



Universidad de las Ciencias Informáticas
“Facultad 4”

Título: “Propuesta de algoritmo para la detección y clasificación de perturbaciones eléctricas en Sistemas SCADA”

**Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas**

Autor: Yailyn Crespo León

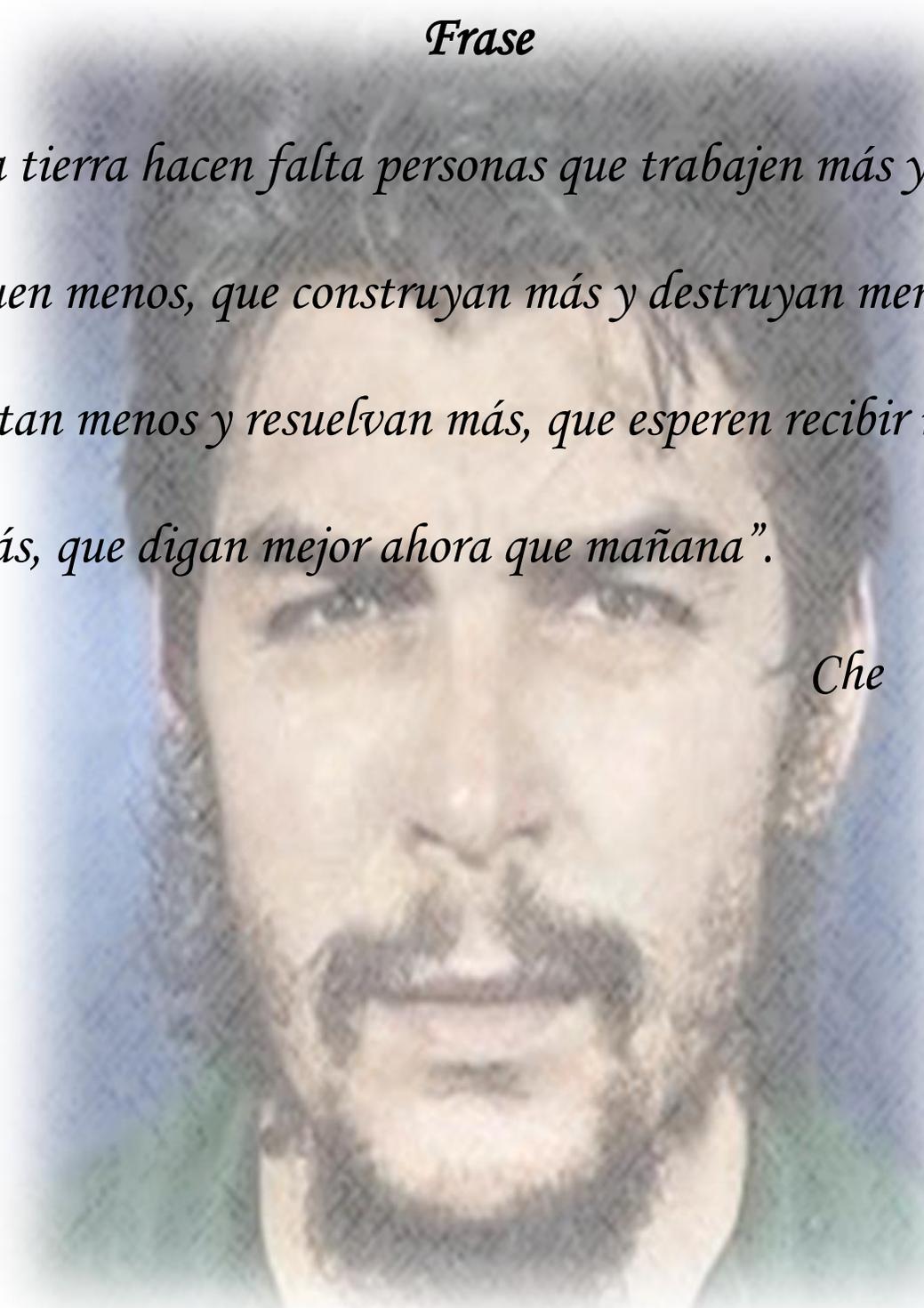
Tutores: MSc. Miguel Angel Socorro Borges
MSc. Andy Hernández Paez

Co-tutor: MSc. Yadira Ramírez Rodríguez

“Año 60 de la Revolución”

La Habana, Cuba

Junio 2018

A portrait of Che Guevara, a man with a beard and mustache, looking directly at the camera. The image is slightly faded and serves as a background for the text.

Frase

“En la tierra hacen falta personas que trabajen más y critiquen menos, que construyan más y destruyan menos, que prometan menos y resuelvan más, que esperen recibir menos y dar más, que digan mejor ahora que mañana”.

Che

DECLARACIÓN DE AUTORÍA

Declaro ser el único autor de la presente tesis que tiene por título: “Propuesta de algoritmo para la detección y clasificación de perturbaciones eléctricas en Sistemas SCADA” y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Yailyn Crespo León
Autor

MSc. Miguel Angel Socorro Borges
Tutor

MSc. Andy Hernández Paez
Tutor

DATOS DEL CONTACTO

MSc. Miguel Angel Socorro Borges: Ingeniero en Ciencias Informáticas (2012), graduado en la Universidad de las Ciencias Informáticas (UCI), más de seis años de experiencia como desarrollador de software. Profesor Instructor de la asignatura Programación 4. Máster en Ciencias Matemáticas (2017), graduado en la Universidad de La Habana. Instituto actual: Centro de Informática Industrial (CEDIN), Facultad 4, UCI.

E-mail: miguelsb@uci.cu

MSc. Andy Hernández Paez: Ingeniero en Ciencias Informáticas (2012), graduado en la Universidad de las Ciencias Informáticas (UCI), siete años de experiencia en las áreas del conocimiento de Ingeniería, Gestión y Calidad de Software en proyectos informáticos, en procesos de desarrollo web y de videojuegos. Profesor Asistente de la disciplina de Ingeniería y Gestión de Software. Máster en Ciencias (2017), graduado de la IV Edición de la Maestría en Calidad de Software de la UCI. Institución actual: Centro de Entornos Interactivos 3D, Vertex, Facultad 4, UCI.

E-mail: andyhp@uci.cu

DEDICATORIA

En especial a mi mamá Jacqueline por apoyarme en todo momento. Por darme fuerzas para continuar y sobre todo por creer en mí.

A mi abuelo Ricardo que aunque ya no está físicamente, estaría muy orgulloso de este logro.

AGRADECIMIENTOS

A mi mamá Jacqueline, gracias por estar siempre a mi lado, por alentarme en cada paso que doy y ser tan fuerte pese a las dificultades.

A mi padrastro Reinaldo, por quererme como una hija y ocuparse de mí desde pequeña. Por estar junto a mi madre en todo momento.

A mis tutores Miguel, Andy y Yadira, sin ustedes no habría sido posible la realización de este trabajo. Gracias por su tiempo y apoyo.

A mi novio Miguel Angel, por apoyarme, malcriarme y sobre todo por tener mucha paciencia.

A mis compañeros de aula, en especial a Las Patipolvas (Dalquerine, Jesslie, Shirley, Dailen y Dianne) y a Los Coyotes (Yasiel y David). Gracias por estos años maravillosos llenos de locura.

A todos los profesores que de alguna manera contribuyeron a mi formación profesional, en especial al profe Luis Eduardo por brindarme su amistad y al profe Andy que a pesar de su rectitud, supo ganarse el respeto y cariño de todos sus alumnos.

RESUMEN

La detección y clasificación de perturbaciones eléctricas es un tema ampliamente tratado en el área del procesamiento digital de señales y en los últimos años ha venido ganando popularidad el desarrollo de diferentes algoritmos que involucran nuevas herramientas matemáticas con este fin. El objetivo de esta investigación es implementar un algoritmo para la detección y clasificación de perturbaciones eléctricas, en sistemas *SCADA*, haciendo uso de la Transformada Wavelet Discreta y el método de clasificación supervisada *k-NN*. Este algoritmo procesa la señal en el espacio bidimensional tiempo-frecuencia y mediante la extracción y análisis de la energía de la señal es capaz de detectar y clasificar las perturbaciones presentes en la misma. El nivel de precisión del algoritmo se valida a través de las métricas: precisión de la clasificación y matriz de confusión, además se simula su comportamiento en un ambiente real mediante un prototipo funcional. Los resultados muestran que el algoritmo propuesto arroja un mayor porcentaje de exactitud en la clasificación que algoritmos similares propuestos en la literatura.

PALABRAS CLAVE

Detección y clasificación de perturbaciones, Energía, *k-NN*, *SCADA*, Transformada Wavelet Discreta.

ÍNDICE

RESUMEN	VI
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	5
Introducción.....	5
1.1 Sistemas SCADA.....	5
1.2 SCADA en sistemas de energía eléctrica.....	6
1.2.1 Análisis y valoración de los sistemas mencionados	8
1.3 Perturbaciones eléctricas	8
1.4 Identificación de perturbaciones eléctricas	9
1.4.1 Procesamiento de señales eléctricas	10
1.4.2 Extracción y optimización de características	14
1.4.3 Selección de la topología o arquitectura de clasificación.....	17
1.4.4 Realización del aprendizaje automático	20
1.4.5 Validación y prueba	20
1.4.6 Clasificación.....	21
1.5 Herramientas y tecnologías.....	21
1.5.1 QtCreator	21
1.5.2 Lenguaje de programación C++	22
1.5.3 Biblioteca QCustomPlot	22
1.5.4 Biblioteca Wavelet 1D/2D	22
1.5.5 Software MATLAB	23
Conclusiones parciales	23

CAPÍTULO 2: DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN	24
Introducción.....	24
2.1 Digitalización de la señal de voltaje.....	24
2.2 Etapa 1. Procesamiento de la señal.....	25
2.2.1 Selección de la herramienta de procesamiento de la señal.....	25
2.2.2 Detección de las perturbaciones.....	26
2.3 Etapa 2. Extracción y optimización de características.....	27
2.3.1 Selección de las características de la señal	27
2.3.2 Extracción de la energía de la señal utilizando TWD.....	28
2.3.3 Extracción de vectores de energía para la clasificación	28
2.4 Etapa 3. Selección de la topología o arquitectura de clasificación	29
2.5 Etapa 4. Realización del aprendizaje automático.....	30
2.6 Etapa 5. Validación y prueba	30
2.6.1 Estimación de los parámetros γ y k para el algoritmo k -NN.....	30
2.7 Propuesta del algoritmo	31
Conclusiones parciales	35
CAPÍTULO 3: EVALUACIÓN DE LA SOLUCIÓN PROPUESTA	36
Introducción.....	36
3.1 Validación del algoritmo basado en métricas	36
3.1.1 Precisión de clasificación.....	36
3.1.2 Matriz de confusión.....	37
3.2 Validación del algoritmo con prototipos.....	38
3.2.1 Propósito de los prototipos	38

3.2.2 Clases de prototipos	39
3.2.3 Prototipo de características seleccionadas.....	39
3.3 Pruebas del algoritmo utilizando el prototipo.....	40
3.4 Análisis de la validación del algoritmo.....	41
Conclusiones parciales	42
CONCLUSIONES	43
Referencias bibliográficas	45
Anexos	50
<i>Anexo 1: Principales productos SCADA comercializados en España.</i>	50
<i>Anexo 2: Esquema de una instalación de Circutor.</i>	51
<i>Anexo 3: Esquema topológico del sistema “Control Vision”.</i>	51
<i>Anexo 4: Arquitectura para una red de distribución para Network Manager de ABB.</i>	52

ÍNDICE DE FIGURAS

Figura 1: Ventana deslizante.	11
Figura 2: Wavelets madres más utilizadas.	13
Figura 3: Análisis Multi-Resolución a tres escalas para una señal (S).	14
Figura 4: Ejemplo de características: a) no separables linealmente, b) separables linealmente. .	16
Figura 5: Técnica de Validación Cruzada.	20
Figura 6: Digitalización de una señal de voltaje.	24
Figura 7: Salida wavelet de los CD, a) Señal ideal, b) Señal perturbada 1.....	27
Figura 8: Salida wavelet de los CD, a) Señal perturbada 2, b) Señal perturbada 3.	27
Figura 9: Ventana deslizante.	32
Figura 10: Diagrama de flujo del algoritmo.	33
Figura 11: Seudocódigo del algoritmo.	34
Figura 12: Prototipo de características seleccionadas.....	40

ÍNDICE DE TABLAS

Tabla 1: Kernels más difundidos en la literatura.	17
Tabla 2: Modelos de ecuaciones paramétricas para perturbaciones eléctricas.	30
Tabla 3: Matriz de confusión	38

INTRODUCCIÓN

Durante los últimos años la reducción del consumo de energía eléctrica ha sido objeto de cuidadosa atención, debido a los altos costos de los combustibles que se utilizan para su generación. Los encargados de generarla se denominan sistemas de energía eléctrica y constituyen la columna vertebral de la civilización técnica contemporánea (Rebizant, Szafran, & Wiszniewski, 2011). Su uso a nivel mundial se ha globalizado tanto que es difícil imaginar el desarrollo de las labores diarias, el funcionamiento de las industrias e incluso el avance de la economía mundial, sin el suministro estable e ininterrumpible de la energía eléctrica.

El suministro de energía no está ajeno de perturbaciones que afectan su calidad, provocadas por la influencia de factores como la simple conexión o desconexión de equipos, cortos circuitos en la red, desastres climáticos, entre otros, y por tal razón se debe supervisar de manera constante. Esta acción se realiza a través de los denominados Sistemas de Supervisión, Control y Adquisición de Datos (*SCADA* o *Supervisory Control And Data Acquisition*, por sus siglas en inglés) que permiten una evaluación automatizada sobre el correcto funcionamiento de los procesos, incluso en situaciones anómalas (Colomer, Meléndez, & Ayza, 2003). Entre las funcionalidades básicas de un *SCADA* se destacan: adquisición y almacenado de datos, representación gráfica y animada de variables de proceso, monitorización de estas por medio de alarmas y control remoto de dispositivos de campo. Pueden utilizarse en disímiles escenarios como oleoductos, yacimientos de gas y petróleo, redes de distribución de gas natural y **sistemas de transmisión de energía eléctrica**.

La implementación de estos sistemas de supervisión según lo expuesto en (Bárzaga, Mompie, & Valdés, 2016) supone recorrer tres etapas fundamentales: la detección de fallos, el diagnóstico de estos y finalmente la reconfiguración del sistema, que debe permitir continuar operando según las especificaciones fijadas. En caso de no realizarse las tres etapas anteriores, se considera el sistema de supervisión como sistema de vigilancia y asistencia al operario, se concibe entonces como un sistema de monitorización.

Una revisión detenida sobre productos *SCADA* que se encontraban en el mercado internacional en el año 2003 fue realizada en (Colomer, Meléndez, & Ayza, 2003) (Ver anexo 1), y permite afirmar que la mayoría de ellos aunque cumplían perfectamente con las tareas de adquisición de datos y control, distaban de ser entornos de supervisión al no contar con las etapas de detección de fallos, diagnóstico de estos y reconfiguración del sistema. El rol supervisor, en estos casos, quedaba relegado a la activación y registro de alarmas, que aunque estas pueden considerarse una forma simple de detección de fallos, faltaban por

cubrir otras tareas de la supervisión como son el diagnóstico y la propuesta de acciones para reconfiguración del sistema, en caso de ser necesario.

Los resultados obtenidos en (Lizarraga de Miguel, 2009) son un poco más recientes y muestran que algunos sistemas *SCADA* empleados específicamente en redes eléctricas, se limitan solo a realizar tareas como: control de armónicos, lectura en tiempo real de datos, generación de históricos, control del consumo general de las instalaciones y adquisición de parámetros energéticos (Ver epígrafe 1.2).

Por otra parte en Cuba, a partir de la entrada al país de modernos metrocontadores digitales, también se han creado sistemas *SCADA* para el control eléctrico. En el año 2009 la Empresa de Ingeniería y Proyectos para la Electricidad (INEL), creó el sistema de Adquisición Automática de Parámetros Eléctricos, Supervisión y Facturación (ASPELFAC). De las prestaciones que brinda, la principal es la adquisición de los parámetros medidos por el metro, tales como, potencia, demandas y consumo (Rodríguez, 2010). Se ofrece información de otro ejemplo en el sitio web de la empresa cubana “Copextel”, el Sistema Automatizado de Supervisión de Parámetros Energéticos, que ya se aplica en instalaciones hoteleras de la Isla con excelentes resultados. Esta herramienta se utiliza en complejos hoteleros o partes del mismo para el monitoreo y análisis del consumo de energía, la adquisición de datos y el análisis de históricos de los indicadores energéticos. También en la Universidad de las Ciencias Informáticas (UCI), específicamente en el Centro de Informática Industrial (CEDIN) se desarrolló el sistema *SCADA* SAINUX 1.0. El mismo se encarga de monitorizar diferentes procesos en la universidad como es el caso del consumo de agua y electricidad. Actualmente el sistema solo realiza tareas como la adquisición de datos y el control de los mismos.

La revisión bibliográfica realizada sobre la diversidad de productos *SCADA*, tanto a nivel nacional como internacional, permite reafirmar lo expuesto en (Colomer, Meléndez, & Ayza, 2003) y (Bárzaga, Mompie, & Valdés, 2016), en su gran mayoría estos sistemas, aunque cumplen con las tareas de adquisición y control de datos, activación de alarmas, análisis de históricos, entre otras, distan de ser entornos de supervisión al no realizar la detección, el diagnóstico de fallos y la reconfiguración del sistema.

La carencia de estas etapas supone grandes riesgos y más cuando un *SCADA* es empleado en sistemas de energía eléctrica. El hecho de no contar con un mecanismo de detección y clasificación de perturbaciones eléctricas, puede afectar la calidad de la energía transmitida, dando lugar a fallas, cuyas consecuencias según (Rebizant, Szafran, & Wiszniewski, 2011) se mencionan a continuación:

- ❖ Daño para la planta generadora, debido a los efectos dinámicos de la corriente de falla.
- ❖ Posibilidad de acontecimientos en efecto de cascada conduciendo a apagones.

- ❖ Averías, calentamiento y vibraciones en los equipos, disminuyendo el tiempo de vida útil de los mismos y provocando pérdidas económicas.

Ante la situación descrita anteriormente surge como **problema de investigación**: ¿Cómo detectar y clasificar perturbaciones eléctricas en Sistemas de Supervisión, Control y Adquisición de Datos?

El **objeto de estudio** lo constituye el reconocimiento de patrones en señales eléctricas enmarcado en el **campo de acción**: detección y clasificación de perturbaciones que afectan la calidad de la energía eléctrica distribuida.

Para brindar solución al problema planteado se define como **objetivo general**: implementar un algoritmo para la detección y clasificación de perturbaciones eléctricas en Sistemas de Supervisión, Control y Adquisición de Datos.

Se propone el desarrollo de las siguientes **tareas de investigación**:

1. Análisis de los referentes teóricos en que se fundamenta el procesamiento de señales eléctricas.
2. Análisis de los referentes teóricos en que se fundamenta la Transformada Wavelet como herramienta para el procesamiento de señales y el algoritmo *k-NN* como herramienta de clasificación.
3. Implementación de un algoritmo para la detección y clasificación de perturbaciones eléctricas en Sistemas SCADA.
4. Ejecución de pruebas experimentales al algoritmo.
5. Validación del algoritmo mediante un prototipo funcional.

Desde el punto de vista metodológico se emplearon los siguientes métodos científicos:

Métodos teóricos:

Analítico – Sintético: Se utiliza para conocer los principales fundamentos relacionados con el procesamiento de señales eléctricas y resumir los elementos más relevantes referentes a las herramientas utilizadas con este fin.

Histórico - Lógico: Se emplea para el estudio de los antecedentes y las tendencias actuales sobre el tema procesamiento de señales eléctricas, con el objetivo de conocer su evolución y complementar las características necesarias para la solución que se propone.

Métodos empíricos:

Observación: Se utiliza para el diagnóstico del problema a investigar y permite obtener información de cada uno de los conceptos o variables tratadas en la investigación.

Experimento: Empleado en la elaboración de un prototipo de características seleccionadas con el objetivo de simular el comportamiento del algoritmo en un ambiente real y comprobar la efectividad del mismo.

El presente documento consta con la siguiente estructura capitular:

Capítulo 1. Fundamentación teórica: En este capítulo se presenta la base teórica de la investigación. Incluye aspectos generales y un análisis de mercado sobre los sistemas *SCADA* empleados para el control eléctrico. Se describe el proceso de identificación de perturbaciones eléctricas, las técnicas que se pueden utilizar en cada etapa del mismo y las herramientas que se utilizan para construir un prototipo de validación.

Capítulo 2. Descripción de la propuesta de solución: En este capítulo se realiza una descripción del algoritmo que se propone para la identificación de perturbaciones eléctricas en sistemas *SCADA*. Se especifican las herramientas de Inteligencia Artificial utilizadas para su desarrollo.

Capítulo 3. Evaluación de la solución propuesta: Este capítulo se refiere a la validación del algoritmo que se propone como solución y se realiza a través de métricas que respaldan el correcto funcionamiento del mismo. Además se presenta la creación de un prototipo que simula el funcionamiento del algoritmo en ambientes reales.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

En el presente capítulo se abordarán aspectos relacionados con los sistemas *SCADA*, las funcionalidades que brindan y los escenarios donde se pueden incluir, haciendo énfasis en el monitoreo y procesamiento de datos en sistemas de transmisión de energía eléctrica. Además, se analizan los métodos y técnicas existentes capaces de procesar señales e identificar perturbaciones en dichos sistemas.

1.1 Sistemas *SCADA*

Los resultados propuestos en (Krutz, 2005) confirman que *SCADA* es la tecnología que permite a un usuario recopilar datos de una o más instalaciones distantes y/o enviar instrucciones de control limitadas a esas instalaciones. Mientras que en (Rodríguez Penin, 2006) tratan como *SCADA* a cualquier software que admita el acceso a datos remotos de un proceso y permita, utilizando las herramientas necesarias en cada caso, el control del mismo.

De manera general se pueden sintetizar como una solución disponible para la adquisición de datos, monitoreo y control de uno o varios procesos a distancia, mediante el uso de tecnologías de comunicación.

Un sistema *SCADA* puede realizar varias funciones, ya que son programas creados para objetivos muy específicos, en (Colomer, Meléndez, & Ayza, 2003) y (Chavarría Meza, 2007) se destacan las siguientes:

- ❖ **Adquisición y almacenado de datos de instalaciones remotas:** El operador puede conocer el estado de desempeño de las instalaciones y los equipos alojados en ellas, lo que permite dirigir las tareas de mantenimiento.
- ❖ **Procesamiento de datos:** El conjunto de datos adquiridos conforman la información que alimenta el sistema, esta información es procesada, analizada, y comparada con datos anteriores, y con datos de otros puntos de referencia, dando como resultado una información confiable y veraz.
- ❖ **Almacenamiento de información histórica:** Cuentan con la opción de almacenar los datos adquiridos, esta información puede analizarse posteriormente, el tiempo de almacenamiento dependerá del operador o del autor del programa.
- ❖ **Control remoto de instalaciones y equipos:** Mediante el sistema, se pueden activar o desactivar los equipos remotamente, de manera automática y también manual (por ejemplo abrir válvulas, activar interruptores, encender motores). Además, es posible ajustar parámetros, valores de referencia y algoritmos de control.

- ❖ **Generación de reportes:** El sistema, permite generar informes con datos estadísticos de los procesos en un tiempo determinado por el operador.
- ❖ **Representación gráfica y animada de variables de proceso:** El sistema, es capaz de brindar imágenes en movimiento que representen el comportamiento de los procesos, dándole al operador la impresión de estar presente dentro de una planta real.
- ❖ **Representación de señales de alarma:** A través de las señales de alarma se logra alertar al operador de la presencia de una condición perjudicial o fuera de lo aceptable. Estas señales pueden ser tanto visuales como sonoras.

Antiguamente, los SCADA eran simplemente sistemas de telemetría, proporcionaban reportes periódicos de señales que representaban medidas y/o condiciones de estado en ubicaciones de campo remotas. Al desarrollarse la tecnología, los ordenadores asumieron el papel de manejar la recolección de datos y agregaron la capacidad de programar el sistema para realizar funciones de control más complejas. Los primeros sistemas automatizados SCADA fueron altamente modificados con programas de aplicación específicos para atender a requisitos de algún proyecto particular. Con esta percepción, numerosas industrias reutilizaron su trabajo previo sobre los nuevos proyectos y perpetuaron su propia visión de los ambientes de control con los cuales tenían experiencia (Chacon, Dijort, & Castrillo, 2001).

Hoy, se pueden encontrar sistemas SCADA adaptados para procesamiento de papel y celulosa, industrias de aceite, yacimientos de gas y petróleo, gerenciamiento y provisión de agua, control de fluidos, **sistemas de energía eléctrica**, entre otros.

1.2 SCADA en sistemas de energía eléctrica

El empleo de sistemas SCADA para el control del consumo eléctrico es una práctica que se ha extendido de manera rápida, incentivada por los costos cada vez más altos de la electricidad. Actualmente muchas empresas en el mundo brindan soluciones al respecto, algunas utilizan sistemas SCADA de propósito general y otras brindan sus propios productos diseñados para fines específicos. A continuación se muestran los trabajos realizados por algunas empresas desarrolladoras de sistemas SCADA, así como una breve descripción de sus programas.

- ❖ **Circuit Monitor 3000 de Merlin Gerin:** *Circuit Monitor CM3250* y *CM3350* han sido diseñados para entornos industriales y grandes infraestructuras comerciales, con el fin de analizar la instalación eléctrica en puntos críticos (acometidas, cargas sensibles, consumos elevados). Este analizador

posee una potencia de procesamiento que proporciona la información necesaria para la toma de decisiones y mantenimiento de los equipos. Entre sus aplicaciones resaltan: control de armónicos, consumos a nivel interno (pero no como contador de energía) y control de la instalación eléctrica en cargas críticas e instalaciones sensibles o problemáticas (Lizarraga de Miguel, 2009).

- ❖ **Power Studio y Power Studio SCADA de Circutor:** La empresa Circutor tiene en el mercado diversos software, de los cuales el *Power Studio SCADA* y el *Power Studio* suelen resultar de interés (Ver anexo 2). Mediante el *Power Studio* se pueden leer en tiempo real los valores de todos los equipos Circutor y genera un histórico de datos que se guarda en una PC para su posterior estudio. Permite al usuario tener un control absoluto de la instalación donde es utilizado, conociendo en tiempo real y de primera mano, el estado de sus líneas de potencia e incluso de consumo general de la instalación. Por otra parte la finalidad del *Power Studio SCADA* es el procesamiento final de los datos y la elaboración de informes, con el objetivo de adoptar medidas preventivas o correctivas en la instalación. Está diseñado para actuar como centralizador y gestor de información. Debido al gran volumen de información que aporta cada central medida, es necesario disponer de un sistema centralizado de recogida de datos. El *Power Studio SCADA* además está pensado para que cualquier usuario pueda crear sus propias pantallas personalizadas (Lizarraga de Miguel, 2009).
- ❖ **Control Visión de SOCOMEC:** El sistema está diseñado para comunicarse con todos los dispositivos conectados a él. Muestra los valores eléctricos medidos y lee el consumo de energía total de las instalaciones donde es utilizado. El usuario puede crear reportes de muchas variables eléctricas en un período de tiempo seleccionado y está habilitado para importar o exportar los parámetros de configuración con Excel. Según el fabricante la facilidad de integración y modificación del número de productos hace de este sistema una de las soluciones más duraderas en cuanto a monitorización de energía (Ver anexo 3) (Lizarraga de Miguel, 2009).
- ❖ **Network Manager de ABB:** El sistema ofrece una amplia gama de funciones para satisfacer las necesidades de los operadores de redes de transmisión y generación de energía. Estas van desde el análisis y la optimización de operaciones del día a día y programación a corto plazo, hasta la gestión y control de generación en tiempo real. Proporciona una arquitectura de sistema abierta y una integración versátil de las aplicaciones empresariales de ABB (Ver anexo 4) (Vetter & Mayur, 2008).

1.2.1 Análisis y valoración de los sistemas mencionados

Las acciones que realizan los sistemas SCADA mencionados con anterioridad, según (Moreno, Bellido, Oterino, & de la Rosa, 2008) están basadas en la monitorización, es decir, la decisión de control, se toma manualmente sobre la base de la experiencia y los datos disponibles en tiempo real, para luego ejecutarse, en su mayoría, a través de una interfaz gráfica de usuario. Serían considerados sistemas de supervisión si realizaran tareas de detección, diagnóstico de fallos y reconfiguración del sistema (Colomer, Meléndez, & Ayza, 2003).

Es importante señalar el valor que cobran estas últimas tareas, cuando los sistemas SCADA son empleados en instalaciones eléctricas. La energía generada y distribuida por estas instalaciones no está ajena a perturbaciones que pueden generar fallas en las redes, ocasionando averías, calentamiento, vibraciones en los equipos, entre otros, lo que provoca pérdidas económicas. Sería por tanto de gran utilidad un SCADA que contara con funciones para obtener los indicios de fallas, clasificarlas como tal, diagnosticar cuales fueron las causas primeras de estas anomalías y finalmente reconfigurar el sistema para mantener los procesos operando según lo establecido.

Para lograr este objetivo se recurre a la utilización de métodos estadísticos y herramientas de Inteligencia Artificial, que permiten el manejo del conocimiento y experiencias de forma automática, en combinación con los entornos de monitorización (Colomer, Meléndez, & Ayza, 2003).

1.3 Perturbaciones eléctricas

La calidad en la energía eléctrica es una combinación entre la calidad de la señal del voltaje y la corriente. La calidad del voltaje es una medida de cuan desviado está el voltaje del ideal y la calidad de la corriente es una medida de cuan desviada está la corriente de la ideal (Bollen & Yu-Hua Gu, 2006).

El voltaje y la corriente ideal se pueden definir como una señal en forma de onda sinusoidal con amplitud y frecuencia constantes, donde la frecuencia y la fase del voltaje son equivalentes a la frecuencia y la fase de la corriente. Cualquier desviación de la forma de onda de voltaje o corriente de la ideal sería considerada entonces una perturbación o disturbio eléctrico (Bollen & Yu-Hua Gu, 2006).

Según (Jianmin Teng, Zhaosheng, & Qiu Song, 2016) estas perturbaciones son agrupadas en siete categorías diferentes:

1. Transitorios: Onda transitoria de corriente, potencial o potencia en un circuito eléctrico. Se pueden clasificar en dos categorías, impulsivo y oscilatorio. Estos términos reflejan la forma de onda de un transitorio de corriente o voltaje.

- 1.1 Impulsivos: Cambio de frecuencia repentino y sin potencia de la condición nominal de voltaje, corriente o ambos, que es unidireccional en polaridad (principalmente positiva o negativa). Se caracterizan por sus tiempos de subida y bajada. Pueden estar dados en nanosegundos, microsegundos o milisegundos.
- 1.2 Oscilatorios: Cambio repentino de frecuencia sin potencia en la condición de estado estable de voltaje, corriente o ambos, que incluye valores de polaridad positiva y negativa. Se describe por su magnitud, duración y contenido espectral (predominantemente frecuencia). Las subclases de contenido espectral definidas son alta, media y baja frecuencia.
2. Variaciones de corta duración: Abarca las categorías de caídas de tensión, subidas de tensión e interrupciones cortas. Cada tipo de variación se puede designar como instantáneo, momentáneo o temporal, según su duración.
3. Variaciones de larga duración: Abarcan las desviaciones en las frecuencias de potencia de más de un minuto de duración. Estas variaciones incluyen las categorías de sobretensiones, sub-tensiones e interrupciones dependiendo de la causa de la variación.
4. Desequilibrio de tensión: Se define como la relación entre la magnitud de la componente de secuencia negativa y la magnitud de la componente de secuencia positiva, expresada como un porcentaje.
5. Distorsión de forma de onda: Desviación del estado estacionario de una senoide de frecuencia ideal, caracterizada principalmente por el contenido espectral de la desviación.
6. Fluctuaciones de tensión: Variaciones sistemáticas de la envolvente de voltaje o una serie de cambios de voltaje aleatorios, cuya magnitud no excede normalmente los rangos de voltaje especificados por el estándar ANSI C84.1-2006.
7. Variaciones de frecuencia eléctrica: Desviación de la frecuencia fundamental del sistema de potencia de su valor nominal específico (50 Hz, 60 Hz).

Para lograr identificar un disturbio eléctrico es necesario analizar constantemente la señal de voltaje recibida en busca de perturbaciones que puedan generarse.

1.4 Identificación de perturbaciones eléctricas

Los resultados propuestos en (Mohan Saha, Izykowski, & Rosolowski, 2010) han confirmado que la identificación de perturbaciones se compone de dos procesos fundamentales, la detección y la clasificación de la perturbación. La mayoría de las veces estos procesos están interrelacionados, y por eso es común verlos como uno solo.

En (Bollen & Yu-Hua Gu, 2006) plantean que un proceso de identificación de perturbaciones se puede dividir en las siguientes etapas:

- ❖ Procesamiento de la señal.
- ❖ Extracción y optimización de características.
- ❖ Selección de la topología o arquitectura de clasificación.
- ❖ Realización del aprendizaje automático ya sea supervisado o no.
- ❖ Validación y prueba.
- ❖ Clasificación.

De manera general la identificación se materializa a través de un proceso de reconocimiento de patrones y tiene como objetivo facilitar la extracción de información relevante en la señal, que permita detectar y clasificar posibles disturbios en la misma. Una vez procesada la señal y extraída sus características se procede entonces a diagnosticar y clasificar las perturbaciones. Las herramientas adecuadas para alcanzar este objetivo utilizan características o atributos de la señal procesada como entrada y clases etiquetadas como clasificadores de los datos en la salida e incluyen el aprendizaje automático, la clasificación y diagnóstico automático.

Para la realización del proceso de identificación de perturbaciones, el presente trabajo se regirá por lo propuesto en (Bollen & Yu-Hua Gu, 2006) ya que el proceso se desglosa en varias etapas para un mejor entendimiento del mismo. A continuación se profundiza en que consiste cada una de las etapas.

1.4.1 Procesamiento de señales eléctricas

El procesamiento de señales es la manipulación matemática de una señal con el objetivo de extraer características e información relevante de la misma (Bollen & Yu-Hua Gu, 2006). Este procedimiento se realiza mediante el uso de transformadas que permiten representar la señal medida en un dominio diferente (el dominio del tiempo, el dominio de la frecuencia o el dominio bidimensional tiempo-frecuencia) dentro del cual es más fácil extraer la información buscada. La elección del más adecuado estará en dependencia de si la señal es o no estacionaria.

- ❖ **Señales estacionarias:** Una señal se considera estacionaria cuando es estadísticamente invariante en el tiempo o sus datos son independientes de este, debido a lo cual su procesamiento solo es necesario realizarlo en el dominio de la frecuencia. Las señales estrictamente estacionarias no existen en los sistemas de energía en la vida real, pues pequeños y grandes cambios estadísticos siempre están presentes en sus parámetros (Bollen & Yu-Hua Gu, 2006).

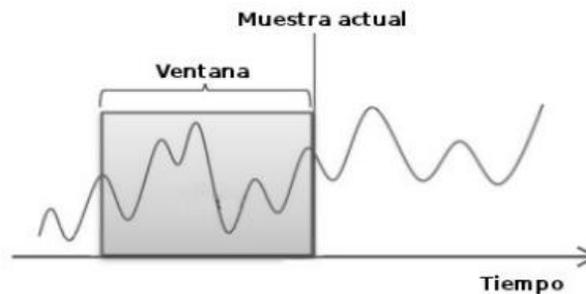
- ❖ **Señales no estacionarias:** Una señal no estacionaria es aquella estadísticamente variante en el tiempo o sus datos son dependientes del mismo, tal es el caso de las perturbaciones eléctricas. Debido a esto se recomienda descomponerla en componentes de frecuencia dependientes del tiempo y procesarla en un dominio bidimensional tiempo-frecuencia (Bollen & Yu-Hua Gu, 2006).

En las representaciones convencionales de una señal tanto en el dominio del tiempo, como en el de la frecuencia, ambas variables tiempo y frecuencia son consideradas mutuamente excluyentes (al obtener una representación en términos de una variable, la otra queda excluida). Una solución adecuada a este inconveniente es hacer una representación de la señal como una función de dos variables, en un espacio bidimensional tiempo-frecuencia, de esta manera es posible analizar la información brindada por la concentración de energía de la señal en dos dimensiones (tiempo y frecuencia) en vez de una sola (tiempo o frecuencia). Esta representación no solo muestra cómo se distribuye la señal en ambas dimensiones, también muestra cómo varía la frecuencia respecto al tiempo (Boashash, 2003).

1.4.1.1 Representación de una señal en el dominio bidimensional tiempo-frecuencia

Para obtener esta representación se emplea una técnica conocida como ventana deslizante (Ver Figura 1). Está basada en la multiplicación de la señal original por una función ventana, con el objetivo de acotarla en cierto intervalo de tiempo, a partir de aquí transformar la señal resultante al dominio de la frecuencia y así obtener una relación tiempo-frecuencia. En caso de desplazar temporalmente la ventana es posible cubrir el dominio de la señal y obtener la información completa tiempo-frecuencia de la misma.

Figura 1: Ventana deslizante



Fuente: (Cedeño Pozo, 2014)).

Para realizar esta transformación de un dominio a otro se utilizan las transformadas matemáticas. Algunas de las más mencionadas en la literatura son: la Transformada de Fourier con Ventana y la Transformada Wavelet (Ver epígrafe 1.4.1.2).

Se debe tener en cuenta en la transformación que una señal no puede ser representada como un punto en un espacio tiempo-frecuencia, solo es posible determinar su posición en este espacio dentro de un rectángulo de área $\Delta t \Delta f$ (Poularikas, 2010). Esta relación también es conocida como desigualdad de Heisenberg y matemáticamente implica que:

Dada una ventana real $g(t)$ localizada en un intervalo centrado $t = 0$ de longitud Δt y su transformada $g(f)$ localizada en una banda centrada $f = 0$ de ancho Δf , es posible obtener una representación tiempo-frecuencia de la señal, en el rectángulo centrado en el punto (t, f) de dimensión $\Delta t \Delta f$ (Poularikas, 2010). La exactitud de esta representación está acotada por el principio de incertidumbre, el cual establece la imposibilidad de reducir indefinidamente la dimensión del rectángulo, debido a la existencia de una constante positiva universal $c = \frac{1}{2}$ tal que: el producto de las resoluciones en tiempo y frecuencia satisface la inecuación $\Delta t \Delta f \geq \frac{1}{2}$

1.4.1.2 Transformada de Fourier con Ventana

La Transformada de Fourier con Ventana (TFV) permite hacer una transformación del dominio del tiempo al dominio de la frecuencia. Esta se basa en la suposición de que una señal $x(t)$ no estacionaria se puede dividir en segmentos limitados por una ventana $w(\tau - t)$, dentro de la cual se pueda considerar como estacionaria (Mohan Saha, Izykowski, & Rosolowski, 2010).

La ventana de observación permite la localización del espectro en el tiempo, pero también difumina el espectro en frecuencia de acuerdo con el principio de incertidumbre. Una vez que se eligen el tipo y el tamaño de la ventana de datos para la TFV, existe una resolución de frecuencia fija en todas las bandas, lo cual constituye una desventaja. Al aumentar el tamaño de la ventana, se reduce la resolución en las bandas de frecuencia y viceversa. Para obtener diferentes resoluciones en diferentes bandas de frecuencia, es necesario volver a hacer la TFV con un tamaño de ventana diferente y luego extraer las salidas de las bandas correspondientes (Bollen & Yu-Hua Gu, 2006).

1.4.1.3 Transformada Wavelet

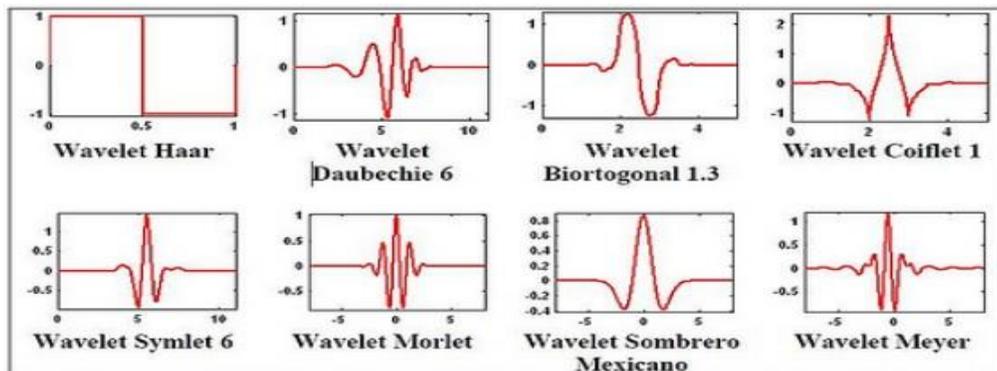
Al igual que la TFV, la Transformada Wavelet (TW) divide la señal en segmentos, limitados por una función ventana dentro de la cual se puede considerar como estacionaria pero, esta emplea una ventana

autoajustable a la distribución de frecuencia analizada y minimiza al máximo el principio de incertidumbre. De esta manera, obtiene la mejor aproximación posible de la señal en el dominio tiempo-escala, donde las escalas son inversamente proporcionales a las frecuencias (Poularikas, 2010).

La escala está relacionada con el tamaño de ventana de la wavelet. Una gran escala realiza un análisis global de la señal y una pequeña escala realiza un análisis detallado de la misma. Para una función wavelet determinada, la reducción de la escala reducirá el tamaño de la ventana y aumentará la resolución en el tiempo, debido a esto una de sus aplicaciones más comunes es la detección y estudio de singularidades en una señal (Sierra, León, Quintana, & Carnesoltas, 2011).

La TW basa su descomposición en la dilatación y contracción de funciones de análisis denominadas wavelets madres (Ver Figura 2). La elección de alguna de ellas depende de la aplicación específica que se le vaya a dar teniendo en cuenta sus propiedades (Daubechies, 1990).

Figura 2: Wavelets madres más utilizadas

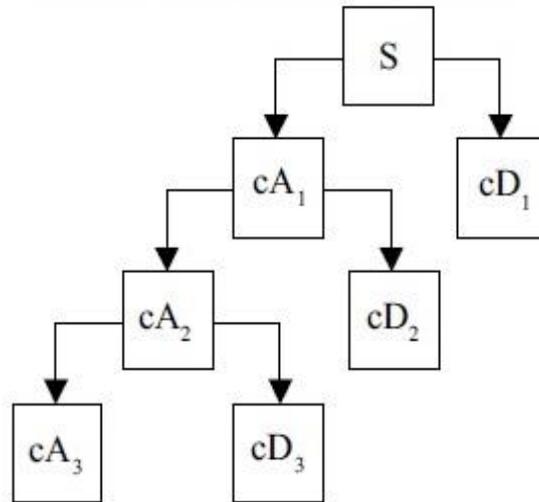


Fuente: (Castañeda & Paternina, 2013)).

En su versión discreta, la TW es una transformación de la señal, que la divide en dos tipos de subseñales, la tendencia o coeficientes de aproximación (CA) y las fluctuaciones o coeficientes de detalle (CD). La tendencia viene a ser una copia de la señal a menor resolución y las fluctuaciones almacenan información referida a los cambios locales en la señal original. El estudio de las fluctuaciones permite detectar anomalías o disfunciones en el comportamiento esperado de la señal inicial (Marínez, Peris, & Rodenas, 2004).

La Transformada Wavelet Discreta (TWD) permite realizar un estudio a varias escalas de la señal conocido como Análisis Multi-Resolución (AMR). Este análisis permite descomponer la señal en dos bandas de frecuencia (CA y CD) para cada escala wavelet, con el fin de procesar la señal en estas múltiples bandas de manera diferente e independiente como se muestra a continuación:

Figura 3: Análisis Multi-Resolución a tres escalas para una señal (S)



Fuente: (Software MATLAB).

La ventaja del AMR radica en la posibilidad de sintetizar numéricamente algoritmos efectivos para la determinación de la expansión wavelet.

1.4.2 Extracción y optimización de características

De manera frecuente las características extraídas de una señal se utilizan como la entrada de un sistema de clasificación en lugar de la propia señal, esto por lo general conduce a una entrada del sistema mucho más simple. Seleccionar un conjunto apropiado de ellas es un importante paso hacia una clasificación con éxito. Algunas de las características utilizadas para determinar las perturbaciones que afectan la calidad de la energía se enumeran a continuación (Bollen & Yu-Hua Gu, 2006):

- ❖ Armónicos y componentes inter-armónicos, así como sus frecuencias y magnitudes.
- ❖ Componentes principales de la señal (amplitud, frecuencia, fase inicial).
- ❖ Distribución espectral de eventos.
- ❖ Estadísticas, distribuciones de probabilidad o histogramas de las perturbaciones.
- ❖ Energía de la señal.

En dependencia del tipo de análisis a realizar y de las herramientas de procesamiento utilizadas, la lista anterior puede ser mucho más larga. Si no existe un conocimiento previo sobre qué características podrían

identificar mejor el tipo de perturbación, una forma de hacerlo es seleccionar varias de ellas. La redundancia que llegue a existir puede ser eliminada a través de un proceso de optimización. La optimización de características tiene diferentes objetivos, entre ellos se encuentra la normalización de los datos, el incremento de la separabilidad entre las clases de clasificación y el establecimiento de un nivel de importancia entre las características. Este proceso presenta relación con la topología del clasificador usado, pues el clasificador puede suplir algunos de los objetivos de la optimización.

1.4.2.1 Normalización

La normalización es aplicada para asegurarse de que cada característica en un vector de características sea imparcial y adecuadamente escalada, de manera que diferentes características tengan igual ponderación en un clasificador, si no existe una información previa sobre cuál de ellas es más importante que las demás (Bollen & Yu-Hua Gu, 2006).

Dado entonces $C_1, C_2, C_3, \dots, C_m$ conjuntos de características diferentes y $X_1, X_2, X_3, \dots, X_n$ vectores de características m-dimensionales extraídos de la señal, tal que cada $X_j = [x_1, x_2, x_3 \dots x_m]$ y $x_i \in C_i$. La normalización de las características se realiza de la siguiente manera:

$$\tilde{x}_i = \frac{x_i - \mu_i}{\sigma_i}, \text{ para } i = 1, 2, 3, \dots, m.$$

Donde μ_i y σ_i son la media y la desviación estándar del i ésimo conjunto de característica de la señal (Bollen & Yu-Hua Gu, 2006).

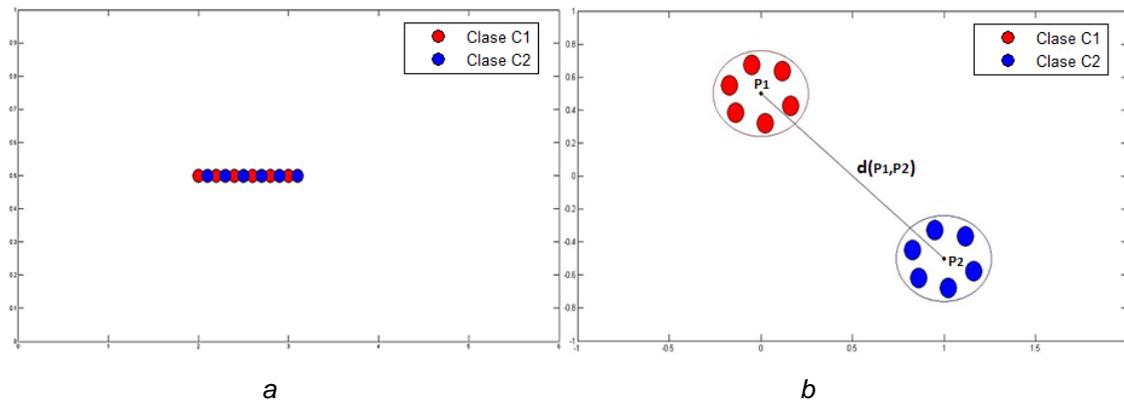
1.4.2.2 Incremento de la separabilidad entre clases

Otro de los objetivos de la optimización de características es maximizar la separabilidad entre las clases a la cual pertenecen. Una de las técnicas más empleadas para lograr este objetivo es el Análisis Discriminante Lineal (ADL), el cual permite reducir la dimensión del espacio de características preservando la mayor parte de la información discriminante entre las clases, pero realiza suposiciones que limitan su aplicación. Estas limitaciones están dadas por el hecho de tomar como medida de separación entre las clases, la dispersión de los datos en función de la media y la varianza, lo cual supone implícitamente una distribución normal de los datos y esto no siempre se cumple (Gómez Chova, 2002). Una alternativa a este inconveniente es el uso de métodos Kernel.

Los métodos Kernel permiten analizar la información transformando los datos del espacio de entrada hacia un espacio de mayor dimensión, llamado espacio de características, en el cual los datos de entrada son más propensos a ser separados (linealmente), según su clase correspondiente, en comparación con el espacio original (Wang L. , 2005).

Se puede obtener una buena separación entre clases, es decir, es posible encontrar una frontera lineal entre las clases en un espacio de mayores dimensiones, si se seleccionan las características adecuadas; tal que, permitan disminuir la dispersión entre los elementos de una misma clase (hacerlos más compactos) y a la vez aumentar la dispersión entre las diferentes clases de clasificación, es decir, maximizar la distancia interclases, como se muestra en la imagen:

Figura 4: Ejemplo de características: a) no separables linealmente, b) separables linealmente.



Fuente: (elaboración propia).

En la primera gráfica de la Figura 4 se observan los elementos de las clases c_1 y c_2 en un espacio de una dimensión, donde no pueden separarse linealmente. En la segunda gráfica se observan los mismos elementos proyectados en un espacio de dos dimensiones, donde pueden separarse linealmente. En este nuevo espacio las clases están bien separadas y sus elementos lo suficientemente compactos para permitir una discriminación lineal.

En (Wang L. , 2005).se plantea que la transformación hacia el espacio de mayores dimensiones es efectuada implícitamente dentro de una función Kernel, la cual se define como un producto interno en el espacio de características o espacio Kernel y se denota como: $K(x_i, x_j) = \langle \varphi(x_i) \varphi(x_j) \rangle$ donde \langle , \rangle denota producto vectorial y $\varphi(\cdot)$ es una función de mapeo no lineal y tal que:

$$\varphi(x) : \mathbb{R}^n \rightarrow \mathbb{R}^m, m > n.$$

El uso de esta función Kernel tiene la ventaja de obtener el producto escalar en el espacio de características, pero realizando los cálculos en el espacio de entrada; para ello basta con evaluar la función $K(x_i, x_j)$ en los

datos de entrada. Otra ventaja del método es que no necesita conocer explícitamente la función $\varphi(\cdot)$ (Wang L. , 2005).

Aunque existen varias funciones Kernels, en la literatura se recomienda especialmente el trabajo con 5 de ellas, las cuales se muestran a continuación:

Tabla 1: Kernels más difundidos en la literatura

Función Kernel	Tipo de clasificador
$K(x_i, x_j) = \langle x_i, x_j \rangle$	Lineal.
$K(x_i, x_j) = [\langle x_i, x_j \rangle + 1]^d$	Polinomial de grado d.
$K(x_i, x_j) = e^{(-\gamma \ x_i - x_j\ ^2)}$	Gaussiano de base radial.
$K(x_i, x_j) = \tanh[\langle x_i, x_j \rangle + 1]^*$	Perceptrón multicapa.
$K(x_i, x_j) = \frac{1}{\sqrt{\ x_i - x_j\ ^2 + b}}$	Cuadrático inverso.

Fuente: (Wang L. , 2005).

1.4.3 Selección de la topología o arquitectura de clasificación

La elección de la arquitectura de clasificación es un proceso complejo basado en la búsqueda de una función de mapeo entre un espacio de entrada y un espacio característico, seguido de la aplicación de algunas reglas de decisión que permitan hacer una clasificación en este espacio característico. Para el caso específico de la clasificación de perturbaciones eléctricas, numerosos autores han investigado y algunos de los clasificadores que proponen se listan a continuación:

1.4.3.1 Redes Neuronales Artificiales propuesto en (Karasu & Sarac, 2017)

Las Redes Neuronales Artificiales (ANN o *Artificial Neural Networks*, por sus siglas en inglés) son un paradigma de aprendizaje y procesamiento automático. Están inspiradas en las redes neuronales del cerebro humano, constituidas por elementos que se comportan de forma similar a la neurona biológica en sus funciones más comunes (Basogain Olabe, 1998).

Una ANN se compone de un conjunto de unidades llamadas neuronas, conectadas entre sí y organizadas por capas según el tipo de arquitectura de la red. Cada neurona recibe una serie de entradas a través de

interconexiones y emite una salida. Esta salida viene dada por las siguientes funciones (Basogain Olabe, 1998):

1. Función de propagación (también conocida como función de excitación), que por lo general consiste en el sumatorio de cada entrada multiplicada por el peso de su interconexión (valor neto).
2. Función de activación, que modifica a la anterior. Puede no existir, siendo en este caso la salida la misma función de propagación.
3. Función de transferencia, que se aplica al valor devuelto por la función de activación. Se utiliza para acotar la salida de la neurona y generalmente viene dada por la interpretación que se le quiera dar a dichas salidas. Algunas de las más utilizadas son la función sigmoidea (para obtener valores en el intervalo $(0,1)$) y la tangente hiperbólica (para obtener valores en el intervalo $(-1,1)$).

Algunas *ANN* presentan la característica de ser "asociativas", significa que para una entrada parcial la red elegirá la entrada más parecida en memoria y generará una salida que corresponda a la entrada completa. La naturaleza de la memoria de las *ANN* permite que la red responda adecuadamente cuando se le presenta una entrada incompleta o con ruido. Esta propiedad suele ser referida como la capacidad de generalización (Basogain Olabe, 1998).

1.4.3.2 Árboles de decisión propuesto en (Mahela Prakash & Shaik Gafoor, 2016)

Un árbol de decisión es una forma de representar reglas de clasificación inherentes a los datos, con una estructura en árbol n -ario que particiona los datos de manera recursiva. Cada rama de un árbol de decisión representa una regla que decide entre una conjunción de valores de un atributo básico (nodos internos) o realiza una predicción de la clase (nodos terminales) (Benítez, Escudero, Samir, & David, 2014).

El proceso de construcción del árbol es un proceso iterativo, en el que, en cada iteración, se selecciona el atributo que mejor particiona el conjunto de entrenamiento. Para realizar este proceso, se debe mirar la bondad de las particiones que genera cada uno de los atributos y, en un segundo paso, seleccionar el mejor. Dos de los grandes inconvenientes que plantea son la elevada fragmentación de los datos en presencia de atributos con muchos valores y el elevado coste computacional que esto implica. Esto hace que no sea un método muy adecuado para problemas con grandes espacios de atributos (Benítez, Escudero, Samir, & David, 2014).

1.4.3.3 k- Vecinos más Cercanos propuesto en (Cai & Thornihill, 2017)

El algoritmo vecinos más cercanos consiste en estimar la clase de un dato desconocido a partir de las características del dato más próximo, según una medida de similitud o distancia. Esta regla tiene propiedades estadísticas bien establecidas y facilidad de aplicación a sistemas reales. El método del vecino

más cercano se puede extender utilizando no uno, sino un conjunto de datos más cercanos para predecir la clase a la cual pertenecen los nuevos datos, en lo que se conoce como los k -vecinos más cercanos (k -NN o *k-Nearest Neighbors* por sus siglas en inglés). El algoritmo k -NN se usa con mayor frecuencia para la clasificación, aunque también se puede usar para la estimación y predicción. Es un ejemplo de aprendizaje basado en instancias. Al considerar más de un vecino, se brinda inmunidad ante ruido y se suaviza la curva de estimación (Larose, 2005).

La clasificación de nuevos ejemplos utilizando k -NN se realiza buscando el conjunto de los k ejemplos más cercanos de entre un conjunto de ejemplos etiquetados, previamente guardados y seleccionando la clase más frecuente de entre sus etiquetas. La generalización se pospone hasta el momento de la clasificación de nuevos ejemplos (Benítez, Escudero, Samir, & David, 2014).

Existen elementos importantes que se deben revisar a la hora de utilizar este método. La definición de la medida de distancia (o similitud) apropiada para el problema a tratar y el valor de la propia k son primordiales.

La medida de distancia más utilizada es la euclidiana pero, otros tipos de distancias pueden ser utilizadas dependiendo del problema a tratar. A continuación se nombran algunas de ellas (Benítez, Escudero, Samir, & David, 2014):

- ❖ Distancia de Manhattan.
- ❖ Distancia Kernel.
- ❖ Distancia de Hamming.

Para la elección de los k vecinos más cercanos, es decir el valor k , puede que no haya una solución obvia. En (Larose, 2005) se plantea que al considerar elegir un valor pequeño para k , entonces es posible que la clasificación o estimación pueda verse afectada por valores atípicos u observaciones inusuales ("ruido"). Con una k pequeña por ejemplo: $k = 1$ el algoritmo simplemente devolverá el valor de la observación más cercana, un proceso que puede llevar al algoritmo a sobreajustar, tendiendo a memorizar el conjunto de datos de entrenamiento a expensas de la generalización. Por otro lado, elegir un valor de k que no sea demasiado pequeño, tenderá a suavizar cualquier comportamiento característico aprendido en el conjunto de entrenamiento.

Es posible que los datos en sí ayuden a resolver este problema, si se sigue un procedimiento de validación cruzada. De esta manera se probarían varios valores de k con diferentes conjuntos de entrenamiento seleccionados al azar y se elegiría el valor de k que minimice la clasificación o el error de estimación, haciendo así el algoritmo muy preciso (Larose, 2005).

1.4.4 Realización del aprendizaje automático

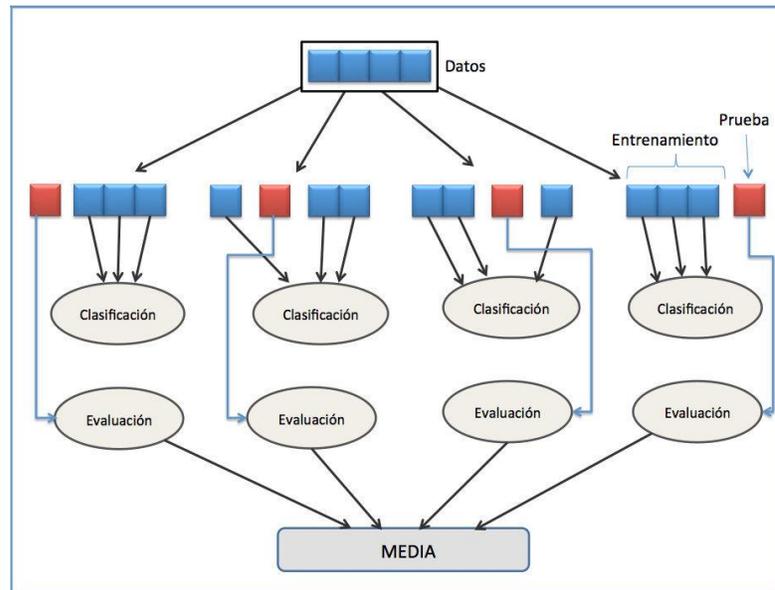
Antes de elegir la arquitectura del clasificador, se debe verificar si los datos de entrenamiento dados (características extraídas de la señal) están previamente etiquetados de acuerdo a sus clases. Una vez hecho esto se considera entonces la variante de aprendizaje automático a realizar ya sea supervisado o no. El aprendizaje supervisado se refiere al entrenamiento de un clasificador utilizando vectores de entrada pre-etiquetados. Contrariamente a esto, en el aprendizaje sin supervisión las clases de los vectores de entrada son desconocidas.

Aunque el entrenamiento no supervisado es necesario con frecuencia en problemas del mundo real, las teorías están menos desarrolladas en comparación con el entrenamiento supervisado.

1.4.5 Validación y prueba

La validación y prueba del clasificador, una vez que ha sido entrenado, tiene como objetivo comprobar la precisión con la cual este es capaz de predecir correctamente las clases asociadas a datos previamente desconocidos. Para ello es necesario realizar una estimación del error de generalización del mismo, es decir, el error en la clasificación de los datos de prueba. Una técnica comúnmente usada con este objetivo es la validación cruzada (Ver figura 5).

Figura 5: Técnica de Validación Cruzada



Fuente: (elaboración propia).

Esta técnica permite estimar el error de generalización por re-muestreo del conjunto de datos de entrenamiento. Para ello se utiliza un tipo de validación cruzada conocida como k iteraciones. Esta variante divide los datos de entrenamiento en k subconjuntos, de los cuales $k - 1$ son subconjuntos usados para el entrenamiento y el subconjunto restante se utiliza para estimar el error de clasificación. Este proceso se repite k veces hasta que cada único subconjunto se ha usado una vez para la estimación. Luego se calcula la media aritmética de los errores de clasificación que fueron obtenidos por cada iteración. Esta media representa una predicción del error de generalización del sistema de clasificación (Bollen & Yu-Hua Gu, 2006).

1.4.6 Clasificación

Una vez diseñado y entrenado el clasificador, la clasificación se puede realizar mediante la introducción al mismo de vectores con características extraídas de los datos de la señal. Un tema importante a tener en cuenta, es que en problemas del mundo real un clasificador debe diseñarse para permitir la actualización y ajuste de sus parámetros de forma dinámica; por ejemplo, si una nueva clase de eventos se obtiene de las mediciones, la información debe ser añadida para enriquecer el clasificador existente.

1.5 Herramientas y tecnologías

Un producto de software tiene calidad en gran medida por las herramientas y tecnologías que se manipulan para desarrollarlo. A continuación se enunciarán las utilizadas en la propuesta de solución. La selección de las mismas, se basa en las políticas de desarrollo del módulo y del proyecto para el cual se realiza el algoritmo.

1.5.1 QtCreator

Es un Entorno de Desarrollo Integrado (IDE por sus siglas en inglés), multiplataforma, programado en C++, JavaScript y QML, donde se pueden editar visualmente interfaces diseñadas. La abreviatura Qt hace referencia a la biblioteca que incluye un conjunto de widgets encargados de proporcionar las funciones estándar de una Interfaz Gráfica de Usuario (GUI por sus siglas en inglés) y además incluye las herramientas necesarias para el diseño de aplicaciones de una forma rápida y sencilla. Todo el código fuente y el contenido publicado puede ser redistribuido y / o modificado bajo los términos de la Licencia Pública General Reducida de GNU, (o más conocida por su nombre en inglés como GNU LGPL) que asegura que el software es libre para todos los usuarios (Díaz Tribaldos, Escobar Ocampo, & Vivas Albán, 2015). Se propone utilizar QtCreator v 3.3.1 en la propuesta de solución.

1.5.2 Lenguaje de programación C++

Se trata de un lenguaje de programación basado en el lenguaje C. Estandarizado por la (ISO/IEC_International Organization for Standardization/International Electrotechnical Commission) y ampliamente difundido. C++ se ha convertido en un lenguaje potente, eficiente y seguro. Posee características superiores a otros lenguajes, las más importantes según (Sierra Ceballos, 2008) son:

- ❖ Programación orientada a objetos: Esta característica permite al programador diseñar aplicaciones pensando en la comunicación entre objetos más que en una secuencia estructurada de código. Además, permite la reutilización del código de una forma más lógica y productiva.
- ❖ Portabilidad: Prácticamente se puede compilar el mismo código C++ en la totalidad de ordenadores y sistemas operativos sin apenas hacer cambios. Por eso es uno de los lenguajes más portados a diferentes plataformas.
- ❖ Brevedad: El código en C++ es muy corto en comparación con otros lenguajes, debido a la facilidad con la que se puede anidar expresiones y la gran cantidad de operadores.
- ❖ Velocidad: El código resultante de una compilación en C++ es muy eficiente debido a su dualidad como lenguaje de alto y bajo nivel y al reducido tamaño del lenguaje mismo.

1.5.3 Biblioteca QCustomPlot

Es una biblioteca de trazado desarrollada en C++, se centra en hacer tramas y gráficos en 2D de buen aspecto y calidad de publicación, además de ofrecer un alto rendimiento para aplicaciones de visualización en tiempo real. QCustomPlot puede exportar a varios formatos, como archivos PDF vectorizados e imágenes rasterizadas como PNG, JPG y BMP. Es un software libre, todo el código fuente y el contenido publicado puede ser redistribuido y / o modificado bajo los términos de la Licencia Pública General de GNU (GNU GPLv3), ya sea la versión 3 de la licencia o cualquier versión posterior (Eichhammer, 2016).

Esta biblioteca será empleada en su versión 2.0.0 -beta para graficar en tiempo real las perturbaciones eléctricas teniendo en cuenta las siete categorías definidas en (Jianmin Teng, Zhaosheng, & Qiu Song, 2016).

1.5.4 Biblioteca Wavelet 1D/2D

Biblioteca desarrollada en C ++ para el trabajo con la TW, se centra en la velocidad y facilidad de uso. Está disponible en las plataformas LINUX y Win32. Tiene como principal característica que permite el análisis de las señales en una y dos dimensiones (1D/2D). Es un software libre; puede ser redistribuido y / o modificado

bajo los términos de la Licencia Pública General de GNU (GNU GPLv2), ya sea la versión 2 de la licencia, o cualquier versión posterior (Hussain, 2011).

1.5.5 Software MATLAB

MATLAB es un potente lenguaje diseñado para la computación técnica. El nombre MATLAB proviene de (Laboratorio de Matrices o *Matrix Laboratory* por sus siglas en inglés). Puede ser utilizado en computación matemática, modelado y simulación, análisis y procesamiento de datos, visualización y representaciones gráficas, así como para el desarrollo de algoritmos. El programa estándar de MATLAB comprende una serie de funciones que pueden ser utilizadas para resolver problemas comunes pero, incorpora además otras bibliotecas llamadas *toolboxes* diseñadas para resolver problemas específicos como por ejemplo el procesamiento de señales, el cálculo simbólico y el diseño de sistemas de control (Gilat, 2005). Está disponible para las plataformas Unix, Windows, Mac OS X y GNU/Linux. Es un software propietario de MathWorks aunque de forma independiente, se ofrecen los *toolboxes*.

Conclusiones parciales

En el estudio realizado se confirma que varios de los sistemas *SCADA* comercializados en la actualidad distan de ser entornos de supervisión al no contar con las etapas de detección de fallos, diagnóstico de estos y reconfiguración del sistema.

Se manifiestan las ventajas del uso de la TWD como herramienta de procesamiento de señales, al brindar la mejor descomposición posible tiempo-frecuencia de la señal.

CAPÍTULO 2: DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN

Introducción

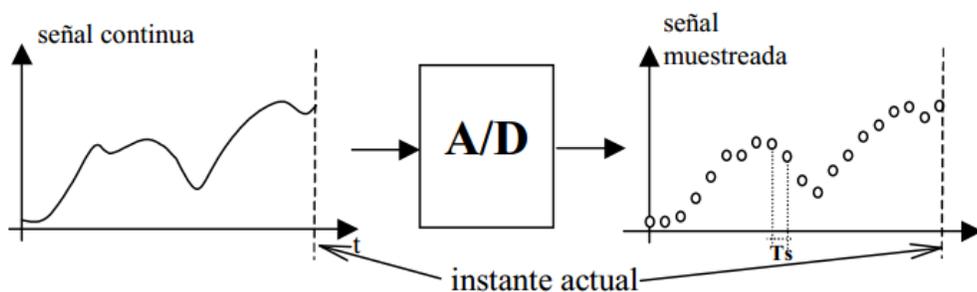
En este capítulo se abordan una serie de elementos que conforman un algoritmo para la detección y clasificación de perturbaciones eléctricas en Sistemas de Supervisión, Control y Adquisición de Datos, mediante el procesamiento de las señales con la TW y utilizando como clasificador k -NN con distancia Kernel.

En primer lugar, se debe tener en cuenta que, el algoritmo que se propone se puede integrar a cualquier sistema SCADA mientras este realice tareas de adquisición o lectura de datos de voltaje, no necesariamente tiene que ser un SCADA eléctrico. El mismo recibe como entrada una señal de voltaje digitalizada y arroja como resultado el tipo de perturbación presente en la señal.

2.1 Digitalización de la señal de voltaje

Para poder analizar y procesar la señal de voltaje es necesario hacer uso de un medidor eléctrico, que permita digitalizar la tensión eléctrica. La digitalización según (Colomer, Meléndez, & Ayza, 2003) es un procedimiento de conversión de la señal analógica o continua en digital, que generalmente es transparente para el usuario del sistema SCADA y como resultado, arroja una secuencia de muestras de la señal original (señal muestreada) con una representación numérica (Ver Figura 6).

Figura 6: Digitalización de una señal de voltaje



Fuente: (Colomer, Meléndez, & Ayza, 2003).

Se deben tomar las muestras de la señal analógica a una frecuencia o tasa de muestreo constante (T_s). El muestreo debe estar basado en el teorema de Nyquist-Shannon el cual plantea: Sea $x_c(t)$ una señal de banda limitada, es decir, una señal cuyo espectro $X_c(\omega)$ es cero para $|\omega| \geq \omega_N$. Entonces $x_c(t)$ está

determinada de forma única por sus muestras $x[n] = x_c(nT)$, $n = 0, \pm 1, \pm 2, \dots$, si se cumple que la frecuencia de muestreo $\omega_s = \frac{2\pi}{T} \geq 2\omega_N$.

Como $\omega_s = \frac{2\pi}{T} \geq 2\omega_N$, la restricción implica que $T \leq \frac{\pi}{\omega_N}$, donde T es el intervalo entre las muestras. El máximo intervalo de tiempo permitido es $T_{max} = \frac{\pi}{\omega_N}$ que se conoce como intervalo de Nyquist y la frecuencia $2\omega_N$, que debe ser menor que la frecuencia de muestreo ω_s , se denomina frecuencia de Nyquist (James, 2002).

La señal analógica del voltaje (en su forma ideal) representa una onda perfectamente sinusoidal, con una frecuencia que varía de 50 a 60 ciclos por segundo, definida matemáticamente como: $y = \text{sen}(2\pi ft)$ con $50 \leq f \leq 60$. Por tanto el proceso de digitalización de la misma requiere un periodo de muestro de $T \leq \frac{\pi}{\omega_N} = \frac{\pi}{2\pi 60} \approx 0.008 \text{ seg}$, con $\omega_N = 2\pi 60$ (mayor frecuencia angular alcanzada por la señal) y así satisfacer las condiciones del teorema de muestreo de Nyquist-Shannon.

Esta señal digitalizada no es estrictamente estacionaria, pues al provenir de un sistema de energía eléctrica real siempre es propensa a presentar alteraciones en sus parámetros (Bollen & Yu-Hua Gu, 2006). Por esta razón, surge la idea de realizar un algoritmo que se pueda integrar a los sistemas SCADA y sea capaz de procesar la señal de voltaje digitalizada en busca de anomalías que afectan la calidad de la energía transmitida.

El algoritmo que se propone hace uso herramientas de Inteligencia Artificial, que permiten la utilización del conocimiento de forma automática y se rige por las etapas propuestas en (Bollen & Yu-Hua Gu, 2006) del proceso de identificación de perturbaciones eléctricas.

2.2 Etapa 1. Procesamiento de la señal

2.2.1 Selección de la herramienta de procesamiento de la señal

Se propone como herramienta de procesamiento de señales el uso de la TW en su versión discreta (Ver epígrafe 1.4.1.3). La elección está basada en el hecho de que esta permite minimizar al máximo el principio de incertidumbre, brindando la representación más exacta de la señal en el dominio tiempo escala, donde la escala es inversamente proporcional a la frecuencia.

Con el objetivo de simplificar el trabajo en esta etapa, se hará uso de la biblioteca Wavelet 1D/ 2D, la misma brinda la función *dwt* (*Discrete Wavelet Transform (DWT)* o Transformada Wavelet Discreta, por sus siglas en inglés) que permite la extracción de los CA y CD. Recibe los siguientes parámetros:

- ❖ `vector<double>&sig`: Vector de entrada con los datos de la señal en tiempo real.
- ❖ `int J`: Máxima escala de descomposición.
- ❖ `string nm`: Nombre de la Wavelet a utilizar.
- ❖ `vector<double> &dwt_output`: Vector de salida de la Transformada Wavelet Discreta. Almacena los CA y CD en el siguiente formato: [CA (J) CD (J) CD (J-1)..... CD (1)].
CA (J): Es el conjunto de coeficientes de aproximación en la escala máxima J.
CD (J), CD (J-1)..... CD (1): Son los conjuntos de los coeficientes de detalle en cada escala.
- ❖ `vector<double> &flag`: Denominado también vector de limpieza. Contiene dos valores:
flag [0]: Toma valor 0 si la señal es par y 1 si la señal es impar.
flag [1]: Contiene la cantidad de niveles de descomposición.
- ❖ `vector<int> &length`: Almacena las longitudes de los respectivos conjuntos de aproximación y detalle.
La última entrada del vector es la longitud de la señal original.

Una vez que se obtienen los CA y CD se procede a detectar si hay presencia o no de perturbaciones.

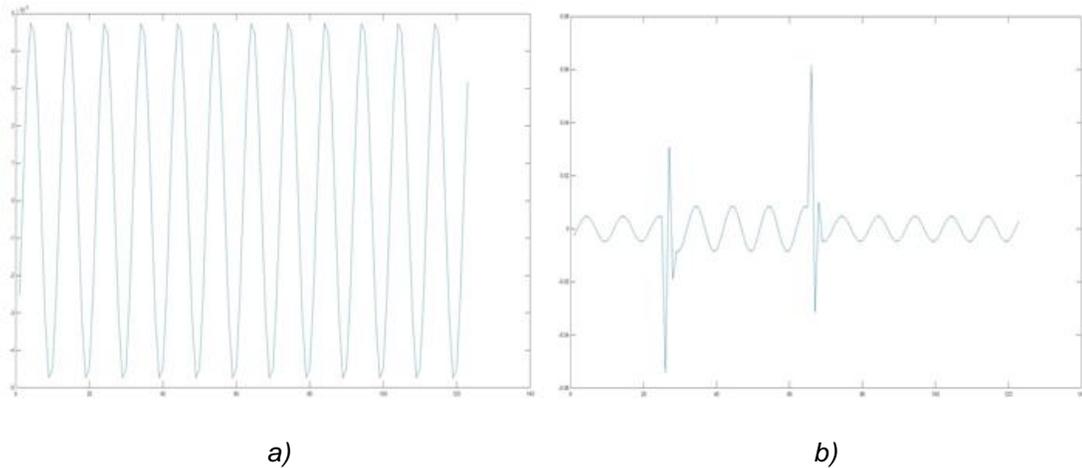
2.2.2 Detección de las perturbaciones

Para la detección de las perturbaciones, basta con realizar un análisis a los CD en la primera escala o nivel de descomposición wavelet, pues esta brinda una mayor resolución en el tiempo. La wavelet funciona de manera similar a un microscopio, mientras menor sea la escala más cerca se podrá observar la señal y así detectar los cambios bruscos o pequeñas singularidades presentes en la misma.

A continuación se muestra un análisis gráfico del comportamiento de los CD en una señal ideal y en diferentes señales perturbadas. La Figura 8.a) representa la gráfica de los CD de una señal de onda perfectamente sinusoidal con una amplitud y frecuencia constante, en la cual los puntos de máximo oscilan en el rango de (0.0047; 0.0049) y los de mínimo en el rango de (-0.0047; -0.0049).

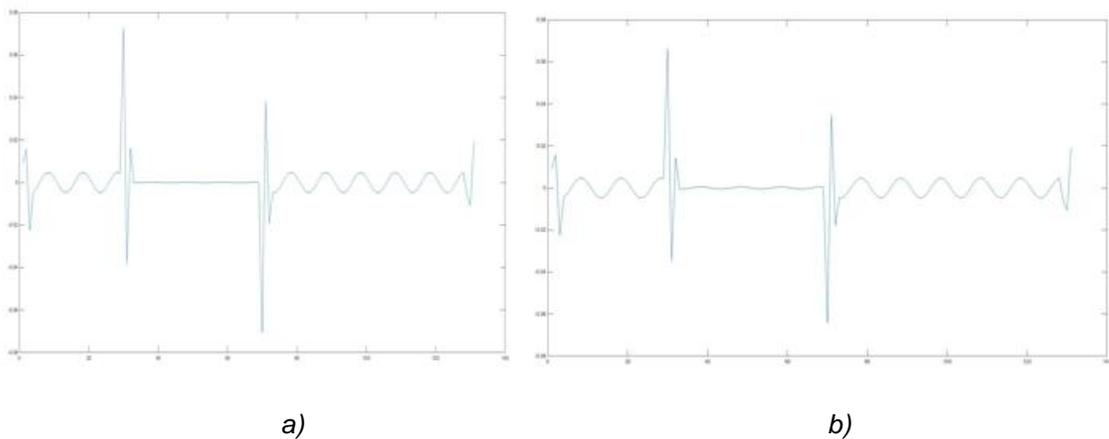
Teniendo en cuenta el concepto de perturbación cualquier desviación de estos valores representa un disturbio eléctrico; como se puede apreciar en la Figura 8.b) y en la Figura 9.a) y 9.b) donde los picos de máximo y mínimo indican el inicio y fin de cada perturbación.

Figura 7: Salida wavelet de los CD, a) Señal ideal, b) Señal perturbada 1.



Fuente: (elaboración propia).

Figura 8: Salida wavelet de los CD, a) Señal perturbada 2, b) Señal perturbada 3.



Fuente: (elaboración propia).

Una vez procesada la señal de voltaje, se conoce si hay existencia o no de perturbaciones. En caso de ser positivo se procede a la segunda etapa del proceso de identificación.

2.3 Etapa 2. Extracción y optimización de características

2.3.1 Selección de las características de la señal

Como se explicó en el Capítulo 1 la selección de un conjunto apropiado de características es un paso fundamental hacia una detección y clasificación de perturbaciones exitosa. Como no todas arrojan resultados satisfactorios cuando se trata de señales, donde la presencia del ruido y demás factores de perturbación afectan los componentes principales de la misma, se propone el uso de la energía de la señal

como principal característica. Esta decisión se toma porque la energía es una característica que puede ser fácil de extraer, tanto en el dominio del tiempo como en el dominio de la frecuencia, y además se conserva al transformar la señal de un dominio a otro como lo plantea en la identidad de Parseval (James, 2002).

2.3.2 Extracción de la energía de la señal utilizando TWD

La TWD permite calcular la energía de la señal a través de sus coeficientes (CA y CD) mediante la identidad de Parseval, la cual establece que la energía de la señal es igual a la suma de los coeficientes al cuadrado de la transformada, como se muestra en (Shukla & Tiwari, 2013)

$$E_{señal} = \sum_{k=-\infty}^{\infty} |CA(k)|^2 + \sum_{j=0}^{\infty} \sum_{k=-\infty}^{\infty} |CD_j(k)|^2.$$

Donde cada j representa una escala de descomposición wavelet y cada k un coeficiente de detalle a una escala j específica.

Según (Li, Monti, & Ponci, 2014) la expresión $E_{señal}$ puede verse como la energía de los CA más la suma de la energía de los CD en cada escala j como se expresa a continuación:

$$E_{señal} = E_{CA_j} + \sum_{j=0}^{\infty} E_{CD_j}, \text{ con}$$

$$E_{CA_j} = \sum_{k=-\infty}^{\infty} |CA(k)|^2 \text{ y } E_{CD_j} = \sum_{k=-\infty}^{\infty} |CD_j(k)|^2.$$

Donde E_{CA_j} es la energía de los CA y E_{CD_j} la energía de los CD en la escala j .

Como función wavelet para la extracción de los coeficientes se usará la wavelet de Daubechies (db4) propuesta en (Milchevski, Kostadinov, & Taskovski, 2013) y (Praveen & Pramond, 2015).

2.3.3 Extracción de vectores de energía para la clasificación

En (Li, Monti, & Ponci, 2014) se propone para la clasificación, el uso de un vector de características formado por la diferencia entre la energía acumulada en los CD en cada escala de la señal perturbada y la energía acumulada en los CD en cada escala de la señal ideal. Matemáticamente queda representado de la siguiente manera:

$$X = E_{perturbada} - E_{ideal} = \Delta E, \text{ con}$$

$$E_{perturbada} = \begin{bmatrix} E_{perturbada_{d_1}} \\ \vdots \\ E_{perturbada_{d_j}} \\ \vdots \\ E_{perturbada_{d_j}} \end{bmatrix}, E_{ideal} = \begin{bmatrix} E_{ideal_{d_1}} \\ \vdots \\ E_{ideal_{d_j}} \\ \vdots \\ E_{ideal_{d_j}} \end{bmatrix} \text{ y } \Delta E = \begin{bmatrix} \Delta E_{d_1} \\ \vdots \\ \Delta E_{d_j} \\ \vdots \\ \Delta E_{d_j} \end{bmatrix}.$$

Donde $E_{perturbada}$ y E_{ideal} son los vectores que contienen en sus componentes la energía acumulada de los CD en cada escala de la señal perturbada e ideal respectivamente y ΔE es el vector de energía resultante. Una vez extraídos los vectores de características se procede entonces a la optimización de los mismos, basada en la normalización (Ver epígrafe 1.4.2.1).

2.4 Etapa 3. Selección de la topología o arquitectura de clasificación

Para la clasificación se propone el uso del algoritmo k -NN, debido a que el mismo es extremadamente simple y está abierto a una amplia variedad de variaciones. Es intuitivo, preciso y aplicable a diversos problemas como se plantea en (Yu, Ji, & Zhang, 2002). En el momento en que se aplica el algoritmo a un problema, la primera decisión que se debe tomar es la medida de distancia y la segunda el valor de k .

Como métrica a usar para el funcionamiento del mismo se propone la distancia en el espacio Kernel, la cual, mientras se extraigan las características adecuadas aumenta la probabilidad de lograr una discriminación lineal entre clases en el nuevo espacio de proyección. Como función Kernel se propone el uso del Kernel Gaussiano visto en la tabla 1, esta función arroja buenos resultados en el procesamiento de señales eléctricas como se puede apreciar en (De Yong, Bhowmik, & Magnago, 2015) y (Muniru Olajide, 2015).

Según (Wang L. , 2005) la distancia al cuadrado en el espacio Kernel se puede definir como:

$$d(x, y) = \|\varphi(x) - \varphi(y)\|^2 = \langle \varphi(x), \varphi(x) \rangle - 2\langle \varphi(x), \varphi(y) \rangle + \langle \varphi(y), \varphi(y) \rangle.$$

Usando funciones Kernel para obtener el producto escalar la distancia quedaría expresada de la siguiente manera.

$$d(x, y) = \|\varphi(x) - \varphi(y)\|^2 = K(x, x) - 2K(x, y) + K(y, y).$$

Y mediante el uso del Kernel gaussiano $K(x, y) = e^{(-\gamma\|x-y\|^2)}$ se reduce a:

$$d(x, y) = \|\varphi(x) - \varphi(y)\|^2 = 2 - 2K(x, y) = 2 - 2e^{-(\gamma\|x-y\|^2)}.$$

Donde γ es un parámetro que ajusta el Kernel Gaussiano a la dispersión de los datos.

En cuanto al valor de k se propone escoger un número impar o primo para minimizar la posibilidad de empates en las votaciones. Plantean en (Benítez, Escudero, Samir, & David, 2014) que para el tratamiento de los empates cuando la k es superior a uno, algunos heurísticos posibles son: no dar predicción en caso de empate o dar la clase más frecuente en el conjunto de entrenamiento de entre las clases seleccionadas para votar.

2.5 Etapa 4. Realización del aprendizaje automático

El modelo creado a través de k -NN se compone solamente de los datos etiquetados disponibles, colocados en algún espacio métrico. En otras palabras, es una excepción al flujo de trabajo general para la construcción de modelos supervisados de aprendizaje automático.

De manera general, para k -NN, no hay una etapa de entrenamiento porque no hay un modelo para construir.

2.6 Etapa 5. Validación y prueba

La validación mide la precisión del modelo contra los datos de entrenamiento. Debido a que en k -NN no se construye ningún modelo, no hay nada que validar; pero aun así es posible ajustar los parámetros del clasificador, a través de un proceso de validación cruzada.

2.6.1 Estimación de los parámetros γ y k para el algoritmo k -NN

Para estimar un valor adecuado de los parámetro γ y k se propone el uso de la técnica conocida como validación cruzada (ver epígrafe 1.4.5), con $N = 10$ iteraciones. Esta técnica permite estimar de manera muy precisa el valor de los parámetros, pues evalúa la capacidad de generalización del clasificador con N combinaciones de datos de entrenamiento y prueba, quedándose al final con los valores que arrojan mejor resultado. En el caso del parámetro γ se recomienda el uso de los valores siguientes: $\gamma = [2^{-5}, 2^{-4}, \dots, 2^0, \dots, 2^4]$ propuestos en (Wang L. , 2005). Para el parámetro k se utilizaran los dígitos impares $k = [3,5,7,9]$ de acuerdo a lo expresado en (Larose, 2005).

Para realizar la validación cruzada es necesario definir un conjunto de vectores de características pre-etiquetados que identifiquen cada perturbación. Los vectores serán extraídos a partir de la señales generadas por los modelos propuestos en (Li, Teng, Tang, & Song, 2016) mostrados a continuación.

Tabla 2: Modelos de ecuaciones paramétricas para perturbaciones eléctricas.

Perturbaciones	Ecuaciones	Parámetros
Señal ideal	$s(t) = \text{sen}(\omega_0 t)$	$\omega_0 = 2 * \pi * 50$
Bajada de voltaje	$s(t) = \left(1 - a((t - t_1) - (t - t_2))\right) * \text{sen}(\omega_0 t)$	$0.1 \leq a \leq 0.9, T \leq t_2 - t_1 \leq 9T$

Subida de voltaje	$s(t) = \left(1 + a((t - t_1) - (t - t_2))\right) * \text{sen}(\omega_0 t)$	$0.1 \leq a \leq 0.8, T \leq t_2 - t_1 \leq 9T$
Interrupción	$s(t) = \left(1 - a((t - t_1) - (t - t_2))\right) * \text{sen}(\omega_0 t)$	$0.9 < a \leq 1, T \leq t_2 - t_1 \leq 9T$
Armónicos	$s(t) = \text{sen}(\omega_0 t) + \sum_{i=2}^n k_i \text{sen}(i\omega_0 t)$	$0.05 < k_i \leq 0.3, i = 3,5,7$
Transitorios	$s(t) = \text{sen}(\omega_0 t) + ae^{\left(-\frac{(t-t_1)}{\tau}\right)}((t - t_1) - (t - t_2)) * \text{sen}(2\pi ft)$	$0.1 \leq a \leq 0.8, 0.5T \leq t_2 - t_1 \leq 3T, 300 \leq f \leq 900$
Fluctuaciones de voltaje	$s(t) = (1 + a \text{sen}(2\pi\beta t)) * \text{sen}(\omega_0 t)$	$0.1 \leq a \leq 0.2, 0.1 \leq \beta \leq 0.5$
Bajada de voltaje + armónicos	$s(t) = \left(1 - a((t - t_1) - (t - t_2))\right) * (\text{sen}(\omega_0 t) + \sum_{i=2}^n k_i \text{sen}(i\omega_0 t))$	$0.1 \leq a \leq 0.9, T \leq t_2 - t_1 \leq 9T, 0.05 < k_i \leq 0.3, i = 3,5,7$
Subida de voltaje + armónicos	$s(t) = \left(1 + a((t - t_1) - (t - t_2))\right) * (\text{sen}(\omega_0 t) + \sum_{i=2}^n k_i \text{sen}(i\omega_0 t))$	$0.1 \leq a \leq 0.8, T \leq t_2 - t_1 \leq 9T, 0.05 < k_i \leq 0.3, i = 3,5,7$

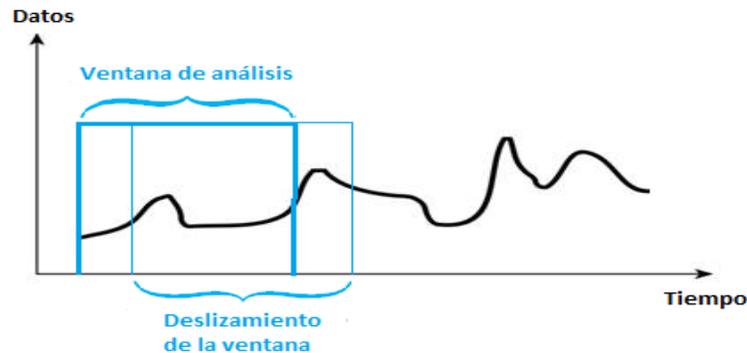
Fuente: (Li, Teng, Tang, & Song, 2016)

2.7 Propuesta del algoritmo

El algoritmo que se propone para la detección y clasificación de perturbaciones eléctricas hace uso de una ventana de análisis deslizante (Ver figura 9), con un ancho fijo de $l = 128$ muestras de voltaje y 0.005 segundos de separación entre cada muestra, esto equivalente a una ventana de 0.64 segundos de duración. Esta ventana se desliza con un paso igual a 32 muestras de voltaje, lo que es equivalente a 0.16 segundos.

Dada entonces la señal discreta $f[KT] = [h_0, h_T, h_{2T}, \dots, h_{nT}]$ con $K = 0,1,2, \dots, n$ y T el intervalo de tiempo entre dos muestras consecutivas, la ventana permite el análisis de la señal en cada intervalo $[h_i, h_{i+l}]$ con $i = 0,1,2, \dots, n - 128$.

Figura 9: Ventana deslizante.



Fuente: (elaboración propia).

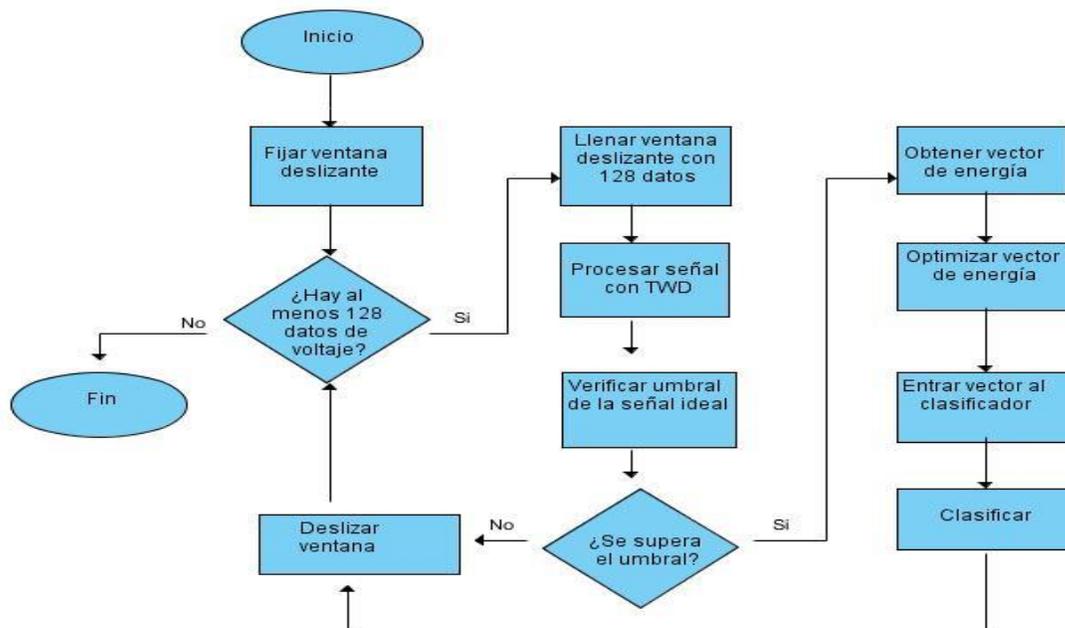
Los pasos para realizar el algoritmo pueden desglosarse de la siguiente manera.

- 1) Entrenamiento del clasificador:
 - a) Generar señales discretas asociadas a cada perturbación modelada en la tabla 2, con un intervalo de muestreo de 0.005 segundos (puede ser menor o igual que 0.008 segundos según el teorema de Nyquist-Shannon).
 - b) Mediante la TWD realizar el AMR de cada señal obtenida.
 - c) Por cada AMR, obtener la energía concentrada en los CD por cada escala y conformar un vector de características con estas energías.
 - d) Hallar el vector diferencia entre cada uno de estos vectores y el vector de energía asociada a los CD por cada escala de la señal de voltaje ideal.
 - e) Normalizar los vectores resultantes de energía.
 - f) Estimar el parámetro γ del Kernel gaussiano y el parámetro k del algoritmo k -NN.
- 2) Digitalización de la señal real:
 - a) Empezar proceso de digitalización de la señal de voltaje distribuido con un periodo de muestreo igual al definido en el paso 1. a
 - b) Iniciar lectura online de los datos h_i digitalizados de la señal.
 - c) Leídos los primeros 128 datos iniciar el procesamiento de la señal dentro de la ventana de análisis situada en el intervalo $[h_i; h_{i+l}]$, con $i = 0$.
- 3) Procesamiento de la señal:
 - a) Obtener los CD de la señal en el intervalo $[h_i; h_{i+l}]$.

- b) Detectar la presencia de una perturbación según lo expuesto en el epígrafe 2.3. En caso de detectar una perturbación realizar paso 3. c. En caso contrario, deslizar la ventana de análisis 0.16 segundos y regresar al paso 3. a.
 - c) Realizar el AMR de los datos definidos en el intervalo.
- 4) Clasificación de perturbaciones:
 - a) Obtener la energía concentrada en los CD por cada escala y conformar un vector de características con estas energías.
 - b) Hallar el vector diferencia entre cada uno de estos vectores y el vector de energía asociada a los CD por cada escala de la señal de voltaje ideal.
 - c) Normalizar los vectores resultantes de energía.
 - 5) Introducir el vector de energía al clasificador y clasificar la posible perturbación.
 - 6) Desplazar la ventana de análisis 0.16 segundos y volver al paso 3. a.

En la siguiente figura se puede observar una síntesis del algoritmo mediante un diagrama de flujo y a continuación el pseudocódigo correspondiente al mismo.

Figura 10: Diagrama de flujo del algoritmo.



Fuente: (elaboración propia).

Figura 11: Seudocódigo del algoritmo.

```
1 //L Lista con los datos de voltaje de la señal
2 //
3 //extraer(L, pos1, pos2) función encargada de extraer de una lista L
4 //los elementos desde la posición pos1 hasta la posición pos2
5 //
6 //extraerCD(output) función encargada de extraer los CD wavelet del vector
7 //de datos output arrojado por la función dwt de la biblioteca wavelet1D
8 //
9 //calcularEnergia(CD) función encargada de extraer la energía de los CD
10 //
11 //optimizar(energia) función encargada de optimizar un vector de energía
12 //
13 //KNN(energia, K) algoritmo de clasificación KNN con K vecinos mas cercanos
14 proceso clasificar(L)
15     pos1 = 0
16     pos2 = 127
17     L1 = extraer(L, pos1, pos2)
18     Mientras L1.Longitud == 128 Hacer
19         dwt(L1, level, "db4", output, flag, length)
20         CD = extraerCD(output)
21         Si max(CD) > umbralMax o min(CD) < umbralMin Entonces
22             energia = calcularEnergia(CD)
23             optimizar(energia)
24             resultado = KNN(energia, K)
25             Escribir resultado
26         Sino
27             Escribir 0
28         FinSi
29         pos1 = pos1 + 32
30         pos2 = pos2 + 32
31         L1 = extraer(L, pos1, pos2)
32     FinMientras
33     Escribir -1
34 FinProceso
```

Fuente: (elaboración propia).

Conclusiones parciales

El algoritmo propuesto toma ventaja de la TWD, para obtener los CA y CD mediante la técnica de AMR, los cuales hacen posible calcular la energía de la señal a varias resoluciones y dicha energía es utilizada para la detección y clasificación de las perturbaciones eléctricas.

La clasificación con el algoritmo k -NN utilizando distancia Kernel aumenta la probabilidad de lograr una discriminación lineal entre clases en el nuevo espacio de proyección y lograr que el mismo sea más preciso. A través de la técnica de validación cruzada se pueden estimar los parámetros γ del Kernel Gaussiano y k del algoritmo k -NN y se obtiene una predicción del error de generalización del clasificador.

CAPÍTULO 3: EVALUACIÓN DE LA SOLUCIÓN PROPUESTA

Introducción

Luego de presentar la propuesta de un algoritmo para la detección y clasificación de perturbaciones eléctricas en Sistemas de Supervisión, Control y Adquisición de Datos se hace necesario validar su correcto funcionamiento, es decir, que tenga las características indispensables para ser considerado un buen algoritmo, como son: ser preciso y aplicable al contexto para el cual fue realizado. Algunas de las métricas que se utilizan para la validación se exponen en el presente capítulo.

3.1 Validación del algoritmo basado en métricas

Específicamente para el caso de un algoritmo de aprendizaje automático son varias las métricas utilizadas para su evaluación. La métrica conocida como precisión de la clasificación es una de las más usadas para medir el rendimiento del algoritmo de clasificación, sin embargo, no es suficiente para juzgar realmente que tan exacto puede ser la solución, debido a lo cual suelen usarse más métricas de prueba como es el caso de la Matriz de confusión (Kota Gopalakrishna & Liotta, 2017).

3.1.1 Precisión de clasificación

La precisión de la clasificación es una métrica usada para evaluar el nivel de precisión del algoritmo durante la clasificación de los datos. Se define como la razón entre el número de predicciones correctas y el número total de muestras de entrada (Kota Gopalakrishna & Liotta, 2017).

$$precisión = \frac{\text{clasificaciones correctas}}{\text{total de clasificaciones}}$$

Esta métrica fue empleada durante el proceso de validación cruzada, con 10 iteraciones, para estimar la capacidad de generalización del algoritmo y ajustar los parámetros k del algoritmo de clasificación k -NN y γ empleado en la distancia Kernel.

Para ello se generaron 60 perturbaciones por cada clase, para un total de 540 perturbaciones. Se dividieron en 10 grupos de manera tal que en cada uno quedara la mayor representación de perturbaciones por clase. Para cada valor de γ y k dentro de los conjuntos $[2^{-5}, 2^{-4}, \dots, 2^0, \dots, 2^4]$ y $[3,5,7,9]$ respectivamente se aplicó la técnica de validación cruzada. Se calculó la *precisión* en cada iteración del proceso para luego estimar la capacidad de generalización del algoritmo, obteniéndose como valor más alto un 98% de precisión de la clasificación con los parámetros ajustados de la siguiente manera ($k = 3$ y $\gamma = 2^{-5}$).

3.1.2 Matriz de confusión

Como su nombre lo indica la Matriz de confusión arroja como salida una matriz, que contiene información sobre los resultados reales y pronosticados dados por un clasificador (Liu, 2007).

Es una herramienta que permite la visualización del desempeño de un algoritmo que se emplee en el aprendizaje supervisado. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real y la diagonal principal contiene la cantidad de predicciones correctas para cada clase. Uno de los beneficios de las matrices de confusión es que facilitan ver si el algoritmo está confundiendo dos o más clases.

Para generar la matriz de confusión se generaron 200 perturbaciones diferentes por cada modelo de perturbación propuesto en la tabla 2, para un total de 1800 perturbaciones a clasificar con el algoritmo propuesto. Para la creación de la matriz de confusión las perturbaciones fueron etiquetadas de la siguiente manera:

P1: señal ideal

P2: bajada de voltaje

P3: subida de voltaje

P4: interrupción

P5: armónicos

P6: transitorios

P7: fluctuaciones de voltaje

P8: bajada de voltaje más armónicos

P9: subida de voltaje más armónicos

	P1	P2	P3	P4	P5	P6	P7	P8	P9	%
P1	200	0	0	0	0	0	0	0	0	100
P2	0	160	0	30	0	0	8	2	0	80
P3	0	0	190	0	0	0	7	0	3	95
P4	0	27	0	163	0	0	0	10	0	81
P5	0	0	0	0	200	0	0	0	0	100
P6	0	0	6	0	0	189	5	0	0	94

P7	0	0	7	0	0	6	187	0	0	93
P8	0	0	0	0	2	0	0	198	0	99
P9	0	0	3	0	0	0	0	0	197	98
% promedio										93

Tabla 3: Matriz de confusión

Según muestra la matriz, son de especial atención algunas perturbaciones específicas. El caso de la señal ideal y los armónicos, donde la precisión de la clasificación es bastante buena, arrojando un 100% de exactitud. Por otro lado en las señales de bajada de voltaje e interrupción se obtuvo el porcentaje de clasificaciones correctas más bajo, con un 80% y un 81% respectivamente. Esto es debido a que ambas señales tienden a confundirse entre sí, lo cual es normal porque son muy parecidas.

Promediando los porcentajes obtenidos en la matriz, también es posible obtener una evaluación general de la precisión del algoritmo; obteniéndose como resultado un 93% de exactitud en las clasificaciones correctas.

3.2 Validación del algoritmo con prototipos

Entre las formas más usuales de analizar un sistema se encuentra el diseño de prototipos. Estos representan una versión inicial del sistema, sencilla, con fines de demostración, evaluación y optimización de tiempo para posteriores etapas del Proceso de Desarrollo de Software (PDS). Al desarrollarlos se deben crear con rapidez, proveer un costo bajo de desarrollo y permitir su evolución a través de un proceso iterativo. Siempre se debe establecer cuál es su objetivo, ya que puede ser útil en diferentes fases del proyecto.

Una ventaja fundamental que presenta la construcción de prototipos desde el punto de vista de la validación radica en que estos modelos, una vez construidos, pueden ser evaluados directamente por los usuarios o expertos en el dominio para validar sobre ellos el análisis y el diseño del sistema (Kendall & Kendall, 2005).

3.2.1 Propósito de los prototipos

En la fase de Análisis de un proyecto, su principal propósito es obtener y validar los requerimientos esenciales, manteniendo abiertas las opciones de implementación. Esto implica que se deben tomar los comentarios de los usuarios, pero se debe regresar a sus objetivos para no perder la atención. En la

fase de Diseño su propósito, basándose en los requerimientos previamente obtenidos, es mostrar las ventanas, su navegación, interacción, controles y botones al usuario y obtener una retroalimentación que permita mejorar el Diseño de Interfaz (Kendall & Kendall, 2005).

3.2.2 Clases de prototipos

La elaboración de prototipos se puede aplicar de manera conveniente ante una situación particular, en lugar de sintetizar todos sus usos en una sola definición. A continuación, se muestran varias concepciones propuestas en (Kendall & Kendall, 2005) para la elaboración de prototipos.

- ❖ **Prototipo corregido:** Representa la construcción de un sistema que funciona pero se corrige simultáneamente. En la ingeniería a este enfoque se le denomina: elaboración de una tabla experimental.
- ❖ **Prototipo no funcional:** Es un modelo no funcional, configurado para probar ciertos aspectos del diseño.
- ❖ **Primer prototipo de una serie:** Involucra la creación de un primer modelo a escala completa de un sistema llamado piloto.
- ❖ **Prototipo de características seleccionadas:** Es la creación de un modelo funcional que incluye algunas de las características que tendrá el sistema final.

La clase de prototipo que se utiliza para validar el algoritmo propuesto es el prototipo de características seleccionadas. Utilizando este modelo se podrá simular el comportamiento del algoritmo en un entorno real.

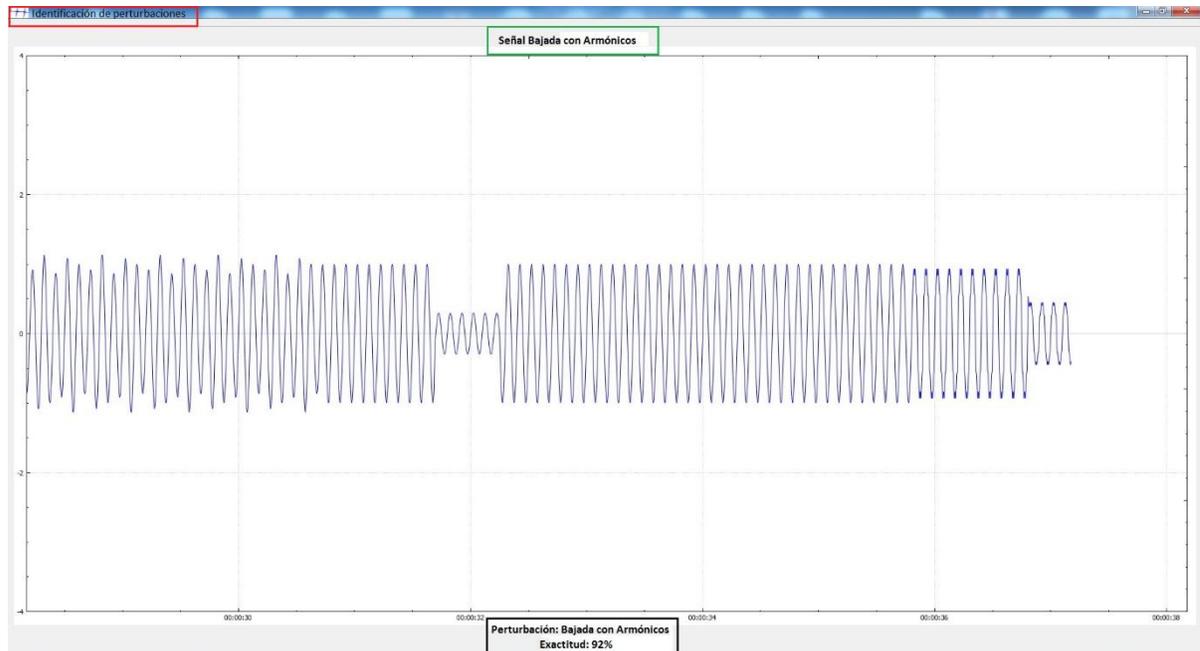
3.2.3 Prototipo de características seleccionadas

En el presente epígrafe se muestran imágenes del prototipo construido para validar el algoritmo de detección y clasificación de perturbaciones eléctricas.

En la Figura 9, se muestra el prototipo realizado, a continuación se expone cada elemento del mismo a través de una leyenda.

- ❖ Rectángulo rojo (situado en la parte superior izquierda): encierra el logo y el nombre del prototipo.
- ❖ Rectángulo verde (centrado en la parte superior): muestra la perturbación real que se está simulando en ese instante de tiempo.
- ❖ Rectángulo negro (centrado en la parte inferior): muestra la salida del algoritmo y el porcentaje de exactitud basado en las últimas 50 clasificaciones realizadas por el algoritmo.

Figura 12: Prototipo de características seleccionadas.



Fuente: (elaboración propia).

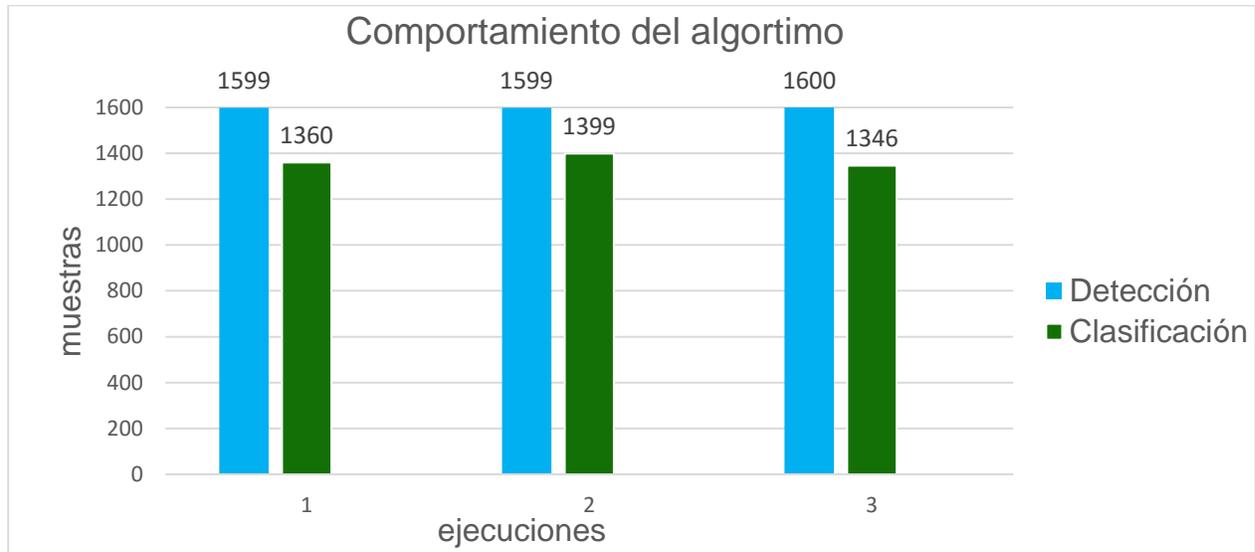
3.3 Pruebas del algoritmo utilizando el prototipo

El prototipo es capaz de generar las nueve clases de perturbaciones vistas en la tabla 2 de manera aleatoria. Cada perturbación es generada durante un tiempo de 2,555 segundos, equivalente a 512 datos con un periodo de 0.005 segundos entre datos consecutivos.

Para simular el comportamiento del algoritmo en un ambiente real se ejecutó el prototipo en tres ocasiones durante 255 segundos, tiempo necesario para generar 100 perturbaciones consecutivas de manera aleatoria, incluyendo la señal ideal. Se utilizó una ventana de análisis deslizante de ancho igual a 0.64 segundos con paso de 0.16 segundos para un análisis en línea de las perturbaciones.

Esta configuración permite analizar una muestra de 128 datos de la señal cada 0.16 segundos, para un total de 1600 muestras consecutivas. El resultado del análisis se muestra en la figura 10 y estuvo basado en la cantidad de muestras detectadas correctamente como perturbaciones o no y en la cantidad de muestras clasificadas correctamente en correspondencia al tipo de perturbación a la cual pertenecía.

Figura 13: Resultado de las pruebas basadas en el prototipo.



Fuente: (elaboración propia).

- ❖ En la primera ejecución se detectaron 1599 muestras correctamente para un 99.9% de exactitud en la detección de perturbaciones y se clasificaron 1360 muestras correctamente para un 85% de exactitud en la clasificación.
- ❖ En la segunda ejecución se detectaron 1599 muestras correctamente para un 99.9% de exactitud en la detección de perturbaciones y se clasificaron 1399 muestras correctamente para un 87% de exactitud en la clasificación.
- ❖ En la tercera ejecución se detectaron 1600 muestras correctamente para un 100% de exactitud en la detección de perturbaciones y se clasificaron 1346 muestras correctamente para un 84% de exactitud en la clasificación.

3.4 Análisis de la validación del algoritmo

El algoritmo analizado consta de dos procesos fundamentales, la detección y la clasificación. Con las pruebas realizadas al mismo se pudo constatar la excelente precisión en la detección de perturbaciones, arrojando siempre un resultado por encima del 99% de exactitud. En cuanto a la clasificación se puede apreciar que los resultados difieren. La evaluación con el prototipo muestra que los resultados son inferiores a los obtenidos con la matriz de confusión. Esto se debe a que el algoritmo recibe un conjunto de perturbaciones por separado, para confeccionar la matriz de confusión, con el objetivo de clasificar cada perturbación de manera independiente. En el caso del prototipo el algoritmo recibe muestras híbridas de

perturbaciones (muestras que contienen en su interior dos perturbaciones consecutivas) debido al uso de la ventana deslizante y al tener que decidir entre una u otra perturbación puede llegar a confundirlas, constituyendo un caso crítico para la clasificación.

Conclusiones parciales

A partir de las pruebas realizadas al algoritmo propuesto se puede afirmar que este muestra un buen desempeño usando TWD y k -NN en la detección y clasificación de perturbaciones eléctricas. Se obtuvo un 93% de exactitud en la clasificación realizando la matriz de confusión; este resultado es comparable con algoritmos similares propuestos en la literatura, como es el caso del algoritmo propuesto por (Chicheng, Li, & Leping, 2015) con un 90.25% de exactitud y el propuesto por (Praveen & Pramond, 2015) con un 92% de exactitud realizando igualmente la matriz de confusión.

CONCLUSIONES

En el estudio realizado se confirma que varios de los sistemas *SCADA* comercializados en la actualidad distan de ser entornos de supervisión al no contar con las etapas de detección de fallos, diagnóstico de estos y reconfiguración del sistema.

Al término de la investigación se obtuvo finalmente la propuesta de un algoritmo que:

- ❖ Toma ventaja de la TWD para obtener los CA y CD mediante la técnica de AMR, los cuales hacen posible calcular la energía de la señal a varias resoluciones y utilizar dicha energía para la detección y clasificación de las perturbaciones eléctricas.
- ❖ Utiliza un conjunto de señales modeladas propuestas en (Li, Teng, Tang, & Song, 2016) que permitieron validar la precisión del algoritmo propuesto para la detección y clasificación de perturbaciones eléctricas.
- ❖ Al desempeñarse con un 93% de precisión en la clasificación es comparable con algoritmos similares propuestos en la literatura, como es el caso del algoritmo propuesto por (Chicheng, Li, & Leping, 2015) con un 90.25% de exactitud y el propuesto por (Praveen & Pramond, 2015) con un 92% de exactitud.

RECOMENDACIONES

Debido a las ventajas que ofrece la aplicación del algoritmo propuesto para la detección y clasificación de perturbaciones eléctricas, se sugiere al seguimiento del mismo y por tal razón se proponen las siguientes recomendaciones:

- ❖ Integrar el algoritmo al sistema *SCADA SAINUX 1.0* desarrollado en la Universidad de las Ciencias Informáticas (UCI).
- ❖ Realizar un estudio de otros métodos de clasificación supervisada que puedan emplearse en el algoritmo propuesto.

REFERENCIAS BIBLIOGRÁFICAS

- Abe, S. (2010). Support Vector Machines for Pattern Classification . London: Springer.
- Baran Roy, R. (2010). Application of SCADA for Controlling Electrical. UITS Journal , 1(2).
- Bárzaga, L., Mompie, R. C., & Valdés, B. (2016). Sistemas SCADA para la automatización de los procesos productivos del CIGB. Revista de Ingeniería Electrónica, Automática y Comunicaciones (RIELAC), 7-8.
- Basogain Olabe, X. (1998). Redes Neuronales Artificiales y sus Aplicaciones. Publicaciones de la Escuela de Ingenieros .
- Benítez, R., Escudero, G., Samir, K., & David, M. R. (2014). Inteligencia Artificial Avanzada. Universitat Oberta de Catalunya (UOC).
- Boashash, B. (2003). Time Frequency Signal Analysis and Processing A Comprehensive Reference. Brisbane,Australia: Elsevier.
- Bollen, M., & Yu-Hua Gu, I. (2006). Signal processing of power quality disturbances. United States of America: IEEE Press.
- Cai, L., & Thornihill, N. F. (2017). Real-Time Detection of Power System Disturbances Based on k-Nearest Neighbor Analysis. IEEE.
- Camps-Valls, G., & Bruzzone, L. (2009). Kernel Methods for Remote Sensing Data Analysis. Reino Unido: WILEY.
- Castañeda, A., & Paternina, J. (2013). Procesamiento embebido de señales cerebrales relacionadas con la imaginación de movimientos para aplicaciones de BCI. Tesis presentada a la Universidad Javeriana como requisito parcial para optar por el título de Ingeniero Electrónico. Bogotá, Colombia.
- Cedeño Pozo, A. (2014). Algoritmo para la reducción de ruido en señales industriales mediante la Transformada Wavelet Discreta. La Habana, La Habana, Cuba.
- Chacon, D., Dijort, O., & Castrillo, J. (2001). Supervisión y control de procesos. EUPVG-UPC.
- Chang, C.-C. a.-J. (2011). LIBSVM: A library for support vector machines. Transactions on Intelligent Systems and Technology (TIST), 2(3), 27.
- Chavarría Meza, L. (2007). Scada System`s & Telemetry México. Atlantic International University.

- Chicheng, X., Li, X., & Leping, B. (2015). An Effective S-transform Feature Extraction Method for Classification of Power Quality Disturbance Signals. *IEEE*, 1555-1560.
- Colomer, J., Meléndez, J., & Ayza, J. (2003). Introducción a la monitorización y supervisión experta de procesos. *Métodos y herramientas. CEA-IFAC*, 72.
- Daubechies, I. (1990). The wavelet transform, time-frequency localization and signal analysis. *IEEE*, 961-1005.
- De Yong, D., Bhowmik, S., & Magnago, F. (2015). An effective Power Quality classifier using Wavelet Transform and Support Vector Machines. *Expert Systems with Applications*, 42(15), 6075--6081.
- Dehghani, M. J. (2009). Comparison of S-transform and wavelet transform in power quality analysis. *World Academy of Science, Engineering and Technology*, 50(4), 395--398.
- Díaz Tribaldos, M. R., Escobar Ocampo, J. M., & Vivas Albán, Ó. A. (2015). Interfaz háptica tipo guante con realimentación vibratoria. *Revista EIA*.
- Eichhammer, E. (2016, septiembre 13). Qt Plotting Widget QCustomPlot. Retrieved from <http://www.qcustomplot.com>
- Gilat, A. (2005). *Matlab: Una introducción con ejemplos prácticos*. Barcelona: Reverté.
- Gómez Chova, L. (2002). *Métodos de clasificación aplicados al reconocimiento de campos de cultivo a partir de imágenes hiperespectrales*. Valencia: GPDS.
- Hussain, R. (2011). Retrieved from C++ Wavelet Libraries: <http://wavelet2d.sourceforge.net/>
- Jalote, P. (2008). *A Concise Introduction to Software Engineering*. Delhi: Springer.
- James, G. (2002). *Matemáticas avanzadas para ingeniería*. México: Pearson educación.
- Jianmin Teng, L., Zhaosheng, T., & Qiu Song, J. (2016). Detection and Classification of Power Quality Disturbances Using Double Resolution S-Transform and DAG-SVMs. *IEEE*.
- Jin, D., & Lin, S. (2012). *Advances in Electronic Engineering, Communication and Management* Vol. 1. Beijing: Springer.
- Karasu, S., & Sarac, Z. (2017). Classification of Power Quality Disturbances with S-Transform and Artificial Neural Networks Method. *IEEE*, 1-4.
- Kendall, K. E., & Kendall, J. E. (2005). *Systems Analysis and Desing*. Pearson Education.

- Kota Gopalakrishna, A., & Liotta, A. (2017). Evaluating Machine Learning Algorithms for Applications with Humans in the Loop. *IEEE*, 459-464.
- Krutz, R. L. (2005). *Securing SCADA Systems*. John Wiley & Sons.
- Kumar, R., Singh, B., Shahani, D., & Chandra, A. (2013). Recognition of Power Quality Events Using S-Transform Based ANN Classifier and Rule Based Decision Tree. *Industry Applications Society Annual Meeting, IEEE*, 1-8.
- Larose, D. T. (2005). *Discovering Knowledge in Data. An Introduction to Data Mining*. New Jersey: Wiley.
- Li, J., Teng, Z., Tang, Q., & Song, J. (2016). Detection and Classification of Power Quality Disturbances Using Double Resolution S-Transform and DAG-SVMs. *IEEE*, 65(10), 2302--2312.
- Li, W., Monti, A., & Ponci, F. (2014). Fault detection and classification in medium voltage DC shipboard power systems with wavelets and artificial neural networks. *IEEE*, 2651-2665.
- Liu, B. (2007). *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. Springer Science.
- Lizarraga de Miguel, A. (2009). *Implantación del software Power Studio Scada en una empresa para la gestión de la energía*. Universidad Politécnica de Catalunya.
- M. Gargoom, A., Ertugrul, N., & L. Soong, W. (2008). Automatic Classification and Characterization of Power Quality Events. *IEEE*, 23(4), 2417-2425.
- Madisetti, V., & Williams, D. (1999). *Digital signal processing*. Georgia: CRC Press.
- Mahela Prakash, O., & Shaik Gafoor, A. (2016). Recognition of Power Quality Disturbances Using S-Transform and Rule-Based Decision Tree. *IEEE*, 1-6.
- Marínez, F., Peris, A., & Rodenas, F. (2004). *Tratamiento de Señales Digitales mediante Wavelets y su uso con MATLAB*. Valencia: Club Universitario.
- Math H, B., & Gu, I. (2006). *Signal processing of power quality disturbances*. John Wiley & Sons.
- Milchevski, A., Kostadinov, D., & Taskovski, D. (2013). Classification of Power Quality Disturbances using Wavelets and Support Vector Machine. *IEEE*, 19(2), 25-30.

- Mohan Saha, M., Izykowski, J., & Rosolowski, E. (2010). *Fault Location on Power Networks*. New York: Springer.
- Morales España, G., Mora Flórez, J., & Vargas Torres, H. (2008). k-NN based regression strategy used to estimate the fault distance in radial power systems. *Revista Facultad de Ingeniería Universidad de Antioquia*, 100--108.
- Moreno, A., Bellido, F., Oterino, D., & de la Rosa, J. (2008). Power Quality monitoring integration into distribution automation through the use of AMR. *IEEE*, 1-6.
- Muniru Olajide, O. (2015). Detection and Classification of Power Quality Event using Discrete Wavelet Transform and Support Vector Machine. 4(6).
- Oppenheim, A. V., & Schaffer, R. W. (2009). *Tratamiento de señales en tiempo discreto*. Madrid: Pearson Education.
- Poularikas, A. (2010). *Transforms and applications*. California: CRC Press.
- Praveen, S., & Pramond, A. (2015). A Neural Network Based Power Quality Signal Classification System using Wavelet Energy Distribution. *IEEE*, 199-204.
- Rebizant, W., Szafran, J., & Wiszniewski, A. (2011). *Digital Signal Processing in Power System Protection and Control*. Londres: Springer.
- Rodríguez Penin, A. (2006). *Sistemas SCADA*. Marcombo.
- Rodríguez, S. A. (2010, enero 23). Desarrollan software capaz de controlar el consumo de electricidad. *Juventud Rebelde*.
- Saha, M., Izykowski, J., & Resolowski, E. (2010). *Fault Location on Power Networks*. Springer.
- Schölkopf, B., & J. Smola, A. (2002). *Learning with Kernels*. London: MIT Press.
- Shukla, K. K., & Tiwari, A. K. (2013). *Efficient algorithms for discrete wavelet transform: with applications to denoising and fuzzy inference systems*. Springer Science .
- Sierra Ceballos, F. J. (2008). *Enciclopedia del lenguaje C++*. Grupo editorial RA-MA.
- Sierra, M., León, A., Quintana, N., & Carnesoltas, M. (2011). Análisis Wavelet Continuo unidimensional de los rolos convectivos horizontales. *Cienc. Tierra y Esp, II Época*.
- Smith, J. C., Hensley, G., & Ray, L. (2009). IEEE Recommended Practice for Monitoring Electric Power Quality. *IEEE*, 1159--1995.

- Soriano González, B. M., Grau Merconchini, F., & Cruz Carnero, L. (2014). Estudio de fenómenos transitorios en circuitos eléctricos de primer orden. *Ingeniería Energética*, 35(3), 295-304.
- Stockwell, R., Mansinha, L., & Lowe, R. (1996). Localization of the Complex Spectrum: The S Transform. *IEEE Transactions on Signal Processing*, 44(4), 998-1001.
- Suykens, J. A., Gestel, J. V., De Brabanter, J., & Vandewalle, J. (2002). *Least Squares Support Vector Machines*. London: World Scientific Publishing.
- Vetter, C., & Mayur, N. (2008). *Con el impulso de la energía*. ABB.
- Vijay K, M., & Douglas B, W. (1999). *Digital signal processing*. Georgia: CRC Press LLC.
- Walnut, D. (2002). *An Introduction To Wavelet Analysis*. Boston: Birkhauser.
- Wang, L. (2005). *Support Vector Machines: Theory and Applications*. Berlin: Springer.
- Wang, L. (2008). Feature Selection with Kernel Class Separability. *IEEE*, 30(9), 1534-1546.
- Yan-nan, L., & Guang-qing, B. B. (2011). Power Quality Disturbance Signals Identification Based On Wavelet Packet and SVM. *IEEE*, 2943-2947.
- Yu, K., Ji, L., & Zhang, X. (2002). Kernel nearest-neighbor algorithm. *Neural Processing Letters*, 147--156.

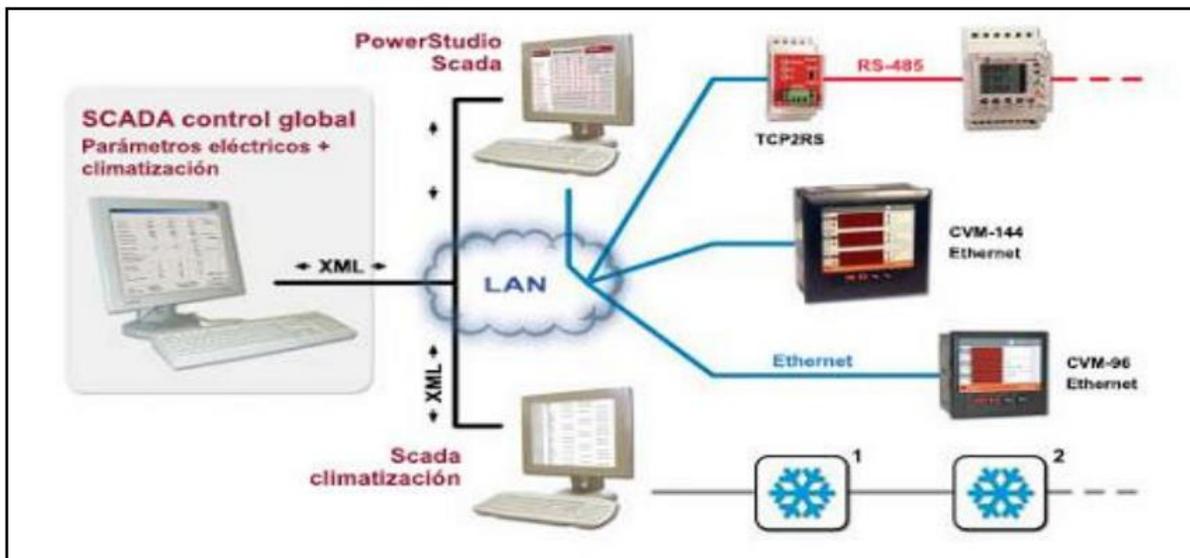
Anexos

Anexo 1: Principales productos SCADA comercializados en España.

Producto	Suministrador / Fabricante	lenguajes programación	Tipo de Enlace	precio min desarrollo	precio mín run-time
Aimax	Desin Instruments, S.A./ T.A. Engineering (USA)	VB, C	OLE, DDE, ODBC	183.000	145.000
All-Done Scada	Freixas i Ros, S.L./ Freixas i Ros, S.L. (E)	propio	DDE	178.000	178.000
All-Done/ BDE	Freixas i Ros, S.L./ Freixas i Ros, S.L. (E)	propio		175.000	40.000
Automainge	Automainge/ Automainge (E)			consultar	consultar
Bridge View	National Instruments / National Instruments (USA)				
Captor	Sisteplant / Sisteplant (E)	VB	ASCII, triggers	consultar	consultar
Checksys Objects	M2R,S.A./ M2R (E)	propio	NetDDE, ficheros, JDBC/SQL	consultar	consultar
CIC	CJM Software, S.A./ CJM Software, S.A. (E)	propio (tipo PLC)	ficheros y memoria compartida	120.000	120.000
Cmi NT	Siemens, ControlMatic Ibérica, S.A./ Idem (E)	C++, VB	DDE, ficheros	300.000	250.000
Cube	ORSI España,SA/ ORSI (I)	IIC1131-3, ladder, iter, C, VB, Fortran, Pascal, VC	ODBC, COM/DCOM, via CUBE-TRACK y via MESAD	600.000	300.000
Digivis	Elsag Bailey Hartmann & Braun, S.A./ Digivis	propio	DDE, OPC	consultar	consultar
Factory Suite 2000	Logitek, S.A./ Wonderware (USA)	propio	ActiveX, DDE, OCX, ODBC	98.000	98.000
Factorylink ECS	Medición y Control, S.A./ USDATA (USA)	Power VB, propio, C, C++	driver nativo, OPC, DDE, bases de datos relacionales	198.000	198.000
FIX DMACS 7.0	Computer Integrated Manufacturing,S.A. / Intellution (USA)	propio	DDE, ODBC, OLE	consultar	consultar
FIX Dynamics	Computer Integrated Manufacturing,S.A. / Intellution (USA)	VBA	ActiveX, ADO, DAO, ODBC, OLE	consultar	consultar
G2	Gensym, s.r.l. / ORSI España, S.A.	propio	Telewindows, G2 Gateway	consultar	consultar
Gefip	Mondragón Sistemas / Mondragón Sistemas (E)	Visual C++, VB	ficheros	consultar	consultar
Genesis CE (Pocket)	Aplein Ingenieros, S.A. / Iconics (USA)			consultar	consultar
Genesis32	Aplein Ingenieros, S.A. / Iconics (USA)	VBA	DDE, OPC, ODBC	consultar	consultar
Glassmaster Control System	Mediterranean Import Trade, S.L./ Precise Control Systems (USA)	VB, C++	DDE, ODBC, OLE	consultar	consultar
GPAO-SAC	Sistemas Avanzados de Control, S.A./ Sistemas Avanzados de Control (E)	VB, Visual C++	ODBC	consultar	400.000
IGSS32	AN Consult España, S.L./ 7-Technologies A/S (DK)	configuración programa	DDE, OLE, OCX, Automation, ODBC	115.000	336.000
Ingeniería de Aplicaciones	Ingeniería de Aplicaciones, S.A./ Ingeniería de Aplicaciones, S.A. (E)	propio	DLL, ODBC, OLE	consultar	consultar

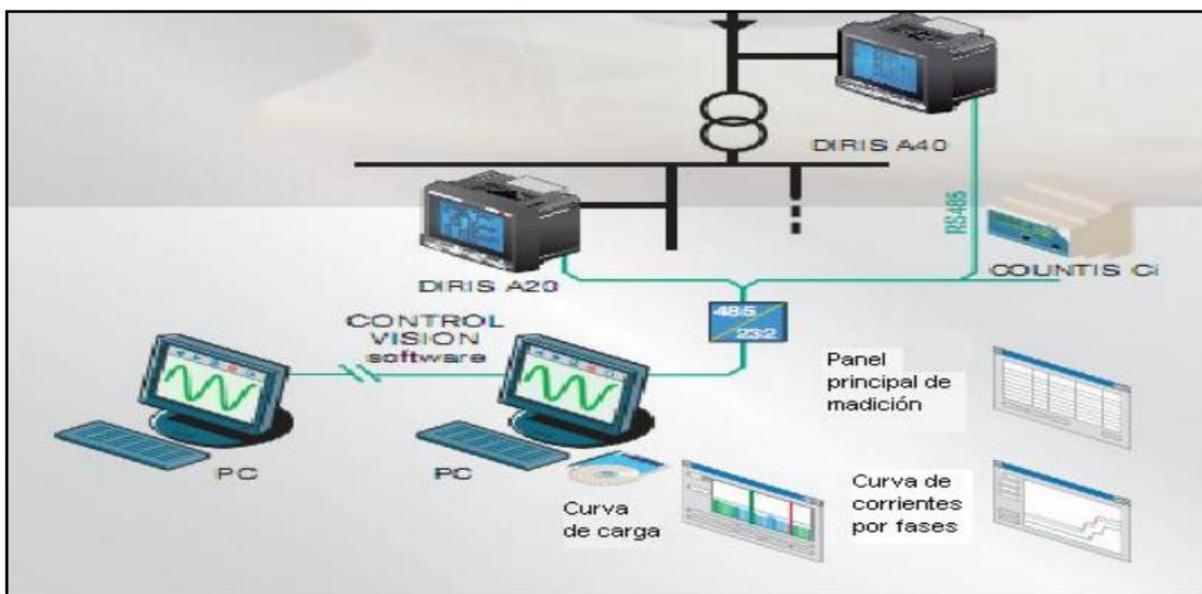
Fuente: (Colomer, Meléndez, & Ayza, 2003).

Anexo 2: Esquema de una instalación de Circutor.



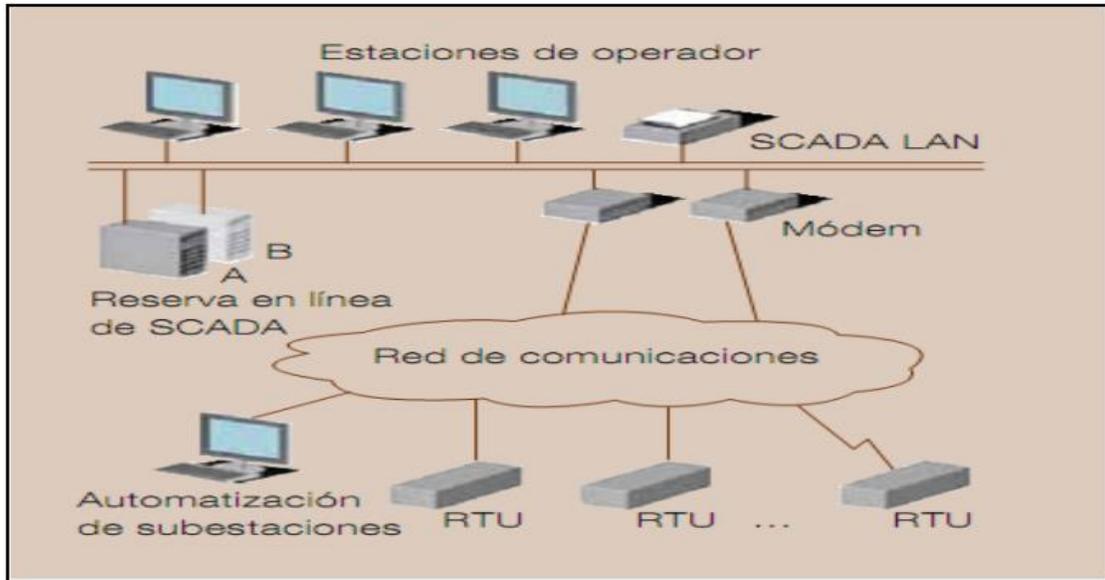
Fuente: (Lizarraga de Miguel, 2009).

Anexo 3: Esquema topológico del sistema "Control Vision".



Fuente: (Lizarraga de Miguel, 2009).

Anexo 4: Arquitectura para una red de distribución para Network Manager de ABB.



Fuente: (Lizarraga de Miguel, 2009).