padre.

A mi madre, cuya valentía admiro.

A mis hermanos.

A la familia Perotto Ghi.

AGRADECIMIENTOS

Deseo expresar mi gratitud a la Universidad Nacional de Río Cuarto, a la Facultad de Ingeniería y a sus directivos, quienes me brindaron la posibilidad de realizar la Maestría en Ciencias de la Ingeniería.

Quiero agradecer muy especialmente al *Doctor Ingeniero Fernando Magnago*, director de mi maestría y doctorado, cuyo profesionalismo y compromiso me motivaron a comenzar este camino. Sin su conocimiento y experiencia el desarrollo de este trabajo hubiera resultado imposible.

También quiero agradecer el apoyo del co-director de este trabajo de tesis, el *Doctor Ingeniero Claudio Reineri*, cuya experiencia en el área de Calidad de Energía influyeron en gran medida a que pudiera finalizar este trabajo.

Además quiero dar las gracias a los Ingenieros *Jorge Federico Aguirre* y *Eduardo Toledo* por sus consejos y aportes.

Este trabajo ha formado parte del Proyecto de Investigación "*Técnicas de procesamiento digital de señales para el monitoreo de transitorios en sistemas eléctricos de potencia*" del Grupo de Análisis de Sistemas Eléctricos de Potencia (GASEP) de la Facultad de Ingeniería. Por este motivo quiero expresar mi más sincero reconocimiento a todos los integrantes del grupo, muy particularmente al *Doctor Ingeniero Luis Aromataris* y al *Ingeniero Juan Alemany* quienes en todo momento me brindaron su apoyo.

Agradezco también la predisposición y confianza de los Ingenieros *Santiago Lovera*, *Ricardo Lima*, *Diego Gagliesi* y *Julián Durigutti*.

Quiero reconocer también al Ingeniero *Tom Grebe* de la empresa Electrotek Inc. quien muy amablemente me facilitó un conjunto de formas de ondas reales obtenidas mediante mediciones de sistemas eléctricos de potencia en los Estados Unidos las cuales fueron muy importantes para el desarrollo de este trabajo.

Agradezco también los consejos de *Sudipto Bhowmik* cuya experiencia en el área me permitió avanzar con mis tareas de investigación.

Finalmente, quiero agradecer a los jurados y revisores de esta tesis, el Doctor Ingeniero Enrique Mombello, Doctor Ingeniero Carlos Amatti, Doctor Ingeniero Juan Fontana y Doctor Ingeniero Guillermo Bossio quienes, con sus consejos, aportaron enormemente en el desarrollo de la presente tesis de maestría.

TABLA DE CONTENIDOS

AGRADECIMIENTOS.....	VI
TABLA DE CONTENIDOS.....	VII
INDICE DE TABLAS	XII
INDICE DE FIGURAS	XIII
RESUMEN	XVI
INTRODUCCIÓN.....	1
PRIMERA PARTE: CONCEPTOS TEÓRICOS.....	9
CAPÍTULO 1: CALIDAD DE ENERGÍA Y SISTEMAS DE MONITOREO.....	11
1.1 INTRODUCCIÓN	12
1.2 CALIDAD DE ENERGIA - DESCRIPCION DE LA PROBLEMÁTICA.....	13
1.2.1 DEFINICIÓN DE CALIDAD DE ENERGÍA	13
1.2.2 INTERÉS EN CALIDAD DE ENERGÍA	13
1.2.3 CLASIFICACIÓN DE EVENTOS DE CALIDAD DE ENERGÍA.....	14
1.2.4 CLASIFICACIÓN DE ACUERDO A LA DURACIÓN	14
1.2.5 CLASIFICACIÓN DE ACUERDO A LA IEC.....	15
1.2.6 CLASIFICACIÓN DE ACUERDO A LA IEEE	15
1.2.6.1 <i>Transitorios</i>	16
1.2.6.2 <i>Variaciones de corta duración</i>	18
1.2.6.3 <i>Variaciones de larga duración</i>	21
1.2.6.4 <i>Desbalance de tensión</i>	22
1.2.6.5 <i>Distorsión de la forma de onda</i>	22
1.2.6.6 <i>Fluctuaciones de tensión</i>	25
1.2.6.7 <i>Variaciones de la frecuencia fundamental</i>	26
1.2.7 MODELOS EMPLEADOS PARA LA SIMULACIÓN DE LOS EVENTOS DE CALIDAD DE ENERGÍA	26
1.3 SISTEMA DE MONITOREO DE EVENTOS DE CALIDAD DE ENERGIA.....	28
1.3.1 NECESIDADES DE MONITOREO.....	28
1.3.2 RESEÑA HISTÓRICA.....	29
1.3.3 CLASIFICACIÓN DE SISTEMAS DE MONITOREO	29
1.3.4 SISTEMAS DE MONITOREO DE CALIDAD DE ENERGÍA	31
1.4 ESTADO DEL ARTE: PROCESO DE ANALISIS DE LOS DATOS	33
1.4.1 DETECCIÓN DE EVENTOS DE CALIDAD DE ENERGÍA	33
1.4.2 CLASIFICACIÓN DE EVENTOS DE CALIDAD DE ENERGÍA.....	35
1.4.3 CARACTERIZACIÓN DE EVENTOS DE CALIDAD DE ENERGÍA	36
1.5 CONCLUSIONES.....	37
CAPÍTULO 2: TRANSFORMADA WAVELET.....	39
2.1 INTRODUCCIÓN	40
2.2 TRANSFORMADA WAVELET	41
2.2.1 CONCEPTOS PREVIOS.....	41
2.2.1.1 <i>Clasificación de señales</i>	41

2.2.1.2	<i>Transformada de Fourier</i>	42
2.2.1.3	<i>Transformada de Fourier dependiente del tiempo</i>	44
2.2.2	LA TRANSFORMADA WAVELET.....	46
2.2.2.1	<i>Wavelets madre</i>	47
2.2.2.2	<i>Expansión y traslación de una wavelet madre</i>	49
2.2.2.3	<i>La transformada Wavelet de Tiempo Continuo (CWT)</i>	50
2.2.2.4	<i>Transformada Wavelet Discreta (DWT)</i>	52
2.2.2.5	<i>Interpretación de la DWT como un banco de filtros</i>	53
2.2.2.6	<i>Teorema de Parseval</i>	56
2.3	ESTADO DEL ARTE: LA TRANSFORMADA WAVELET PARA LA DETECCION DE PERTURBACIONES	58
2.4	CONCLUSIONES	60
CAPÍTULO 3: REDES NEURONALES ARTIFICIALES		61
3.1	INTRODUCCIÓN	62
3.2	CONCEPTOS BASICOS DE LAS REDES NEURONALES	64
3.2.1	SIMILITUDES CON LAS REDES NEURONALES BIOLÓGICAS.....	66
3.3	CARACTERISTICAS DE LAS REDES NEURONALES	68
3.3.1	APRENDIZAJE ADAPTATIVO.....	68
3.3.2	GENERALIZACIÓN.....	68
3.3.3	ABSTRACCIÓN.....	68
3.3.4	AUTO-ORGANIZACIÓN.....	68
3.3.5	TOLERANCIA A FALLOS.....	68
3.3.6	OPERACIÓN EN TIEMPO REAL.....	68
3.3.7	FÁCIL INSERCIÓN DENTRO DE LA TECNOLOGÍA EXISTENTE.....	68
3.4	ARQUITECTURAS DE LAS REDES NEURONALES	69
3.4.1	REDES MONOCAPA.....	70
3.4.2	REDES MULTICAPA.....	70
3.5	AJUSTE DE LOS PESOS Y METODOS DE ENTRENAMIENTO	71
3.5.1	ENTRENAMIENTO SUPERVISADO.....	71
3.5.1.1	<i>Aprendizaje por corrección de error</i>	71
3.5.1.2	<i>Aprendizaje por refuerzo</i>	71
3.5.1.3	APRENDIZAJE ESTOCÁSTICO	72
3.5.2	ENTRENAMIENTO NO SUPERVISADO.....	72
3.6	FUNCIONES DE ACTIVACION	73
3.6.1	FUNCIÓN IDENTIDAD.....	73
3.6.2	FUNCIÓN ESCALÓN.....	73
3.6.3	FUNCIÓN SIGMOIDEA BINARIA.....	74
3.6.4	FUNCIÓN TANGENTE HIPERBÓLICA.....	75
3.7	RED NEURONAL BACK PROPAGATION	76
3.7.1	ARQUITECTURA DE LA RED BACKPROPAGATION.....	76
3.7.2	FUNCIONES DE ACTIVACIÓN.....	77
3.7.3	NOMENCLATURA.....	77

3.7.4	ALGORITMO DE ENTRENAMIENTO	78
3.8	CONSIDERACIONES DE CONFIGURACION	82
3.8.1	SELECCIÓN DEL VALOR INICIAL DE LOS BIAS Y LOS PESOS	82
3.8.2	SELECCIÓN DEL TAMAÑO DEL SET DE ENTRENAMIENTO	82
3.8.3	NÚMERO DE CAPAS OCULTAS	82
3.8.4	NÚMERO DE NEURONAS EN LA CAPA OCULTA.....	82
3.9	ESTADO DEL ARTE; EMPLEO DE REDES NEURONALES EN SISTEMAS DE MONITOREO	84
3.10	CONCLUSIÓN	87
CAPÍTULO 4: MÁQUINA DE VECTORES DE SOPORTE.....		89
4.1	INTRODUCCIÓN	90
4.2	TEORIA DE APRENDIZAJE ESTADISTICO	92
4.2.1	APRENDIZAJE SUPERVISADO.....	92
4.2.2	LIMITACIONES EN EL APRENDIZAJE.....	92
4.2.3	PROCESO DE APRENDIZAJE PARA LAS MÁQUINAS DE VECTORES DE SOPORTE.....	93
4.3	TEORIA DE LA OPTIMIZACION MATEMATICA	94
4.3.1	CONCEPTOS	94
4.3.2	FORMULACIÓN DEL PROBLEMA.....	94
4.3.3	DEFINICIONES IMPORTANTES	95
4.3.3.1	<i>Programación lineal.....</i>	<i>95</i>
4.3.3.2	<i>Programación cuadrática</i>	<i>95</i>
4.3.3.3	<i>Función cóncava y convexa.....</i>	<i>95</i>
4.3.3.4	<i>Transformación de las restricciones</i>	<i>95</i>
4.3.4	TEORÍA DE LAGRANGE	96
4.3.5	DUALIDAD	98
4.4	MAQUINAS DE SOPORTE VECTORIAL.....	99
4.4.1	CLASIFICADOR DE VECTORES DE SOPORTE	99
4.4.2	CLASIFICADOR DE MÁXIMO MARGEN	99
4.4.2.1	<i>Clasificador linealmente separables</i>	<i>100</i>
4.4.2.2	<i>Clasificador no linealmente separables</i>	<i>105</i>
4.4.3	TRANSFORMACIÓN DE SUB ESPACIO USANDO FUNCIONES KERNEL	106
4.4.3.1	<i>Funciones kernel</i>	<i>108</i>
4.4.3.2	<i>Incorporación del kernel al SVM</i>	<i>109</i>
4.4.4	MÁQUINAS DE SOPORTE DE VECTORES MULTICLASE	110
4.4.4.1	<i>Uno contra el resto (One versus rest)</i>	<i>110</i>
4.4.4.2	<i>Uno contra uno (One vs One)</i>	<i>110</i>
4.4.4.3	<i>Árbol de jerarquía de clasificadores</i>	<i>110</i>
4.5	ESTADO DEL ARTE: APLICACIÓN DE MAQUINAS DE SOPORTE VECTORIAL PARA LA CLASIFICACION DE PERTURBACIONES.....	112
4.6	CONCLUSIONES.....	114
SEGUNDA PARTE: DESARROLLO DE LA HERRAMIENTA		115
CAPÍTULO 5: SOFTWARE PARA EL ANÁLISIS DE EVENTOS DE CALIDAD DE ENERGÍA.....		117
5.1	INTRODUCCIÓN	118

5.2	SOFTWARE PARA EL ANALISIS DE EVENTOS DE CALIDAD DE ENERGIA	120
5.3	ENTRENAMIENTO DEL MODULO CLASIFICADOR	121
5.3.1	GENERACIÓN DE SEÑALES DE ENTRENAMIENTO Y PRUEBA	121
5.3.2	EXTRACCIÓN DE PARÁMETROS CARACTERÍSTICOS	121
5.3.2.1	<i>Adquisición de la señal.....</i>	<i>121</i>
5.3.2.2	<i>Eventanado de la señal.....</i>	<i>122</i>
5.3.2.3	<i>Transformada Wavelet.....</i>	<i>123</i>
5.3.2.4	<i>Conformación del vector característico</i>	<i>124</i>
5.4	ENTRENAMIENTO DEL CLASIFICADOR.....	125
5.4.1	AJUSTE DE LOS PARÁMETROS DEL CLASIFICADOR	126
5.4.2	ENTRENAMIENTO SUPERVISADO DEL CLASIFICADOR	128
5.4.3	CONTROL DE LA EFICIENCIA.....	128
5.4.4	ALMACENAMIENTO DE LOS PARÁMETROS DE ENTRENAMIENTO	129
5.5	ANALISIS DEL COMPORTAMIENTO DE LOS ALGORITMOS EN SITUACIONES REALES	130
5.5.1	ADQUISICIÓN DE LAS SEÑALES Y LAS ESTRUCTURAS DEL CLASIFICADOR	130
5.5.2	PRE-PROCESAMIENTO	130
5.5.3	EXTRACCIÓN DE LOS PARÁMETROS CARACTERÍSTICOS.....	130
5.5.4	DETECCIÓN	131
5.5.5	LOCALIZACIÓN TEMPORAL	131
5.5.6	CLASIFICACIÓN DE LA PERTURBACIÓN	132
5.5.7	POST-PROCESAMIENTO	132
5.5.7.1	<i>Ganador toma todo</i>	<i>132</i>
5.5.7.2	<i>Algoritmo basado en reglas.....</i>	<i>133</i>
5.6	SOFTWARE DE MONITOREO – CASO DE USO	134
5.6.1	ACCESO A LA HERRAMIENTA	134
5.6.2	GENERACIÓN DE SEÑALES DE ENTRENAMIENTO Y PRUEBA	135
5.6.3	EXTRACCIÓN DE PARÁMETROS CARACTERÍSTICOS DE LAS SEÑALES DE ENTRENAMIENTO Y PRUEBA.	137
5.6.4	ENTRENAMIENTO Y PRUEBA DEL CLASIFICADOR.....	139
5.6.4.1	<i>Red Neuronal Backpropagation.....</i>	<i>139</i>
5.6.4.2	<i>Support Vector Machine.....</i>	<i>140</i>
5.6.5	INCORPORACIÓN DE SEÑALES REALES.....	142
5.6.6	SIMULACIÓN DE LA CLASIFICACIÓN	143
5.7	CONCLUSION	145
CAPÍTULO 6: RESULTADOS DE LAS EXPERIENCIAS		147
6.1	RESULTADOS Y SIMULACIONES.....	148
6.1.1	EXPERIENCIA 1: CLASIFICADOR SVM BASADO EN UN ÁRBOL DE DECISIÓN BINARIO	148
6.1.1.1	<i>Conformación del set de entrenamiento y prueba</i>	<i>148</i>
6.1.1.2	<i>Resultados obtenidos.....</i>	<i>149</i>
6.1.2	EXPERIENCIA 2: COMPARACIÓN DE UN CLASIFICADOR ONE VS ALL SVM CON UNA ANN BACKPROPAGATION.	150
6.1.2.1	<i>Conformación del set de entrenamiento y prueba</i>	<i>151</i>
6.1.2.2	<i>Resultados obtenidos.....</i>	<i>152</i>
6.2	CONCLUSIONES.....	153
6.2.1	CONCLUSIONES DE LA EXPERIENCIA 1	153

6.2.2	CONCLUSIONES DE LA EXPERIENCIA 2	154
6.2.3	COMPARACIÓN EXPERIENCIA 1 CON EXPERIENCIA 2	155
CAPÍTULO 7: CONCLUSIONES Y TRABAJOS FUTUROS.....		157
7.1	CONCLUSIONES.....	158
7.2	TRABAJOS FUTUROS	161
CAPÍTULO 8: REFERENCIAS.....		163

INDICE DE TABLAS

Capítulo 1

Tabla 1.1 - Principales fenómenos que causan perturbaciones electromagnéticas.....	15
Tabla 1.2 - Categorías y Características de fenómenos electromagnéticos en un SEP.....	16
Tabla 1.3 - Modelo de perturbaciones.....	27

Capítulo 6

Tabla 6.1 - Set de Entrenamiento	149
Tabla 6.2 - Matriz de confusión del Clasificador SVM.....	150
Tabla 6.3 - Set de entrenamiento para el clasificador SVM	151
Tabla 6.4 - Matriz de confusión de clasificador ANN	152
Tabla 6.5 - Matriz de confusión del clasificador SVM	153

INDICE DE FIGURAS

Capítulo 1

Figura 1.1 - Transitorio Oscilatorio de alta frecuencia.....	17
Figura 1.2 - Transitorio Oscilatorio de baja frecuencia.....	18
Figura 1.3 - Interrupción ideal.....	19
Figura 1.4 (a) Variación de la tensión instantánea ante una Interrupción. (b) Variación del valor eficaz.....	19
Figura 1.5 - (a) Tensión instantánea del sistema ante un Hueco de tensión. (b) Tensión eficaz del sistema ante un Hueco de tensión.....	20
Figura 1.6 - (a) Tensión Instantánea del sistema ante una Sobretensión. (b) Tensión Eficaz del sistema ante una Sobretensión.....	21
Figura 1.7 - a) Tensión instantánea del sistema conteniendo Armónicos. b) Espectro de la señal.....	23
Figura 1.8 - Tensión instantánea de un sistema ante una Muesca.....	24
Figura 1.9 - Tensión instantánea del sistema contaminada con Ruido.....	25
Figura 1.10 - Tensión instantánea del sistema ante Fluctuaciones.....	26
Figura 1.11 - Sistema de Monitoreo [2].	31
Figura 1.12 - Análisis de las muestras.....	33

Capítulo 2

Figura 2.1 - Señal estacionaria.....	41
Figura 2.2 - Señal no estacionaria.....	42
Figura 2.3 - Señal estacionaria.....	43
Figura 2.4 - Señal no estacionaria.....	44
Figura 2.5 - Transformación de dominios de la STFT.....	45
Figura 2.6 - Transformada de Fourier de tiempo Corto de un Hueco de Tensión.....	46
Figura 2.7 - a) Wavelet Haar b) Wavelet Sombrero Mexicano.....	48
Figura 2.8 - Wavelet Gaussiana Compleja b) Wavelet Shannon Compleja.....	48
Figura 2.9 - a) Wavelet Meyer b) Wavelet Morlet.....	48
Figura 2.10 - Wavelet madre Daubechie 4.....	49
Figura 2.11 - Traslación de la Wavelet Madre.....	50
Figura 2.12 - Escalamiento de la Wavelet Madre.....	50
Figura 2.13 - Plano Escala-Tiempo de la Transformada Wavelet.....	52
Figura 2.14 - Aproximación mediante banco de filtros.....	53
Figura 2.15 - Banco de filtros y sub muestreo.....	54
Figura 2.16 - Coeficientes A y D de una señal senoidal más ruido blanco.....	54
Figura 2.17 - Árbol de descomposición.....	55
Figura 2.18 - Árbol de descomposición para una perturbación tipo Muesca.....	56

Capítulo 3

Figura 3.1 - Esquema de una Neurona.....	64
Figura 3.2 - Conexión entre neuronas.....	65
Figura 3.3 - Neurona biológica [35].....	66
Figura 3.4 - Red Neuronal Monocapa.....	69
Figura 3.5 - Red Neuronal Multicapa.....	70
Figura 3.6 - Función de activación lineal.....	73
Figura 3.7 - Función de activación escalón.....	74
Figura 3.8 - Función de activación sigmoide binaria.....	74
Figura 3.9 - Función de activación tangente sigmoide.....	75
Figura 3.10 - Arquitectura de una red multi-capa.....	77

Capítulo 4

Figura 4.1 - Fronteras de decisión.....	100
Figura 4.2 - Caso linealmente separable.....	101
Figura 4.3 - Hiperplano de separación.....	102
Figura 4.4 - Hiperplano para el caso no linealmente separable.....	106
Figura 4.5 - Transformación del espacio de características usando funciones Kernel.....	108
Figura 4.6 - Estructura jerárquica de tipo árbol para un clasificador SVM multiclase.....	111

Capítulo 5

Figura 5.1 - Sistema de procesamiento de perturbaciones.....	120
Figura 5.2 - Proceso de entrenamiento de un clasificador.....	121
Figura 5.3 - Algoritmo de detección y caracterización de eventos de calidad de energía.....	122
Figura 5.4 - Ventaneo de una señal.....	123
Figura 5.5 - Transformada Wavelet de 9 niveles.....	123
Figura 5.6 - Algoritmo de entrenamiento del clasificador.....	126
Figura 5.7 - Arquitectura de un clasificador basado en SVM.....	127
Figura 5.8 - Algoritmo de clasificación de señales reales.....	130
Figura 5.9 - Vector característico.....	131
Figura 5.10 - Umbralización del coeficiente d1 de la TW.....	132
Figura 5.11 - Acceso a la herramienta.....	134
Figura 5.12 - Panel principal.....	135
Figura 5.13 - Generador de señales de prueba.....	136
Figura 5.14 - Generador automático de señales perturbadas.....	136
Figura 5.15 - Panel de generación del vector característico.....	138
Figura 5.16 - Clasificador basado en una Red Neuronal Backpropagation.....	139
Figura 5.17 - Clasificador basado en Máquinas de Soporte Vectorial.....	140
Figura 5.18 - Hueco de tensión adquirido de la red.....	142
Figura 5.19 - Señal real normalizada.....	142

Figura 5.20 - Localización y clasificación de una señal real..... 143

Capítulo 6

Figura 6.1 - Arquitectura de un clasificador SVM del tipo árbol de decisión binario 148

Figura 6.2 - Arquitectura de un clasificador SVM Uno contra el resto 151

RESUMEN

En los últimos años la Calidad de la Energía (*PQ por sus siglas en inglés*) en un Sistema Eléctrico de Potencia (SEP) se ha vuelto una materia de extrema importancia tanto para las compañías proveedoras y distribuidoras de energía eléctrica como también para los clientes. Debido a los crecientes niveles de demanda de energía a nivel global se hizo imprescindible la incorporación del concepto conocido como *Redes Eléctricas Inteligentes (Smart Grids)* para permitir, entre otras cosas, la integración de la red eléctrica convencional con nuevas fuentes de energía distribuida, limpia y renovable. La naturaleza no estacionaria de la energía renovable y las maniobras involucradas en la integración de las fuentes de energía distribuida con la red tradicional, generan una innumerable cantidad de problemas relacionados con la calidad de energía eléctrica.

Por este motivo el proceso de detección, clasificación, caracterización y propagación de perturbaciones presentes en un sistema eléctrico, la localización de las fuentes que los producen, el efecto que estas perturbaciones tienen sobre los equipos sensibles a ellas y, las posibles formas de mitigar estos eventos, son tareas de suma importancia.

El objetivo del presente trabajo de tesis es estudiar distintas técnicas de procesamiento de señales para la detección y extracción de parámetros característicos de una perturbación. Además se plantea como objetivo la comparación de diferentes máquinas de aprendizaje para la clasificación de perturbaciones electromagnéticas de diferente naturaleza.

Con este objetivo se desarrolló una herramienta informática que permite la detección, localización y clasificación de un conjunto fenómenos electromagnéticos en las formas de onda adquiridas de un SEP.

Como primera medida se describen los fundamentos teóricos básicos de las herramientas matemáticas que se emplean en el procesamiento de las señales. Luego, se estudian e implementan, técnicas de detección de perturbaciones basadas en la *Transformada Wavelet*. Posteriormente se tratan en detalle técnicas que luego son empleadas para clasificar patrones en forma automática, centrando el trabajo principalmente las *Máquinas de Soporte Vectorial*, en muchos aspectos, como una alternativa superadora a las Redes Neuronales Artificiales.

Los métodos desarrollados se implementaron dentro de un software de monitoreo denominado *Analizador PQ*. Este software, permite detectar, clasificar y caracterizar eventos de calidad de energía presentes en señales provenientes de un entorno de simulación producidas por el mismo programa, o señales provenientes de la red eléctrica.

Finalmente se presenta una comparación entre los diferentes sistemas de clasificación desarrollados donde se detallan las ventajas de cada método.

ABSTRACT

Recently, Power Quality (PQ) problems in Electrical Power Systems become a very important issue not only for utilities but also for customers. In addition, due to the increasing energy demand, a new concept known as Smart Grid has been established to allow the integration of conventional generation with clean renewable and distributed energy sources. The non stationary nature of the renewable energy and the operation of the distribution generation in order to integrate them into the electrical grid, produce considerable problems related to the PQ issue.

Therefore, the process of detection, classification, characterization, propagation and localization of the PQ signals, and the way of mitigation their effect, becomes a relevant matter.

The aim of this thesis is to analyze different signals processing techniques to detect and extract useful information of an electromagnetic event. In addition, a comparison between different power quality classification methods based on learning machines algorithms is presented in this thesis.

For this reason a software tool that allows the detection, localization classification of PQ events was developed.

First the theoretical concepts are detailed, then several techniques based on the Wavelet Transform to detect PQ events are developed. Then, classification methods based on Support Vector Machines are presented in order to automatic classify different events.

All developed methods are integrated into a software tool named PQ analyzer. This tool allows detecting, classifying and characterizing real electrical signals and simulated ones.

Finally, the different methodologies developed are compared in order to discuss the advantages that they present.

Introducción

Calidad de Energía

Si bien existen numerosas definiciones del término *Calidad de Energía (Power Quality)*, la que más se adapta al contenido presentado en el presente trabajo es la siguiente:

“Se entiende por eventos de calidad de energía a toda desviación en los parámetros que gobiernan las formas de onda (de tensión o de corriente) en relación a una señal ideal.”[1].

De aquí en adelante se utilizará el término *perturbación* como todo evento que provoque una alteración en las formas de onda de la señal deteriorando la calidad del servicio eléctrico.

Existe una gran variedad de fenómenos que provocan eventos electromagnéticos no deseados en un Sistema Eléctrico de Potencia (SEP). Algunos de ellos son:

- Fallas o cortocircuitos en la red.
- Conexión intempestiva de grandes cargas.
- Conexión de bancos de condensadores.
- Dispositivos de conmutación de estado sólido.
- Cargas altamente no lineales.
- Cargas electrónicamente conmutadas.
- Sistemas de potencia no balanceados.
- Sistemas de control de iluminación.
- Computadoras y centros de procesamiento.
- Bombas hidráulicas.
- Rectificadores e inversores industriales.

Dentro de las fuentes generadoras de este tipo de fenómenos electromagnéticos, tal vez la más importante sea la incorporación de dispositivos electrónicos a los SEP. Este tipo de tecnología se utiliza cada vez con mayor frecuencia aunque, por su naturaleza no lineal, representan una inherente fuente de eventos de calidad de energía. Por otro lado, la familia de dispositivos electrónicos sufre graves alteraciones en su rendimiento y/o funcionamiento ante la presencia de perturbaciones en la red eléctrica donde se desempeñan. Paradójicamente, los equipos que generan en mayor grado los eventos de calidad de energía son los que más los sufren.

Las perturbaciones en un sistema eléctrico generan una innumerable cantidad de problemas, la mayoría de los cuales se traduce en pérdidas económicas tanto para la empresa proveedora de energía como para el usuario final. Dentro de las consecuencias

de una pobre calidad de energía se pueden mencionar: el funcionamiento de las máquinas eléctricas por debajo de su rendimiento óptimo; el detrimento en la vida útil de los dispositivos sensibles a los eventos de calidad de energía; y las frecuentes reparaciones a las que estos dispositivos se ven sometidos. Además, es importante destacar que un equipo dañado o funcionando en niveles inaceptables puede ser causante de la salida de servicio de una línea de producción de una fábrica, lo que también deviene en una importante pérdida monetaria para el usuario. Por este y otros motivos, el estudio de los eventos de calidad de energía ha sido una materia que ha cobrado mucho interés en estos últimos años.

Por otro lado, debido a los crecientes niveles de demanda de energía a nivel global, uno de los principales desafíos que una red eléctrica afronta a futuro es el de satisfacer éste incremento con una energía limpia, confiable, sustentable y de bajo costo. Ésta problemática motivó a la incorporación de un concepto nuevo denominado *Redes Eléctricas Inteligentes (Smart Grids)* cuyo objetivo apunta a la generación de nuevos productos, servicios y procesos que mejoren la eficiencia de las redes eléctricas actuales y permitan la integración de la red eléctrica convencional con nuevas fuentes de energía distribuida, limpia y renovable. La naturaleza no estacionaria de la energía renovable y las maniobras involucradas en la integración de las fuentes de energía distribuida con la red tradicional, generan una innumerable cantidad de problemas relacionados con la calidad de energía eléctrica. Por este motivo el desarrollo de las redes eléctricas inteligentes deberá necesariamente desarrollar marcos de trabajo y estándares para asegurar que la integración y migración hacia las redes del futuro no afecte la calidad de la energía percibida por el usuario.

Sistemas de monitoreo

La principal estrategia adoptada por la industria para la disminución del impacto de las perturbaciones en un SEP es la incorporación de una entidad capaz de detectar, clasificar y localizar eventos de calidad de energía. Usualmente a esta entidad se la conoce como *Sistema de monitoreo*. Un sistema de monitoreo está compuesto por equipamiento de adquisición de datos, bases de datos, algoritmos de disparo, un sistema de comunicación, sistemas de procesamiento y análisis de los datos y un sistema de visualización de la información.

Como se aprecia en la Figura 1 un sistema de procesamiento está compuesto por tres bloques funcionales. El sub sistema de detección, el sub sistema de clasificación y, finalmente, el sub sistema de caracterización. Como su nombre lo indica, el objetivo del bloque de detección es localizar la una perturbación dentro de un conjunto de muestras y extraer una serie de parámetros característicos de las muestras para ser utilizados en el proceso de clasificación. El proceso de clasificación se encarga de tipificar en diferentes clases los parámetros obtenidos en el proceso de detección. El sub sistema de caracterización tiene la finalidad de generar parámetros que indican el estado del nodo

que se está monitoreando en función de la cantidad y características de los fenómenos electromagnéticos detectados y clasificados por los dos sub bloques que le preceden.

El avance de la tecnología electrónica permite implementar en la actualidad algoritmos que involucran herramientas matemáticas complejas pero de gran utilidad para lograr los objetivos propuestos. El estudio de estas técnicas de procesamiento digital, permite obtener una gran cantidad de información sobre el estado del sistema en tiempo real, por lo que la selección de las mejores técnicas para monitorear las perturbaciones, es fundamental para tener una calidad de potencia aceptable en el SEP.



Figura 1 - Procesamiento y análisis de las señales

Este trabajo se enfoca en el *Procesamiento y análisis de las muestras* obtenidas por medio de la adquisición de mediciones de un SEP.

Objetivos

El objetivo del presente trabajo de tesis es estudiar distintas técnicas de procesamiento de señales para la detección y extracción de parámetros característicos de una perturbación. Además se plantea como objetivo la comparación de diferentes máquinas de aprendizaje para la clasificación de perturbaciones electromagnéticas de diferente naturaleza.

Para ello se desarrolló una herramienta informática que permita la detección, localización y clasificación de un conjunto de fenómenos electromagnéticos en las formas de onda adquiridas de un SEP. La herramienta desarrollada está compuesta por un módulo destinado a la detección, localización y extracción de parámetros característicos empleando la *Transformada Wavelet (WT)*, y un módulo para la clasificación de los eventos de calidad de energía basado en *Redes Neuronales Artificiales (ANN)* y *Máquinas de Vectores de Soporte (SVM)*.

La herramienta desarrollada posee la capacidad de procesar señales simuladas así como también señales provenientes de mediciones reales.

Finalmente, un objetivo adicional es generar una base de conocimientos para el desarrollo de una herramienta que pueda ser implementada para la detección, localización y clasificación de fenómenos electromagnéticos en tiempo real en un SEP.

El desarrollo de la herramienta permitirá a docentes de las cátedras relacionadas con el estudio de los SEP el contar con una herramienta pedagógica que podría ser empleada para el dictado de las materias.

Para el desarrollo de la herramienta propuesta se diseñó una plataforma basada en MATLAB© que permite simular señales perturbadas, incorporar formas de onda que contengan eventos de calidad de energía adquiridas desde un sistema real y la comparación de la performance de los modelos desarrollados para la obtención de resultados y conclusiones.

Organización de la tesis

La tesis se divide en dos cuerpos principales: La primera parte aborda los estándares involucrados en el monitoreo de un SEP, así como los fundamentos teóricos y matemáticos de diferentes métodos de procesamiento de señales y técnicas de clasificación de patrones, que luego son utilizados en el método propuesto. En la segunda parte se presenta la herramienta desarrollada para la detección, localización y clasificación de perturbaciones en un SEP.

Primera parte – Fundamentación teórica

Para establecer una normativa que caracterice los tipos de anomalías que se pueden presentar en un SEP y las estrategias de monitoreo que se deben llevar a cabo para realizar las mediciones, diferentes organismos han desarrollado sus propios estándares. El que se adoptará en este trabajo, es el “*IEEE Std 1159-2009*” [2]. El *Capítulo 1* de esta tesis presenta los aspectos más destacados del mencionado estándar.

Tal vez los componentes más importantes de un sistema de monitoreo sean sus algoritmos específicamente diseñados para detectar, clasificar y localizar perturbaciones provenientes de mediciones de la red eléctrica. Estos algoritmos son desarrollados como una combinación de métodos de disparo, códigos de decisiones basados en reglas, técnicas de procesamiento de señales y máquinas de aprendizaje lineal, las que, mancomunadamente, permiten recabar información de una red eléctrica y afrontar con mayor conocimiento las causas que generan una disminución de la calidad de energía.

Dentro de las técnicas de procesamiento de señales, la más ampliamente difundida y adoptada en la comunidad científica avocada al estudio de los fenómenos eléctricos es la *Transformada Wavelet*. Esta herramienta permite un análisis simultáneo en el dominio del tiempo y de la frecuencia, lo que facilita la tarea de la detección y localización en el tiempo de fenómenos electromagnéticos en un SEP. El *Capítulo 2* de este trabajo abordará conceptualmente la *Transformada Wavelet* y se presentará una reseña de los trabajos más importantes relacionados a la utilización de ésta técnica en el área de Calidad de Energía.

En el *Capítulo 3* se presenta una estrategia para la clasificación de patrones dentro de un conjunto de datos. La técnica seleccionada, conocida como *Redes Neuronales Artificiales (ANN, por sus siglas en inglés)*, se trata de una técnica de procesamiento en paralelo, compuesto por unidades computacionales denominadas neuronas que, al igual que el cerebro humano, son capaces de adquirir conocimiento a través de la experiencia. Las ANN representan una buena alternativa para el reconocimiento y clasificación de

patrones; sin embargo, presentan una serie de desventajas como por ejemplo: el tiempo que toma el proceso de adiestramiento; el atascamiento del entrenamiento en un mínimo local; la carencia de un sustento matemático que permita el establecimiento de reglas para definir su arquitectura. El *Capítulo 3* abordará un tipo de Red Neuronal en particular denominado *feed forward backpropagation* que será posteriormente empleado, tal como se presenta en el *Capítulo 5*, para el desarrollo del sistema de procesamiento de señales de un SEP. Finalmente éste capítulo presentará una serie de artículos donde los autores proponen la utilización de las redes neuronales para la clasificación de perturbaciones.

Debido a las mencionadas desventajas de las ANN, en el *Capítulo 4* se presentará una técnica de máquinas de aprendizaje que se fundamenta en dos bases sólidas: *La teoría de aprendizaje estadístico* y *la optimización matemática*. Este método, denominado *Máquinas de Vectores de Soporte (SVM, por sus siglas en inglés)* presenta una serie de ventajas frente a las ANN, dentro de las cuales se destacan: su gran habilidad de generalización; su sólido fundamento matemático; la eliminación del problema del posible atascamiento en un mínimo local; los escasos parámetros empleados para realizar su configuración. En el *Capítulo 4* también se presentará un breve recorrido por las publicaciones científicas que adoptaron a los SVM como método de clasificación de perturbaciones en un SEP.

Segunda parte – Desarrollo y prueba de un sistema de monitoreo

En la segunda parte del trabajo de tesis se presentan los algoritmos desarrollados para la detección y clasificación de eventos de calidad de energía y los resultados obtenidos. Los métodos de procesamiento de señales utilizados en el desarrollo presentado en esta segunda parte del informe se sustentan en el marco teórico y matemático presentado en los capítulos que conforman la primera parte de la presente tesis.

Los objetivos del desarrollo del software se listan a continuación:

- Implementar las técnicas de procesamiento de señales y clasificación de patrones presentadas en los capítulos 2, 3 y 4 de este trabajo.
- Desarrollar una herramienta académica para la investigación de distintas técnicas y algoritmos de análisis de eventos de calidad de energía en un sistema eléctrico de potencia.
- Comparar el rendimiento de las diferentes técnicas de clasificación.
- Desarrollar un software que pueda ser utilizado en un sistema de monitoreo de eventos de calidad de energía en tiempo real.

En el *Capítulo 5* se describen los algoritmos desarrollados en MATLAB®. Este software permite la generación de señales sintéticas para el entrenamiento y prueba de los algoritmos de clasificación; la adquisición de señales reales provenientes de una base de datos; la detección de eventos de calidad de energía utilizando técnicas basadas

en la *Transformada Wavelet*; la clasificación de dichos eventos utilizando *Redes Neuronales Artificiales* y *Máquina de Vectores de Soporte*.

En el *Capítulo 6* se presentan los resultados de las experiencias realizadas para la comparación entre las diferentes técnicas abordadas en este trabajo.

Primera Parte:
Conceptos Teóricos

Capítulo 1: Calidad de Energía y Sistemas de Monitoreo

1.1 INTRODUCCIÓN

La calidad de la energía en un Sistema Eléctrico de Potencia (SEP) se ha vuelto una materia de extrema importancia tanto para las compañías proveedoras y distribuidoras de energía eléctrica como también para los clientes. Los clientes se han vuelto cada día menos tolerantes a las perturbaciones en la energía eléctrica ya que el detrimento en el rendimiento de los equipos industriales genera cada año millonarias pérdidas económicas.

Los equipamientos implementados por los clientes son los mayores generadores de perturbaciones en los sistemas de distribución pero, paradójicamente, estos equipos son cada vez más sensibles a dichas perturbaciones.

Para mejorar la calidad de la energía se ha desarrollado una normativa que estandariza los tipos de anomalías que se pueden presentar en un SEP. En la sección 1.2.4 del presente capítulo se presenta un resumen de los estándares que definen los tipos de perturbaciones según los parámetros característicos de las formas de onda y las posibles causas que pueden provocar cada tipo de evento en una red eléctrica.

Con el objetivo de disminuir el impacto que los eventos de calidad de energía tienen en un sistema eléctrico de potencia, se han desarrollado herramientas para el monitoreo de la calidad de energía. Estas herramientas permiten la identificación, localización y clasificación de un evento en tiempo real. En la sección 1.3 se presentan los aspectos más importantes relacionados a un *sistema de monitoreo de eventos de calidad de energía*.

1.2 CALIDAD DE ENERGIA - DESCRIPCION DE LA PROBLEMÁTICA

1.2.1 Definición de Calidad de Energía

En la comunidad científica no existe una unicidad de criterios a la hora de dar una definición de lo que el término calidad de energía abarca. Según el Instituto de Ingenieros Electrónicos y Electricistas (IEEE) *calidad de energía se refiere a un concepto de alimentación y puesta a tierra de equipamiento sensible en términos si los mismos son adecuados para el funcionamiento de dicho equipamiento*. Para la Comisión Internacional de Electrotecnia (IEC) *calidad de energía se define como Las características de la electricidad en un determinado punto del sistema eléctrico*. La primera de las definiciones aquí presentadas hace referencia a la energía que alimenta a los equipos sensibles a eventos de calidad de energía, mientras que la segunda se enfoca en los puntos de medición.

Para el desarrollo que en esta tesis se presenta, la definición más apropiada es aquella que se enfoca en los eventos de calidad de energía. *Se entiende por eventos de calidad de energía, tanto en las formas de onda de tensión como las de corriente, a toda desviación de los parámetros que gobiernan a estas señales tomando como referencia a una señal senoidal ideal*.

Una señal de tensión ideal se entiende como una forma de onda sinusoidal cuya frecuencia y amplitud son constantes y se corresponden con sus valores nominales. De la misma manera, una forma de onda ideal de corriente también se considera como una señal sinusoidal cuyos parámetros de frecuencia y magnitud se corresponden con sus valores nominales, son constantes y, adicionalmente, la frecuencia y fase de la corriente se deben corresponder con la frecuencia y fase de la señal de tensión. Cualquier desviación tanto de corriente o voltaje de la forma de onda ideal se conoce como *evento de calidad de energía*.

1.2.2 Interés en Calidad de Energía

En los últimos años la comunidad científica involucrada en el estudio de los sistemas eléctricos de potencia ha demostrado un creciente interés en la investigación de los fenómenos de calidad de energía. Si bien existen muchas razones que justifican este fenómeno aquí solo se detallan algunas de las más importantes.

Las industrias cada vez se han vuelto más dependientes de los equipamientos eléctricos. Por otro lado, dicho equipamiento se ha vuelto día a día más sensible a las perturbaciones en la forma de onda de tensión. En contraposición a estas tendencias, las industrias se han vuelto menos tolerantes a la detención de sus líneas de producción por lo que el interés de las compañías en contar con una energía libre de perturbaciones se ha incrementado drásticamente en las últimas décadas.

A pesar de ser más sensibles a los eventos de calidad de energía, estos equipos son uno de los principales generadores de perturbaciones en la forma de onda de la corriente. El

incremento en el empleo de convertidores electrónicos de energía eléctrica agrava el problema de las distorsiones generando un amplio espectro de perturbaciones en un SEP.

La desregulación de los mercados de la energía eléctrica ha generado un incremento en la necesidad de contar con indicadores de calidad de energía. Los clientes, a su vez demandan mayor información acerca del estado de la red, cosa que en el pasado no ocurría.

La incorporación de la generación distribuida y energía renovable a la red eléctrica ha creado nuevas fuentes de perturbaciones.

La incorporación de equipos de consumo eficiente se ha transformado en una fuente importante de problemas de calidad de energía. Dispositivos de velocidad ajustable y lámparas de bajo consumo son ejemplos de equipos que introducen distorsiones en las formas de onda de la corriente y, a su vez, son dispositivos altamente sensibles a dichas perturbaciones.

1.2.3 Clasificación de Eventos de Calidad de Energía

Los principales motivos por los cuales es recomendable clasificar los eventos en diferentes categorías se deben a que existen diferentes maneras de realizar mediciones, resolver problemas, extraer parámetros característicos o establecer métodos de clasificación que dependen del tipo o clase de evento en cuestión.

Existen diversos criterios para la clasificación de eventos de calidad de energía de acuerdo a la variación o los parámetros que difieren de la forma de onda ideal. En este trabajo se presenta un criterio de división de acuerdo a la duración de la perturbación y luego se abordan los criterios establecidos por la IEC y la IEEE.

1.2.4 Clasificación de acuerdo a la duración

Una importante división dentro de los tipos de perturbaciones, es aquella que se realiza teniendo en cuenta la duración del evento. Existen fenómenos de carácter estacionario (o cuasi estacionario) que persisten por períodos de tiempo mucho mayores a los períodos de análisis. En tal caso, la perturbación suele llamarse *variación* [1]. Un ejemplo de una variación es una alteración en la frecuencia fundamental de la forma de onda de tensión.

El segundo tipo de variaciones, conocidas como *perturbaciones repentinas*, son eventos que comienzan y finalizan dentro de un período corto de tiempo que usualmente coincide con el tiempo de análisis (también llamado *ventana de análisis*). Una caída de tensión por debajo de un umbral predeterminado, durante un tiempo reducido es un ejemplo de *perturbación repentina* también conocida como *perturbación de corta duración*.



Esta clasificación es importante a la hora de definir metodologías y parámetros característicos para la detección, clasificación y caracterización de perturbaciones en un SEP. Un claro ejemplo de esto es el método empleado para la ejecución de los algoritmos de detección y clasificación. Mientras las perturbaciones del tipo estacionarias no requieren un disparador, los eventos de corta duración necesitan un método de disparo para ejecutar los mecanismos de registro y análisis de la información.

1.2.5 Clasificación de acuerdo a la IEC.

En la IEC 61000-2-5 1995 [3] los eventos electromagnéticos se clasifican de acuerdo a lo mostrado en la Tabla 1.1.

Tabla 1.1 - Principales fenómenos que causan perturbaciones electromagnéticas

Grupo	Ejemplos
Fenómenos de baja frecuencia - Conducidos	Armónicos, Interarmónicos
	Señales de sistema (Power Line Carrier)
	Fluctuaciones de Voltaje
	Huecos e Interrupciones de Voltaje
	Desbalance de Voltaje
	Variaciones en la frecuencia
	Voltajes inducidos de baja frecuencia
Tensiones de continua en redes de corriente alterna.	
Fenómenos de baja frecuencia – Radiados	Campos Magnéticos
	Campos Eléctricos
Fenómenos de alta frecuencia - Conducidos	Señales de continua inducidas (Corriente o Voltaje)
	Transitorios unidireccionales
	Transitorios oscilatorios
Fenómenos de alta frecuencia – Radiados	Campos Magnéticos
	Campos Eléctricos
	Campos Electromagnéticos
	Ondas continuas
	Transitorios
Fenómenos de descarga electro estática (ESD)	
Pulsos Electromagnéticos de origen Nuclear (NEMP)	

1.2.6 Clasificación de acuerdo a la IEEE

En la Tabla 1.2 se detalla lo propuesto por el estándar 1159-2009 de la IEEE [2]. Esta normativa trata sobre las prácticas recomendadas para el monitoreo de calidad de energía eléctrica.

El estándar propone una clasificación en 7 grandes grupos de acuerdo a parámetros como la duración de la perturbación, su contenido espectral y su magnitud típica.

En las siguientes secciones se describirán las características distintivas de cada una de las categorías.

Debido a que el presente trabajo se enfoca en el procesamiento de señales para la detección y clasificación de perturbaciones, aquí no se profundizará en las causas que los generan, ni en el estudio de las consecuencias que estos eventos traen aparejados en un sistema eléctrico de potencia.

Tabla 1.2 - Categorías y Características de fenómenos electromagnéticos en un SEP

Categoría	Contenido espectral típico	Duración típica	Magnitud de tensión típica
1.0 Transitorios			
1.1 Impulsivo			
1.1.1 nanosegundos	5 nseg (levantamiento)	< 50 nanoseg	
1.1.2 microsegundos	1 microseg (levantamiento)	50 nanoseg - 1 miliseg	
1.1.3 milisegundos	0.1 miliseg (levantamiento)	> 1 miliseg	
1.2 Oscilatorio			
1.2.1 Baja frecuencia	< 5 KHz	0.3 - 50 miliseg	0 - 4 pu*
1.2.2 Media frecuencia	5 - 500 KHz	20 microseg	0 - 8 pu
1.2.3 Alta frecuencia	0.5 - 5 MHz	5 microseg	0 - 4 pu
2.0 Variaciones de corta duración			
2.1 Instantáneo			
2.1.1 Sag		0.5 - 30 ciclos	0.1 - 0.9 pu
2.1.2 Swell		0.5 - 30 ciclos	1.1 - 1.8 pu
2.2 Momentáneo			
2.2.1 Interrupción		0.5 ciclos - 3 seg	< 0.1 pu
2.2.2 Sag		30 ciclos - 3 seg	0.1 - 0.9 pu
2.2.3 Swell		30 ciclos - 3 seg	1.1 - 1.4 pu
2.3 Temporánea			
2.3.1 Interrupción		3 seg - 1 min	< 0.1 pu
2.3.2 Sag		3 seg - 1 min	0.1 - 0.9 pu
2.3.3 Swell		3 seg - 1 min	1.1 - 1.2 pu
3.0 Variaciones de larga duración			
3.1 Interrupción sostenida		> 1 min	0.0 pu
3.2 Subtensiones		> 1 min	0.8 - 0.9 pu
3.3 Sobretensiones		> 1 min	1.1 - 1.2 pu
4.0 Desbalance de tensión		estado estacionario	0.5 - 2 %
5.0 Distorsión de la forma de onda			
5.1 Offset de continua		estado estacionario	0 - 0.1 %
5.2 Armónicos	Orden de armónico : 0 - 100	estado estacionario	0 - 20 %
5.3 Interarmónicos	0 - 6 KHz	estado estacionario	0 - 2 %
5.4 Muestras		estado estacionario	
5.5 Ruido	Banda ancha	estado estacionario	0 - 1 %
6.0 Fluctuaciones de tensión	< 25 Hz	intermitente	0.1 - 7 %
7.0 Variaciones de la frecuencia fundamental		< 10 seg	

*Nota: La cantidad pu se refiere a una cantidad por unidad (per unit) que es adimensional. La cantidad 1.0 pu se corresponde al 100% del valor nominal.

1.2.6.1 Transitorios

a) Transitorios impulsivos (en inglés *Impulsive Transient*)

Un transitorio impulsivo es un cambio súbito y unidireccional (positivo o negativo) en la condición de estado estable de la tensión, la corriente, o ambos y de frecuencia diferente a la del sistema eléctrico de potencia. Un transitorio impulsivo usualmente es caracterizado por su tiempo de subida y el tiempo de decaimiento. Este tipo de fenómeno puede producirse por una descarga eléctrica o por fuentes de iluminación.

b) Transitorios oscilatorios (en inglés *Oscillatory Transient*)

Un transitorio oscilatorio es un cambio repentino y bidireccional de la condición de estado estacionario de la tensión, corriente, o ambos, que incluye valores positivos y negativos. Según la Tabla 1.2 los transitorios oscilatorios se pueden clasificar como de alta, media y baja frecuencia dependiendo del contenido espectral del evento.

Los transitorios oscilatorios con una componente de frecuencia primaria mayor a los 500 [KHz] y una duración típica medida en micro segundos (varios ciclos de la frecuencia fundamental) son considerados como *transitorios oscilatorios de alta frecuencia*. Estos fenómenos se producen mayoritariamente por alguna maniobra de conmutación, o bien resultan de una respuesta oscilatoria de un sistema ante una excitación del tipo transitorio impulsivo.

Un transitorio cuyo componente de frecuencia primaria se encuentre entre los 5 y los 500 [KHz] y cuya duración sea de decenas de microsegundos (varios ciclos de la frecuencia fundamental) se considera como un *transitorio oscilatorio de media frecuencia*. La energización de un arreglo *back to back* de capacitores resulta en corrientes oscilatorias cuya frecuencia supera la decena de [KHz].

Un transitorio con un componente de frecuencia primaria que no supera los 5 [KHz] y cuya duración va de los 0.3 a los 50 [ms] se considera como un *transitorio oscilatorio de baja frecuencia*. Esta categoría de fenómenos se encuentran en los sistemas de sub transmisión y distribución y son causados por diversos tipos de eventos, principalmente por la energización de un banco de capacitores.

La Figura 1.1 muestra un transitorio oscilatorio de alta frecuencia y la Figura 1.2 muestra un transitorio de baja frecuencia.

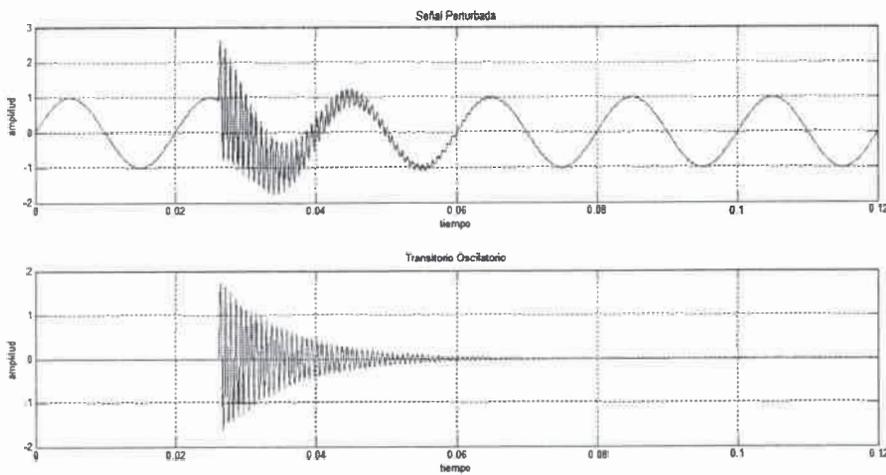


Figura 1.1 - Transitorio Oscilatorio de alta frecuencia

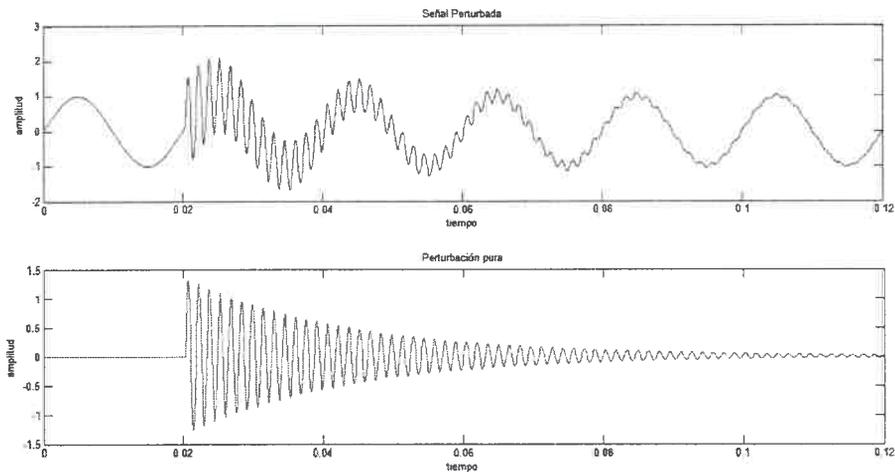


Figura 1.2 - Transitorio Oscilatorio de baja frecuencia.

1.2.6.2 Variaciones de corta duración

El estándar divide a las variaciones de corta duración en tres categorías: *Huecos* (en inglés *Sag**), *Subas* (en inglés *Swell*) e *Interrupción* (en inglés *Outage*). Cada una de estas categorías se subdividen de acuerdo a la duración como: *Instantáneas*, *Momentáneas* o *Temporarias*. La mayoría de las variaciones de corta duración son causadas por condiciones de falla o la alimentación de grandes cargas que requieren altas corrientes de arranque. Otro motivo se puede encontrar en la pérdida intermitente de conexión con el cableado de alimentación.

*Nota: Sag es la terminología empleada por el estándar propuesto por la IEEE. Un término equivalente comúnmente empleado por la IEC es "Voltage Dip".

a) Interrupciones momentáneas y temporales (en inglés *Outage*)

Una interrupción ocurre cuando el voltaje de alimentación o la corriente de carga disminuye a un valor inferior al $0.1 [p.u.]$ por un período de tiempo que no excede un minuto. Las interrupciones pueden ser el resultado de fallas en los sistemas eléctricos, falla en los equipos, o el funcionamiento inadecuado de los sistemas de control. El tiempo de inicio para la medición de una interrupción tiene lugar cuando la magnitud del voltaje cae por debajo del 10% del valor nominal.

Las interrupciones, generadas por una falla, generalmente se ven precedidas por un hueco de tensión cuya duración está relacionada con el tiempo de reacción de los dispositivos de protección. La duración de la interrupción dependerá mayoritariamente de la capacidad de re cierre de los dispositivos de protección. La Figura 1.3 muestra una interrupción ideal ya que los tiempos de decaimiento y de subida son nulos.

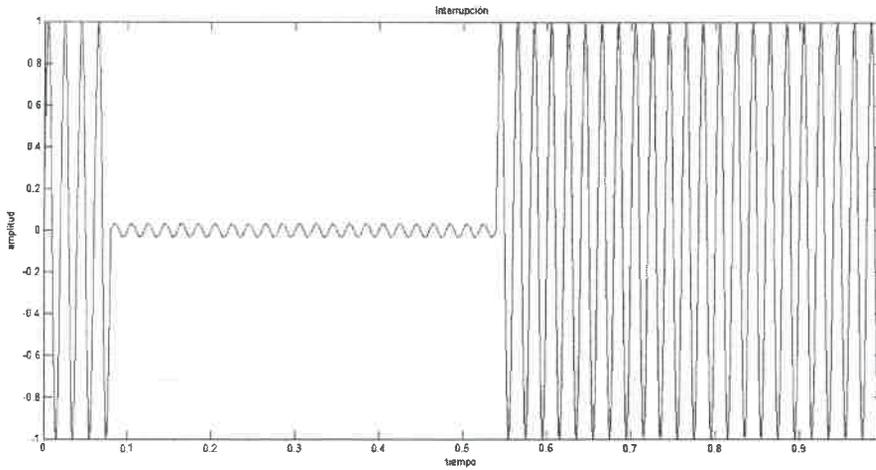


Figura 1.3 - Interrupción ideal.

La tensión (o su valor rms) generalmente no cae a cero en forma instantánea ante una operación de apertura de un interruptor, debido a las cargas inductivas y/o motores conectados a la red. Esto se produce por ejemplo cuando hay conectada una significativa cantidad de motores de inducción en un SEP. Estos efectos pueden observarse en la Figura 1.4 (a) y (b).

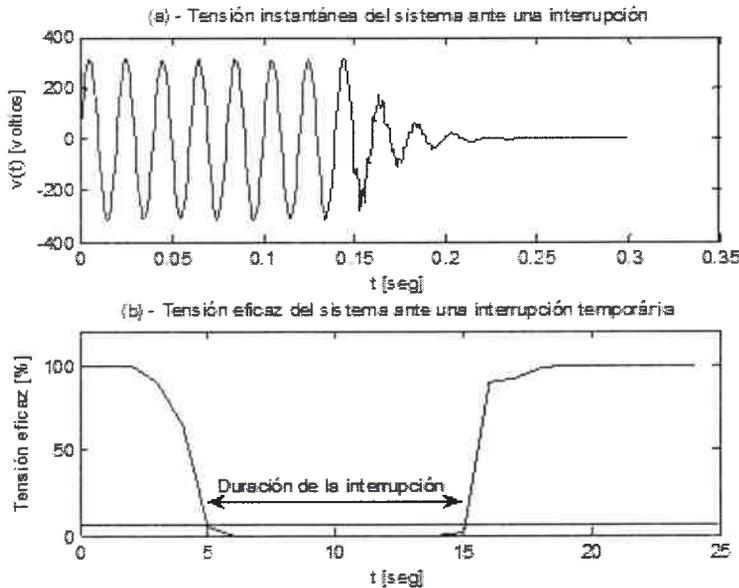


Figura 1.4 (a) Variación de la tensión instantánea ante una Interrupción. (b) Variación del valor eficaz

b) Huecos de tensión (en inglés Voltage Sag)

Un hueco de tensión es una caída en el valor de la amplitud del voltaje entre 0.1 [p.u.] y 0.9 [p.u.] por un tiempo que va desde los 0.5 ciclos hasta un minuto. La

terminología utilizada para describir un hueco de tensión suele ser algo ambigua. Un hueco del 20% puede significar como resultado de una caída del voltaje de $0.8 [p.u.]$ o de $0.2 [p.u.]$ dependiendo si los términos empleados hacen referencia al voltaje resultante o a la caída de voltaje. El estándar propone utilizar el voltaje resultante para ser empleado como terminología común. En este caso, un hueco de tensión del 80% significa una perturbación que resulta en una tensión de $0.8 [p.u.]$.

Los huecos de tensión se dan como resultado de una falla en el sistema, de la conmutación de grandes cargas o el encendido de motores grandes.

La duración de un hueco de tensión ha sido definida desde los $2 [ms]$ (aproximadamente la décima parte de un ciclo) a varios minutos en diferentes publicaciones. Caídas de tensión cuya duración no supera la mitad de un ciclo no pueden ser caracterizadas efectivamente por un cambio en el valor eficaz de la frecuencia fundamental. Por otro lado, caídas de tensión que duran más de un minuto son clasificadas como eventos de larga duración.

La Figura 1.5 (a) muestra un Hueco de tensión ideal ya que las transiciones ocurren en un tiempo nulo mientras que en la Figura 1.5 (b) se puede ver el valor eficaz instantáneo de la señal.

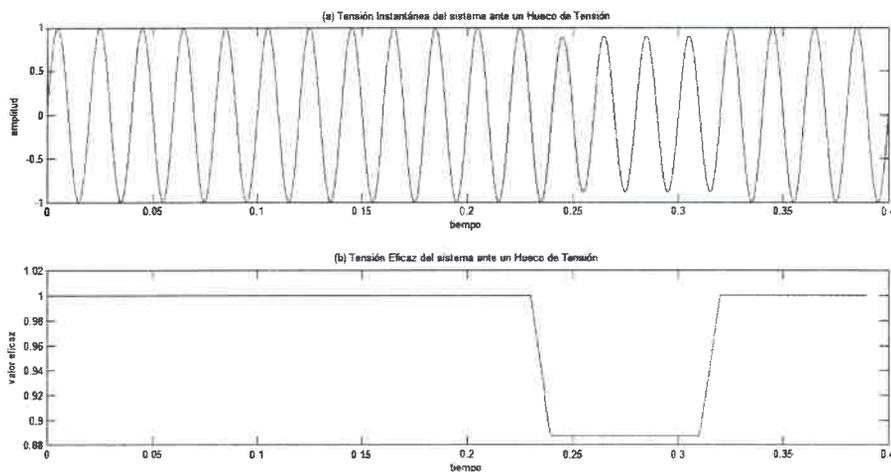


Figura 1.5 - (a) Tensión instantánea del sistema ante un Hueco de tensión. (b) Tensión eficaz del sistema ante un Hueco de tensión.

c) *Sobretensión (en inglés Voltage Swell)*

Una sobretensión es un incremento en el valor rms de la señal de voltaje por encima del valor $1.1 [p.u.]$ por un tiempo que puede ir desde los 0.5 ciclos hasta 1 minuto. Las magnitudes típicas de sobretensión van desde 1.1 a $1.2 [p.u.]$.

Al igual que los huecos de tensión, las sobretensiones suelen estar asociadas con condiciones de fallas del sistema aunque este tipo de perturbaciones son mucho menos comunes. Las sobretensiones suelen también ser causadas por la conmutación de grandes cargas, divisores de cargas o grandes bancos de capacitores.

La Figura 1.6 (a) muestra una sobretensión ideal ya que los tiempos de elevación y decaimiento del evento ocurren en tiempos idealmente nulos. La Figura 1.6 (b) ilustra el valor eficaz instantáneo de la señal.

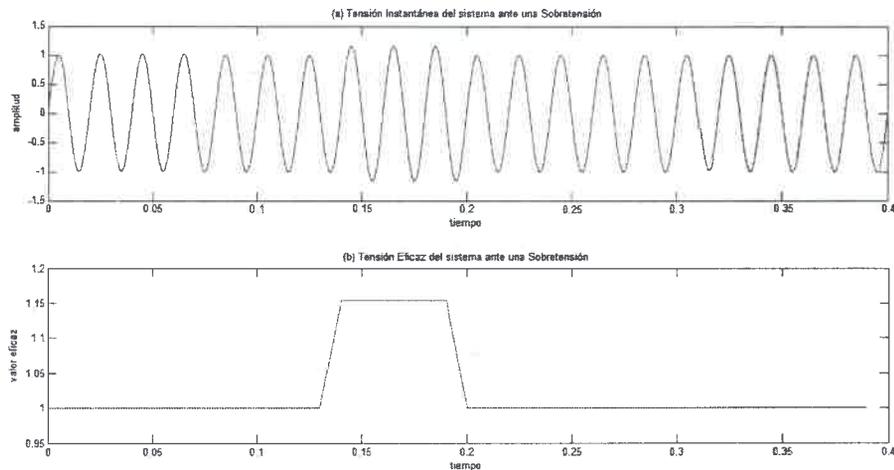


Figura 1.6 - (a) Tensión Instantánea del sistema ante una Sobretensión. (b) Tensión Eficaz del sistema ante una Sobretensión.

La severidad de una sobretensión durante una condición de falla es función de la localización de la falla, la impedancia del sistema y del método de puesta a tierra del sistema.

1.2.6.3 Variaciones de larga duración

Las variaciones de larga duración se corresponden con desviaciones del valor eficaz de una señal por un tiempo superior al minuto. En la Tabla 1.2 se puede ver que las variaciones de larga duración pueden tratarse tanto de sobretensiones como subtensiones y generalmente son causadas por variaciones en el sistema y operaciones de conmutación. A continuación se abordan los distintos tipos de variaciones de larga duración.

a) Sobretensión (En inglés *Overvoltage*)

Una sobretensión es un incremento del valor eficaz de una tensión por un valor superior a 1.1 [p.u.] cuya duración supera el minuto. Los valores típicos de la sobretensión se encuentran entre 1.1 [p.u.] y 1.2 [p.u.]. Las sobretensiones pueden ser el resultado de la

conmutación de cargas, la conmutación de bancos de capacitores o la configuración incorrecta del tap de un transformador.

b) Subtensión (En Inglés Undervoltaje)

Una caída prolongada en la tensión es un decremento del valor eficaz de tensión por un valor inferior a 0.9 [p.u.] por un tiempo superior a un minuto. Los valores típicos de las caídas de tensión se encuentran entre 0.8 [p.u.] y 0.9 [p.u.].

Algunas de las causas de una subtensión pueden ser: la conexión de grandes cargas, la desconexión de bancos de capacitores o la sobrecarga de los circuitos.

c) Interrupciones sostenidas (en inglés Sustained Interruptions)

La caída de tensión por un valor menor al 10% del valor nominal y por un período de tiempo que supera el minuto se considera como una interrupción sostenida. Interrupciones que superan el minuto usualmente son consideradas como permanentes ya que requieren de la intervención manual para restablecer las condiciones normales de operación.

1.2.6.4 Desbalance de tensión

Se define como desbalance de un sistema trifásico a la relación entre la componente de la secuencia negativa respecto de la positiva, expresada en porcentaje. Esta definición puede aplicarse tanto para el voltaje como para la corriente. El desbalance típico de un SEP se encuentra por debajo del 3% medidos en tensión.

1.2.6.5 Distorsión de la forma de onda

Los fenómenos clasificados como perturbaciones de distorsión de forma de onda, de acuerdo a lo detallado en la Tabla 1.2, se clasifican en: *Componente de Continua, Armónicos, Interarmónicos, Muecas y Ruido*. Cada una de estas categorías se explican brevemente a continuación:

a) Corrimiento de continua (en inglés DC offset)

La presencia de componentes de continua en un sistema de corriente y tensión alterna se denomina como corrimiento de continua. Este fenómeno ocurre como resultado de perturbaciones geomagnéticas o debido a la presencia de rectificadores de media onda en el Sistema Eléctrico de Potencia. Este tipo de perturbaciones pueden generar pérdidas, debido al incremento de la saturación en los transformadores y su calentamiento asociado.

b) Armónicos (en inglés Harmonics)

Los armónicos han existido desde que se adoptó a la corriente alterna como medio de transmisión de la energía eléctrica. Sin embargo este efecto se ha hecho más significativo en estos últimos tiempos debido al incremento de los dispositivos no lineales. Un dispositivo no lineal produce corrientes no sinusoidales aún cuando la tensión lo sea. En el caso ideal, las fuentes de tensión son sinusoidales. Las corrientes distorsionadas originadas en las distintas cargas no lineales del sistema producen caídas de tensión distorsionadas aún cuando las impedancias del sistema son consideradas lineales. Estas caídas de tensión, sumadas a la fuente ideal sinusoidal, reproducen en las barras de carga tensiones que finalmente son distorsionadas.

Los dispositivos, cuyo funcionamiento se basan en electrónica de potencia, son los mayores contribuyentes de armónicos en los SEP. La forma de caracterizar la distorsión armónica es mediante un gráfico que contenga la magnitud y fase de cada componente armónica del sistema. En un típico gráfico del espectro de frecuencias de una señal con contenido armónico, se puede apreciar una fuerte presencia de energía en las frecuencias correspondientes a múltiplos enteros de la frecuencia fundamental del sistema. Existe un parámetro, denominado *Distorsión Total Armónica (por sus siglas en inglés THD)*, que es comúnmente empleado para cuantificar la distorsión armónica presente en la red.

La Figura 1.7 a) Muestra la tensión instantánea de un sistema contaminado con segundo, tercer, cuarto y quinto armónico. En la Figura 1.7 b) se puede ver el resultado de la transformada de Fourier de la señal contaminada con armónicos.

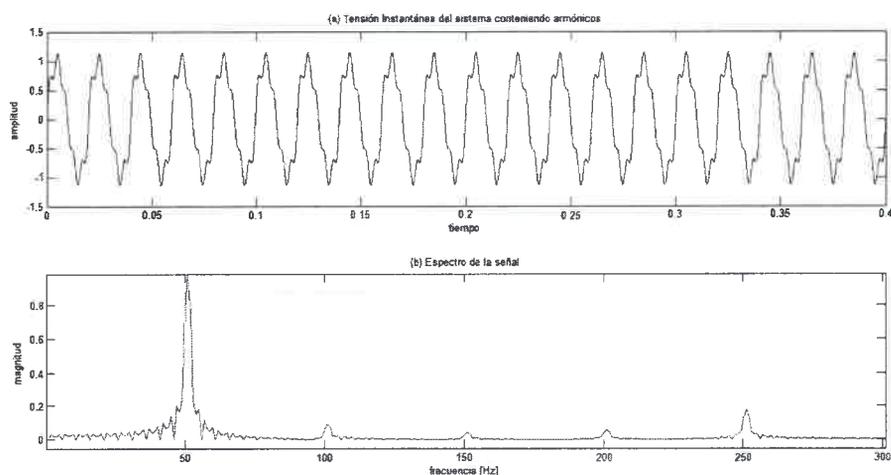


Figura 1.7 - a) Tensión instantánea del sistema conteniendo Armónicos. b) Espectro de la señal.

c) *Interarmónicos (en inglés Interharmonics)*

Se denominan interarmónicos a aquellos componentes del espectro de frecuencia de la señal perturbada que no se corresponden con múltiplos enteros de la frecuencia fundamental del sistema eléctrico.

Las principales causas de los interarmónicos son los convertidores estáticos de frecuencia, los ciclo convertidores, los hornos de inducción y los dispositivos de arco, especialmente aquellos cuyo control no se encuentra apropiadamente sincronizado con la frecuencia del sistema.

d) *Muestras (en inglés Notch)*

Las muescas son una perturbación periódica provocada por la operación de dispositivos que emplean la electrónica de potencia cuando la corriente se conmuta de una fase a otra. Este fenómeno representa un caso especial ya que es periódico, por lo que podría interpretarse como un fenómeno de estado estacionario, y a su vez presenta componentes de alta frecuencia. Esto hace difícil enmarcarlo como un transitorio de corta duración o una distorsión de forma de onda. Los convertidores trifásicos (inversores) son los mayores generadores de este tipo de perturbación en un sistema de potencia.

La Figura 1.8 muestra una señal contaminada con una perturbación del tipo muesca. En esta figura se puede apreciar la periodicidad de la perturbación.

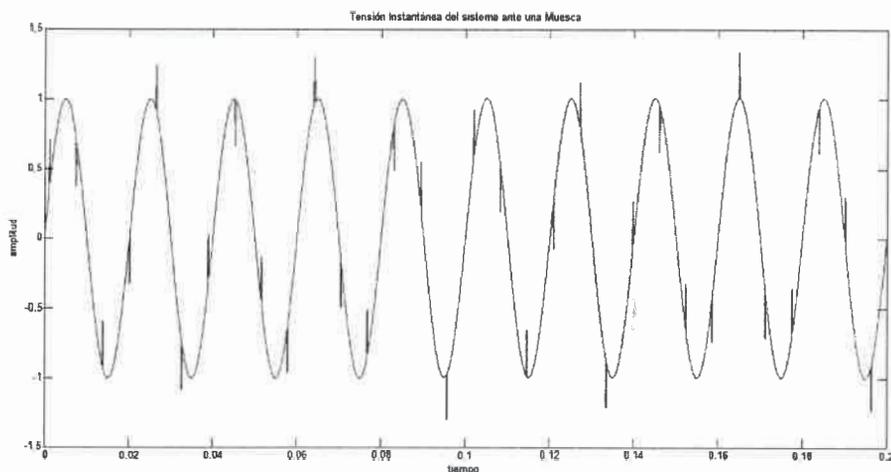


Figura 1.8 - Tensión instantánea de un sistema ante una Muesca

e) *Ruido (En inglés Noise)*

Un ruido es una señal eléctrica no deseada, con una amplia composición espectral, típicamente con valores máximos de $200 [KHz]$, que se agrega a la señal de tensión o corriente de manera aditiva. Se define ruido a toda distorsión

no deseada que no puede ser clasificada como distorsión armónica o perturbaciones del tipo transitorios. El ruido puede ser generado por equipos de electrónica de potencia, circuitos de control, equipos que utilizan arcos voltaicos, cargas con rectificadores de estado sólido y fuentes de energía conmutada.

En la Figura 1.9 se puede ver una señal de tensión de un sistema eléctrico de potencia contaminada con una perturbación del tipo *Ruido*.

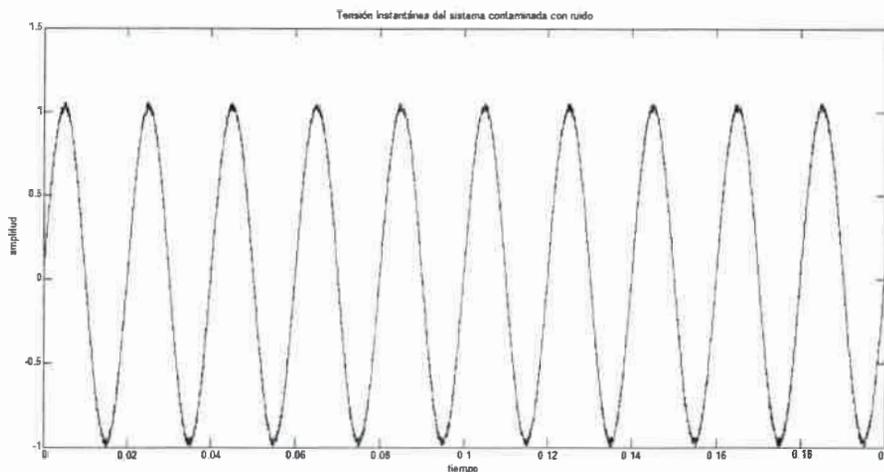


Figura 1.9 - Tensión instantánea del sistema contaminada con Ruido

1.2.6.6 Fluctuaciones de tensión

Las fluctuaciones de tensión son variaciones sistemáticas de la envolvente de la señal del voltaje o una serie de cambios aleatorios en la magnitud, que normalmente no exceden el rango comprendido entre los $0.95 [p.u.]$ y los $1.05 [p.u.]$.

Cualquier carga que tenga importantes variaciones cíclicas, especialmente su componente reactiva, puede producir fluctuaciones de voltaje. Usualmente suele emplearse el término flicker de manera errónea para denominar a las fluctuaciones de tensión. Mientras que las fluctuaciones de tensión se refieren a un fenómeno electromagnético, el término flicker está relacionado al resultado no deseado de estas variaciones percibidas por el ojo humano como una fluctuación en la intensidad lumínica de una lámpara.

Los hornos de arco eléctrico son los generadores más comunes de este tipo de fenómeno.

Debido a que las fluctuaciones de tensión suelen aparecer como una modulación en la magnitud de la forma de onda fundamental (similarmente a lo que ocurre en la modulación AM de una señal de radio). Se puede definir la magnitud de la fluctuación como la magnitud rms de la señal modulante. Esto puede fácilmente

obtenerse mediante la demodulación de la envolvente a través de la remoción de la frecuencia fundamental y mediante la medición de la magnitud de la componente moduladora.

La Figura 1.10 muestra la tensión instantánea de un sistema eléctrico de potencia ante la presencia de fluctuaciones en su magnitud.

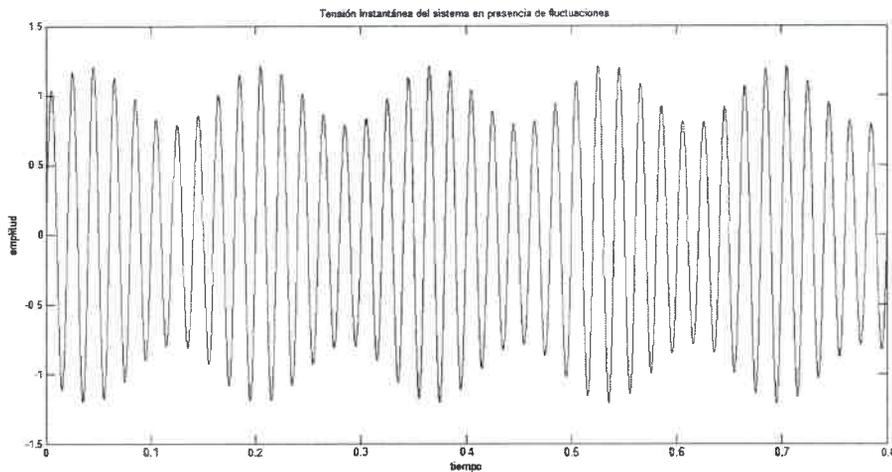


Figura 1.10 - Tensión instantánea del sistema ante Fluctuaciones.

1.2.6.7 Variaciones de la frecuencia fundamental

Este tipo de fenómeno se explica como una desviación de la frecuencia nominal del sistema eléctrico de potencia (normalmente $50 [Hz]$ o $60 [Hz]$). La frecuencia fundamental en estado estacionario está directamente relacionada con la velocidad de rotación de los generadores del sistema. En cualquier instante de tiempo, la frecuencia fundamental depende del balance entre la carga y la capacidad disponible de generación. Cuando este balance varía, pequeños cambios en la frecuencia pueden ocurrir. La desviación de la frecuencia y su duración son variables que dependen de la carga y de la capacidad de respuesta del sistema de generación ante las modificaciones en la carga.

1.2.7 Modelos empleados para la simulación de los eventos de calidad de energía

En la Tabla 1.3 se resumen los modelos matemáticos empleados en este trabajo para la simulación de las formas de onda perturbadas. Estos modelos fueron propuestos por Panigrahi y Pandi (2009) [4].

Tabla 1.3 - Modelo de perturbaciones.

Perturbaciones	Modelo	Variación de los parámetros
Hueco de tensión	$x(t) = A\{1 - \alpha [u(t - t_1) - u(t - t_2)]\} \text{sen}(\omega t)$ $t_2 > t_1 \text{ y } u(t) = \begin{cases} 1 & \geq 0 \\ 0 & \leq 0 \end{cases}$	$0.05 \leq \alpha \leq 0.85$ $T \leq t_2 - t_1 \leq 7T$
Sobre tensión	$x(t) = A\{1 + \alpha [u(t - t_1) - u(t - t_2)]\} \text{sen}(\omega t)$ $t_2 > t_1 \text{ y } u(t) = \begin{cases} 1 & \geq 0 \\ 0 & \leq 0 \end{cases}$	$0.05 \leq \alpha \leq 0.8$ $T \leq t_2 - t_1 \leq 7T$
Armónicos	$x(t) = A[\alpha_1 \text{sen}(\omega t) + \alpha_2 \text{sen}(2\omega t)$ $+ \alpha_3 \text{sen}(3\omega t) + \alpha_5 \text{sen}(5\omega t)$ $+ \alpha_7 \text{sen}(7\omega t)]$	$\alpha_1 = 1$ $\alpha_2 - \alpha_7 = 0.0 - 0.3$
Fluctuaciones	$x(t) = A\{1 + \alpha_f [\text{sen}(\beta_f \omega t)]\} \text{sen}(\omega t)$	$0.01 \leq \alpha_f \leq 0.25$ $2[\text{Hz}] \leq f_f \leq 8[\text{Hz}]$
Transitorio Oscilatorio	$x(t) = A\{[\text{sen}(\omega t) + b e^{-\gamma(t-t_1)}] \text{sen}(\omega_{tr}(t-t_1))\}$	$-2 \leq b \leq 2$ $50 \leq \gamma \leq 100$ $500[\text{Hz}] \leq f_{tr} \leq 1500[\text{Hz}]$
Interrupción	$x(t) = A\{1 - \alpha [u(t - t_1) - u(t - t_2)]\} \text{sen}(\omega t)$ $t_2 > t_1 \text{ y } u(t) = \begin{cases} 1 & \geq 0 \\ 0 & \leq 0 \end{cases}$	$0.88 \leq \alpha \leq 0.85$ $T \leq t_2 - t_1 \leq 7T$
Armónicos y Hueco	$A\{1 - \alpha [u(t - t_1) - u(t - t_2)]\} [\alpha_1 \text{sen}(\omega t)$ $+ \alpha_2 \text{sen}(2\omega t)$ $+ \alpha_3 \text{sen}(3\omega t) + \alpha_5 \text{sen}(5\omega t)$ $+ \alpha_7 \text{sen}(7\omega t)]$	$0.05 \leq \alpha \leq 0.85$ $T \leq t_2 - t_1 \leq 7T$ $\alpha_1 = 1$ $\alpha_2 - \alpha_7 = 0.0 - 0.3$
Armónicos y Sobretensión	$A\{1 - \alpha [u(t - t_1) - u(t - t_2)]\} [\alpha_1 \text{sen}(\omega t)$ $+ \alpha_2 \text{sen}(2\omega t)$ $+ \alpha_3 \text{sen}(3\omega t) + \alpha_5 \text{sen}(5\omega t)$ $+ \alpha_7 \text{sen}(7\omega t)]$	$0.05 \leq \alpha \leq 0.8$ $T \leq t_2 - t_1 \leq 7T$ $\alpha_1 = 1$ $\alpha_2 - \alpha_7 = 0.0 - 0.3$

1.3 SISTEMA DE MONITOREO DE EVENTOS DE CALIDAD DE ENERGIA

1.3.1 Necesidades de monitoreo

En la mayoría de las disciplinas tecnológicas, el primer paso para resolver cualquier problema radica en la detección de la fuente que lo origina. El monitoreo de un SEP permite la recolección de información de las características del estado de la red desde el punto de vista del comportamiento de las variables más importantes que la gobiernan en condiciones de operación normal y/o transitoria [1]. Las mencionadas variables se corresponden a los valores de amplitud, frecuencia y fase de las formas de onda de tensión y corriente. Los datos recolectados son procesados con el objetivo de obtener información acerca de los tipos de perturbaciones que se presentan en el SEP, las posibles causas que las generan y las acciones que se deben tomar para mitigarlas.

Según las definiciones del estándar IEEE Std 1159-2009 *Prácticas recomendadas para el monitoreo de calidad de energía* [2] el monitoreo de un sistema eléctrico de potencia es una herramienta imprescindible para detectar, clasificar, caracterizar y localizar los fenómenos electromagnéticos.

No existe un único y común objetivo para el monitoreo de las variables que gobiernan un SEP. En algunos casos, el monitoreo se realiza para diagnosticar incompatibilidades entre las fuentes de energía y la carga; en otros casos el monitoreo es empleado para evaluar las condiciones eléctricas de una ubicación particular con el fin de refinar técnicas de modelado para desarrollar normas y referencias de calidad de energía. La elección de las técnicas, los equipamientos de medición, los métodos de recolección de información, los disparadores, los umbrales y las técnicas de análisis de datos a emplear se deben realizar teniendo en cuenta el objetivo del sistema de monitoreo. Por ejemplo, si un sistema eléctrico presenta reiteradas salidas de servicio, el propósito del monitoreo de la red puede ser el de detectar las perturbaciones que generan condiciones de operación no previstas en equipamientos sensibles. En este caso, el equipamiento y las técnicas de medición deben ser seleccionados de modo que permitan la detección de las condiciones que provocan que los equipos funcionen fuera de los rangos de tolerancia exigidos por el o los fabricante/s.

Sin importar los objetivos técnicos involucrados en la implementación de un sistema de monitoreo, el principal motivo que fundamenta su implementación radica en las *pérdidas económicas generadas por los eventos de calidad de energía*. Los efectos en el equipamiento y en los procesos de una industria pueden incluir la mala operación, el daño del equipamiento y la detención de una línea productiva.

Todos estos efectos tienen como consecuencia importantes pérdidas económicas tanto para el usuario como para la empresa proveedora de energía eléctrica. El recambio de un equipo dañado, las pérdidas económicas que se generan por mantener una línea productiva fuera de operación, la incorrecta medición de la energía consumida, el detrimento de la vida útil de las instalaciones, etc., son algunas de las pérdidas

económicas producidas por eventos de calidad de energía que un usuario y la empresa proveedora deben afrontar.

1.3.2 Reseña Histórica

En la década de 1920 General Electric desarrolló lo que denominó una grabadora de arco. Este artefacto estaba conformado por un motor que movía una tira de papel desde un tambor hacia otro. Un par de electrodos generaban un arco a través del papel que dejaba unas marcas proporcionales a las mediciones realizadas. Este dispositivo era capaz de grabar en el papel información cualitativa de la amplitud de una señal así como el tiempo y la fecha en que el evento era registrado. Un técnico capacitado debía ser capaz de interpretar las marcas en el papel y en sus habilidades se basaba la inteligencia del sistema de monitoreo.

En los comienzos de la década de 1960 Martzloff aplicó un arreglo inteligente con una cámara fotográfica y un osciloscopio analógico de alta persistencia para registrar eventos de calidad de energía. En este caso, lo que se modificó es la forma en que las mediciones eran registradas, pero los datos necesariamente debían ser interpretados visualmente por un operario por lo que la inteligencia de este dispositivo también recaía en la pericia de los técnicos que operaban el artefacto.

En los 70 aparecieron los equipos de monitoreo de eventos de calidad de energía que proporcionaban indicadores basados en texto, caracterizando una perturbación de acuerdo a su tipo y duración. Estos equipos gradualmente comenzaron a incorporar herramientas para la visualización gráfica y resúmenes estadísticos de los eventos capturados.

A mediados de 1980 los instrumentos comenzaron a contar con la capacidad de muestreo en el orden de los mega Hertz [MHz] para proveer información detallada de los transitorios de altas frecuencias.

Desde la década del 2000 a la actualidad los dispositivos de monitoreo incorporan procesadores digitales cada día con mayor potencia de cálculo. Además la capacidad de almacenamiento crece continuamente, la integración con una computadora ya está estandarizada y pueden encontrarse en el mercado cualquier interfaz de comunicación para incorporar los dispositivos de monitoreo a los distintos tipos de red. Estos adelantos se realizaron con el objetivo de mejorar la interoperabilidad de los equipos y fabricantes mediante la estandarización de interfaces, base de datos e instrumentación para permitir mayor flexibilidad y reducir costos.

1.3.3 Clasificación de sistemas de monitoreo

Una tipificación comúnmente empleada para los sistemas de monitoreo es la división entre equipos portables y equipos fijamente instalados. La implementación de cada uno de ellos depende de cada necesidad en particular.

- Equipos portables: Para solucionar problemas en instalaciones edilicias, industrias o eventos puntuales en sistemas de distribución, se suelen emplear dispositivos de medición portable ya que las necesidades de almacenamiento de información no son determinantes.
- Equipos de monitoreo permanentemente: Estos equipos son empleados para la gestión de grandes sistemas eléctricos de potencia donde se necesita realizar la recolección de información de rendimiento y confiabilidad de las instalaciones eléctricas para proveer funcionalidades de alarmas ante la aparición de eventos de calidad de energía. Para este tipo de soluciones, el sistema de monitoreo suele ser integrado con los sistemas de control de procesos para correlacionar información de los eventos de calidad de energía y determinar su impacto en los procesos productivos.

De acuerdo a la complejidad y a las funcionalidades proporcionadas por los sistemas de monitoreo se puede realizar la siguiente clasificación:

- Indicadores de eventos: Los indicadores de eventos son los más simples y baratos de los dispositivos de monitoreo de perturbaciones. Estos dispositivos son capaces de clasificar varios tipos de eventos mostrando su ocurrencia a través de luces indicadoras, gráficos de barra, alarmas audibles o una combinación de estas tres funcionalidades. Este tipo de dispositivos generalmente no es capaz de almacenar el tiempo de inicio y duración del evento detectado.
- Osciloscopios: Los osciloscopios digitales pueden almacenar formas de onda de voltaje y corriente. Algunos osciloscopios digitales permiten el cálculo de parámetros básicos como el valor pico, el valor medio, el valor eficaz, entre otros. Las formas de onda se almacenan manteniendo un sincronismo temporal entre las señales de tensión y corriente. Generalmente los osciloscopios son diseñados para que el usuario intervenga después de realizar una medición y no son capaces de capturar y almacenar repetitivamente eventos de calidad de energía de manera automática.
- Sistemas de monitoreo de calidad de energía: Un sistema de monitoreo de eventos de calidad de energía está conformado por una serie de dispositivos permanentemente instalados que cumplen con la función de adquisición y procesamiento de información.

Debido a que el presente trabajo de tesis se enfoca en los sistemas de monitoreo (principalmente en la etapa de procesamiento de información) este tipo de arreglo será abordado más profundamente en la siguiente sección.

1.3.4 Sistemas de monitoreo de calidad de energía

La Figura 1.11 muestra un diagrama funcional de un sistema de monitoreo de calidad de energía.

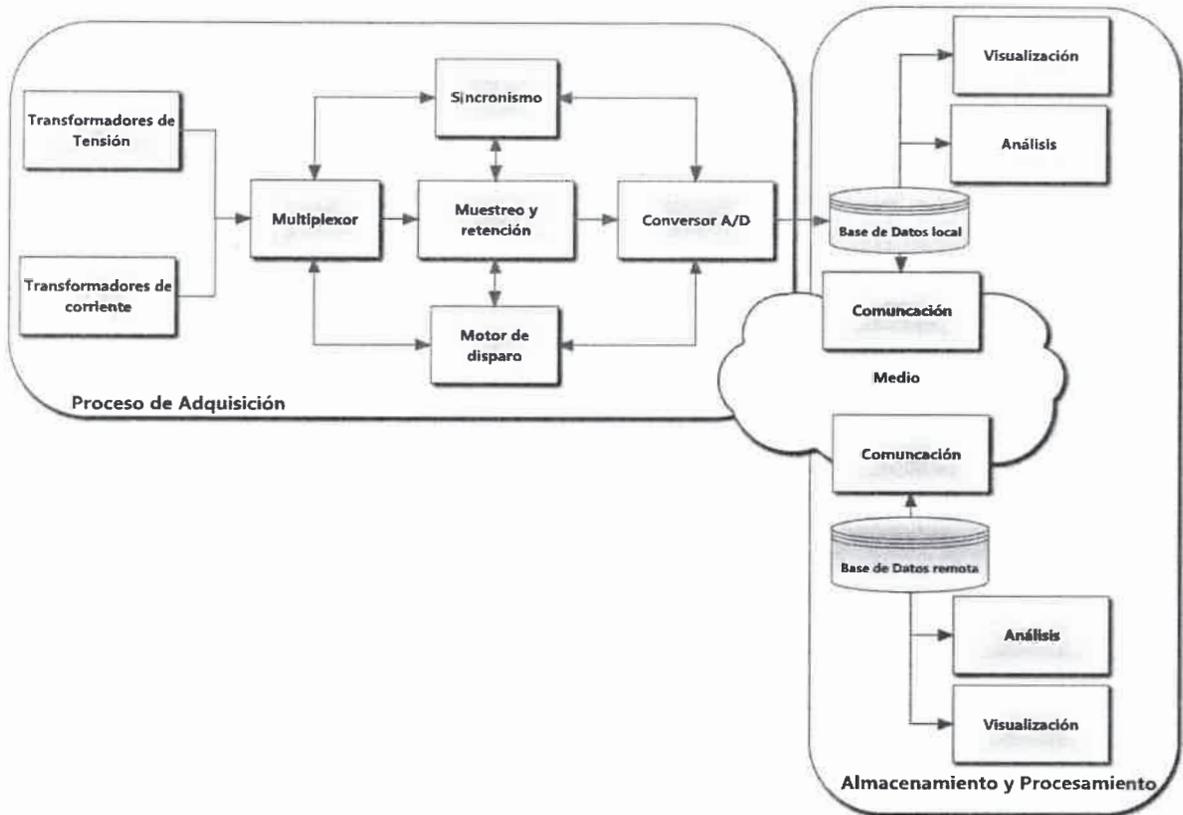


Figura 1.11 - Sistema de Monitoreo [2].

Según lo que se muestra en la Figura 1.11 un sistema de monitoreo está conformado por dos bloques de procesos:

- Procesos de adquisición de datos: Aunque el presente trabajo se enfoca en el estudio de diferentes técnicas de procesamiento de señales, es importante destacar que un sistema de monitoreo involucra equipamiento de adquisición de datos cuyas prestaciones son determinantes para el procesamiento de la información. Esta etapa del sistema de monitoreo está conformada por transformadores de potencia, transformadores de corriente, transductores, muestreadores, convertidores Analógico/Digital, dispositivos de sincronismo, multiplexores y complejos algoritmos de disparo.
- Procesos de análisis, almacenamiento y visualización de los datos: Esta etapa de un sistema de monitoreo está conformada por el hardware y software de almacenamiento, procesamiento y visualización de la información relacionada con los eventos de calidad de energía en un sistema eléctrico de potencia. La Figura 1.11 muestra que las actividades relacionadas con el

análisis y visualización de la información se puede realizar localmente o de manera remota empleando redes de comunicaciones. El proceso de *Análisis* de la información obtiene la información de la base de datos generada por el *proceso de adquisición de datos*, extrae y calcula parámetros relacionados a los eventos de calidad de energía (tiempos de ocurrencia, duración, magnitud, energía, etc.) y prepara la información resultante del análisis para ser almacenada y/o graficada.

Este trabajo de tesis se enfoca principalmente en el análisis y comparación de diferentes técnicas de procesamiento de datos para la detección y clasificación de eventos de calidad de energía. Por este motivo, en la sección 1.4, se abordará con más detalle el proceso *Análisis* mostrado en la Figura 1.11.

1.4 ESTADO DEL ARTE: PROCESO DE ANALISIS DE LOS DATOS

Se dice que los datos son información sin contexto. La adquisición de señales es una materia importante para un sistema de monitoreo, pero más importante aún es transformar esas señales en información útil para el usuario y presentarla de manera precisa y amigable. La mayoría de los instrumentos de monitoreo colectan información de manera continua con altas velocidades de muestreo. Si no existiera una herramienta para extraer información de los datos colectados los sistemas de almacenamiento rápidamente colapsarían.

El proceso de *Análisis* de los datos obtenidos del muestreo de las formas de onda, cumple la importante función de extraer información útil de los datos colectados por el proceso de adquisición.

El proceso de *Análisis* está conformado por tres etapas que se muestran en la Figura 1.12



Figura 1.12 - Análisis de las muestras.

1.4.1 Detección de eventos de calidad de energía

Los eventos de calidad de energía se presentan de manera aleatoria y rara vez su ocurrencia puede ser predicha. Debido a la naturaleza azarosa de las perturbaciones un sistema de monitoreo debe continuamente muestrear y almacenar información de las formas de onda aunque estas no contengan perturbación alguna.

Para que la base de datos del sistema de monitoreo no se vea inundada de información no relevante, es preciso realizar el procesamiento de las muestras para descartar la información que describa al sistema en un estado de operación normal y almacenar solo aquellos datos que representan una perturbación. Por este motivo, un sistema de monitoreo cuenta con módulo de detección cuya función principal es la de identificar la presencia de una perturbación dentro de un conjunto de muestras almacenadas en una base de datos.

A su vez, el módulo de detección, es el encargado de extraer parámetros que caracterizan una perturbación minimizando la cantidad de datos a ser almacenados y procesados por el módulo de clasificación.

Las investigaciones realizadas en el área proponen distintos métodos para la detección de perturbaciones y la extracción de parámetros que caracterizan al evento de calidad de

energía. A continuación se realizará un recorrido de las publicaciones más importantes relacionadas con la detección de anomalías en un sistema eléctrico de potencia.

La Transformada de Fourier ha sido históricamente empleada para el análisis del contenido espectral de una señal, independientemente de su naturaleza. Roscoe; Burt, y McDonald (2009) proponen el uso de la *Transformada de Fourier* para la medición de perturbaciones estacionarias del tipo armónicos [5].

Heydt, Fjeld, Liu y Pierce (1999) justifican el uso de la *Transformada de Fourier dependiente del tiempo* para la detección de eventos de calidad de energía de corta duración [6]. Por otro lado, Zhang, Geng y Yuan (2001) emplean la Transformada ventaneada de Fourier para disminuir el error introducido por la *Transformada rápida de Fourier* (por sus siglas en inglés FFT) en la medición de armónicos [7].

Soo-Hwan Cho, Gilsoo Jang y Sae-Hyuk Kwon (2010) emplean la Transformada *Gabor-Wigner* para la detección de fenómenos estacionarios y de corta duración [8].

La técnica más extensamente aceptada por la comunidad científica se basa en la *Transformada Wavelet*. El trabajo de Santoso, Powers, Grady y Hofmann, (1996) fue uno de los pioneros en el uso la Transformada Wavelet para la detección de perturbaciones en un sistema eléctrico de potencia [9].

La Transformada Wavelet Discreta Modificada, comúnmente denominada *Transformada S*, fue propuesta por Stockwell, Mansinha y Lowe (1996) como una alternativa a la Transformada Wavelet Discreta [10]. Posteriormente Dash, Panigrahi y Panda (2003) emplearon esta técnica como una herramienta potente para la detección, localización y clasificación de problemas relacionados con la calidad de energía [11]. Siguiendo la misma línea de investigación Fengzhan y Rengang (2007) demostraron que la Transformada S es un método rápido, eficiente, escalable y muy poco sensible al ruido [12].

En el trabajo propuesto por Radil, Ramos y Cruz Serra (2008) se discute la técnica denominada *Ajuste senooidal* con mínimos cuadrados para la detección de perturbaciones y el cálculo de la distorsión armónica total de una señal perturbada [13].

Finalmente se puede mencionar la publicación realizada por Caujolle, Petit, Fleury y Berthet (2010) en la que se presenta un método de detección basado en técnicas de filtrado de Kalman [14].

En el Capítulo 2 de este trabajo se profundizará en el estudio de la Transformada Wavelet como método para la detección de perturbaciones en un sistema eléctrico de potencia y la extracción de parámetros que resultan de utilidad para la etapa de clasificación.

1.4.2 Clasificación de eventos de calidad de energía

La etapa de detección es la encargada de descubrir la presencia de una anomalía en una señal de tensión o corriente, pero es el proceso de clasificación el que realiza el diagnóstico acerca del tipo de perturbación presente en las muestras. El proceso de clasificación, por lo tanto, relaciona un conjunto de datos de entrada con un conjunto de patrones de perturbaciones conocido.

Existen aplicaciones en donde el clasificador emplea la totalidad de las muestras de la señal para decidir el tipo de evento presente en dicha señal. Sin embargo, una alternativa es analizar un conjunto reducido de datos (de dimensión inferior al número de muestras) provenientes de la etapa de detección usualmente conocidos como parámetros característicos de la perturbación.

A continuación se realizará una breve reseña de las técnicas empleadas por la comunidad científica para la clasificación de perturbaciones en un sistema eléctrico de potencia.

Tradicionalmente las *Redes Neuronales Artificiales* han sido ampliamente utilizadas en los procesos de reconocimiento de patrones. Santoso, Powers, Grady y Parsons (2000) implementaron un sistema de monitoreo que combina la Transformada Wavelet con las *Redes Neuronales Artificiales* [15]. Existen muchos aportes que utilizan los distintos tipos de redes neuronales para la clasificación de eventos de calidad de energía por lo que los mismos serán abordados con más profundidad en el Capítulo 3 de este trabajo.

Jaehak, Powers, Grady y Bhatt (2002) propusieron un método de clasificación que combina un sistema basado en reglas, la Transformada Wavelet y el *Modelo oculto de Markov* [16].

Una combinación de técnicas basadas en *Redes Neuronales* y *Lógica Difusa* fue discutido en el trabajo de Jiansheng, Negnevitsky y Nguyen (2002) [17]

En el trabajo de Reaz, Choong, Sulaiman (2007) se presenta un esquema que emplea la Transformada Wavelet para la extracción de parámetros característicos de la señal perturbada, las *Redes Neuronales Artificiales* para la clasificación de patrones y un post-procesamiento basado en *Lógica Difusa* para mejorar la precisión del proceso de clasificación [18].

En su trabajo Panigrahi y Pandi (2009) presentan un sistema automático de clasificación que emplea un algoritmo genético denominado fuzzy k-nearest neighbour para seleccionar las características que mejor caracterizan a un conjunto de perturbaciones.

En los últimos años la comunidad científica ha mostrado creciente interés en la aplicación de las *Máquinas de Soporte Vectorial* (Support Vector Machine) para la clasificación de eventos de calidad de energía. En su publicación Whei-Min, Chien-Hsien, Chia-Hung y Fu-Sheng (2008) presentan un sistema de monitoreo que utiliza el paquete de Transformada Wavelet para la detección, y las *Máquinas de Soporte*

Vectorial (SVM por sus siglas en inglés) para la clasificación de eventos de calidad de energía [19].

Debido a que las Máquinas de Soporte Vectorial presentan muchas ventajas frente a la mayoría de métodos de clasificación de patrones, muchos aportes en esta área han sido realizados en los últimos años. En el Capítulo 4 de este trabajo se abordará con mayor profundidad el empleo de esta técnica en un sistema de monitoreo de eventos de calidad de energía.

1.4.3 Caracterización de eventos de calidad de energía

La etapa de caracterización consiste en la generación de parámetros que proporcionen una medida del estado de las ondas de tensión y corriente en un sistema eléctrico de potencia. Esta información puede ser empleada por los fabricantes para diseñar los componentes de la red eléctrica, por los entes reguladores para establecer indicadores de calidad de energía, por los usuarios para realizar reclamos o bien por las empresas proveedoras de energía para efectivizar penalizaciones. El módulo de caracterización es el encargado de calcular estos indicadores basándose en las señales perturbadas.

El estudio las técnicas de caracterización de los eventos de calidad de energía se encuentran fuera del alcance de los objetivos del presente trabajo, razón por la cual esta problemática no se abordará en profundidad.

1.5 CONCLUSIONES

Los conceptos que aquí se abordaron demuestran que el monitoreo de eventos de calidad de energía en un sistema eléctrico de potencia representa una herramienta fundamental para disminuir las pérdidas económicas para las empresas proveedoras de energía eléctrica y para sus clientes.

Debido a la amplia variedad de fenómenos electromagnéticos que se pueden encontrar en un SEP es necesario adoptar un estándar que defina los tipos de perturbaciones que se pueden encontrar así como también las causas que generan las perturbaciones, los instrumentos empleados para medirlas y las prácticas recomendadas para limitar sus efectos. Este capítulo, a su vez, presenta los aspectos más importantes del estándar más reciente que la IEEE recomienda.

Un sistema de monitoreo está conformado por una etapa de adquisición de datos, una etapa de análisis, procesamiento, almacenamiento y visualización de la información y una etapa de comunicación (para aquellos sistemas de monitoreo distribuido en grandes áreas geográficas). En este capítulo se explicaron los aspectos básicos de cada etapa realizando un enfoque más profundo en el estudio de diversas técnicas para el procesamiento de la información obtenida por la etapa de adquisición de datos.

En la última parte de este capítulo se presentó una revisión bibliográfica de la metodología utilizada por la comunidad científica para el procesamiento de las señales perturbadas con el fin de cotejar las técnicas adoptadas en este trabajo con los métodos más comúnmente utilizados.

Capítulo 2: Transformada Wavelet

2.1 INTRODUCCIÓN

En el análisis de señales existe un gran número de herramientas que se han ido desarrollando con el paso de los años, entre las que se destaca, sin lugar a dudas, la Transformada de Fourier. Sin embargo, en los últimos 20 años, se han desarrollado nuevas herramientas que permiten analizar señales desde otra perspectiva, surgiendo sobre todo de la necesidad de poder analizar señales que no se comportan de manera estacionaria o que presentan cambios bruscos en intervalos de tiempo muy pequeños.

Como se presentó en el *Capítulo 1* del presente trabajo de tesis existe un gran número de perturbaciones eléctricas que poseen la característica de ser no estacionarias.

Como se abordará en el presente capítulo, la *Transformada Wavelet* es una técnica de procesamiento de señales que permite el análisis de una señal simultáneamente en el dominio del tiempo y la frecuencia. Esta característica hace que la Transformada Wavelet sea ideal para la detección de perturbaciones cuyo carácter sea no estacionario.

Generalmente los artículos o libros que abordan la *Transformada Wavelet* lo hacen desde el punto de vista matemático, el cual es altamente complejo.

El objetivo de este capítulo es presentar los conceptos más importantes relacionados con la *Transformada Wavelet*, y las características que hacen que esta herramienta sea la adecuada para la detección de fenómenos electromagnéticos en un sistema eléctrico de potencia. Los conceptos aquí desarrollados servirán para comprender la función que la *Transformada Wavelet* cumple en un sistema de monitoreo.

En la primera sección de este capítulo se presenta la necesidad de contar con una herramienta de análisis de señales que permitan extraer información en el dominio del tiempo y la frecuencia de manera simultánea. Se explicarán las herramientas de procesamiento de señales tradicionales como la *Transformada de Fourier* y la *Transformada de Fourier de tiempo corto* para demostrar que estos métodos presentan falencias a la hora de procesar señales no estacionarias.

En la segunda sección del capítulo se aborda la Transformada Wavelet desde un punto de vista práctico. Se presentará la *Transformada Wavelet Continua (CWT)*, la *Transformada Wavelet Discreta (DWT)* y la representación alternativa de la *DWT* como un banco de filtros seguidos por un proceso de diezmado.

Finalmente se presentará una revisión de los trabajos más destacados que utilizan la *Transformada Wavelet* para el análisis de eventos de calidad de energía.

2.2 TRANSFORMADA WAVELET

2.2.1 Conceptos previos

2.2.1.1 Clasificación de señales

Una de las principales maneras de clasificar una señal es de acuerdo a su comportamiento en el tiempo. Las señales pueden ser *estacionarias* o *no estacionarias*.

Se dice que una señal es *estacionaria* o invariante en el tiempo cuando la forma de onda es definida por sus parámetros característicos que no varían en el tiempo. Un ejemplo de señal estacionaria puede ser una función senoidal, ya que la misma queda definida por un conjunto de parámetros (amplitud, frecuencia y fase) que no se modifican en el tiempo.

Por otro lado, las señales *no estacionarias* son aquellas en las que sus parámetros característicos varían con el tiempo.

La Figura 2.1 muestra un ejemplo de una señal estacionaria continua. La Figura 2.2 muestra la representación de una señal no estacionaria continua.

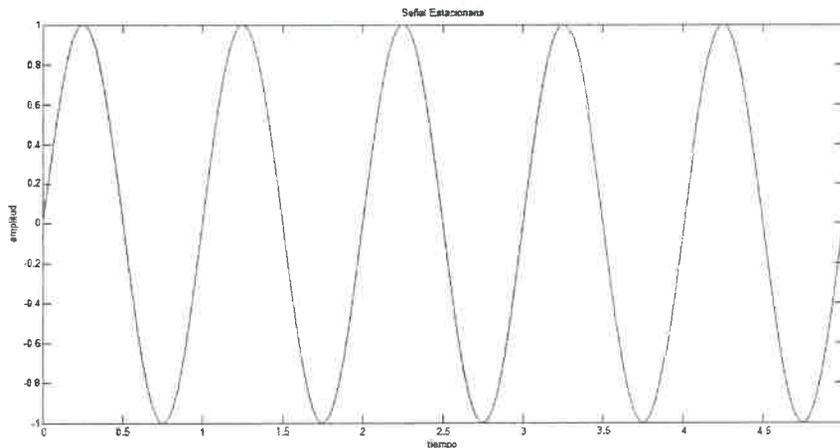


Figura 2.1 - Señal estacionaria.

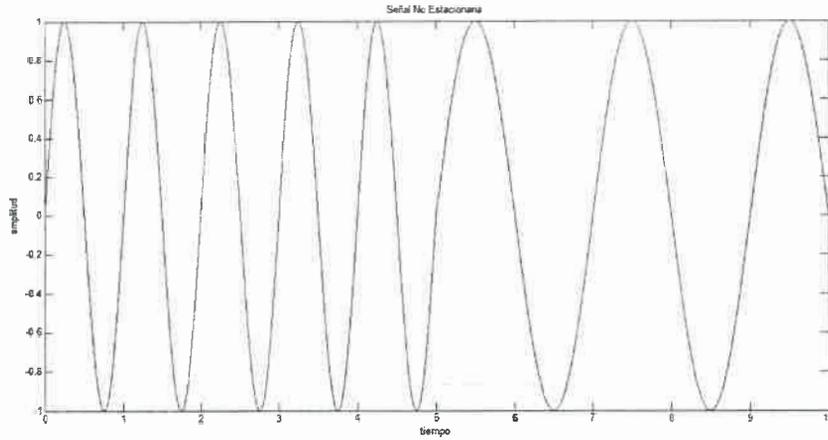


Figura 2.2 - Señal no estacionaria.

2.2.1.2 Transformada de Fourier

El estudio clásico del procesamiento de señales ha enfocado sus esfuerzos principalmente en el estudio de sistemas y señales de carácter estacionario. En este sentido la *Transformada de Fourier* ha sido la herramienta de análisis con mayor difusión para el procesamiento de señales.

En términos generales se puede definir a la *Transformada de Fourier* como una herramienta matemática empleada para representar una señal mediante una combinación lineal de exponenciales complejas infinitesimalmente separadas en frecuencia.

Sea $f(t)$ una función no periódica cuya variable real es t , la transformada de Fourier de esta señal viene dada por:

$$\mathfrak{F}\{f(t)\} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-j\omega t} \delta t \quad (2.1)$$

donde ω es una variable que representa la frecuencia y se mide en radianes por segundo [rad/seg]. Usualmente la frecuencia también puede expresarse en ciclos por segundo y su símbolo representativo es la letra f . La relación entre ω y f viene dada por:

$$f = \frac{\omega}{2\pi} \left[\frac{\text{ciclos}}{\text{segundos}} \right] \quad (2.2)$$

Para obtener una señal a través de su Transformada de Fourier se debe aplicar la transformada inversa de Fourier. Esta transformada viene dada por la siguiente ecuación:

$$\mathfrak{F}^{-1}\{F(\omega)\} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(\omega) e^{j\omega t} \delta \omega \quad (2.3)$$

La siguiente ecuación se corresponde a la Transformada de Fourier para señales de tiempo discreto (*DTFT por sus siglas en inglés*)

$$\mathfrak{F}\{x[n]\} = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \quad (2.4)$$

La transformada de Fourier, al tratarse de una herramienta ideada para el análisis de señales estacionarias, presenta limitaciones para el análisis de fenómenos de corta duración también conocidos como fenómenos transitorios. Para aclarar este concepto es conveniente analizar los siguientes ejemplos:

La Figura 2.3 a) muestra una señal senoidal en el dominio del tiempo y la Figura 2.3 b) su *Transformada de Fourier*. Al tratarse de una señal estacionaria, los parámetros característicos (Amplitud, frecuencia y fase) de la señal senoidal no varían en el tiempo de análisis. En este caso el espectro de frecuencias obtenido por la *Transformada de Fourier* se considera como una representación alternativa de la señal ya que, partiendo de la información del diagrama espectral, la señal puede ser completamente reconstruida en el dominio del tiempo.

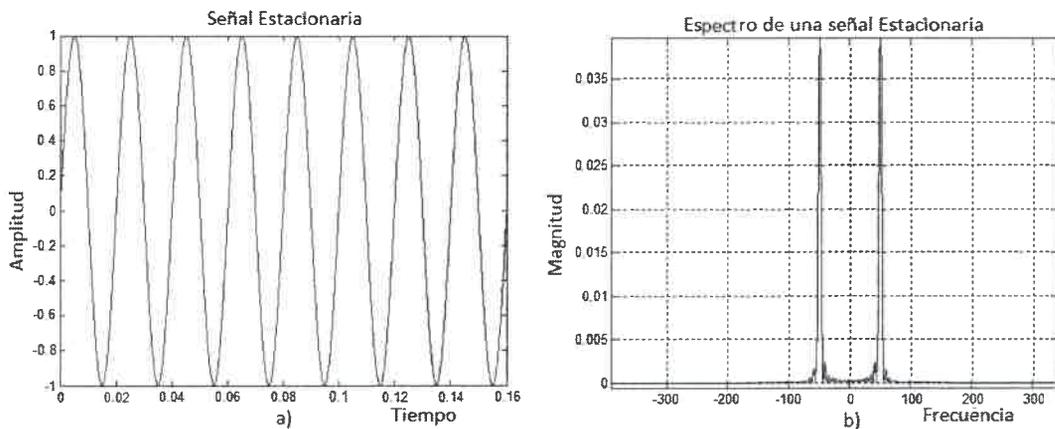


Figura 2.3 - Señal estacionaria

Para el segundo ejemplo se propone la señal mostrada en la Figura 2.4 a). Esta señal es de carácter no estacionario ya que su frecuencia, originalmente de 50 [Hz] , se desplaza hacia una frecuencia de oscilación de 300 [Hz] en un determinado instante de tiempo.

En el diagrama espectral de la señal, mostrado en la Figura 2.4 b), se pueden identificar los dos pulsos de frecuencia ubicados en los 50 [Hz] y 300 [Hz] . A primera vista se podría concluir que la *Transformada de Fourier* representa completamente a la señal en el dominio del tiempo. Sin embargo, analizando más profundamente el ejemplo propuesto se puede notar que *el diagrama de frecuencias no presenta información del tiempo en que se produce la transición en la frecuencia de la señal.*

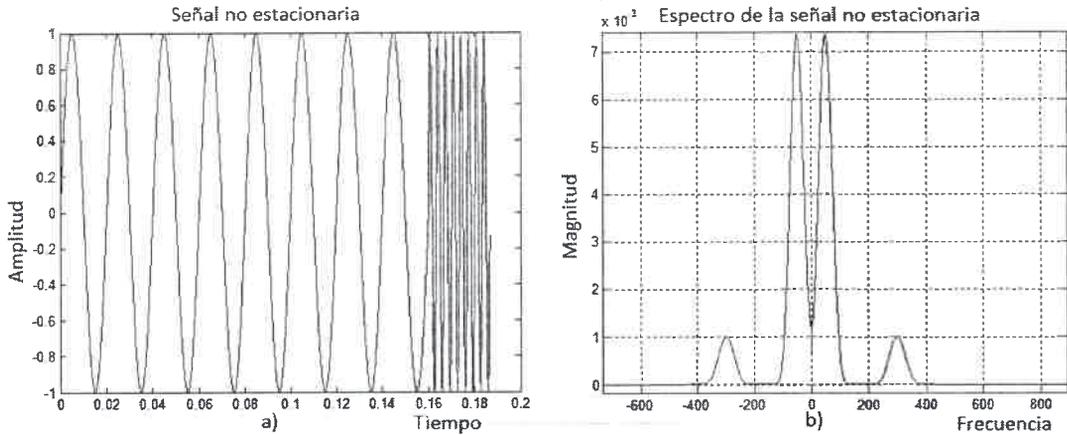


Figura 2.4 - Señal no estacionaria

Mediante este ejemplo se ilustra una importante limitación en la *Transformada de Fourier* para representar una señal que contenga transiciones en el dominio del tiempo.

En el capítulo 1 de este trabajo se explicó que las perturbaciones en un sistema eléctrico de potencia pueden ser del tipo estacionarias (perturbaciones armónicas o fluctuaciones) o no estacionarias (huecos, interrupciones, sobre tensión temporal, fenómenos transitorios, muescas, entre otras).

El análisis de Fourier, a pesar de ser una herramienta de gran utilidad para el procesamiento de perturbaciones estacionarias, resulta a la vista *insuficiente* para el procesamiento de perturbaciones de carácter transitorio. Por este motivo resulta claro que se necesita una herramienta matemática que permita el análisis de una señal en el dominio del tiempo y la frecuencia de manera simultánea.

Es importante aclarar que los diagramas en frecuencia mostrados en las gráficas de la Figura 2.3 y Figura 2.4 no se corresponden con el diagrama espectral real ya que:

- Las señales senoidales empleadas en el ejemplo no son señales de tiempo continuo. Por el contrario, estas señales se conforman mediante muestras de una señal senoidal graficadas como señales de tiempo continuo.
- El método obtenido para obtener el espectro de la señal, utiliza técnicas de enventanado. Cualquier método que emplea una porción de la señal para obtener el espectro introduce distorsión en el resultado obtenido. Por este motivo el espectro de una señal senoidal, la cual idealmente debería corresponderse a un impulso en el dominio de la frecuencia, se muestra como un pulso que abarca más de una única componente espectral.

2.2.1.3 Transformada de Fourier dependiente del tiempo

Para salvar las limitaciones que la Transformada de Fourier presenta para el análisis de señales no estacionarias, *Denis Gabor (1946)* desarrolló una modificación en el método para restringir el análisis a un determinado intervalo de tiempo. Para ello se divide la

señal en diferentes secciones en el tiempo y se considera que la señal es periódica dentro de cada uno de dichos intervalos. Al tratar a cada porción de la señal como si fuera periódica se puede aplicar la Transformada de Fourier tradicional y se presenta un espectro de frecuencias por cada una de las secciones en las que la señal original fue dividida. Una forma de presentar los resultados es mediante una grilla bidimensional cuyos ejes son las frecuencias y el tiempo. En esta gráfica se conoce como *espectrograma*.

Esta herramienta se denomina *Transformada de Fourier dependiente del tiempo* o *Transformada de Fourier de Tiempo Corto (STFT por sus siglas en inglés)* y puede ser expresada mediante la siguiente ecuación:

$$STFT(\tau, f) = \int_{-\infty}^{\infty} x(t)w(t - \tau)e^{-j\omega t} \delta t \quad (2.5)$$

donde w se corresponde a la función empleada para el ventaneo cuya forma y longitud determinan la resolución en el dominio del tiempo y frecuencia y τ es una variable empleada para el desplazamiento de la ventana a lo largo de la señal.

Una expresión equivalente para las señales en tiempo discreto puede ser expresada como:

$$STFT[m, \omega] = \sum_{n=-\infty}^{\infty} x[n]w[n - m]e^{-j\omega n} \quad (2.6)$$

donde w nuevamente se corresponde a la ventana.

m es la variable empleada para trasladar la ventana a lo largo de la señal.

Debido a que una misma ventana se emplea para el análisis de la señal indistintamente de la frecuencia la resolución del método, es la misma en todas las ubicaciones del plano tiempo frecuencia.

En la Figura 2.5 se puede ver como la transformada de Fourier de tiempo corto mapea una señal en el dominio del tiempo al dominio tiempo-frecuencia. En la Figura 2.6 se puede ver la STFT de un hueco de tensión.

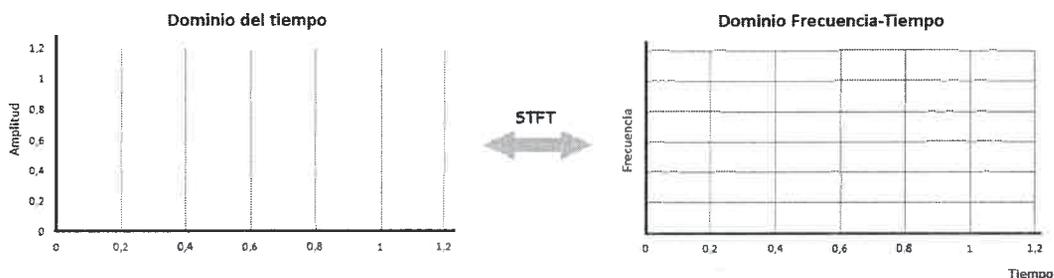


Figura 2.5 - Transformación de dominios de la STFT.

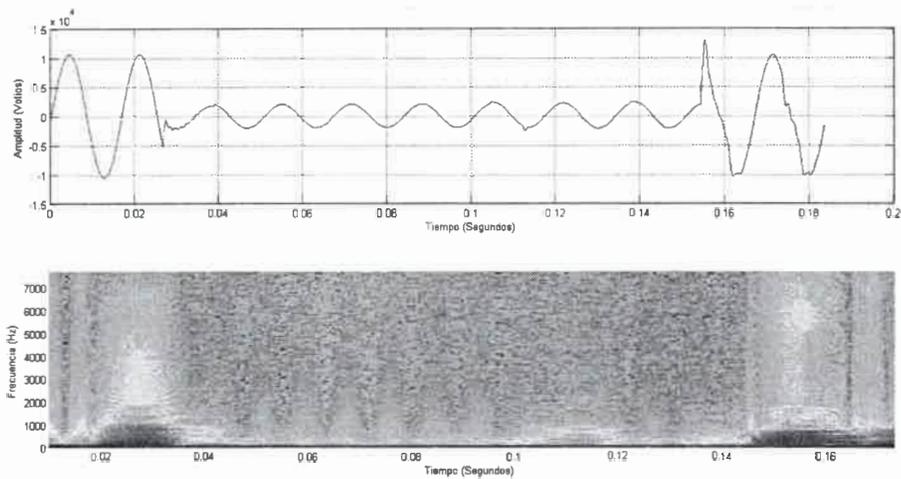


Figura 2.6 - Transformada de Fourier de tiempo Corto de un Hueco de Tensión.

En este momento se presenta una relación de compromiso entre la resolución en el dominio del tiempo y en el dominio de la frecuencia. Si se hace el ancho de la ventana lo suficientemente grande se obtendrán resultados con una buena resolución en frecuencia pero una pobre resolución en el tiempo. Para ventanas muy grandes, la *Transformada de Fourier dependiente del tiempo* se parecería mucho a la *Transformada de Fourier* tradicional. Por otro lado, hacer la ventana muy estrecha brindará una buena resolución en el tiempo en detrimento de la información utilizada para obtener la señal en el dominio de la frecuencia. Dada la forma de una ventana es imposible ajustar la resolución del método en el dominio del tiempo sin obtener el no deseado deterioro en la resolución de los resultados en el dominio de la frecuencia. Esta definición es usualmente conocida como *el principio de incertidumbre de Heisenberg*.

Una segunda limitación de este método radica en la carga computacional producto del ventaneo solapado entre las secciones de la señal a analizar.

El análisis de perturbaciones en un SEP, las señales de tensión o corriente pueden presentar componentes en frecuencia que van desde la corriente continua a componentes de muy alta frecuencia. Esta variedad de perturbaciones hace que una técnica que utilice una resolución fija sea inadecuada.

2.2.2 La Transformada Wavelet

La *Transformada Wavelet (WT)*, herramienta matemática desarrollada a mediados de los años '80, representa un método eficiente para el análisis local de señales no estacionarias y de rápida transitoriedad.

La clásica *Transformada de Fourier* es una herramienta que transforma una señal del dominio del tiempo al dominio de la frecuencia por medio de una representación que emplea exponenciales complejas infinitesimalmente separadas en frecuencia. Cada

coeficiente de la *Transformada de Fourier* se puede interpretar como una proyección de la señal en cada una de las bases ortonormales formadas por exponenciales complejas.

La *Transformada Wavelet*, por el contrario, en lugar de utilizar exponenciales complejas de duración y energía infinita, emplea un grupo de funciones base conocida como onditas o, por su traducción al inglés, *wavelets*. La WT, por lo tanto, se entiende como una herramienta que realiza una proyección de la señal en distintas bases ortonormales dadas por funciones especiales denominadas *onditas* (en inglés *wavelets*).

Las familias de onditas, representadas simbólicamente como $\psi(t)$, son un grupo especial de señales cuya principal característica es su flexibilidad para adaptarse a cualquier tipo de evento de corta duración.

En este capítulo se detallarán los aspectos matemáticos más relevantes de la *Transformada Wavelet* en tiempo continuo para luego hacer extensivo el análisis a señales en tiempo discreto.

2.2.2.1 Wavelets madre

Las wavelets son familias de funciones que se encuentran definidas en el espacio o tiempo. Cada componente de una familia de wavelets se genera a partir de una función denominada *wavelet madre* a través de las variables de escala (a) y la variable de traslación en el tiempo (b).

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad (2.7)$$

Las wavelets madre son un tipo especial de señales que deben cumplir con las siguientes condiciones:

- 1) Una wavelet debe tener energía finita:

$$E = \int_{-\infty}^{\infty} |\psi(t)|^2 dt < \infty \quad (2.8)$$

- 2) La función debe cumplir con el criterio de la constante de admisibilidad:

$$C_{\psi} = \int_0^{\infty} \frac{|\hat{\psi}(f)|^2}{f} \delta f < \infty \quad (2.9)$$

donde f es la frecuencia y la condición implica que la wavelet debe tener una componente de frecuencia $\psi(f=0) = 0$ o, lo que resulta equivalente, debe tener valor medio nulo.

- 3) Para wavelets complejas la transformada de Fourier $\psi(f)$ debe ser real y desvanecida para frecuencias negativas.

Algunas de las wavelet madre más comúnmente empleadas pueden verse en la Figura 2.7, Figura 2.8 y Figura 2.9.

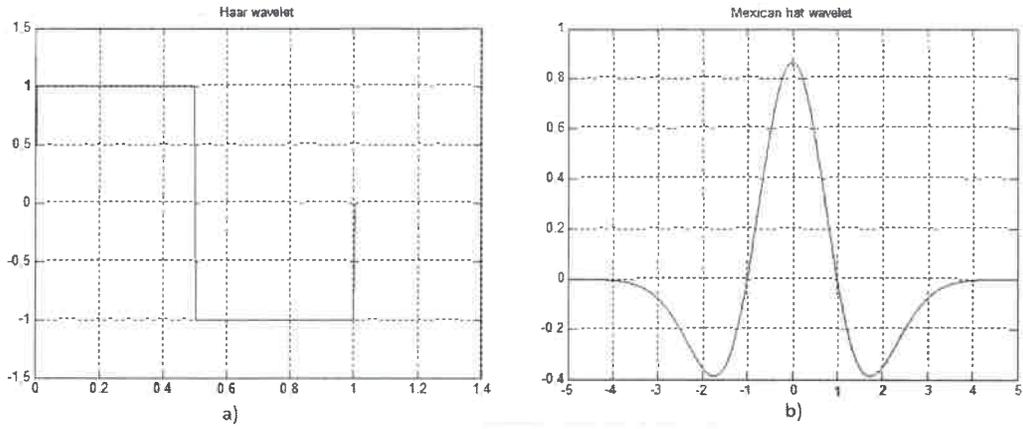


Figura 2.7 - a) Wavelet Haar b) Wavelet Sombrero Mexicano.

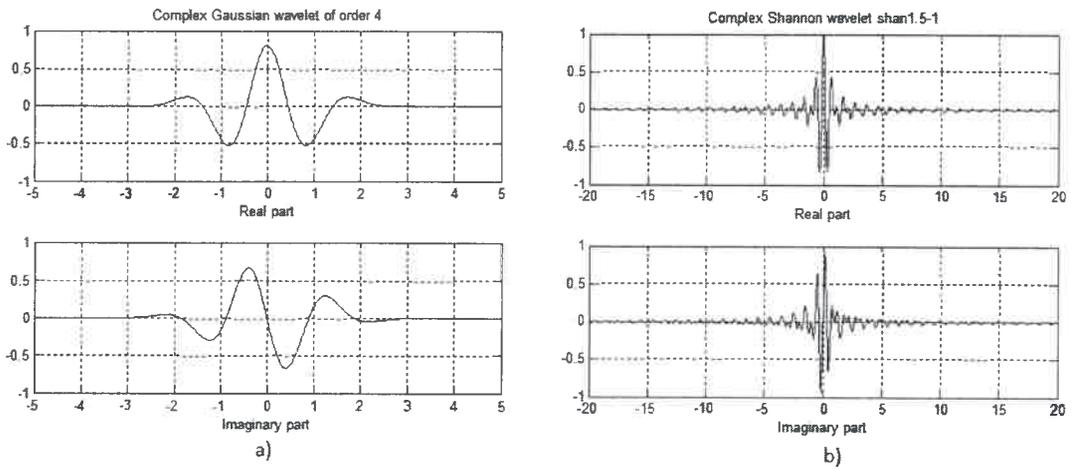


Figura 2.8 - Wavelet Gaussiana Compleja b) Wavelet Shannon Compleja.

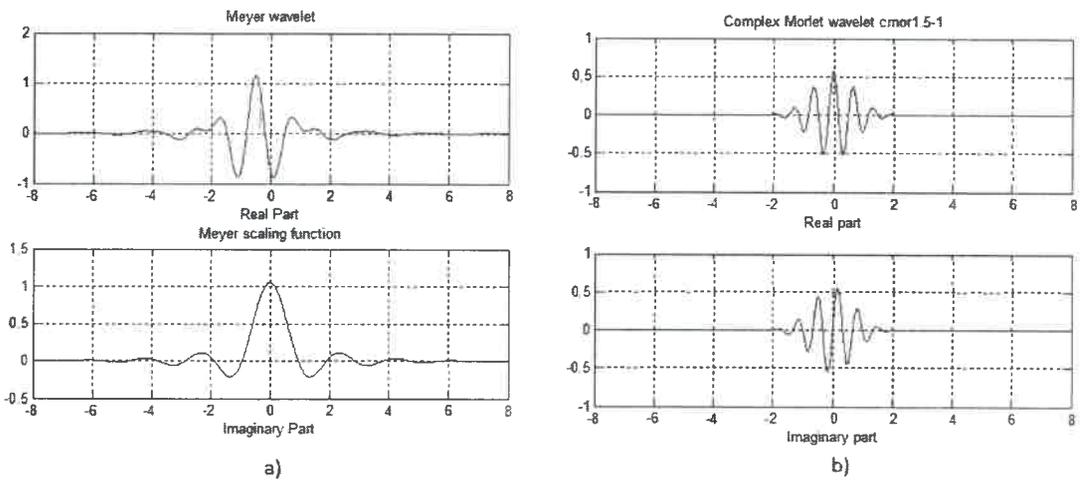


Figura 2.9 - a) Wavelet Meyer b) Wavelet Morlet.

La selección de la wavelet madre más apropiada depende de la aplicación y no existe una regla general para realizar esta tarea. Por ejemplo, Srivastava, Siddharth y Singh presentaron un trabajo donde se concluye que la wavelet madre Gaussiana Compleja es la más apropiada para el *análisis de perturbaciones del tipo armónico* [20].

El aporte realizado por Vega, Duarte, Ordonez y Kagan compara diferentes tipos de wavelets madre para la detección de las perturbaciones del tipo hueco, sobre tensión, transitorios, fluctuaciones y armónicos. Este trabajo concluye que la mejor wavelet madre es la Biorotogonal 3.9 [21].

Sin embargo, la wavelet madre más comúnmente empleada para la detección de eventos de calidad de energía es la Daubechie número 4 que se muestra en la Figura 2.10: [9,22-24]

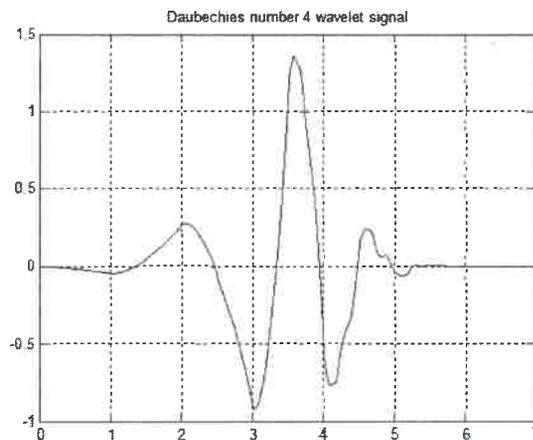


Figura 2.10 - Wavelet madre Daubechie 4.

2.2.2.2 Expansión y traslación de una wavelet madre.

De acuerdo a lo explicado en la sección 2.2.2.1 una familia de wavelet se genera a partir de la traslación y escalamiento en el tiempo de la función wavelet madre.

La Figura 2.11 muestra el efecto de la traslación en el tiempo de una wavelet madre del tipo Morlet. El término traslación se relaciona con la localización de la ventana a medida que ésta se desplaza a lo largo de la señal. Obviamente este término corresponde a la información del tiempo en el dominio transformado.

La Figura 2.12 ilustra el efecto de compresión o dilatación de la wavelet madre, producto de la operación denominada escalamiento. En términos de frecuencia, las bajas frecuencias (altas escalas) corresponden a una información global que comúnmente abarca toda la señal, mientras que las altas frecuencias (escalas bajas) corresponden a una información detallada de una característica oculta en la señal, que se produce en un tiempo relativamente pequeño.

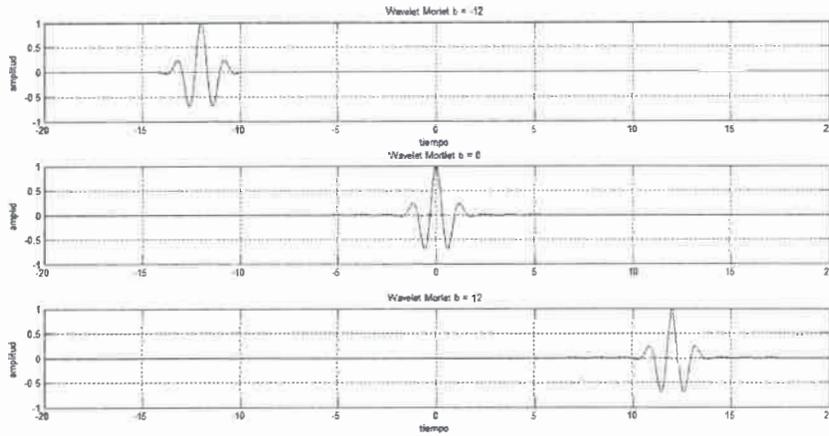


Figura 2.11 - Traslación de la Wavelet Madre.

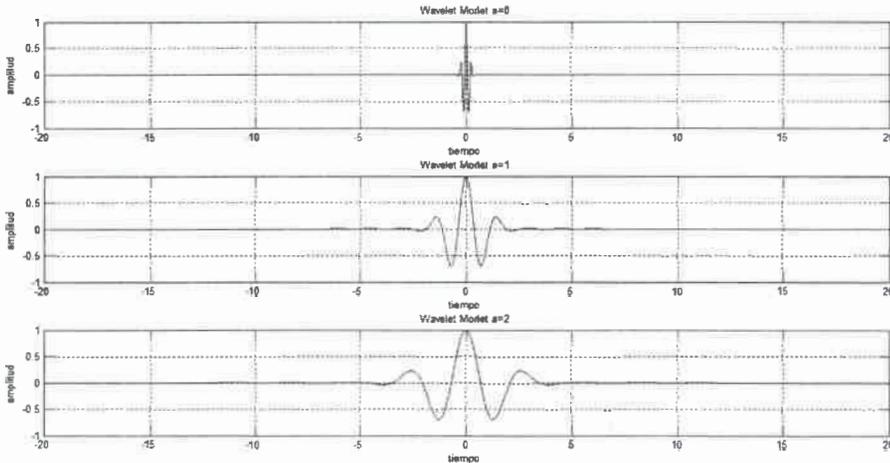


Figura 2.12 - Escalamiento de la Wavelet Madre.

2.2.2.3 La transformada Wavelet de Tiempo Continuo (CWT)

La transformada Wavelet de tiempo continuo se define como:

$$CWT(a, b; \psi) = a^{-\frac{1}{2}} \int_{-\infty}^{\infty} f(t) \psi^* \left(\frac{t - b}{a} \right) \delta t \tag{2.10}$$

La ecuación correspondiente a la transformada wavelet continua (CWT) se puede interpretar de manera similar a la transformada de Fourier de tiempo corto (STFT). En ambos casos existe una función que multiplica a la señal y en ambos casos esta función se traslada a lo largo del tiempo. La función wavelet madre $\psi(t)$ hace de veces de la función empleada para el ventaneo de la señal $w(t)$ en la transformada de Fourier dependiente del tiempo. La principal diferencia entre ambas transformadas radican en que, para la STFT, el ancho de la función empleada para el ventaneo permanece constante, mientras que, para la CWT, la wavelet madre es comprimida o expandida por un factor a .

El proceso de cálculo de la transformada wavelet continua es abordado de manera práctica por Miiti, Openheim y Poggio [25] y se puede resumir de la siguiente manera:

1. Se selecciona una wavelet madre.
2. Se calcula el coeficiente C que indica el grado de correlación entre la wavelet y la primera porción de la señal. Mientras más alto sea el coeficiente mayor será la similitud de la sección de la señal con la wavelet. Observar que el grado de similitud está íntimamente ligado a la elección de la wavelet madre.
3. Se traslada la wavelet mediante el parámetro b y se repite el paso 2 hasta cubrir toda la señal a analizar. En este punto se obtienen distintos coeficientes de correlación para un único valor de la escala a y para distintos valores de traslación b .
4. Se modifica la escala de la wavelet y se repiten los pasos 2 y 3.
5. Se repiten los pasos 2, 3 y 4 para todas las escalas.

Cuando se han finalizado los pasos se obtiene un conjunto de coeficientes de correlación vinculados a distintos valores de escala a y traslación b .

Para realizar una representación que sea fácil de interpretar se puede realizar un gráfico en donde el eje x se corresponda a la posición de la wavelet con respecto a la señal, en el eje y se represente el valor de la escala, y una escala de colores represente la magnitud del parámetro C para cada par de valores (a,b) .

Es importante establecer que existe una correspondencia entre las escalas y la frecuencia de la señal a analizar:

Para bajas escalas (valores chicos del parámetro a) la wavelet madre se comprime, lo cual la hace adecuada para el análisis de los detalles y cambios rápidos de la señal. Los cambios rápidos se corresponden a altas frecuencias. En resumen, bajas escalas tienen una correspondencia directa a altas frecuencias.

Para escalas altas (valores grandes del parámetro a) la wavelet madre se ensancha, lo cual la hace apropiada para la detección de fenómenos lentos. Fenómenos lentos se corresponden a frecuencias bajas. Por lo tanto, escalas altas tienen una relación directa con bajas frecuencias.

Una representación gráfica tiempo-escala se muestra en la siguiente Figura 2.13.

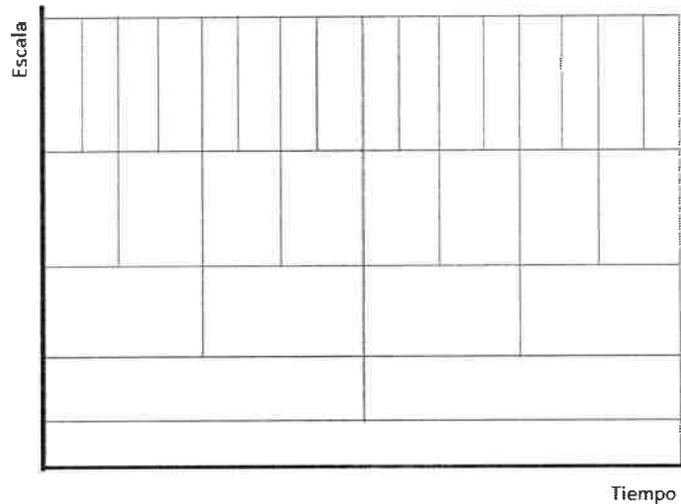


Figura 2.13 - Plano Escala-Tiempo de la Transformada Wavelet.

El área de cada división es fija y viene determinada por la función de la wavelet madre. Al comparar el plano Escala - Tiempo de la CWT con el plano Frecuencia - Tiempo de la STFT se concluye que la transformada wavelet es un método multi-resolución ya que el método permite ajustar la resolución para detectar los detalles (altas frecuencias) de una señal.

2.2.2.4 Transformada Wavelet Discreta (DWT)

El cálculo de la CWT se trata de un procedimiento bastante complejo que involucra la resolución de complicadas ecuaciones analíticas (si la transformada se resuelve de manera analítica) o el procesamiento de una gran cantidad de datos (si la transformada es resuelta a través de métodos numéricos).

En la DWT los parámetros de escalamiento a y traslación b pueden asumir un número discreto de valores, lo que facilita en gran medida los cálculos.

Se ha demostrado que si se seleccionan los valores de escalamiento y traslación empleando una potencia de dos (también llamada escala diádica) los resultados de los análisis resultan mucho más eficientes.

La función continua $\psi_{a,b}(t)$ se transforma en la wavelet discreta $\psi_{j,k}(n)$ dada por la siguiente ecuación:

$$\psi_{j,k}(n) = a^{-\frac{j}{2}} \psi(a^{-j}n - kb) \quad (2.11)$$

Los parámetros a y b se definen empleando la escala diádica donde j representa el nivel de descomposición y k el factor de traslación tal como se muestra en la siguiente ecuación:

$$a = 2^j; b = k2^j \quad a > 0; b > 1 \quad (j; k) \in Z \quad (2.12)$$

La Transformada Wavelet Discreta viene dada por:

$$DWT(j, k) = a^{-\frac{1}{2}} \int f[n] \psi(a^{-j}n - kb) dn \quad (2.13)$$

2.2.2.5 Interpretación de la DWT como un banco de filtros

Una forma eficiente para representar la DWT fue presentada en 1998 por Mallat [26].

Para muchas señales, el contenido de bajas frecuencias representa la parte más importante de la señal. El contenido de frecuencias, por otro lado, se considera como un componente que representa matiz o detalle.

Esta distinción es lo que en la Transformada Wavelet se conoce como aproximación y detalle. La aproximación se corresponde con altas escalas o bajas frecuencias mientras que el detalle se corresponde con bajas escalas o altas frecuencias. Un banco de filtros que cumple con la función de diferenciar estas componentes puede visualizarse en la Figura 2.14.

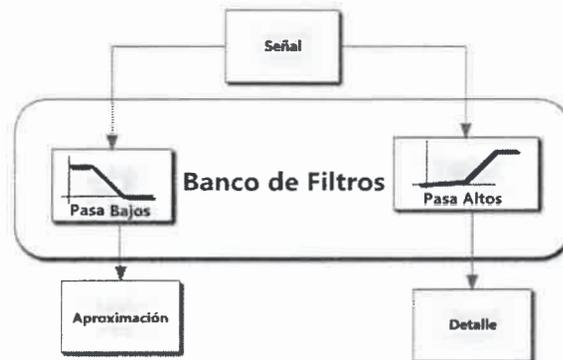


Figura 2.14 - Aproximación mediante banco de filtros.

La señal ingresa a la etapa de bancos de filtros y, como resultado, dos señales emergen. Es importante notar que esta operación multiplica considerablemente la cantidad de información que debe procesarse. Si la señal cuenta con 1000 muestras, la salida de cada filtro estará formada por esa misma cantidad de muestras. El proceso de filtrado duplicó la cantidad de información a procesar.

Por este motivo es preciso incorporar una técnica denominada sub-muestreo. El sub-muestreo básicamente consiste en eliminar muestras que se consideran redundantes. Si se elimina una de cada dos muestras se dice que el sub-muestreo es de orden $M=2$.

El procedimiento de filtrado y sub-muestreo mostrado en la Figura 2.15 genera los coeficientes de aproximación y detalle de la DWT.

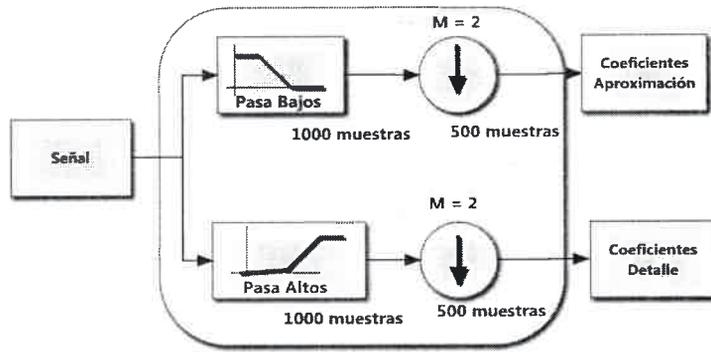


Figura 2.15 - Banco de filtros y sub muestreo.

Como ejemplo se introduce al banco de filtros una señal senoidal corrupta con ruido blanco gaussiano. La Figura 2.16 muestra los coeficientes de aproximación y detalle.

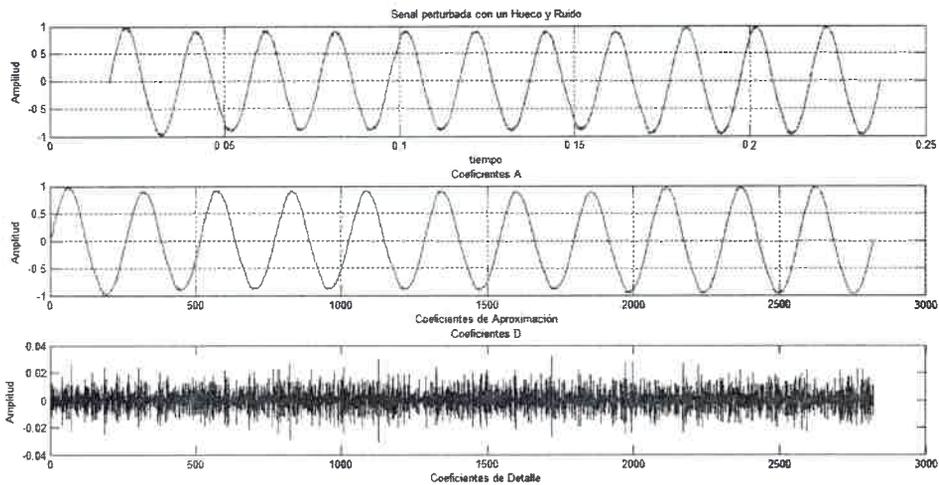


Figura 2.16 - Coeficientes A y D de una señal senoidal más ruido blanco

Este proceso puede iterarse, obteniendo sucesivas aproximaciones de forma tal que la señal sea descompuesta en varios componentes de menor resolución. Al esquema mostrado en Figura 2.17 se lo conoce como *árbol de descomposición*.

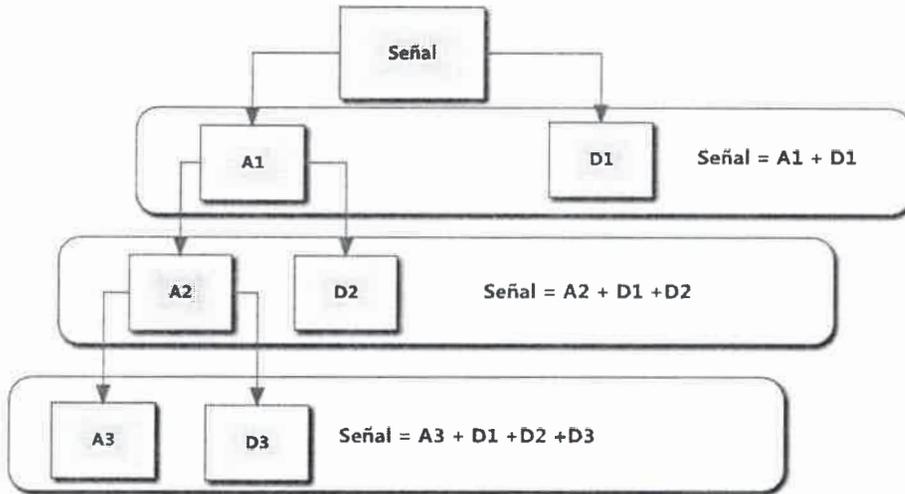


Figura 2.17 - Árbol de descomposición

El árbol de descomposición representa una herramienta fundamental para el análisis de perturbaciones en sistemas eléctricos de potencia. En la Figura 2.18 se muestra un árbol de descomposición para una perturbación del tipo *Muesca*. Los coeficientes de detalle brindan información importante relacionada con la periodicidad y duración de los eventos. Estos coeficientes muestran información relacionada con los cambios rápidos en la señal original. De la misma forma, los coeficientes de aproximación contienen información de la señal de baja frecuencia.

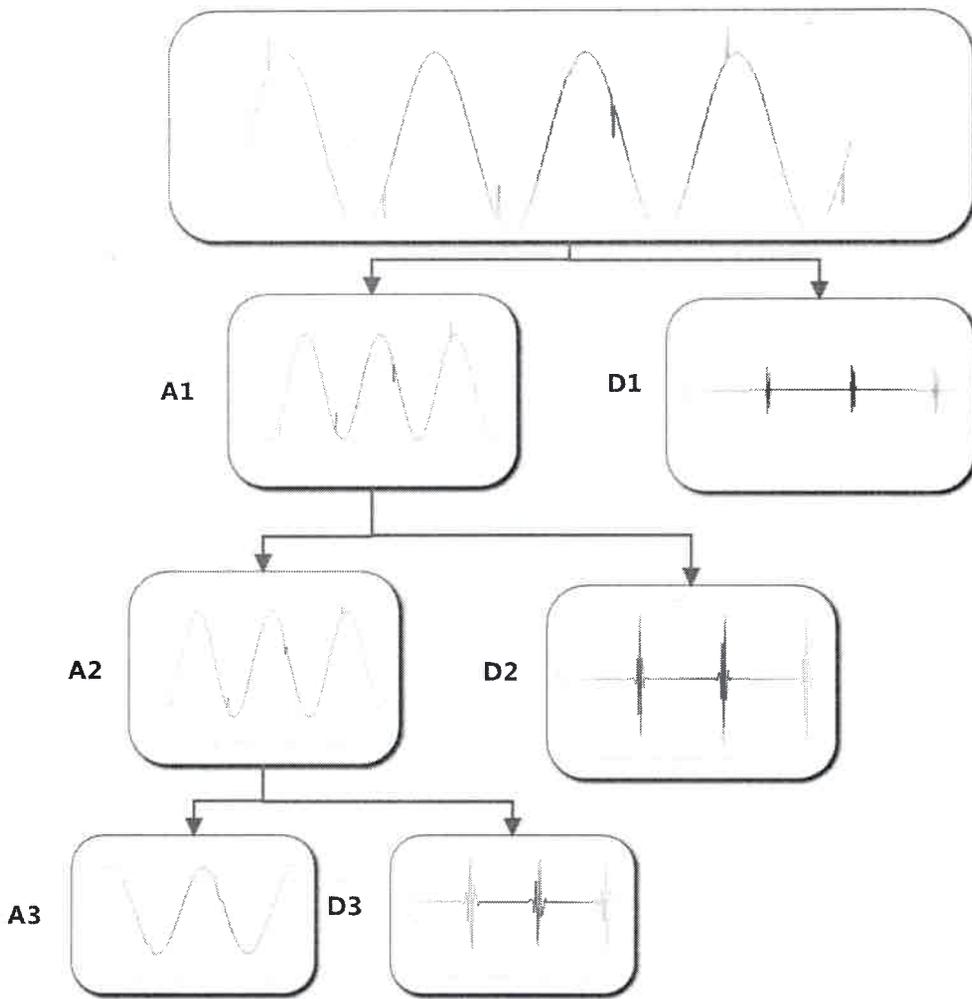


Figura 2.18 - Árbol de descomposición para una perturbación tipo Muesca.

2.2.2.6 Teorema de Parseval

Este teorema establece una relación entre la energía total de la señal $x(t)$ y la energía asociada a cada uno de los coeficientes de la descomposición.

Esta relación puede expresarse a través de la siguiente ecuación:

$$\int |x(t)|^2 = \sum_{l=-\infty}^{\infty} |c[l]|^2 + \sum_{j=0}^{\infty} \sum_{k=-\infty}^{\infty} |d_j[k]|^2 \tag{2.14}$$

La energía total de la señal está dividida en tiempo, a través de k y en escala o frecuencia, por medio de j . Esta es la partición tiempo-frecuencia que efectúa la *DWT*.

Daubechies [27] mostró que es posible para las funciones de escalado y funciones wavelet, constituir un paquete compacto que tome valores distintos de cero solamente

en una región de duración finita de manera de localizar en el tiempo el contenido espectral.

El sistema wavelet permite contar con características similares a la Transformada de Fourier dependiente del tiempo, pero con la diferencia que aquí es posible tener ortogonalidad y buena resolución tiempo frecuencia, característica imposible de lograr en la STFT.

2.3 ESTADO DEL ARTE: LA TRANSFORMADA WAVELET PARA LA DETECCION DE PERTURBACIONES

De todos los métodos para la detección de eventos de calidad de energía y la extracción de parámetros característicos, la técnica más extensamente empleada se basa en la *Transformada Wavelet*. En este sentido, uno de los aportes más importantes es el trabajo presentado por Santoso, Powers, Grady y Hofmann, (1996) [9]. Este artículo es uno de los precursores en lo referente al empleo de la *Transformada Wavelet* para la detección de perturbaciones en un sistema eléctrico de potencia.

En su trabajo, Possion (1999), concluye que la Transformada Wavelet Continua (CWT) es una buena herramienta para la detección de fenómenos transitorios rápidos con buena precisión en frecuencia, aunque no presenta buenas prestaciones para el análisis de fenómenos estacionarios. Este trabajo compara la (CWT) con otros métodos como un algoritmo de análisis Multi Resolución (MRA), la Transformada de Fourier (FT) clásica y la Transformada Cuadrática (QT) [28] aludiendo las ventajas y desventajas de cada uno de ellos.

Ese mismo año Gaouda, Salama, Sultan y Chikhani (1999) presentaron un importante aporte donde se fundamenta el uso de una técnica de descomposición multiresolución basada en la *Transformada Wavelet Discreta* (por sus siglas en inglés DWT) para el análisis de eventos transitorios [22].

Un segundo trabajo fue presentado por estos autores en donde se aplica la transformada multiresolución para el análisis de eventos de corta duración (2000) [29].

Weiming, S. Xuelei, L. Jingbo and Z. Zhiheng (2006) presentaron un método que combina la Transformada de Fourier dependiente del tiempo y el Paquete de Transformadas Wavelet para la extracción de parámetros característicos de una señal perturbada con el objetivo de reducir la dimensión de los datos manejados por un sistema de clasificación [30].

Más recientemente Barros (2008) propone el uso *Paquete de Transformada Wavelet* para el análisis de armónicos en un sistema eléctrico de potencia [31].

Además de la utilización de la *Transformada Wavelet* para la detección, localización y extracción de parámetros característicos de un evento de calidad de energía, esta técnica también ha demostrado ser un poderoso método de compresión de datos. En este sentido Santoso (1997) presentó un trabajo donde se emplea una técnica de umbralización para descartar información y reducir de este modo el tamaño del archivo que contiene la información de la señal perturbada en una tasa que va desde un tercio a un sexto de su tamaño original [32].

En numerosos trabajos se ha demostrado que el rendimiento de la Transformada Wavelet sufre una degradación importante frente al ruido aditivo. Para mejorar la

capacidad los sistemas de monitoreo basados en la WT Hong-Tzer and L. Chiung-Chou presentaron un esquema de cancelación de ruido [33].

2.4 CONCLUSIONES

En este capítulo se abordaron algunos conceptos básicos de la Transformada de Fourier y la Transformada de Fourier dependiente del tiempo para el análisis de señales estacionarias y no estacionarias demostrando las limitaciones y desventajas que ambos métodos presentan frente a la Transformada Wavelet.

Luego fueron presentadas algunas nociones teóricas y analíticas relativas a la Transformada Wavelet y se demostraron sus ventajas frente a los métodos de procesamiento de señales tradicionales. Dentro de los temas abordados en este capítulo se encuentra la Transformada Wavelet Continua (CWT), la Transformada Wavelet Discreta (DWT) y una útil interpretación de la transformada wavelet discreta representada por un banco de filtros.

Se demostró que la WT, al tratarse de un método de análisis multi resolución, permite la localización temporal precisa, con una aceptable resolución en frecuencias, de un evento de calidad de energía en un sistema eléctrico de potencia.

Finalmente se presentan los aportes más relevantes de distintos autores que utilizan la WT para detectar y clasificar eventos de calidad de energía.

Las claras ventajas que la WT presentadas frente a distintos métodos de procesamiento de señales justifican su elección para el desarrollo del software de detección y clasificación de eventos de calidad de energía presentado en Capítulo 5 del presente trabajo de tesis.

Capítulo 3: Redes Neuronales Artificiales

3.1 INTRODUCCIÓN

En la actualidad los sistemas de procesamiento se encuentran difundidos alrededor de todo el mundo y representan una de las herramientas más formidables con las que ha contado la humanidad, ya que permiten implementar fácilmente algoritmos para resolver multitud de problemas y cálculos que antes resultaban engorrosos y complicados. Sin embargo, debido a que la mayoría de los sistemas de procesamiento emplean una programación rígida, la siguiente importante limitación se presenta:

Los sistemas de procesamiento tradicional son capaces de resolver sólo aquellos problemas que admiten tratamiento algorítmico pero no pueden resolver problemas que resultan naturalmente sencillos para el cerebro humano como es el caso, por ejemplo, de la clasificación de objetos por rasgos comunes. [34]

Esta limitación impulsó a muchos científicos a estudiar las capacidades del cerebro humano como fuente de ideas para el desarrollo de una nueva generación de máquinas, lo que dio lugar a una reciente y prominente rama en la ciencia de la computación denominada *Inteligencia Artificial*. Así, la inteligencia artificial es un intento por descubrir y describir aspectos de la inteligencia humana que pueden ser simulados mediante máquinas.

Las *Redes Neuronales Artificiales* (ANN por sus siglas en inglés) se corresponden con un área de estudio dentro de la Inteligencia Artificial y pueden entenderse como una forma de emular ciertas características propias de los humanos, como la capacidad de memorizar y de asociar hechos.

El humano es capaz de resolver situaciones complejas mediante lo que comúnmente se conoce como experiencia. Se llama experiencia a toda enseñanza adquirida por medio de acontecimientos transcurridos en el pasado como, por ejemplo, la capacidad del ser humano de reconocer objetos mediante el recuerdo y asociación de patrones aprendidos tiempo atrás. De esta manera parece claro que una forma de aproximarse al problema consiste en la construcción de sistemas que sean capaces de reproducir la capacidad humana de emplear la experiencia en la resolución de problemas complejos.

Las *Redes Neuronales* no son más que un modelo artificial y simplificado del cerebro humano, que es el ejemplo más perfecto del que se dispone para un sistema que es capaz de adquirir conocimiento a través de la experiencia. Una red neuronal es un sistema para el tratamiento de la información, cuya unidad básica de procesamiento está inspirada en la célula fundamental del sistema nervioso humano: *La neurona*.

Las características especiales de los sistemas de computación neuronal permiten que esta técnica de cálculo sea utilizada en una extensa variedad de aplicaciones, dentro de las cuales se encuentran la clasificación de patrones, la asociación de patrones, la generación de señales no lineales, entre otras áreas de aplicación.

La capacidad de las Redes Neuronales para el reconocimiento y clasificación de patrones y su habilidad para el manejo de datos contaminados con ruido hizo que fueran

utilizadas para la clasificación de patrones de perturbaciones en un sistema eléctrico de potencia.

En este capítulo se introducirá al lector a los conceptos básicos relacionados a las Redes Neuronales. Se abordarán aspectos generales como las características principales que definen una red neuronal, las arquitecturas conocidas, las funciones de activación y los algoritmos de entrenamiento.

A su vez se abordará un tipo de red neuronal en particular conocido como *feedforward backpropagation* ya que este modelo será el empleado en capítulo 5 para el desarrollo del sistema de procesamiento de información.

Finalmente, en este capítulo, se realizará un recorrido por las publicaciones más relevantes que proponen a las Redes Neuronales para el monitoreo de eventos de calidad de energía.

3.2 CONCEPTOS BASICOS DE LAS REDES NEURONALES

Una ANN es un sistema de procesamiento de información que tiene ciertas características en común con las redes neuronales biológicas [34].

Las ANN han sido desarrolladas como una generalización de modelos matemáticos que emulan el funcionamiento del cerebro humano en los siguientes aspectos:

- El tratamiento de la información no se realiza en una única unidad de procesamiento, si no que ocurre en múltiples elementos denominados *neuronas*.
- Las señales se pasan de neurona a neurona a través de enlaces de conexión.
- Cada enlace entre neuronas tiene un peso asociado, el que, en una red típica, multiplica la señal transmitida.
- Cada neurona aplica una función de activación (usualmente no lineal) a la suma de sus entradas ponderadas por los pesos para determinar la señal de salida.

Una red neuronal está conformada por un número de elementos de procesamiento simple, llamados *neuronas*, *unidades*, *celdas* o *nodos*. Cada neurona se conecta a otra por medio de enlaces de comunicación, y cada enlace tiene asociado un determinado *peso*. Este peso representa la información que va a ser utilizada por la red para resolver el problema. Cada neurona tiene asociado un estado denominado *activación* o *nivel de actividad* el cual, a su vez, es función de sus entradas. Es importante destacar que una neurona puede enviar solamente un estado o nivel al mismo tiempo a una o más neuronas.

Por ejemplo, considere que la neurona ilustrada en la Figura 3.1 ocupa la posición k dentro de una determinada arquitectura y que recibe información proveniente de sus n entradas.

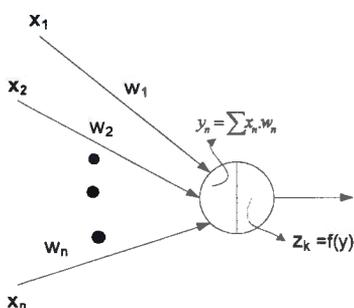


Figura 3.1 - Esquema de una Neurona.

Cada valor x_i puede tratarse de un valor de entrada a la red neuronal o bien de la salida de una neurona de capas anteriores. Esto dependerá de la arquitectura y la posición de la k -ésima neurona dentro de la red. Los pesos w_{ki} representan la relevancia de la i -ésima conexión de la k -ésima neurona.

Cada k -ésima neurona combina cada entrada x_i con sus correspondientes pesos w_{ik} determinando una salida descrita por:

$$y_k = \bar{x} \cdot \bar{w} = \sum_{i=1}^n x_i w_{ik} \quad (3.15)$$

El resultado de esta ecuación representa la entrada a una función denominada *función de activación*. La sección 3.6 detalla algunas de las funciones de activación comúnmente empleadas.

La función de activación es uno de los parámetros fundamentales del diseño de la red neuronal y su elección tendrá influencia en el comportamiento del sistema. Las funciones más comúnmente empleadas son: *función lineal*, *sigmoidea*, *sigmoidea logarítmica*, *umbral* y *gaussiana*.

Supongamos ahora que la neurona z se conecta a dos neuronas m_1 y m_2 a través de dos conexiones cuyos pesos son V_1 y V_2 respectivamente. La Figura 3.2 muestra el ejemplo propuesto:

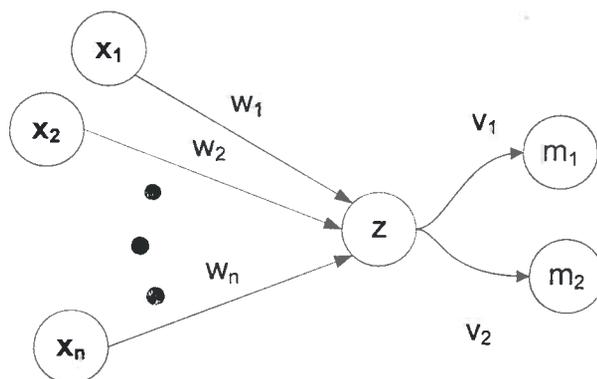


Figura 3.2 - Conexión entre neuronas.

El nivel de activación de la neurona m_1 no será igual al de la neurona m_2 debido a que es altamente probable que los pesos de las conexiones V_1 y V_2 sean diferentes. La salida de la neurona m_1 nuevamente estará dada por la suma de las entradas multiplicadas por los pesos de las conexiones, y procesadas con una determinada función de activación.

Para definir el comportamiento de la red, los pesos de las conexiones de todas las neuronas se ajustan mediante métodos denominados *algoritmos de entrenamiento*.

Este ejemplo presenta una primera idea de cómo una red neuronal se caracteriza por:

- a) *Arquitectura de la red*: patrón de conexión entre cada neurona.
- b) *Algoritmo de entrenamiento*: Método por el cual se ajustan los pesos de las conexiones entre neuronas.

c) *Función de activación*: Función lineal o no lineal con que cada neurona responde ante los estímulos en las conexiones conocidas como entradas.

3.2.1 Similitudes con las redes neuronales biológicas

Existe una analogía cercana entre la estructura de una neurona biológica y el elemento de procesamiento de una ANN.

Una neurona biológica tiene tres tipos de componentes que tienen un particular interés en el entendimiento de una red neuronal. Tal como se pueden ver en la Figura 3.3 estas son: Dendrita, Soma y Axón.

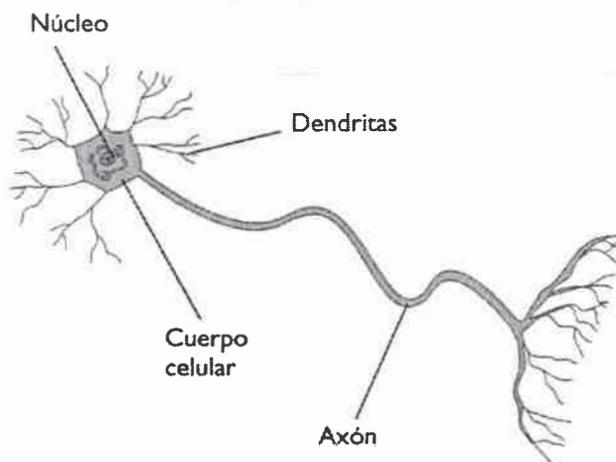


Figura 3.3 - Neurona biológica [35].

Las múltiples neuronas reciben señales de otras neuronas. Estas señales se corresponden con impulsos eléctricos que son transmitidos a través del axón y las dendritas hacia otra neurona.

- El soma, también llamado cuerpo celular, suma las señales que ingresan a través de todas las dendritas. Cuando la suma de las señales de la entrada llega a un determinado nivel, la célula se enciende y transmite una señal a través del axón hacia las neuronas vecinas. Comúnmente se entiende que una neurona se encuentra apagada o encendida para un determinado instante de tiempo. Los elementos de procesamiento reciben múltiples señales de entrada.
- Las señales pueden ser modificadas por un peso en la neurona receptora.
- Los elementos de procesamiento suman las entradas.
- Bajo determinadas circunstancias (suficiente nivel de entrada), las neuronas transmiten una salida.
- La salida de una neurona puede ir a otra neurona a través del axón.
- La información se procesa localmente.
- La memoria es distribuida.

- La memoria a largo plazo reside en la sinapsis de las neuronas.
- La memoria a corto plazo se corresponde con las señales enviadas por las neuronas.
- El vínculo sináptico entre dos neuronas se modifica con la experiencia.
- Los neurotransmisores para la sinapsis pueden ser excitatorios o inhibitorios.

3.3 CARACTERÍSTICAS DE LAS REDES NEURONALES

Las ANN, al margen de "parecerse" al cerebro, presentan una serie de características propias del mismo. Por ejemplo las ANN aprenden de la experiencia, generalizan de ejemplos previos a ejemplos nuevos y abstraen las características principales de una serie de datos. Algunas de las características más importantes son:

3.3.1 Aprendizaje Adaptativo

Capacidad de adquirir el conocimiento de un problema por medio del estudio, ejercicio o experiencia. Las ANN pueden cambiar su comportamiento en función del entorno. Se les presenta un conjunto de entradas y ellas mismas se ajustan para producir salidas consistentes.

3.3.2 Generalización

Es facultad de las Redes Neuronales responder apropiadamente cuando se les presentan datos o situaciones a las que no habrían sido expuestas anteriormente. Esta característica es muy importante cuando se tiene que solucionar problemas en los que la información de entrada no es muy clara; además permite que el sistema dé una solución, incluso cuando la información de entrada está especificada de forma incompleta.

3.3.3 Abstracción

Las ANN pueden considerar por separado las cualidades de un dato. Algunas ANN son capaces de abstraer la esencia de un conjunto de datos de entrada que aparentemente no presentan aspectos comunes o relativos.

3.3.4 Auto-organización

Una red neuronal puede crear su propia organización o representación de la información que recibe mediante una etapa de aprendizaje.

3.3.5 Tolerancia a fallos

Las Redes Neuronales fueron los primeros métodos computacionales con la capacidad inherente de tolerancia a fallos. Comparados con los sistemas computacionales tradicionales, que pierden su funcionalidad cuando sufren un pequeño error de memoria, en las Redes Neuronales, si se produce un fallo en un número no muy grande de neuronas, aunque el comportamiento del sistema se ve influenciado, el sistema no sufre una caída repentina.

3.3.6 Operación en tiempo real

Los cómputos neuronales pueden ser realizados en paralelo; para esto se diseñan y fabrican máquinas con hardware especial para obtener esta capacidad.

3.3.7 Fácil inserción dentro de la tecnología existente

Se pueden obtener chips especializados para Redes Neuronales que mejoran su capacidad en ciertas tareas. Ello facilitará la integración modular en los sistemas existentes.

3.4 ARQUITECTURAS DE LAS REDES NEURONALES

Es común representar a las Redes Neuronales como un conjunto de neuronas agrupadas en una o más capas. Cada neurona dentro de una capa se comporta de una manera similar.

A su vez, es importante destacar que los factores que priman a la hora de determinar el comportamiento de una neurona, son su función de transferencia y los valores de los pesos de las conexiones con las neuronas vecinas. Usualmente, dentro de una misma capa, las neuronas presentan la misma función de activación y el mismo patrón de conexión con las neuronas vecinas. El arreglo de las neuronas en capas y el patrón de conexión entre capas se denominan *Arquitectura de la red neuronal*. En la mayoría de las arquitecturas, las neuronas de entrada a la red se activan directamente con el valor de los patrones o señales de entrada.

Las ANN usualmente se clasifican de acuerdo al número de capas como monocapa y multicapa. En el cálculo del número de capas de una arquitectura de una ANN la capa de entrada no es tomada en cuenta ya que sus neuronas no realizan ningún cómputo. La Figura 3.4 ilustra una ANN monocapa y la Figura 3.5 ilustra una ANN multicapa.

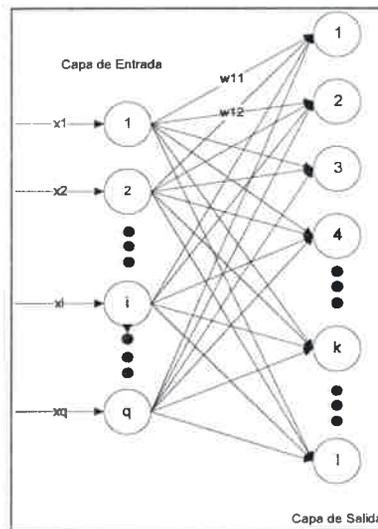


Figura 3.4 - Red Neuronal Monocapa.

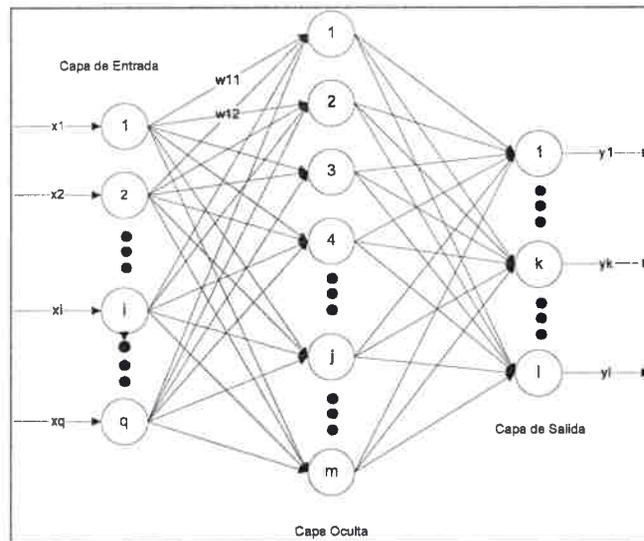


Figura 3.5 - Red Neuronal Multicapa.

3.4.1 Redes Monocapa

En la red monocapa presentada en la Figura 3.4 cada neurona de entrada está conectada con todas las neuronas de la capa de salida pero las neuronas de la capa de entrada y las de salida no se encuentran interconectadas entre sí. Existe una arquitectura donde cada neurona de la capa de salida se conecta con todas las neuronas de esta capa. Esta configuración da a lugar a lo que se llama red competitiva.

3.4.2 Redes Multicapa

Tal como muestra la Figura 3.5 una red multicapa está compuesta por una capa de entrada, una o más capas ocultas y una capa de salida. Las ANN multicapa pueden resolver con éxito problemas que las redes monocapa no son capaces de solucionar, sin embargo el entrenamiento de estas redes es muchas veces más complejo.

3.5 AJUSTE DE LOS PESOS Y METODOS DE ENTRENAMIENTO

En la sección anterior se puso en relieve la importancia de la arquitectura de una red neuronal a la hora de determinar su comportamiento. Además de la arquitectura de la red, la forma en que se calculan los pesos representa un factor importante para definir el comportamiento de una ANN. Se llama *entrenamiento* al proceso que ajusta los pesos de cada capa de la ANN con el objetivo de que la red *aprenda* a resolver un determinado problema [34]. El entrenamiento de la red neuronal reforzará los pesos de las conexiones que favorecen a la convergencia hacia la solución deseada, disminuyendo aquellos pesos que hacen que la red se aleje del resultado esperado. El ajuste de los pesos se realiza de manera iterativa empleando un criterio provisto por el algoritmo de entrenamiento. La etapa finaliza cuando los pesos se estabilizan en un determinado valor que puede o no ser el valor apropiado.

La red se encontrará en condiciones de realizar las tareas para que fue entrenada solo cuando la fase de aprendizaje haya finalizado. Si los pesos no se modifican mientras la red está operativa, se dice que la modalidad de entrenamiento es *off-line*. Del mismo modo, si los pesos son modificados aún cuando la red se encuentra operando, decimos que la modalidad de entrenamiento es *on-line*.

Existen dos tipos de entrenamiento: entrenamiento supervisado y entrenamiento no supervisado.

3.5.1 Entrenamiento supervisado

Es el tipo de entrenamiento más comúnmente seleccionado para ajustar los pesos de la neurona. El entrenamiento supervisado se caracteriza porque el proceso de aprendizaje se realiza mediante un entrenamiento controlado por un agente externo que determina la respuesta que la red debería entregar ante una determinada entrada. El supervisor controla la salida de la red y, en caso que la salida no coincida con la respuesta esperada, dicho proceso modifica los pesos de la red de manera que la respuesta se aproxime a la deseada. Por este motivo se debe contar con un vector que contenga los valores de entrada asociado a un segundo vector que contenga los resultados esperados. A este par de vectores se los conoce como *set de entrenamiento*. Los tres tipos de entrenamiento supervisado más comunes son:

3.5.1.1 Aprendizaje por corrección de error

Consiste en ajustar los pesos de las conexiones de la red en función de la diferencia entre los valores deseados y los obtenidos a la salida, es decir, en función del error cometido en la salida. Esta técnica de ajuste de pesos puede encontrarse en la regla de aprendizaje del Perceptron, la regla del mínimo error cuadrático (LMS) y la regla empleada por las redes Backpropagation denominada LMS multicapa.

3.5.1.2 Aprendizaje por refuerzo

Se basa en la premisa de no disponer de un ejemplo completo del comportamiento deseado. Es decir, en no indicar durante el entrenamiento exactamente la salida deseada.

En esta técnica, la función del supervisor es indicar mediante una señal de refuerzo si la salida obtenida en la red se ajusta a la salida deseada (éxito = 1 o fracaso = -1) y en función de ello se ajustan los pesos basándose en un mecanismo de probabilidades.

3.5.1.3 Aprendizaje estocástico

Consiste en realizar cambios aleatorios en los valores de los pesos de las conexiones de red y evaluar su efecto a partir del objetivo deseado y de distribución de probabilidad. El algoritmo realiza un cambio aleatorio de los valores de los pesos y determina la energía de la red. Si la energía es menor después del cambio, es decir, si el comportamiento de la red se aproxima al deseado, se acepta el cambio. Si, por el contrario, la energía no es menor, se aceptaría el cambio en función de una determinada y preestablecida distribución de probabilidad.

3.5.2 Entrenamiento no supervisado

El entrenamiento no supervisado es típico de las redes denominadas auto organizadas. Una red auto organizada agrupa patrones de entrada de acuerdo a su *similitud* sin necesidad que un agente externo le indique los resultados esperados. En consecuencia, el *set de entrenamiento* no contendrá información relativa a las salidas deseadas.

En este tipo de redes los pesos son ajustados de tal forma que las entradas más similares entre sí sean asignadas a la misma salida (clúster).

Dentro de las redes que utilizan este tipo de entrenamiento se encuentran las *Kohonen-Self Organizing Maps (SOM)* y las redes adaptivas que emplean la teoría de la resonancia.

3.6 FUNCIONES DE ACTIVACION

Como ya se mencionó anteriormente, la arquitectura, el método de entrenamiento y la función de activación de las neuronas son los tres parámetros que definirán el comportamiento de la ANN. En las siguientes sub secciones se presentan las funciones de activación más comúnmente empleadas. [36]

3.6.1 Función identidad

Esta función es generalmente empleada en la capa de entrada de toda red neuronal. En la mayoría de los casos, para poder alcanzar las ventajas de una arquitectura multicapa, las funciones de activación no lineal son requeridas. La ecuación que rige esta función de activación es:

$$f(x) = x \quad \forall x \quad (3.16)$$

La Figura 3.6 muestra la función aquí presentada.

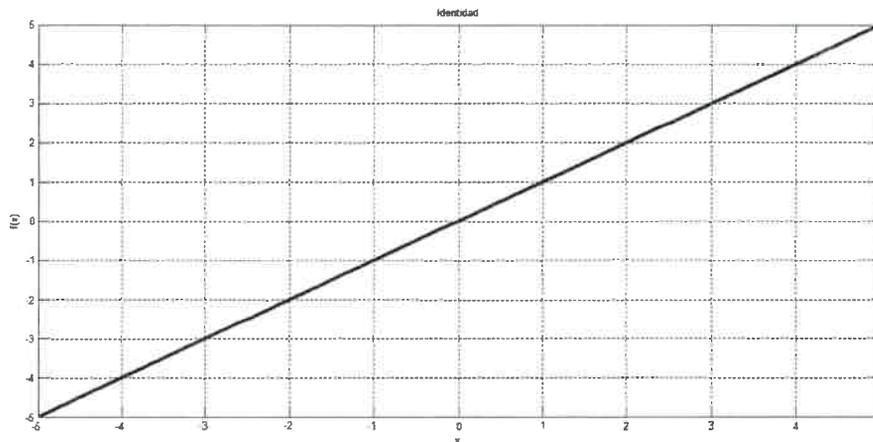


Figura 3.6 - Función de activación lineal.

3.6.2 Función escalón

Las Redes Neuronales monocapa usualmente emplean este tipo de función de activación para convertir la entrada de la red, la cual suele ser una función continua, en una función de salida binaria. Esta conversión se realiza por medio de un umbral. Si la suma de las entradas ponderadas de una neurona supera un cierto umbral, la salida se activa con un 1. Por el contrario, si la mencionada suma es menor que el valor del umbral, la salida de dicha neurona estará en un valor inactivo, el cual puede ser 0 ó -1, dependiendo de la convención utilizada. La ecuación que gobierna esta función de activación es:

$$f(x) = \begin{cases} 1 & \text{Si } x \geq \emptyset \\ 0 & \text{Si } x < \emptyset \end{cases} \quad (3.17)$$

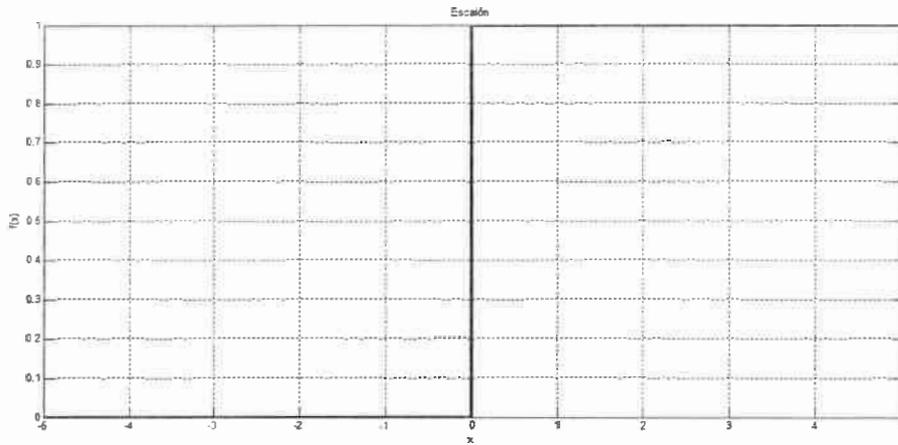


Figura 3.7 - Función de activación escalón.

3.6.3 Función sigmoidea binaria

La principal característica de las funciones sigmoideas (funciones con forma de S) es que pertenecen a las denominadas funciones suaves o funciones de derivada continua. Esta característica es muy importante cuando se implementan algoritmos de entrenamiento que implican la derivada de la función de transferencia de la red neuronal. La función sigmoidea binaria, también llamada función logística, es una función que cambia suavemente del valor 0 al valor 1. La función sigmoidea binaria se representa mediante la siguiente ecuación:

$$f(x) = \frac{1}{1 + \exp(-ax)} \quad (3.18)$$

La Figura 3.8 muestra la influencia del valor de a en la forma de la función.

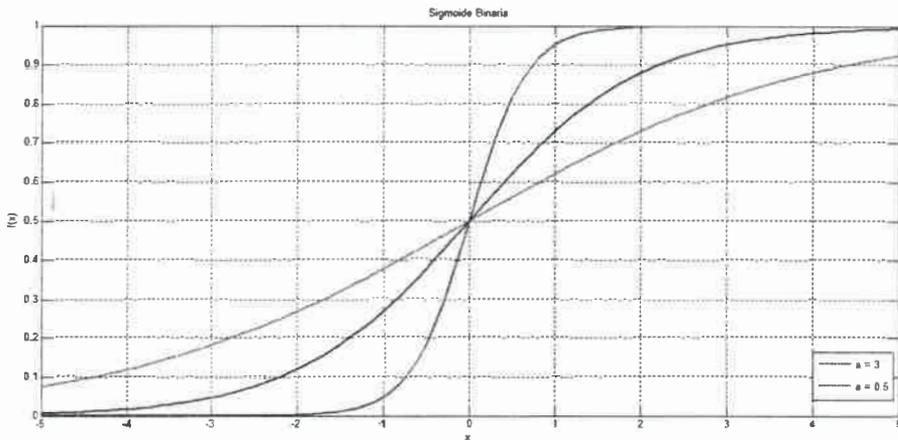


Figura 3.8 - Función de activación sigmoidea binaria.

3.6.4 Función tangente hiperbólica

La función tangente hiperbólica es una función análoga a la función sigmoidea binaria pero se utiliza para representar una transición suave en el rango que abarca desde -1 a 1. La ecuación empleada para representar esta función es:

$$f(x) = \frac{\exp(ax) - \exp(-ax)}{\exp(ax) + \exp(-ax)} \quad (3.19)$$

y es representada en la Figura 3.9 para distintos valores de a .

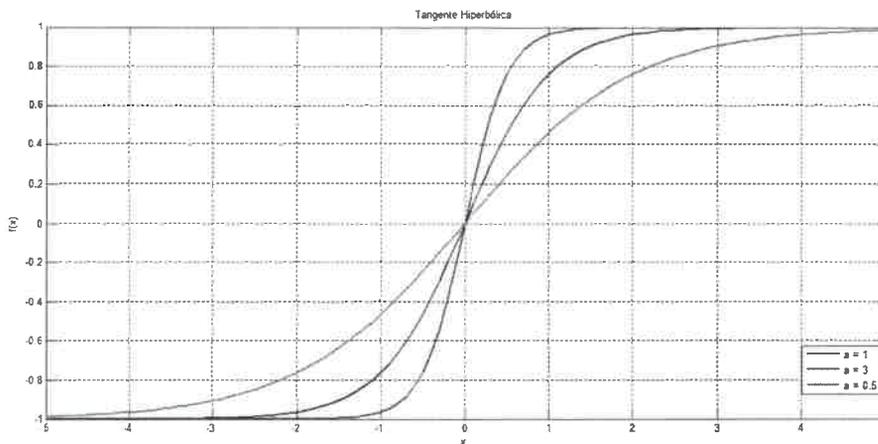


Figura 3.9 - Función de activación tangente sigmoide.

3.7 RED NEURONAL BACK PROPAGATION

Si bien existe una extensa variedad de Redes Neuronales que fueron diseñadas para resolver una amplia gama de problemas, no es objetivo de este trabajo ser exhaustivo en la presentación de los tipos de Redes Neuronales. En esta sección solo se presentará la red neuronal *feedforward backpropagation*, la cual ha probado ser eficientes a la hora de asociar un patrón de entrada con un determinado patrón esperado a la salida.

Estas Redes Neuronales deben su nombre al método de entrenamiento denominado como *backpropagation* (Propagación hacia atrás) o *generalized delta rule* (Regla delta generalizada).

Una Red Neuronal *backpropagation* es una red de propagación hacia adelante (*feedforward*) multicapa entrenada con el algoritmo *backpropagation*.

El algoritmo *backpropagation* se trata de un método que utiliza la técnica del gradiente descendiente para minimizar el error cuadrático de la salida computada por la red.

Este tipo de redes se puede encontrar en cualquier campo donde se empleen Redes Neuronales para resolver problemas que involucran la asociación de patrones.

La asociación de patrones implica que la red neuronal aprenda a tomar decisiones sobre cuál será el patrón de salida más apropiado ante un determinado patrón de entrada. Debido a esto, el tipo de entrenamiento necesariamente es supervisado y se debe contar con un set de patrones de entrada con un correspondiente a un set de patrones esperados a la salida.

Como en la mayoría de las Redes Neuronales, el objetivo del entrenamiento es alcanzar un balance entre la habilidad de responder correctamente a un patrón de entrada empleado en el entrenamiento (memorización) y la habilidad para dar una respuesta razonable a entradas similares pero no idénticas a las usadas en el entrenamiento (generalización).

Como veremos más adelante, el entrenamiento de una red neuronal consiste en las siguientes etapas: La propagación hacia delante de los patrones de entrenamiento en la red; el cálculo de la diferencia entre la respuesta de la red y la respuesta esperada; la propagación del error hacia atrás y el ajuste de los pesos en dirección a aproximar la respuesta de la red a la respuesta esperada. [34]

3.7.1 Arquitectura de la red Backpropagation

La Figura 3.10 muestra la arquitectura de una red multicapa compuesta por una capa de entrada (Unidades X), una única capa oculta (Unidades Z), una capa de salida (Unidades Y), un bias (también conocido como sesgo) en la capa oculta y un segundo bias en la capa de salida.

Los bias en la capa de salida Y_k se denotan por w_{0k} y los bias en la capa de oculta Z_j se denotan por v_{0j}

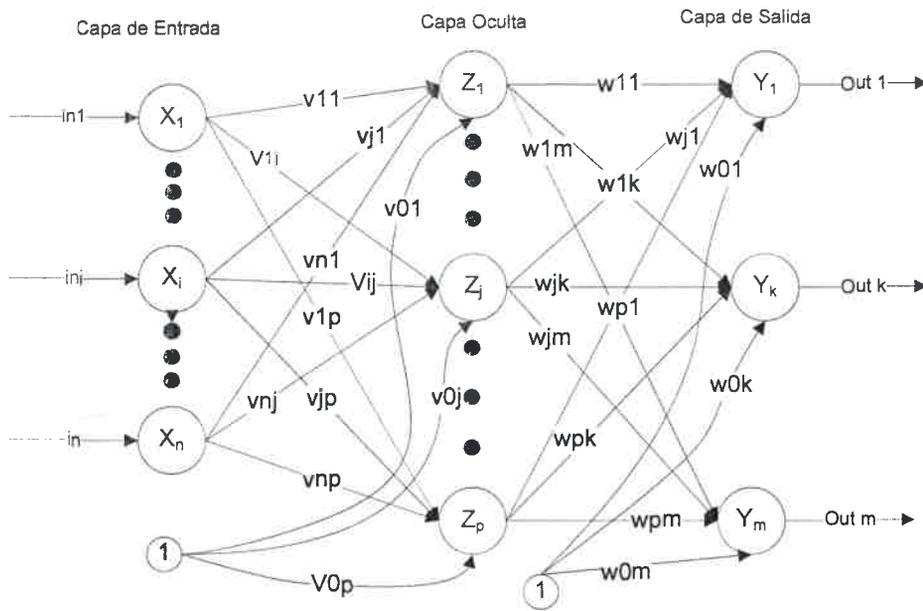


Figura 3.10 - Arquitectura de una red multi-capas.

3.7.2 Funciones de activación

La función de activación de una red backpropagation debe ser continua, diferenciable y monótonamente no decreciente. Cualquier ecuación que cumpla con estas características es candidata para servir de función de activación en una red backpropagation; sin embargo la mayoría de los autores se han concentrado en aquellas funciones cuyas derivadas sean relativamente fáciles de calcular.

Las funciones sigmoidea y sigmoidea bipolar presentadas en la sección 3.6 cumplen con las características mencionadas y además su derivada es conocida y fácil de calcular.

3.7.3 Nomenclatura

- x vector de entrenamiento $x = [x_1\ x_2\ \dots\ x_i\ \dots\ x_n]$.
- t : vector de salidas (target) $t = [t_1\ t_2\ \dots\ t_i\ \dots\ t_m]$.
- δ_k Porción del error para la corrección de los pesos w_{jk} que se produce debido al error a la salida de la unidad Y_k .
- δ_j Porción del error para la corrección de los pesos v_{ij} que se produce debido a la propagación hacia atrás de la información del error desde la capa de salida hacia las unidades de la capa oculta Z_j .
- α : Tasa de aprendizaje.
- X_i : Neuronas de la capa de entrada. La salida de la capa de entrada se llamará x_j .
- v_{0j} : Sesgo de la capa oculta (bias).
- Z_j : Neuronas de la capa oculta.

- La entrada a cada neurona de la capa oculta se representa por: $z_{in j} = v_{0j} + \sum_i x_i v_{ij}$.
- La salida de cada neurona de la capa oculta se representa por $z_j = f(z_{in j})$.
- w_{0k} : Sesgo en la unidad de salida k .
- Y_k : Neurona de salida k .
- La entrada a la unidad Y_k se denota por $y_{in k} = w_{0j} + \sum_j z_j w_{jk}$.
- La salida de la unidad Y_k se denota por $y_k = f(y_{in k})$.

3.7.4 Algoritmo de entrenamiento

Al igual que la mayoría de las ANN, la backpropagation funciona de manera *off-line*. Esto significa que la red solo aprenderá durante la fase de entrenamiento. Una vez finalizada la fase de entrenamiento los pesos de la red permanecerán fijos y la red cumplirá con las funciones para que fuera entrenada.

Como ya se mencionó, el algoritmo de entrenamiento backpropagation involucra tres etapas: a) La propagación hacia adelante de un patrón de entrenamiento, b) La propagación hacia atrás del error y c) El ajuste de los pesos.

Durante la etapa de propagación hacia adelante, cada componente de un patrón de entrenamiento se presenta a su correspondiente neurona de la capa de entrada (X_i). La capa de entrada envía la información a cada neurona de la capa oculta Z_j . Cada unidad de la capa oculta computa las señales que reciben, incorporando en el cálculo el valor del sesgo (bias). La salida de cada unidad de la capa oculta es propagada hacia adelante dentro de la red para servir de entradas de las unidades de la capa de salida. Cada neurona de la capa de salida (Y_k) computa su activación (y_k) para formar la respuesta de la red a un determinado patrón de entrada.

Luego, cada neurona de salida compara su activación y_k con el correspondiente valor del vector objetivo t_k para determinar la diferencia entre la salida obtenida y la salida esperada. Basándose en esta diferencia se genera un vector δ_k el cual es empleado para distribuir el error desde la salida hacia atrás o, lo que es lo mismo, hacia todas las neuronas de las capas ocultas. Este mismo vector es empleado, posteriormente, para la actualización de todos los pesos entre la capa de salida y la capa oculta w_{ij} .

De manera similar, se computa el vector δ_j para cada unidad de la capa oculta Z_j . En este punto, no es necesario propagar el error hacia la capa de entrada, pero δ_j será empleado para ajustar los pesos entre la capa oculta y la capa de entrada w_{jk} .

El algoritmo puede explicarse paso a paso de la siguiente manera:

Paso 0: Inicialización de los pesos.

Paso 1: Chequeo de condición de parada.

Paso 2: Para cada par de entrada hacer Paso 3 y Paso 4.

Inicio del proceso de propagación hacia adelante.

Paso 3: Cada neurona de entrada X_i ($i = 1, \dots, n$) recibe cada componente de la señal de entrada x_i y reenvía esta señal a todas las neuronas de la capa oculta.

Paso 4: Cada neurona de la capa oculta Z_j ($j = 1, \dots, p$) suma sus entradas ponderadas por los respectivos pesos:

$$z_{in j} = v_{0j} + \sum_i x_i v_{ij} \quad (3.20)$$

y aplica la función de activación para obtener:

$$z_j = f(z_{in j}) \quad (3.21)$$

luego envía esta señal a todas las unidades de las capas superiores.

Paso 5: Cada neurona de salida (Y_k $k = 1, \dots, m$) suma sus entradas ponderadas con los respectivos pesos para obtener:

$$y_{in k} = w_{0j} + \sum_j z_j w_{jk} \quad (3.22)$$

y aplica su función de activación para obtener la salida de la red:

$$y_k = f(y_{in k}) \quad (3.23)$$

Inicio del proceso de propagación hacia atrás (backpropagation).

Paso 6: Cada neurona de salida (Y_k $k = 1, \dots, m$) se compara con un patrón esperado correlativo a cada entrada y se computa el término de información del error dado por:

$$\delta_k = (t_k - y_k) f'(y_{in k}) \quad (3.24)$$

posteriormente se calcula el término de corrección de los pesos:

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (3.25)$$

luego se calcula el término de corrección de los bias:

$$\Delta w_{0k} = \alpha \delta_k \quad (3.26)$$

y, finalmente, envía δ_k a las unidades de las capas inferiores.

Paso 7: Cada neurona de la capa oculta ($Z_j, j = 1, \dots, p$) suma su información delta del error (proveniente de las capas superiores) dada por:

$$\delta_{in j} = \sum_{k=1}^m \delta_k w_{jk} \quad (3.27)$$

la multiplica por la derivada de su función de activación para calcular el término de información del error:

$$\delta_j = \delta_{in j} f'(z_{in j}) \quad (3.28)$$

calcula su término de corrección de pesos (utilizado para ajustar v_{ij})

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (3.29)$$

y calcula el término de corrección del bias (utilizado posteriormente para actualizar v_{0j})

$$\Delta v_{0j} = \alpha \delta_j \quad (3.30)$$

Actualización de los pesos y los bias

Paso 8: Cada neurona de salida ($Y_k, k = 1, \dots, m$) actualiza los vías y los pesos ($j = 0, \dots, p$):

$$w_{jk(\text{nuevo})} = w_{jk(\text{viejo})} + \Delta w_{jk} \quad (3.31)$$

cada neurona de la capa oculta ($Z_j = 1, \dots, p$) actualiza sus bias y pesos ($i = 0, \dots, n$):

$$v_{ij(\text{nuevo})} = v_{ij(\text{viejo})} + \Delta v_{ij} \quad (3.32)$$

Paso 9: Se chequea la condición de parada

La condición de parada del algoritmo usualmente se define cuando la diferencia entre la salida obtenida y la salida esperada se encuentra por debajo de un umbral establecido.

El entrenamiento de una RNA backpropagation *no siempre converge* a un valor deseado. Los algoritmos de aprendizaje pueden caer en lo que se denomina mínimos locales y el entrenamiento, sin una segunda condición de parada, podría prolongarse indefinidamente. Se dice que la etapa de aprendizaje cumple con una *época* (en inglés *epoch*) cuando el set de entrenamiento completo es introducido en la RNA. Típicamente se necesitan varias épocas para finalizar con el entrenamiento de una red backpropagation. Además de la comparación del valor del error con un umbral definido, se establece una segunda condición de parada, la cual se emplea para terminar con el algoritmo en caso que el entrenamiento no converja. Esta condición establece un valor

máximo de épocas que, una vez superado, hará que el entrenamiento finalice aún sin alcanzar el error esperado.

Puede observarse que la velocidad de ajuste de los pesos de la red depende del factor α denominado *tasa de aprendizaje*. La tasa de aprendizaje es un factor que determina la velocidad de convergencia del algoritmo de entrenamiento. Tasas de aprendizaje relativamente altas hacen que el algoritmo realice ajustes grandes en los pesos al final de cada época, lo que no garantiza que el entrenamiento converja rápidamente ya que el algoritmo podrá oscilar a medida que se acerque al valor de error esperado. Por el contrario, tasas de aprendizaje pequeñas provocarán que el algoritmo de entrenamiento se aproxime muy lentamente al error esperado.

Las bases matemáticas para el algoritmo backpropagation se basan en una técnica de optimización denominada como *gradiente descendiente*. El gradiente indica la dirección donde una función crece más rápidamente (en este caso la función es el error y las variables independientes son los pesos de la red). El negativo del gradiente indica la dirección en la cual la función decrece más rápidamente.

3.8 CONSIDERACIONES DE CONFIGURACION

Además de la selección de la arquitectura, la función de activación de las neuronas, el algoritmo de entrenamiento y las condiciones de parada del entrenamiento, existen ciertos parámetros que el diseñador podrá seleccionar a la hora de configurar una red.

3.8.1 Selección del valor inicial de los bias y los pesos

La elección de un valor inicial para estos parámetros tiene mucha influencia en la factibilidad y velocidad de convergencia del algoritmo de entrenamiento. Generalmente, estos valores se inicializan de manera aleatoria evitando que la selección de los mismos provoque que una función de activación (o su derivada) sea igual a cero. Un procedimiento común es seleccionar los pesos (y bias) de manera aleatoria dentro del intervalo comprendido entre -1 y 1.

3.8.2 Selección del tamaño del set de entrenamiento

La pregunta a ser respondida en este punto es ¿bajo qué circunstancias se puede asegurar que una red entrenada para clasificar un determinado número de patrones de entrenamiento de manera correcta también va a clasificar correctamente los patrones de pruebas obtenidos del mismo espacio muestral de donde se obtuvieron los patrones de entrenamiento?

Existe una relación entre los patrones de entrenamiento disponibles P , el número de pesos a ser entrenados W y la precisión de la clasificación esperada, e . Esta relación se resume en la siguiente regla:

$$\frac{W}{P} = e \quad (3.33)$$

Por ejemplo, con un error de 0.1 , una red con 80 pesos va a requerir 800 patrones de entrenamiento para asegurar la clasificación de 90% de los patrones de prueba, asumiendo que la red fue entrenada para clasificar correctamente el 95% de los patrones de entrenamiento.

3.8.3 Número de capas ocultas

Para Redes Neuronales con más de una capa oculta el algoritmo backpropagation no sufre alteraciones considerables. Si bien una única capa oculta es suficiente para aproximar cualquier mapeo continuo de un patrón de entrada a un patrón de salida, con un error arbitrario, algunas redes con más de una capa oculta pueden entrenarse más fácilmente en algunas situaciones.

3.8.4 Número de Neuronas en la capa oculta

No hay una manera directa de determinar el número óptimo de neuronas en la capa oculta sin antes entrenar varias redes y estimar el error de generalización.

Muy pocas neuronas en la capa oculta conducirán a un alto error de entrenamiento y un alto error de generalización provocado por el sub-entrenamiento. Por el contrario, muchas neuronas en la capa oculta podrían disminuir el error de entrenamiento pero el

error de generalización podría permanecer en valores no aceptables en este caso debido al sobre-entrenamiento.

Existen reglas informalmente definidas que se conocen como *reglas de oro (rules of thumbs)* que suelen utilizarse para diseñar inicialmente una red neuronal. Estas reglas no definirán de manera exacta la mejor configuración para una red en particular. Por el contrario, la mejor arquitectura deberá ser encontrada teniendo en cuenta el problema en particular por medio de la creación de varias configuraciones y la medición del error de generalización. Algunas de estas reglas para una única capa oculta son:

- La cantidad de neuronas de la capa oculta se puede calcular como las dos terceras partes del número de neuronas de la capa de entrada [37].
- El número de neuronas de la capa oculta debería ser menor que el doble del número de neuronas de la capa de entrada [38].
- El número de neuronas de la capa oculta se debe encontrar entre el número de neuronas de la capa de entrada y el número de neuronas de la capa de salida [39].

3.9 ESTADO DEL ARTE: EMPLEO DE REDES NEURONALES EN SISTEMAS DE MONITOREO

El reconocimiento de patrones de perturbaciones en las muestras de las formas de onda es una tarea compleja debido al gran número de clases y categorías de perturbaciones que pueden verse involucradas en la clasificación. Por este motivo es muy importante contar con un estándar [2] que defina una normativa para establecer una clasificación de los tipos de perturbaciones que se pueden encontrar en un sistema eléctrico de potencia. Como en la mayoría de las tareas de clasificación, el objetivo es etiquetar correctamente el objeto desconocido que, en nuestro caso, es un evento de calidad de energía coincidente con un tipo o clase de perturbación.

El estudio de los problemas que acarrea un sistema que presenta una degradación en los parámetros de calidad de energía generalmente se realiza siguiendo la siguiente secuencia: inicialmente se comienza con la inspección del equipamiento que ha fallado, recabando todos los archivos de registro guardados por el dispositivo. En segundo lugar se obtienen los registros del sistema de monitoreo y se trata de establecer una correlación entre la falla del equipamiento y el tipo de perturbación detectado por el sistema de monitoreo. En tercer lugar se clasifican los eventos de calidad de energía. En cuarto lugar se confirma la relación entre la perturbación y la falla. En quinto y último lugar se realizan acciones para mitigar futuras fallas. En este esquema se resalta la importancia de los sistemas de monitoreo en el estudio de la calidad de energía.

Los primeros sistemas de monitoreo se basaban en métodos de disparo soportados por umbrales rígidos y pre configurados. Un umbral rígido era difícil de configurar ya que si el mismo era establecido con un valor muy alto, una gran cantidad de eventos de calidad de energía ocurrían sin ser detectados por el sistema de monitoreo. Por el contrario, si el umbral se configuraba muy bajo, se detectaba una enorme cantidad de eventos de calidad de energía y los usuarios de los sistemas debían lidiar con grandes cantidades de información para realizar la discriminación entre los distintos tipos de perturbaciones por medio de la inspección visual. Esta primera generación de sistemas de monitoreo fue reemplazada por sistemas con umbrales de detección variables en el tiempo y con sistemas de clasificación automáticos.

El reconocimiento de patrones es un mecanismo mediante el cual se percibe un determinado patrón de un objeto empleando el conocimiento previamente adquirido. Esta tarea, relativamente sencilla para el humano, es muy compleja para ser llevada a cabo por una computadora. Dentro de las áreas de aplicación donde se involucra la tarea del reconocimiento de patrones podemos mencionar el reconocimiento de patrones de textos escritos a mano, identificación de huellas digitales, sistemas de reconocimiento de rostros, aplicaciones de detección de señales oceánicas, solo por mencionar unas pocas.

En el *Capítulo 2* de este trabajo de tesis se presentaron varias estrategias para la extracción de la información característica de las perturbaciones y varias metodologías para realizar la detección automática de los eventos de calidad de energía.

El trabajo de Gosh and Lubkeman (1995) describe una metodología basada en Redes Neuronales para la clasificación de perturbaciones. En esta publicación los autores proponen la extracción de parámetros característicos de las señales perturbadas como los valores pico, valores rms y varios componentes espectrales obtenidos por medio de la *Transformada de Fourier dependiente del tiempo* para el entrenamiento de las *Redes Neuronales*. Finalmente los autores concluyen que la red *Time Delay Neural Network (TDNN)* es mucho más precisa que la red *Feedforward Neural Network (FFNN)* [40].

Surya Santoso (2000) fue uno de los primeros en utilizar un esquema combinado basado en la *Transformada Wavelet* para la extracción de parámetros característicos de las perturbaciones y *Redes Neuronales Artificiales* para la clasificación de perturbaciones. A diferencia del trabajo presentado por Gosh and Lubkeman, Santoso propuso un esquema donde los parámetros característicos se obtienen en el dominio de la *Transformada Wavelet* y se utiliza un esquema de múltiples Redes Neuronales del tipo *Learning Vector Quantization (LVQ por sus siglas en inglés)* para la clasificación de patrones. Su trabajo fue publicado en dos artículos, el primero enfocado en los aspectos teóricos [41] y el segundo apuntó a una implementación práctica de los algoritmos [15].

El principal inconveniente de los trabajos presentados anteriormente radica en su escasa flexibilidad para adaptarse a nuevas perturbaciones. Por este motivo Wijayakulasooriya, Putrus y Minns presentaron un trabajo que utiliza un tipo de *Redes Neuronales Auto Organizadas* denominado *Self Adapting Neural Networks (SANN)*. Los autores proponen la *Transformada Discreta de Fourier (DFT)* para la extracción de parámetros característicos, razón por la cual el sistema de monitoreo puede detectar y clasificar eventos de calidad de energía, pero es incapaz localizar la perturbación en el dominio del tiempo [42].

Monedero, Leon, Roper, Garcia, Elena y Montano (2007) desarrollaron un sistema de monitoreo basado en la *Transformada Wavelet* y una arquitectura de cuatro redes del tipo *Perceptron Multi Layer Neural Network* para la clasificación de eventos de calidad de energía en tiempo real. El trabajo aborda metodologías y técnicas de bajo consumo computacional para lograr una aplicación en tiempo real [43].

El trabajo de Reaz, Choong, Sulaiman Mohd-Yasin y Kamada (2007) presenta un clasificador que utiliza redes neuronales aleatoriamente optimizadas conocidas como *Univariate Random Optimized Neural Network (URONN)* en combinación con la *Transformada Wavelet* y tecnologías basadas en Lógica Difusa [18].

Otro tipo de red neuronal denominado *Probabilistic Neural Network (PNN)* se emplea en un sistema de clasificación desarrollado por Mishra, Bhende y Panigrahi (2008). Este tipo de red neuronal se implementa utilizando un modelo probabilístico conocido como proceso Bayesiano. Las ventajas de este tipo de red neuronal es que no requiere

procesos de aprendizaje y no necesita una configuración previa de sus pesos y, si el número de muestras es lo suficientemente grande, la red neuronal converge a un clasificador bayesiano [44].

Finalmente se destaca el trabajo presentado por Cheng (2012) donde se emplea una red neuronal adaptiva lineal comúnmente conocida como *Adaptive Lineal Neuran Network (ADALINE)*. El empleo de este tipo de red neuronal se ve justificado por su mecanismo de ajuste online, su fácil implementación y sus características propicias para la cancelación de ruido [45].

3.10 CONCLUSIÓN

La primera parte de este capítulo presentó los aspectos generales de una Red *Neuronal Artificial* como su funcionamiento, arquitectura, funciones de activación y algoritmos de entrenamiento.

La segunda parte del capítulo se enfocó en el desarrollo de un tipo de red en particular denominado *feedforward backpropagation*. Este tipo de redes se estudia en detalle debido a su habilidad para resolver problemas de asociación y clasificación de patrones, característica que las presenta como aptas para ser implementadas en un sistema de monitoreo. El capítulo presenta la arquitectura, las características que deben cumplir las funciones de activación y el algoritmo de entrenamiento de una *ANN feedforward backpropagation*. Además se presentan las reglas de diseño utilizadas para este tipo de arreglo.

Finalmente, en la última parte del capítulo, se realiza un breve recorrido por las publicaciones que aplican con éxito las redes neuronales artificiales para la clasificación de eventos de calidad de energía.

Capítulo 4: Máquina de Vectores de Soporte

4.1 INTRODUCCIÓN

En el *Capítulo 3* se presentaron las *Redes Neuronales Artificiales (ANN)* como método para la clasificación de eventos de calidad de energía. Estas herramientas fueron ampliamente utilizadas debido a su estructura de procesamiento en paralelo, su capacidad de aprender y, por lo tanto, de generalizar el conocimiento adquirido [34]. Las propiedades distintivas de una *ANN* son: aprendizaje de conjuntos linealmente no separables, capacidad de relacionar elementos de entrada con elementos de salida, adaptación, tolerancia a fallos y uniformidad de análisis y diseño.

Sin embargo, también es importante señalar que las *ANN* presentan una serie de problemas como: los algoritmos de aprendizaje generalmente involucran un gran número de muestras de entrenamiento; el proceso de entrenamiento generalmente toma mucho tiempo; el entrenamiento puede no finalizar de la manera esperada debido a un atascamiento en un mínimo local y a la no existencia de reglas teóricas para definir la arquitectura de una *ANN* si no que la misma queda definida de manera empírica.

Algunos de estos problemas no se presentan en la técnica de aprendizaje empleada en las *Máquinas de Vectores de Soporte* ya que estos se sustentan en una sólida base estadística por lo que han cobrado un interés creciente en los últimos años.

Las *Máquinas de Vectores de Soporte (SVM por sus siglas en inglés)* es una técnica de máquinas de aprendizaje que se fundamenta en dos bases sólidas: la *teoría de aprendizaje estadístico* y la *optimización matemática*. Esta técnica ha venido siendo aplicada con éxito en una gran variedad de problemas, los cuales van desde reconocimiento de caracteres hasta el reconocimiento de objetos basado en sus patrones de apariencia.

El objetivo de este capítulo es presentar las aplicaciones de los SVM para la clasificación de eventos de calidad de energía en sistemas eléctricos de potencia. Aquí se evitará realizar un exhaustivo análisis de todas las teorías matemáticas involucradas en los SVM, ya que las mismas podrán encontrarse en la bibliografía en este capítulo referenciada.

Algunos de los aspectos que fundamentan el éxito de esta técnica frente a otros métodos de aprendizaje lineal (como las *Redes Neuronales Artificiales*) están relacionados con el hecho que los SVM exhiben una gran habilidad de generalización en muchos problemas de la vida real, la técnica se encuentra bien fundamentada matemáticamente, el proceso de entrenamiento elimina el posible atascamiento en un mínimo local, el modelo presenta pocos parámetros para ajustar y la arquitectura no necesita ser configurada por medio de métodos empíricos. Debido a que esta técnica es relativamente nueva, existe poca bibliografía que aborde este tema desde el punto de vista práctico. A su vez, debido a que la matemática involucrada en este tipo de máquinas de aprendizaje lineal es compleja, este capítulo se enfocará en proporcionar una introducción a los conceptos esenciales acerca de las *Máquinas de Vectores de Soporte*.

En este contexto, la primera parte del capítulo introducirá al lector en las teorías de aprendizaje y la optimización matemática. Estos conceptos serán utilizados en la segunda sección del capítulo para la explicación conceptual de un clasificador binario basado en Máquinas de Vectores de Soporte.

La tercera parte del capítulo detalla las diferentes estrategias para la confección de un clasificador multiclase basado en clasificadores SVM binarios.

Finalmente, en la última sección del capítulo, se detalla la aplicación de esta técnica para la clasificación de eventos de calidad de energía en un sistema eléctrico de potencia.

4.2 TEORIA DE APRENDIZAJE ESTADISTICO

Existe una gran variedad de problemas que no pueden ser resueltos por técnicas de programación clásica ya que no siempre se puede formular un modelo matemático que responda a un problema de la vida real. Un ejemplo comúnmente citado es el problema de reconocimiento de patrones en la escritura a mano alzada. Para resolver este tipo de problemas las *máquinas de aprendizaje* se asoman como la mejor solución.

El proceso de aprendizaje se conoce como *la habilidad de una entidad inteligente para adecuarse a nuevos entornos, para resolver nuevos problemas o, en base a la experiencia, solucionar estos problemas de manera más eficiente.* [46]

4.2.1 Aprendizaje supervisado

Una alternativa para resolver el problema del reconocimiento de texto es realizar un programa que permita relacionar la entrada con la salida por medio de ejemplos, de la misma manera que un ser humano aprende a reconocer las letras por medio de la inspección de los patrones dibujados en un papel. La aproximación que utiliza ejemplos para sintetizar un programa se conoce como metodología de aprendizaje y, cuando se utilizan varios ejemplos para generalizar una relación de entrada y salida, el entrenamiento se denomina *aprendizaje supervisado*.

Si un problema de aprendizaje supervisado tiene como salida una variable binaria se conoce como *clasificación binaria*. Por el contrario, un problema de aprendizaje con un espacio de salida compuesto por más de dos elementos (reconocimiento de patrones de escritura) se conoce como *clasificación multiclase*. Un problema del tipo *regresión* se da cuando el espacio de la salida está compuesto por una función real.

Un problema fundamental de este tipo de entrenamiento es que, incluso cuando es posible encontrar una hipótesis consistente que relacione de manera correcta un dato de entrada con la salida deseada, puede que no sea válida para clasificar correctamente datos que no han sido representados en el set de entrenamiento. La habilidad de un sistema para clasificar correctamente una clase de datos no presente en el set de entrenamiento se conoce como *generalización*, y esta es la propiedad más importante de una máquina de aprendizaje lineal que se desea optimizar.

Para que una máquina de aprendizaje logre una buena habilidad de generalización se debe realizar un correcto entrenamiento. Si el set de entrenamiento está compuesto por un número muy grande de componentes es posible que la máquina memorice la relación en lugar de aprenderla. En este caso se dice que se ha *sobre entrenado* la máquina de aprendizaje y su capacidad de generalización se ve afectada. Una máquina sobre entrenada responderá incorreladamente ante una entrada que difiera (aunque en proporciones ínfimas) a cualquier entrada usada para el entrenamiento.

4.2.2 Limitaciones en el aprendizaje

El método de aprendizaje tiene cuatro limitaciones que son importantes destacar:

- Los algoritmos de aprendizaje pueden atascarse en un mínimo local.
- El tamaño de la hipótesis de salida puede volverse muy grande e impreciso.
- Si hay únicamente un número limitado de ejemplos demasiado ricos en información, un clasificador podrá sobre entrenarse y la habilidad de generalización se verá afectada.
- Usualmente los algoritmos de entrenamiento son controlados por un gran número de parámetros que complejizan al sistema y lo hacen difícil de utilizar.

4.2.3 Proceso de Aprendizaje para las Máquinas de Vectores de Soporte

Los SVM son sistemas de aprendizaje que utilizan espacios de hipótesis de funciones lineales en un espacio característico de mayor dimensión, entrenados con un algoritmo que se basa en la teoría de optimización matemática y que implementan un sesgo en el entrenamiento proveniente de la teoría de aprendizaje estadístico.

4.3 TEORIA DE LA OPTIMIZACION MATEMATICA

4.3.1 Conceptos

El entrenamiento de SVM involucra la resolución de un problema de optimización matemática. Por este motivo, con el objetivo de entender el funcionamiento de la etapa de aprendizaje del algoritmo de SVM es importante un conocimiento previo sobre la teoría de *optimización matemática*.

La teoría de la optimización es la rama de las matemáticas relacionada con la tarea de hallar la mejor solución dentro de un conjunto de posibles respuestas a un problema y desarrollar algoritmos que encuentren dichas soluciones de manera eficiente.

En el caso más simple, un problema de optimización consiste en maximizar o minimizar una función real eligiendo sistemáticamente valores de entrada (tomados de un conjunto permitido) y computando el valor de la función. De forma general, la optimización incluye el descubrimiento de los "mejores valores" de alguna función objetivo dado un dominio definido, incluyendo una variedad de diferentes tipos de funciones objetivo y diferentes tipos de dominios [47].

En esta sección solo se describen los casos en los que la función principal a optimizar es una función cuadrática convexa y todas sus restricciones son lineales. Esta clase de problemas es la que se utiliza para el entrenamiento de los SVM.

4.3.2 Formulación del problema

El objetivo de resolver un problema de optimización matemática es *encontrar el valor máximo o el valor mínimo de una función sujeto a diferentes tipos de restricciones*. La definición formal es:

Dadas las funciones $f, g_i \ i = 1, 2, \dots, k$ y $h_i \ i = 1, 2, \dots, m$ definidas en el dominio $\Omega \subseteq \mathbb{R}^n$

$$\text{minimizar } f(w) \quad w \in \mathbb{R}^n \quad 4.1$$

Sujeto a

$$g_i(w) \leq 0 \quad i = 1, 2, \dots, k \quad 4.2$$

$$h_i(w) = 0 \quad i = 1, 2, \dots, m \quad 4.3$$

donde $f(w)$ se denomina *función objetivo* y las restantes funciones son llamadas respectivamente *restricciones de desigualdad* (g_i) y *restricciones de igualdad* (h_i).

El valor óptimo de la función objetivo se conoce como valor del problema de optimización.

La región del dominio donde la función objetivo es definida y donde todas las restricciones son satisfechas se llama *región de factibilidad* y se denota por:

$$R = \{w \in \Omega, g(w) \leq 0, h(w) = 0\} \quad 4.4$$

Una solución al problema de optimización se corresponde con el punto $w^* \in \mathbb{R}^n$ de tal forma que no exista otro punto $w \in \mathbb{R}$ tal que $f(w) < f(w^*)$. Este punto es conocido como *mínimo global*.

Un punto $w^* \in \Omega$ se llama *mínimo local* de $f(w)$ si $\exists \xi > 0$ tal que:

$$f(w) \geq f(w^*), \forall w \in \Omega \text{ tal que } \|w - w^*\| \leq \xi \quad 4.5$$

4.3.3 Definiciones importantes

El entrenamiento de un SVM involucra un problema de optimización matemática cuyas restricciones serán limitadas a *funciones lineales* y la función objetivo será una función *convexa cuadrática*. Antes de abordar el método de entrenamiento de un SVM es preciso realizar las siguientes definiciones:

4.3.3.1 Programación lineal

Un problema de optimización matemática cuya función objetivo y las restricciones son funciones lineales es llamado problema de *programación lineal*.

4.3.3.2 Programación cuadrática

Un problema de optimización matemática cuya función objetivo es una función cuadrática y las restricciones son funciones lineales es llamado *problema de optimización cuadrática*.

4.3.3.3 Función cóncava y convexa

Una función real f definida en un intervalo (o en cualquier conjunto convexo C de algún espacio vectorial) se dice que es cóncava, si para dos puntos x e y cualesquiera definidas en su dominio C , y para cualquier t en $[0, 1]$, se cumple:

$$f(tx + (1 - t)y) \geq tf(x) + (1 - t)f(y) \quad 4.6$$

Además, $f(x)$ es cóncavo en $[a, b]$ si y sólo si la función $-f(x)$ es convexa en $[a, b]$.

Si una función $f(x)$ es *convexa* para todo x entonces si existe un valor x^* para el cual la función $f(x^*)$ asume su valor mínimo; entonces se puede asegurar que ese valor se corresponde con un *mínimo global*. [48]

4.3.3.4 Transformación de las restricciones

Existe una estrategia para transformar las restricciones de desigualdad en restricciones de igualdad. El método consiste en incorporar a la función una variable ξ denominada *variable de relajación* de la siguiente manera.

$$g_i(w) \leq 0 \Leftrightarrow g_i(w) + \xi_i = 0 \quad \xi \geq 0 \quad 4.7$$

4.3.4 Teoría de Lagrange

El propósito de la teoría de Lagrange radica en caracterizar la solución de un problema de optimización matemática donde inicialmente se propondrá que no existan restricciones de desigualdad. Antes de abordar la teoría de Lagrange es preciso realizar la primera definición:

Definición 1: Teorema de Fermat

Una condición necesaria para que w^* sea un mínimo de $f(w)$, $f \in C^1$ es que la derivada de la función en ese punto sea nula.

$$\frac{\partial f(w^*)}{\partial w} = 0 \quad 4.8$$

Una función f es de clase C^1 si su derivada primera existe y es continua.

La teoría se fundamenta en dos conceptos básicos: los multiplicadores de Lagrange y las funciones lagrangianas.

Definición 2: Multiplicador de Lagrange y función lagrangiana

Dado un problema de optimización matemática, con función objetivo $f(w)$ y restricciones de igualdad $h_i(w) = 0$ para $i = 1, 2, \dots, m$ se define la función lagrangiana como:

$$\mathcal{L}(w, \beta) = f(w) + \sum_{i=1}^m \beta_i h_i(w) \quad 4.9$$

donde β_i se conoce como *multiplicador lagrangiano*.

Si un punto w^* es un mínimo local para un problema de optimización con únicamente condiciones de igualdad es posible que $\frac{\partial f(w^*)}{\partial w} \neq 0$. Esto quiere decir que es posible moverse en alguna dirección de w , violando algunas de las restricciones, y encontrar que $f(w)$ disminuye su valor.

Para respetar todas las restricciones de igualdad h_i es preciso moverse de manera perpendicular a $\frac{\partial h_i(w^*)}{\partial w}$. De esta manera, para respetar todas las restricciones nos debemos mover de manera perpendicular al sub espacio V creado por:

$$\left\{ \frac{\partial h_i(w)}{\partial w} \quad i = 1, 2, \dots, m \right\} \quad 4.10$$

Si todas las $\frac{\partial h_i(w^*)}{\partial w}$ son linealmente independientes, ningún movimiento legal va a cambiar el valor de la función objetivo, sin importar donde $\frac{\partial h_i(w^*)}{\partial w}$ se encuentre en el sub espacio V . En otras palabras, cuando existe un valor de β_i tal que:

$$\frac{\partial f(w)}{\partial w} + \sum_{i=1}^m \beta_i h_i(w^*) = 0 \quad 4.11$$

Esta observación fundamenta la siguiente definición.

Definición 3: Multiplicador de Lagrange y función lagrangiana

Una condición necesaria que un punto w^* debe cumplir para ser un mínimo de $f(w)$ sujeto a $h_i(w)=0, i=1,2,\dots,m$ con $f, h_i \in C^1$ es:

$$\frac{\partial \mathcal{L}(w^*, \beta^*)}{\partial w} = 0 \quad 4.12$$

$$\frac{\partial \mathcal{L}(w^*, \beta^*)}{\partial \beta} = 0 \quad 4.13$$

para algunos valores de β^* . Las condiciones arriba mencionadas son también suficientes debido a que $\mathcal{L}(w, \beta^*)$ es también una función convexa de w .

Las tres definiciones presentadas hasta el momento abren el camino para una definición más completa, que contempla no sólo las restricciones de igualdad si no que también contempla las restricciones de desigualdad.

Definición 4: Teorema de Kuhn Tucker

Dado un problema de optimización matemática cuya función objetivo es una función convexa $\Omega \subseteq \mathbb{R}$

$$\text{minimizar } f(w) \quad w \in \Omega \quad 4.14$$

Sujeto a

$$g_i(w) \leq 0 \quad w \in \Omega \quad 4.15$$

$$h_i(w) = 0 \quad w \in \Omega \quad 4.16$$

con $f \in C^1$ convexa y g_i, h_i son restricciones de desigualdad e igualdad respectivamente. Las condiciones necesarias y suficientes para que un punto w^* sea un valor óptimo es la existencia de un α^* y β^* tal que:

$$\frac{\partial \mathcal{L}(w^*, \alpha^*, \beta^*)}{\partial w} = 0 \quad 4.17$$

$$\frac{\partial \mathcal{L}(w^*, \alpha^*, \beta^*)}{\partial \beta} = 0 \quad 4.18$$

$$\alpha_i^* g_i(w^*) = 0 \quad i = 1, \dots, k \quad 4.19$$

$$g_i(w^*) \leq 0 \quad i = 1, \dots, k \quad 4.20$$

$$\alpha_i^* \geq 0 \quad i = 1, \dots, k \quad 4.21$$

La tercera condición se conoce como condición complementaria de *Kurush-Kuhn-Tucker (KKT)*. Esta condición implica que para restricciones activas $\alpha^* \geq 0$ y para restricciones inactivas $\alpha^* = 0$

Una forma de interpretar los resultados presentados anteriormente implica que la solución pueda encontrarse en una o dos posibles posiciones con respecto a la restricción de desigualdad, o bien en el interior de la región de factibilidad, con la

restricción inactiva, o bien en el límite de la región de factibilidad cuando la restricción se encuentra activa. En el primer caso las condiciones de optimalidad para tales restricciones están dadas por el *teorema de Fermat* y, por lo tanto α_i debe ser cero. Para el segundo caso se utiliza el *teorema de Lagrange* con un valor no cero de α_i . Entonces las condiciones *KKT* implican si una restricción es activa, significa que $g_i(w^*) = 0$, o que el multiplicador satisface que $\alpha_i = 0$. Eso se representa en la ecuación $\alpha^* g_i(w^*) = 0$.

4.3.5 Dualidad

La resolución de problemas de optimización convexa empleando el tratamiento de Lagrange puede realizarse por medio de una descripción alternativa dual que hace que sea más fácil resolver del problema principal ya que la solución de las restricciones de desigualdad suele ser una tarea compleja. El problema dual se obtiene por medio de la introducción de multiplicadores lagrangianos, también llamados *variables duales*.

Los métodos duales se basan en la idea de que las *variables duales* son la incógnita principal del problema de optimización. Se puede transformar un problema primario en un problema dual fijando a cero la derivada del lagrangiano con respecto a las variables primarias y substituyendo las relaciones obtenidas en el lagrangiano removiendo de esta manera la dependencia de las variables primarias. Esto se corresponde a computar la siguiente función:

$$\theta(\alpha, \beta) = \min_{w \in \Omega} \mathcal{L}(w, \alpha, \beta) \quad 4.22$$

La función resultante contiene únicamente variables duales y debe ser maximizada bajo restricciones simples.

4.4 MAQUINAS DE SOPORTE VECTORIAL

Las Máquinas de Vectores de Soporte (SVM) pueden ser aplicadas en problemas de reconocimiento de patrones, problemas de regresión, extracción de características o detección de novedades. Como el objetivo del presente trabajo de tesis es estudiar algoritmos de reconocimiento y clasificación de patrones aquí solo se abordarán los aspectos relacionados a este tipo de problemas.

Una característica importante de este método es que el algoritmo es entrenado empleando representaciones duales y ralas de las restricciones, lo que resulta en algoritmos extremadamente eficientes. La eficiencia es una de las características fundamentales que distingue a un clasificador basado en *Máquinas de Vectores de Soporte* frente a otras máquinas lineales como las *Redes Neuronales Artificiales* por ejemplo.

Otra característica importante de las aproximaciones que se basan en un SVM radica en el hecho que el problema siempre involucra problemas de optimización matemática convexa lo que elimina la posibilidad de la aparición de mínimos locales. Los mínimos locales son un gran problema en el entrenamiento de otros algoritmos de clasificación.

4.4.1 Clasificador de vectores de soporte

El objetivo de un clasificador que emplea vectores de soporte es el de proveer una forma computacional y eficiente de aprendizaje que realice una óptima separación de dos clases por medio de un hiperplano ubicado en un espacio característico de mayor dimensión.

Un SVM primero mapea todos los puntos de entrada a un espacio característico de mayor dimensión. Por ejemplo, si los datos de entrada están representados en \mathbb{R}^2 los SVM mapean estos puntos a una dimensión en \mathbb{R}^3 para luego encontrar un hiperplano óptimo que los separe. Existen varios métodos para encontrar el hiperplano óptimo, dentro de los cuales se puede destacar la estrategia de optimizar el margen máximo, optimizar la distribución del margen u optimizar el número de vectores de soporte.

El modelo más simple, y por lo tanto el primero en ser abordado en este trabajo, es el *clasificador de máximo margen*.

4.4.2 Clasificador de máximo margen

Si se tiene un problema de clasificación de patrones de datos linealmente separables como el mostrado en la Figura 4.1 el problema del clasificador de máximo margen es el de encontrar un hiperplano que separe ambas clases maximizando el margen m .

Un problema de clasificación binario puede ser de dos clases: *linealmente separables* o *no linealmente separables*. El primero de los dos problemas se corresponde con el caso más sencillo y, por este motivo, será detallado en primer lugar para luego hacer extensivo el método a los casos no linealmente separables.

4.4.2.1 Clasificador linealmente separables

Maximizar el margen m es un problema de optimización matemática con una función objetivo *cuadrática convexa* y puede ser resuelto mediante la proposición de un problema dual incorporando multiplicadores de Lagrange.

Sin ningún conocimiento del mapeo, un SVM encuentra el hiperplano óptimo utilizando el producto punto con funciones en el espacio de características que son llamadas *kernels*. La solución del hiperplano óptimo puede ser descrita como la combinación de unos pocos puntos de entrada que son llamados *vectores de soporte*.

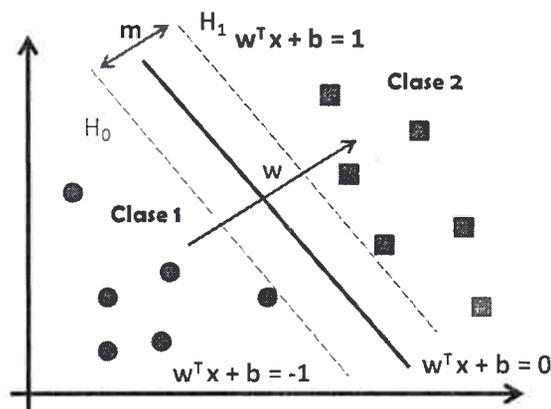


Figura 4.1 - Fronteras de decisión.

Supongamos el caso mostrado en la Figura 4.2 en donde, para un mismo conjunto de datos, se encuentran dos planos que cumplen con la función de separar las dos clases presentes originalmente.

Se define el margen m como la distancia entre dos puntos dados, de ambas clases, más próximos a la superficie de decisión que, en nuestro caso, es un hiperplano.

Como se puede ver, el margen m_1 encontrado en la figura Figura 4.2 a) es más estrecho al margen m mostrado en la figura Figura 4.2 b). Este ejemplo muestra que no hay un único plano que separa ambas clases de datos, sin embargo existe un plano óptimo que maximiza el margen m .

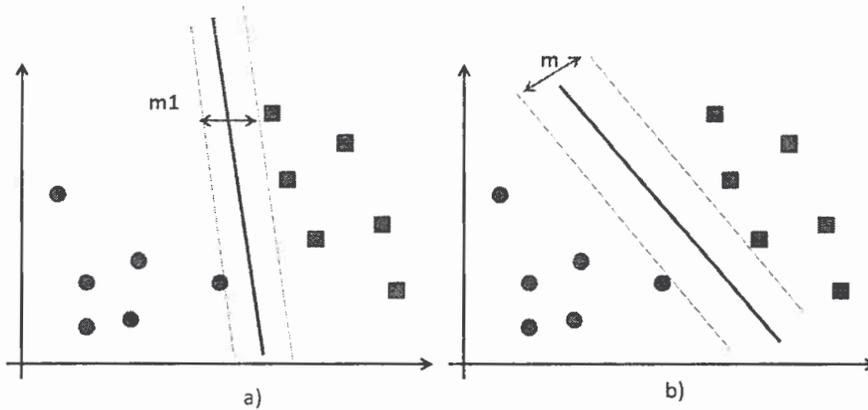


Figura 4.2 - Caso linealmente separable

La *teoría de aprendizaje* permite controlar la capacidad de generalización y la posibilidad de sobre-entrenamiento (*overfitting*) a través del control de los parámetros del hiperplano [46].

En este trabajo se considera la manera más común de encontrar el plano óptimo de separación, la cual consiste en utilizar herramientas de optimización matemática para reducir la norma de un vector w también conocido como vector peso [49].

Considere un conjunto de puntos de entrada S , dado $x_i \in \mathbb{R}^n$ con $i = 1, 2, \dots, N$. Cada punto x_i pertenece a una de dos clases establecidas por lo que se puede establecer una etiqueta $y_i \in \{1, -1\}$. Supongamos que existe un hiperplano que separa ambas clases de datos. Como se puede ver en la Figura 4.2 los puntos sobre el hiperplano que satisfacen $w x + b = 0$ donde w es un hiperplano perpendicular (normal) al hiperplano, $\frac{|b|}{\|w\|}$ es una distancia perpendicular del hiperplano al origen y $\|w\|$ es la norma Euclidiana de w . Sea d^+ la menor distancia entre el hiperplano de separación y los puntos que representan la clase positiva (+1) y d^- la menor distancia entre el hiperplano de separación y los puntos más próximos a la frontera negativa (-1). La frontera del hiperplano debe ser, por lo tanto $d^+ + d^-$.

El algoritmo de SVM procura un hiperplano de separación con un margen m máximo que puede ser construido como sigue:

Suponiendo que todos los datos de entrenamiento satisfacen las siguientes restricciones:

$$x_i w + b \geq +1 \quad \text{para } y_i = +1 \quad 4.23$$

$$x_i w + b \leq -1 \quad \text{para } y_i = -1 \quad 4.24$$

Estas dos ecuaciones pueden ser combinadas en una única restricción de la siguiente manera:

$$y_i(x_i w + b) - 1 \geq 0 \quad \forall i = 1, 2, \dots, N \quad 4.25$$

Aquellos puntos que cumplen con la igualdad propuesta en la ecuación 4.23 se corresponden al hiperplano H_1 de la Figura 4.1, que poseen una norma w y una distancia al origen dada por $\frac{[1-b]}{\|w\|}$. De la misma manera los puntos correspondientes a la igualdad de la ecuación 4.24 se corresponde con el hiperplano H_0 , nuevamente con una norma dada por w y una distancia perpendicular al origen dada por $\frac{[-1-b]}{\|w\|}$. Por lo tanto, $d^+ = d^- = \frac{1}{\|w\|}$ y el margen es simplemente $\frac{2}{\|w\|}$.

Es importante destacar que H_1 y H_2 son paralelos (tienen la misma norma) y no hay ningún punto entre ellos.

Por lo tanto, una definición del hiperplano óptimo como superficie de decisión para la clasificación de dos clases de datos es la siguiente:

Dado un conjunto de datos linealmente separables, representados de la siguiente manera $S = [(x_1, y_1), \dots, (x_l, y_l)]$, un hiperplano representado por $wx + b = 0$ puede ser encontrado solucionando el siguiente problema de optimización:

$$\begin{aligned} & \text{minimizar}_{w,b} (w \cdot w) && 4.26 \\ & \text{sujeto a } y_i(x_i w + b) \geq 1 \quad \forall i = 1, 2, \dots, l \end{aligned}$$

Una solución para el caso bidimensional deberá tener la forma representada en la Figura 4.3. Los puntos para los que aplica la igualdad dada en la ecuación 4.25 son aquellos que están sobre los hiperplanos H_0 y H_1 y se ven resaltados de sus pares por medio de una representación sin color de relleno. Estos puntos son llamados *vectores de soporte*.

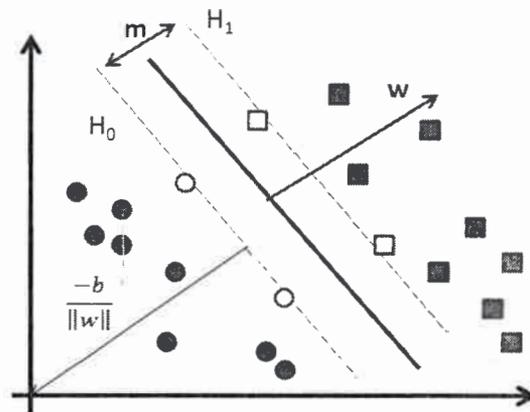


Figura 4.3 - Hiperplano de separación

Para transformar el problema presentado en 4.26 en un problema de optimización cuadrática convexa se debe realizar la siguiente transformación:

$$\begin{aligned} & \text{minimizar}_{w,b} \frac{1}{2} (w \cdot w) && 4.27 \\ & \text{sujeto a } y_i(x_i w + b) \geq 1 \quad \forall i = 1, 2, \dots, l \end{aligned}$$

Conforme lo presentado en la sección 4.3.4 se puede utilizar el método de los *multiplicadores Lagrange* para resolver el problema de optimización cuadrático arriba mostrado.

- La restricción presentada en 4.25 es sustituida por una nueva restricción que se define en función de los multiplicadores de Lagrange, lo que la hace más fácil de resolver computacionalmente.
- En esta formulación del problema, los datos de entrenamiento aparecen en forma de un producto interno entre vectores.

Introduciendo los multiplicadores de Lagrange $\alpha_i, i = 1, \dots, l$, se realiza el producto entre la restricción de la ecuación 4.25 y los multiplicadores positivos y ese producto es sustraído de la función objetivo para formar la función de Lagrange. Por lo tanto, para solucionar el problema de optimización propuesto en la ecuación 4.27 se debe encontrar el punto silla de la siguiente función de Lagrange:

$$\mathcal{L}_p = \frac{1}{2}(w, w)^2 - \sum_{i=1}^l \alpha_i y_i (x_i w + b) + \sum_{i=1}^l \alpha_i \quad 4.28$$

De acuerdo a las ecuaciones 4.13 y 4.14 las derivadas de \mathcal{L}_p con relación a w y b deben ser nulas y esto se corresponde al hecho que el punto óptimo tiene las siguientes ecuaciones de punto silla:

$$\frac{\partial \mathcal{L}_p(w, b, \alpha)}{\partial w} = w - \sum_{i=1}^l y_i \alpha_i x_i = 0 \quad 4.29$$

$$\frac{\partial \mathcal{L}_p(w, b, \alpha)}{\partial b} = \sum_{i=1}^l y_i \alpha_i = 0 \quad 4.30$$

Sustituyendo las relaciones obtenidas se tiene:

$$w = \sum_{i=1}^l y_i \alpha_i x_i \quad 4.31$$

$$\sum_{i=1}^l y_i \alpha_i = 0 \quad 4.32$$

Las nuevas restricciones formuladas en las ecuaciones 4.31 y 4.32 pueden ser sustituidas en la ecuación 4.28 de forma que:

$$\mathcal{L}_D = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle x_i x_j \rangle \quad 4.33$$

Es importante destacar que \mathcal{L}_p es el problema primario y \mathcal{L}_D se denomina el problema dual. Una solución es encontrada por medio de la minimización de \mathcal{L}_p o

la maximización de \mathcal{L}_D . Existe un multiplicador de Lagrange para cada punto del entrenamiento.

Además de las restricciones arriba expuestas, las condiciones complementarias de *Karush-Kuhn-Tucker* (*KKT*) presentadas en la sección 4.3.3 proporcionan información útil sobre la estructura de la solución. Las condiciones dicen que, para una solución óptima α^*, w^* y b^* deben satisfacer:

$$\alpha_i^* [y_i(w^* x_i + b^*) - 1] = 0 \text{ para } i = 1, \dots, l \tag{4.34}$$

Para problemas convexos, las condiciones de *KKT* son necesarias y suficientes para que α^*, w^* y b^* sean solución, entonces, encontrar una solución para un SVM es equivalente a encontrar una solución para las condiciones de *KKT*.

Como los llamados vectores de soporte poseen α_i^* no nulos, ellos son los únicos involucrados en la expresión de los pesos w , por lo tanto, el vector que representa el hiperplano de margen máximo es calculado como una combinación lineal de la siguiente manera:

$$w^* = \sum_{i=1}^l y_i \alpha_i^* x_i \tag{4.35}$$

Puede ser rescrita en función de los vectores de soporte:

$$w^* = \sum_{\text{vectores de soporte}} y_i \alpha_i^* x_i \tag{4.36}$$

Reescribiendo el problema nuevamente, ahora colocando la expresión para w^* en función del problema dual de Lagrange y teniendo en cuenta las condiciones de *KKT* el problema cuadrático convexo de SVM se transforma en el siguiente:

Dado un $\alpha_0 = (\alpha_1^0, \dots, \alpha_l^0)$ una solución para el problema arriba propuesto, entonces

$$W(\alpha) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l y_i y_j \alpha_i \alpha_j \langle x_i x_j \rangle \tag{4.37}$$

Sujeto a

$$\begin{aligned} \sum_{i=1}^l y_i \alpha_i &= 0 \\ \alpha_i &\geq 0 \quad i = 1, \dots, l \end{aligned} \tag{4.38}$$

una norma del vector w que corresponde al hiperplano óptimo es igual a:

$$\|w\|^2 = 2W(\alpha_0) = \sum_{\text{vectores de soporte}} \alpha_i^0 \alpha_j^0 \langle x_i x_j \rangle y_i y_j \quad 4.39$$

La regla de separación, con base en el hiperplano óptimo es la función indicadora siguiente:

$$f(x) = \text{sgn} \left(\sum_{\text{vectores de soporte}} \alpha_i^0 \alpha_j^0 \langle x_i x_j \rangle y_i y_j \right) \quad 4.40$$

en el cual x_i son los vectores de soporte y α_0 son los coeficientes de Lagrange correspondientes a los vectores de soporte y b_0 es un umbral constante dado por:

$$b_0 = \frac{1}{2} [\langle w_0 x^*(1) \rangle + \langle w_0 x^*(-1) \rangle] \quad 4.41$$

donde $x^*(1)$ se corresponde a cualquier vector de soporte que pertenece a la primera clase y $x^*(-1)$ se corresponde a cualquier vector de soporte perteneciente a la segunda clase.

Un clasificador de mínimo margen queda restringido a aplicaciones cuando los datos son linealmente separables.

4.4.2.2 Clasificador no linealmente separables

Para tratar los casos linealmente no separables es necesario relajar las restricciones del problema. Si el conjunto S no es linealmente separable, violaciones a la clasificación deben ser permitidas en la formulación de la SVM.

De este modo, las restricciones presentadas en las ecuaciones 4.23 y 4.24 denominadas restricciones de margen duro (*hard margin*) son transformadas en restricciones de margen blando (*soft margin*) mediante la incorporación de una variable de relajación ξ_i $i = 1, \dots, l$ de la siguiente manera:

$$x_i w + b \geq +1 - \xi_i \quad \text{para } y_i = +1 \quad 4.42$$

$$x_i w + b \leq -1 + \xi_i \quad \text{para } y_i = -1 \quad 4.43$$

Esta estrategia permite una cierta tolerancia a datos denominados *outliers*. Un *outlier* es un dato muy distante de los restantes puntos de la clase a la que pertenecen. Por lo tanto la expresión $\sum_{i=1}^N \xi_i$ representa el número de errores de tratamiento. Debido a esto, con el fin de poder representar los costos extras por los errores que resultan de la adición de las variables de holgura, existe la necesidad de cambiar la función objetivo presentada en la ecuación 4.27 para el caso *linealmente separable* de la siguiente manera:

$$\frac{1}{2} \|w\|^2 + C \left(\sum_{i=1}^N \xi_i \right)^k \quad 4.44$$

donde el parámetro C es una constante que actúa como una función de penalidad previniendo que los *outliers* afecten el cálculo del hiperplano óptimo.

Un valor grande de C hace que se asuma mayor penalidad por incorporar errores. Por tratarse de problemas de programación convexa k puede ser cualquier valor entero positivo en particular $k=1$ y $k=2$ se trata de un problema de programación cuadrática. Para el caso del valor más pequeño de $k>0$ tiene la ventaja de no ser necesario que ξ_i y sus multiplicadores de Lagrange aparezcan en el problema dual. El problema con esta modificación se vuelve:

$$\mathcal{L}_D \equiv \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle x_i x_j \rangle \tag{4.45}$$

sujeto a:

$$\sum_{i=1}^l y_i \alpha_i = 0 \tag{4.46}$$

$$0 \leq \alpha_i \leq C \quad i = 1, \dots, l$$

y la solución viene dada por

$$w = \sum_{\text{vectores de soporte}} \alpha_i y_i x_i \tag{4.47}$$

Así que la única diferencia para el caso del hiperplano óptimo es que α_i tiene un límite superior en C . Esta situación se ve representada en la Figura 4.4.

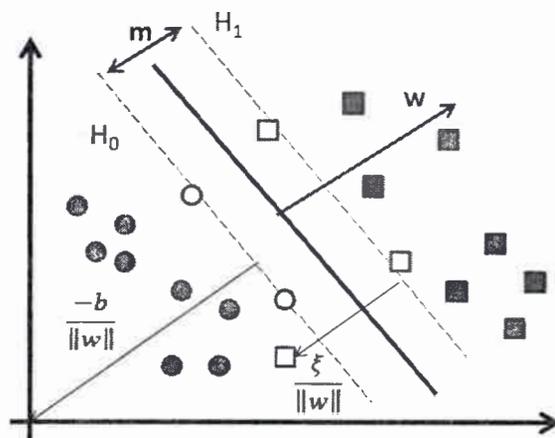


Figura 4.4 - Hiperplano para el caso no linealmente separable

4.4.3 Transformación de sub espacio usando funciones Kernel

De acuerdo al teorema de Cover un problema complejo de clasificación de patrones tiene más probabilidades de ser separable linealmente mientras mayor sea la dimensión del espacio característico {Cristianini, 2000 #76}.

Considerando un espacio de entrada en donde los patrones no son linealmente separables, el teorema de Cover dice que ese espacio puede ser transformado en un

nuevo espacio característico donde los patrones tienen mayor posibilidad de ser linealmente separables sobre las siguientes dos condiciones: la transformación nunca puede ser lineal y la dimensión del nuevo espacio característico debe ser de mayor que la dimensión del espacio característico de los datos de entrada.

Una vez transformado el espacio característico la idea nuevamente recae en construir un hiperplano de margen máximo de acuerdo a lo descrito en la sección 4.4.2 para clasificar las dos clases de datos pero con una diferencia fundamental: *el hiperplano de margen máximo ahora se construye como una función lineal de vectores del espacio característico transformado en lugar de hacerlo con los vectores del espacio característico original.*

Vapnik [46] observó que, para la construcción del hiperplano de margen máximo en un espacio característico Z , no es necesario considerar todo el espacio de forma explícita. En su lugar, se puede simplemente calcular el producto interno entre los vectores de soporte y los vectores del espacio característico.

Suponiendo que se establecieran patrones $x \in \mathbb{R}^N$ donde la mayor parte de la información está contenida en los monomios de grado d donde:

$$x_{j_1}, \dots, x_{j_d}$$

donde $j_1, \dots, j_d \in \{1, \dots, N\}$

En tal situación, sería interesante extraer primero los monomios (o productos de las posiciones específicas dentro de los vectores de características) y trabajar en un espacio de características Z en representación de todos los monomios de grado d . Tomando como ejemplo $x \in \mathbb{R}^2$ se pueden seleccionar los monomios de grado 2 a través del siguiente mapeo no lineal

$$\begin{aligned} \phi: \mathbb{R}^2 &\rightarrow Z = \mathbb{R}^3 && 4.48 \\ (x_1, x_2) &\rightarrow (x_1^2, x_2^2, x_1x_2) && 4.49 \end{aligned}$$

En ciertas situaciones hay una manera de computar los productos internos sin realizar un mapeo a través de los *kernels no lineales*.

Para computar productos internos de la forma $\langle \phi(x), \phi(y) \rangle$ se propone la siguiente representación para un *kernel*.

$$K(x, y) = \langle \phi(x), \phi(y) \rangle \tag{4.50}$$

Supongamos que $N = d = 2$ por lo que los monomios deben ser de grado $d=2$:

$$C_2(x): \langle x_1x_2 \rangle \rightarrow (x_1^2, x_2^2, x_1x_2, x_2x_1) \tag{4.51}$$

Supongamos ahora la misma transformación para y :

$$C_2(y): \langle y_1y_2 \rangle \rightarrow (y_1^2, y_2^2, y_1y_2, y_2y_1) \tag{4.52}$$

El producto interno entre ambos monomios nos da:

$$\langle C_2(x)C_2(y) \rangle \rightarrow x_1^2y_1^2 + x_2^2y_2^2 + 2x_1x_2y_1y_2 = (x \cdot y)^2 \quad 4.53$$

La distribución de los monomios brinda las condiciones para que el *kernel* K deseado se obtenga simplemente mediante el cuadrado del producto interno de los espacios de entrada.

Sea C_d una función que mapea $x \in \mathbb{R}^N$ para el vector $C_d(x)$. Las entradas son todos los posibles productos (monomios) ordenados de grado d de la variable x . Entonces, una función *kernel* que es equivalente a realizar el producto interno de los vectores mapeados por C_d y:

$$k(x, y) = \langle C_d(x)C_d(y) \rangle = (x \cdot y)^d \quad 4.54$$

El objetivo de este ejemplo es el de demostrar que un producto interno puede ser calculado sin el mapeo tradicional por medio de las funciones *kernel*.

Las representaciones *kernel* trabajan como proyecciones del espacio donde los datos residen en un espacio de características de más alta dimensión para permitir la clasificación de los datos que, en su espacio característico original, son no linealmente separables. En la Figura 4.5 se puede ver cómo un conjunto de datos *no linealmente separables* se transforma en un conjunto *linealmente separable* en un nuevo espacio característico, de mayor dimensión, creado por la función *kernel* ϕ .

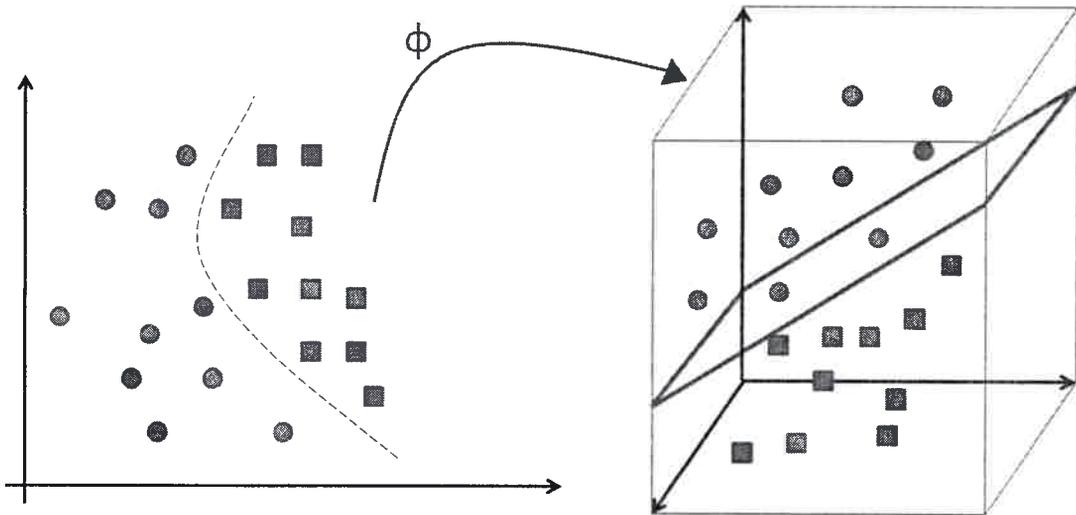


Figura 4.5 - Transformación del espacio de características usando funciones Kernel

4.4.3.1 Funciones kernel

Seleccionar la correcta función *kernel* es de vital importancia para una buena configuración de un SVM. Tal como se mencionó en el *Capítulo 3* de este trabajo de tesis, una de las características más importantes para la configuración de una red neuronal radica en la selección de su arquitectura. De manera análoga, para el caso de

los SVM, la selección de la arquitectura es equivalente a seleccionar una apropiada función *kernel*. [49]

Las condiciones que una función *kernel* debe cumplir son:

- Debe ser simétrica $K(x, z) = \langle \phi(x), \phi(z) \rangle = K(z, x)$
- Debe cumplir con la desigualdad de Cauchy-Schwarz

$$K(x, z)^2 = (\langle \phi(x), \phi(z) \rangle)^2 \leq \|\phi(x)\|^2 \|\phi(z)\|^2$$

$$= \langle \phi(x), \phi(x) \rangle \langle \phi(z), \phi(z) \rangle = K(x, x)K(z, z)$$

- Debe cumplir con el teorema de Mercer: La condición de Mercer garantiza que una función simétrica $K(x, z) = \sum_i \phi(x)_i \phi(z)_i$ es un kernel se cumple la siguiente desigualdad:

$$\iint K(x, z)g(x)g(z)\delta x\delta z \geq 0$$

para

$$\int g^2(x)\delta x \leq \infty$$

Las funciones Kernel más comúnmente empleadas son:

- Máquinas de aprendizaje polinomial: Una máquina de aprendizaje polinomial de grado d está especificada por:

$$k(x, x_i) = [\langle x x_i \rangle + 1]^d \quad 4.55$$

- Máquinas de funciones de base radial: Es producida por una red del tipo RBF y está dada por:

$$k(x, x_i) = e^{-\frac{\|x-x_i\|^2}{2\sigma^2}} \quad 4.56$$

- Redes neuronales de dos capas: se puede definir

$$k(x, x_i) = S[v(x, x_i) + c] \quad 4.57$$

donde v y c son parámetros escalares y $S(u)$ es la función sigmoidea.

Cada función kernel debe ser estudiada teniendo en cuenta el problema en particular.

4.4.3.2 Incorporación del kernel al SVM

Con la introducción de la función Kernel el problema de SVM no se ve alterado en gran medida. Para encontrar los α_i es preciso resolver el siguiente problema:

$$\text{maximizar } \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l y_i y_j \alpha_i \alpha_j K(x_i x_j) \quad 4.58$$

Sujeto a:

$$\sum_{i=1}^l y_i \alpha_i = 0 \quad 4.59$$

$$0 \leq \alpha_i \leq C \quad i = 1, \dots, l$$

Es importante observar que la forma de la ecuación es idéntica a la que se utilizó para encontrar el hiperplano de margen máximo excepto que el producto interno entre $x_i x_j$ ahora es realizado a través de la función kernel $K(x_i x_j)$.

4.4.4 Máquinas de Soporte de Vectores Multiclase

En este capítulo, primero se abordó el problema de la clasificación entre dos clases de datos linealmente separables. Debido a que este tipo de problemas rara vez se presenta en la vida real, se agregó la teoría de los márgenes relajados (*softmargin*) y las funciones *kernel*, para la clasificación de patrones compuestos por dos clases que no cumplen con la condición de ser linealmente separables.

En esta sección se presentará una estrategia para hacer extensivo el uso de los SVM cuando el problema involucre más de dos clases de datos [50].

4.4.4.1 Uno contra el resto (One versus rest)

En este método N diferentes clasificadores son construidos, uno para cada clase. El clasificador l -th se entrena sobre todo el conjunto de datos para aprender a distinguir la clase l contra las restantes. Los miembros de una clase son rotulados con un $(+1)$ y los restantes datos son rotulados con (-1) .

4.4.4.2 Uno contra uno (One vs One)

Para cada una de las clases posibles un clasificador binario es entrenado. Cada clasificador es entrenado utilizando parte del conjunto de entrenamiento que contiene algunos ejemplos de las dos clases involucradas. En este caso una cantidad de $N(N-1)/2$ clasificadores son construidos y entrenados. En este caso se necesita un esquema de votación para verificar cuál de los datos se aproximó a los datos de prueba para calcular la clasificación final.

4.4.4.3 Árbol de jerarquía de clasificadores

Mediante este método el problema de clasificación es descompuesto en una serie de sub-problemas de clasificación binaria organizados en un esquema jerárquico. [51]

La Figura 4.6 muestra un clasificador que utiliza seis etapas de SVM en una estructura tipo árbol para la clasificación de siete diferentes clases de datos.

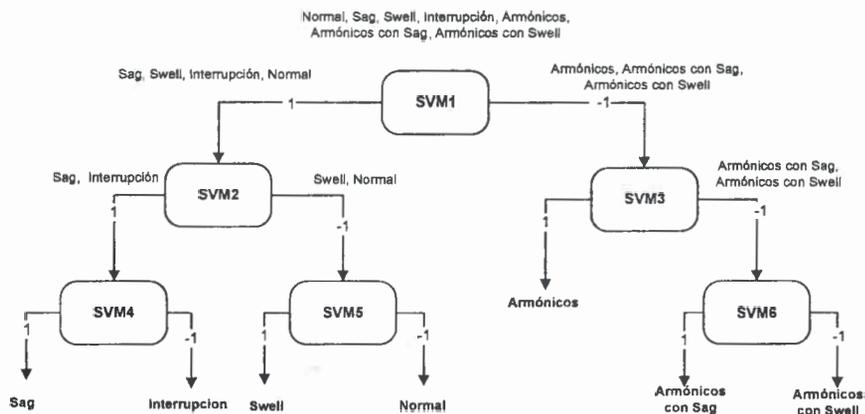


Figura 4.6 - Estructura jerárquica de tipo árbol para un clasificador SVM multiclase.

Este método soluciona el problema de la clasificación de N clases, sin embargo a medida que se desciende en el árbol jerárquico se contará con menos datos para realizar el entrenamiento del SVM.

En trabajo de Chih-Wei y Chih-Jen (2002) se presenta una comparación entre las diferentes estrategias para desarrollar un SVM multi clase [52].

4.5 ESTADO DEL ARTE: APLICACIÓN DE MAQUINAS DE SOPORTE VECTORIAL PARA LA CLASIFICACION DE PERTURBACIONES

En su trabajo Peisheng y Weilin (2006) presentan una comparación del rendimiento de un clasificador basado en una *Red Neuronal Artificial (ANN)* del tipo *backpropagation* y un segundo clasificador basado en *Máquinas de Vectores de Soporte (SVM)* para la clasificación de las perturbaciones en un SEP. En ambos casos se emplea la *Transformada Wavelet (WT)* y la *Transformada de Fourier dependiente del Tiempo (STFT)* para la extracción de patrones característicos de las señales perturbadas. La estrategia elegida por los autores para la confección del clasificador SVM multiclase es *uno contra uno*. Este trabajo concluye en que el clasificador basado en SVM es más rápido, el set de entrenamiento necesario para entrenarlo es más reducido y presenta un mayor porcentaje de aciertos y un mejor comportamiento frente a las muestras corruptas con ruido [53].

La publicación de Janik and T. Lobos (2006) presenta un método para la clasificación de perturbaciones basado en *SVM* configurado con una función *kernel* del tipo *Radial Basis Function (RBF por sus siglas en inglés)*. Para establecer comparaciones los autores proponen un segundo clasificador que utiliza *ANN* del tipo *RBF*. Para la extracción de parámetros característicos de un sistema trifásico se utiliza un método basado en *fases* complejos. En el trabajo los autores concluyen que el clasificador *SVM* presenta un mejor rendimiento que el clasificador que utiliza una *RBF ANN* [54].

Weiming, Xuelei, Jingbo y Zhiheng (2006), por su parte, publicaron un artículo donde utilizan los *Paquetes de Transformadas Wavelet (WP por sus siglas en inglés)* para la extracción de parámetros característicos de una señal perturbada y un sistema de clasificación que emplea una arquitectura del tipo *árbol de decisión binaria basado en SVM* para la clasificación de los eventos de calidad de energía. Los resultados obtenidos por los autores muestran que el clasificador propuesto emplea menos tiempo de entrenamiento, menos tiempo de prueba, más confiabilidad y más exactitud en la clasificación en comparación con un clasificador basado en una *ANN* [30].

Los sistemas de clasificación que basan su funcionamiento en métodos de inteligencia artificial (*AI por sus siglas en inglés*) deben ser entrenados antes de ser utilizados en un escenario real. Por este motivo, para aplicaciones comerciales, es esperable que el usuario no tenga que entrenar el sistema clasificador antes de utilizarlo. En su lugar dicho clasificador debería encontrarse configurado y entrenado por el fabricante. Teniendo en cuenta esta problemática los autores Axelberg, Gu y Bollen (2007) publicaron un trabajo donde se simula el comportamiento de un clasificador basado en *SVM* en situaciones reales cuando el algoritmo es entrenado con datos reales y datos simulados. La estrategia seleccionada en esta publicación para configurar un clasificador multiclase fue *uno contra todos* en una arquitectura tipo árbol de decisión binaria [55].

La arquitectura utilizada para los clasificadores multiclase basados en *SVM* usualmente utiliza un esquema *uno contra uno* o *uno contra el resto*. El tamaño de estas redes crece

marcadamente a medida que se agregan nuevas clases a los datos a clasificar. Por este motivo, Whei-Min, Chien-Hsien, Chia-Hung y Fu-Sheng (2008) presentaron un sistema de procesamiento que utiliza la *WT* para la extracción de parámetros característicos de una señal y una estrategia denominada *disturbance vs normal (DVN por sus siglas en inglés)* para la clasificación de las perturbaciones. En este esquema un SVM por cada tipo de perturbación es entrenado para distinguir un evento de calidad de energía de una señal limpia de perturbaciones. El sistema de procesamiento propuesto por los autores permite, además de la detección y clasificación, la localización de un evento de calidad de energía en el SEP [19].

El trabajo de Milchevski y Taskovski (2011) presenta un método que utiliza la energía de los coeficientes de la *WT* para la extracción de parámetros característicos de una señal perturbada. Adicionalmente, para facilitar la tarea del sistema de clasificación, los autores proponen un método de umbralización de la amplitud de la señal cuya finalidad es la extracción de tres parámetros característicos adicionales. El sistema de clasificación propuesto por los autores se basa en un árbol de decisión binaria donde un modelo *SVM* es creado para cada nodo [56].

Un reciente trabajo de Erisi y Demir (2012) presenta un completo sistema de procesamiento de perturbaciones basado en la *Transformada Wavelet (WT por sus siglas en inglés)* y las *SVM* para la detección, localización y clasificación de las perturbaciones en un sistema eléctrico de potencia. El sistema de procesamiento fue desarrollado en MATLAB, utilizando una librería llamada LibSVM la cual utiliza el método *uno contra todos* para la clasificación de patrones multi clase. La función *kernel* elegida por los autores fue *Radial Basis Function (RBF)* y se utilizó un proceso de búsqueda basado en grillas para encontrar los parámetros de configuración óptimos de SVM [57].

En su trabajo, Biswall, Dash y Mishra presentan un esquema conformado por una transformada del dominio del tiempo al dominio del tiempo (Transformada T-T) para la clasificación de patrones de perturbación no estacionarios en señales provenientes de sistemas eléctricos de potencia. La Transformada T-T no es ni más ni menos que una variante de la transformada S y utiliza una nueva función para el ventaneo que es proporcional a la frecuencia de la señal a analizar. Los parámetros característicos derivados de la transformada se utilizan para el reconocimiento automático empleando SVM basado en clustering. El trabajo también presenta un algoritmo de optimización inmune para mejorar el rendimiento del sistema Transformada T-T – SVM [58].

4.6 CONCLUSIONES

A pesar de resultar una herramienta de notoria complejidad matemática, la literatura indica que las Máquinas de Soporte Vectorial presentan resultados equivalentes o superiores frente a otras técnicas de clasificación.

En la primera parte del capítulo se presentaron las herramientas necesarias para el entendimiento de la matemática involucrada en el entrenamiento de una Máquina de Vectores de Soporte. Con tal fin se presentaron algunos conceptos relacionados con la *Teoría de Aprendizaje* y la *Optimización Matemática*.

En la segunda parte del capítulo se presentaron los conceptos teóricos y matemáticos involucrados en el uso y entrenamiento de un clasificador basado en Máquinas de Vectores de Soporte. En esta sección también fueron abordadas varias estrategias para transformar un clasificador SVM binario en un clasificador multiclase.

Finalmente fueron presentadas las publicaciones más relevantes donde se utilizan las Máquinas de Soporte de Vectores para conformar un sistema de clasificación multiclase de eventos de calidad de energía.

Segunda Parte:
Desarrollo de la
herramienta

Capítulo 5: Software para el Análisis de Eventos de Calidad de Energía

5.1 INTRODUCCIÓN

Los eventos y actores involucrados en la generación de perturbaciones de diversa naturaleza son cada día más comunes en un Sistema Eléctrico de Potencia (SEP). Paradójicamente, tanto para clientes industriales como en los consumidores minoristas, se ha incrementado drásticamente la utilización de dispositivos sensibles a los eventos de calidad de energía. Como consecuencia de esta problemática, muchos esfuerzos apuntaron a la detección, clasificación y caracterización de perturbaciones en los SEP.

Un aspecto crítico para la disminución del impacto de las perturbaciones en un sistema eléctrico radica en la habilidad de la entidad que realiza el monitoreo para detectar, localizar y clasificar una perturbación de manera automática.

Se entiende por sistema de monitoreo de calidad de potencia a una combinación de hardware y software específicamente diseñados para detectar y registrar eventos provenientes de mediciones de la red eléctrica. En el *Capítulo 1* de este trabajo de tesis se detallan los aspectos fundamentales relacionados a un sistema de monitoreo.

Dentro de las características de hardware más importantes se destacan los rangos de corriente y energía en los que el dispositivo funciona de manera adecuada, la cantidad de canales de medición, el modo de extracción de los datos, duración de la batería, la frecuencia máxima de muestreo soportada, la frecuencia máxima de entrada que el equipo permite analizar, el peso del dispositivo y la portabilidad del mismo.

El software presenta diferentes características dentro de las cuales se pueden destacar funciones que permiten la configuración de la frecuencia de muestreo y la frecuencia fundamental de la señal bajo análisis, los tipos de eventos de calidad de energía que pueden ser detectados, clasificados y registrados, la precisión con que los eventos son localizados en el tiempo, la frecuencia armónica máxima que puede ser detectada, la forma en que los resultados son presentados en pantalla y exportados, entre otras.

En el presente capítulo se describen los algoritmos que han sido desarrollados con el objetivo de detectar y clasificar eventos de calidad de energía provenientes de una señal muestreada.

La herramienta, desarrollada en MATLAB, cuenta con una interfaz gráfica que permite el acceso de manera intuitiva a todas sus funcionalidades.

El software se ha desarrollado con dos objetivos:

- Contar con una herramienta de monitoreo de eventos de calidad de energía en tiempo real.
- Contar con una herramienta académica para estudiar distintas técnicas y algoritmos de análisis de eventos de calidad de energía en un sistema eléctrico de potencia.

Como más adelante se detallará en este capítulo el software permite el análisis de perturbaciones obtenidas mediante el muestreo de señales reales así como también de señales generadas de manera artificial. A su vez, la herramienta cuenta con un módulo para la generación de señales perturbadas, permitiendo la configuración de las características más importantes que gobiernan a cada tipo de evento como, por ejemplo, el tiempo de duración, la magnitud o la frecuencia del evento.

Este capítulo también presenta el desarrollo de un algoritmo basado en la *Transformada Wavelet* (ver *Capítulo 2*) el cual permite detectar la presencia de un evento calidad de energía y extraer distintas características de dicha señal que son de utilidad para el proceso de clasificación.

Finalmente en este capítulo se exhibe el desarrollo de un módulo que realiza la clasificación de los eventos de calidad de energía de acuerdo a las características extraídas en el módulo de detección. Los algoritmos de clasificación están basados en *Redes Neuronales Artificiales* (ver *Capítulo 3*) y *Máquinas de Vectores de Soporte* (ver *Capítulo 4*).

5.2 SOFTWARE PARA EL ANALISIS DE EVENTOS DE CALIDAD DE ENERGIA

El software fue desarrollado para permitir el análisis de distintos métodos de detección y clasificación de perturbaciones en un sistema eléctrico de potencia. A su vez el diseño de la herramienta involucró procedimientos para simular el comportamiento de los algoritmos para situaciones reales.

Como ya se mencionó en el *Capítulo 1* un sistema de procesamiento de señales perturbadas está conformado principalmente por tres bloques funcionales tal como se muestra en la Figura 5.1.

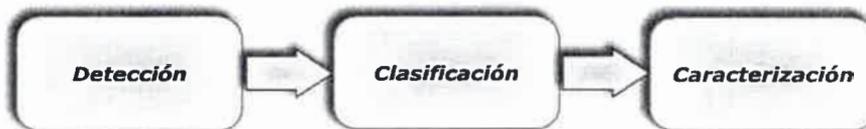


Figura 5.1 - Sistema de procesamiento de perturbaciones

El módulo de detección fue diseñado utilizando algoritmos que se basan en técnicas de procesamiento de señales para la detección de perturbaciones, la extracción de parámetros característicos y la localización de eventos de calidad de energía. Por lo mencionado en el *Capítulo 2* de esta tesis, la técnica de procesamiento de señales elegida para el desarrollo del módulo es la *Transformada Wavelet*.

El módulo de clasificación utiliza algoritmos basados en *Redes Neuronales Artificiales* y *Máquinas de Soporte Vectorial* cuya característica principal es que se deben entrenar en el reconocimiento y clasificación de patrones antes de ser empleadas en un sistema de monitoreo en tiempo real. Por este motivo, la herramienta fue desarrollada para permitir:

- El entrenamiento del módulo clasificador.
- El análisis del comportamiento de los algoritmos ante situaciones reales.

En las siguientes secciones del capítulo se explican los algoritmos utilizados para llevar a cabo estas dos funciones.

5.3 ENTRENAMIENTO DEL MODULO CLASIFICADOR

Antes de poder ser utilizado en un sistema de monitoreo en tiempo real, el módulo de clasificación debe ser previamente entrenado.

La Figura 5.2 muestra las actividades involucradas en el entrenamiento del módulo de clasificación.

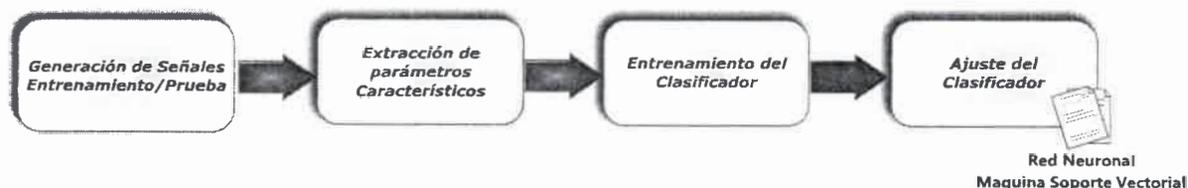


Figura 5.2 - Proceso de entrenamiento de un clasificador

5.3.1 Generación de Señales de Entrenamiento y Prueba

Para poder realizar el entrenamiento de los algoritmos del sistema clasificador es preciso contar con un número grande y variado de señales de prueba. Debido a que es complejo contar con un número grande de señales reales de variada naturaleza, se desarrolló una herramienta para la generación de perturbaciones de manera artificial. Mediante esta herramienta se generan dos vectores, el primero contiene una secuencia de perturbaciones y el segundo contiene una etiqueta que vincula a cada perturbación del primer vector con el tipo de evento de calidad de energía correspondiente.

5.3.2 Extracción de parámetros característicos

El objetivo de este bloque es el de evitar que el algoritmo del clasificador analice todas las muestras de cada perturbación para realizar la clasificación. Para ello se desarrolló un proceso que reduce el volumen de información empleado para el entrenamiento del algoritmo clasificador. Los parámetros extraídos de cada perturbación deben caracterizar de forma completa al evento.

La Figura 5.3 muestra el diagrama de flujo del software desarrollado para la detección y la extracción de parámetros característicos de un evento de calidad de energía. Las siguientes sub secciones detallan los aspectos más importantes de cada uno de los componentes del proceso de extracción de parámetros de una perturbación.

5.3.2.1 Adquisición de la señal

Este proceso se encarga de incorporar las señales de entrenamiento generadas de acuerdo a lo explicado en la sección 5.3.1.

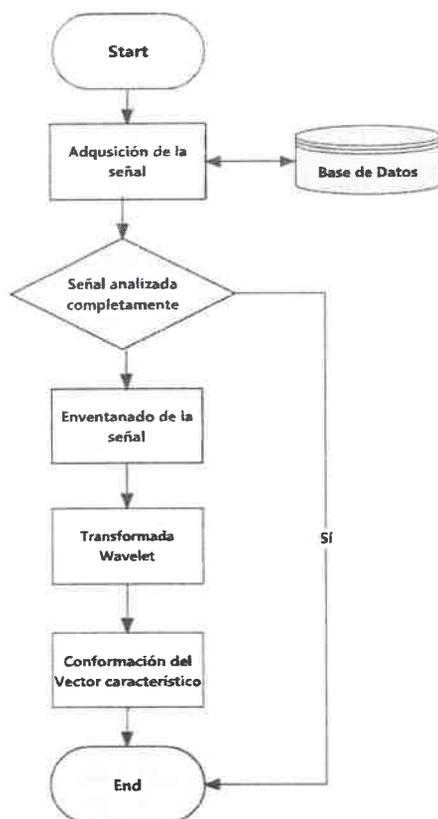


Figura 5.3 - Algoritmo de detección y caracterización de eventos de calidad de energía

5.3.2.2 Enventanado de la señal

Las señales empleadas para el entrenamiento y prueba de los algoritmos de detección y clasificación pueden estar conformadas por un conjunto de perturbaciones localizadas en distintos instantes de tiempo. La Figura 5.4 muestra una señal que contiene tres tipos de perturbaciones: *Hueco*, *Armónicos* y *Flicker*. Debido a que el análisis de cada perturbación debe realizarse de manera independiente es preciso procesar la señal tramo a tramo. El procesamiento tramo a tramo se realiza mediante la técnica de ventaneo que consiste en tomar un tramo de la señal a analizar y descartar el resto. Para el ejemplo propuesto, la longitud de la ventana comprende veinte ciclos de la frecuencia fundamental y el análisis se completa cuando se finalice el procesamiento de 3 secciones de la señal cuya duración será de 20 ciclos.

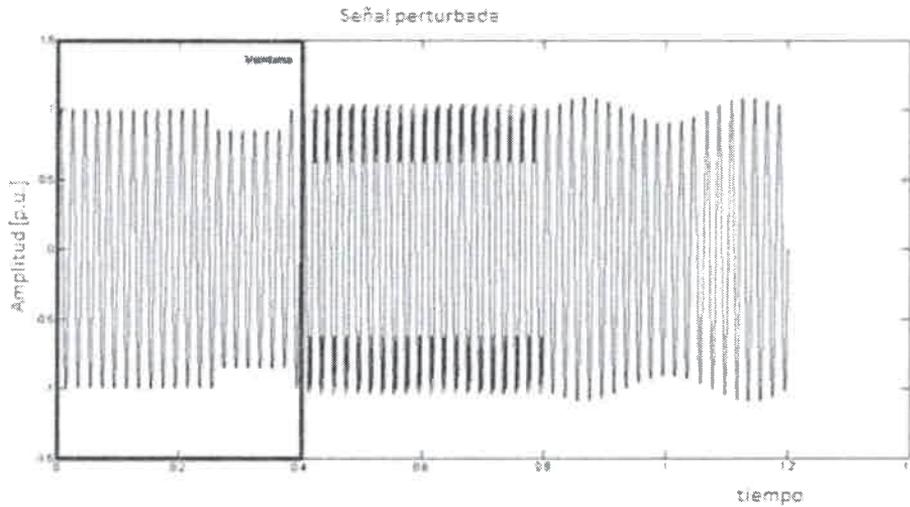


Figura 5.4 - Ventaneo de una señal.

5.3.2.3 Transformada Wavelet

Este proceso evalúa la *Transformada Wavelet* utilizando por defecto la wavelet madre Daubechie 4. Se utilizan además nueve niveles o coeficientes de detalle [53]. La Figura 5.5 muestra el resultado de la *Transformada Wavelet* correspondiente al primer tramo de la señal ventaneada.

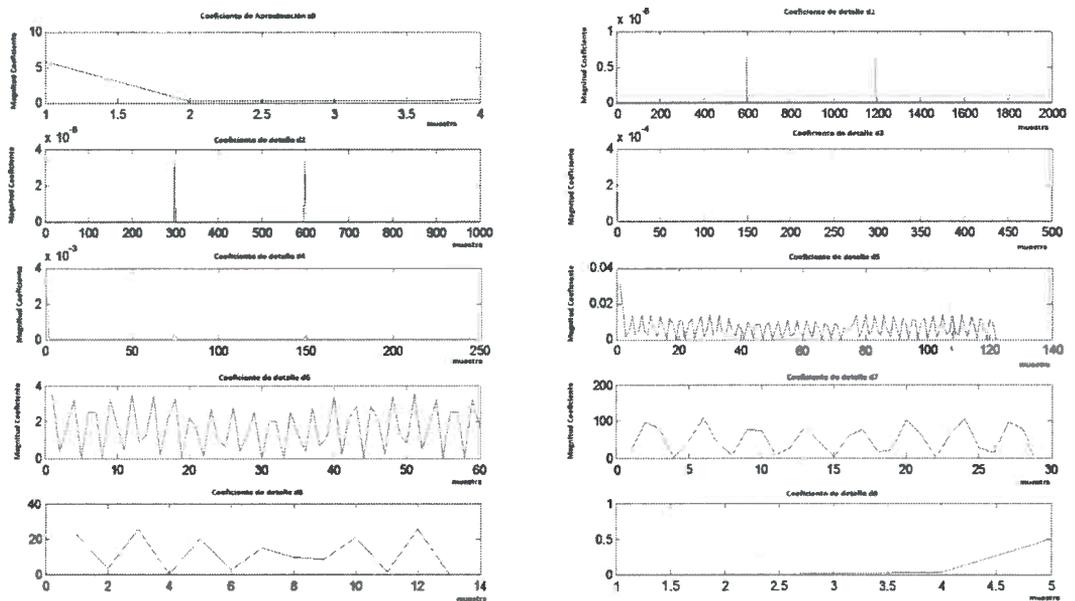


Figura 5.5 - Transformada Wavelet de 9 niveles.

5.3.2.4 Conformación del vector característico

El algoritmo desarrollado emplea los vectores de la *Transformada Wavelet* para extraer los primeros diez componentes característicos de la señal. Cada uno de estos componentes se corresponden con la diferencia normalizada de las energías de cada nivel de la *Transformada Wavelet* de la señal perturbada y la señal sin perturbación. La energía de cada nivel se calcula mediante el teorema de Parseval presentado en la siguiente ecuación 2.14.

$$\begin{aligned}
 Ed_{9Norm} &= \frac{(Ed_9 \text{ sen} - Ed_{9perturbada})}{Esen} \\
 Ed_{8Norm} &= \frac{(Ed_8 \text{ sen} - Ed_{8perturbada})}{Esen} \\
 Ed_{7Norm} &= \frac{(Ed_7 \text{ sen} - Ed_{7perturbada})}{Esen} \\
 Ed_{6Norm} &= \frac{(Ed_6 \text{ sen} - Ed_{6perturbada})}{Esen} \\
 Ed_{5Norm} &= \frac{(Ed_5 \text{ sen} - Ed_{5perturbada})}{Esen} \\
 Ed_{4Norm} &= \frac{(Ed_4 \text{ sen} - Ed_{4perturbada})}{Esen} \\
 Ed_{3Norm} &= \frac{(Ed_3 \text{ sen} - Ed_{3perturbada})}{Esen} \\
 Ed_{2Norm} &= \frac{(Ed_2 \text{ sen} - Ed_{2perturbada})}{Esen} \\
 Ed_{1Norm} &= \frac{(Ed_1 \text{ sen} - Ed_{1perturbada})}{Esen} \\
 Ea_{0Norm} &= \frac{(Ea_0 \text{ sen} - Ea_{0perturbada})}{Esen}
 \end{aligned} \tag{5.34}$$

Dónde:

Ed_iNorm es la energía normalizada correspondiente al d_i coeficiente de la Transformada Wavelet.

Ed_iSen es la energía correspondiente al d_i coeficiente de la Transformada Wavelet de una señal sin perturbar.

$Ed_iperturbada$ es la energía correspondiente al d_i coeficiente de la Transformada Wavelet de una señal perturbada.

$Esen$ es la energía total de la señal sin perturbar.

Los valores normalizados de energía probaron no ser suficientes para caracterizar completamente todas las perturbaciones que se analizan en este trabajo. Por este motivo este trabajo propone la incorporación al vector característico de 4 parámetros adicionales los cuales serán llamados k_1 , k_2 , k_3 y k_4 . Las ecuaciones 5.35, (5.36 y 5.37 muestran que los valores k_1, k_2 y k_3 están relacionados con la distribución de la amplitud de las muestras, mientras que la ecuación 5.38 muestra que k_4 es un parámetro que viene dado por la cantidad de muestras del coeficiente $d1$ de la transformada wavelet que superan un umbral predefinido.

$$k_{1\text{ Norm}} = \frac{\sum_{i=1}^N k_1(i)}{N} \quad k_1 = \begin{cases} 1 & \text{Si muestra}(i) \geq 1 \\ 0 & \text{Si muestra}(i) \leq 1 \end{cases} \quad (5.35)$$

$$k_{2\text{ Norm}} = \frac{\sum_{i=1}^N k_2(i)}{N} \quad k_2 = \begin{cases} 1 & \text{Si muestra}(i) \leq 0.1 \\ 0 & \text{Si muestra}(i) \geq 1 \end{cases} \quad (5.36)$$

$$k_{3\text{ Norm}} = \frac{\sum_{i=1}^N k_3(i)}{N} \quad k_3 = \begin{cases} 1 & \text{Si muestra}(i) \leq 1 \\ 0 & \text{Si muestra}(i) \geq 0.1 \end{cases} \quad (5.37)$$

$$k_{4\text{ Norm}} = \frac{\sum_{i=1}^N d_1(i)}{M} \quad k_4 = \begin{cases} 1 & \text{Si } d_1(i) \geq \text{Umbral} \\ 0 & \text{Si } d_1(i) \leq 0.1 \end{cases} \quad (5.38)$$

La ecuación 5.39 muestra que el vector característico queda conformado por:

$$\text{Vector Característico} = \begin{bmatrix} Ed_{9\text{ Norm}} \\ Ed_{8\text{ Norm}} \\ Ed_{7\text{ Norm}} \\ Ed_{6\text{ Norm}} \\ Ed_{5\text{ Norm}} \\ Ed_{4\text{ Norm}} \\ Ed_{3\text{ Norm}} \\ Ed_{2\text{ Norm}} \\ Ed_{1\text{ Norm}} \\ Ea_{0\text{ Norm}} \\ k_{\text{ Norm}1} \\ k_{2\text{ Norm}} \\ k_{3\text{ Norm}} \\ k_{4\text{ Norm}} \end{bmatrix} \quad (5.39)$$

5.4 Entrenamiento del clasificador

El módulo del clasificador de perturbaciones está conformado por un conjunto de herramientas desarrolladas con el objetivo de permitir la configuración, entrenamiento y ajuste de los algoritmos de clasificación basados en *Redes Neuronales Artificiales* y *Máquinas de Soporte Vectorial*. Estos métodos se encuentran dentro de la familia de las máquinas de aprendizaje supervisado los cuales deben ser entrenados antes de poder ser empleadas. La Figura 5.6 muestra el diagrama de flujo empleado en el desarrollo de los algoritmos de entrenamiento del módulo de clasificación.

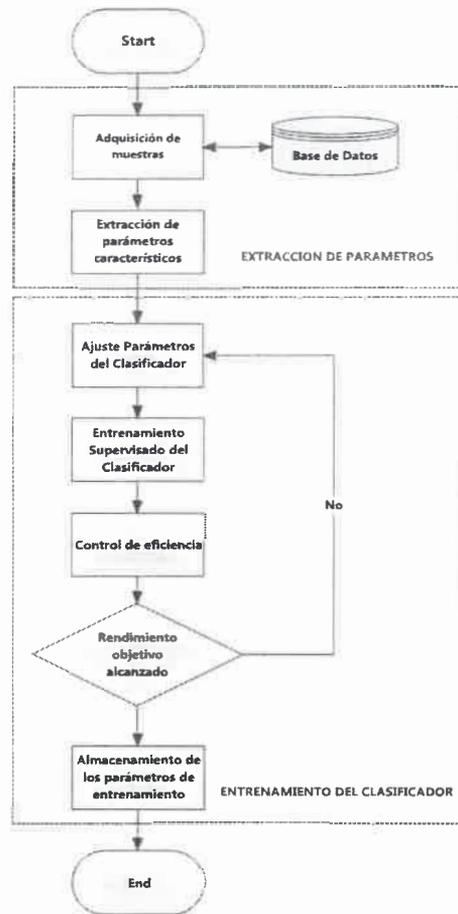


Figura 5.6 - Algoritmo de entrenamiento del clasificador

El módulo destinado a la extracción de parámetros característicos se corresponde al ya detallado en la sección 5.3.2. A continuación se detallan las funcionalidades correspondientes al entrenamiento del clasificador.

5.4.1 Ajuste de los parámetros del clasificador

El módulo clasificador cuenta con dos algoritmos que realizan esencialmente la misma tarea, los cuales están basados en:

- **Redes Neuronales Artificiales:** Como ya se ha explicado en el *Capítulo 3* existen varios parámetros que gobiernan el comportamiento de una ANN. Dentro de estos parámetros se encuentran, la arquitectura (cantidad de capas, cantidad de neuronas por capa), las funciones de activación de cada capa, la tasa de aprendizaje, la función de entrenamiento, los parámetros de salida de entrenamiento (cantidad de épocas y performance). El ajuste de los parámetros de una ANN backpropagation consiste en la búsqueda de los parámetros óptimos para resolver de la forma más eficiente el problema de clasificación de eventos de calidad de energía. El algoritmo desarrollado se basa en la función de MATLAB *feedforwardnet* para la creación de una red neuronal del tipo back-propagation.

- Máquinas de Soporte Vectorial: Una máquina de soporte vectorial es un algoritmo de aprendizaje supervisado que permite la clasificación entre dos clases de parámetros de entrada. Para conformar un clasificador multi clase basado en SVM es necesario conformar una estructura de más de un SVM como se explica con más detalle en el *Capítulo 4*. El software desarrollado emplea la estrategia denominada *One vs All*, mediante la cual se entrena una Máquina de Soporte Vectorial por cada tipo de perturbación. Por ejemplo, como puede verse en la *Figura 5.7*, la máquina de soporte vectorial llamada SVM 1 es entrenada para presentar un 1 a su salida si el vector característico de entrada se corresponde con un *Hueco* y un -1 si se trata de cualquier otro tipo de perturbación. Cada uno de los SVM fueron generados utilizando la función *svmtrain* perteneciente al grupo de herramientas de aprendizaje estadístico de la toolbox de bioinformática de MATLAB. La configuración de la arquitectura *One Vs All* se realiza mediante código desarrollado por el tesista.

Para mejorar el rendimiento del clasificador se pueden ajustar ciertos parámetros que gobiernan a cada uno de los nodos SVM. Debido a que el desarrollo presentado en esta tesis utiliza la función kernel conocida como *Radial Basis Function (RBF, por sus siglas en inglés)* los parámetros que se ajustan son *boxconstraint* y *Rbf_sigma*.

El parámetro *boxconstraint* se utiliza para ajustar el factor de penalidad C para el caso donde las clases no son separables. Este problema es conocido como un problema de decisión de margen blando (*soft margin*).

Rbf sigma, por otro lado, se corresponde con un factor de escala para ajustar la función Kernel RBF que es la que se emplea en cada uno de los nodos del clasificador.

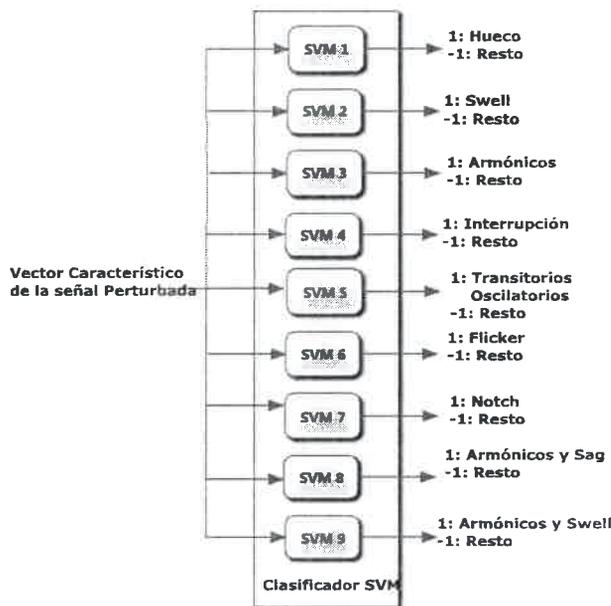


Figura 5.7 - Arquitectura de un clasificador basado en SVM

5.4.2 Entrenamiento supervisado del clasificador

El algoritmo del clasificador, al pertenecer a la familia de las máquinas lineales, debe ser entrenado antes de poder ser empleado en un sistema de monitoreo en tiempo real. Los datos de entrenamiento generalmente consisten de pares de objetos (normalmente vectores) donde una componente del par son los datos de entrada y el otro, los resultados deseados. La salida de la función puede ser un valor numérico (como en los problemas de regresión) o una etiqueta de clase (como en los de clasificación). El objetivo del aprendizaje supervisado es el de crear una función capaz de predecir el valor correspondiente a cualquier objeto de entrada válida después de haber analizado una serie de ejemplos (*).

Para el sistema de monitoreo estudiado en el presente capítulo los datos de entrenamiento se corresponden con un conjunto de pares conformados por vectores característicos y las correspondientes etiquetas que vinculan el vector con una clase o tipo de perturbación.

El algoritmo desarrollado para el entrenamiento del clasificador basado en la red neuronal se basa en la función de MATLAB *trainlm*. Esta función utiliza la técnica de optimización *Levenberg-Marquardt* para la actualización de los pesos de la red neuronal.

Por otro lado, las Máquinas de Vectores de Soporte se entrenan mediante la función *svmtrain* de MATLAB.

(*) Nota: Para más detalle de los procedimientos de entrenamiento de las redes Neuronales referirse al capítulo 3 de este trabajo de tesis. Para más información acerca de los procesos de entrenamiento para las Máquinas de Soporte Vectorial referirse al capítulo 4 del presente trabajo.

5.4.3 Control de la eficiencia

Esta rutina fue desarrollada para realizar el control de la eficiencia del clasificador una vez que ha sido entrenado. Para ello un conjunto conocido de vectores característicos son evaluados por el clasificador y se almacenan los resultados de la clasificación.

El algoritmo desarrollado emplea la función *sim* de MATLAB para obtener una clasificación del módulo basado en la red neuronal. Del mismo modo, la función *svmclassify* de MATLAB es utilizada para obtener los resultados del algoritmo basado en las Máquinas de Vectores de Soporte.

Los vectores característicos utilizados para el control de eficiencia y ajuste de parámetros se corresponden con un grupo de perturbaciones conocido. Por lo tanto, el algoritmo de control de eficiencia puede comparar los resultados obtenidos del proceso de clasificación con las etiquetas de las perturbaciones que previamente se conocían. El algoritmo de control de eficiencia desarrollado, por lo tanto, devuelve la cantidad de perturbaciones correcta e incorrectamente clasificadas.

En función de los resultados obtenidos en el proceso de control de eficiencia permite realizar el ajuste manual de los parámetros del algoritmo para volver a entrenar el clasificador y, de este modo, aproximar los resultados con una precisión deseada.

5.4.4 Almacenamiento de los parámetros de entrenamiento

Una vez que se obtuvo el rendimiento deseado, los parámetros del clasificador pueden ser almacenados en un archivo para ser empleados en el sistema de monitoreo en tiempo real.

5.5 ANALISIS DEL COMPORTAMIENTO DE LOS ALGORITMOS EN SITUACIONES REALES

La Figura 5.8 muestra un diagrama esquemático del software de monitoreo desarrollado para la prueba de los algoritmos de detección y clasificación presentados en la sección anterior. Las siguientes sub secciones explican cada etapa del proceso de prueba.

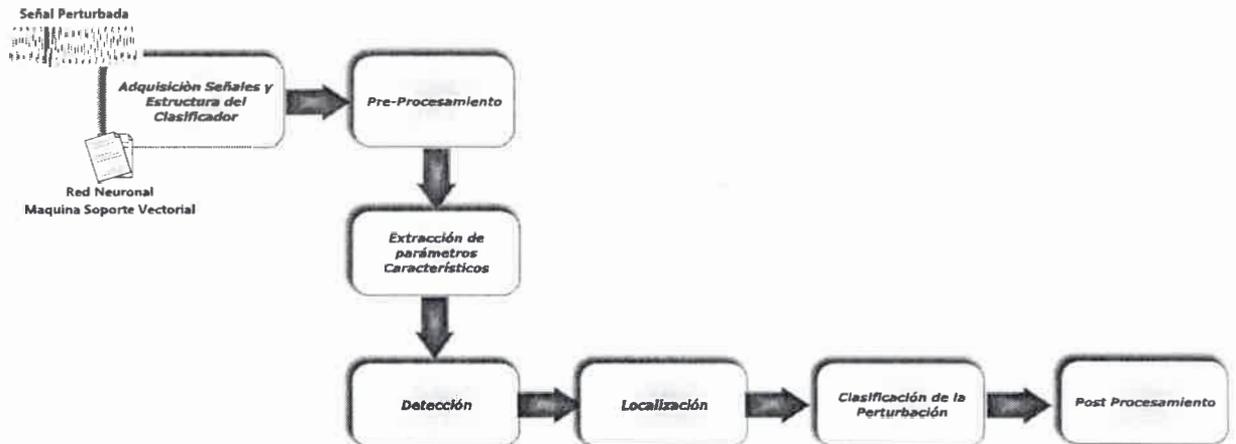


Figura 5.8 - Algoritmo de clasificación de señales reales

5.5.1 Adquisición de las señales y las estructuras del clasificador

Esta etapa es la encargada de incorporar la señal que se va a analizar. Aquí se debe configurar la ventana de análisis de tal manera que la misma se ajuste a un número entero de ciclos de la señal. Además de las señales, en esta fase se permite la incorporación de los archivos que contienen la configuración de los clasificadores entrenados de acuerdo a lo expuesto en la sección anterior. Finalmente, en esta etapa, se realiza la normalización de la amplitud de la señal utilizando los valores nominales de la red con la finalidad de operar con valores pu.

5.5.2 Pre-Procesamiento

Debido a que mediciones de las señales pueden verse corruptas por ruido, el cual usualmente es modelado como ruido blanco gaussiano aditivo, muchas veces, es necesario realizar un pre-procesamiento de la señal para eliminar cualquier componente espurio que dificulte el proceso de detección. El ruido blanco, al tener componentes de alta frecuencia, afecta principalmente el coeficiente $d1$ de la *Transformada Wavelet* lo que dificulta en gran medida la detección y localización de un evento de corta duración [33].

5.5.3 Extracción de los parámetros característicos

Este proceso, cuya entrada es el vector que contiene las muestras de la señal presuntamente contaminada con una perturbación, genera un arreglo cuyos catorce

componentes caracterizan completamente la señal bajo análisis. La forma en que este proceso genera el vector característico es idéntica a la descrita en la sección 5.3.2.

5.5.4 Detección

Este proceso tiene por objetivo la detección de alguna condición anormal en la señal analizada. En caso de no detectar alguna condición que difiera de la normalidad el proceso descarta la muestra y queda a la espera de un nuevo vector característico para analizar.

El análisis para la detección de una perturbación se realiza monitoreando los valores anormales del vector característico. A modo de ejemplo, la Figura 5.9 muestra el vector característico de una señal sinusoidal pura y una señal contaminada con armónicos. En tal caso, la presencia de armónicos se detecta monitoreando los componentes 5, 6 y 11 del vector característico. En caso que estos parámetros difieran de los valores nominales para una señal sinusoidal pura (*), el algoritmo detectará la presencia de una perturbación, almacenará el vector característico y continuará con el flujo para localizar y clasificar el evento. Este análisis puede hacerse extensivo para todas las perturbaciones analizadas en este capítulo.

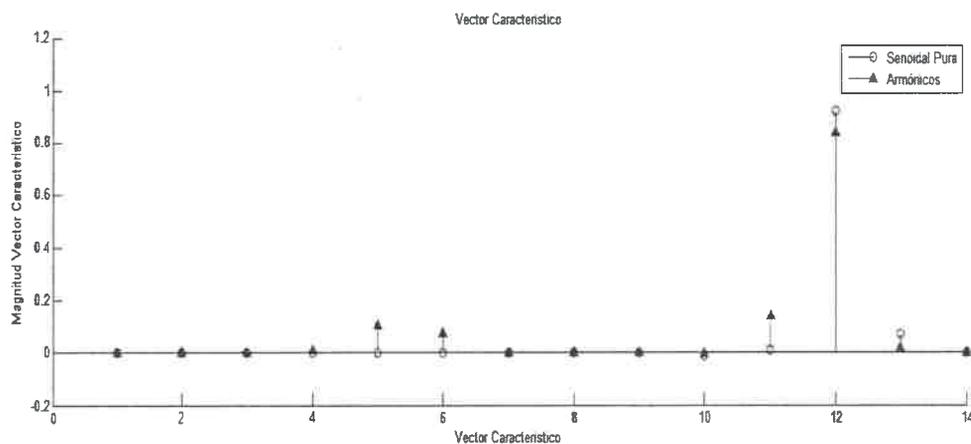


Figura 5.9 - Vector característico

(*)Nota: La magnitud de la diferencia de los componentes de una señal ideal y una señal contaminada por armónicos se encuentra estipulado en el estándar IEC 61000-2-2.

5.5.5 Localización temporal

Este proceso fue desarrollado con el objetivo de localizar una perturbación en el tiempo. Para ello, el software de monitoreo utiliza el coeficiente $d1$ de la *Transformada Wavelet* el cual, como se explicó en el *Capítulo 2*, se corresponde con el coeficiente de detalle que proporciona información de altas frecuencias. La Figura 5.10 muestra el coeficiente $d1$ de la señal presentada en la sección 5.3.2.3 del presente capítulo. En el gráfico puede verse que el coeficiente $d1$ presenta dos impulsos en las muestras correspondientes al inicio y al fin de la perturbación.

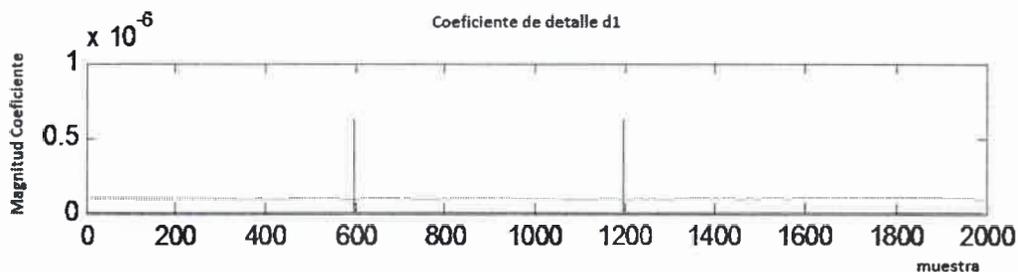


Figura 5.10 - Umbralización del coeficiente $d1$ de la TW

El proceso desarrollado compara muestra a muestra, de izquierda a derecha, el coeficiente de detalle $d1$ de la *Transformada Wavelet* con un umbral pre establecido. Cuando se detecta que una muestra supera dicho umbral, el proceso identifica el comienzo de una perturbación de corta duración. Luego, el algoritmo recorre el vector $d1$ de derecha a izquierda en búsqueda de un segundo valor de $d1$ que supere el umbral. En caso de encontrarlo el software identifica el fin de la perturbación. Finalmente, mediante la frecuencia de muestreo, el software calcula el tiempo de inicio de la perturbación y el tiempo de fin de la perturbación.

5.5.6 Clasificación de la perturbación

En caso de haber detectado la presencia de una perturbación y de haberla localizado en el tiempo (para aquellas perturbaciones de corta duración) se realiza el proceso de clasificación del fenómeno electromagnético.

La clasificación de una perturbación es un procedimiento que tiene por objetivo la vinculación del vector característico obtenido en la sección 5.5.3 con una etiqueta correspondiente a uno de los nueve tipos de perturbaciones para los cuales el clasificador será entrenado. Por lo tanto, el clasificador será entrenado para clasificar los nueve tipos de perturbaciones presentadas en la sección 1.2.7.

Para permitir la comparación del rendimiento de diferentes técnicas se desarrollaron dos algoritmos basados en:

- Redes Neuronales Artificiales.
- Máquinas de Soporte Vectorial.

5.5.7 Post-Procesamiento

Las respuestas del algoritmo de clasificación muchas veces necesitan ser procesadas como paso previo a la presentación en pantalla. Dos técnicas de post-procesamiento fueron empleadas en el software desarrollado.

5.5.7.1 Ganador toma todo

Las Redes Neuronales emplean una técnica de post-procesamiento denominada *Ganador toma todo* (en inglés, *Winner Take All*). Este principio computacional se aplica en modelos de redes neuronales cuya capa de salida está conformada por más de una neurona. El algoritmo de post-procesamiento genera que únicamente aquella neurona con el mayor nivel de activación obtenga un 1 lógico forzando a un 0 lógico a las restantes neuronas de la capa de salida.

5.5.7.2 Algoritmo basado en reglas

Debido a que las Máquinas de Soporte Vectorial son técnicas que permiten la clasificación solamente entre dos clases de objetos, para conformar un clasificador multi-clase se necesita la configuración y entrenamiento de más de una máquina de soporte vectorial tal como se explica en la sección 5.4.

La interpretación de los resultados de este algoritmo necesita un post procesamiento basado en reglas. Estas reglas son estructuras fijas de código que emplean diferentes parámetros para interpretar los resultados obtenidos en la etapa de clasificación.

5.6 SOFTWARE DE MONITOREO – CASO DE USO

El objetivo de esta sección es la descripción del proceso completo de entrenamiento y prueba de los algoritmos diseñados para detección y clasificación de eventos de calidad de energía ilustrando el procedimiento por medio de capturas de pantalla del software desarrollado.

El procedimiento comprende las siguientes etapas:

1. Acceso a la herramienta.
2. Generación de señales de entrenamiento y prueba.
3. Extracción de parámetros característicos de las señales de entrenamiento y prueba.
4. Entrenamiento y prueba del clasificador.
5. Incorporación de señales reales.
6. Simulación.

En el APENDICE A se describen todas las pantallas que conforman el software.

5.6.1 Acceso a la herramienta

Para acceder a la herramienta se debe ejecutar la aplicación MATLAB y configurar el directorio raíz donde se encuentran los scripts del software. En la ventana de comandos, como se muestra en la Figura 5.11, se invoca al panel principal de la herramienta por medio de la siguiente instrucción:

```
>> Analizador_de_señales
```

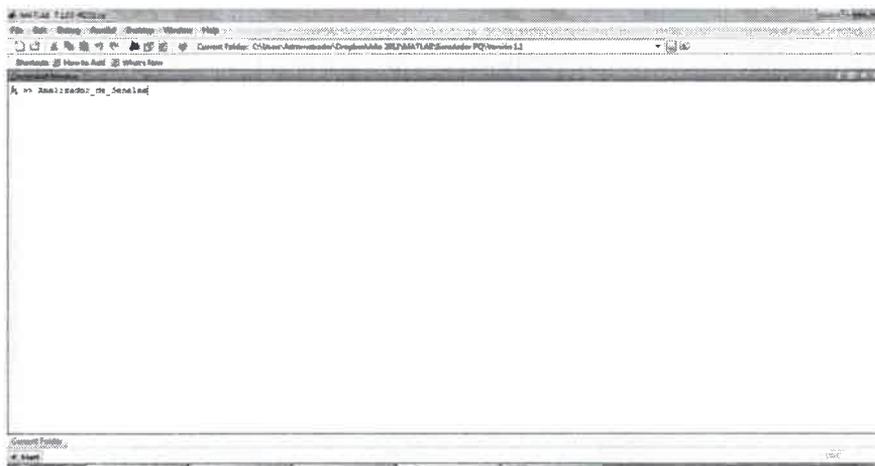


Figura 5.11 - Acceso a la herramienta

Al ejecutar este comando se despliega el panel principal de la aplicación mostrado en la Figura 5.12. Esta interfaz permite al usuario el acceso a los siguientes módulos cuyas funcionalidades serán explicadas en detalle en el presente capítulo:

- Módulo adquisidor de Señales
- Módulo generador de Señales
- Módulo de detección de Eventos
 - Transformada Wavelet
- Módulo de clasificación de Eventos
 - ANN BackPropagation
 - Support Vector Machine

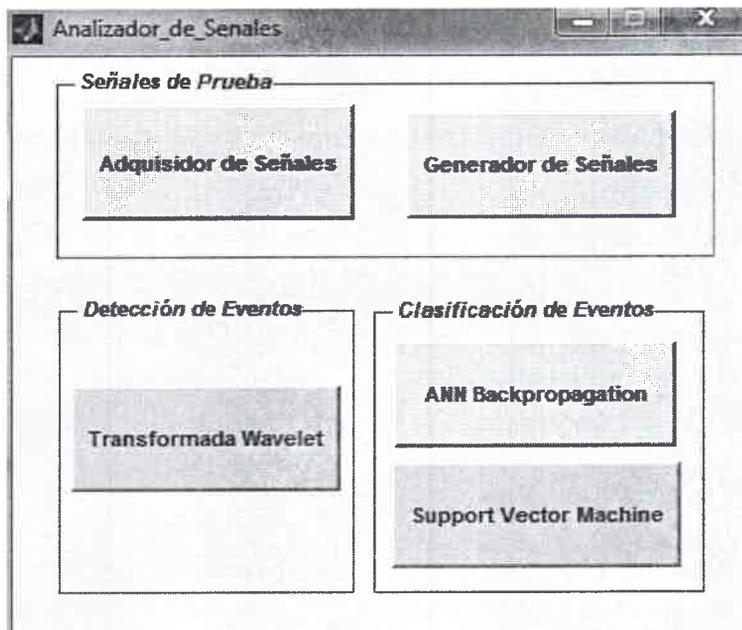


Figura 5.12 - Panel principal

5.6.2 Generación de señales de Entrenamiento y Prueba

Para generar las señales de entrenamiento y prueba se debe presionar la opción *Generador de Señales* del panel principal. Al realizar esta acción se despliega la interfaz gráfica que se muestra en la Figura 5.13.

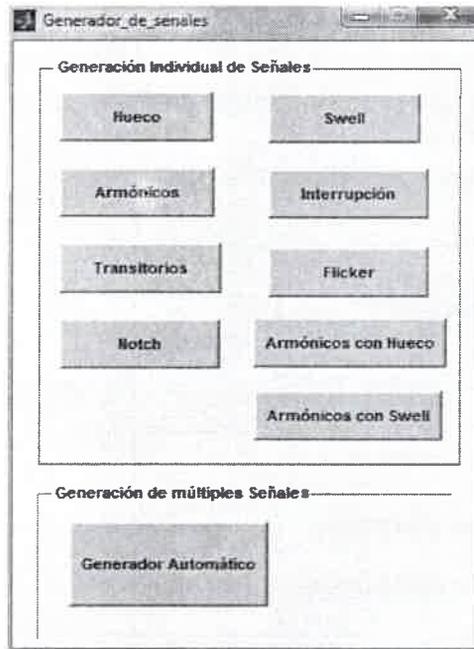


Figura 5.13 - Generador de señales de prueba

Presionando la opción *Generador Automático* se accede a la interfaz gráfica mostrada en la Figura 5.14 que permite la generación de señales de prueba y entrenamiento.

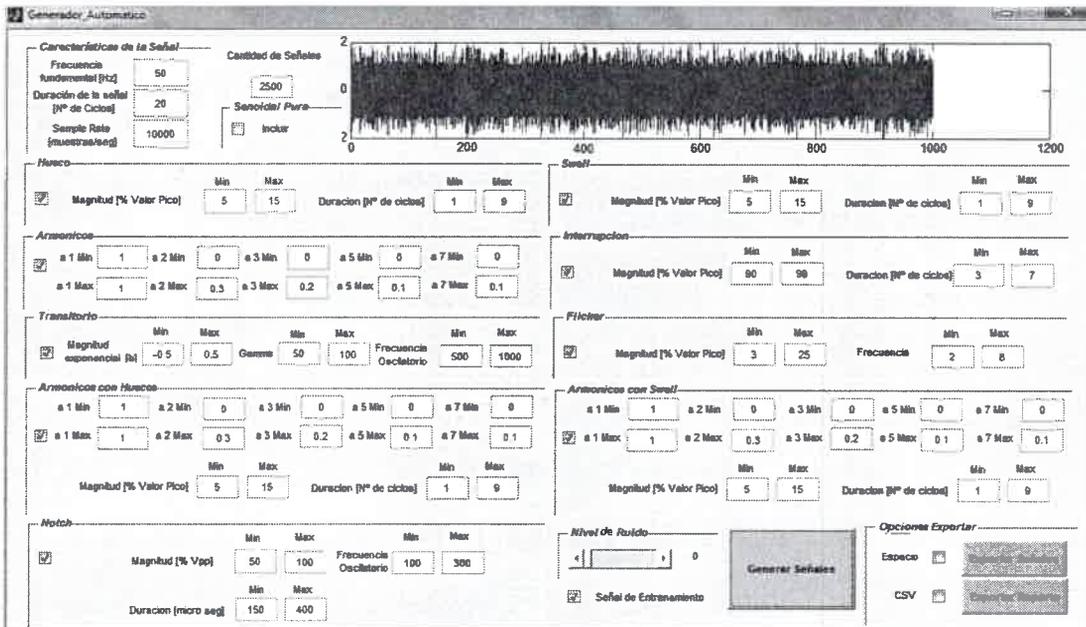


Figura 5.14 - Generador automático de señales perturbadas

Para nuestro ejemplo se crea un set de entrenamiento conformado por 2500 señales de los distintos tipos de perturbaciones que el software permite simular. La frecuencia fundamental seleccionada es de 50 [Hz], la longitud de la ventana se configura en 20 ciclos y la frecuencia de muestreo en 10000 [muestras/segundo].

Para generar la señal se configuran todos los campos correspondientes a las perturbaciones que se van incluir en la señal resultante, se seleccionan los correspondientes check box y luego se presiona el botón *Generar Señal*. Este botón genera un vector que contiene la cantidad de señales perturbadas de acuerdo a lo configurado en el campo *Cantidad de Señales* incluyendo solo aquellas perturbaciones que hayan sido debidamente seleccionadas. La herramienta fue desarrollada para que la generación de cualquier tipo de perturbación tenga la misma probabilidad. Como resultado se obtiene un vector que contiene las muestras de diferentes perturbaciones generadas de manera aleatoria, cuyas características principales se ven regidas de acuerdo a parámetros también generados aleatoriamente, aunque respetando los valores máximos y mínimos configurados en la interfaz gráfica.

Para generar la señal de entrenamiento se debe seleccionar el check box denominado *Señal de Entrenamiento*. Del mismo modo, para generar una señal del tipo señal de prueba se debe cuidar que el mencionado check box no se encuentre marcado.

5.6.3 Extracción de parámetros característicos de las señales de entrenamiento y prueba.

El acceso a la interfaz gráfica mostrada en la Figura 5.15 se realiza seleccionando la opción Transformada Wavelet del Panel Principal presentado en la sección 5.6.1.

Para generar el vector característico de entrenamiento se deben introducir los parámetros relativos al set de entrenamiento generado en la sección anterior. Para nuestro ejemplo se debe introducir la frecuencia fundamental ($50 [Hz]$), la longitud de la ventana ($20 [ciclos]$), la frecuencia de muestreo ($10000 [muestras/segundo]$) y, finalmente, la cantidad de señales (2500).

Al presionar el botón *Importar Perturbación*, la herramienta automáticamente incorpora la señal de Entrenamiento generada en la etapa anterior. Para indicar que los vectores característicos generados se utilizarán para entrenar los algoritmos de clasificación se debe marcar el check box denominado *Señal de Entrenamiento*.

Para generar un set de prueba se debe acceder al módulo *Generador de Señales*, generar un conjunto de señales cuidando de no marcar el check box *Señal de Entrenamiento* y repetir el proceso explicado en esta sección.

La Figura 5.15 muestra el panel desarrollado para la generación del vector característico.

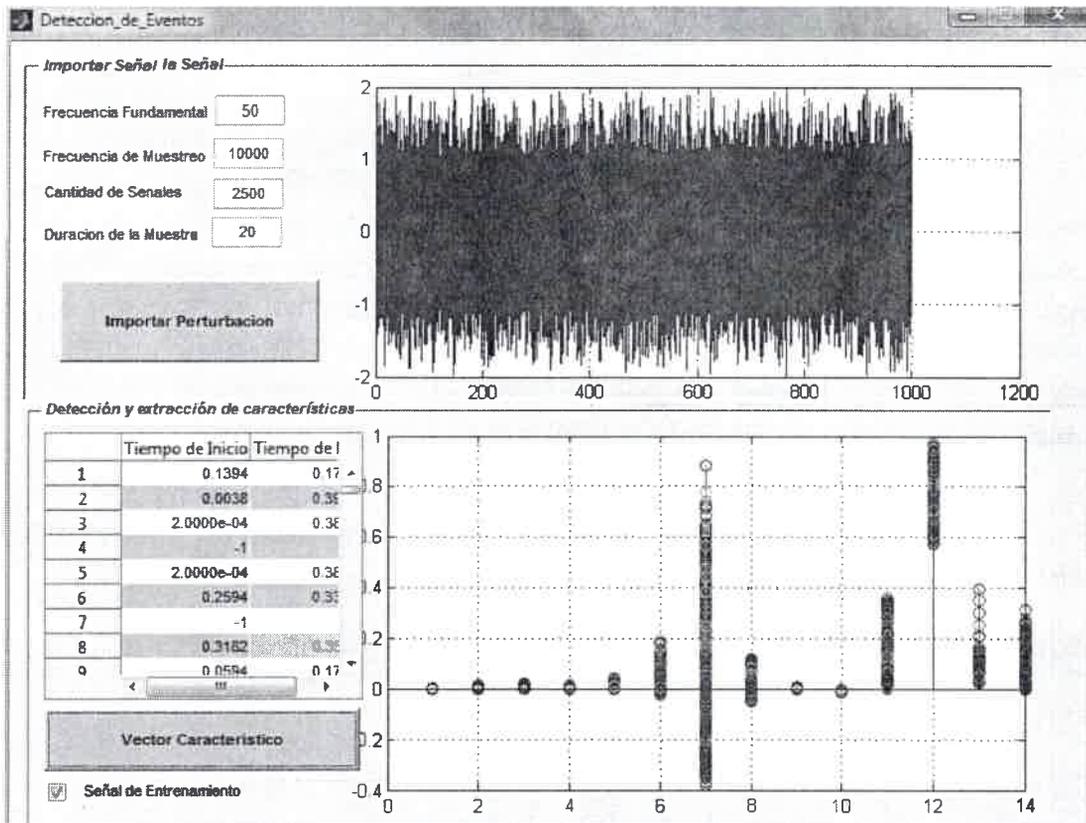


Figura 5.15 - Panel de generación del vector característico

Una vez que se generaron las señales y los vectores característicos de entrenamiento y prueba, el Workspace de MATLAB se encuentra conformado por las siguientes señales:

- SeñalGenerada: Vector que contiene la señal de prueba.
- SeñalGeneradaEntrenamiento: Vector que contiene la señal de entrenamiento.
- reporteSeñal: Vector que contiene las etiquetas correspondientes al vector de las señales de prueba.
- reporteSeñalEntrenamiento: Vector que contiene las etiquetas correspondientes al vector de las señales de entrenamiento.
- vectorCaracterístico: Matriz que contiene un vector característico de catorce componentes por cada una de las perturbaciones del vector de señales de prueba.
- vectorCaracterísticoEntrenamiento: Matriz que contiene un vector característico de 14 componentes por cada una de las perturbaciones del vector de señales de entrenamiento.

5.6.4 Entrenamiento y prueba del clasificador

Como ya se ha mencionado previamente en este capítulo el módulo de clasificación posee dos herramientas para la clasificación de eventos de calidad de energía: *Red Neuronal Backpropagation* y *Support Vector Machine*.

El acceso a cada uno de estos módulos se realiza a través del *Panel Principal* mencionado en la sección 5.6.1.

5.6.4.1 Red Neuronal Backpropagation

En la interfaz gráfica mostrada en la Figura 5.16 se ingresan los parámetros básicos para configurar una red neuronal del tipo backpropagation. Por defecto para la prueba se configuran dos capas ocultas conformadas correspondientemente por 12 y 10 neuronas. Además se configura como 1000 el número máximo de épocas y el error mínimo permitido es de 1×10^{-3} [59].

The screenshot shows a window titled 'BackPropagation' with a sub-title 'Red Neuronal Back Propagation'. It is divided into several sections:

- Configuracion de la Red:**
 - Neuronas capa Oculta 1: 12
 - Neuronas capa Oculta 2: 10
 - Tasa Aprendizaje: 0.3
 - Epocas: 1000
 - Error Máximo: $1e-3$
- Resultado de la Prueba:**
 - Cantidad de Señales: 2500
 - Clasificadas Correctamente: 2459
 - Porcentaje Aciertos: 98.36
- Resultado del Entrenamiento:**
 - Epocas: 66
 - Error Máximo: 0.000984587
 - Tiempo de Entrenamiento: 175.175

Buttons for 'Entrenar', 'Prueba', and 'Exportar' are visible at the bottom of the interface.

Figura 5.16 - Clasificador basado en una Red Neuronal Backpropagation

Al presionar el botón *Entrenar*, el software conforma el set de entrenamiento utilizando la matriz *vectorCaracterísticoEntrenamiento* y el vector *reporteSeñalEntrenamiento*.

Una vez que el entrenamiento finaliza los campos destinados a los *Resultados del Entrenamiento* son poblados automáticamente. Para el ejemplo propuesto, el entrenamiento culminó alcanzando el error máximo configurado. El proceso del entrenamiento tomó 175.175 [segundos] y se emplearon 66 épocas para lograr un error de 0.000984587.

Para comprobar el rendimiento del clasificador se presiona el botón *Prueba*. El software utiliza la matriz *vectorCaracterístico* para obtener un vector con los resultados de la clasificación el que es cotejado con el vector *reporteSeñal* con el objetivo de detectar la

cantidad de clasificaciones correctas y la cantidad de clasificaciones erróneas. Los resultados obtenidos para nuestro ejemplo indican que de 2500 señales de prueba 2458 fueron clasificadas correctamente. El porcentaje de efectividad del clasificador es de 98.36%.

Es importante destacar que para corroborar la propiedad de *generalización* del clasificador no se utiliza el mismo conjunto de datos para el entrenamiento y prueba de los algoritmos de clasificación.

Si el porcentaje de efectividad del clasificador no se corresponde con un valor esperado es posible modificar los parámetros de configuración de la red neuronal y volver a ejecutar el entrenamiento.

Cuando la red neuronal alcanza el porcentaje de efectividad deseado se presiona el botón *Exportar*. Este botón genera un archivo denominado *parametrosRed.net* dentro del directorio donde se encuentra la aplicación que contiene los parámetros que caracterizan el comportamiento de la red (pesos, bias, arquitectura, etc.). Este archivo será utilizado por un software de monitoreo en tiempo real cuyo clasificador se basa en una red neuronal del tipo backpropagation.

5.6.4.2 Support Vector Machine

Para realizar una clasificación empleando algoritmos basados en máquinas de soporte vectorial se presiona la opción *Support Vector Machine* del panel principal presentado en la sección 5.6.1.

Tal como se explicó en la sección 5.4 este clasificador está conformado por nueve máquinas de soporte vectorial que serán entrenadas para diferenciar cada tipo de perturbación del resto. La Figura 5.17 muestra que existen dos parámetros (rbf_sigma y box constraint) que gobiernan el comportamiento de cada una de las 9 etapas del clasificador.

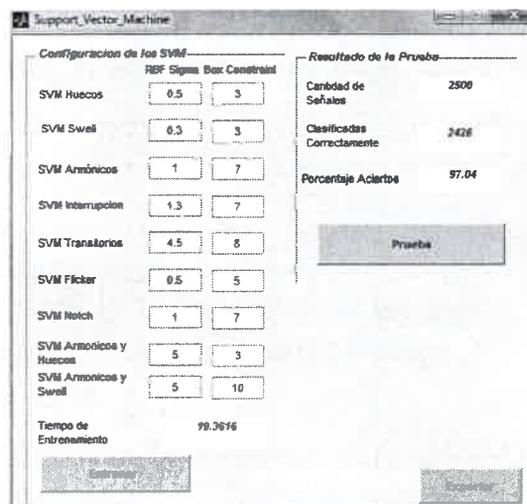


Figura 5.17 - Clasificador basado en Máquinas de Soporte Vectorial

Al presionar el botón *Entrenar* el software conforma el set de entrenamiento utilizando la matriz *vectorCaracterísticoEntrenamiento* y el vector *reporteSeñalEntrenamiento*.

Una vez que el entrenamiento finaliza el campo tiempo de Entrenamiento es poblado de manera automática. Para el ejemplo propuesto el proceso del entrenamiento tomó *19.3676 [segundos]*.

Para comprobar el rendimiento del clasificador se presiona el botón *Prueba*. Del mismo modo que el módulo basado en la red neuronal backpropagation, el software utiliza la matriz *vectorCaracterístico* para obtener un vector con los resultados de la clasificación el cual se compara con el vector *reporteSenal* para detectar la cantidad de clasificaciones correctas. Los resultados obtenidos para nuestro ejemplo indican que de 2500 señales de prueba 2426 fueron clasificadas correctamente. El porcentaje de efectividad del clasificador es de 97.04%.

Nuevamente, si el porcentaje de clasificaciones correctas no se corresponde con lo esperado pueden modificarse los parámetros que gobiernan a cada máquina de soporte vectorial y volver a entrenar el clasificador en búsqueda de los parámetros que optimicen el porcentaje de clasificaciones correctas.

Cuando el clasificador alcanza el porcentaje de efectividad deseado se presiona el botón *Exportar*. Este botón genera nueve archivos denominados *SVM1.mat*, *SVM2.mat*,...,*SVM9.mat* dentro del directorio donde se encuentra la aplicación. Estos archivos contienen los parámetros que caracterizan el comportamiento del clasificador y serán utilizados por el software de monitoreo en tiempo real cuyo clasificador se basa en máquinas de soporte vectorial.

5.6.5 Incorporación de señales reales

La Figura 5.18 ilustra una señal obtenida mediante el muestreo de una de las fases del sistema eléctrico de potencia de una planta de procesamiento de aceites.

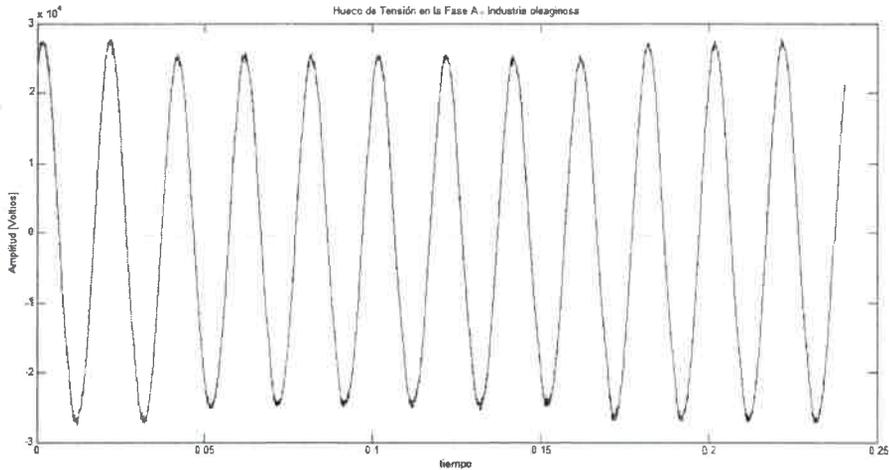


Figura 5.18 - Hueco de tensión adquirido de la red.

Para poder ser procesada por el sistema de monitoreo, la señal es normalizada y enventanada con una ventana de longitud de 11 ciclos, tal como se muestra en la Figura 5.19.

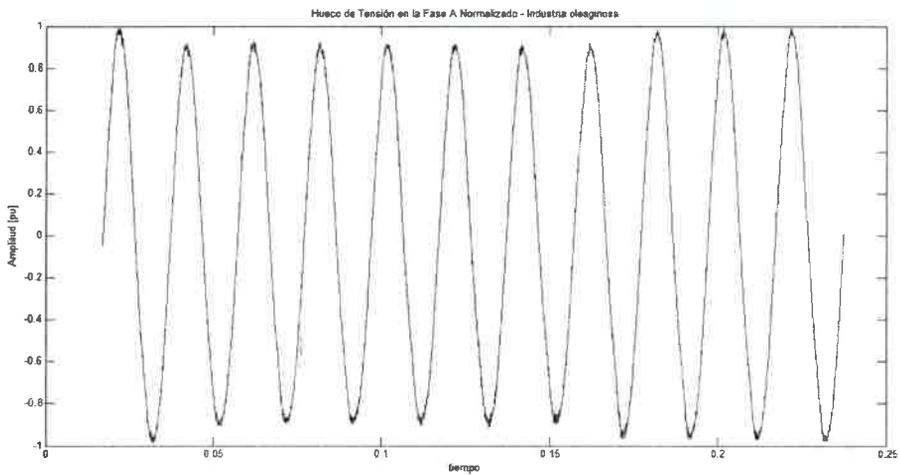


Figura 5.19 - Señal real normalizada.

5.6.6 Simulación de la clasificación

Para emular el comportamiento del módulo de detección y clasificación se desarrolló un módulo de simulación cuya interfaz gráfica se muestra en la Figura 5.20.

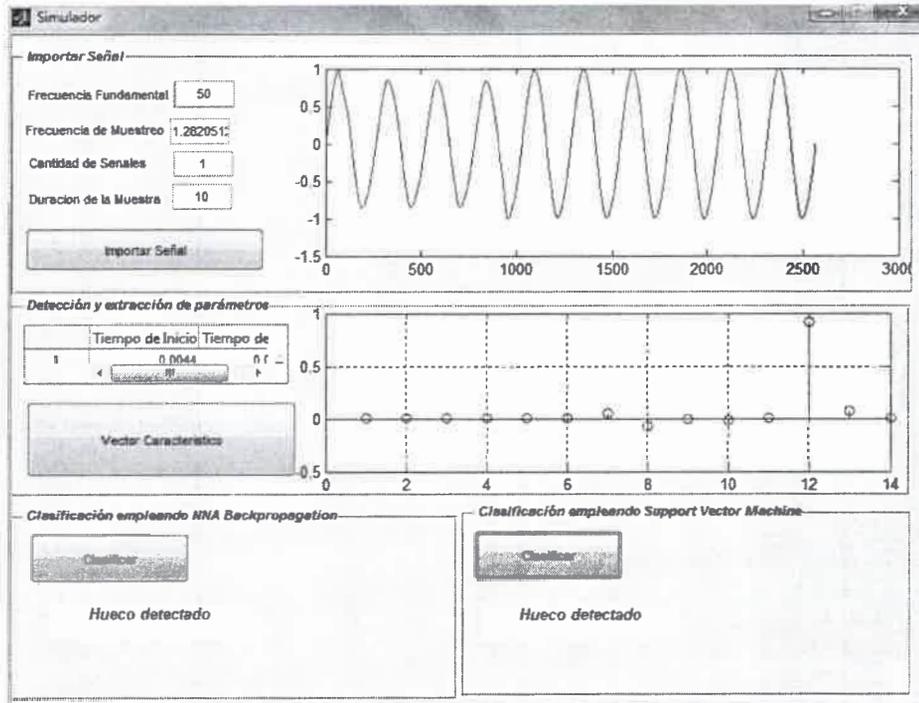


Figura 5.20 - Localización y clasificación de una señal real.

Para importar una señal desde un archivo se presiona el botón *Importar Señal*. Esta acción hace que emerja un cuadro de diálogo que permite la selección de un archivo de texto que contiene la señal perturbada. Además se carga la frecuencia fundamental, la frecuencia de muestreo, la cantidad de señales que contiene el archivo de texto a analizar y la duración de la muestra (longitud de la ventana).

Luego se ejecuta el algoritmo de extracción de parámetros característicos presionando el botón *Vector Característico*. Esta acción además ejecuta el algoritmo que localiza el evento dentro de la ventana de análisis y estima el tiempo de inicio y fin de la perturbación.

El módulo de Simulación fue desarrollado para probar los algoritmos de clasificación diseñados y entrenados tal como se explica en la sección 5.6.4. Cuando se presiona el botón *Clasificar* correspondiente a la Red Neuronal Backpropagation la herramienta busca en el directorio raíz el archivo denominado *parametrosRed.mat* que contiene los parámetros de la red neuronal y lo utiliza para clasificar el vector característico mostrando en pantalla el resultado de la clasificación.

De la misma manera, presionando el botón *Clasificar* correspondiente al clasificador basado en Máquinas de Soporte Vectorial, la herramienta automáticamente carga los archivos *SVM1.data*, *SVM2.data*, ... , *SVM9.data* los cuales contienen los parámetros

del clasificador entrenado como se detalla en la sección 5.6.4. Estos archivos son empleados para realizar la clasificación cuyo resultado es mostrado en pantalla.

5.7 CONCLUSION

Debido a la dificultad de contar con una gran variedad de señales obtenidas por medio de la medición de una red eléctrica, la primera parte de este capítulo presenta el desarrollo de una herramienta que permite la generación de perturbaciones de manera artificial. Este módulo probó ser útil para generar una gran variedad de eventos de calidad de energía los cuales fueron destinados a la confección de conjuntos de señales de entrenamiento y prueba para ser usados en la sintonización del sistema de clasificación.

En la segunda parte del capítulo se presentaron dos algoritmos, el primero basado en la Transformada Wavelet para la extracción de parámetros característicos de las perturbaciones y su localización temporal, y el segundo para el entrenamiento de los sistemas de clasificación.

En la tercera sección del capítulo se presentó el conjunto de etapas de un proceso de análisis de perturbaciones extraídas de una red eléctrica. Basándose en este proceso se desarrolló el software de procesamiento de señales aquí presentado.

Finalmente, en este capítulo se detalló un *caso de uso* del software desarrollado donde se destacó la utilización de la herramienta para la generación de señales artificiales, la confección de los set de entrenamiento y prueba, el entrenamiento de los sistemas de clasificación, la detección de las perturbaciones, la localización temporal de los eventos de calidad de energía y la utilización del software para el análisis de una señal adquirida en un sistema eléctrico de potencia de una planta industrial.

Capítulo 6: Resultados de las Experiencias

6.1 RESULTADOS Y SIMULACIONES

En este capítulo se presentan dos escenarios donde se medirá el rendimiento de los algoritmos desarrollados. Para ello se evaluarán dos parámetros característicos de cada clasificador, el porcentaje de acierto y el tiempo de entrenamiento.

La experiencia 1 evaluará el rendimiento de un algoritmo de clasificación *SVM multiclase* implementado con una arquitectura del tipo *árbol de decisión binaria*.

La experiencia 2, por otro lado, presenta una comparación del rendimiento de un clasificador basado en una *Red Neuronal backpropagation* y un algoritmo *SVM multiclase* implementado con una arquitectura *uno contra el resto*.

6.1.1 Experiencia 1: Clasificador SVM basado en un árbol de decisión binario

En este escenario se configura un clasificador de eventos multiclase del tipo árbol de decisión, en donde cada nodo de la estructura se corresponde con una Máquina de Soporte Vectorial binario. Como se muestra en la Figura 6.1 el clasificador SVM1, ubicado en el nodo raíz, es el responsable de realizar la primera clasificación separando todos los patrones de entrada en dos sub grupos. El nodo raíz es entrenado para diferenciar las perturbaciones de corta duración de aquellas que presentan perturbaciones estacionarias (armónicos). De manera similar, cada nodo del árbol es entrenado para separar en dos grupos las perturbaciones que se le presentan en sus respectivas entradas. En los nodos de menor jerarquía es donde recae la mayor complejidad de la estructura ya que los mismos se entrenan para separar las perturbaciones que son similares entre sí, lo que las hace difíciles de diferenciar.

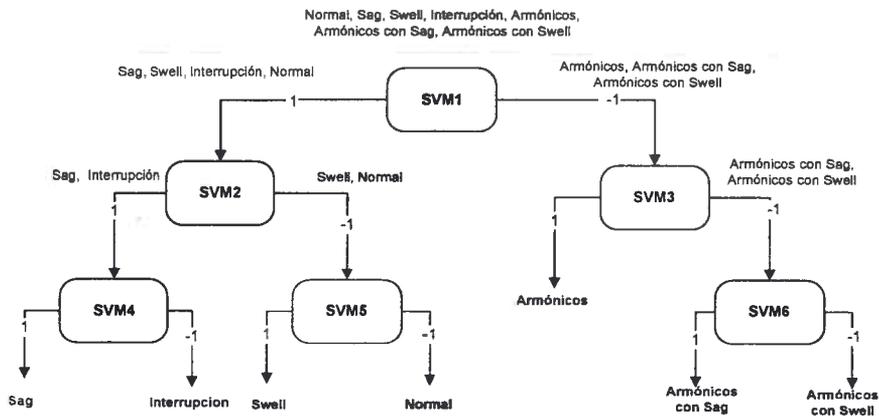


Figura 6.1 - Arquitectura de un clasificador SVM del tipo árbol de decisión binario

6.1.1.1 Conformación del set de entrenamiento y prueba

Para entrenar el árbol de decisiones se creó un set de entrenamiento específico para cada nodo. Seis tipos de perturbaciones fueron consideradas en este escenario.

Las señales empleadas para el entrenamiento del clasificador fueron simuladas utilizando la herramienta de generación de señales que forma parte del software análisis de perturbaciones presentado en el *Capítulo 5*.

Los parámetros fundamentales que gobiernan la simulación de las perturbaciones son:

- Longitud de la ventana: *20 ciclos*
- Frecuencia fundamental de *50 [Hz]*.
- Frecuencia de muestreo: *10000 [Muestras/segundo]*

La Tabla 6.1 resume los tipos de perturbaciones utilizados en cada set de entrenamiento. Para la creación de las señales de entrenamiento y prueba se utilizó la herramienta desarrollada para la generación de señales sintéticas presentada en el *Capítulo 5* del presente trabajo.

Tabla 6.1 - Set de Entrenamiento

<i>Nodo</i>	<i>Tipos de perturbaciones</i>	<i>Cantidad</i>	<i>Frecuencia de muestreo [Hz]</i>
SVM1	Normal - Sag - Swell - Interrupción - Armónicos - Armónicos con Sag - Armónicos con Swell	300	10000
SVM2	Sag - Swell - Interrupción - Normal	300	10000
SVM3	Armónicos, Armónicos con Sag, Armónicos con Swell	300	10000
SVM4	Sag – Interrupción	300	10000
SVM5	Swell – Normal	300	10000
SVM6	Armónicos y Sag- Armónicos y Swell	300	10000

Además de las señales de entrenamiento, un set de señales fue generado para la prueba de las del sistema clasificador. El set fue conformado por *300* señales de cada tipo de perturbación por lo que se emplearon un total de *2100* señales de prueba.

6.1.1.2 Resultados obtenidos

Una vez que el sistema de monitoreo fue entrenado, se realizaron las pruebas cuyos resultados se resumen en la Tabla 6.2.

El algoritmo de clasificación basado en Máquinas de Vectores de Soporte ha probado ser eficiente a la hora de clasificar eventos de calidad de energía ya que logró catalogar correctamente cerca del *92,5%* de las perturbaciones simuladas.

Tabla 6.2 - Matriz de confusión del Clasificador SVM

<i>Perturbación</i>	<i>Senoidal</i>	<i>Hueco</i>	<i>Swell</i>	<i>Interrupción</i>	<i>Armónicos</i>	<i>Armónicos con Hueco</i>	<i>Armónicos con Swell</i>	<i>Eficiencia (%)</i>
Senoidal	300	0	0	0	0	0	0	100
Hueco	0	296	2	2	0	0	0	98,66
Swell	0	0	297	3	0	0	0	99
Interrupción	0	13	0	278	0	0	9	92,6
Armónicos	0	0	0	26	274	0	0	91,3
Armónicos con Hueco	0	0	0	23	20	243	14	81
Armónicos con Swell	0	0	0	6	30	6	258	86

6.1.2 Experiencia 2: Comparación de un clasificador One vs All SVM con una ANN backpropagation.

En este escenario se plantea la comparación de dos tipos de clasificadores: el primero, basado en una Red Neuronal Artificial (ANN) backpropagation y el segundo basado en Máquinas de Soporte de Vectores (SVM).

La ANN fue diseñada con una arquitectura de dos capas ocultas conformadas por 12 y 10 neuronas respectivamente. Para el entrenamiento de la red se utilizó el algoritmo *Levenberg-Marquardt*. La función de activación seleccionada fue la función tangente hiperbólica para ambas capas ocultas y la tasa de aprendizaje seleccionada fue de 0.3.

Por otro lado, la arquitectura del clasificador multiclase basado en SVM es la denominada *uno contra el resto (one against rest)* como se muestra en la Figura 6.2. Esta arquitectura implica que se debe entrenar un SVM por cada tipo de perturbación. Por ejemplo, la máquina de soporte vectorial llamada SVM 1 es entrenada para presentar un 1 a su salida si el vector característico de entrada se corresponde con un Hueco y un -1 si se trata de cualquier otro tipo de perturbación.

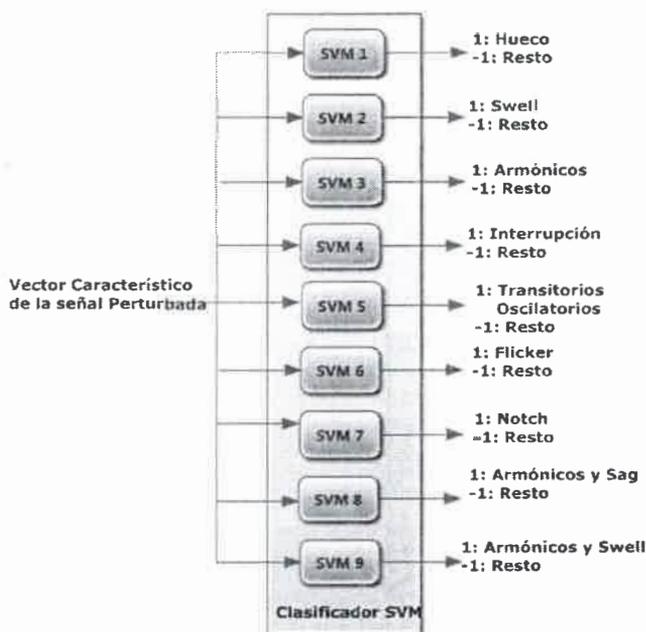


Figura 6.2 - Arquitectura de un clasificador SVM Uno contra el resto

6.1.2.1 Conformación del set de entrenamiento y prueba

Para el entrenamiento de la Red Neuronal se generó un set de señales compuesto por 1000 formas de onda por cada tipo de perturbación. Las 9000 señales fueron generadas por la herramienta de generación de perturbaciones presentada en el Capítulo 5.

Para que la comparación sea representativa, el mismo conjunto de señales utilizado para entrenar la Red Neuronal se utilizó para entrenar el clasificador SVM multiclase. Nueve diferentes grupos de perturbaciones de entrenamiento fueron creados de acuerdo a lo mostrado en la Tabla 6.3.

Tabla 6.3 - Set de entrenamiento para el clasificador SVM

Nodo	Tipos de perturbaciones	Cantidad	Frecuencia de muestreo [Hz]
SVM1	Huecos (1) – Resto (-1)	1000	10000
SVM2	Swell (1) – Resto (-1)	1000	10000
SVM3	Armónicos (1)- Resto (-1)	1000	10000
SVM4	Interrupcion (1) -. Resto (-1)	1000	10000
SVM5	Transitorios (1)– Resto(-1)	1000	10000
SVM6	Flicker (1) – Resto (-1)	1000	10000
SVM7	Notch(1) – Resto (-1)	1000	10000
SVM8	Armónicos y Sag (1) – Resto (-1)	1000	10000
SVM9	Armónicos y Swell (1) – Resto (-1)	1000	10000

Los parámetros fundamentales que gobiernan la simulación de las perturbaciones son:

- Longitud de la ventana: 20 ciclos
- Frecuencia fundamental de 50 [Hz].
- Frecuencia de muestreo: 10000 [Muestras/segundo]

6.1.2.2 Resultados obtenidos

En la Tabla 6.4 se presenta la matriz de confusión del clasificador basado en la Red Neuronal.

De las 9000 muestras empleadas para la prueba 8881 fueron correctamente clasificadas. Esto se corresponde con el 98,7% del total de las muestras empleadas para la prueba. El restante 1,33% de las muestras fueron erróneamente clasificadas.

Tabla 6.4 - Matriz de confusión de clasificador ANN

	<i>Hueco</i>	<i>Swell</i>	<i>Arm</i>	<i>Int</i>	<i>Trans</i>	<i>Flicker</i>	<i>Arm y Hueco</i>	<i>Arm y Swell</i>	<i>Notch</i>	<i>Sin Clasificar</i>
Huecos	997	0	0	0	0	0	0	0	3	0
Swell	0	1000	0	0	0	0	0	0	0	0
Armónicos	1	0	991	0	0	3	2	2	1	0
Interrupción	0	0	0	1000	0	0	0	0	0	0
Transitorios	1	0	1	0	985	0	0	0	13	0
Flicker	0	0	0	0	0	1000	0	0	0	0
Armónicos y Hueco	2	0	0	7	0	0	986	0	5	0
Armónicos y Swell	0	14	58	0	4	0	0	923	0	0
Notch	0	0	0	0	0	0	0	1	999	0

En la Tabla 6.5 se resumen los resultados obtenidos con el clasificador SVM multiclase. En este caso, del total de las muestras de prueba 8763 fueron clasificadas correctamente lo que nos da un porcentaje de acierto del 97,4%.

En la matriz de confusión se encuentra que un total de 215 muestras fueron erróneamente clasificadas, valor que se corresponde al 2,15% del total del set de pruebas.

Finalmente, en este ejemplo, se encuentra un tercer grupo compuesto por 22 muestras que no fueron clasificadas en ninguna de las clases analizadas.

Es importante destacar que, aunque algunas muestras no son clasificadas, el algoritmo de detección las descubre con éxito. El reporte del sistema de procesamiento de las perturbaciones indica la detección de 9000 muestras, la clasificación de 8978

perturbaciones (8763 correctas y 215 erradas) y el tiempo de inicio y fin de cada perturbación de carácter no estacionario.

Tabla 6.5 - Matriz de confusión del clasificador SVM

	Hueco	Swell	Arm	Int	Trans	Flicker	Arm y Hueco	Arm y Swell	Notch	Sin Clasificar
Huecos	1000	0	0	0	0	0	0	0	0	0
Swell	0	1000	0	0	0	0	0	0	0	0
Armónicos	3	1	994	0	0	1	0	0	0	1
Interrupción	0	0	0	1000	0	0	0	0	0	0
Transitorios	1	0	0	0	979	0	0	0	20	0
Flicker	0	0	0	0	0	1000	0	0	0	0
Armónicos y Hueco	4	0	28	0	0	0	954	0	0	14
Armónicos y Swell	0	1	87	0	0	0	0	905	0	7
Notch	69	0	0	0	0	0	0	0	931	0

6.2 CONCLUSIONES

6.2.1 Conclusiones de la Experiencia 1

Considerando los resultados presentados en la Tabla 6.2, se concluye que muchas de las clasificaciones erróneas se deben a una marcada similitud entre los eventos de calidad de energía como, por ejemplo, los huecos de tensión con una interrupción.

Para las perturbaciones del tipo *armónicos con huecos* el clasificador comete treinta errores al detectar solo la presencia de los armónicos sin detectar la presencia de los huecos. Esta clasificación podría considerarse como *parcialmente correcta* lo que provocaría que la eficiencia del esquema seleccionado sea aún mayor al 92.5%.

La arquitectura tipo árbol empleada en el clasificador presenta la cualidad de ser más rápida que un esquema normal versus perturbación (NVD). Para realizar una clasificación, el sistema de monitoreo presentado, a lo sumo, emplea tres etapas para evaluar una perturbación, mientras que un esquema que evalúa empleando una estrategia NVD debería evaluar una etapa por cada tipo de evento.

Una desventaja de la arquitectura seleccionada radica en la dificultad de sintonizar las 6 etapas de manera simultánea para obtener un rendimiento adecuado.

Un error en el nodo raíz puede llevar a una confusión entre dos tipos de perturbaciones aunque los eventos sean marcadamente de diferente naturaleza. Por ejemplo, en la Tabla 6.2 se puede ver que seis perturbaciones del tipo *armónicos con swell* fueron clasificados como una *interrupción*. Esta equivocación se produce en el nodo raíz y, por lo tanto, es arrastrada hacia abajo por todo el árbol de decisión. Al manejar muchos

tipos de perturbaciones a la vez, el nodo raíz es el más propenso a cometer errores, es difícil de entrenar y el más difícil de sintonizar.

La conformación de un árbol de decisión binaria implica la programación de un código que modele la arquitectura del árbol. Esto representa una desventaja ya que, ante la aparición de un nuevo tipo de perturbación no solamente se deben re entrenar cada uno de los clasificadores binarios si no que, además, se deberá desarrollar un código que contemple la nueva arquitectura del árbol.

La clasificación tiene lugar comenzando desde el nodo raíz y continúa hacia los niveles inferiores del árbol de decisión binaria. La jerarquía inferior del árbol se conforma por clasificadores binarios que están entrenados para diferenciar por una de dos posibles clases. Por lo tanto esta arquitectura de clasificación siempre presentará un resultado unívoco a su salida. Esto representa una ventaja frente a los clasificadores One vs All ya que, al no presentarse resultado ambiguos, no se necesitará un esquema de post procesamiento.

6.2.2 Conclusiones de la Experiencia 2

Ambos clasificadores presentan una notable paridad en lo referente al porcentaje de muestras correctamente clasificadas siendo el clasificador basado en las ANN, con un porcentaje de 98,7%, levemente superior al clasificador SVM cuyo porcentaje de clasificación es de 97,4% aproximadamente.

El tiempo de entrenamiento del clasificador ANN es de tres a cuatro veces mayor al tiempo de entrenamiento del clasificador SVM. Para la cantidad de muestras propuesta en la experiencia 2, el entrenamiento de la Red Neuronal se realiza en un tiempo que va desde los 45 a los 75 segundos mientras que, para el mismo set de entrenamiento, el clasificador SVM es entrenado en un tiempo que va desde los 11 a los 16 segundos. Esto representa una ventaja en aquellos clasificadores cuyo entrenamiento se ajusta mientras la red neuronal se encuentra operativa. Para el esquema de monitoreo presentado en esta tesis la red neuronal se entrena una única vez antes de considerarse operativa, por lo que el tiempo de entrenamiento no resulta de extrema importancia.

En algunos casos el entrenamiento de la Red Neuronal debió ser repetido producto del atascamiento del algoritmo en un mínimo local. En tal caso el entrenamiento finalizó por la expiración de la cantidad máxima de iteraciones (épocas) configurado en la interfaz gráfica. Este problema no se presenta en el clasificador basado en SVM.

Debido a la arquitectura empleada para el clasificador SVM multiclase, existen casos en donde una muestra no es clasificada con ninguna categoría propuesta (ver Tabla 6.5). Esto no ocurre con el clasificador ANN ya que su salida siempre se corresponde con alguna categoría válida.

La salida esperada de la clasificación de una red neuronal debería ser un uno para aquella neurona correspondiente a un tipo de perturbación en particular y un cero para el resto de las neuronas de la capa de salida. Sin embargo, debido a que las neuronas de la

capa de salida responden con valores que no pertenecen al conjunto binario es necesario emplear un esquema de post procesamiento para reconocer aquella neurona cuyo valor sea cercano a uno y aquellas neuronas cuyo valor se aproxima a cero. Este esquema de post procesamiento emplea una estrategia de umbralización para etiquetar con un uno aquellas neuronas cuyo valor de salida sea próximo a la unidad y etiquetar con un cero a aquellas neuronas que se aproximan al valor nulo.

Idealmente se espera que, ante un vector característico de entrada, un clasificador que emplea una arquitectura One vs All presente un 1 en la salida solamente de aquel nodo que fue entrenado para clasificar la perturbación en cuestión y un 0 en el resto de los nodos. Sin embargo, esta arquitectura no está exenta de resultados ambiguos ya que, para un mismo vector característico, más de un nodo SVM puede responder con un 1. Para resolver los casos donde se presentan ambigüedades en la clasificación, un esquema de post procesamiento denominado fue implementado.

Por lo tanto, para la interpretación correcta de los resultados entregados por ambos clasificadores fue necesario desarrollar un sistema de post-procesamiento.

6.2.3 Comparación Experiencia 1 con Experiencia 2

El objetivo de esta sección es comparar el clasificador de la experiencia 1 correspondiente a un algoritmo basado en técnicas de SVM que emplea una arquitectura del tipo árbol de decisión binaria presentado en la sección 6.1.1 con el clasificador SVM cuya arquitectura es uno contra el resto de la experiencia 2 presentado en la sección 6.1.2.

La conclusión más importante está relacionada al rendimiento. El clasificador SVM de la experiencia 2, con un porcentaje de acierto de 97,4%, es notoriamente más preciso que el clasificador SVM de la experiencia 1 cuyo porcentaje de acierto es de 92,5%

El clasificador de la experiencia 1 es claramente más difícil de sintonizar ya que los parámetros modificados en este clasificador binario afectará todas las clasificaciones que se realizarán por el resto de los clasificadores por debajo en la jerarquía del árbol.

Debido a que su arquitectura tipo árbol de decisión, el clasificador de la experiencia 1 es más complicado de modificar ante la aparición de un nuevo tipo de perturbación. La efectividad en la clasificación de este método se ve reducida a medida que se incorporan nuevos patrones. Por este motivo, en la experiencia 1 solo se tuvieron en cuenta 6 tipos diferentes de perturbaciones.

Capítulo 7: Conclusiones y Trabajos Futuros

7.1 CONCLUSIONES

A lo largo de este trabajo se ha demostrado que el monitoreo de eventos de calidad de energía en un Sistema Eléctrico de Potencia representa una herramienta fundamental para disminuir pérdidas económicas producto de los eventos de Calidad de Energía tanto para las empresas proveedoras de energía eléctrica como así también para sus clientes.

En la primera parte del trabajo se abordaron los aspectos teóricos y matemáticos más relevantes de las técnicas que se emplearon posteriormente en el desarrollo del sistema de procesamiento de formas de onda perturbadas.

Con el fin de tipificar las perturbaciones que se presentan en un sistema eléctrico de potencia de acuerdo a las variaciones relativas de los parámetros que las gobiernan, este trabajo adoptó el estándar IEEE Std 1159-2009. Este estándar fue también empleado como un marco de trabajo para el desarrollo de los distintos componentes que conforman el sistema desarrollado. Por este motivo, en este trabajo de tesis se presentaron los aspectos más relevantes del estándar en lo referente a la tipificación de las perturbaciones y la conformación de un sistema de monitoreo.

Se abordaron los conceptos básicos de la *Transformada de Fourier (FT)* y la *Transformada de Fourier dependiente del tiempo (STFT)* para el análisis de señales estacionarias y no estacionarias demostrando las limitaciones y desventajas que ambos métodos presentan frente a la *Transformada Wavelet*. Se presentaron también los conceptos teóricos y analíticos relativos a la *Transformada Wavelet continua (CWT)*, la *Transformada Wavelet Discreta (DWT)* y una útil interpretación de la transformada wavelet discreta representada por un banco de filtros.

Además se presentaron los aspectos generales de las *Redes Neuronales Artificiales* como su funcionamiento, arquitectura, funciones de activación y algoritmos de entrenamiento, para luego enfocarse en un tipo de red en particular denominado *feedforward backpropagation*. Este tipo de redes fue seleccionado debido a su probada habilidad para resolver problemas de asociación y clasificación de patrones, característica que las volvió ideales para el desarrollo del sistema de monitoreo. Se presentaron la arquitectura, las características que deben cumplir las funciones de activación, el algoritmo de entrenamiento y las reglas de diseño utilizadas para una red *backpropagation*.

A pesar de resultar una herramienta de notoria complejidad matemática, la literatura indica que las *Máquinas de Soporte Vectorial* presentan resultados equivalentes o superiores frente a otras técnicas de clasificación. Es por este motivo que en este trabajo se presentaron los conceptos teóricos y matemáticos involucrados en el uso y entrenamiento de un clasificador basado en *Máquinas de Vectores de Soporte*. También se analizaron varias estrategias para transformar un clasificador SVM binario en un clasificador multiclase.

Como una de las contribuciones más relevantes de este trabajo de tesis se menciona el desarrollo de un software de procesamiento de señales que permite la detección, clasificación y localización de perturbaciones en una forma de onda adquirida de un Sistema Eléctrico de Potencia. Esta herramienta fue desarrollada para permitir:

- La generación de perturbaciones sintéticas: Este módulo probó ser útil para generar una gran variedad de eventos de calidad de energía los cuales fueron destinados a la confección de conjuntos de señales de entrenamiento y prueba para ser usados en la sintonización del sistema de clasificación.
- Extracción de parámetros característicos de las perturbaciones y su localización temporal: El algoritmo desarrollado está basado en la *Transformada Wavelet* que utiliza los coeficientes de alta frecuencia para la detección de las perturbaciones y el cálculo del tiempo de inicio y duración de cada evento de calidad de energía.
- Generación de los vectores característicos: Se desarrolló un algoritmo que utiliza los coeficientes de la *Transformada Wavelet*, el Teorema de Parseval y una técnica de extracción de parámetros de la amplitud de la señal para la generación de un vector característico de 14 componentes.
- Clasificación de las perturbaciones: Se desarrollaron dos algoritmos para la clasificación de perturbaciones. El primero de ellos fue basado en las *Redes Neuronales Artificiales (ANN)* y el segundo en *Máquinas de Soporte Vectorial (SVM)*. El software desarrollado permite el entrenamiento y la prueba de estos algoritmos.

Todos los desarrollos mencionados en este trabajo fueron realizados en *MATLAB*. Las Toolbox empleadas fueron:

- Wavelet Transform Toolbox
- Neural Network Toolbox
- BioinformaticsToolbox (SVM)

De los resultados obtenidos se concluye que:

- El rendimiento del clasificador basado en la *Red Neuronal* fue el más preciso logrando un 98,7% de aciertos en sus clasificaciones.
- El clasificador *SVM* multiclase configurado con una arquitectura del tipo *uno contra el resto* logró una efectividad de aciertos similar al clasificador *ANN* ya que los resultados de la experiencia mostraron un porcentaje de aciertos de aproximadamente del 97,4%.
- El clasificador *SVM* multiclase configurado con un arquitectura *árbol de decisión binario* logró el peor rendimiento de los tres clasificadores desarrollados. Su porcentaje de aciertos fue de 92.5%.
- La arquitectura tipo árbol de decisión binaria presenta la característica de ser *más rápida que uno contra el resto*. El clasificador emplea, a lo sumo, tres

etapas para evaluar una perturbación, mientras que un esquema que evalúa empleando una estrategia *uno contra el resto* debería evaluar una etapa por cada tipo de evento.

- El tiempo de entrenamiento del clasificador *NNA* desarrollado, es de tres a cuatro veces mayor al tiempo de entrenamiento del clasificador *SVM*. Esta conclusión no resulta determinante para la aplicación presentada en este trabajo de tesis ya que el entrenamiento del clasificador se realiza en una instancia previa a su aplicación práctica.
- En algunos casos el entrenamiento del clasificador basado en la *ANN* finalizó por la expiración de la cantidad máxima de iteraciones (épocas). Este problema se conoce como atascamiento en un mínimo local de la función de entrenamiento. Este problema no se presenta en el clasificador basado en *SVM*.

7.2 TRABAJOS FUTUROS

En base a los métodos desarrollados, estos se podrían extender a futuro de la siguiente manera:

- El software fue íntegramente desarrollado empleando las Toolbox nativas de MATLAB. Sin embargo existen otras librerías, desarrolladas en lenguajes como C, C++, que podrían ser compiladas y empleadas para comparar el rendimiento con las herramientas que MATLAB presenta por defecto. Algunas de estas librerías son:
 - LibSVM
 - SVR
 - OSU SVM
 - SVM-KM Toolbox
- Como parte de trabajos futuros se pretende realizar pruebas con un software de Data Mining desarrollado por la Universidad de Waikato llamado Weeka. Este software, que corre sobre una plataforma Java, es una herramienta libre que permite comparar diferentes métodos de clasificación. Esta herramienta podría utilizarse en trabajos futuros para la comparación del rendimiento de distintos métodos de clasificación de eventos de calidad de energía.
- De acuerdo a lo que se ha explicado en el *Capítulo 5* de este trabajo, las formas de onda utilizadas para el entrenamiento y prueba de los sistemas de clasificación fueron señales obtenidas mediante la simulación matemática de los eventos de calidad de energía. Estudios futuros podrían contemplar el empleo de señales reales para el entrenamiento y prueba de la herramienta desarrollada para este trabajo de tesis.
- A pesar que la herramienta desarrollada permite la generación de perturbaciones corruptas con ruido los estudios realizados en este trabajo no contemplaron el análisis de técnicas de pre procesamiento para la eliminación de componentes ruidosas. Para el estudio y comparación del rendimiento de los clasificadores con señales corruptas con ruido se podría incorporar una etapa de pre procesamiento con algún tipo de estrategia de eliminación de ruido (*denoising* en inglés).
- Como la herramienta fue desarrollada de manera modular, se podrían desarrollar funcionalidades adicionales que contemplen:
 - La incorporación de métodos de procesamiento de señales alternativos a la *Transformada Wavelet* para la detección y extracción de parámetros característicos de una señal.
 - La incorporación de otros métodos de clasificación.
- Luego de haber probado y comparado diferentes algoritmos de procesamiento de señales y técnicas de clasificación de patrones un trabajo

futuro podría ser el desarrollar la herramienta en un lenguaje de programación como *C*, *C++*, *C#* o *Java*.

- Un estudio futuro podría enfocarse en corroborar el rendimiento de los algoritmos desarrollados en lo referente al consumo del tiempo. Esto permitiría contar con información importante para depurar el software y volverlo óptimo en cuanto al consumo de tiempo. El código optimizado podría formar parte de un sistema de monitoreo de señales en tiempo real.

Capítulo 8: Referencias

- [1] M. H. Bollen and I. Gu, "Signal Processing of Power Quality Disturbances," 2006.
- [2] "IEEE Recommended Practice for Monitoring Electric Power Quality," Journal, vol., no. Issue, pp. c1-81, 2009
- [3] B. W. Jaekel, "Description and classification of electromagnetic environments - revision of IEC 61000-2-5," Electromagnetic Compatibility, 2008. EMC 2008. IEEE International Symposium on, pp. 1-4, 2008
- [4] B. K. Panigrahi and V. R. Pandi, "Optimal feature selection for classification of power quality disturbances using wavelet packet-based fuzzy k-nearest neighbour algorithm," Generation, Transmission & Distribution, IET, vol. 3, no. 3, pp. 296-306, 2009.
- [5] A. J. Roscoe, G. M. Burt and J. R. McDonald, "Frequency and fundamental signal measurement algorithms for distributed control and protection applications," Generation, Transmission & Distribution, IET, vol. 3, no. 5, pp. 485-495, 2009.
- [6] G. T. Heydt, P. S. Fjeld, C. C. Liu, D. Pierce, L. Tu and G. Hensley, "Applications of the windowed FFT to electric power quality assessment," Power Delivery, IEEE Transactions on, vol. 14, no. 4, pp. 1411-1416, 1999.
- [7] Z. Fusheng, G. Zhongxing and Y. Wei, "The algorithm of interpolating windowed FFT for harmonic analysis of electric power system," Power Delivery, IEEE Transactions on, vol. 16, no. 2, pp. 160-164, 2001.
- [8] C. Soo-Hwan, J. Gilsoo and K. Sae-Hyuk, "Time-Frequency Analysis of Power-Quality Disturbances via the Gabor-Wigner Transform," Power Delivery, IEEE Transactions on, vol. 25, no. 1, pp. 494-499, 2010.
- [9] S. Santoso, E. J. Powers, W. M. Grady and P. Hofmann, "Power quality assessment via wavelet transform analysis," Power Delivery, IEEE Transactions on, vol. 11, no. 2, pp. 924-930, 1996.
- [10] R. G. Stockwell, L. Mansinha and R. P. Lowe, "Localization of the complex spectrum: the S transform," Signal Processing, IEEE Transactions on, vol. 44, no. 4, pp. 998-1001, 1996.
- [11] P. K. Dash, B. K. Panigrahi and G. Panda, "Power quality analysis using S-transform," Power Delivery, IEEE Transactions on, vol. 18, no. 2, pp. 406-411, 2003.
- [12] Z. Fengzhan and Y. Rengang, "Power-Quality Disturbance Recognition Using S-Transform," Power Delivery, IEEE Transactions on, vol. 22, no. 2, pp. 944-950, 2007.
- [13] T. Radil, P. M. Ramos and A. C. Serra, "Detection and extraction of harmonic and non-harmonic power quality disturbances using sine fitting methods," Harmonics and Quality of Power, 2008. ICHQP 2008. 13th International Conference on, pp. 1-6, 2008
- [14] M. Caujolle, M. Petit, G. Fleury and L. Berthet, "Reliable power disturbance detection using wavelet decomposition or harmonic model based Kalman filtering," Harmonics and Quality of Power (ICHQP), 2010 14th International Conference on, pp. 1-6, 2010
- [15] S. Santoso, E. J. Powers, W. M. Grady and A. C. Parsons, "Power quality disturbance waveform recognition using wavelet-based neural classifier. II. Application," Power Delivery, IEEE Transactions on, vol. 15, no. 1, pp. 229-235, 2000.
- [16] C. Jaehak, E. J. Powers, W. M. Grady and S. C. Bhatt, "Power disturbance classifier using a rule-based method and wavelet packet-based hidden Markov model," Power Delivery, IEEE Transactions on, vol. 17, no. 1, pp. 233-241, 2002.
- [17] H. Jiansheng, M. Negnevitsky and D. T. Nguyen, "A neural-fuzzy classifier for recognition of power quality disturbances," Power Delivery, IEEE Transactions on, vol. 17, no. 2, pp. 609-616, 2002.

- [18] M. B. I. Reaz, F. Choong, M. S. Sulaiman, F. Mohd-Yasin and M. Kamada, "Expert System for Power Quality Disturbance Classifier," *Power Delivery, IEEE Transactions on*, vol. 22, no. 3, pp. 1979-1988, 2007.
- [19] L. Whei-Min, W. Chien-Hsien, L. Chia-Hung and C. Fu-Sheng, "Detection and Classification of Multiple Power-Quality Disturbances With Wavelet Multiclass SVM," *Power Delivery, IEEE Transactions on*, vol. 23, no. 4, pp. 2575-2582, 2008.
- [20] A. Srivastava, A. Siddharth and A. K. Singh, "Some observations on selection of most suitable mother wavelet for measurement of power system harmonics using CWT," *TENCON 2009 - 2009 IEEE Region 10 Conference*, pp. 1-6, 2009
- [21] V. Vega, C. Duarte, G. Ordonez and N. Kagan, "Selecting the best wavelet function for power quality disturbances identification patterns," *Harmonics and Quality of Power, 2008. ICHQP 2008. 13th International Conference on*, pp. 1-6, 2008
- [22] A. M. Gaouda, M. M. A. Salama, M. R. Sultan and A. Y. Chikhani, "Power quality detection and classification using wavelet-multiresolution signal decomposition," *Power Delivery, IEEE Transactions on*, vol. 14, no. 4, pp. 1469-1476, 1999.
- [23] V. Vega, N. Kagan, G. Ordonez and C. Duarte, "Automatic power quality disturbance classification using wavelet, Support Vector Machine and Artificial Neural Network," *Electricity Distribution - Part 1, 2009. CIRED 2009. 20th International Conference and Exhibition on*, pp. 1-4, 2009
- [24] V. Vega, C. Duarte, G. Ordonez and N. Kagan, "Selecting the best wavelet function for power quality disturbances identification patterns," *Harmonics and Quality of Power, 2008. ICHQP 2008. 13th International Conference on*, vol., no., pp. 1-6, 2008.
- [25] Y. M. M. Miiti, G. Openheim y J. M. Pogy, "Wavelet Toolbox, User's Guide Version 2," *The Math Works, Inc 2000*, vol., no., pp., 2000.
- [26] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 11, no. 7, pp. 674-693, 1989.
- [27] I. Daubechies, "Ten Lectures on Wavelets," *Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania*, vol., no., pp., 1992.
- [28] O. Poisson, P. Rioual and M. Meunier, "New signal processing tools applied to power quality analysis," *Power Delivery, IEEE Transactions on*, vol. 14, no. 2, pp. 561-566, 1999.
- [29] A. M. Gaouda, M. M. A. Salama, M. R. Sultan and A. Y. Chikhani, "Application of multiresolution signal decomposition for monitoring short-duration variations in distribution systems," *Power Delivery, IEEE Transactions on*, vol. 15, no. 2, pp. 478-485, 2000.
- [30] T. Weiming, S. Xuelei, L. Jingbo and Z. Zhiheng, "Detection and Classification of Power Quality Disturbances Based on Wavelet Packet Decomposition and Support Vector Machines," *Signal Processing, 2006 8th International Conference on*, pp., 2006
- [31] J. Barros and R. I. Diego, "Analysis of Harmonics in Power Systems Using the Wavelet-Packet Transform," *Instrumentation and Measurement, IEEE Transactions on*, vol. 57, no. 1, pp. 63-69, 2008.
- [32] S. Santoso, E. J. Powers and W. M. Grady, "Power quality disturbance data compression using wavelet transform methods," *Power Delivery, IEEE Transactions on*, vol. 12, no. 3, pp. 1250-1257, 1997.
- [33] Y. Hong-Tzer and L. Chiung-Chou, "A de-noising scheme for enhancing wavelet-based power quality monitoring system," *Power Delivery, IEEE Transactions on*, vol. 16, no. 3, pp. 353-360, 2001.

- [34] L. Fausett, "Fundamentals of Neural Networks: Architectures, Algorithms, and Applications," 2006.
- [35] http://ar.kalipedia.com/ecologia/tema/neurona.html?x=20070417klpcnavid_152.Kes, Journal, vol., no. Issue, pp.,
- [36] B. a. H. Demuth, "Neural Network Toolbox," Mathworks, vol., no., pp., 2000.
- [37] Z. G. Boger, Hugo, "Knowledge extraction from artificial neural network models," Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on, vol., no., pp., 1997.
- [38] M. J. A. Berry, and Linoff, G., "Data Mining Techniques," 1997.
- [39] A. lum, "Neural Networks in C++," 1992.
- [40] A. K. Ghosh and D. L. Lubkeman, "The classification of power system disturbance waveforms using a neural network approach," Power Delivery, IEEE Transactions on, vol. 10, no. 1, pp. 109-115, 1995.
- [41] S. Santoso, E. J. Powers, W. M. Grady and A. C. Parsons, "Power quality disturbance waveform recognition using wavelet-based neural classifier. I. Theoretical foundation," Power Delivery, IEEE Transactions on, vol. 15, no. 1, pp. 222-228, 2000.
- [42] J. V. Wijayakulasooriya, G. A. Putrus and P. D. Minns, "Electric power quality disturbance classification using self-adapting artificial neural networks," Generation, Transmission and Distribution, IEE Proceedings-, vol. 149, no. 1, pp. 98-101, 2002.
- [43] I. Monedero, C. Leon, J. Roperro, A. Garcia, J. M. Elena and J. C. Montano, "Classification of Electrical Disturbances in Real Time Using Neural Networks," Power Delivery, IEEE Transactions on, vol. 22, no. 3, pp. 1288-1296, 2007.
- [44] S. Mishra, C. N. Bhende and K. B. Panigrahi, "Detection and Classification of Power Quality Disturbances Using S-Transform and Probabilistic Neural Network," Power Delivery, IEEE Transactions on, vol. 23, no. 1, pp. 280-287, 2008.
- [45] I. C. Cheng, "Virtual Multifunction Power Quality Analyzer Based on Adaptive Linear Neural Network," Industrial Electronics, IEEE Transactions on, vol. 59, no. 8, pp. 3321-3329, 2012.
- [46] V. Vapnik, "The Nature of Statistical Learning Theory," 2000.
- [47] G. B. Dantzig, "Linear programming and extensions," 1998.
- [48] L. Lasdon, "Optimization theory for large systems," 1970.
- [49] N. Cristianini and J. Shawe-Taylor, "An Introduction to Support Vector Machines and Other Kernel-based Learning Methods," 2000.
- [50] J. Watson, "Support Vector Machine for Multi-Class Pattern Recognition," Department of Computer Science, Royal Holloway, University of London, vol., no., pp., 1999.
- [51] H. Osman, "Novel Multiclass SVM-Based Binary Decision Tree Classifier," Signal Processing and Information Technology, 2007 IEEE International Symposium on, pp. 880-883, 2007
- [52] H. Chih-Wei and L. Chih-Jen, "A comparison of methods for multiclass support vector machines," Neural Networks, IEEE Transactions on, vol. 13, no. 2, pp. 415-425, 2002.
- [53] G. Peisheng and W. Weilin, "Power Quality Disturbances Classification using Wavelet and Support Vector Machines," Intelligent Systems Design and Applications, 2006. ISDA '06. Sixth International Conference on, pp. 201-206, 2006
- [54] P. Janik and T. Lobos, "Automated classification of power-quality disturbances using SVM and RBF networks," Power Delivery, IEEE Transactions on, vol. 21, no. 3, pp. 1663-1669, 2006.

- [55] P. G. V. Axelberg, I. Y. H. Gu and M. H. J. Bollen, "Support Vector Machine for Classification of Voltage Disturbances," *Power Delivery, IEEE Transactions on*, vol. 22, no. 3, pp. 1297-1303, 2007.
- [56] A. Milchevski and D. Taskovski, "Classification of power quality disturbances using wavelet transform and SVM decision tree," *Electrical Power Quality and Utilisation (EPQU), 2011 11th International Conference on*, pp. 1-5, 2011
- [57] Eris, x, H. ti and Y. Demir, "Automatic classification of power quality events and disturbances using wavelet transform and support vector machines," *Generation, Transmission & Distribution, IET*, vol. 6, no. 10, pp. 968-976, 2012.
- [58] B. Biswal, M. K. Biswal, P. K. Dash and S. Mishra, "Power quality event characterization using support vector machine and optimization using advanced immune algorithm," *Neurocomputing*, vol. 103, no. 0, pp. 75-86, 2013.
- [59] B. a. H. Demuth, "Neural Network Toolbox " MATLAB, vol., no., pp., 2009.

ANEXO A

MANUAL DE USUARIO DEL SOFTWARE

A.1 INTRODUCCION

En el presente capítulo se describen las funcionalidades del Software *Analizador PQ* y sus aspectos más importantes.

La herramienta, enteramente desarrollada en MATLAB, cuenta con una interfaz gráfica que permite el acceso de manera intuitiva a todas las funcionalidades del software.

Este software permite el análisis de señales obtenidas mediante el muestreo de señales reales así como también de señales generadas de manera artificial.

La herramienta cuenta con un módulo para la generación señales perturbadas permitiendo la configuración de las características más importantes que gobiernan a *cada tipo de perturbación como, por ejemplo, el tiempo de duración, la magnitud o la frecuencia del evento.*

El software permite detectar la presencia de un evento de calidad de energía y extraer distintas características de dicha señal que son de utilidad para el proceso de clasificación.

El módulo de clasificación permite, en función de las características de cada perturbación, la tipificación de las mismas de acuerdo conjunto pre determinado de posibles eventos de calidad de energía.

A.2 SOFTWARE PARA EL ANALISIS DE EVENTOS DE CALIDAD DE ENERGIA

A.2.1 Panel principal

Para acceder a la herramienta se debe ejecutar la aplicación *MATLAB* y configurar el directorio raíz donde se encuentran los scripts del software. En la ventana de comandos, como se muestra en la Figura A.1 , se invoca al panel principal de la herramienta por medio de la siguiente instrucción: *Analizador_de_señales*.



Figura A.1 - Invocación al panel principal

Al ejecutar este comando se despliega el panel principal de la aplicación mostrado en la Figura A.2. Esta interfaz permite al usuario el acceso a los siguientes módulos cuyas funcionalidades las cuales serán explicadas en detalle en el presenta capítulo:

1. Módulo Adquisidor de Señales
2. Módulo generador de Señales
3. Módulo de detección de Eventos
 - a. Transformada Wavelet
4. Módulo de clasificación de Eventos
 - a. NNA BackPropagation
 - b. Support Vector Machine

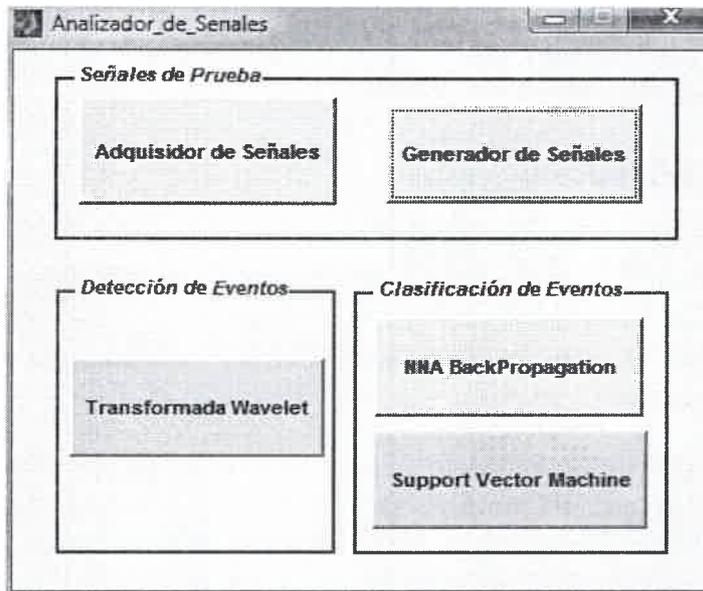


Figura A.2 - Panel Principal

A.2.2 Módulo Adquisidor de Señales

Al seleccionar la opción Adquisidor de Señales del panel principal se despliega un panel como el que se muestra en la siguiente Figura A.3:

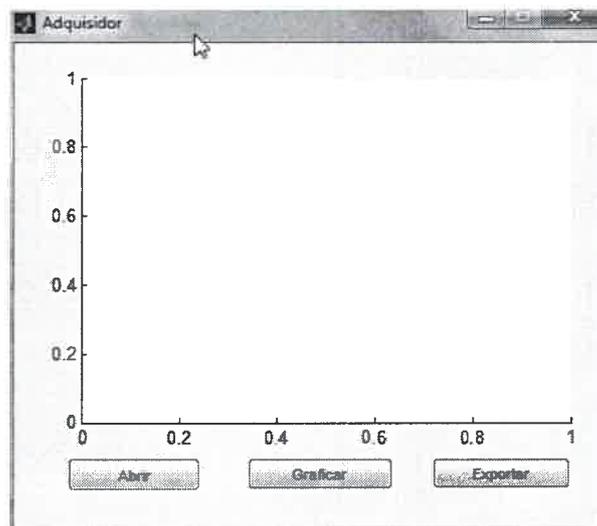


Figura A.3 - Módulo Adquisidor de Señales

Mediante este módulo el usuario puede incorporar señales almacenadas en una base de datos para su posterior análisis. Estas señales pueden haber sido obtenidas del muestreo de una señal real o bien pueden haber sido creadas por algún software con el propósito de contar con un conjunto de señales de prueba que permitan probar los algoritmos de detección y clasificación.

Este módulo ventana que sirve de interfaz entre el usuario y la base de datos donde se encuentra la información, permite al usuario incorporar datos, graficarlos o exportarlos en un archivo ASCII.

A.2.3 Módulo Generador de Señales

Si bien un sistema de monitoreo debe ser capaz de analizar formas de onda reales obtenidas del muestreo de un sistema eléctrico, para los fines de probar el rendimiento de los algoritmos de detección y clasificación de perturbaciones, es de vital importancia contar con una herramienta que permita simular los diferentes eventos que se pueden producir en una red eléctrica.

El objetivo del desarrollo de este módulo es el de permitir la generación de señales corruptas con diferentes tipos de perturbaciones permitiendo la configuración de los parámetros más importantes que gobiernan a cada tipo de perturbación.

Para acceder a este módulo se debe presionar la opción *Generador de Señales* del *Panel Principal* lo que provocará que emerja el cuadro de diálogos mostrado en la Figura A.4:

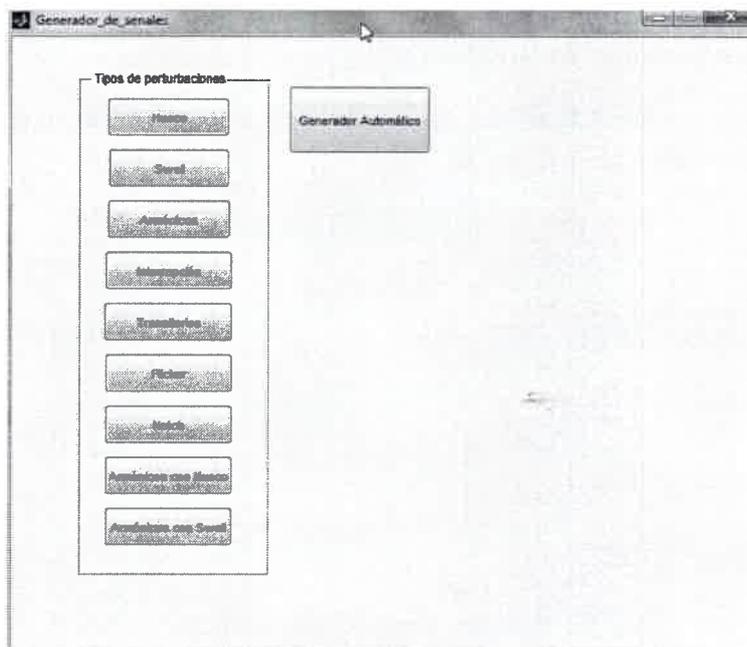


Figura A.4 - Generador de Señales.

El panel *Generador de Señales* presenta las siguientes funcionalidades:

- 1) Hueco
- 2) Swell
- 3) Armónicos
- 4) Interrupción

- 5) Transitorios
- 6) Flicker
- 7) NotchArmónicos con Huecos
- 8) Armónicos con Sag
- 9) Generador Automático

A.2.3.1 Generación de perturbaciones del tipo Huecos

Al seleccionar la opción Huecos en el panel Generador de Señales se despliega la interfaz mostrada en la Figura A.5.

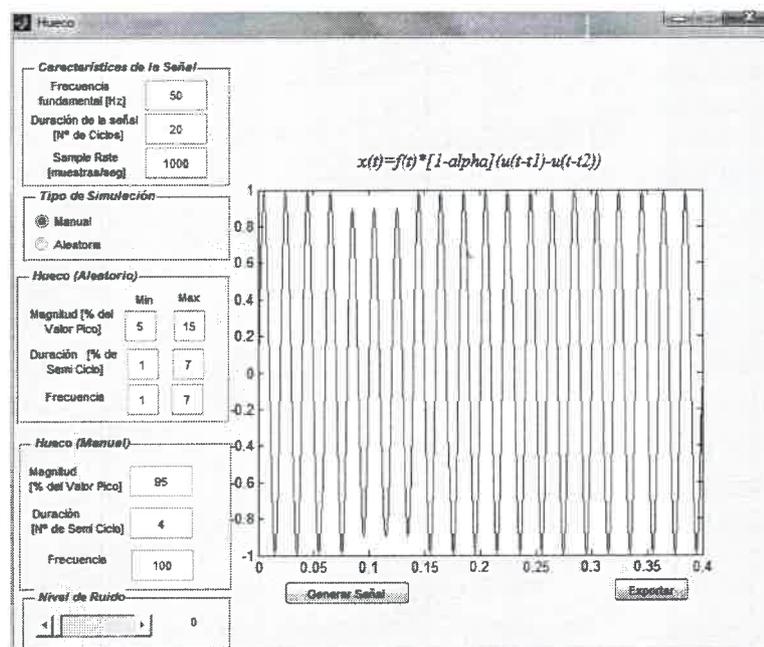


Figura A.5 - Panel de generación de Huecos

Esta interfaz gráfica permite el ingreso de varios parámetros característicos de una perturbación tipo Hueco, los cuales son detallados a continuación:

Sección Características de la señal: En esta sección el usuario puede ingresar las características relativas a la señal sinusoidal independientemente del tipo de perturbación con la cuál será distorsionada. Estas características son:

Frecuencia fundamental: Esta sección permite el ingreso de la variable frecuencia fundamental de la señal sinusoidal. Por defecto el panel está configurado con una frecuencia de 50 [Hz] ya que este valor es el empleado en los sistemas eléctricos de potencia en Argentina.

Duración de la señal: Esta variable se utiliza para determinar la longitud de la ventana de muestreo y se mide en cantidad de ciclos de la frecuencia fundamental.

Frecuencia de muestreo: Esta variable se utiliza para cargar la frecuencia de muestreo simulada. La misma indica la cantidad de muestras que se emplean en un segundo para generar la forma de onda perturbada.

Sección Tipo de Simulación: Esta sección contiene dos campos del tipo check box. Mediante estos campos se puede seleccionar si la generación de la señal se realiza de manera manual o de manera aleatoria.

Generación Manual: Si esta opción se selecciona la señal se generará de acuerdo a los parámetros cargados en la sección *Hueco (Manual)*.

Generación Aleatoria: Por el contrario, si esta opción se habilita, el software utilizará los parámetros cargados en la sección *Hueco (Aleatoria)*.

Sección Hueco (Aleatorio): En caso de haber sido seleccionado el check box Tipo de simulación Aleatorio, la señal se generará de acuerdo a los parámetros que se carguen en esta sección. Una perturbación tipo Hueco queda determinada por los siguientes dos parámetros:

Duración (Máxima/Mínima): En esta sección se pueden cargar los parámetros máximos y mínimos de la duración de la perturbación. Estos campos se miden en número de ciclos. Por ejemplo, si en el campo duración mínima se carga el número 3 y en el campo duración máxima se carga el número 7, el software generará de manera aleatoria una perturbación tipo Hueco cuya duración quedará delimitada dentro del intervalo 3 y 7 ciclos de la frecuencia fundamental.

Amplitud (Máxima/Mínima): En esta sección se pueden cargar los parámetros máximos y mínimos de la amplitud de la perturbación. Estos campos se miden en relación al valor pico de la señal sin perturbar (en porcentaje). Por ejemplo, si en el campo magnitud mínima se carga el número 5 y en el campo magnitud máxima se carga el número 15, el software generará de manera aleatoria una perturbación tipo Hueco cuya magnitud quedará delimitada entre un 5% y un 15% del valor nominal y cuya duración se corresponderá con lo cargado en la sección *Duración (máxima/mínima)*

Sección Hueco (Manual): En caso de haber sido seleccionado el check box Tipo de simulación Manual, la señal se generará de acuerdo a los parámetros que se carguen en esta sección.

Duración: En este campo se carga la duración de la perturbación. Se mide en cantidad de ciclos de la frecuencia fundamental.

Amplitud: La amplitud de la perturbación se generará de acuerdo al valor cargado en este campo. Se mide en porcentaje de decaimiento por sobre la señal sin perturbar.

Nivel de Ruido: Esta sección del panel permite incorporar ruido tiene las propiedades de tener una distribución gaussiana y de incorporarse a la señal de manera aditiva. La varianza del ruido se ajusta por medio de la barra deslizante denominada Nivel de Ruido.

Generar Señal: Este botón ejecuta la generación de la perturbación. Una vez que la perturbación se genera, la misma se grafica automáticamente en el recuadro mostrado en la Figura A.5.

Exportar Señal: Este botón permite la generación de un archivo ASCII el cual contendrá el vector señal y el vector tiempo correspondiente.

A.2.3.2 Generación de perturbaciones del tipo Swell

El panel empleado para la generación de una perturbación tipo Swell estructuralmente es similar al empleado para generar la perturbación tipo Huecos y, en líneas generales, presenta las mismas características que el resto de las interfaces destinadas a generar todas perturbaciones soportadas por esta herramienta. Por este motivo, en lo sucesivo, solo se detallarán aquellos aspectos que difieran de lo ya explicado en la sección anterior.

La Figura A.6 muestra la interfaz que se despliega si se selecciona la opción *Swell* en el panel de generación de señales.

Duración: Los campos relacionados con la duración, tanto para la generación manual como para la generación automática son utilizados para modelar la ventana de muestreo emulada. Este campo se mide en número de ciclos.

Amplitud: Los campos relacionados a la amplitud se emplean para cargar los parámetros relacionados con la magnitud de la perturbación. Estos campos se miden en relación al valor pico de la señal sin perturbar (en valor porcentual).

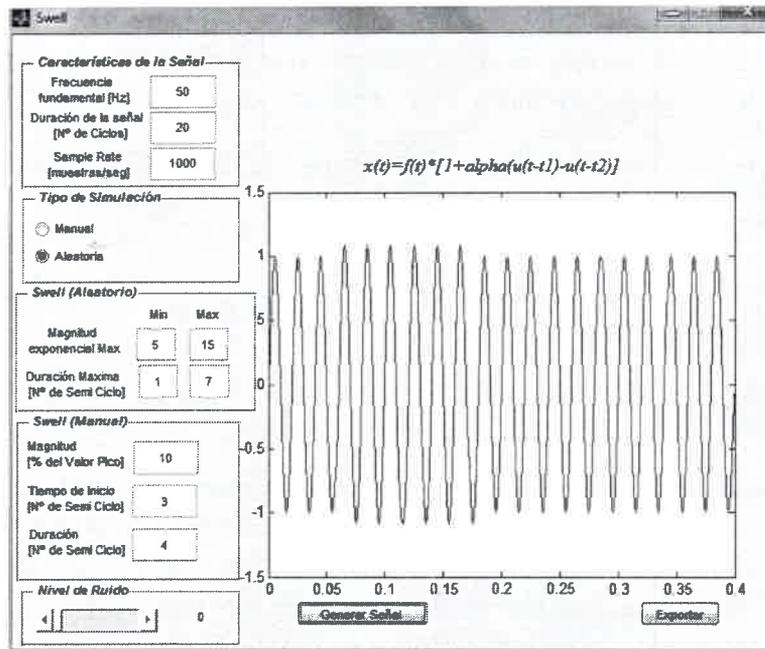


Figura A.6 - Panel de generación de Swell.

A.2.3.3 Generación de perturbaciones del tipo Armónicos

La Figura A.7 muestra la interfaz desarrollada para generar perturbaciones del tipo armónicos.

Una perturbación del tipo armónico puede ser vista como la superposición de una señal sinusoidal de una determinada frecuencia fundamental con una serie de señales sinusoidales, de menor amplitud, cuyas frecuencias se corresponden con un número entero de la frecuencia fundamental de la señal original. Los siguientes campos se emplean para modelar este tipo de perturbación:

Alpha 1-3-5-7: Estos campos, tanto para el tipo de generación manual como para el tipo de generación aleatoria, son empleados para configurar la magnitud de la segunda, tercera, quinta y séptima armónica. Su unidad de medida es en magnitud normalizada al valor nominal de una señal sinusoidal pura. Por ejemplo, si la variable alpha 2 se configura con un valor de 0.3 la señal perturbada tendrá componentes de segunda armónica con una magnitud de 0.3.

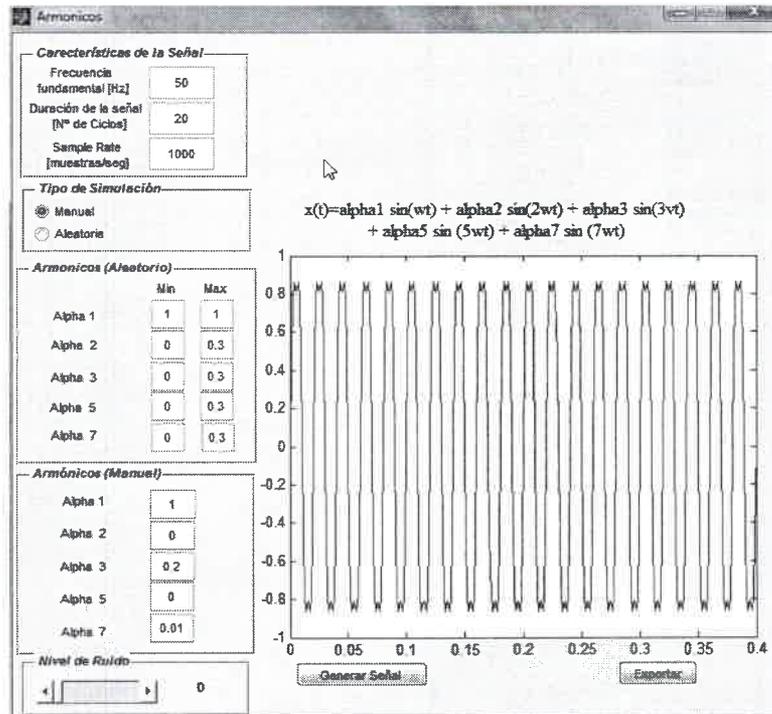


Figura A.7 - Generación de perturbaciones del tipo Armónicos.

A.2.3.4 Generación de perturbaciones del tipo Interrupción

De manera similar a la interfaz presentada para la generación de Huecos de tensión, en esta pantalla se pueden cargar parámetros relacionados con la magnitud y duración de la perturbación. La Figura A.8 muestra la interfaz desarrollada para la generación de eventos del tipo Interrupción.

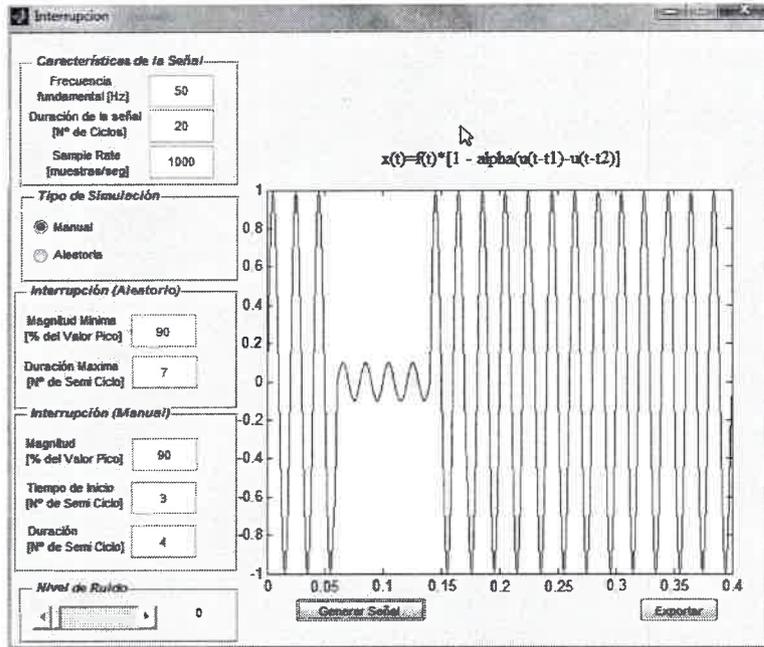


Figura A.8 - Generación de perturbaciones del tipo Interrupcion.

A.2.3.5 Generación de perturbaciones del tipo Transitorios

La Figura A.9 muestra la interfaz gráfica desarrollada para la generación de perturbaciones del tipo Transitorios Oscilatorios.

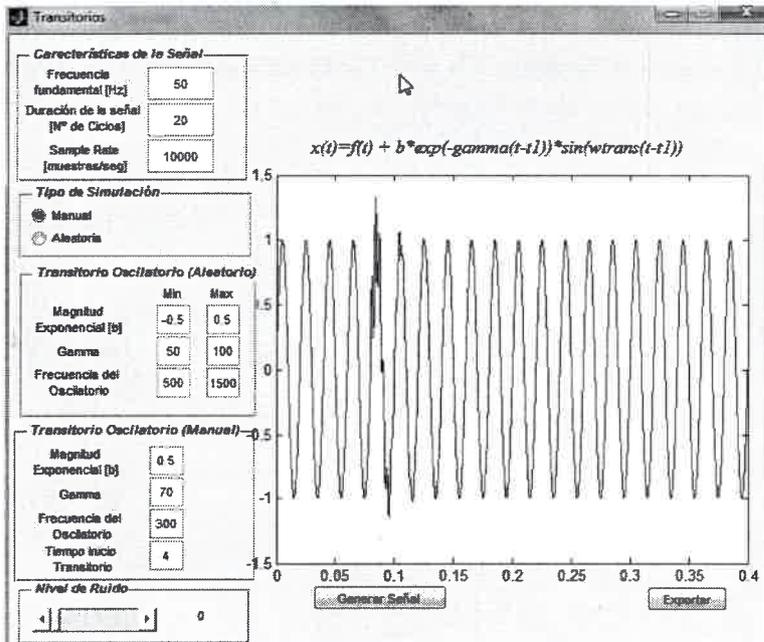


Figura A.9 - Generación de perturbaciones del tipo Transitorios.

Los parámetros característicos empleados para modelar este tipo de perturbaciones son:

Magnitud de la Exponencial [b]: Este campo, tanto para la generación manual como para la generación aleatoria, es empleado para configurar la magnitud de función exponencial empleada para modelar un transitorio.

Gamma: Los campos relacionados con el parámetro Gamma, tanto para la generación manual como para la generación aleatoria, se emplean para definir el parámetro de decaimiento de la exponencial empleada para modelar la perturbación.

Frecuencia del Oscilatorio: Este campo se emplea para modelar la frecuencia fundamental la perturbación. Su unidad es [Hertz].

A.2.3.6 Generación de perturbaciones del tipo Flicker

La Figura A.10 muestra la interfaz desarrollada para la generación de perturbaciones del tipo Flicker.

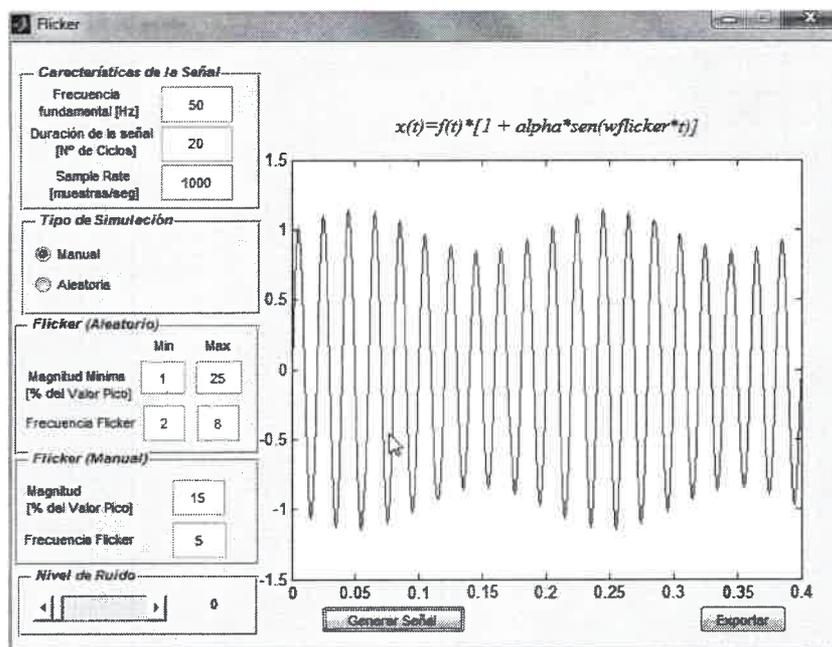


Figura A.10 - Generación de perturbaciones del tipo Flicker

Esta interfaz permite la configuración de los siguientes parámetros:

Magnitud del Flicker: Este campo está destinado a cargar el valor relativo de la magnitud del flicker respecto al valor nominal de la señal sin perturbar. Este campo está medido como un porcentaje de la relación de la amplitud del evento con respecto a la señal sin perturbar.

Frecuencia del Flicker: Este campo es empleado para cargar la frecuencia del evento. Su unidad de medida es el Hz.

A.2.3.7 Generación de perturbaciones del tipo Notch

La Figura A.11 ilustra la interfaz gráfica utilizada para la generación de perturbaciones del tipo Notch, también conocida como muesca.

Esta interfaz cuenta con los siguientes parámetros empleados para la generación de este tipo de perturbaciones:

Magnitud: Este campo es empleado para configurar la magnitud del notch. El mismo viene dado por un valor porcentual normalizado respecto al valor máximo de la señal sin perturbación.

Duración: Este campo es empleado para configurar la duración del evento y su unidad es micro segundo [ms].

Frecuencia: Mediante este campo se carga el valor la frecuencia de con que las muescas se repiten.

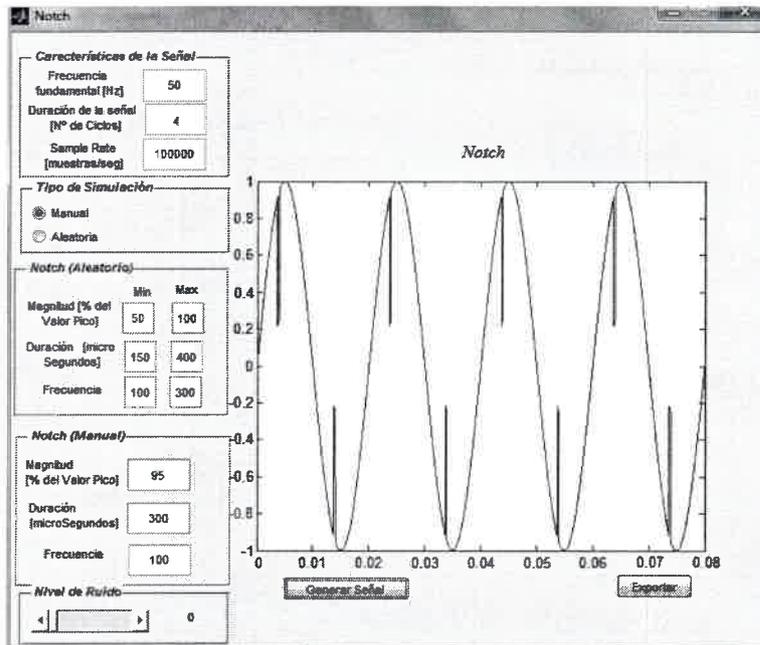


Figura A.11 - Generación de perturbaciones del tipo Notch.

A.2.3.8 Generación de perturbaciones del tipo Armónicos y Huecos

Esta interfaz es utilizada para generar una combinación de perturbaciones que usualmente suelen presentarse en sistema eléctrico. Tal es el caso de perturbaciones de tipo Armónicos combinadas con perturbaciones de tipo Huecos. Los parámetros que esta interfaz permite configurar son idénticos a los que se configuran en las interfaces destinadas a generar Huecos y Armónicos de manera independiente.

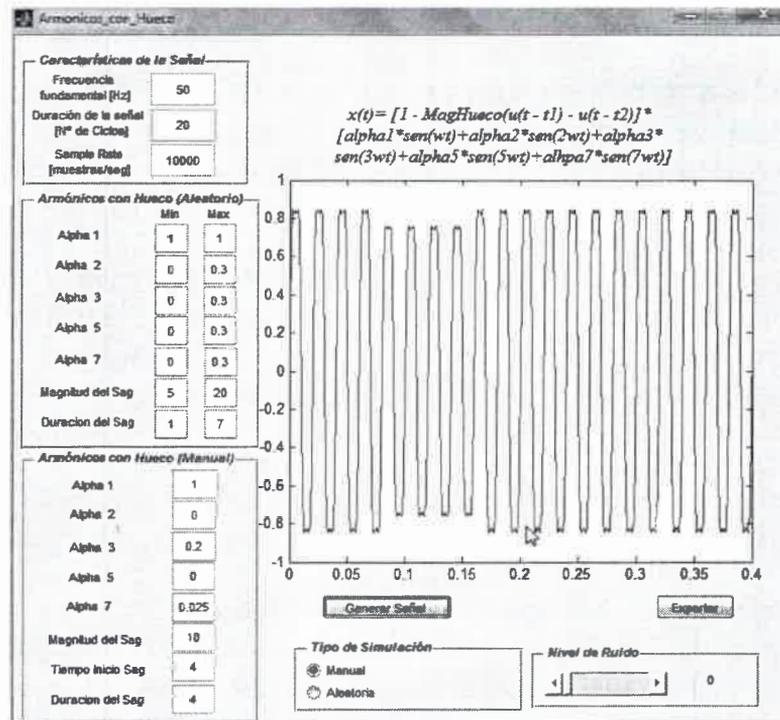


Figura A.12 - Generación de perturbaciones del tipo Armónicos con Huecos.

A.2.3.9 Generación de perturbaciones del tipo Armónicos y Swell

Al igual que la sección anterior, la combinación de Armónicos y Swell es bastante común en un sistema eléctrico de potencia. Esta interfaz se emplea para configurar este tipo de perturbación.

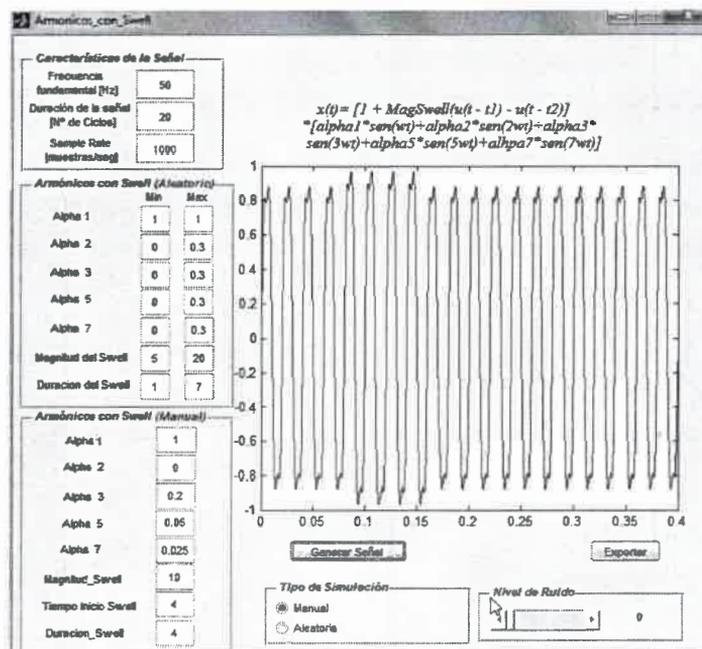


Figura A.13 - Generación de perturbaciones del tipo Armónicos con Swell.

A.2.4 Generador Automático de Señales

Las interfaces hasta ahora presentadas son de gran utilidad para estudiar cada fenómeno de manera individual sin embargo, para probar la eficiencia de los algoritmos de detección y clasificación, es de gran utilidad el contar con una base de datos de perturbaciones lo más grande y variada posible.

La tarea de generar individualmente cada tipo de perturbación puede resultar una tarea excesivamente lenta y agobiante. Por este motivo se desarrolló una herramienta para generar diversas perturbaciones en una única señal. La configuración de los parámetros de cada perturbación se realiza mediante la interfaz gráfica mostrada en la Figura A.14.

La mencionada interfaz gráfica permite indicar la cantidad de perturbaciones que se desean generar. Además permite modificar la frecuencia fundamental de la señal, la frecuencia de muestreo y la longitud de la ventana temporal utilizada en cada tipo de perturbación.

El software presenta, por cada tipo de evento de calidad de energía, una sección donde se encuentran todos los campos que gobiernan a la perturbación en cuestión. Cada una de estas secciones contiene un campo del tipo check box el cual, en caso de encontrarse seleccionado, incluirá la perturbación en la simulación de la señal.

Para generar la señal se configuran todos los campos correspondientes a las perturbaciones que se van incluir en la señal resultante, se seleccionan los correspondientes check box y luego se presiona el botón *Generar Señal*. Este botón genera un vector que contiene una la cantidad de señales perturbadas de acuerdo a lo configurado en el campo *Cantidad de Señales* incluyendo aquellas perturbaciones que hayan sido debidamente seleccionadas. La herramienta fue desarrollada para que la generación de cualquier tipo de perturbación tenga la misma probabilidad. Como resultado se obtiene un vector que contiene las muestras de diferentes perturbaciones generadas de manera aleatoria, cuyas características principales se ven regidas de acuerdo a parámetros también generados aleatoriamente aunque respetando los valores máximos y mínimos configurados en la interfaz gráfica.

El campo denominado *Señal de Entrenamiento* es empleado para generar una señal que posteriormente será empleada en distintos módulos del software para *entrenar* los algoritmos de clasificación. Si este campo no se encuentra seleccionado, la señal generada será interpretada como una señal de prueba.

Dependiendo de la opción seleccionada, la señal puede ser exportada a un archivo ASCII, con un formato CSV (cada muestra se separa con una coma) o formato el formato común en donde cada muestra es separada por un espacio.

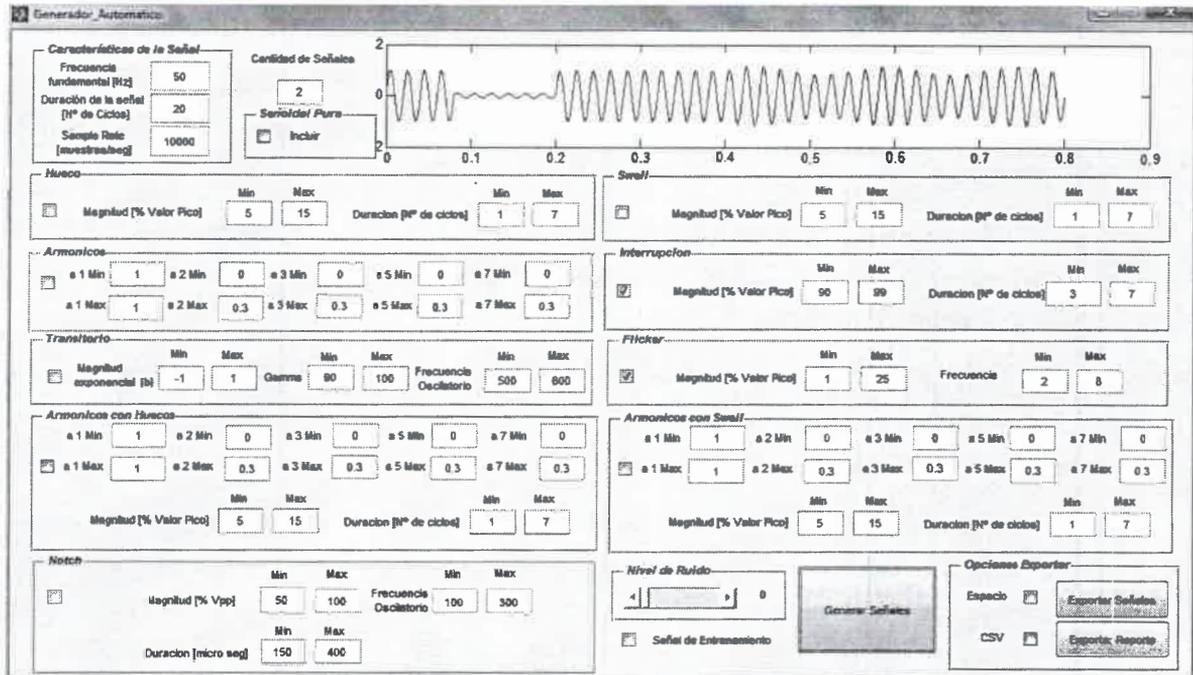


Figura A.14 – Generación Automático de perturbaciones.

A.2.5 Módulo de Detección de Perturbaciones

El acceso a la interfaz gráfica mostrada en la Figura A.15 se realiza seleccionando la opción *Transformada Wavelet* del *Panel Principal* presentado en la sección A.2

El principal objetivo de esta interfaz gráfica es la de permitir la ejecución de los algoritmos de detección de perturbaciones y extracción de parámetros característicos. (Nota: En el capítulo 5 se detalla el desarrollo de los algoritmos de detección y clasificación)

La interfaz gráfica desarrollada cuenta con las siguientes secciones:

La sección *Importación de Señal* permite el ingreso de parámetros de la señal importada los cuales resultan imprescindibles para el algoritmo de detección de eventos. Estos parámetros son:

Frecuencia fundamental: Permite indicar la frecuencia fundamental de la señal importada. La unidad de este campo es [ciclos/segundo] o [Herz].

Frecuencia de muestreo: Permite ingresar la frecuencia de muestreo de la señal. Se mide en [muestras/segundo].

Duración de la ventana de muestreo: Permite ingresar la duración de la ventana. Se mide en [cantidad de ciclos]

Cantidad de señales: Los datos importados muchas veces se corresponden con el muestreo sucesivo de una señal durante más de un intervalo de muestreo. El algoritmo desarrollado contempla la funcionalidad de generar un vector característico por cada uno de los

intervalos de muestreo. En este campo se configura la cantidad de ventanas de muestreo que el algoritmo de detección va a contemplar.

La sección Detección y Extracción de Características está conformado por las siguientes sub-secciones:

Vector Característico: ejecuta el algoritmo de detección y extracción parámetros característicos de la perturbación. La ejecución de este algoritmo automáticamente grafica el o los vector/es característico/s.

Tiempo Inicio/Fin: Para el caso de eventos de corta duración, la interfaz gráfica cuenta con una tabla donde se muestra el tiempo de inicio y el tiempo de fin (relativo al inicio de la ventana de muestreo) de cada perturbación.

Señal de Entrenamiento: La herramienta desarrollada para la clasificación de patrones emplea algoritmos de aprendizaje supervisado los cuales, antes de ser utilizados, deben ser previamente entrenados. Con el objetivo de distinguir entre los vectores característicos empleados para el entrenamiento de aquellos vectores utilizados para el algoritmo de clasificación, la interfaz gráfica presenta un campo del tipo check box denominado Señal de Entrenamiento que permite indicar si los vectores característicos generados serán empleados para el entrenamiento de los algoritmos de clasificación o bien serán utilizados como vectores de prueba.

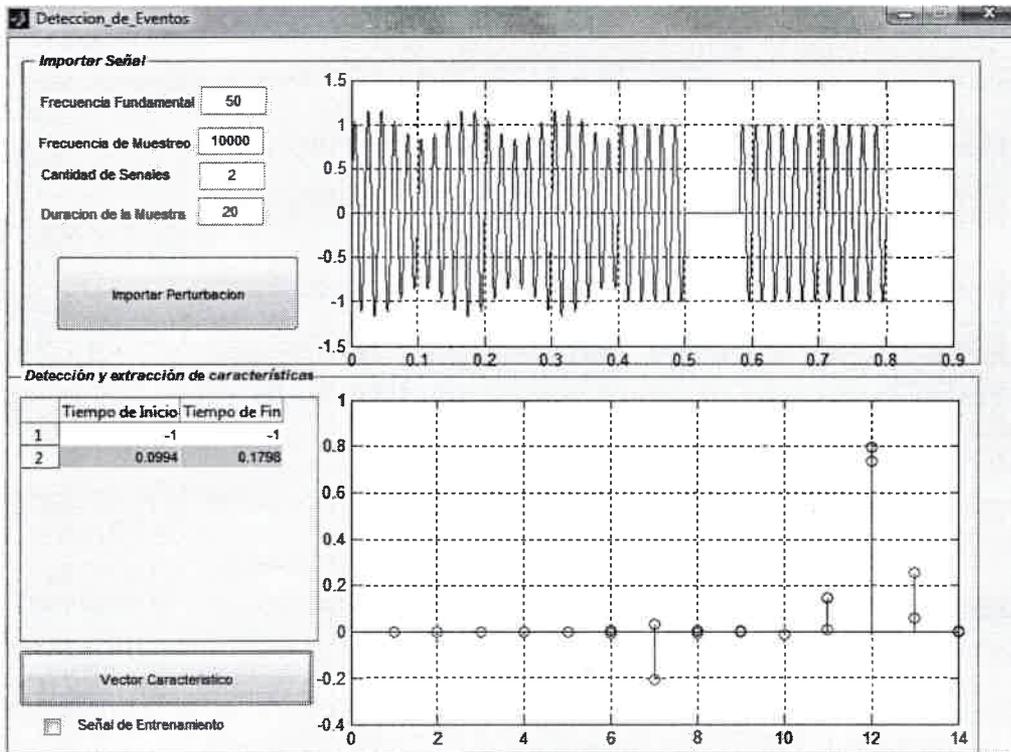


Figura A.15 - Módulo de detección y extracción de parámetros característicos.

A.2.6 Módulo de Clasificación de Perturbaciones

El módulo de clasificación de perturbaciones fue desarrollado con el objetivo de permitir la configuración, entrenamiento y prueba de algoritmos de clasificación basados en métodos de aprendizaje supervisado.

A través de este módulo se obtiene una red neuronal entrenada la cual se podrá exportar y estará en condiciones de ser utilizada para la clasificación de patrones de perturbaciones.

El acceso a este módulo de *Clasificación de perturbaciones* se realiza a través del *Panel Principal* explicado en la sección A.2. La figura muestra que la herramienta cuenta con dos algoritmos de clasificación de patrones

- 1) Red Neuronal Back Propagation
- 2) Support Vector Machine

A.2.6.1 Red Neuronal Back Propagation

El objetivo de esta interfaz gráfica mostrada en la Figura A.16 es el de permitir la configuración de los principales parámetros que gobiernan el entrenamiento de una red neuronal back propagation de dos capas ocultas.

Para acceder a este módulo se debe presionar la opción *NNA BackPropagation* del *Panel Principal*.

The screenshot displays the 'BackPropagation' window, which is divided into several sections for configuring and testing a neural network. The 'Configuracion de la Red' section includes input fields for 'Neuronas capa Oculta 1' (12), 'Neuronas capa Oculta 2' (10), 'Tasa Aprendizaje' (0.3), 'Epocas' (1000), and 'Error Máximo' (1e-3), with an 'Entrenar' button below. The 'Resultado de la Prueba' section shows 'Cantidad de Señales' (500), 'Clasificadas Erroneamente' (35), and 'Porcentaje Aciertos' (93), with a 'Prueba' button. The 'Resultado del Entrenamiento' section at the bottom shows 'Epocas' (32), 'Error Máximo' (0.000916145), and 'Tiempo de Entrenamiento' (16.161), with an 'Exportar' button.

Configuracion de la Red	
Neuronas capa Oculta 1	12
Neuronas capa Oculta 2	10
Tasa Aprendizaje	0.3
Epocas	1000
Error Máximo	1e-3

Resultado de la Prueba	
Cantidad de Señales	500
Clasificadas Erroneamente	35
Porcentaje Aciertos	93

Resultado del Entrenamiento	
Epocas	32
Error Máximo	0.000916145
Tiempo de Entrenamiento	16.161

Figura A.16 - Módulo de clasificación basado en una NNA Backpropagation.

La interfaz gráfica está conformada por tres secciones:

Configuración de la Red: Esta sección de la interfaz gráfica permite la configuración de parámetros que gobiernan la arquitectura y entrenamiento de la red neuronal. Los parámetros que se podrán configurar mediante esta interfaz gráfica son:

Neuronas de la capa oculta 1: Cantidad de neuronas de la primera capa oculta.

Neuronas de la capa oculta 2: Cantidad de neuronas de la segunda capa oculta.

Tasa de Aprendizaje: Mediante este campo se puede ingresar la tasa de aprendizaje de la red neuronal. Este parámetro es empleado por el algoritmo de entrenamiento y representa una medida de la velocidad con que los pesos de la red neuronal serán ajustados.

Épocas: En este campo se configura la cantidad máxima de veces que el set de entrenamiento completo se utiliza para el entrenamiento de la red. Si el entrenamiento alcanza el *error máximo* antes de cumplir con lo configurado en este campo, el entrenamiento de la red culmina indicando que la misma alcanzó el error mínimo permitido.

Error Máximo: En este campo se configura el error mínimo permitido para la salida de la red. Este parámetro es una medida de la performance del entrenamiento que la red alcanza. La salida de la red, ante una determinada entrada, no diferirá de la salida esperada por un valor superior a lo configurado en este campo.

Entrenar: Este botón ejecuta el algoritmo de entrenamiento. El software utiliza por defecto la señal del tipo Señal de Entrenamiento creada por el módulo *Generador Automático* presentado (sección 0) la cual debe ser procesada por el módulo de *Detección de perturbaciones* (sección 0).

Resultado del Entrenamiento: Esta sección es poblada de manera automática por el algoritmo de entrenamiento de la red neuronal. Los siguientes campos ilustran el resultado del entrenamiento:

Épocas: Este campo indica la cantidad de épocas empleada por el algoritmo de entrenamiento. Si el valor de este campo se corresponde con el valor configurado en la sección *Entrenamiento de la red* la red no habrá alcanzado el error mínimo esperado.

Error Máximo: En este campo se podrá verificar el error máximo alcanzado en el entrenamiento. Este campo indica la precisión con que la red clasificará los patrones de entrada.

Tiempo de Entrenamiento: En este campo se visualiza el tiempo empleado para entrenar la red.

Resultado de la Prueba: Esta sección de la interfaz gráfica presenta los resultados de la prueba de la red neuronal.

Cantidad de Señales: Este campo indica la cantidad de señales que conforman el set de prueba

Clasificadas Correctamente: El algoritmo de prueba de la red neuronal utiliza un set de pruebas para comparar los resultados obtenidos con los resultados esperados. En este campo el algoritmo presenta la cantidad de clasificaciones exitosas.

Porcentaje de Aciertos: Este campo indica, en valor porcentual, la efectividad de la red neuronal para clasificar perturbaciones.

Prueba: Este botón inicia el algoritmo de prueba de la red neuronal. La señal de prueba utilizada se corresponde con una señal creada por el módulo *Generador Automático* presentado (sección 0) la cual debe ser procesada por el módulo de *Detección de perturbaciones* (sección 0) teniendo en cuenta que el campo check box *Señal de Entrenamiento* debe estar desmarcado.

Exportar: Una vez que se alcanza un porcentaje de aciertos deseado es útil que la red pueda ser exportada para ser empleada en la clasificación de señales reales. Por este motivo la herramienta cuenta con una funcionalidad que permite exportar una estructura que contiene los parámetros de la red ya entrenada. La estructura se exporta con el nombre *parametrosRed.net* que, por defecto, será almacenada en el directorio donde se encuentra ubicado el software.

A.2.6.2 Support Vector Machine

La interfaz gráfica mostrada en la Figura A.17 fue desarrollada permitir la configuración de los parámetros que gobiernan el entrenamiento de una red compuesta por Maquinas de Soporte Vectorial (SVM) que, al igual que una red neuronal, su objetivo es clasificar las perturbaciones según sus parámetros característicos.

La interfaz gráfica está compuesta por las siguientes secciones:

Configuración de los SVM: En esta sección se configura los parámetros que gobiernan el entrenamiento de los SVM especializados en clasificar cada una de las perturbaciones.

RBF Sigma: Parámetro relacionado con la función Kernel denominada Gaussian Radial Basis utilizada por defecto.

Box Constraint: Este parámetro se conoce como margen blando (soft margin) en el entrenamiento de los SVM.

Tiempo de Entrenamiento: En este campo se visualiza el tiempo empleado para entrenar la red de SVM.

Resultados de la Prueba

Cantidad de Señales: Este campo indica la cantidad de señales que conforman el set de prueba.

Clasificadas Correctamente: El algoritmo de prueba utiliza un set de pruebas para comparar los resultados obtenidos con los valores esperados. En este campo el algoritmo presenta la cantidad de clasificaciones exitosas.

Porcentaje de Aciertos: Este campo indica, en valor porcentual, la efectividad de la red de SVM para clasificar perturbaciones.

Prueba: Este botón inicia el algoritmo de prueba de la red de SVM. La señal de prueba utilizada se corresponde con una señal creada por el módulo *Generador Automático* presentado (sección 0) la cual debe ser procesada por el módulo de *Detección de perturbaciones* (sección 0) teniendo en cuenta que el campo check box *Señal de Entrenamiento* debe estar desmarcado.

Exportar: Una vez que se alcanza un porcentaje de aciertos deseado es útil que la red pueda ser exportada para ser empleada en la clasificación de señales reales. Por este motivo la herramienta cuenta con una funcionalidad que permite exportar una estructura que contiene los parámetros de la red ya entrenada. La estructura se exporta con el nombre *parametrosRed.net* que, por defecto, será almacenada en el directorio donde se encuentra ubicado el software.

Configuración de los SVM		
	RBF Sigma	Box Constraint
SVM Huecos	0.5	3
SVM Swell	0.3	3
SVM Armónicos	1	7
SVM Interrupcion	1.3	7
SVM Transitorios	4.5	8
SVM Flicker	0.5	5
SVM Notch	1	7
SVM Armonicos y Huecos	5	3
SVM Armonicos y Swell	5	10

Tiempo de Entrenamiento: 3.01693

Resultado de la Prueba	
Cantidad de Señales	500
Clasificadas Correctamente	407
Porcentaje Aciertos	81.4

Figura A.17 - Clasificador basado en Maquinas de Soporte Vectorial.

A.2.7 Módulo de Clasificación de Perturbaciones

Este módulo fue desarrollado para simular el comportamiento de los algoritmos de detección y clasificación presentados en las secciones anteriores

Para acceder a este módulo se debe configurar el directorio raíz donde se encuentran los scripts del software y, como se muestra en la Figura A.18, se ejecutar la siguiente instrucción: *Simulador*

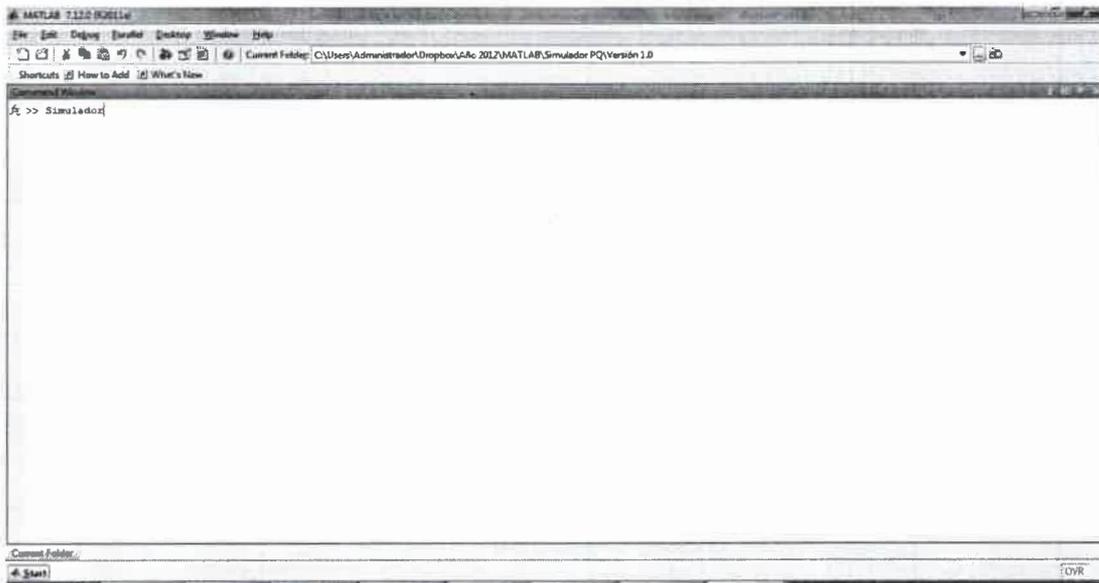


Figura A.18 - Acceso a la interfaz gráfica del Simulador.

La interfaz gráfica desarrollada para la simulación del comportamiento de los algoritmos de clasificación se muestra en la Figura A.19.

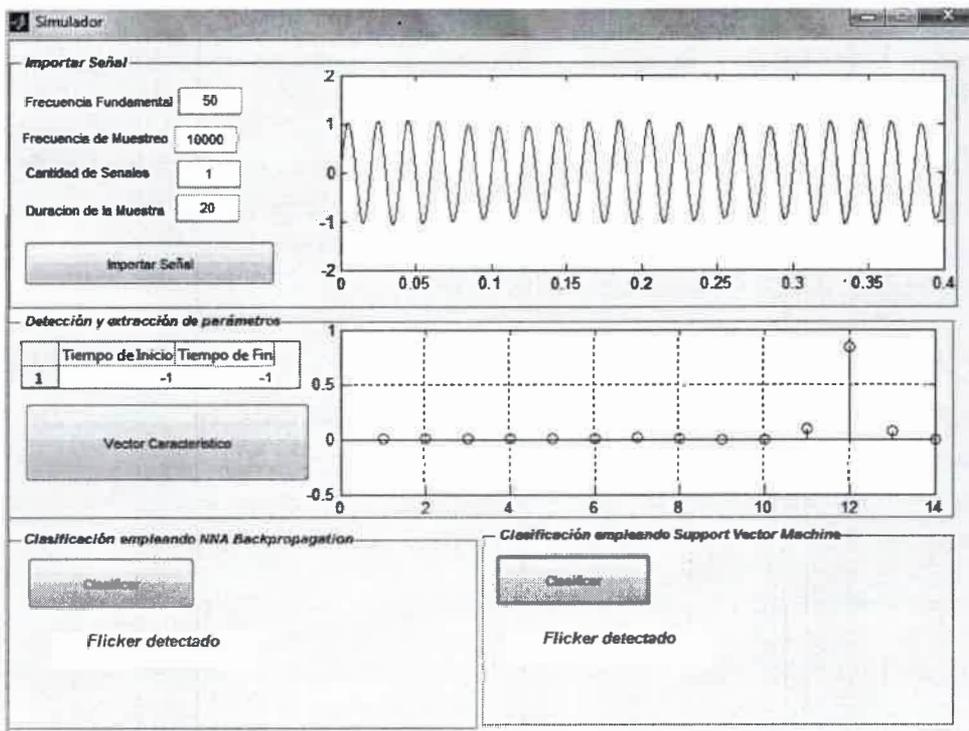


Figura A.19 - Interfaz grafica del clasificador.

Esta interfaz está dividida en cuatro secciones:

Importar Señal

Frecuencia fundamental: Permite indicar la frecuencia fundamental de la señal importada. La unidad de este ampo es [ciclos/segundo] o [Herz].

Frecuencia de muestreo: Permite ingresar la frecuencia de muestreo de la señal. Se mide en [muestras/segundo].

Duración de la ventana de muestreo: Permite ingresar la duración de la ventana. Se mide en [cantidad de ciclos].

Cantidad de señales: Los datos importados muchas veces se corresponden con el muestreo sucesivo de una señal durante más de un intervalo de muestreo. El algoritmo desarrollado contempla la funcionalidad de generar un vector característico por cada uno de los intervalos de muestreo. En este campo se configura la cantidad de ventanas de muestreo que el algoritmo de detección va a contemplar

Importar Señal: Esta opción despliega un cuadro de diálogo que permite buscar el archivo de texto que contiene la señal a analizar. Una vez importada, la señal se grafica de manera automática.

Detección y extracción de parámetros

Vector característico: Este botón ejecuta el algoritmo de detección y extracción de parámetros característicos de la señal importada. El algoritmo genera y grafica el vector característico correspondiente a la señal bajo análisis.

Tiempo Inicio/Fin: En caso que la señal importada contenga una perturbación de corta duración el algoritmo identifica el tiempo de inicio y de fin del evento y lo muestra en estos campos de la interfaz gráfica. Si la perturbación detectada fuera del tipo estacionaria el algoritmo configura estos campos con el valor -1.

Clasificación empleando NNA Back Propagation.

Clasificar: Esta función ejecuta el algoritmo de clasificación basado en la NNA Backpropagation. Este algoritmo utiliza la estructura de la Red Neuronal entrenada por el módulo de clasificación anteriormente mostrado para realizar la clasificación de la señal de bajo análisis.

Clasificación empleando Support Vector Machine.

Clasificar: Esta función ejecuta el algoritmo de clasificación basado en la SVM. Este algoritmo utiliza las 9 estructuras de los SVM entrenados por el módulo de clasificación anteriormente mostrado para realizar la clasificación de la señal de bajo análisis.

ANEXO B
PUBLICACIONES REALIZADAS

- [1] "Educational Software for Power Quality Analysis", D. De Yong, C. Reineri, F Magnago, IEEE Latin America Transactions, Vol. 11, No. 1, pp. 479-485, Febrero 2013.
- [2] "Sistema de Detección de Eventos de Calidad de Energía Basado en Máquinas de Vectores de Soporte", D. DeYong, C. Reineri, F. Magnago, ERIAC XV, Foz de Iguazú, Mayo 2013.
- [3] "Utilización de Redes Auto-Expansibles para una Red Inteligente Eléctrica", F. Aguirre, F. Magnago, D. DeYong, E. Tardivo, ERIAC XV, Foz de Iguazú, Mayo 2013.
- [4] "Software Educativo para en Análisis Eventos de Calidad de Energía Eléctrica", ", D. DeYong, C. Reineri, F. Magnago, Argencon2012, Córdoba, Junio 2012.

73210

U.N.R.C
Biblioteca Central



73210