



REPÚBLICA DEL ECUADOR

Escuela Politécnica Nacional

" E S C I E N T I A H O M I N I S S A L U S "

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

***Respeto hacia sí mismo y hacia los demás.***

# ESCUELA POLITÉCNICA NACIONAL

## FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

### DIRECCIONALIDAD BASADA EN PATRONES OBTENIDOS MEDIANTE LA TRANSFORMADA WAVELET PARA UN RELÉ DE SOBRECORRIENTE INSTALADO EN UNA LÍNEA DE TRANSMISIÓN

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERA  
ELÉCTRICA

CRISTINA JENNYFER TRÁVEZ GUAYASAMÍN

[crisstra\\_137@hotmail.com](mailto:crisstra_137@hotmail.com)

DIRECTOR: Dr.-Ing. FABIÁN ERNESTO PÉREZ YAULI

[fabian.perez@epn.edu.ec](mailto:fabian.perez@epn.edu.ec)

Quito, Octubre 2017

## DECLARACIÓN

Yo, Cristina Jennyfer Trávez Guayasamín, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derecho de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

Cristina Jennyfer Trávez Guayasamín

## CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por la señorita Cristina Jennyfer Trávez Guayasamín bajo mi supervisión.

---

**Dr.-Ing. Fabián Pérez Yauli**  
**DIRECTOR DEL PROYECTO**

## AGRADECIMIENTO

A ti, mi Dios y mi Señor, por tanto amor, por salvarme y rescatarme, por el privilegio de conocerte, y tu fidelidad a pesar de mi infidelidad. Es por tu gracia todo lo soy y lo que tengo. Me has enseñado el camino en que debo andar, y fijaste sobre mí Tus ojos. Llenaste mi corazón y Tu Luz me alumbró en momentos de oscuridad. Has disfrutado con mis risas y has enjugado mis lágrimas. Tu Presencia me dio propósito para vivir. Gracias por amarme primero, porque enamoraste mi corazón, y ahora yo te pertenezco.

A ti, mamita, Martha Guayasamín, la bendición que Dios puso en mi vida para cuidarme y protegerme. Gracias por tu amor, valentía, apoyo y esfuerzo incondicional. Sin ti no hubiese sido posible llegar a esta estación del camino. Eres la mejor porque sé que me has dado lo mejor. Que el Cielo te recompense mil veces más. ¡Te amo mamita!

A ti, ñañita linda, Marjory, mi compañera de toda la vida, *mi “alma gemela”*... Eres el regalo perfecto que Dios me dio desde que estuvimos en el vientre de mamá. Ha sido y es maravilloso tenerte a mi lado. Eres única, maravillosa y confiable como nadie en esta tierra. Gracias por tu comprensión, por sostenerme, y animarme.

A mis amigos, *“y no quiero dar sus nombres ni apellidos, ellos lo saben y se dan por aludidos”*. Con ustedes he reído y llorado con libertad. Me honran con su respeto y me hacen crecer con sus regaños. Me quieren como soy. ¡Gracias!

Al Dr. Fabián Pérez, por acompañarme a lo largo de este trabajo. Gracias por su dirección y experiencia profesional que me permitió culminar esta etapa. Gracias por su calidad de gente y actitud desinteresada, siempre presto a ayudar.

Al Dr. Franklin Quilumba por su valiosa colaboración que fue vital para el desarrollo del presente trabajo.

*“Todo el que quiera ser sabio  
que comience por obedecer a Dios;  
conocer al Dios santo  
es dar muestras de inteligencia.”  
Prov. 9:1*

**DEDICATORIA**

*Al Único y Sabio Dios*

*Todo honor y gloria*

*Es para ti,*

**Jesucristo,**

*La razón de mi existir.*

*«Tú eres mi Señor; no hay para mí bien fuera de Ti.»*

## CONTENIDO

<b>DECLARACIÓN</b> .....	<b>I</b>
<b>CERTIFICACIÓN</b> .....	<b>II</b>
<b>AGRADECIMIENTO</b> .....	<b>III</b>
<b>DEDICATORIA</b> .....	<b>IV</b>
<b>CONTENIDO</b> .....	<b>V</b>
<b>GLOSARIO</b> .....	<b>1</b>
<b>RESUMEN</b> .....	<b>2</b>
<b>PRESENTACIÓN</b> .....	<b>4</b>
<b>CAPÍTULO 1. INTRODUCCIÓN</b> .....	<b>5</b>
<b>1.1. OBJETIVOS</b> .....	<b>5</b>
1.1.1. <i>OBJETIVO GENERAL</i> .....	<b>5</b>
1.1.2. <i>OBJETIVOS ESPECÍFICOS</i> .....	<b>5</b>
<b>1.2. JUSTIFICACIÓN</b> .....	<b>6</b>
<b>CAPÍTULO 2. MARCO TEÓRICO</b> .....	<b>8</b>
<b>2.1. CONCEPTOS BÁSICOS DE PROTECCIONES ELÉCTRICAS</b> .....	<b>8</b>
2.1.1. <i>COMPONENTES DE UN SISTEMA DE PROTECCIÓN</i> .....	<b>9</b>
2.1.2. <i>ZONAS DE PROTECCIÓN</i> .....	<b>10</b>
2.1.2.1. <i>Protección primaria</i> .....	<b>11</b>
2.1.2.2. <i>Protección de respaldo</i> .....	<b>13</b>
2.1.3. <i>CARACTERÍSTICAS FUNCIONALES DE UN SISTEMA DE PROTECCIÓN</i> .....	<b>14</b>
2.1.3.1. <i>Sensibilidad</i> .....	<b>14</b>
2.1.3.2. <i>Selectividad</i> .....	<b>15</b>
2.1.3.3. <i>Velocidad</i> .....	<b>15</b>
2.1.3.4. <i>Confiabilidad</i> .....	<b>17</b>
2.1.4. <i>CLASIFICACIÓN DE LOS RELÉS DE PROTECCIÓN</i> .....	<b>17</b>
2.1.4.1. <i>Según su tecnología</i> .....	<b>18</b>
2.1.4.2. <i>Según sus características de funcionamiento o tipo de protección</i> .....	<b>18</b>
2.1.5. <i>RELÉS DE SOBRECORRIENTE</i> .....	<b>19</b>
2.1.5.1. <i>Ecuación característica de los relés de sobrecorriente</i> .....	<b>20</b>
2.1.5.2. <i>Curvas características de los relés de sobrecorriente</i> .....	<b>21</b>
2.1.5.3. <i>Clasificación de relés de sobrecorriente según su sentido de detección de corriente</i> .....	<b>22</b>
2.1.5.4. <i>Relé de sobrecorriente direccional</i> .....	<b>23</b>
2.1.5.4.1. <i>Principio de operación</i> .....	<b>25</b>

2.1.5.4.2.	Conexiones de los relés direccionales de sobrecorriente .....	25
2.1.5.4.3.	Algunas aplicaciones .....	28
2.1.5.4.4.	Protección de Líneas de Transmisión con Relés de Sobrecorriente Direccional .....	28
<b>2.2.</b>	<b>DESCRIPCIÓN DE HERRAMIENTAS EMPLEADAS PARA PROCESAMIENTO DE DATOS.....</b>	<b>30</b>
2.2.1.	<i>TRANSFORMADA WAVELET</i> .....	30
2.2.1.1.	La Transformada Wavelet Continua .....	33
2.2.1.2.	La Transformada Wavelet Discreta .....	36
2.2.1.3.	El Análisis Multi-Resolución de una señal .....	37
2.2.1.4.	Aplicaciones a SEP .....	40
2.2.1.4.1.	Calidad de energía .....	41
2.2.1.4.2.	Protección del sistema eléctrico .....	42
2.2.1.4.3.	Transitorios del sistema de potencia .....	43
2.2.1.4.4.	Mediciones del sistema de potencia .....	44
2.2.1.4.5.	Descargas parciales .....	44
2.2.1.4.6.	Estimación de la demanda .....	45
2.2.2.	<i>MINERÍA DE DATOS</i> .....	46
2.2.2.1.	Aprendizaje Automático .....	47
2.2.2.2.	Máquinas de Soporte Vectorial .....	48
2.2.2.2.1.	Análisis del caso linealmente separable .....	50
2.2.2.2.2.	Análisis del caso linealmente no separable .....	55
2.2.2.2.3.	Multclasificación con SVMs .....	59
<b>CAPÍTULO 3.</b>	<b>DESCRIPCIÓN DE LA METODOLOGÍA.....</b>	<b>61</b>
3.1.	<b>DIAGRAMA DE LA METODOLOGÍA .....</b>	<b>61</b>
3.2.	<b>IMPLEMENTACIÓN DEL SISTEMA DE PRUEBA .....</b>	<b>63</b>
3.2.1.	<i>GENERADORES</i> .....	63
3.2.2.	<i>TRANSFORMADORES</i> .....	64
3.2.3.	<i>LÍNEAS DE TRANSMISIÓN</i> .....	64
3.2.3.1.	Modelación de Línea de Transmisión para simular fallas.....	65
3.2.4.	<i>CARGAS</i> .....	66
3.3.	<b>CREACIÓN DE ARCHIVOS BASE EN ATP .....</b>	<b>66</b>
3.4.	<b>SIMULACIÓN DE CORTOCIRCUITOS VINCULANDO ATP Y MATLAB .....</b>	<b>67</b>
3.4.1.	<i>ENTRADAS DE LA FUNCIÓN DE SIMULACIÓN “runatpmatlab”</i> .....	67
3.4.2.	<i>INICIALIZACIÓN DE VARIABLES EN “runatpmatlab”</i> .....	68
3.4.3.	<i>SALIDAS DE LA FUNCIÓN “runatpmatlab”</i> .....	69
3.5.	<b>BASE DE DATOS CON CORRIENTES DE FALLA .....</b>	<b>70</b>
3.5.1.	<i>TIPOS DE CORTOCIRCUITOS</i> .....	70
3.6.	<b>APLICACIÓN DE LA TRANSFORMADA WAVELET DISCRETA EN LA BASE DE DATOS.....</b>	<b>71</b>
3.6.1.	<i>SELECCIÓN DE DATOS: CORRIENTES DE FALLA</i> .....	72
3.6.2.	<i>COMANDO wavedec</i> .....	73

3.6.2.1.	Descripción del comando <i>wavedec</i> .....	73
3.6.3.	<i>APLICACIÓN DEL COMANDO wavedec</i> .....	74
<b>3.7.</b>	<b>FORMACIÓN DE MATRICES DE ENTRADA Y SALIDA</b> .....	<b>74</b>
3.7.1.	<i>MATRICES DE ENTRADA</i> .....	75
3.7.1.1.	Matriz A .....	75
3.7.1.2.	Matriz B .....	77
3.7.2.	<i>VECTORES DE SALIDA</i> .....	78
<b>3.8.</b>	<b>SELECCIÓN DE CONJUNTOS DE ENTRENAMIENTO Y VALIDACIÓN</b> .....	<b>80</b>
3.8.1.	<i>ENTRENAMIENTO, VALIDACIÓN Y EVALUACIÓN</i> .....	80
3.8.2.	<i>PORCENTAJES DE ENTRENAMIENTO Y VALIDACIÓN</i> .....	81
<b>3.9.</b>	<b>ENTRENAMIENTO CON SVMs</b> .....	<b>81</b>
3.9.1.	<i>COMANDO fitcecoc</i> .....	82
3.9.1.1.	Descripción del comando <i>fitcecoc</i> .....	83
3.9.2.	<i>APLICACIÓN DEL COMANDO fitcecoc</i> .....	83
<b>3.10.</b>	<b>VALIDACIÓN</b> .....	<b>84</b>
3.10.1.	<i>COMANDO plotconfusion</i> .....	85
3.10.1.1.	Descripción del comando <i>plotconfusion</i> .....	85
3.10.2.	<i>APLICACIÓN DEL COMANDO plotconfusion</i> .....	86
<b>3.11.</b>	<b>ESTRATEGIAS DE TRANSFORMACIÓN DE DATOS</b> .....	<b>88</b>
3.11.1.	<i>NORMALIZACIÓN</i> .....	88
3.11.1.1.	Normalización Min-Max.....	88
3.11.2.	<i>COMANDO mapminmax</i> .....	89
3.11.2.1.	Descripción del comando <i>mapminmax</i> .....	89
3.11.3.	<i>APLICACIÓN DEL COMANDO mapminmax</i> .....	89
<b>3.12.</b>	<b>EVALUACIÓN O PRUEBA</b> .....	<b>91</b>
<b>3.13.</b>	<b>ANÁLISIS DE RESULTADOS</b> .....	<b>91</b>
<b>CAPÍTULO 4. APLICACIÓN DE LA METODOLOGÍA Y ANÁLISIS DE RESULTADOS</b> .....		<b>92</b>
<b>4.1.</b>	<b>IMPLEMENTACIÓN DEL SISTEMA DE PRUEBA</b> .....	<b>92</b>
4.1.1.	<i>GENERADORES</i> .....	93
4.1.2.	<i>TRANSFORMADORES</i> .....	94
4.1.3.	<i>LÍNEAS DE TRANSMISIÓN</i> .....	95
4.1.3.1.	Pestaña <i>Model</i> .....	97
4.1.3.2.	Pestaña <i>Data</i> .....	98
4.1.4.	<i>CARGAS</i> .....	99
4.1.5.	<i>EQUIPOS DE MEDICIÓN</i> .....	102
<b>4.2.</b>	<b>ARCHIVOS BASE EN ATP</b> .....	<b>102</b>
<b>4.3.</b>	<b>SIMULACIÓN DE CORTOCIRCUITOS VINCULANDO ATP Y MATLAB</b> .....	<b>103</b>
4.3.1.	<i>ENTRADAS DE LA FUNCIÓN DE SIMULACIÓN "runatp matlab"</i> .....	106

<b>4.4.</b>	<b>MATRICES DE ENTRADA Y SALIDA.....</b>	<b>106</b>
<b>4.5.</b>	<b>ANÁLISIS DE RESULTADOS.....</b>	<b>111</b>
4.5.1.	<i>NORMALIZACIÓN.....</i>	<i>111</i>
4.5.1.1.	Normalización de toda la matriz .....	111
4.5.1.2.	Normalización por filas .....	112
4.5.1.3.	Normalización por columnas .....	113
4.5.2.	<i>RESULTADOS DE ENTRENAMIENTO.....</i>	<i>115</i>
4.5.3.	<i>RESULTADOS DE VALIDACIÓN.....</i>	<i>115</i>
4.5.3.1.	Modelos A y B Predeterminados .....	116
4.5.4.	<i>RESULTADOS DE EVALUACIÓN.....</i>	<i>119</i>
	<b>CAPÍTULO 5. CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>122</b>
5.1.	CONCLUSIONES.....	122
5.2.	RECOMENDACIONES.....	124
	<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>126</b>
	<b>ANEXOS.....</b>	<b>129</b>
ANEXO 1.	<b>MÉTODOS DE APRENDIZAJE.....</b>	<b>130</b>
ANEXO 2.	<b>MANUAL DE USUARIO - GUI.....</b>	<b>131</b>
A2.1.	<i>INSTALACIÓN EN PC.....</i>	<i>131</i>
A2.2.	<i>ORGANIZACIÓN DE LA BASE DE DATOS.....</i>	<i>132</i>
A2.3.	<i>INICIO DEL PROGRAMA.....</i>	<i>133</i>
A2.3.1.	Ventana CARÁTULA.....	133
A2.3.2.	Ventana ENTRENAMIENTO.....	134
A2.3.2.1.	Opción 1 .....	134
A2.3.2.2.	Opción 2 .....	136
A2.3.3.	Ventana VALIDACIÓN.....	137
A2.3.3.1.	Ubicación Señales Internas .....	137
A2.3.3.2.	Seleccionar Falla .....	138
A2.3.3.3.	Gráficos .....	138
A2.3.3.4.	Predicciones .....	138
A2.3.4.	Ventana EVALUACIÓN.....	139
A2.3.4.1.	Señales Externas.....	140
A2.3.4.2.	Señales Internas.....	140
A2.3.5.	Ventana FIN .....	140
ANEXO 3.	<b>ANÁLISIS DE UNA MATRIZ DE CONFUSIÓN.....</b>	<b>141</b>

## GLOSARIO

PLC: Comunicación por Onda Portadora (Power Line Carrier / Power Line Communications).

PQ: Calidad de Energía (Power Quality).

PD: Descargas Parciales (Partial Discharges).

SVM: Máquinas de Soporte Vectorial (Support Vector Machines).

FT: Transformada de Fourier (Fourier Transform).

STFT: Transformada Rápida de Fourier (Short-Time Fourier Transform).

WT: Transformada Wavelet (Wavelet Transform).

CWT: Transformada Wavelet Continua (Continuous Wavelet Transform).

DWT: Transformada Wavelet Discreta (Discrete Wavelet Transform).

MRA: Análisis Multi-Resolución (Multi-Resolution Analysis).

DM: Minería de Datos (Data Mining).

ML: Aprendizaje Automático (Machine Learning).

GUI: Interfaz Gráfica de Usuario (Graphical User Interface).

## RESUMEN

Este trabajo presenta un modelo de patrones capaz de proporcionar la característica de direccionalidad a un relé de sobrecorriente instalado en una línea de transmisión, mediante el uso de dos herramientas matemáticas: la Transformada Wavelet y las Máquinas de Soporte Vectorial. Adicionalmente, se muestra un segundo modelo con la capacidad de identificar el tipo de falla producida en la línea protegida por el relé.

El sistema eléctrico de prueba se implementó en el programa ATP/EMTP, el cual se vinculó con MATLAB para simular diez tipos de cortocircuitos a diferentes distancias, tanto en la línea de transmisión protegida por el relé, como en la línea anterior. Los resultados de las simulaciones se agruparon por clase y se almacenaron en una misma base de datos.

En el aprendizaje supervisado se debe conocer los atributos de cada elemento que será entrenado, así como la clase a la que pertenece dicho objeto. De allí que se requirieron las matrices de entrada y salida, las cuales fueron construidas con los Coeficientes de Detalle, como resultado de aplicar la Transformada Wavelet sobre la base de datos. A partir de estas matrices, se entrenaron los modelos de patrones mediante las Máquinas de Soporte Vectorial, que constituyen un método de aprendizaje basado en la búsqueda de hiperplanos de separación para clasificar y establecer un modelo.

Además, se implementó una interfaz gráfica en MATLAB para que el usuario pueda apreciar de forma sencilla y didáctica la metodología propuesta, y la efectividad de la misma. Como parte del programa, se elaboraron las etapas de: Creación de matrices, Entrenamiento, Validación y Evaluación de modelos.

En la fase de Validación se consideró un conjunto de fallas diferente al empleado durante el Entrenamiento. A su vez, la Evaluación se realizó con un nuevo conjunto (no utilizado en etapas anteriores), que sirvió para ratificar la validez de los modelos.

El desarrollo del presente trabajo involucró múltiples entrenamientos para conseguir excelentes modelos. En efecto, se logró obtener ejemplares que

presentan el 100% de aciertos en la predicción de sus variables. A estos modelos se los denominó *predeterminados*.

En base a lo expuesto, se puede afirmar que los resultados muestran una alta efectividad en cuanto a la identificación de la dirección y tipo de una falla. De esta manera, se asegura una operación correcta para el relé de sobrecorriente direccional, ante cualquier tipo de cortocircuito producido en la línea protegida.

## PRESENTACIÓN

Los relés de sobrecorriente, al igual que cualquier relé de protección, tienen como objetivo detectar una falla, y aislarla del sistema de potencia mediante la apertura de un interruptor. Los relés direccionales de sobrecorriente tienen la particularidad de operar únicamente en una dirección específica. Esta función se consigue monitoreando la señal de voltaje, además de la corriente que es habitual en los relés no direccionales. Es decir, el voltaje, denominado señal de referencia, constituye una señal esencial para establecer la dirección de la falla y determinar si el relé debe operar o no.

El aporte principal de la herramienta computacional propuesta en este trabajo radica en añadir la característica de direccionalidad al relé de sobrecorriente (no direccional) sin necesidad de sensor la señal de voltaje. En virtud de ello, el documento se ha dividido en cinco capítulos que se detallan a continuación:

En el Capítulo 1, se presenta una breve introducción del tema desarrollado. Aquí se incluye la justificación, así como los objetivos del presente trabajo.

En el Capítulo 2, se exponen los aspectos teóricos esenciales que fueron utilizados en este estudio. Los conceptos presentados se relacionan con Protecciones Eléctricas, Transformada Wavelet y Máquinas de Soporte Vectorial.

El Capítulo 3 describe la metodología propuesta para obtener el modelo de patrones deseado. Se detalla desde la implementación del sistema eléctrico de prueba en el cual se realizaron simulaciones de diferentes cortocircuitos, hasta la obtención y validación de los patrones.

En el Capítulo 4, se presentan básicamente los resultados de los procesos de normalización, validación, entrenamiento y evaluación de datos, así como el análisis correspondiente de los *modelos predeterminados de patrones*.

En el Capítulo 5, se presentan las conclusiones obtenidas en el desarrollo de este trabajo, y algunas recomendaciones que pueden ser consideradas en estudios posteriores o afines a éste.

# CAPÍTULO 1

## INTRODUCCIÓN

Un relé de sobrecorriente direccional instalado en una línea de transmisión detecta fallas que ocurren dentro de la misma, considerando una dirección. Para ello, el relé requiere supervisar señales de corriente y contar con una señal de voltaje que sirva de referencia y proporcione la característica de direccionalidad; de este modo el relé puede identificar en qué dirección ocurrió la falla. Consecuentemente, se debe disponer de instrumentos de medida tales como el transformador de potencial o voltaje (TP) y transformador de corriente (TC). Numerosos equipos de medición, así como un conexionado abundante se ven reflejados, de manera directa, en los costos de instalación.

Muchas investigaciones se han dedicado a la búsqueda de nuevos dispositivos de protección que prescindan de la señal de referencia proveniente de un TP, para reducir los costos y el conexionado del sistema de protección. En ese sentido, el presente trabajo se enmarca en dichas propuestas, trabajando exclusivamente con las señales de corriente y herramientas matemáticas como la Transformada Wavelet, Máquinas de Soporte Vectorial, entre otras.

### 1.1. OBJETIVOS

#### 1.1.1. OBJETIVO GENERAL

- Utilizar una metodología para determinar, mediante patrones característicos obtenidos con la transformada Wavelet, la dirección de fallas en líneas de transmisión a fin de prescindir de la señal de voltaje requerida en relés direccionales convencionales.

#### 1.1.2. OBJETIVOS ESPECÍFICOS

- Presentar una breve descripción de los fundamentos de la Transformada Wavelet y su aplicación en sistemas eléctricos de potencia, así como la base teórica de las Máquinas de Soporte Vectorial (SVMs).

- Simular un conjunto de fallas en una línea de transmisión perteneciente a un sistema eléctrico de prueba modelado en el programa ATP/EMTP.
- Utilizar una metodología para obtener un modelo de patrones, que permita identificar la dirección de ocurrencia de fallas en una línea de transmisión, mediante el procesamiento de señales de corriente derivadas de las simulaciones. Esto se realizará en MATLAB utilizando la transformada Wavelet y la técnica SVM.
- Validar los patrones obtenidos utilizando la metodología empleada y un conjunto de fallas simuladas diferente al original.

## **1.2. JUSTIFICACIÓN**

El funcionamiento de los relés direccionales se fundamenta en la utilización de la Transformada Rápida de Fourier que permite obtener los fasores de corriente y de voltaje, mismos que al ser comparados (se compara el ángulo de dichos fasores) dan a conocer la dirección en la que ocurrió una falla. Sin embargo, la alimentación de la señal de voltaje supone la instalación de un elemento adicional de medida como el transformador de potencial (TP), y esto a su vez implica un costo extra en el sistema de protección.

Por otro lado, la Transformada Wavelet es una herramienta matemática que permite obtener datos de una señal en el dominio tiempo-frecuencia, y analizarlos en forma separada.

A diferencia de la transformada de Fourier, la Transformada Wavelet permite trabajar sobre señales no estacionarias como corrientes de cortocircuitos, con discontinuidades o picos. Esta característica es aprovechada y aplicada en la presente propuesta para obtener un modelo de patrones representativos y útiles en la clasificación de señales, con el propósito de detectar la dirección de fallas que ocurren en líneas de transmisión, únicamente al procesar señales de

corriente, prescindiendo de la señal de voltaje necesaria para la referencia en un relé de sobrecorriente direccional-convencional.

Tomando en cuenta que siempre es conveniente minimizar costos, sin comprometer la calidad, operación ni funcionalidad de los equipos, se propone eliminar costos debidos a TPs y sus respectivas conexiones.

## CAPÍTULO 2

### MARCO TEÓRICO

Este capítulo resume y define los conceptos básicos teóricos utilizados en este trabajo. En primer lugar, se describen los sistemas de protección en cuanto a sus componentes, características y clasificaciones. Luego, se analiza específicamente al relé de sobrecorriente direccional, y en último término, se mencionan las herramientas relevantes que fueron empleadas en la obtención del modelo de patrones de direccionalidad. Este sustento teórico tiene por objetivo familiarizar al lector con las áreas que intervienen en el desarrollo del presente trabajo.

#### 2.1. CONCEPTOS BÁSICOS DE PROTECCIONES ELÉCTRICAS

Protecciones Eléctricas es una expresión utilizada para designar tanto a la ciencia, como a la operación de los dispositivos de protección dentro de una estrategia controlada, con el objetivo de maximizar la continuidad del servicio y minimizar los daños a la propiedad y al personal, en caso de un comportamiento anormal del sistema. En otras palabras, los relés de protección tienen como objetivo “proteger”<sup>1</sup> al personal, a los equipos y al sistema de potencia del efecto de un componente fallado, provocando la desconexión inmediata del mismo, cuando sufre un cortocircuito o empieza a funcionar de manera irregular, tal que pueda causar daños o interferir de alguna forma con el funcionamiento efectivo del resto del sistema.

En esta tarea, los relés trabajan juntamente con los disyuntores<sup>2</sup>, los cuales son capaces de desconectar el elemento defectuoso, cuando son llamados a hacerlo por los relés. Los disyuntores se localizan generalmente de modo que cada generador, transformador, barra, línea de transmisión, etc., pueda ser desconectado completamente del sistema de potencia. Estos disyuntores deben tener la capacidad suficiente para transportar momentáneamente la corriente

---

<sup>1</sup> Cuando se dice que los relés "protegen" se quiere decir que, junto con otros equipos, los relés ayudan a minimizar el daño y mejorar el servicio [1].

<sup>2</sup> Disyuntores, también conocidos como Interruptores de Potencia, o simplemente Interruptores.

máxima de cortocircuito que pueda fluir a través de ellos, y luego interrumpir esta corriente. También deben soportar el cierre en un cortocircuito y luego interrumpirlo de acuerdo con ciertos estándares prescritos [1], [2].

### 2.1.1. COMPONENTES DE UN SISTEMA DE PROTECCIÓN

Las protecciones eléctricas abarcan una gran gama de dispositivos que pueden variar desde un simple fusible hasta un complejo esquema con comunicación por onda portadora (PLC<sup>3</sup>) para transmitir la información de una subestación a otra. De manera general, se establecen los siguientes componentes para un sistema de protección [4]:

- 1) **Transformadores de medida:** Son los transformadores de voltaje (TPs) y de corriente (TCs), que constituyen el enlace entre el sistema de potencia y los relés de protección. Estos dispositivos permiten obtener la información sobre las condiciones de operación del sistema, convirtiendo las magnitudes originales de voltajes y corrientes en magnitudes proporcionalmente inferiores que puedan ser conectadas a las entradas de los relés.
- 2) **Relés:** Son dispositivos que controlan el estado de un interruptor mediante una entrada eléctrica, y son los encargados de discriminar entre las condiciones normales y anormales del sistema. Al detectar condiciones anormales, entran en operación, cerrando o abriendo sus contactos de acuerdo a la lógica programada en los relés. De esta manera, ordenan operar a los interruptores, para desconectar los elementos comprometidos, debido a las perturbaciones o fallas. Por su funcionamiento, los relés pueden ser eléctricos, electrónicos, transistorizados, transductorizados, etc.
- 3) **Interruptores de Potencia:** Son los encargados de aislar a los equipos afectados, por una anomalía del sistema. Los interruptores, al ser comandados por los relés; o bien, voluntariamente por los operadores,

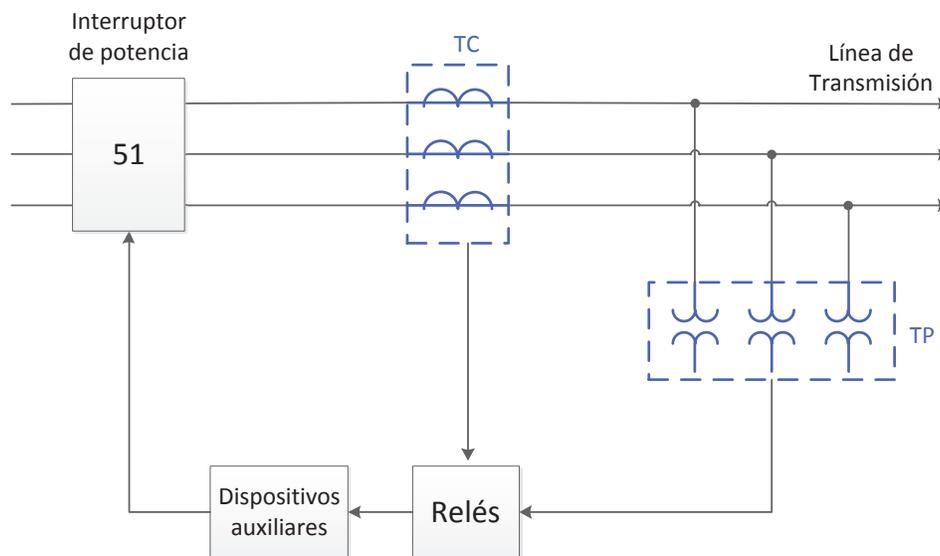
---

<sup>3</sup> PLC: Por sus siglas en inglés *Power Line Carrier* o *Power Line Communications*. Existen varias traducciones al español como: Comunicación a través de la Red de Energía, Transmisión por Onda Portadora de Corriente, Comunicación por Onda Portadora, Onda Portadora por Línea Eléctrica, Comunicación por Línea Eléctrica, y otras similares. Se trata fundamentalmente, de la transmisión de voz y datos a través de la red eléctrica [3].

maniobran de forma directa el sistema de potencia, conectando o desconectando elementos de las redes eléctricas.

- 4) Circuitos de control:** Son los elementos y dispositivos que interconectan los tres componentes anteriores. Entre ellos se pueden citar alambrados, comunicadores de tecnología PLC, dispositivos auxiliares como switches, lámparas indicadoras, relés auxiliares, alarmas, entre otros. Los circuitos de control, además de contener al sistema de comunicaciones, que informa sobre el estado de interruptores y relés para analizar la condición del sistema de potencia, también engloban al sistema de alimentación del sistema de protecciones. Generalmente, la alimentación de interruptores y relés debe ser independiente del sistema protegido, con el fin de asegurar la autonomía de la operación. Es común que los sistemas de protección estén alimentados por baterías.

En la Figura 2.1 se muestran los componentes de un sistema de protección, descritos anteriormente, para el caso de una línea de transmisión.



**Figura 2.1.** Componentes de un Sistema de Protección [4].

### 2.1.2. ZONAS DE PROTECCIÓN

Se debe tener en cuenta que, cualquier tipo de falla tendría efectos simultáneos en diversas partes del sistema de potencia, por lo cual sería ilógico diseñar un

sistema de protección que ordene la apertura de todos los interruptores donde se detecten los efectos de dicha falla. Lo ideal es despejar la falla abriendo o desconectando el mínimo número de interruptores. Esto se puede lograr dividiendo el sistema de protección en regiones denominadas *zonas de protección*. Considerando el equipo de protección contra cortocircuitos, que son las fallas más frecuentes, existen dos grupos [1], [5]:

- *Protecciones primarias o principales*: Es la primera línea de defensa contra fallas del sistema.
- *Protecciones de respaldo*: Operan cuando la protección primaria no lo hace.

Esto supone evidentemente que, al ocurrir la falla, ambas protecciones inician su operación, pero la principal, por tener menor tiempo de operación, cumple primero su función [4].

#### **2.1.2.1. Protección primaria**

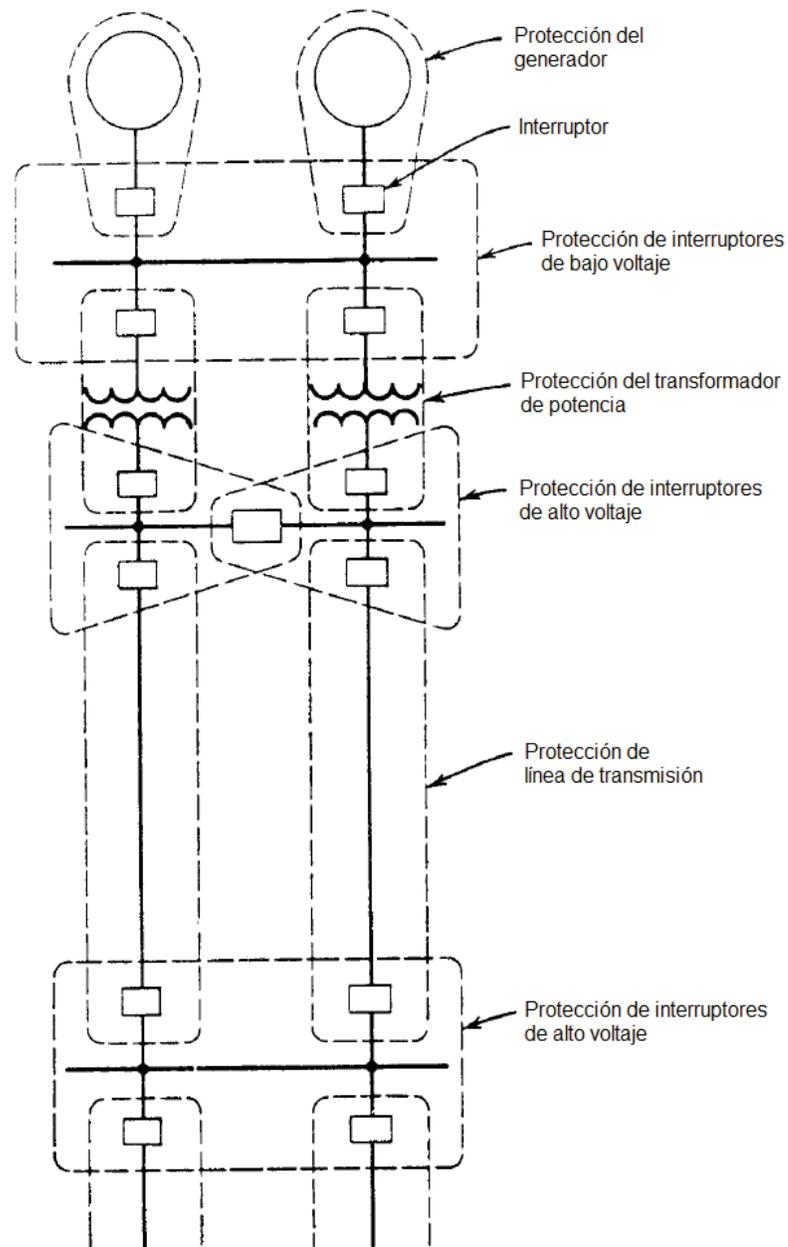
La protección primaria o principal debe operar inmediatamente después de que uno de sus elementos detecte una falla, en virtud de lo cual, los relés de protección ordenan la apertura del mínimo número de interruptores en el menor tiempo posible. Para este propósito, se utilizan relés de alta velocidad (1 a 3 ciclos exclusivos del tiempo de operación del interruptor).

En la Figura 2.2 se muestra un ejemplo de protección primaria. Se puede observar, en primer lugar, que los interruptores están localizados en las conexiones de cada elemento del sistema de potencia. Esta disposición hace posible desconectar solo el elemento defectuoso.

La segunda observación es que se establece una zona de protección separada alrededor de cada elemento del sistema. Esto significa que cualquier falla que ocurra dentro de una zona dada originará el disparo (apertura) de todos los interruptores dentro de esa zona y de solo esos interruptores.

Finalmente, se observa que las zonas adyacentes de protección de la Figura 2.2 se superponen alrededor de un interruptor. Esta es la práctica preferida porque, para fallas en cualquier lugar, excepto en la región de superposición, es necesario

que se dispare el mínimo número de interruptores. Se excluyen las zonas solapadas, debido a la preferencia de proteger todo el sistema de potencia, pese a que implique abrir un número mayor de interruptores al mínimo necesario; de otra forma, una parte del sistema quedaría sin protección [1], [5], [6].



**Figura 2.2.** Parte de un sistema eléctrico de potencia con protección primaria [1].

### 2.1.2.2. Protección de respaldo

La protección de respaldo tiene como objetivo aislar una falla cuando, por cualquier razón, la protección principal no ha operado. El funcionamiento de la protección de respaldo será más lento que la protección primaria (a fin de permitir el tiempo necesario para que la protección principal opere primero), y como resultado, una mayor parte del sistema de potencia quedará fuera de servicio [7].

Esta protección se emplea únicamente para protección de cortocircuitos, debido a que son el tipo predominante de falla en un sistema de potencia, y hay más posibilidades de que falle la protección primaria. La experiencia ha demostrado que la protección de respaldo no es económicamente justificable, para casos que no sean cortocircuitos [1]. Aún para este tipo de falla no siempre se emplea, y su uso obedece a los siguientes factores [4]:

- Importancia y costo del equipo protegido.
- Probabilidad de fallas.
- Importancia de no producir perturbaciones en el resto del sistema.
- Consecuencia de las pérdidas del suministro.

La protección de respaldo también es utilizada a menudo para proporcionar protección primaria, cuando el equipo que debería ocuparse de esto se encuentra fuera de servicio por mantenimiento o reparación [1].

Un relé puede proporcionar protección de respaldo simultáneamente a diferentes componentes del sistema, y de igual manera, un mismo equipo puede tener varios relés de protección de respaldo diferentes. En efecto, es muy común que un relé actúe como protección primaria para un equipo y como respaldo para otro. El respaldo puede variar desde la total duplicación de los componentes de un esquema de protección, hasta la duplicación únicamente de los relés que, según sea el caso, son de diferente característica de funcionamiento.

La protección de respaldo puede ser local o remota. La protección de respaldo local se encuentra dentro de la zona en la que se produce la falla y dispara el disyuntor principal o los disyuntores en zonas adyacentes. La protección de

respaldo remoto se encuentra en zonas adyacentes y en general sólo dispara disyuntores en su zona [4], [6], [8].

### 2.1.3. CARACTERÍSTICAS FUNCIONALES DE UN SISTEMA DE PROTECCIÓN

Todas las protecciones que cubren un sistema eléctrico, deben cumplir una serie de características, para que en conjunto, cada esquema de protección trabaje asociado al resto, con el objetivo de despejar fallas y/o perturbaciones. A continuación se describen dichas características funcionales [4]:

#### 2.1.3.1. Sensibilidad

La sensibilidad en un sistema de protección es la capacidad de ese sistema para identificar una condición anormal que excede un valor de *pick-up*<sup>4</sup> o de umbral de detección, e iniciar una acción protectora cuando las cantidades detectadas exceden ese umbral. Se dice que los relés o esquemas de protección son sensibles si sus parámetros operativos primarios son bajos [2], [7].

Esta característica permite al sistema de protecciones ser suficientemente sensible para operar de manera segura bajo cualquier condición de falla (máxima o mínima) que se presente en el sistema eléctrico protegido, tomando en cuenta la condición real que provoca la menor tendencia a la operación. Además, la sensibilidad permite que el sistema de protección permanezca inalterable para fallas externas a su zona [1], [4].

En relés antiguos, electromecánicos, la sensibilidad se consideró en relación al movimiento de medición, y se midió en términos de su consumo “voltaje-amperio” para provocar la operación. Por el contrario, en los relés modernos, digitales y numéricos, la sensibilidad que puede alcanzarse rara vez está limitada por el diseño del dispositivo, sino por su aplicación y los parámetros asociados de los transformadores de corriente y voltaje [7].

---

<sup>4</sup> El término *pick-up* como una característica de los relés, se define como el nivel mínimo (corriente, voltaje, potencia, etc.) que inicia una acción. El *pick-up* de un relé de protección de sobrecorriente ha sido generalmente considerado el valor mínimo de corriente que hace que el relé cierre sus contactos [6].

### 2.1.3.2. Selectividad

La selectividad en un sistema de protección, se refiere al diseño global de la estrategia de protección, en la cual sólo aquellos disyuntores cuyo funcionamiento es necesario para aislar una falla, funcionan para desconectar el componente averiado. Esto implica, seleccionar entre aquellas condiciones que requieren un funcionamiento rápido, una operación con tiempo de retardo, o aquellas que no requieren ninguna operación. De ahí que, la selectividad está relacionada directamente con las condiciones de operación instantánea, temporizada y de no operación, a fin de obtener la funcionalidad selectiva deseada. De este modo, se puede aislar exclusivamente el equipo o zona fallada, manteniendo en servicio aquel equipo o zona que no es imprescindible desconectar. Esta propiedad de disparo selectivo también se denomina *discriminación* [1], [2], [4], [7].

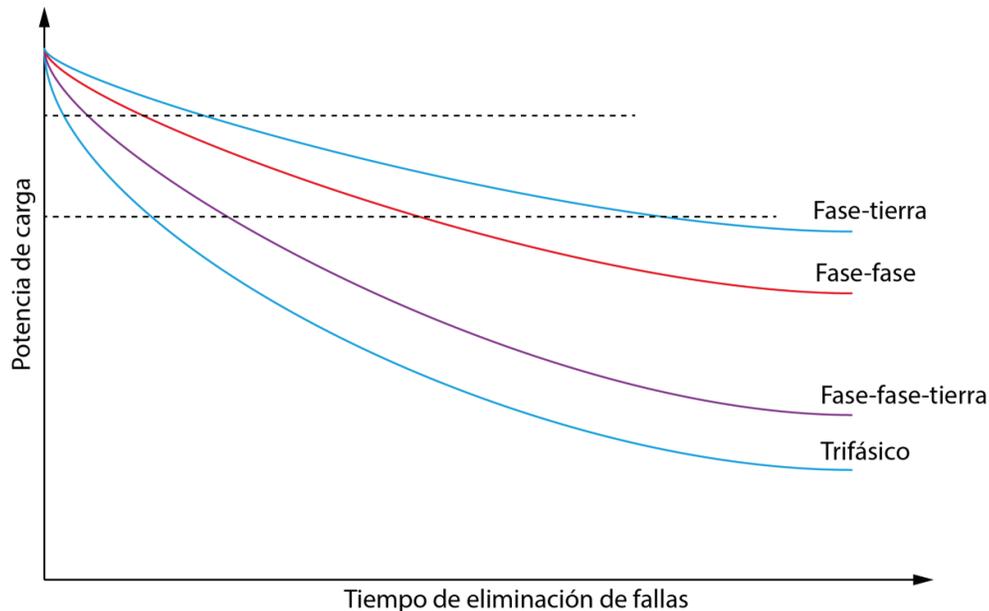
### 2.1.3.3. Velocidad

Otra característica con la que debe contar un sistema de protección es la velocidad, puesto que es necesario desconectar el equipo fallado lo más pronto posible. Las ventajas de contar con una alta velocidad de operación y aislamiento de la falla, son las siguientes [1], [4]:

- Disminuye el daño de los equipos.
- Mejora la estabilidad del sistema.
- Disminuye las perturbaciones en el resto del sistema.
- Aumenta la efectividad de las reconexiones automáticas.
- Evita que una falla simple, se propague al resto de las fases.

Uno de los principales objetivos que se tiene al aislar las fallas a la brevedad posible, es salvaguardar la continuidad del suministro antes de que la falla conduzca a una pérdida generalizada de sincronismo, y en consecuencia, al colapso del sistema de potencia. A medida que aumenta la carga en un sistema de potencia, también aumenta el desplazamiento de fase entre los voltajes de las diferentes barras del sistema, y por lo tanto, también aumenta la probabilidad de que el sincronismo se pierda cuando una falla perturba el sistema. Es por eso que, cuanto mayor sea la carga del sistema, más corto será el tiempo que se

permita mantener a una falla en el sistema. La Figura 2.3 muestra las relaciones típicas entre la carga del sistema y los tiempos de eliminación de fallas. Se puede observar que, para una misma potencia de carga, el tiempo en que debe aislarse un cortocircuito trifásico es menor que en los otros casos; por tanto tiene un efecto más marcado en la estabilidad del sistema que un cortocircuito a tierra, ya que este último dispone de un tiempo mayor para ser despejado.



**Figura 2.3.** Relación típica de potencia-tiempo para varios tipos de fallas [7].

Sin embargo, la estabilidad del sistema no es la única consideración que debe hacerse. El funcionamiento rápido de las protecciones garantiza minimizar el daño (causado por la falla) a equipos. La energía perjudicial liberada durante una falla es proporcional al tiempo que esa falla está presente, por eso es importante que la protección funcione lo más rápidamente posible. No obstante, la velocidad de operación debe compararse con la economía. Por ejemplo, los circuitos de distribución, normalmente no requieren una eliminación rápida de fallas. En contraste, las plantas de generación y los sistemas de EHV<sup>5</sup> exigen protecciones de máxima velocidad y fiabilidad [7].

<sup>5</sup> **EHV:** Extra Alto Voltaje o Voltaje Extra Alto. Son las siglas del vocablo inglés *Extra High Voltage*. De acuerdo a la Norma IEEE Std 1312-1993, se refiere al voltaje máximo de un sistema, mayor a 242 kV, pero inferior a 1000 kV [9].

#### **2.1.3.4. Confiabilidad**

La confiabilidad es un requisito básico del sistema de protecciones, y se define como la probabilidad de que dicho sistema funcione correctamente cuando se requiere que actúe. Esta confiabilidad tiene dos aspectos:

1. El sistema debe funcionar en presencia de una falla que esté dentro de su zona de protección.
2. El sistema debe abstenerse de funcionar, innecesariamente, en caso de fallas fuera de su zona de protección, o en ausencia de falla [2].

La simplicidad y la robustez son otras características que contribuyen a la confiabilidad de un sistema de protección. En realidad, existen muchas consideraciones de diseño que podrían mencionarse, como la mano de obra, la presión de contacto, los materiales de contacto y las disposiciones para prevenir la contaminación, que son sumamente importantes. Pero ninguno de los atributos mencionados, por sí solos, representa la solución completa.

Por otra parte, conviene señalar que el sistema de protección permanece inactivo la mayor parte del tiempo, a diferencia de los otros componentes del sistema eléctrico de potencia. Algunos relés de protección pueden actuar una vez en varios años, incluso los relés con mayor frecuencia de operación, que son aquellos ubicados en las líneas de transmisión, pueden funcionar pocas veces al año. Por esta razón, es indispensable realizar una compensación mediante pruebas y procedimientos de mantenimiento para asegurar que los equipos de protección actuarán correctamente cuando el sistema de potencia lo requiera [1].

#### **2.1.4. CLASIFICACIÓN DE LOS RELÉS DE PROTECCIÓN**

Los relés de protección pueden ser clasificados considerando diferentes criterios, por ejemplo, en base a su función general, principio de operación o construcción, tecnología, señal de entrada, tipo de protección, función específica, entre otros. En esta sección, únicamente se enumeran las diferentes clases de relés de acuerdo a su tecnología, por ser las más mencionadas y, de acuerdo a su tipo de protección, por permitir una introducción a los conceptos de direccionalidad.

#### **2.1.4.1. Según su tecnología**

A lo largo de los años, los relés han cambiado su tecnología de fabricación. Los relés más antiguos, en todas sus formas han sido renovados progresivamente, y cada cambio ha traído consigo reducciones de tamaño y mejoras de funcionalidad. Los niveles de confiabilidad también se han mantenido o incluso han mejorado, y la disponibilidad ha aumentado significativamente debido a técnicas no disponibles con los anteriores tipos de relés. A continuación se listan las diferentes clases [7]:

- Electromecánicos.
- Estáticos.
- Digitales.
- Numéricos.

#### **2.1.4.2. Según sus características de funcionamiento o tipo de protección**

Existe una gran variedad de protecciones que los relés brindan a los elementos de un sistema de potencia. A continuación se presentan los principales tipos de relés conforme a sus características de funcionamiento [8], [10]:

- Relés de sobrecorriente.
- Relés diferenciales.
- Relés de distancia.
- Relés de potencia inversa.
- Relés de tiempo inverso.
- Relés de tiempo definido.
- Relés de sobre y bajo voltaje.
- Relés de fase
- Relés de tierra.
- Relés pilotos:
  - Relés de comparación de fase.
  - Relés de comparación direccional.
  - Relés de corriente diferencial.

Debido a los objetivos de este trabajo, en la siguiente sección se presentan algunos conceptos concernientes a la protección de sobrecorriente, tema que abarca varias subsecciones, incluyendo fundamentos de un relé direccional de sobrecorriente (convencional).

### 2.1.5. RELÉS DE SOBRECORRIENTE

La protección contra el exceso de corriente fue, justamente, el primer sistema de protección que se desarrolló. Es la protección más antigua, sencilla, económica y relativamente confiable. Su principio de operación se basa en el parámetro de la corriente de falla. Si los relés son del tipo electromecánico, utilizan dos principios fundamentales: atracción electromagnética e inducción electromagnética. Pero como se vio en la Sección 2.1.4.1, existen otros relés modernos que pueden ser estáticos, digitales y numéricos.

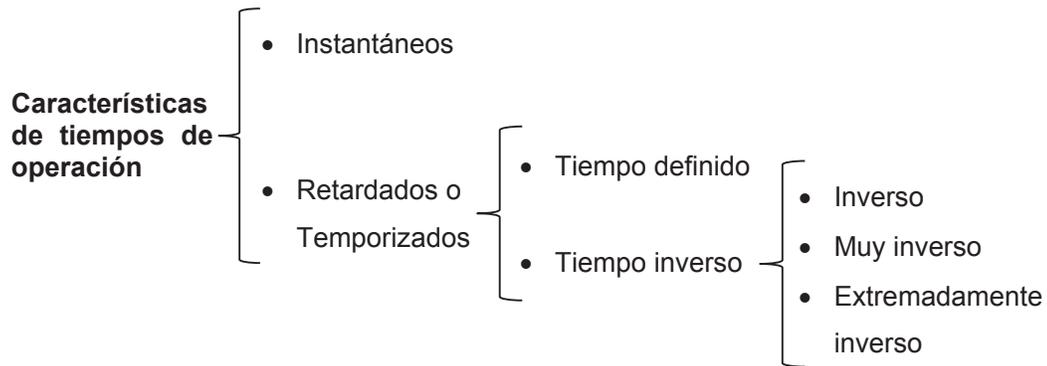
El objetivo primario de toda protección es detectar la falla y aislarla mediante la apertura del interruptor. La protección de sobrecorriente cumple totalmente con esta función. El segundo objetivo es la selectividad y velocidad del despeje de falla, que puede conseguirse en cierto grado con una adecuada coordinación de relés de sobrecorriente por escalonamiento de tiempo, de corriente; y con la función de direccionalidad [8].

Para una correcta aplicación de los relés de sobrecorriente, es necesario considerar dos variables fundamentadas en términos de *selectividad* y *velocidad* (ver Sección 2.1.3). Estas variables son:

1. Nivel de corriente mínima de operación.
2. Característica de tiempos de operación.

La corriente mínima de operación  $I_{op}$ , también denominada *corriente de pick-up*, es aquella que produce el cambio de estado del relé (los contactos de disparo pasan de abierto a cerrado).

Por su parte, las características de tiempos de operación son variadas y se pueden resumir en la Figura 2.4 [4]:



**Figura 2.4.** Características de tiempos de operación [4].

### 2.1.5.1. Ecuación característica de los relés de sobrecorriente

Las ecuaciones (2.1) y (2.2) definen las características de tiempo *Instantáneo* y *Tiempo Definido*.

$$\text{Instantáneo:} \quad t = 0 \text{ para } I \geq I_{op} \quad (2.1)$$

$$\text{Tiempo definido:} \quad t = K \text{ para } I \geq I_{op} \quad (2.2)$$

Donde:

$K$ : Tiempo después del cual actuará el relé en [s].

La ecuación (2.1) representa un caso ideal: se considera que el relé no requiere ningún periodo de tiempo para actuar. En realidad, los relés siempre toman unos milisegundos debido a la inercia de la protección. Por ejemplo, en la Figura 2.5 se tiene un caso específico de Schneider Electric, donde la protección cuenta con un tiempo de 30 ms. En este caso, la ecuación sería:  $t = 0,03 \text{ s}$ .

Por otra parte, los relés temporizados inversos se modelan generalmente mediante ecuaciones exponenciales, las cuales se indican a continuación [11]:

Norma IEC 255-3:

$$t = Dial \times \frac{\beta}{\left[\frac{I}{I_s}\right]^\alpha - 1} \quad (2.3)$$

Norma ANSI / IEEE Std. C37.112-1996:

$$t = Dial \times \left( \frac{\beta}{\left[ \frac{I}{I_s} \right]^\alpha - 1} + c \right) \quad (2.4)$$

Donde:

- $t$ : Tiempo de operación del relé en [s].
- $Dial$ : Ajuste del multiplicador de tiempo. Factor que permite obtener diferentes tiempos de operación para un mismo tipo de curva de operación en [s].
- $I$ : Corriente de operación del relé en [A].
- $I_s$ : Corriente de ajuste o TAP en [A].

Las constantes  $\alpha$ ,  $\beta$ ,  $c$ , determinan el grado de característica inversa del relé, y sus valores normalizados se indican en la Tabla 2.1 y Tabla 2.2. De las ecuaciones presentadas, la ecuación (2.3) establecida por la Norma IEC 255-3, es la más utilizada para todas las curvas temporizadas [11].

**Tabla 2.1.** Constantes para la ecuación exponencial – Norma IEC 255-3 [11].

TIPO DE CURVA	$\alpha$	$\beta$
Normalmente inversa	0,02	0,14
Muy inversa	1,00	13,50
Extremadamente inversa	2,00	80,00

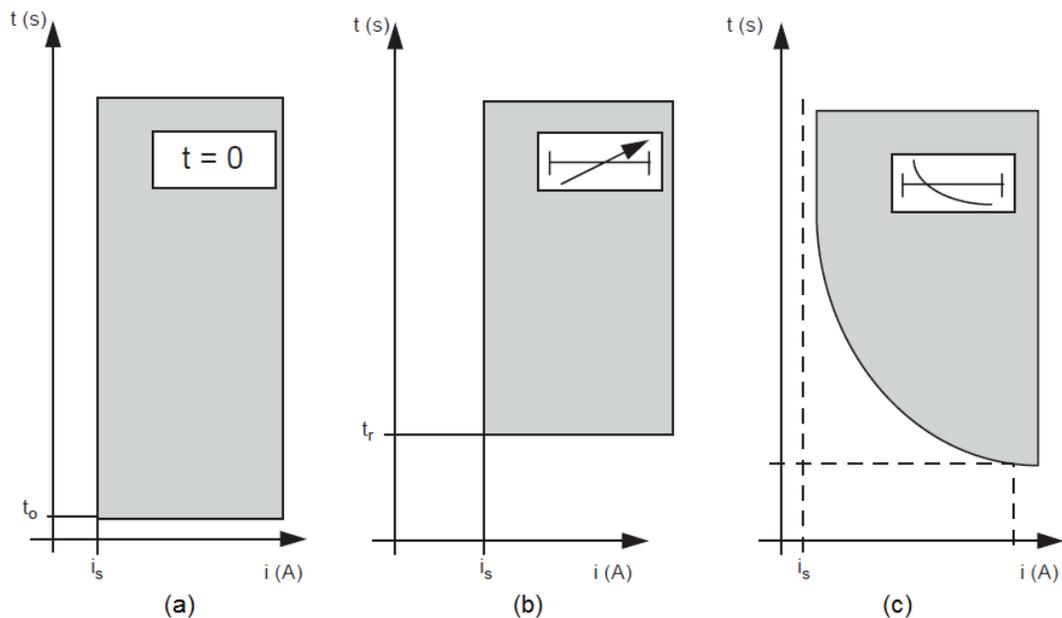
**Tabla 2.2.** Constantes para la ecuación exponencial – Norma IEEE Std. C37.112 [12].

TIPO DE CURVA	$\alpha$	$B$	$c$
Moderadamente inversa	0,02	0,0515	0,114
Muy inversa	2,00	19,61	0,491
Extremadamente inversa	2,00	28,20	0,1217

### 2.1.5.2. Curvas características de los relés de sobrecorriente

De acuerdo a su característica de operación, los relés de sobrecorriente pueden agruparse en tres clases que se listan a continuación, y cuyas gráficas se muestran en la Figura 2.5, respectivamente.

1. De corriente definida o instantánea
2. De tiempo definido
3. De tiempo inverso



**Figura 2.5.** Características de operación tiempo-corriente de los relés de sobrecorriente a) Instantánea, b) Temporizada a tiempo definido, c) Temporizada por curva (tiempo inverso) [13].

### 2.1.5.3. Clasificación de relés de sobrecorriente según su sentido de detección de corriente

Los relés de sobrecorriente se pueden clasificar en dos tipos de acuerdo al sentido de detección de la corriente [11]:

- *No direccional*: Opera para sobrecorrientes que circulan en ambos lados de su ubicación. Ejecuta su función sin importar el sentido en que circula la corriente. En la norma ANSI / IEEE Std. C37.2-2008 [14] se designan por los números 50 y 51. Normalmente son utilizados en sistemas radiales.
- *Direccional*: Solo opera cuando verifica que la corriente sobrepasa el valor de arranque, y que además circula en el sentido de ajuste asignado, dentro de su zona protegida. La direccionalidad se logra gracias a una señal adicional de referencia: el voltaje. En la norma ANSI / IEEE Std. C37.2-2008, este relé se designa por el número 67.

Por supuesto que las características de tiempos de operación, utilizadas en las protecciones de sobrecorriente no direccionales, sean instantáneas, de tiempo definido o de tiempo inverso, son las mismas que se emplean en las protecciones de sobrecorriente direccionales.

En la siguiente sección, se amplía la descripción del relé de sobrecorriente direccional, que es uno de los elementos de estudio en este trabajo.

#### **2.1.5.4. Relé de sobrecorriente direccional**

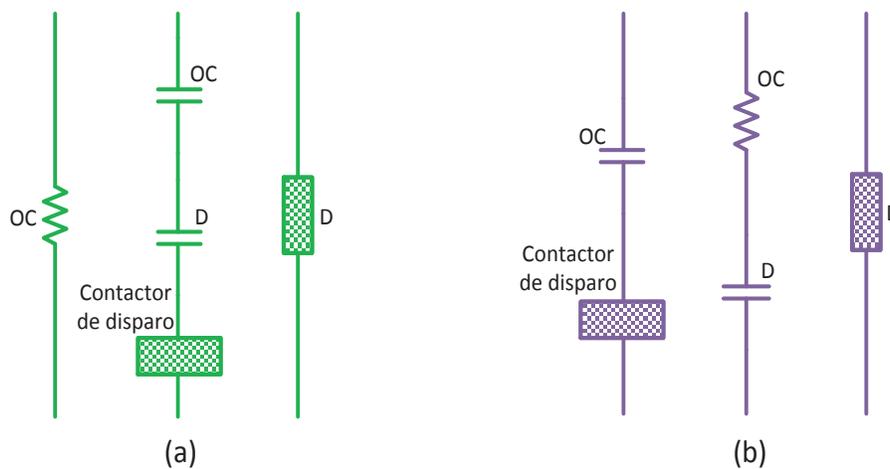
De manera general, la función principal de las unidades direccionales es limitar la operación del relé en una dirección especificada. En ese sentido, los relés de sobrecorriente direccionales están constituidos por dos unidades, una que mide la magnitud de la corriente (unidad de sobrecorriente), y otra que determina la dirección del flujo de potencia en el elemento del sistema asociado (unidad direccional), es decir, que puede controlar o supervisar direccionalmente a los esquemas de protección de sobrecorriente.

La segunda unidad, además de la corriente del relé, también requiere una señal de referencia para medir el ángulo de la falla y así determinar si el relé debe operar o no. Generalmente, la señal de referencia o de polarización es un voltaje, pero también puede ser una corriente de entrada. Con el trabajo conjunto de las dos unidades, el disparo del interruptor solo se produce cuando la dirección del flujo concuerda con la ajustada en el relé, y la corriente que circula es mayor al valor de *pick-up* preestablecido. Si únicamente la corriente supera ese valor de *pick-up*, y el sentido del flujo no coincide con el configurado, no hay orden de apertura [8], [15].

Básicamente existen dos maneras en las que el elemento direccional actúa sobre el elemento de sobrecorriente para obtener la dirección del flujo de potencia:

- Supervisión
- Control

Ambos casos están ilustrados en la Figura 2.6, donde D indica la unidad direccional y OC la unidad de sobrecorriente [8].



**Figura 2.6.** Métodos para obtener la dirección del flujo de potencia.  
a) Por supervisión, b) Por control [8].

*Supervisión direccional:* Los contactos del relé direccional y de sobrecorriente se conectan en serie con el contactor de disparo; es decir, se requiere que operen los dos elementos (cerrando sus contactos) para cerrar el circuito y permitir que el interruptor se abra. En este caso, se dice que el relé de sobrecorriente está direccionalmente supervisado.

*Control direccional:* Los contactos del relé direccional y de sobrecorriente se combinan de tal manera que, mientras no opere la unidad direccional, tampoco se habilita la unidad de sobrecorriente. Como se muestra en la Figura 2.6b, el circuito de disparo del interruptor solo incluye contactos del relé de sobrecorriente, puesto que el elemento direccional debe operar previamente. Se dice entonces que el relé de sobrecorriente está direccionalmente controlado.

Es recomendable usar el método controlado para determinar la dirección del flujo de potencia, ya que la unidad de sobrecorriente solo se habilita cuando el flujo está en la dirección correcta; mientras que en el método de supervisión, la unidad de sobrecorriente podría activarse de forma errónea para una dirección incorrecta del flujo. Además, cuando el interruptor es abierto en un sistema en anillo, los flujos de corriente cambiarán y esto puede conducir a la posibilidad de perder la coordinación [8].

#### 2.1.5.4.1. Principio de operación

El principio de operación del relé de sobrecorriente direccional es similar al de un medidor de potencia o vatímetro, en el hecho de tomar las mismas señales (voltaje y corriente) para su operación. Sin embargo, el principio del relé es diferente porque el voltaje es usado como una referencia para establecer la dirección de la corriente y, pese a contar con estas señales, no puede responder a la potencia actual del sistema. El torque de operación del relé direccional puede definirse como [8]:

$$T = K \phi_1 \phi_2 \sin \theta \quad (2.5)$$

Donde:

- $\phi_1$ : Valor de polarización proporcional a la corriente.
- $\phi_2$ : Valor de polarización proporcional al voltaje.
- $\theta$ : Ángulo entre  $\phi_1$  y  $\phi_2$ .

El torque es positivo si  $0^\circ < \theta < 180^\circ$ , y negativo si  $180^\circ < \theta < 360^\circ$ . Debe notarse que  $\theta$  está en fase con la corriente pero atrasado con respecto al voltaje puesto que  $V = -\frac{d\phi}{dt}$ .

Si la corriente y el voltaje están en fase, los flujos están desfasados en  $90^\circ$ . Por lo tanto, el ángulo para máximo torque ocurre cuando la corriente y el voltaje del relé están en fase. Esto puede obtenerse muy fácilmente usando la corriente y el voltaje de la misma fase. Sin embargo, esto no es práctico ya que para una falla en una fase, el voltaje de esa fase puede reducirse a cero. Es por lo tanto, práctica común usar el voltaje de una fase diferente [8].

#### 2.1.5.4.2. Conexiones de los relés direccionales de sobrecorriente

El voltaje de polarización de los relés direccionales de sobrecorriente tiene que ser escogido apropiadamente para que en el momento de un cortocircuito, cuando el relé debe operar, no se haga cero; de lo contrario, el torque del relé se haría cero y el relé no podría operar. El voltaje de la fase afectada durante un cortocircuito puede tener valores muy bajos (podría reducirse a cero); por esta razón, como regla general, no se polariza un relé con el mismo voltaje de la fase

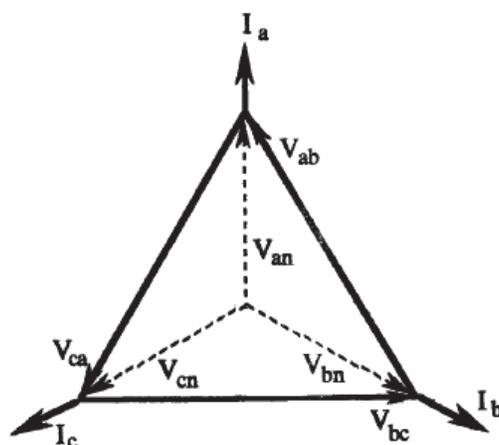
cortocircuitada. En cuanto a la corriente, no hay riesgo de que se haga cero, ya que se trata de la corriente de cortocircuito.

También debe escogerse adecuadamente el ángulo de máximo torque con el fin de favorecer la operación del relé. Este ángulo, que se mide respecto al voltaje de referencia, se da cuando la corriente aplicada al relé se encuentra en fase con el voltaje aplicado a dicho relé. En condiciones de falla, para conseguir un factor de potencia igual a uno en el relé (y por tanto el ángulo de máximo torque) es necesario elegir un voltaje y una corriente tales que su desfase sea igual al ángulo de la impedancia de la línea, estando la corriente en adelanto con respecto al voltaje. De este modo, al ocurrir un cortocircuito, la corriente se retrasará en el ángulo de impedancia de la línea quedando en fase con el voltaje [5], [8].

Se han estudiado varias combinaciones de voltajes de polarización, cuyo nombre indica el número de grados en que una corriente a un factor de potencia unitario adelanta al voltaje de polarización. Estas conexiones son [5]:

- Conexión a  $90^\circ$  o en cuadratura (Figura 2.7).
- Conexión a  $60^\circ$  con transformadores de corriente en estrella (Figura 2.8).
- Conexión a  $60^\circ$  con transformadores de corriente en delta o triángulo (Figura 2.9).
- Conexión a  $30^\circ$  (Figura 2.10).

De todas ellas, la más común es la conexión en cuadratura [5].



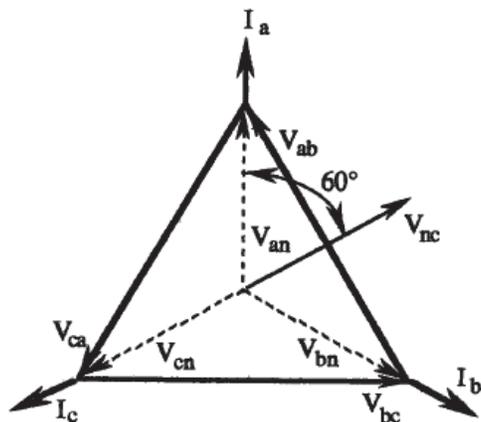
**Voltajes y corrientes utilizados:**

**Relé 1:  $I_a, V_{bc}$**

**Relé 2:  $I_b, V_{ca}$**

**Relé 3:  $I_c, V_{ab}$**

**Figura 2.7.** Conexión a  $90^\circ$  o conexión en cuadratura [5].



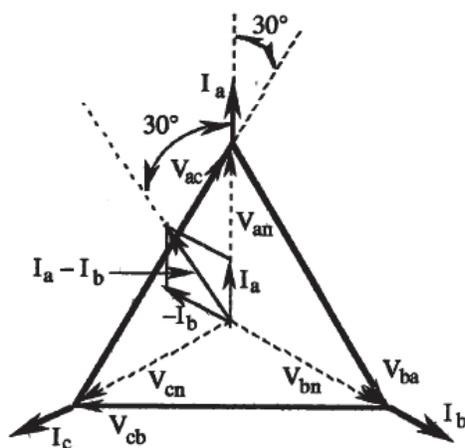
**Voltajes y corrientes utilizados:**

**Relé 1:  $I_a, V_{nc}$**

**Relé 2:  $I_b, V_{na}$**

**Relé 3:  $I_c, V_{nb}$**

**Figura 2.8.** Conexión de  $60^\circ$  con transformadores de corriente en estrella [5].



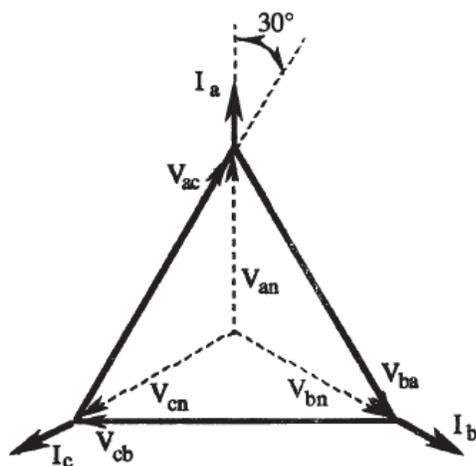
**Voltajes y corrientes utilizados:**

**Relé 1:  $I_a - I_b, V_{ac}$**

**Relé 2:  $I_b - I_c, V_{ba}$**

**Relé 3:  $I_c - I_a, V_{cb}$**

**Figura 2.9.** Conexión de  $60^\circ$  con transformadores de corriente en triángulo [5].



**Voltajes y corrientes utilizados:**

**Relé 1:  $I_a, V_{ac}$**

**Relé 2:  $I_b, V_{ba}$**

**Relé 3:  $I_c, V_{cb}$**

**Figura 2.10.** Conexión de  $30^\circ$  [5].

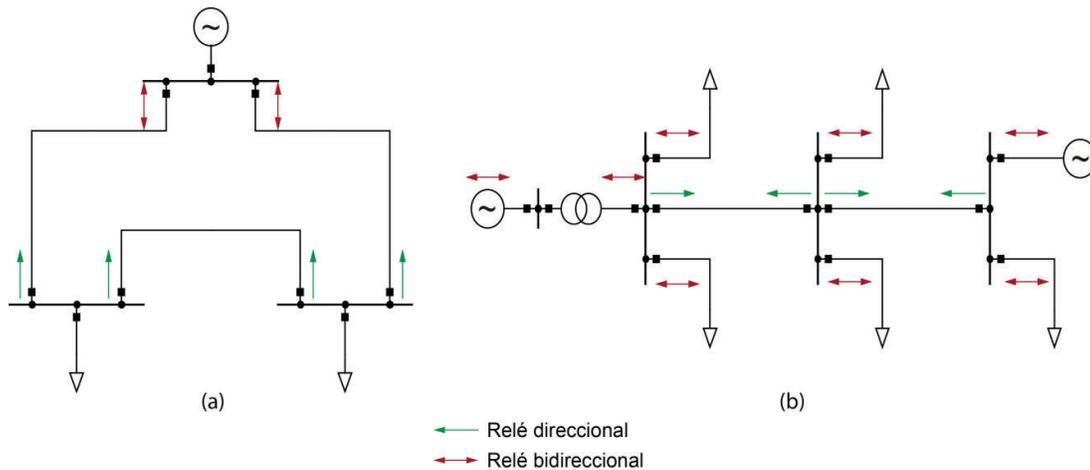
La unidad direccional de fase debe ser rápida debido a la posibilidad de reversión repentina de la dirección de la corriente cuando se elimina una falla y se reanuda la carga. Los relés de fase pueden detectar fallas a tierra, pero generalmente se utiliza un relé polarizado con el voltaje de secuencia cero para detectar este tipo de fallas. Este relé, denominado relé de neutro o de tierra, se puede ajustar con una corriente de operación muy baja, ya que mide la corriente del neutro, que en condiciones normales debería ser cero, mientras que los de fase tienen que ajustarse con una corriente superior a la de carga [5].

#### *2.1.5.4.3. Algunas aplicaciones*

La protección de sobrecorriente direccional es usada cuando es necesario proteger el sistema contra corrientes de falla que pueden circular en ambas direcciones a través de un elemento del sistema, y cuando la protección de sobrecorriente no direccional puede producir desconexión innecesaria de circuitos. Esto puede suceder en sistemas en anillo o mallados, y en sistemas de múltiples fuentes. En tales casos, sería imposible obtener una correcta selectividad mediante el uso de un relé de sobrecorriente no direccional. Si la misma magnitud de la corriente de falla pudiera fluir en cualquier dirección respecto a la ubicación de un relé no direccional, la coordinación con los relés adelante y atrás no puede lograrse excepto en configuraciones de sistema muy inusuales. Por lo tanto, la protección de sobrecorriente se hace direccional para proporcionar la coordinación del relé entre todos los demás que puedan ver una falla dada. Como ejemplo se muestra el uso de relés de sobrecorriente direccionales en los dos casos de la Figura 2.11 [8], [16].

#### *2.1.5.4.4. Protección de Líneas de Transmisión con Relés de Sobrecorriente Direccional*

Las líneas de transmisión son la parte del sistema de potencia más expuesta a fallas a causa de las descargas atmosféricas, rotura de sus aisladores, daño de sus torres, etc. de manera análoga se puede decir que las líneas de transmisión son el sistema circulatorio de un país a través del cual fluye la energía de los centros de generación hasta los centros de consumo.



**Figura 2.11.** Ejemplos de aplicación de los relés direccionales de sobrecorriente  
a) Sistema en anillo, b) Sistema mallaado [8].

De acuerdo al nivel de su voltaje, que está ligado a la magnitud de la potencia que transportan, las líneas se clasifican en líneas de transmisión, subtransmisión y distribución. Sin embargo su nivel de protección depende de su importancia dentro del sistema de potencia, desde el punto de vista de continuidad del servicio, estabilidad del sistema, entre otros.

La protección de líneas con relés de sobrecorriente es la más económica y sencilla, pero la más difícil de aplicar por los problemas de coordinación y selectividad que se presentan, y la que más rápido necesita reajuste a medida que cambia el sistema. Sin embargo, utilizando la amplia gama de características de tiempo de operación que existe y las restricciones de tipo direccional, se puede superar estos problemas, pero generalmente, a cambio de tiempos de operación muy grandes [5]. Se le usa por lo general para protección contra la falla fase-fase o fase a tierra, en los circuitos de servicios propios de la subestación, en los circuitos de distribución y en los sistemas industriales.

Una vez descritos los conceptos básicos relacionados con los sistemas de protecciones, conviene describir las herramientas que se utilizaron en el presente trabajo de titulación para el tratamiento de datos.

## 2.2. DESCRIPCIÓN DE HERRAMIENTAS EMPLEADAS PARA PROCESAMIENTO DE DATOS

En esta sección se describen dos herramientas sumamente importantes en el desarrollo de este trabajo, con base en las cuales fue posible la obtención de un modelo de patrones que determina la direccionalidad de una falla dada. Estas herramientas son:

- Transformada Wavelet Discreta.
- Máquinas de Soporte Vectorial.

### 2.2.1. TRANSFORMADA WAVELET

Existen numerosos algoritmos basados en herramientas tradicionales que se usan en el análisis de distorsiones en sistemas eléctricos de potencia. Muchos de ellos han sido útiles para señales en estado estacionario; sin embargo se necesita una herramienta más potente para analizar sistemas que presenten eventos con formas de onda no periódicas, o distorsiones no estacionarias.

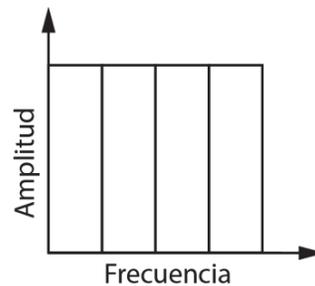
La Transformada de Fourier (FT) es una herramienta matemática que permite el análisis de señales, dando a conocer su contenido frecuencial. La expresión analítica de la FT se indica en la ecuación (2.6) [17].

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \quad (2.6)$$

La información que se obtiene mediante la FT se deriva de un promedio de la contribución de las frecuencias, esto hace que se pierda la localización de los fenómenos en el dominio del tiempo, y por tanto es aplicable únicamente a señales estacionarias, es decir, señales cuyo contenido frecuencial no varíe con el tiempo. Los resultados de la FT se presentan en el dominio de la frecuencia, como se observa en la Figura 2.12.

Si se desea analizar una señal no estacionaria, como corrientes de cortocircuitos o eventos transitorios en general, se necesita conocer simultáneamente la información temporal y frecuencial, esto significa disponer de una representación

tiempo-frecuencia de la señal, y por ello la FT es inadecuada. Para resolver este problema, se introdujo la *Short-Time Fourier Transform* (STFT).



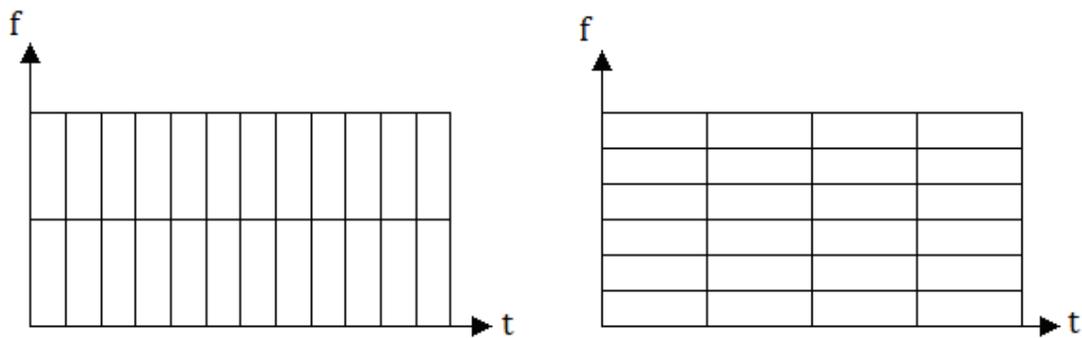
**Figura 2.12.** Dominio de frecuencia para la FT [18].

La STFT es una versión modificada de la Transformada de Fourier. Aquí la señal no estacionaria es dividida en pequeños trozos, en los que se supone que la señal sí es estacionaria, y a los cuales se aplica la FT. Esto se realiza utilizando una función ventana de ancho  $w$  que va siendo desplazada y multiplicada con la señal. Su expresión matemática se indica a continuación [17]:

$$STFT(b, f) = \int_t [x(t) w(t - b)] e^{-j2\pi ft} dt \quad (2.7)$$

Donde  $w(t)$  es la función ventana de anchura finita y  $b$  es el parámetro de traslación de la ventana.

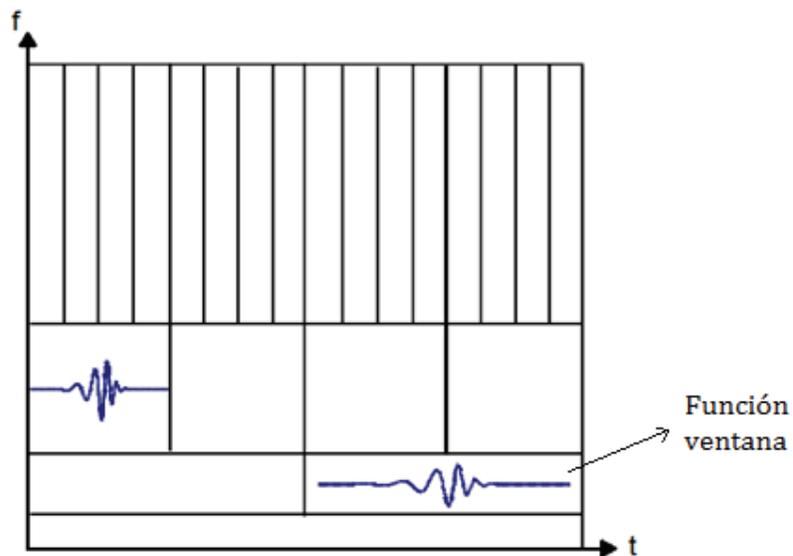
En la STFT, las resoluciones frecuenciales y temporales son fijas y quedan determinadas por el ancho de la función ventana, misma que se selecciona una sola vez durante todo el análisis, es decir que la resolución tanto en tiempo como en frecuencia permanecen constantes. De esta manera se debe elegir una ventana apropiada para el análisis, dependiendo de la aplicación: si la ventana es estrecha la resolución temporal es buena pero la resolución frecuencial es pobre, siendo difícil saber exactamente las componentes frecuenciales que existen en la señal; si la ventana es ancha, mejora la resolución frecuencial, pero la resolución temporal es pobre, además una ventana ancha puede infringir la condición de estacionariedad. En la Figura 2.13 se muestran las dos condiciones mencionadas en el plano tiempo-frecuencia.



**Figura 2.13.** Dominio tiempo-frecuencia para la STFT [18].

La Transformada Wavelet (WT), en contraste con la STFT, analiza la señal a distintas frecuencias con diferentes resoluciones, es decir, cada componente espectral no se analiza con la misma resolución. La WT utiliza un análisis multi-resolución, que implica ventanas estrechas en frecuencias altas y ventanas anchas en bajas frecuencias; en otras palabras, cuando la resolución de frecuencia aumenta, la resolución temporal disminuye, y viceversa.

Ya que la WT incluye evidentemente información de tiempo y frecuencia, su representación gráfica se realiza en el denominado plano *tiempo-frecuencia*, como se muestra en la Figura 2.14.



**Figura 2.14.** División del plano tiempo-frecuencia mediante WT [17].

Cada celda de la Figura 2.14 representa una porción igual del plano tiempo-frecuencia, pero con variaciones en la resolución temporal y resolución frecuencial. Esto quiere decir que las celdas tienen una superficie constante, y están determinadas por el Principio de Incertidumbre de Heisenberg. A bajas frecuencias, la altura del rectángulo es menor (mejor resolución de frecuencia) pero la anchura es grande (pobre resolución temporal). Por el contrario, a altas frecuencias la anchura del rectángulo decrece (mayor resolución temporal) y la altura aumenta (menor resolución frecuencial). Resumiendo, se tiene:

- i) *Para señales de baja frecuencia:* alta resolución en frecuencia y baja resolución en tiempo.
- ii) *Para señales de alta frecuencia:* alta resolución en tiempo y baja resolución en frecuencia.

Justamente el sello distintivo de las wavelets (funciones ventana o de análisis en la WT) es su capacidad de cambiar la escala de observación para estudiar características con diferentes escalas.

Debido a las ventajas que presenta la WT en el análisis y procesado de señales no estacionarias, existen múltiples aplicaciones que han ido incrementándose en los últimos años, dentro de diversos campos relacionados con la compresión de datos, las comunicaciones, el procesado de voz, imagen, señales sísmicas o geológicas, señales médicas, señales en sistemas de potencia, etc. Esta última es de particular interés, ya que en un sistema eléctrico se presentan variaciones temporales de frecuencia, huecos de voltaje, sobrevoltajes, cortocircuitos, interrupciones, y demás distorsiones. En estos casos, la Transformada Wavelet se considera una herramienta muy apropiada [17].

A continuación se detallan dos tipos de Transformada Wavelet que pueden utilizarse en diferentes aplicaciones.

#### **2.2.1.1. La Transformada Wavelet Continua**

La Transformada Wavelet Continua (CWT) se define mediante la siguiente ecuación [17]:

$$X_{WT}(a, b) = \int_{-\infty}^{\infty} x(t) \Psi_{a,b}(t) dt \quad (2.8)$$

Donde  $x(t)$  es la señal a analizar, y  $\Psi_{a,b}$  es la “wavelet madre” o función básica (función ventana), que está definida por [17]:

$$\Psi_{a,b} = a^{-1/2} \Psi\left(\frac{t-b}{a}\right) \quad (2.9)$$

La wavelet madre  $\Psi$ , es una función ventana de longitud finita, corta y de naturaleza oscilatoria, esto significa que su promedio es cero y decae rápidamente en ambos extremos. En otras palabras, para que  $\Psi(t)$  pueda ser wavelet madre debe satisfacer las siguientes condiciones [17]:

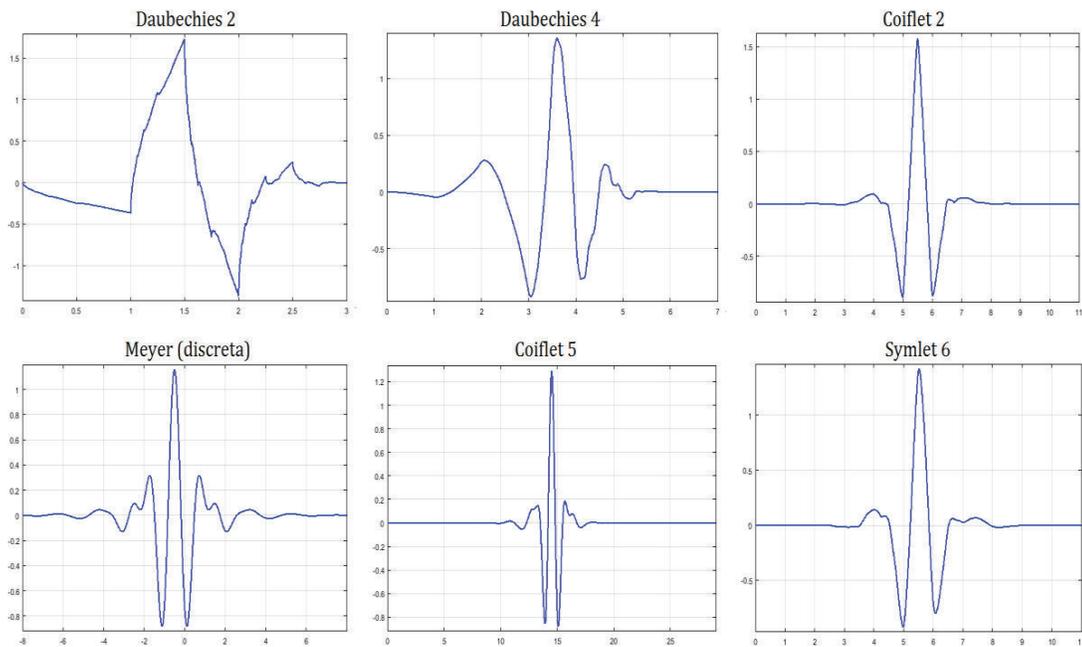
- i) Debe ser continua e integrable.
- ii) Debe tener una Transformada de Fourier  $\Psi(w) = 0$  para  $w < 0$ .
- iii) Debe tener un valor medio nulo, es decir, debe cumplir que:

$$\int_{-\infty}^{\infty} \Psi(t) dt = 0 \quad \text{y} \quad \int_{-\infty}^{\infty} \frac{|\Psi(w)|^2}{w} dw = 0.$$

El término *wavelet* significa “onda pequeña”, refiriéndose como *pequeña* al hecho que la función integral (2.8) es finita, y el término *onda* se refiere a la condición de poseer naturaleza oscilatoria. El término *madre* da a entender que es un prototipo para generar otras funciones ventana que se usan en el proceso de transformación [19].

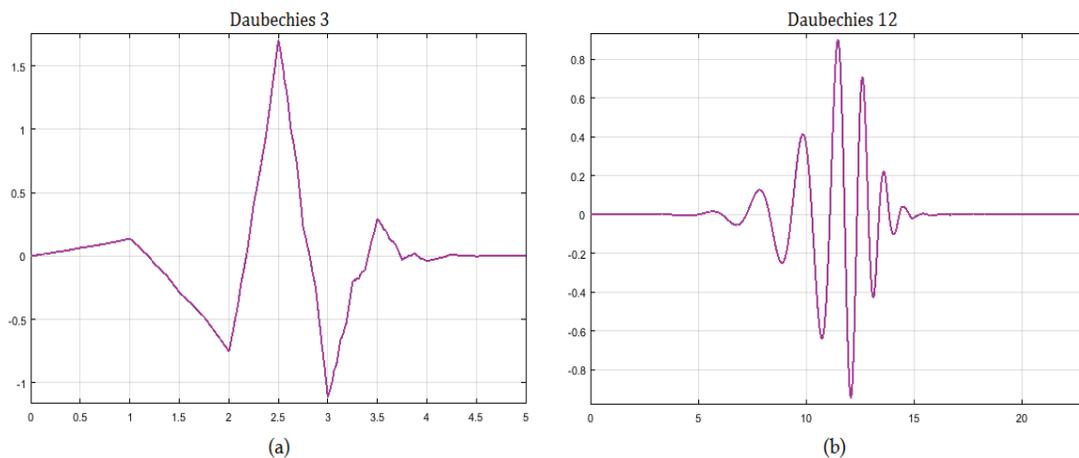
Al igual que la FT representa una señal basándose en la suma de funciones senoidales a diferentes frecuencias, la WT utiliza como elemento básico a la wavelet madre  $\Psi(t)$ . La diferencia consiste en que la FT descompone la señal únicamente con funciones seno; mientras que la *wavelet madre* puede ser una de muchas funciones [20] que han sido desarrolladas por investigadores en distintas disciplinas, como Jean Morlet y Alex Grossmann, J. Strömberg, Stephane Mallat, Yves Meyer, Ingrid Daubechies, C. K. Chui, entre otros. Sus estudios han contribuido significativamente en el desarrollo de técnica WT.

Algunas de las wavelets madres más utilizadas se muestran en la Figura 2.15.



**Figura 2.15.** Algunas de las wavelets madre más utilizadas [17].

La forma final de la *wavelet madre* depende del número de coeficientes que se han utilizado para la generación de la función de escalonamiento. A menor número de coeficientes la señal es más abrupta, y a un mayor número, resulta una función más suavizada [21]. Por ejemplo, en la Figura 2.16 se muestran dos wavelets de la familia “Daubechies”, una con 3 y otra con 12 coeficientes.



**Figura 2.16.** Familia Wavelet “Daubechies”.  
a) 3 coeficientes, b) 12 coeficientes [21].

La Transformada Wavelet depende de dos parámetros:

- El parámetro  $b$ , traslación, está relacionado con el desplazamiento de la ventana a lo largo del dominio temporal de la señal y por tanto este término corresponde a la información temporal de la señal.
- El parámetro  $a$ , escala, es definido como  $[1/frecuencia]$  y corresponde a la información frecuencial. El escalado es una operación matemática que dilata o comprime una señal. Las escalas pequeñas corresponden a señales comprimidas (alta frecuencia) y las escalas grandes a señales dilatadas (baja frecuencia). Si  $a > 1$  la señal se contrae y si  $a < 1$  la señal se expande.

La Transformada Wavelet Continua CWT se calcula cambiando la escala de la ventana de análisis, desplazándola en el tiempo, multiplicándola por la señal e integrando en el tiempo. Puesto que la transformación se consigue dilatando y trasladando la wavelet madre en forma continua, se genera información redundante. Para ello, los parámetros de escala (dilatación) y traslación se discretizan como  $a = a_0^m$  y  $b = nb_0 a_0^m$ , donde  $a_0$  y  $b_0$  son constantes fijas tal que  $a_0 > 1$  y  $b_0 > 0$  y  $m, n \in Z$  [17].

Así, la wavelet madre discretizada se define como:

$$\Psi_{m,n} = a_0^{-1/2} \Psi \left( \frac{t - nb_0 a_0^m}{a_0^m} \right) \quad (2.10)$$

### 2.2.1.2. La Transformada Wavelet Discreta

La Transformada Wavelet Discreta (DWT) es una versión muestreada de la CWT, y se utiliza para evitar la extensa generación de datos en forma de coeficientes wavelet, ya que esto conduce a una gran carga computacional. La DWT reduce significativamente el tiempo y los recursos empleados por la CWT, y además es sencilla de implementar y proporciona suficiente información para el análisis y síntesis de una señal [17].

Una señal de tiempo continuo puede representarse en forma discreta siempre que la frecuencia de muestreo se elija correctamente. Esto se hace usando el teorema

de muestreo de Nyquist que establece: la frecuencia de muestreo usada para convertir la señal continua en una señal discreta debe ser al menos dos veces mayor que la frecuencia más alta presente en la señal [22].

En la práctica, la aplicación de la WT se consigue en ordenadores digitales mediante la aplicación de DWT en muestras discretizadas.

La ecuación de la DWT es [17]:

$$DWT_{\psi}X[m, n] = \frac{1}{\sqrt{a_0^m}} \sum_k x[k] \Psi \left[ \frac{n - ka_0^m}{a_0^m} \right], \quad a = a_0^m \quad b = ka_0^m \quad (2.11)$$

Donde:

- $t$ : Es la wavelet madre.
- $x$ : Es la señal a analizar.
- $a, b$ : Son los parámetros de escalado y traslación.

### 2.2.1.3. El Análisis Multi-Resolución de una señal

El Análisis de Múltiples Resoluciones o Multi-Resolución (MRA), también conocido como un algoritmo piramidal, fue desarrollado por Mallat para descomponer una señal discreta y obtener una representación tiempo-escala. La técnica multi-resolución necesita dos funciones básicas relacionadas: además de la wavelet  $\Psi(t)$ , se requiere otra función denominada *función escalado*  $\varphi(t)$ .

Después de escogida la wavelet madre, la señal a analizar  $x(t)$  pasa a través de dos filtros:

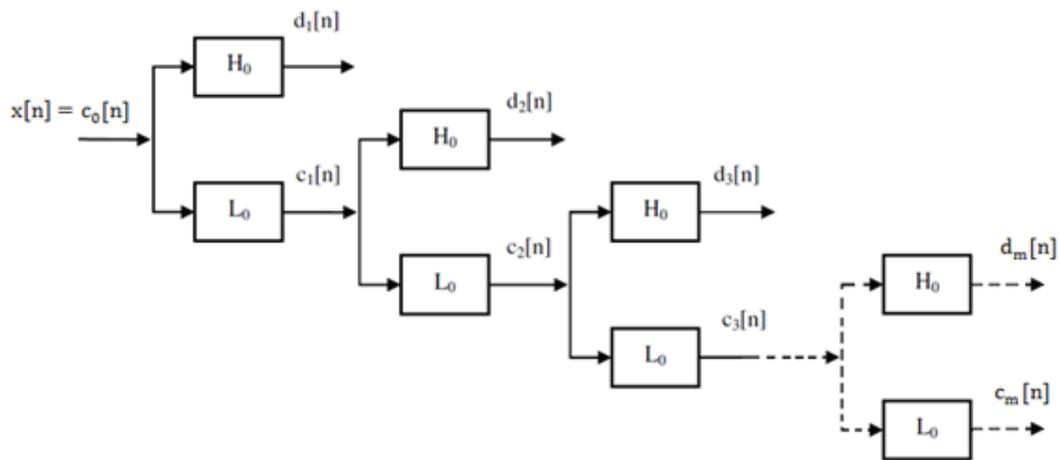
- Pasa bajo: filtro  $L$ , se forma una familia de funciones escalado  $\varphi(t)$ .
- Pasa alto: filtro  $H$ , se forma una familia wavelet  $\Psi(t)$ .

De esta forma, la señal  $x(t)$  puede desarrollarse en series que utilizan las funciones  $\Psi(t)$  y  $\varphi(t)$ :

$$x(t) = \sum_k c_k \varphi_{j_0, k}(t) + \sum_k \sum_{j=j_0}^{\infty} d_{jk} \Psi_{j, k}(t) \quad (2.12)$$

Los pesos de las combinaciones lineales de dichas funciones, que en conjunto representan a una señal  $x(t)$ , se denominan *Coefficientes Wavelet*. Es así que, en la ecuación (2.12), los valores  $c_k$  se denominan *Coefficientes de Aproximación* en la escala  $j_0$ , y los factores  $d_{jk}$  representan los *Coefficientes de Detalle* a diferentes escalas. La resolución de la señal puede modificarse mediante operaciones de filtrado, y la escala se cambia reduciendo la velocidad de muestreo o eliminando ciertas muestras de la señal (submuestreo) [17], [20].

El proceso de filtrado se puede repetir con el fin de descomponer la aproximación obtenida en un nuevo nivel, hasta alcanzar el grado deseado. En la Figura 2.17 se muestra un árbol de descomposición wavelet de  $m$  niveles. La señal de entrada se denota por  $c_0[n]$ , el filtro pasa bajo es  $L_0$ , y el filtro pasa alto es  $H_0$ . En cada nivel, el filtro pasa alto produce información de detalle  $d[n]$  y el filtro pasa bajo, asociado con la función de escalado, produce información de aproximación  $c[n]$ .



**Figura 2.17.** Árbol de descomposición wavelet (nivel  $m$ ) [17].

Sea  $c_0[n]$  la señal de entrada. En la escala 1 dicha señal se descompone en dos señales:  $c_1[n]$  (aproximada) y  $d_1[n]$  (detalle), definidas por:

$$d_1[n] = \sum_k H(k - 2n) x(k) \quad (2.13)$$

$$c_1[n] = \sum_k L(k - 2n) x(k) \quad (2.14)$$

Donde  $L(n)$  y  $H(n)$  son los coeficientes de los filtros pasa bajo y pasa alto.

$c_1$  contiene las componentes de baja frecuencia de  $x[n]$ .

$d_1$  contiene las componentes de alta frecuencia de  $x[n]$ .

En el siguiente nivel de descomposición se utiliza  $c_1[n]$ :

$$d_2[n] = \sum_k H(k - 2n)c_1(k) \quad (2.15)$$

$$c_2[n] = \sum_k L(k - 2n)c_1(k) \quad (2.16)$$

$c_2$  contiene las componentes de baja frecuencia de  $c_1[n]$ .

$d_2$  contiene las componentes de alta frecuencia de  $c_1[n]$ .

Análogamente se pueden construir nuevos niveles de descomposición  $m$ , como se observa en la Figura 2.17.

$$d_m[n] = \sum_k H(k - 2n)c_{m-1}(k) \quad (2.17)$$

$$c_m[n] = \sum_k L(k - 2n)c_{m-1}(k) \quad (2.18)$$

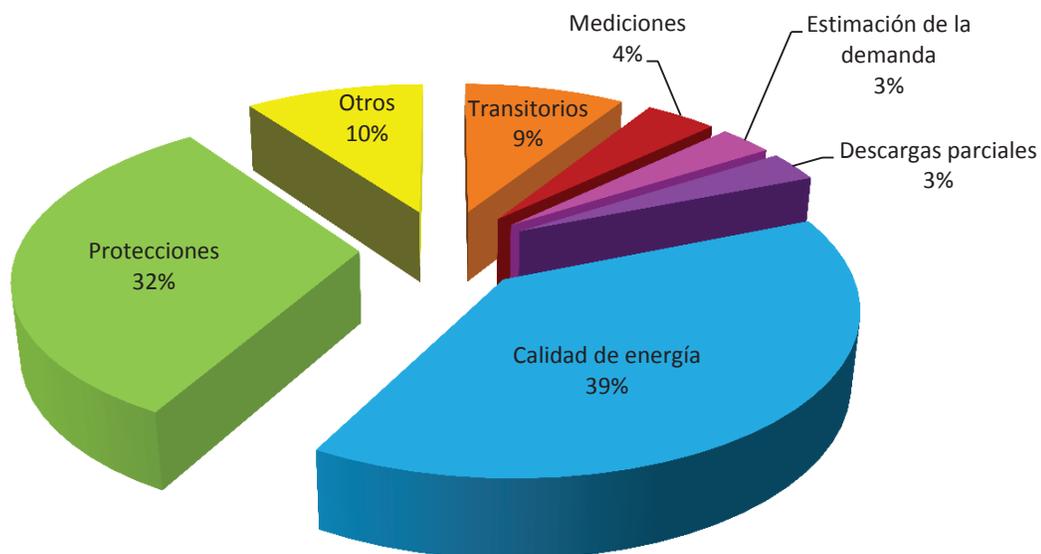
Los filtros eliminan la mitad de las frecuencias y después del filtrado la mitad de las muestras pueden ser eliminadas sin pérdida de información (según Nyquist) puesto que la frecuencia más alta es ahora la mitad del nivel anterior. En cada nivel se obtienen la mitad de muestras que en el nivel precedente, consiguiéndose por tanto el doble de resolución en frecuencia y la mitad de resolución temporal. La salida del filtro pasa bajo se vuelve a filtrar para otra descomposición. La descomposición puede continuar hasta obtener finalmente una muestra en el último nivel. Esta descomposición reduce a la mitad la resolución en el tiempo en cada etapa, como consecuencia de la reducción a la mitad del número de muestras de la señal. Sin embargo duplica la resolución en frecuencia, ya que ahora la banda de frecuencia abarca solamente la mitad de la banda de frecuencia de la etapa anterior. Esta característica es muy relevante, dado que en

la mayoría de las aplicaciones con wavelets, se requiere que la señal original sea sintetizada a partir de los coeficientes wavelet [17].

#### 2.2.1.4. Aplicaciones a SEP

Los investigadores David Robertson y Paulo Ribeiro fueron quienes aplicaron inicialmente esta herramienta a un sistema eléctrico en el año 1994. Desde esta fecha el número de publicaciones relacionadas al análisis y aplicación de wavelets ha ido en constante aumento.

De un total de 196 artículos IEEE, se puede concluir que las áreas de aplicación de wavelets más sobresalientes son el campo de calidad de energía y el campo de protección del sistema de potencia, tal como se muestra en la Figura 2.18.



**Figura 2.18.** Porcentaje de artículos acerca de aplicaciones wavelet en diferentes áreas del sistema de potencia [20].

A continuación se realiza una descripción generalizada de las aplicaciones de wavelet más populares dentro de los sistemas de potencia [20].

- i) Calidad de energía.
- ii) Protección del sistema eléctrico.
- iii) Transitorios del sistema eléctrico.

- iv) Mediciones del sistema de potencia.
- v) Descargas parciales.
- vi) Estimación de la demanda.

#### *2.2.1.4.1. Calidad de energía*

Los estudios dentro de este campo han sido enfocados principalmente en la detección y localización de perturbaciones utilizando la WT como una eficaz y útil herramienta para analizar sags, swells, interrupciones y otros parámetros relacionados con las señales no estacionarias. Las perturbaciones son de cambio lento, las cuales contienen contenido espectral en rangos de baja frecuencia. Por tanto, al examinar los coeficientes de la WT en niveles muy altos de descomposición se puede establecer la ocurrencia y el tiempo en que se producen los eventos de perturbación.

De esta manera y corroborando la información antes mencionada, se puede verificar que la técnica DWT ha sido empleada correctamente en el análisis de eventos de perturbación de los sistemas de potencia.

Las características de la señal original, en teoría, están incluidas en las escalas de la CWT. No obstante, si se tomaran todos los niveles de la CWT sería difícil clasificar en forma precisa los eventos de calidad de energía, puesto que el tiempo y la memoria para que el sistema de reconocimiento pueda alcanzar un nivel adecuado, serían mucho mayores. Además, si sólo se usara el primer nivel de la CWT, algunas características importantes de los otros niveles podrían ser ignoradas. La decisión de cuántos niveles usar depende de cada aplicación.

Por otro lado, con el avance del equipo de monitoreo en calidad de la energía, la cantidad de datos obtenidos durante la última década se ha vuelto enorme, provocando problemas en el almacenamiento y comunicación desde los monitores locales hacia los ordenadores centrales de procesamiento; por lo cual, la compresión de datos se convirtió en un tema esencial dentro del campo de la calidad de energía. Una técnica de compresión de datos, contiene una transformación para extraer las características contenidas en los datos, además de una lógica eliminar las redundancias en las características extraídas.

Aunque la WT presenta una gran capacidad en la detección y localización de los eventos de calidad de energía, su capacidad se ve degradada a menudo en aplicaciones reales debido al ruido, particularmente el ruido blanco con un espectro plano que se monta en la señal. Por lo tanto, para superar las dificultades y capturar las perturbaciones fuera de los ruidos de fondo, se integra un algoritmo de supresión de ruidos con la WT. En los últimos años se ha prestado mucha atención a estos métodos de supresión de ruido con diferentes interpretaciones. Sin embargo, es difícil establecer el umbral en la detección de la existencia de eventos de calidad de energía.

#### *2.2.1.4.2. Protección del sistema eléctrico*

Se han reconocido muchos beneficios al aplicar la WT para mejorar el rendimiento de los relés de protección. En 1996, Chaari introduce la Teoría Wavelet en protecciones de sistemas de distribución de energía para analizar señales transitorias de fallas a tierra, en una red de puesta a tierra de 20 kV generada en el programa EMTP. En el mismo año, otro investigador, Momoh y sus compañeros, presentan un algoritmo para desarrollar un modelo de características adecuado para el entrenamiento de una red neuronal artificial para diagnosticar fallas empleando la Transformada Wavelet, en este caso los datos del caso se obtuvieron de forma experimental.

En 1998, Magnago y Abur desarrollaron una nueva línea de investigación en el área de la localización de fallas empleando las wavelets, para lo cual la onda viajera asociada a la falla generada se procesa con la transformada wavelet para mostrar los tiempos que existen entre el instante en que se produce la falla y la localización de la misma por los relés. Para comprobar y validar el método de localización de fallas propuesto emplean simulaciones en el programa EMTP. En 1999 los mismos autores extienden el método a la identificación de fallas laterales en sistemas radiales de distribución.

El empleo de las wavelets al estudio del disparo monopolar de protecciones se desarrolla para acelerar la apertura de las líneas de transmisión. La transformada wavelet se emplea para analizar los transitorios asociados a la falla generados por un segundo arco y las fallas permanentes; los resultados numéricos revelan que

ciertos componentes de la wavelet pueden ser usados para detectar e identificar de forma eficiente las características relevantes de la falla en los sistemas eléctricos de transmisión, como también para la distinción entre fallas transitorias y permanentes.

La transformada wavelet también se ha aplicado a la protección de barras, de motores, de generadores y de transformadores, en la mayoría de estos casos, el espectro de las señales se analiza con la transformada wavelet para desarrollar un algoritmo para detección de la degradación del aislante, corriente de conexión (inrush) y para una precisa discriminación entre fallas internas y externas [23].

#### *2.2.1.4.3. Transitorios del sistema de potencia*

Las perturbaciones de voltaje que son de menor duración que los sags o swells se clasifican como transitorias y son causadas por cambios repentinos en el sistema de potencia. Sobre la base de la duración, los sobrevoltajes transitorios se pueden dividir en:

- Sobrevoltaje de maniobra (duración en el intervalo de los milisegundos).
- Pico de impulso (duración en el intervalo de los microsegundos).

Los sobrevoltajes son pulsos de alta energía provenientes de perturbaciones en el sistema de potencia, ya sea directamente o como resultado de circuitos de resonancia asociados con dispositivos de conmutación, particularmente de capacitores. Por otro lado, los impulsos resultan de impactos directos o indirectos de rayos, arcos, rupturas del aislamiento, etc.

Una investigación bibliográfica realizada por autores y experiencias pasadas demuestra que para las perturbaciones transitorias cortas y rápidas son mejores las wavelets db4 y db6, mientras que para las perturbaciones transitorias lentas son más adecuadas las wavelets db8 y db10 [20].

A la escala más baja, en el nivel 1, la wavelet madre oscila rápidamente en un periodo muy corto de tiempo. A medida que la wavelet va a una escala más alta, las ondas de análisis se vuelven más localizadas y oscilan menos debido a la naturaleza de dilatación de análisis de la WT. Por lo tanto, las perturbaciones

transitorias rápidas y cortas se detectan a escalas más bajas, mientras que las perturbaciones transitorias lentas y largas serán detectadas a escalas más altas.

#### *2.2.1.4.4. Mediciones del sistema de potencia*

En mediciones de potencia / energía y de valores eficaces (RMS), la WT brinda la ventaja de proporcionar la distribución de potencia y energía con respecto a las bandas de frecuencia individuales asociadas con cada nivel del análisis wavelet.

Hay dos enfoques principales al campo de los armónicos. El primero, lleva a cabo un MRA en una primera etapa, usando bancos de filtros wavelet, y usualmente, la aplicación de la CWT a las sub-bandas en una segunda etapa. El segundo enfoque, utiliza un análisis complejo en su mayoría de la CWT.

Por otro lado, la DWT ha sido parcialmente aplicada en mediciones RMS o de potencia. Sin embargo, cabe mencionar que en el MRA implementado por bancos de filtros DWT para extraer información detallada, la señal es descompuesta en componentes de sub-banda de frecuencia no uniforme en el dominio de tiempo. No obstante, es más útil para propósitos de identificación de armónicos si la señal se descompone en sub-bandas de frecuencia uniforme.

#### *2.2.1.4.5. Descargas parciales*

Las descargas parciales (PDs) son difíciles de detectar debido a sus señales de corta duración, baja amplitud y alta frecuencia. Sin embargo, la WT tiene la capacidad de ampliar señales con discontinuidades en el tiempo, marcando así una ventaja comparándola con la transformada de Fourier, y permite identificar las variaciones locales de la señal. Además, la transformada Wavelet ha sido utilizada en trabajos de detección de descargas parciales en devanados de transformadores, en cables y en subestaciones aisladas con gas (GIS).

Por otro lado, el monitoreo de las descargas parciales ha presentado problemas de interferencia electromagnética. Este ruido se presenta como parte de las PDs de muy bajo nivel detectadas por los sensores. Esto hace que la detección de las descargas parciales sea difícil, particularmente para el monitoreo de PDs de bajo

nivel. Adicionalmente, hay radiofrecuencias relacionadas con la telefonía móvil, entre otras.

#### *2.2.1.4.6. Estimación de la demanda*

La predicción o estimación de la demanda es una clave para la gestión eficiente de los sistemas eléctricos de potencia. Se han desarrollado trabajos para pronosticar la carga eléctrica a corto plazo combinando la WT con redes neuronales artificiales (ANNs). Desde el punto de vista del análisis de señales, puesto que la carga eléctrica se supone usualmente como una combinación lineal de diferentes componentes, ésta puede también considerarse como una combinación lineal de frecuencias diferentes. Cada componente de carga puede ser representado por una o varias frecuencias. A través de la WT, la carga histórica es descompuesta en una parte de aproximación asociada con bajas frecuencias y varias partes de detalles asociadas con frecuencias altas. Luego, se desarrolla el pronóstico de la parte de la carga futura mediante una red neural, o se ajusta la carga usando un método de regresión.

Por otra parte, la integración de energía eólica en la red eléctrica de varios países plantea nuevos desafíos debido a su intermitencia y volatilidad. El pronóstico de la energía eólica desempeña un papel clave en la solución de estos desafíos.

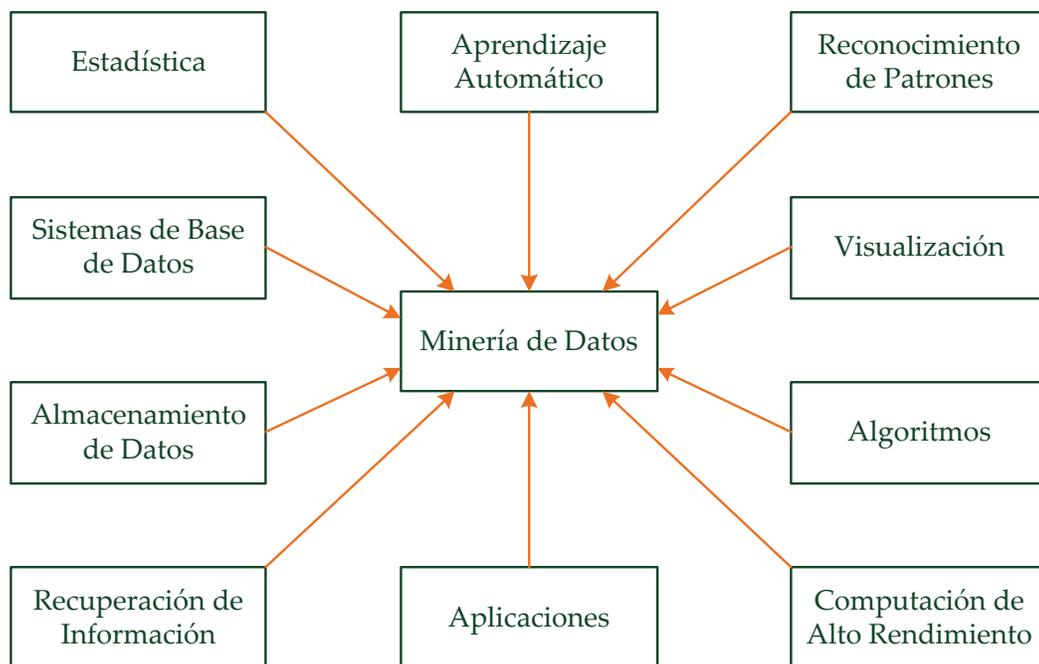
Por otro lado, el pronóstico de precios en un mercado de electricidad real se vuelve esencial para la gestión de riesgos en mercados de electricidad no regulados. Las series de precios son altamente volátiles y de naturaleza no estacionaria. En este sentir, para el tratamiento series no estacionarias se puede considerar las wavelets de Daubechies. Para estas familias de wavelets, la regularidad aumenta conforme aumenta también el orden de las funciones. La regularidad es útil para conocer la suavidad de la señal reconstruida. Sin embargo, conforme el orden se incrementa, los intervalos de soporte también lo hacen, lo cual podría conducir a un análisis local deficiente, provocando que la estimación se deteriore. Por tanto, son generalmente aconsejables las funciones wavelet de orden inferior. La wavelet de Daubechies de orden 1 (db1) es la única wavelet de esta familia que es de naturaleza discontinua, por lo que podría no ser adecuada para análisis de señales de carga, viento o precio. No obstante, según lo

manifiestan varias investigaciones, la wavelet madre Db4 ha sido utilizada en la mayoría de aplicaciones de estimación en sistemas de potencia.

### 2.2.2. MINERÍA DE DATOS

Antes de iniciar con la descripción de las Máquinas de Soporte Vectorial, se presenta una breve introducción que identifica a esta herramienta dentro de los conceptos globales de la Minería de Datos.

La Minería de Datos<sup>6</sup> (DM) es de naturaleza interdisciplinaria en cuanto a investigación y desarrollo, lo cual ha permitido incorporar muchas técnicas de otros dominios que contribuyen significativamente al éxito de la DM, y a sus extensas aplicaciones. En la Figura 2.19 se muestra que la Minería de Datos adopta varias disciplinas, tales como: Estadística, Aprendizaje Automático, Reconocimiento de Patrones, Recuperación de Información, Visualización, Algoritmos, entre otras [24].



**Figura 2.19.** La minería de datos adopta técnicas de diferentes dominios [24].

<sup>6</sup> Minería de Datos: Traducción del término *Data Mining*. Existen otras expresiones que tienen un significado conveniente; por ejemplo: Extracción de conocimiento a partir de datos, Extracción de conocimiento, Análisis de datos / patrones, Arqueología de datos, Exploración de datos, etc. [24]

### 2.2.2.1. Aprendizaje Automático

El Aprendizaje Automático<sup>7</sup> investiga cómo las computadoras pueden aprender (o mejorar su rendimiento) basándose en la información de datos. El área de investigación principal es que las computadoras aprendan automáticamente a reconocer patrones complejos y tomar decisiones inteligentes basadas en los datos, es decir, que actúen sin haber sido explícitamente programadas. En la última década, el aprendizaje automático ha proporcionado vehículos que funcionan sin conductor, reconocimiento práctico de voz, búsqueda efectiva en la web, etc. Actualmente, el uso del aprendizaje automático está tan generalizado que las personas lo usan docenas de veces al día sin darse cuenta [25].

El aprendizaje consiste en la estimación de los parámetros o las relaciones entre los valores de los atributos, que permiten que el clasificador se adapte a un problema específico. Un clasificador puede aprender de dos formas principalmente: Aprendizaje Supervisado y Aprendizaje No Supervisado. No obstante, en los métodos de aprendizaje se puede añadir un tercer algoritmo denominado Semi-Supervisado.

El *Aprendizaje Supervisado* es básicamente un sinónimo de clasificación. La supervisión proviene de una base de datos formada por un conjunto de ejemplos etiquetados  $T = \{(x_1, y_1), \dots, (x_r, y_r)\}$ , que se utilizará en el entrenamiento. Esto quiere decir que, además de contar con los atributos de cada objeto ( $X$ ), se conoce plenamente a qué clase ( $Y$ ) corresponde cada uno.

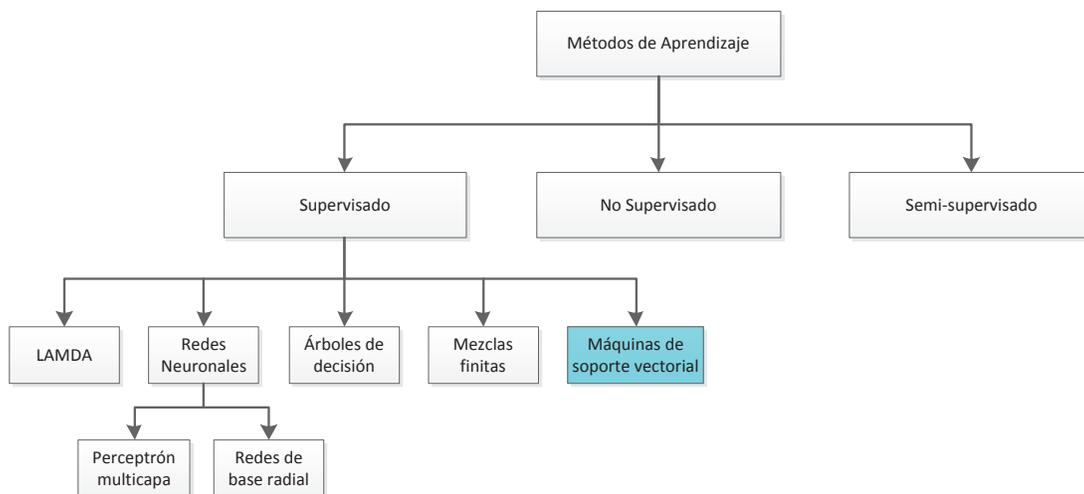
El *Aprendizaje No Supervisado* es esencialmente un sinónimo de agrupamiento (*clustering*). Se utiliza cuando existe una base de datos de la cual solamente se conoce el valor de los atributos de cada objeto ( $X$ ), es decir que los ejemplos de entrada no están etiquetados en ninguna clase. Entonces el conjunto de entrenamiento se puede representar por  $T = \{x_1, \dots, x_r\}$ . Normalmente, se usa el *clustering* para establecer una relación entradas-salidas, y descubrir las clases dentro de los datos. Estas clases deben estar relacionadas con características significativas del problema analizado.

---

<sup>7</sup> Aprendizaje Automático: Traducción del vocablo inglés *Machine Learning*. Se traduce también como Aprendizaje de Máquinas o Métodos de Aprendizaje.

El *Aprendizaje Semi-Supervisado* combina las dos técnicas anteriores, haciendo uso de ejemplos etiquetados y no etiquetados cuando se aprende un modelo. Por un lado, se utilizan ejemplos etiquetados para aprender modelos de clase; y por otro, en el proceso de agrupamiento se usan ejemplos no etiquetados para refinar o ajustar los límites entre dichas clases [24], [26].

La Figura 2.20 muestra los algoritmos más representativos del Aprendizaje Supervisado, resaltando el método de las Máquinas de Soporte Vectorial, el cual será estudiado en la siguiente sección. En el Anexo 1 se presenta un cuadro completo, con los diferentes métodos de aprendizaje.



**Figura 2.20.** Principales métodos del Aprendizaje Supervisado [26].

#### 2.2.2.2. Máquinas de Soporte Vectorial

Las Máquinas de Soporte Vectorial o simplemente SVMs, son un potente algoritmo de aprendizaje supervisado para resolver una variedad de problemas de clasificación/aprendizaje y estimación de funciones, tales como reconocimiento de patrones (facial, voz, diferentes imágenes), regresiones, bioinformática, inversión de operador, predicción de estructura de proteínas, etc. Este método puede clasificar datos lineales y no lineales.

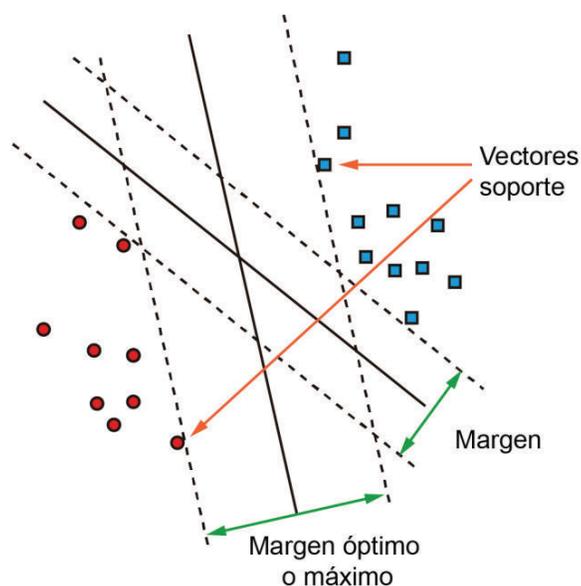
El primer artículo sobre SVMs fue presentado en 1992 por Vladimir Vapnik y sus colegas Bernhard Boser e Isabelle Guyon, usando herramientas estadísticas, de optimización y de teoría de aprendizaje. Sin embargo, la base para SVMs ha

existido desde los años 60, incluyendo los primeros trabajos de Vapnik y Alexei Chervonenkis sobre la teoría del aprendizaje estadístico.

La utilidad de las SVMs se refleja en características como la ausencia de mínimos locales, la alta capacidad de generalización, el control y uso de funciones de núcleo o *kernel*, que a diferencia de otros métodos de clasificación, confieren la ventaja de evitar requerimientos sobre la densidad de probabilidad de los rasgos, y son convenientes en cuanto a la dimensionalidad del problema.

Aunque el tiempo de entrenamiento de las SVMs puede ser extremadamente lento, son muy precisos, debido a su capacidad de modelar límites de decisión no lineales muy complejos. Además, son mucho menos propensos al sobreajuste (*overfitting*) que otros métodos. Las SVMs pueden usarse tanto para la predicción numérica como para la clasificación [24], [26].

De manera simplificada, la metodología que utilizan las SVMs se basa en un mapeo no lineal que transforma los datos originales de entrenamiento en una dimensión superior. Dentro de esta nueva dimensión, las SVMs buscan un hiperplano lineal de separación ("límite de decisión" que divide una clase de otra) que equidiste de los ejemplos más cercanos de cada clase, con el fin de conseguir un margen máximo a cada lado del hiperplano (Ver Figura 2.21). Esta característica permite desarrollar una buena capacidad de generalización.



**Figura 2.21.** Búsqueda del margen óptimo en SVMs [27].

Si los ejemplos de entrada son separables, las SVMs buscan hiperplanos en el mismo espacio original, mientras que si los ejemplos no son linealmente separables, el hiperplano se determina en un espacio transformado (espacio de características), normalmente de muy alta dimensión.

En términos sencillos, las SVMs realizan implícitamente la búsqueda del hiperplano óptimo, utilizando el producto punto con funciones del espacio de características que son llamadas *kernel*. La solución de este hiperplano puede ser escrita como la combinación de unos pocos puntos de entrada de cada clase, que caen justo en la frontera de los márgenes, como se puede apreciar en la Figura 2.21. Estos ejemplos reciben el nombre de *vectores soporte*. La tarea de maximizar el margen es un problema de programación cuadrática y puede ser resuelto por su dual introduciendo multiplicadores de Lagrange [27], [28].

La teoría, tanto como las aplicaciones de las SVMs, constituye un campo muy extenso, que no es el propósito de este trabajo. En esta sección solo se abarcará un extracto del amplio mundo de las Máquinas de Soporte Vectorial, cuyo sustento teórico se encuentra en las referencias [24], [26] y [28].

#### 2.2.2.2.1. Análisis del caso linealmente separable

Para explicar este caso, se analizará el ejemplo más simple: un problema de dos clases que son linealmente separables. Dado un conjunto de ejemplos  $S = \{(\mathbf{x}_1, y_1); \dots; (\mathbf{x}_n, y_n)\}$ , donde  $\mathbf{x}_i \in \mathcal{R}^d$  es el conjunto de datos de entrenamiento asociados a etiquetas de clase  $y_i$ . Cada  $y_i$  puede tomar uno de dos valores: +1 o -1, entonces  $y_i \in \{+1, -1\}$ .<sup>8</sup>

Un hiperplano de separación se define por:

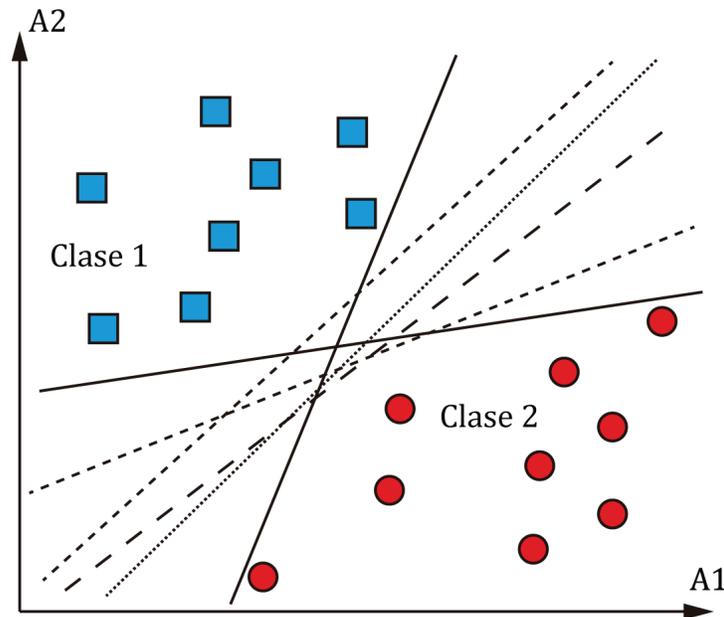
$$D(\mathbf{x}) = (w_1x_1 + \dots + w_dx_d) + b = \mathbf{w} \cdot \mathbf{x} + b = 0 \quad (2.19)$$

Donde:

- $\mathbf{w}$  Es un vector de peso, es decir,  $\mathbf{w} = \{w_1, w_2, \dots, w_d\}$ .
- $d$  Es el número de atributos.
- $b$  Es un escalar, a menudo referido como un sesgo.

<sup>8</sup> De acuerdo a la notación utilizada en este texto, un símbolo que represente un vector se muestra en letra cursiva negrita y minúscula.

Para una mejor visualización, se toma como ejemplo un conjunto de datos que contiene dos atributos de entrada:  $A_1$  y  $A_2$ , como se indica en la Figura 2.22.



**Figura 2.22.** Conjunto de elementos pertenecientes a dos clases. Cada ejemplo está determinado por dos atributos  $A_1$  y  $A_2$  ( $R^2$ ) [28].

Los elementos de entrenamiento están definidos como:  $x = (x_1, x_2)$ , donde  $x_1$  y  $x_2$  son los valores de los atributos  $A_1$  y  $A_2$ , respectivamente. Si se piensa en  $b$  como un peso adicional,  $w_0$ , la ecuación (2.19) se puede reescribir como:

$$w_0 + w_1x_1 + w_2x_2 = 0 \quad (2.20)$$

De la Figura 2.22 se puede deducir fácilmente que existen infinitos hiperplanos que separan los datos de entrada en la Clase 1 y Clase 2. Es evidente la necesidad de establecer un criterio adicional para definir un hiperplano de separación óptimo. Para ello, primero se define el concepto de *margen* de un hiperplano de separación, denotado por  $T$ , como la distancia entre dicho hiperplano y el ejemplo más cercano de cualquiera de las dos clases. A partir de esta definición, un hiperplano de separación se denominará óptimo si su margen es de tamaño máximo y equidista del ejemplo más cercano de cada clase.

Para maximizar el margen, se proponen dos planos paralelos  $H_1$  y  $H_2$  que contienen los puntos más cercanos al hiperplano óptimo. De la ecuación (2.20),

se puede afirmar que cualquier punto que se encuentre por encima del hiperplano de separación satisface la ecuación (2.21), y cualquier punto por debajo del mismo hiperplano cumple con la ecuación (2.22).

$$w_0 + w_1x_1 + w_2x_2 > 0 \quad (2.21)$$

$$w_0 + w_1x_1 + w_2x_2 < 0 \quad (2.22)$$

Entonces, los pesos se pueden ajustar para que los hiperplanos  $H_1$  y  $H_2$  (ver Figura 2.23) que definen cada "lado" del margen, puedan escribirse como sigue:

$$H_1: D(x) = w_0 + w_1x_1 + w_2x_2 \geq 1 \quad \text{para } y_i = +1 \quad (2.23)$$

$$H_2: D(x) = w_0 + w_1x_1 + w_2x_2 \leq -1 \quad \text{para } y_i = -1 \quad (2.24)$$

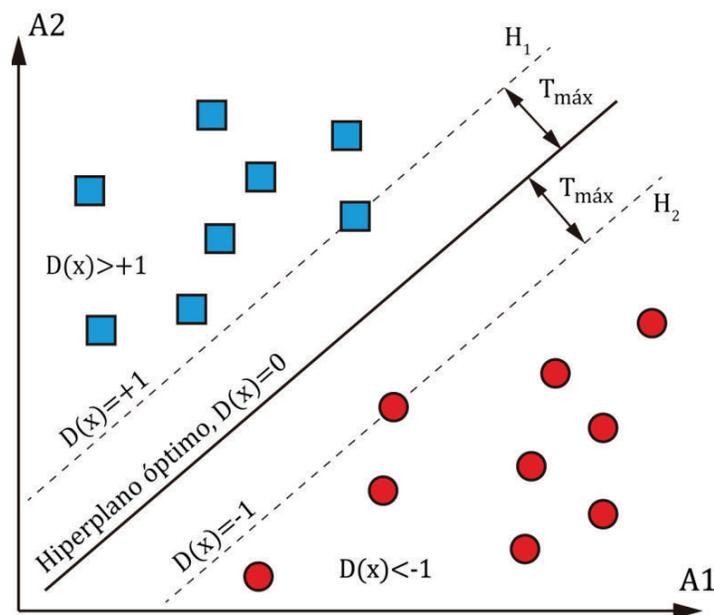


Figura 2.23. Hiperplanos  $H_1$  y  $H_2$  [28].

Cualquier elemento que cae por encima o en  $H_1$  pertenece a la clase +1, y cualquier elemento que cae por debajo o en  $H_2$  pertenece a la clase -1. Aquellos que caen sobre los hiperplanos  $H_1$  y  $H_2$ , son los *vectores soporte* (ver Figura 2.24). Combinando las dos desigualdades (2.23) y (2.24), se obtiene:

$$y_i(w_0 + w_1x_1 + w_2x_2) \geq 1, \quad \forall i \quad (2.25)$$

O de forma general:

$$y_i(\mathbf{w} \cdot \mathbf{x} + b) \geq 1, \quad \forall i \quad (2.26)$$

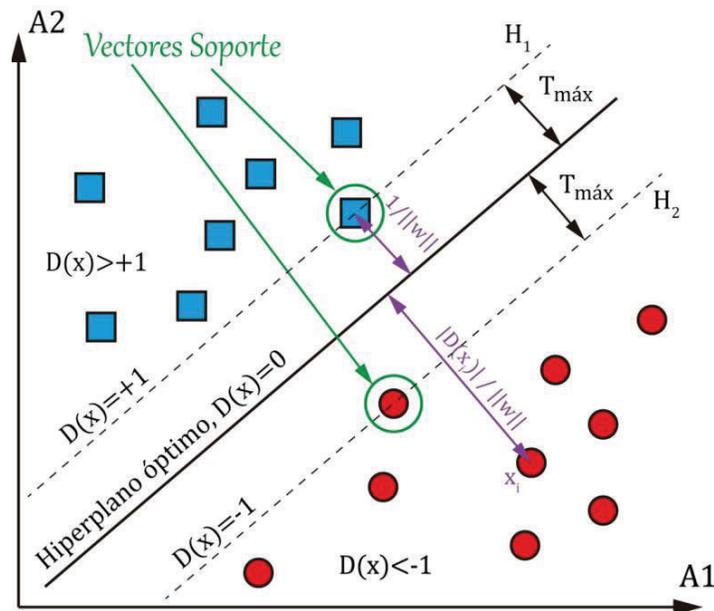


Figura 2.24. Ejemplo de Vectores Soporte [28].

La distancia desde el hiperplano de separación a cualquier ejemplo  $x_i$  es  $\frac{|D(x_i)|}{\|\mathbf{w}\|}$ , donde  $\|\mathbf{w}\|$  es la norma euclidiana de  $\mathbf{w}$ , es decir,  $\sqrt{\mathbf{w} \cdot \mathbf{w}}$ . En particular, si dicho ejemplo pertenece al conjunto de *vectores soporte*, la distancia al hiperplano será siempre  $\frac{1}{\|\mathbf{w}\|}$ , ya que cumplen con  $|D(x)| = 1$ . Por lo tanto, el margen definido como la distancia perpendicular entre  $H_1$  y  $H_2$  está dado por  $\frac{2}{\|\mathbf{w}\|}$ , valor que debe ser maximizado para encontrar el *margen de separación óptimo*, teniendo en cuenta la restricción (2.26). Esto es equivalente a resolver el siguiente problema planteado:

$$\min_{\mathbf{w}, b} \frac{1}{2} (\mathbf{w} \cdot \mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 \quad (2.27)$$

$$\text{sujeto a: } y_i(\mathbf{w} \cdot \mathbf{x} + b) \geq 1, \quad \forall i \quad (2.28)$$

La función (2.27) se llama función objetivo, y junto con (2.28) se conforma un problema de optimización cuadrático con restricciones. Pese a que las herramientas matemáticas utilizadas en la resolución del problema, están más allá del alcance de este trabajo, se presenta una síntesis del proceso de solución. El problema descrito es abordable mediante la *teoría de optimización*, la cual establece que un problema de optimización, denominado *primal*, tiene una forma *dual* si la función a optimizar y las restricciones son funciones estrictamente convexas. En estas circunstancias, resolver el problema *dual* permite obtener la solución del problema *primal*.

En este caso, primero se construye un problema de optimización sin restricciones introduciendo el método de multiplicadores de *Lagrange*.

$$L(\mathbf{w}, \alpha, b) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^n \alpha_i [y_i (\mathbf{w} \cdot \mathbf{x} + b) - 1] \quad (2.29)$$

Donde los  $\alpha_i > 0$  son los denominados multiplicadores de Lagrange.

El segundo paso consiste en aplicar las condiciones de *Karush-Kuhn-Tucker* (KKT). Estas son reemplazadas en (2.29), eliminando las variables primarias  $w$  y  $b$ , así se llega al *problema de optimización dual de Wolfe*, el cual se resuelve mediante las siguientes ecuaciones [26]:

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) \quad (2.30)$$

$$\text{sujeto a: } \alpha_i \geq 0 \quad \forall i \quad \text{y} \quad \sum_{i=1}^n \alpha_i y_i = 0 \quad (2.31)$$

Este problema dual puede ser resuelto con métodos de programación cuadrática estándar. Una vez obtenido el vector  $\alpha^*$ , se pueden obtener los parámetros  $\mathbf{w}^*$  en (2.32),  $b^*$  en (2.33), y la solución del problema primal, es decir, la ecuación del *margen de separación óptimo*, para lo cual se sustituye dicha solución en la expresión (2.32) y el resultado se reemplaza en la ecuación (2.19), obteniendo (2.34) como resultado final.

$$\mathbf{w}^* = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i \quad (2.32)$$

$$b^* = y_i - \mathbf{w}^* \cdot \mathbf{x}_i \quad (2.33)$$

$$D(\mathbf{x}) = \sum_{i=1}^n \alpha_i^* y_i (\mathbf{x}_i \cdot \mathbf{x}) + b^* \quad (2.34)$$

#### 2.2.2.2.2. Análisis del caso linealmente no separable

Se considera un conjunto  $S = \{(\mathbf{x}_1, y_1); \dots; (\mathbf{x}_n, y_n)\}$  que es linealmente no separable, debido a la existencia de datos erróneos, ruido, alto traslapamiento de clases en los datos de entrenamiento, entre otros. Para estos casos, la solución presentada en la sección anterior puede contener una gran cantidad de errores y no ser una buena opción.

Se proponen básicamente dos formas para tratar datos con estas características. La primera opción considera un planteamiento diferente para las SVMs, basándose en permitir la posibilidad de ejemplos que violen la restricción (2.28), por la consideración de variables de relajación, como se presenta en la Figura 2.25. La segunda opción consiste en transformar los datos de entrada original en un espacio de mayor dimensión utilizando un mapeo no lineal. Los mapeos no lineales más comunes se mencionarán después. Una vez que los datos se han transformado en el nuevo espacio superior, se busca un hiperplano de separación lineal en el nuevo espacio, y de nuevo se termina con un problema de optimización cuadrática que puede ser resuelto usando la formulación de SVMs lineal. El hiperplano marginal máximo encontrado en el nuevo espacio corresponde a un hiperplano no lineal en el espacio original.

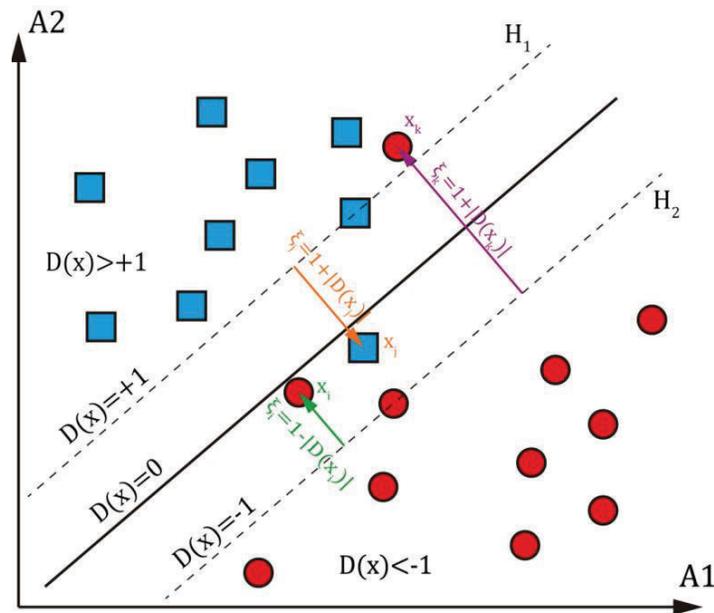
- *Primera opción*

En la Figura 2.25 se tiene que:

$$\xi_i \geq 0, \quad \forall i \quad (2.35)$$

La nueva restricción es:

$$y_i(\mathbf{w} \cdot \mathbf{x} + b) \geq 1 - \xi_i, \quad \forall i \quad (2.36)$$



**Figura 2.25.** Hiperplano de separación para el caso no separable permitiendo infracciones en la clasificación [28].

Luego se encuentra el clasificador que mejor generaliza, controlando su capacidad de clasificación (con  $\|\mathbf{w}\|$ ), y el límite superior del número de errores de entrenamiento ( $\sum_{i=1}^n \xi_i$ ).

Una posible forma de obtener el *hiperplano óptimo de separación con margen blando*<sup>9</sup> es minimizando la función (2.37) sujeta a (2.38) [28].

$$\min_{\mathbf{w}, b} \frac{1}{2}(\mathbf{w} \cdot \mathbf{w}) + C \sum_{i=1}^n \xi_i \quad (2.37)$$

$$\text{sujeto a: } y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \forall i \quad (2.38)$$

Se debe considerar que un valor alto del parámetro C corresponde a una alta penalización a los errores. Con los multiplicadores de Lagrange el problema se transforma en (2.39), sujeto a (2.40).

<sup>9</sup> **Margen blando:** Viene del inglés *soft margin*. Este caso es opuesto al que se analizó en la Sección 2.2.2.1, donde los datos son perfectamente separables y se obtiene un hiperplano óptimo de separación, también conocido como *hiperplano de separación de margen duro*, que viene del inglés *hard margin*.

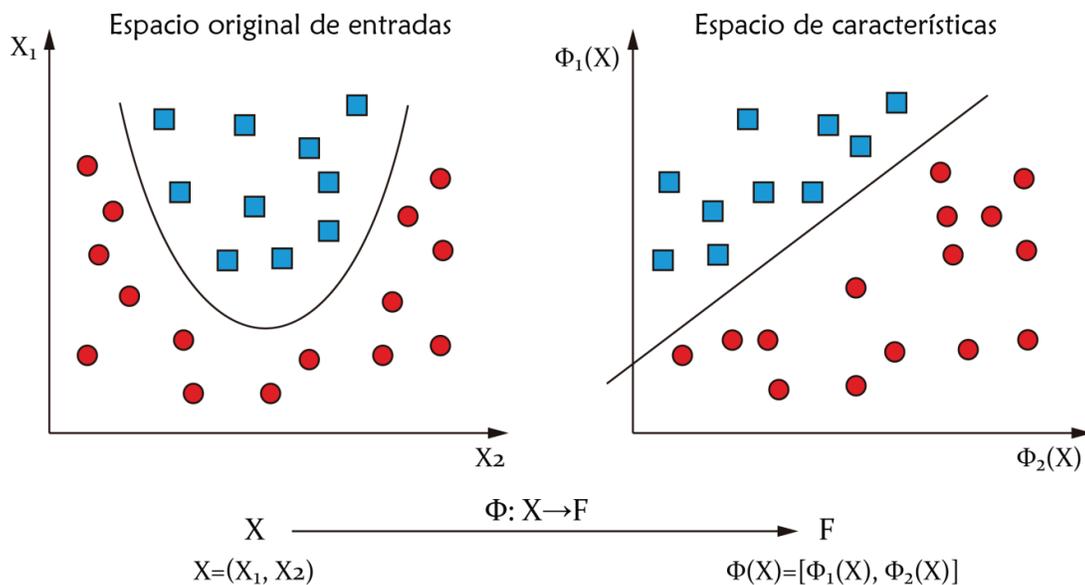
$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (2.39)$$

$$\text{sujeto a: } 0 \leq \alpha_i \leq C, \forall i \quad \text{y} \quad \sum_{i=1}^n \alpha_i y_i = 0 \quad (2.40)$$

En resumen, el hiperplano separador de solución se puede expresar como (2.34).

- *Segunda opción*

El principio de las SVMs no lineales consiste en “mapear” o establecer una relación entre el espacio de entrada y un espacio de representación de dimensión alta, a través de una función no lineal ( $\Phi$ ) elegida a priori, tal como se observa en la Figura 2.26.



**Figura 2.26.** Relación del espacio de entrada con el espacio de características [28].

$\phi$

Por medio de la función  $\Phi$  se trazan los datos de entrada ( $X_i \in R^N$ ) a algún espacio de mayor dimensión y con producto punto definido. Este espacio se llama *espacio característico* ( $F$ ).

$$\Phi: R^N \rightarrow F \quad (2.41)$$

Al resolver el problema de optimización cuadrática de las SVMs lineales (es decir, cuando se busca SVMs lineales en el nuevo espacio de mayor dimensión), los datos de entrenamiento aparecen solo en forma de productos punto, es decir,  $\Phi(x_i) \cdot \Phi(x)$ . En lugar de calcular el producto punto sobre los datos transformados, resulta que es matemáticamente equivalente aplicar una función *kernel* a los datos de entrada originales. Es decir, se define una función que sea el producto punto de los vectores en el espacio característico [24], [26]:

$$k(x_i, x) = \Phi(x_i) \cdot \Phi(x) \quad (2.42)$$

En otras palabras, en todas partes que  $\Phi(x_i) \cdot \Phi(x)$  aparece en el algoritmo de entrenamiento, puede ser reemplazado por  $k(x_i, x)$ . De esta manera, todos los cálculos se realizan en el espacio de entrada original, que es potencialmente de una dimensión mucho menor, y el cálculo del producto punto tiene un menor costo computacional. En realidad, ni siquiera se tiene que saber qué es el mapeo. Después de aplicar este truco del *kernel*, se procede a encontrar un hiperplano de separación óptimo. El procedimiento es similar al descrito anteriormente.

Algunos de los *kernel* admisibles más utilizados se presentan en la Tabla 2.3.

**Tabla 2.3.** Tipos de *kernel* más utilizados [24].

NOMBRE	KERNEL
Polinomial (grado $d$ )	$k(x, y) = (x \cdot y + 1)^d$
Función de base radial (RBF)	$k(x, y) = e^{-\frac{\ x-y\ ^2}{2\sigma^2}}$
Sigmoide	$k(x, y) = \tanh(\kappa(x \cdot y) - \delta)$

La función de base radial (RBF) también es conocida como *kernel* gaussiano, y probablemente es el *kernel* más útil y comúnmente aplicado. No hay reglas absolutas para determinar qué *kernel* resultará ser el más preciso en el entrenamiento. En la práctica, el *kernel* elegido generalmente no hace una diferencia abismal en la precisión resultante [24].

Incluyendo la función *kernel* descrita, se puede reescribir (2.39) sujeta a (2.40), de la siguiente manera [26]:

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \quad (2.43)$$

$$\text{sujeto a: } 0 \leq \alpha_i \leq C, \forall i \quad \text{y} \quad \sum_{i=1}^n \alpha_i y_i = 0 \quad (2.44)$$

Las ecuaciones del margen óptimo de separación (2.34) se reescribe como:

$$D(\mathbf{x}) = \sum_{i=1}^n [\alpha_i^* y_i k(\mathbf{x}_i \cdot \mathbf{x})] + b^* \quad (2.45)$$

#### 2.2.2.2.3. Multiclasificación con SVMs

Hasta ahora, se ha descrito las SVMs lineales y no lineales para la clasificación binaria (clases con valores de  $\pm 1$ ), pero muchos problemas tienen más de dos clases ( $y_i \in \{1, 2, \dots, l\}$ ,  $l > 2$ ). En este caso, las SVMs binarias pueden combinarse para resolver el problema de multiclasificación, para lo cual se admiten dos tipos de arquitectura:

- i) Máquina multiclasificadora.
- ii) Máquinas biclasificadoras generalizadas.

##### i) Máquina multiclasificadora

Esta máquina construye una función clasificadora global, considerando todas las posibles clases. Se intenta resolver el problema modificando la función objetivo de las SVMs de tal manera que permita calcular una única máquina capaz de trabajar con todas las clases a la vez.

La principal desventaja de esta generalización se encuentra en el problema de optimización cuadrática, donde hay  $n(l-1)$  variables sujetas a  $2n(l-1) + 2l$  restricciones de desigualdad.

Esta desventaja se ha confirmado con experimentos donde se muestra que este tipo de solución es lenta. Además, no hay referencias de técnicas que mejoren la robustez del sistema o de estudios sobre la cota de error [26].

## ii) Máquinas biclasificadoras generalizadas

Este tipo de máquinas construyen una función clasificadora global a partir de un conjunto de funciones biclasificadoras. Existen técnicas de descomposición y reconstrucción que permiten a las SVMs biclasificadoras manejar problemas de multclasificación con mayor simplicidad y/o menor tiempo de respuesta que una SVM generalizada a multclasificación.

### *Arquitecturas de descomposición*

En el esquema de descomposición estándar se construyen  $m$  máquinas biclasificadoras, en paralelo que son entrenadas sobre modificaciones del conjunto de aprendizaje. Para esto se crea una matriz de descomposición, donde los elementos de unas clases son asignados a salidas positivas, los de otras a salidas negativas. Algunas arquitecturas de descomposición son [26]:

- Uno contra el resto.
- Uno contra uno.
- ECOC<sup>10</sup>.

### *Métodos de reconstrucción*

Cada máquina biclasificadora entrenada emite una respuesta en forma numérica  $z^i = g_i(x)$  a una entrada  $x$ . La información más importante en esta respuesta, en principio, se encuentra en el signo que adopta la función de decisión.

Existen esquemas de reconstrucción para las tres arquitecturas de descomposición mencionadas arriba, sin embargo, la forma de reconstrucción más habitual son los *Esquemas de votación*, que posee diferentes variantes. Algunas de ellas son: Votación por unanimidad, Votación por mayoría absoluta, Votación por mayoría simple [26].

---

<sup>10</sup> **ECOC**: Del inglés *Error-Correcting Output Code*, se puede traducir como “Código de salida que corrige errores”. Este método reduce un problema multiclase, a un conjunto de clasificadores binarios (como las SVMs).

## CAPÍTULO 3

### DESCRIPCIÓN DE LA METODOLOGÍA

La metodología hace uso básicamente de dos programas computacionales. El primero es, principalmente, una herramienta de simulación para la industria de la energía eléctrica: ATP/EMTP. El segundo software constituye el lenguaje de programación utilizado en el procesamiento de datos: MATLAB. Este capítulo describe el desarrollo de la metodología propuesta, fundamentada en la aplicación de la Transformada Wavelet y las Máquinas de Soporte Vectorial, para la obtención de un modelo de patrones que cumpla con el objetivo general planeado, al contar con la capacidad de establecer la dirección de una falla ocurrida una línea de transmisión.

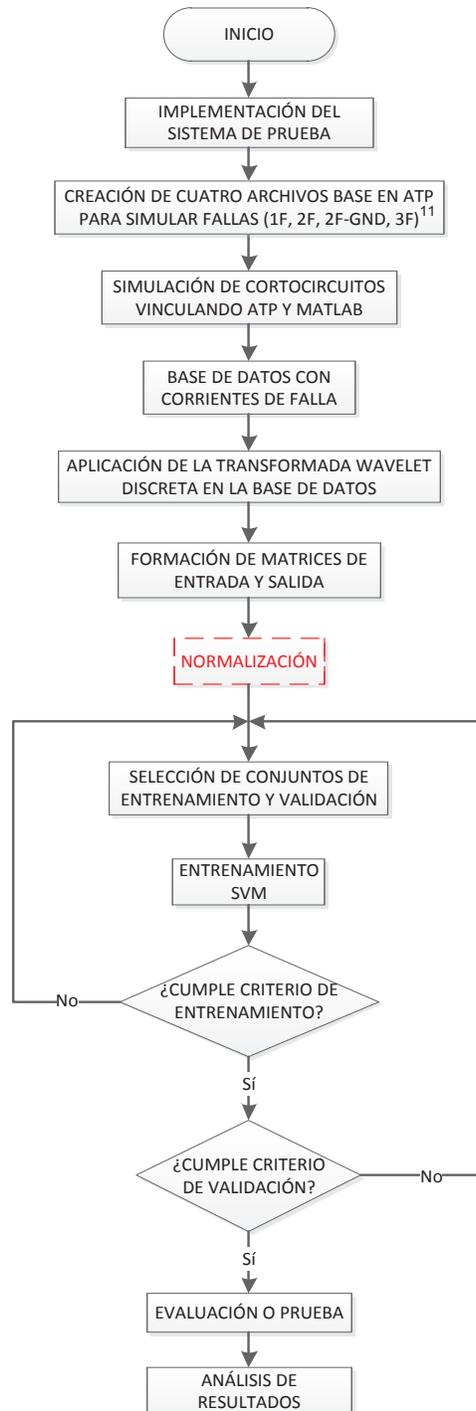
En resumen, una vez modelado el sistema de prueba en ATP / EMTP, se simulan cortocircuitos en la línea protegida por un relé de sobrecorriente, y en la línea anterior al mismo. Se extraen las corrientes de diferentes tipos de falla, y se conforma la base de datos. Luego, se sigue con los procesos de entrenamiento y validación, concluyendo con la obtención y evaluación del modelo deseado.

#### 3.1. DIAGRAMA DE LA METODOLOGÍA

La metodología propuesta involucra una serie de pasos que convergen a la obtención de un modelo apto para predecir correctamente la dirección de ocurrencia de una falla, y cumplir con los objetivos planteados. Esta metodología se expresa en un diagrama de flujo (ver Figura 3.1), cuya composición señala el procedimiento que se siguió para obtener los resultados esperados.

Es interesante notar que la Figura 3.1 incluye un recuadro entrecortado de color rojo con el nombre “*Normalización*” (Sección 3.11), puesto que se trata de un paso recomendable, pero no indispensable en este procedimiento. También es importante resaltar que los pasos subsiguientes a la elaboración de “*Base de datos con corrientes de falla*”, están contenidos en una interfaz gráfica de usuario (GUI) que se desarrolló para una mejor y resumida apreciación del trabajo

realizado. La interfaz mencionada se implementó en MATLAB, y cuenta con un manual de usuario detallado en el Anexo 2.



**Figura 3.1.** Diagrama de flujo de la metodología propuesta [Elaboración propia].

<sup>11</sup> Simbología de fallas simuladas: 1F = monofásicas; 2F = bifásicas; 2F-GND = bifásicas a tierra y 3F = trifásicas.

Las secciones posteriores detallan cada actividad que compone el diagrama de flujo, incluyendo las variables y técnicas utilizadas.

### **3.2. IMPLEMENTACIÓN DEL SISTEMA DE PRUEBA**

El sistema de prueba se implementó en el programa ATP/EMTP (o sencillamente ATP), el cual ofrece algunos beneficios como:

- Simulación precisa (mediante implementación detallada de modelos de elementos de red eléctrica) de transitorios de alta frecuencia que se producen en intervalos de tiempo cortos.
- Facilidad en el manejo de los resultados de simulación, que pueden ser trazados o impresos como funciones de tiempo y almacenados en archivos para procesamiento posterior.

Debido a estas razones, el área de protecciones eléctricas se ha convertido en uno de los mayores consumidores de ATP [29]. Además, el programa ATP es usado en todo el mundo para análisis de sobrevoltajes, coordinación de aislamiento y estudios de oscilación torsional de eje, modelado de relés de protección, estudio de armónicos y de calidad de energía, modelado HVDC y FACTS, entre otros [30].

El sistema de prueba fue implementado en el paquete computacional ATP y consta de varios elementos que se describen a continuación.

#### **3.2.1. GENERADORES**

Un generador es una máquina eléctrica rotativa que transforma energía mecánica en energía eléctrica, gracias a la interacción de sus dos elementos principales: rotor y estator. En la implementación del sistema de potencia, un generador puede representar a una central de generación, ya que son el elemento principal de la misma. Allí se produce energía en corriente alterna sinusoidal a voltajes intermedios, generalmente entre 6 y 13 kV [31].

En este caso, el generador se modeló a través de una fuente trifásica balanceada de AC, llamada “AC source (1&3)”.

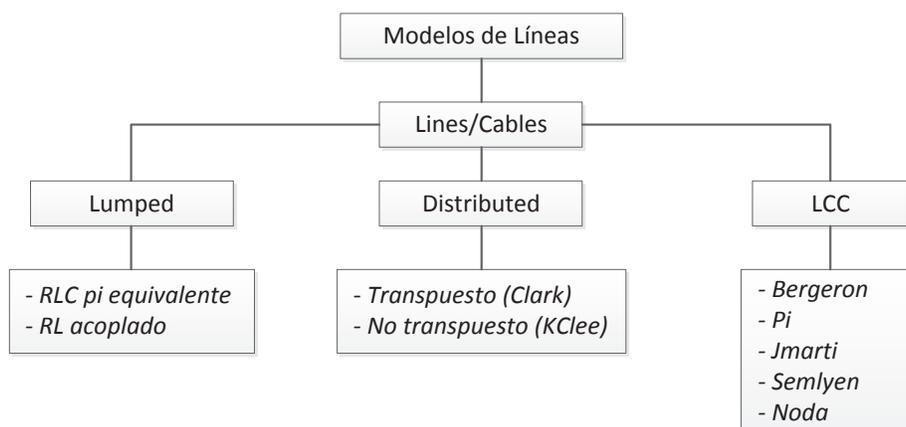
### 3.2.2. TRANSFORMADORES

Los transformadores de potencia son los encargados de aumentar o reducir los niveles de voltaje en el sistema, manteniendo la potencia constante. La implementación de este elemento en el sistema de prueba se realizó mediante el “*Hybrid transformer*” de ATP. Este componente llamado XFMR se agregó a la versión 4.2 de ATPDraw en junio de 2005. El componente XFMR proviene del trabajo realizado por el Profesor Bruce Mork en Michigan Tech. El principio del modelado consiste en derivar un modelo topológicamente correcto con el núcleo conectado a un devanado artificial en la superficie del núcleo [30].

### 3.2.3. LÍNEAS DE TRANSMISIÓN

Las líneas de transmisión están encargadas de enlazar las centrales con los puntos de utilización de energía eléctrica, para lo cual la energía debe ser transportada, frecuentemente a gran distancia de su centro de producción. Las líneas de transmisión están interconectadas entre sí en configuración mallada, de manera que puedan transportar electricidad entre puntos muy alejados, en cualquier sentido. Están generalmente construidas sobre grandes torres metálicas y a voltajes superiores a 66 kV [31].

ATP posee varios modelos de líneas [32] que se indican en la Figura 3.2.



**Figura 3.2.** Modelos de líneas disponibles en ATPDraw [32].

La implementación de las líneas en el sistema de prueba se realizó con el modelo *JMartí*, cuyos parámetros están en función de la frecuencia, y es el más utilizado

para estudio de transitorios de alta frecuencia. Por esta razón, el presente estudio acopla perfectamente a este modelo ya que se manejan primordialmente señales de cortocircuitos que son ondas de carácter transitorio a una alta frecuencia.

### 3.2.3.1. Modelación de Línea de Transmisión para simular fallas

Para facilitar la simulación de numerosas fallas en una línea de transmisión, se utilizó una herramienta presentada en [29] al sistema de prueba. El prototipo se indica en la Figura 3.3.

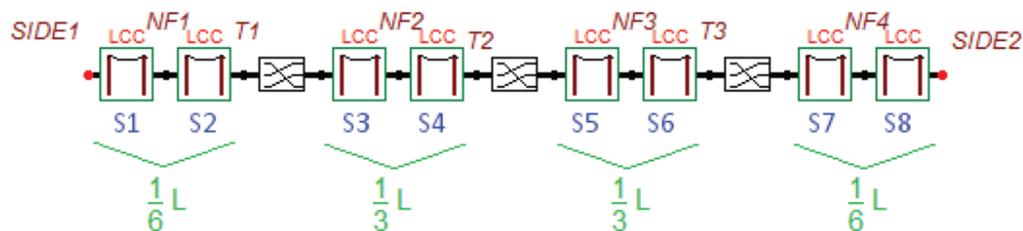


Figura 3.3. Modelo de L/T para simular fallas [29].

Se puede apreciar que la línea de transmisión está dividida en cuatro tramos, debido a las tres transposiciones existentes. Tienen una longitud de  $L/6$ ,  $L/3$ ,  $L/3$  y  $L/6$ , respectivamente, donde  $L$  es la longitud total de la línea.

Cada tramo se subdividió en dos secciones, resultando en cuatro nodos internos: NF1, NF2, NF3 y NF4. La longitud de las secciones S1, S2, S7 y S8 es de  $L/12$ . Las restantes (S3, S4, S5, S6) tienen una longitud de  $L/6$ .

Todas las secciones de línea se implementaron utilizando el modelo LCC que genera un archivo *.lib*, que es leído e incluido por el ATP al ejecutar la simulación. La rutina LCC se incluye en el ATP y calcula las constantes principales de la línea de transmisión desde su geometría y los conductores de línea. En ATP están disponibles varios modelos de LCC: JMarti, Pi, Bergeron, etc. El modelo de JMarti fue elegido para representar la línea de la Figura 3.3, porque es un modelo dependiente de la frecuencia. La geometría y los conductores se consideran iguales a lo largo de la línea de transmisión.

### 3.2.4. CARGAS

Las cargas son componentes eléctricos que consumen determinada potencia. En otras palabras, son consumidores de la energía generada y que luego es transportada por medio de las líneas de transmisión. Una carga también es conocida como la demanda del sistema [31], [33].

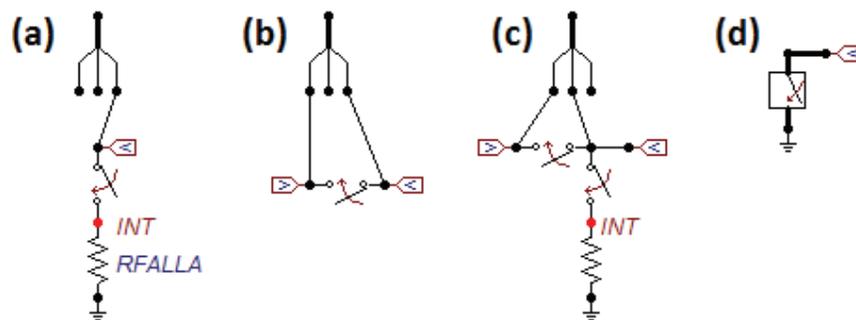
El modelado de las cargas se realizó basándose en los valores de las potencias  $P$  y  $Q$ , que son datos constantes y conocidos.

### 3.3. CREACIÓN DE ARCHIVOS BASE EN ATP

Antes de ejecutar las simulaciones como tal, se requiere primero crear los *archivos base*, que tienen una extensión *.acp* y son los encargados de generar los archivos *.atp*, sobre los cuales trabaja MATLAB con los diferentes cortocircuitos. Se consideraron cuatro tipos de falla en una línea aérea de transmisión, y por tanto es necesario generar cuatro *archivos base*:

1. Falla monofásica.
2. Falla bifásica.
3. Falla bifásica – tierra.
4. Falla trifásica.

Cada archivo base se obtiene insertando en el nodo  $NFi$  de la Figura 3.3, los esquemas ilustrados en la Figura 3.4. De esta manera, cuando se compila el archivo (en ATPDraw), se genera el código ATP para cada tipo de falla, que posteriormente será modificado por MATLAB.



**Figura 3.4.** Esquemas modelados en ATP para: (a) Falla monofásica, (b) Falla bifásica, (c) Falla bifásica-tierra, (d) Falla trifásica [29].

### 3.4. SIMULACIÓN DE CORTOCIRCUITOS VINCULANDO ATP Y MATLAB

Al seleccionar un modelo LCC para la línea de transmisión en la que se simulará la perturbación, se habilitan dos subrutinas adicionales en la función “*runatpmatlab*”<sup>12</sup>. Estas son *generate\_line* y *write\_lib*. Particularmente, los modelos *Jmarti*, al ejecutar con ATP el archivo *.dat*, generan un archivo *.pch*. Con el archivo *.pch*, la subrutina llamada *generate\_line* se encarga de crear un archivo *.lib* perteneciente a una línea de transmisión LCC de longitud *x*. En este caso, *x* es la variable que va cambiando el punto de perturbación. La subrutina *write\_lib* se utiliza para escribir en disco todas las variables que contienen la información de las secciones de línea que se observan en la Figura 3.3: *S1*, *S2*, *S3*, *S4*, *S5*, *S6*, *S7*, *S8*. Los nombres de archivo *.lib* que se han dado a estas 8 secciones en todos los casos base son: *S1.lib*, *S2.lib*, *S3.lib*, *S4.lib*, *S5.lib*, *S6.lib*, *S7.lib* y *S8.lib*.

El programa se complementa con una subrutina llamada *save\_file*, que tiene como objetivo guardar en disco los cambios en los archivos *.dat* y *.lib* generados por las subrutinas *generate\_line* y *write\_lib*, y también escribir en disco el archivo *name\_case.atp* antes de iniciar cada simulación.

Aprovechando que cada tramo está dividido en dos secciones, las fallas se sitúan entre ellas. Así las longitudes de sección se ajustan consecutivamente y la longitud total de la línea y de los tramos se mantiene constante. El archivo base *.atp* se mantiene invariable, sin embargo, lee al disco los archivos *.lib* que son manipulados por el programa para mover el punto de falla. Por lo tanto, cada vez que se ejecuta un caso de simulación, se generan 8 nuevos archivos *.lib*, correspondientes a las 8 secciones de línea, mismos que son leídos por el archivo base *.atp* [29].

#### 3.4.1. ENTRADAS DE LA FUNCIÓN DE SIMULACIÓN “*runatpmatlab*”

La simulación comienza en la plataforma MATLAB con la ejecución de la función de simulación “*runatpmatlab*”. Se debe ingresar cinco parámetros en esta función,

---

<sup>12</sup> “*runatpmatlab*”: Es la función de simulación desarrollada en MATLAB. (Más detalles en [29]).

que se especifican conforme al tipo de falla. A continuación, se explica cada argumento de la sentencia:

```
[verifica] = runatpmatlab(fault_type,Rrfault,phi,x0,step_x)
```

1. *fault\_type*: Tipo de falla seleccionada

El primer argumento es un identificador del tipo de falla, y está asociado con el archivo base *.atp* creado previamente en ATPDraw. Los valores que puede tomar este argumento son:

- 1 = para falla monofásica.
- 2 = para falla bifásica.
- 3 = para falla bifásica-tierra.
- 4 = para falla trifásica.

2. *Rrfault*: Valor de la resistencia de falla en ohm

3. *phi*: Ángulo de inicio de la falla en grados

Este parámetro indica que la falla ocurrirá *phi grados* después del tiempo de referencia *t0* (cruce por cero).

4. *x0*: Valor inicial de la primera falla en km

El cuarto parámetro indica el kilómetro donde se produce la primera falla dentro de la línea de transmisión. El inicio de la línea es el *Lado 1*<sup>13</sup>.

5. *step\_x*: Longitud de desplazamiento en km

El quinto parámetro es la variable *step\_x*, e indica la longitud en km que se desplaza la falla a lo largo de la línea de transmisión.

### 3.4.2. INICIALIZACIÓN DE VARIABLES EN “runatpmatlab”

Antes de iniciar el programa, se deben incluir algunos argumentos dentro de la función de simulación *runatpmatlab*. Los principales argumentos que deben ser definidos son [29]:

---

<sup>13</sup> *Lado 1*: Se considera el lado de la línea de transmisión donde se ubica el relé de protección.

- *directory*: Directorio de salida de la simulación.
- *L*: Longitud total de la línea.
- *t0*: Tiempo de referencia. Es el momento en que la señal de voltaje en el *Lado 1* pasa por cero.

Inicialmente la función *runatpmatlab* calcula ciertas variables en su proceso de ejecución. Por ejemplo, en el Espacio de código 3.1 se puede observar las variables *delay*, *Dtime*, *Dtx*. Una de las más importantes es *x*, que se define con dos argumentos de entrada especificados más atrás (Sección 3.4.1):

- *x*: Distancia de falla medida desde el *Lado 1* de la línea de transmisión. Se inicializa en *x0* y se incrementa en intervalos de *step\_x* hasta llegar a la longitud total de la línea *L*.

```
%-----
directory='C:\EEUG09\ATPDraw\Atp'; %output
L=240; %Total line length
% step_x = Displacement length along the f
% x0 = Initial fault distance
% x = Fault distance measured from side 1
% stretch = stretch of the line (1 2 or 3.
% Lstretch = length of the stretch
% phi = Fault inception angle in degrees
% Rfault = Fault resistance value. Could b
%         diferents values. %!!!! ONLY US
delay=phi/360*(1/60); %is the inception an
                    %constant 0.0167 i
                    %of power frequenc

angS1=0;
angS2=0;

Dtime=(1/60)*(angS1-angS2)/360; %is the d
t0= 0.016098; %Reference time. It is the

Dtx=Dtime/L; %is the diphase angle (Dtime)
```

**Espacio de código 3.1.** Inicialización de variables en la función de simulación *runatpmatlab* [Impresión de pantalla de MATLAB].

### 3.4.3. SALIDAS DE LA FUNCIÓN “runatpmatlab”

Los archivos de salida se guardan en la carpeta designada por la variable *directory* (Sección 3.4.2) que se registra en el script “*runatpmatlab*”. Estos archivos se almacenan como datos propios de MATLAB, es decir, con extensión *.mat*, y toman su nombre conforme a los parámetros de simulación.

En la Tabla 3.1 se presenta la estructura del nombre de los archivos *.mat* de acuerdo a su tipo de falla.

**Tabla 3.1.** Nombres de los archivos \*.mat [Elaboración propia].

<i>Tipo de Falla</i>	<i>Nombre del archivo .mat</i>		
	<i>Texto</i>	<i>Longitud (km)</i>	<i>Resistencia de falla</i>
Monofásica (A, B, C)	faultsinglephd	24	r10
Bifásica (AB, BC, CA)	faultisotwophd	56	r0
Bifásica - Tierra(AB-T, BC-T, CA-T)	faulttwophgrd	17	r10
Trifásica	faultthreephd	108	r0

Por ejemplo, un archivo de nombre *faultsinglephd24r10.mat*, corresponde a una falla monofásica simulada al kilómetro 24, con una resistencia de falla de 10  $\Omega$ .

La función de simulación tiene solo una variable de salida en el *Workspace* de MATLAB, que se denomina *verifica*, dentro de la cual el programa registra los datos necesarios para la revisión del usuario, en otras palabras, sirve como variable de control para comprobar una simulación correcta.

### 3.5. BASE DE DATOS CON CORRIENTES DE FALLA

La elaboración de la base de datos consiste en recopilar todas las señales de corriente, resultado de los diferentes cortocircuitos simulados. Dichas perturbaciones fueron ejecutadas a lo largo de dos líneas de transmisión:

- Línea protegida por el relé.
- Línea anterior al mismo.

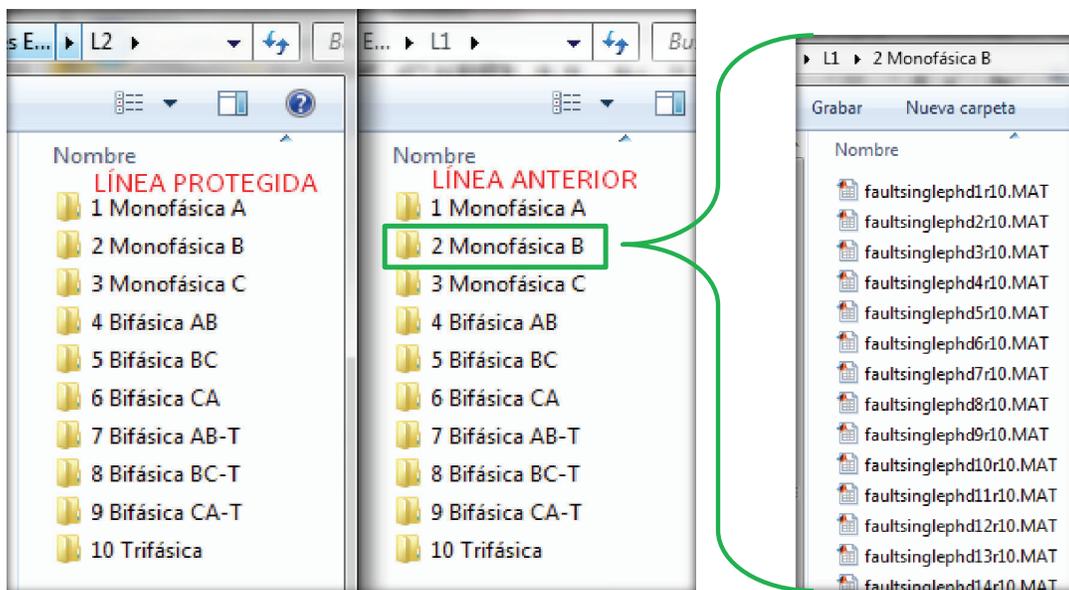
#### 3.5.1. TIPOS DE CORTOCIRCUITOS

En un sistema eléctrico pueden producirse varios tipos de cortocircuitos. En total se simularon diez tipos, y se listan a continuación:

1. Monofásico fase A.
2. Monofásico fase B.
3. Monofásico fase C.
4. Bifásico AB.

5. Bifásico BC.
6. Bifásico CA.
7. Bifásico – Tierra AB.
8. Bifásico – Tierra BC.
9. Bifásico – Tierra CA.
10. Trifásico.

Cada una de estas fallas puede ser simulada a diferentes longitudes de la línea de transmisión. En este caso, se ejecutaron cortocircuitos en cada kilómetro de ambas líneas. Todos los archivos obtenidos fueron organizados en varias carpetas, de acuerdo a su ubicación y tipo de falla. De este modo, se consigue la base de datos mostrada en la Figura 3.5. Dentro de cada carpeta se encuentran los archivos *.mat*, que son resultado de las simulaciones.



**Figura 3.5.** Base de datos con corrientes de falla [Elaboración propia].

### 3.6. APLICACIÓN DE LA TRANSFORMADA WAVELET DISCRETA EN LA BASE DE DATOS

La descripción de todos los comandos de MATLAB aplicados en el presente trabajo, está basada en la documentación de su página oficial: [www.mathworks.com](http://www.mathworks.com) [34].

En esta sección se aplica el comando *wavedec*, mediante el cual se realiza la descomposición wavelet, usando una de sus familias.

### 3.6.1. SELECCIÓN DE DATOS: CORRIENTES DE FALLA

Debido a que las variables recolectadas en los archivos *.mat*, contienen voltajes y corrientes en diversos puntos del sistema de potencia, es necesario extraer únicamente las señales de corriente que detecta el relé. En este punto, se tomaron las variables *[iX4aX0007a]*; *[iX4bX0007b]*; *[iX4cX0007c]*, que equivalen a las corrientes de fase *a*, *b* y *c*, respectivamente.

En el Espacio de código 3.2 se asignan los datos de corrientes a las variables *sa*, *sb* y *sc*, para ser transformadas posteriormente.

```
sa = [iX4aSide1a];
sb = [iX4bSide1b];
sc = [iX4cSide1c];
```

**Espacio de código 3.2.** Corrientes de falla: a, b y c  
[Impresión de pantalla de MATLAB].

El análisis de una enorme cantidad de datos complejos, y su correspondiente trabajo en *Minería* (descubrir patrones en grandes volúmenes de conjuntos de datos) podría tardar mucho tiempo. Se pueden aplicar *técnicas de reducción de datos* para obtener una representación reducida del conjunto original, que es mucho menor en volumen, pero mantiene estrechamente la integridad de los datos. De esta manera, la *minería* en el conjunto de datos reducido, debería ser más eficiente y producir los mismos resultados analíticos (o casi los mismos) que con el conjunto inicial [24].

En el presente estudio, se decidió aplicar la Transformada Wavelet Discreta a las señales de corriente (*sa*, *sb* y *sc*) de cada tipo de cortocircuito. La DWT utiliza un algoritmo jerárquico de pirámide que reduce los datos a la mitad, en cada iteración (ver Sección 2.2.1.3), dando como resultado una velocidad computacional más rápida. Además, la técnica WT también funciona para eliminar el ruido sin excluir las características predominantes de los datos [24].

### 3.6.2. COMANDO *wavedec*

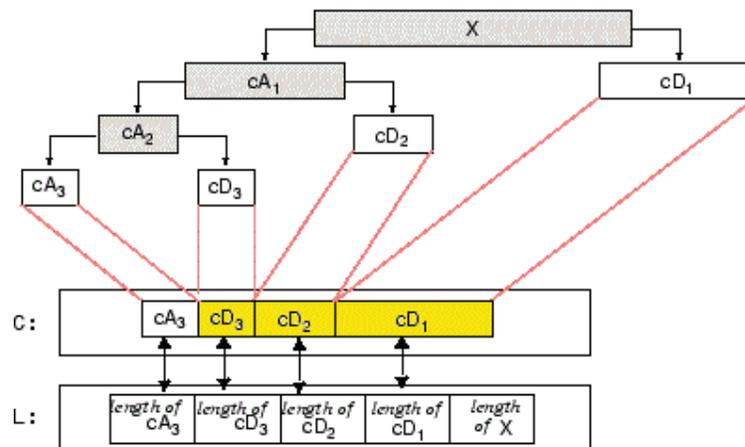
El comando *wavedec* realiza un análisis *wavelet multinivel*, y fue usado en la aplicación de la Transformada Wavelet sobre la base de datos obtenida.

#### 3.6.2.1. Descripción del comando *wavedec*

Se ejecuta a través de la sentencia:  $[C, L] = \text{wavedec}(X, n, 'wname')$ , misma que devuelve la descomposición wavelet de la señal  $X$  a nivel  $n$ , usando una familia de wavelets  $'wname'$ .  $n$  debe ser un entero positivo. La estructura de la salida contiene el vector  $C$  de descomposición wavelet, y el vector  $L$  que devuelve la longitud de cada componente.

Por ejemplo, para una descomposición wavelet de nivel 3, se tiene como salida:

- El vector  $C$ , se compone de los coeficientes de una descomposición de tercer nivel (es decir, el coeficiente de aproximación de tercer nivel y los coeficientes de detalle nivel 1, 2 y 3).
- El vector  $L$ , muestra la longitud de cada componente (Ver Figura 3.6).



**Figura 3.6.** Descomposición Wavelet de nivel 3 [34].

Para extraer los Coeficientes de Detalle, desde el vector  $C$ , se escribe el comando:  $cDn = \text{detcoef}(C, L, n)$ ; de acuerdo al nivel  $n$  que se desee.

### 3.6.3. APLICACIÓN DEL COMANDO *wavedec*

Se decidió utilizar la “*Daubechies 4*” (*D4*) como familia wavelet para la descomposición, ya que la *D4* es una señal más localizada en el tiempo que, por ejemplo, su homóloga *D20*. Por tanto es adecuada para la detección de transitorios rápidos, como los impulsos, cortocircuitos, etc. [21]. Además, se aplicó el nivel 8 en la Transformada Wavelet debido al gran número de elementos que contenían las corrientes de falla, considerando que los Coeficientes de Detalle en cada nivel de la descomposición, son la mitad de la señal precedente, y además la resolución en frecuencia es mejor (Sección 2.2.1.2).

A continuación, se muestra un ejemplo de la descomposición de nivel 8 aplicando la familia wavelet Daubechies 4 ‘db4’ (ver Espacio de código 3.3).

```
[Ca,La] = wavedec(sa,8,'db4');
[Cb,Lb] = wavedec(sb,8,'db4');
[Cc,Lc] = wavedec(sc,8,'db4');
```

**Espacio de código 3.3.** Descomposición Wavelet de nivel 8 [Impresión de pantalla de MATLAB].

Con el Espacio de código 3.4 se extraen los Coeficientes de Detalle de nivel 8; para las corrientes de falla de cada fase.

```
cD8a = detcoef(Ca,La,8);
cD8b = detcoef(Cb,Lb,8);
cD8c = detcoef(Cc,Lc,8);
```

**Espacio de código 3.4.** Coeficientes de Detalle, nivel 8 [Impresión de pantalla de MATLAB].

## 3.7. FORMACIÓN DE MATRICES DE ENTRADA Y SALIDA

Para cumplir con los objetivos planteados en el Capítulo 1 se debe crear un modelo de patrones, capaz de determinar la dirección de una falla en una línea de transmisión, el mismo que a partir de ahora, tomará el nombre de *Modelo A*. Además de cumplir con este propósito, se consideró oportuno añadir un segundo modelo, que en adelante se lo denominará *Modelo B*, cuya función es actuar

únicamente cuando la falla se produce en la línea protegida por el relé, y determinar el tipo de cortocircuito ocurrido.

Como se vio en la Sección 2.2.2.1, el aprendizaje supervisado debe contar con dos grupos de datos para el entrenamiento:

1. Datos de entrada, o atributos de cada objeto.
2. Datos de salida, etiquetas o *respuestas correctas*.

Por facilidad, a estos conjuntos se les dio un nombre específico para su identificación. Al primer grupo se le asignó el nombre de *Matriz de Entrada*; y al segundo, *Vector de Salida*, en vista de sus respectivas dimensiones.

### 3.7.1. MATRICES DE ENTRADA

La matriz de entrada asociada al *Modelo A*, en adelante **Matriz A**, unifica las corrientes de falla en la línea protegida y la línea anterior a ésta, con la finalidad de clasificar objetivamente en qué línea se produjo el cortocircuito; por tanto, la dirección de la falla.

Por su parte, la matriz de entrada del *Modelo B*, denominada **Matriz B**, debe contener solamente las señales de cortocircuitos simulados en la línea protegida, y así determinar el tipo de falla ocurrida.

#### 3.7.1.1. Matriz A

De manera general, esta matriz contiene los Coeficientes de Detalle, nivel  $n$  ( $cDn$ ) de toda la base de datos. Para formar esta matriz, se colocan los  $cDn$  de las tres fases ( $a$ ,  $b$ ,  $c$ ) de la misma falla, en una sola fila. Se consideró conveniente esta distribución de datos debido a que cada individuo o fila de la matriz, representa un tipo de falla, y cuando ocurre un cortocircuito, las corrientes de las tres fases son afectadas de cierta manera, es decir, que el comportamiento de las tres fases influye en la caracterización del tipo de falla.

Luego, se recorre hacia abajo las diferentes longitudes de cada tipo de cortocircuito. El mismo proceso se realiza con los 10 tipos de fallas simuladas en la línea anterior, seguidos de los 10 tipos en la línea protegida por el relé.

En la Tabla 3.2, las celdas sombreadas muestran el esquema completo de la **Matriz A**. Su dimensión es de  $[p \times q]$ , donde  $p$  es el número total de fallas simuladas. En este trabajo, se realiza un mismo número de simulaciones ( $p/20$ ) para cada tipo de falla. En total, son 20 casos ya que en cada línea se producen 10 tipos de cortocircuitos (ver Sección 3.5.1). La variable  $q$  representa el número total de elementos de cada fila. Entonces, el número de elementos de cada fase es el mismo e igual a  $(q/3)$ . Otra forma de leer la **Matriz A**, es decir que cuenta con  $p$  individuos, y cada uno posee  $q$  atributos.

**Tabla 3.2.** Formación de la **Matriz A** [Elaboración propia].

Línea fallada	Tipo de falla	Longitud de falla (km)	Dim. $[p \times q]$	Fase a			Fase b			Fase c		
				1	...	$q/3$	$q/3 + 1$	...	$2q/3$	$2q/3 + 1$	...	$q$
Línea anterior	1 $\phi$ - A	1	1	cDn_a1			cDn_b1			cDn_c1		
		⋮	⋮	cDn_a1			cDn_b1			cDn_c1		
		$p/20$	$p/20$	cDn_a1			cDn_b1			cDn_c1		
	1 $\phi$ - B	1	$p/20 + 1$	cDn_a2			cDn_b2			cDn_c2		
		⋮	⋮	cDn_a2			cDn_b2			cDn_c2		
		$p/20$	$2p/20$	cDn_a2			cDn_b2			cDn_c2		
	1 $\phi$ - C	1	$2p/20 + 1$	cDn_a3			cDn_b3			cDn_c3		
		⋮	⋮	cDn_a3			cDn_b3			cDn_c3		
		$p/20$	$3p/20$	cDn_a3			cDn_b3			cDn_c3		
	2 $\phi$ - AB	1	$3p/20 + 1$	cDn_a4			cDn_b4			cDn_c4		
		⋮	⋮	cDn_a4			cDn_b4			cDn_c4		
		$p/20$	$4p/20$	cDn_a4			cDn_b4			cDn_c4		
	2 $\phi$ - BC	1	$4p/20 + 1$	cDn_a5			cDn_b5			cDn_c5		
		⋮	⋮	cDn_a5			cDn_b5			cDn_c5		
		$p/20$	$5p/20$	cDn_a5			cDn_b5			cDn_c5		
	2 $\phi$ - CA	1	$5p/20 + 1$	cDn_a6			cDn_b6			cDn_c6		
		⋮	⋮	cDn_a6			cDn_b6			cDn_c6		
		$p/20$	$6p/20$	cDn_a6			cDn_b6			cDn_c6		
	2 $\phi$ Gnd - AB	1	$6p/20 + 1$	cDn_a7			cDn_b7			cDn_c7		
		⋮	⋮	cDn_a7			cDn_b7			cDn_c7		
		$p/20$	$7p/20$	cDn_a7			cDn_b7			cDn_c7		
	2 $\phi$ Gnd - BC	1	$7p/20 + 1$	cDn_a8			cDn_b8			cDn_c8		
		⋮	⋮	cDn_a8			cDn_b8			cDn_c8		
		$p/20$	$8p/20$	cDn_a8			cDn_b8			cDn_c8		
	2 $\phi$ Gnd - CA	1	$8p/20 + 1$	cDn_a9			cDn_b9			cDn_c9		
		⋮	⋮	cDn_a9			cDn_b9			cDn_c9		
		$p/20$	$9p/20$	cDn_a9			cDn_b9			cDn_c9		
	3 $\phi$	1	$9p/20 + 1$	cDn_a10			cDn_b10			cDn_c10		
		⋮	⋮	cDn_a10			cDn_b10			cDn_c10		
		$p/20$	$p/2$	cDn_a10			cDn_b10			cDn_c10		

Línea protegida	1 $\phi$ - A	1	$p/2 + 1$	cDn_a11	cDn_b11	cDn_c11
		$\vdots$	$\vdots$	cDn_a11	cDn_b11	cDn_c11
		$p/20$	$11p/20$	cDn_a11	cDn_b11	cDn_c11
	1 $\phi$ - B	1	$11p/20 + 1$	cDn_a12	cDn_b12	cDn_c12
		$\vdots$	$\vdots$	cDn_a12	cDn_b12	cDn_c12
		$p/20$	$12p/20$	cDn_a12	cDn_b12	cDn_c12
	1 $\phi$ - C	1	$12p/20 + 1$	cDn_a13	cDn_b13	cDn_c13
		$\vdots$	$\vdots$	cDn_a13	cDn_b13	cDn_c13
		$p/20$	$13p/20$	cDn_a13	cDn_b13	cDn_c13
	2 $\phi$ - AB	1	$13p/20 + 1$	cDn_a14	cDn_b14	cDn_c14
		$\vdots$	$\vdots$	cDn_a14	cDn_b14	cDn_c14
		$p/20$	$14p/20$	cDn_a14	cDn_b14	cDn_c14
	2 $\phi$ - BC	1	$14p/20 + 1$	cDn_a15	cDn_b15	cDn_c15
		$\vdots$	$\vdots$	cDn_a15	cDn_b15	cDn_c15
		$p/20$	$15p/20$	cDn_a15	cDn_b15	cDn_c15
	2 $\phi$ - CA	1	$15p/20 + 1$	cDn_a16	cDn_b16	cDn_c16
		$\vdots$	$\vdots$	cDn_a16	cDn_b16	cDn_c16
		$p/20$	$16p/20$	cDn_a16	cDn_b16	cDn_c16
	2 $\phi$ Gnd - AB	1	$16p/20 + 1$	cDn_a17	cDn_b17	cDn_c17
		$\vdots$	$\vdots$	cDn_a17	cDn_b17	cDn_c17
$p/20$		$17p/20$	cDn_a17	cDn_b17	cDn_c17	
2 $\phi$ Gnd - BC	1	$17p/20 + 1$	cDn_a18	cDn_b18	cDn_c18	
	$\vdots$	$\vdots$	cDn_a18	cDn_b18	cDn_c18	
	$p/20$	$18p/20$	cDn_a18	cDn_b18	cDn_c18	
2 $\phi$ Gnd - CA	1	$18p/20 + 1$	cDn_a19	cDn_b19	cDn_c19	
	$\vdots$	$\vdots$	cDn_a19	cDn_b19	cDn_c19	
	$p/20$	$19p/20$	cDn_a19	cDn_b19	cDn_c19	
3 $\phi$	1	$19p/20 + 1$	cDn_a20	cDn_b20	cDn_c20	
	$\vdots$	$\vdots$	cDn_a20	cDn_b20	cDn_c20	
	$p/20$	$p$	cDn_a20	cDn_b20	cDn_c20	

### 3.7.1.2. Matriz B

Al igual que en la Sección 3.7.1.1, la **Matriz B** contiene los Coeficientes de Detalle, nivel  $n$  (cDn), pero sólo de los cortocircuitos simulados en la línea protegida. La ubicación de estos coeficientes sigue el mismo procedimiento de la **Matriz A**, posicionando a los coeficientes de fase  $a$ ,  $b$  y  $c$  del mismo cortocircuito, en una sola fila; y colocándolos verticalmente hasta completar los 10 tipos de falla. Siendo así, la **Matriz B** es prácticamente la mitad inferior de la **Matriz A**.

En la Tabla 3.3, se expone el esquema completo de la **Matriz B**. Su dimensión es de  $[r \times s]$ . Ya que se realizó el mismo número de simulaciones ( $r/10$ ) para cada uno de los 10 tipos de falla, la variable  $r$  es el número total de fallas simuladas. La variable  $s$  representa el número total de elementos de cada fila, considerando que el número de elementos por cada fase ( $a$ ,  $b$ ,  $c$ ) es el mismo e igual a  $s/3$ .

**Tabla 3.3.** Formación de la **Matriz B** [Elaboración propia].

Tipo de falla	Longitud de falla (km)	Dim. [r x s]	Fase a			Fase b			Fase c		
			1	...	s/3	s/3 +1	...	2s/3	2s/3 +1	...	s
1 $\phi$ - A	1	1	cDn_a1			cDn_b1			cDn_c1		
	⋮	⋮	cDn_a1			cDn_b1			cDn_c1		
	r/10	r/10	cDn_a1			cDn_b1			cDn_c1		
1 $\phi$ - B	1	r/10 +1	cDn_a2			cDn_b2			cDn_c2		
	⋮	⋮	cDn_a2			cDn_b2			cDn_c2		
	r/10	2r/10	cDn_a2			cDn_b2			cDn_c2		
1 $\phi$ - C	1	2r/10 +1	cDn_a3			cDn_b3			cDn_c3		
	⋮	⋮	cDn_a3			cDn_b3			cDn_c3		
	r/10	3r/10	cDn_a3			cDn_b3			cDn_c3		
2 $\phi$ - AB	1	3r/10 +1	cDn_a4			cDn_b4			cDn_c4		
	⋮	⋮	cDn_a4			cDn_b4			cDn_c4		
	r/10	4r/10	cDn_a4			cDn_b4			cDn_c4		
2 $\phi$ - BC	1	4r/10 +1	cDn_a5			cDn_b5			cDn_c5		
	⋮	⋮	cDn_a5			cDn_b5			cDn_c5		
	r/10	5r/10	cDn_a5			cDn_b5			cDn_c5		
2 $\phi$ - CA	1	5r/10 +1	cDn_a6			cDn_b6			cDn_c6		
	⋮	⋮	cDn_a6			cDn_b6			cDn_c6		
	r/10	6r/10	cDn_a6			cDn_b6			cDn_c6		
2 $\phi$ Gnd - AB	1	6r/10 +1	cDn_a7			cDn_b7			cDn_c7		
	⋮	⋮	cDn_a7			cDn_b7			cDn_c7		
	r/10	7r/10	cDn_a7			cDn_b7			cDn_c7		
2 $\phi$ Gnd - BC	1	7r/10 +1	cDn_a8			cDn_b8			cDn_c8		
	⋮	⋮	cDn_a8			cDn_b8			cDn_c8		
	r/10	8r/10	cDn_a8			cDn_b8			cDn_c8		
2 $\phi$ Gnd - CA	1	8r/10 +1	cDn_a9			cDn_b9			cDn_c9		
	⋮	⋮	cDn_a9			cDn_b9			cDn_c9		
	r/10	9r/10	cDn_a9			cDn_b9			cDn_c9		
3 $\phi$	1	9r/10 +1	cDn_a10			cDn_b10			cDn_c10		
	⋮	⋮	cDn_a10			cDn_b10			cDn_c10		
	r/10	r	cDn_a10			cDn_b10			cDn_c10		

### 3.7.2. VECTORES DE SALIDA

Contienen las *respuestas correctas* de la **Matriz A** y **Matriz B**. A partir de ahora, estos vectores toman el nombre de **Vector A** y **Vector B**, respectivamente. En la Tabla 3.4 se describe el contenido de los mismos, con sus respectivas dimensiones. También se indica el significado de cada número utilizado en MATLAB, de acuerdo con la dirección y el tipo de falla. Por ejemplo, en el **Vector A**, el número 2 señala una falla en la *Línea protegida*. En el **Vector B**, el número 2 denota una falla *Monofásica B*; el número 8, una falla *Bifásica BC – Tierra*.

**Tabla 3.4.** Detalle de Vectores de Salida **A** y **B** [Elaboración propia].

Vector A			Vector B		
Dim. [ $\rho \times 1$ ]	Falla en:	SIGNIFICADO	Dim. [ $r \times 1$ ]	Tipo de falla:	SIGNIFICADO
1	1	Línea anterior	1	1	Monofásica A
2	1	Línea anterior	⋮	1	Monofásica A
3	1	Línea anterior	$r/10$	1	Monofásica A
⋮	1	Línea anterior	$r/10 + 1$	2	Monofásica B
$p/2 - 2$	1	Línea anterior	⋮	2	Monofásica B
$p/2 - 1$	1	Línea anterior	$2r/10$	2	Monofásica B
$p/2$	1	Línea anterior	$2r/10 + 1$	3	Monofásica C
$p/2 + 1$	2	Línea protegida	⋮	3	Monofásica C
$p/2 + 2$	2	Línea protegida	$3r/10$	3	Monofásica C
$p/2 + 3$	2	Línea protegida	$3r/10 + 1$	4	Bifásica AB
⋮	2	Línea protegida	⋮	4	Bifásica AB
$p - 2$	2	Línea protegida	$4r/10$	4	Bifásica AB
$p - 1$	2	Línea protegida	$4r/10 + 1$	5	Bifásica BC
$p$	2	Línea protegida	⋮	5	Bifásica BC
			$5r/10$	5	Bifásica BC
			$5r/10 + 1$	6	Bifásica CA
			⋮	6	Bifásica CA
			$6r/10$	6	Bifásica CA
			$6r/10 + 1$	7	Bifásica AB - Tierra
			⋮	7	Bifásica AB - Tierra
			$7r/10$	7	Bifásica AB - Tierra
			$7r/10 + 1$	8	Bifásica BC - Tierra
			⋮	8	Bifásica BC - Tierra
			$8r/10$	8	Bifásica BC - Tierra
			$8r/10 + 1$	9	Bifásica CA - Tierra
			⋮	9	Bifásica CA - Tierra
			$9r/10$	9	Bifásica CA - Tierra
			$9r/10 + 1$	10	Trifásica
			⋮	10	Trifásica
			$r$	10	Trifásica

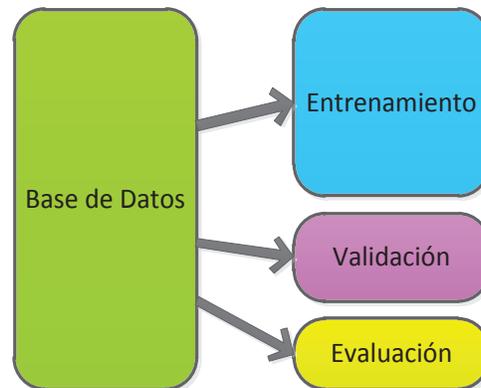
### 3.8. SELECCIÓN DE CONJUNTOS DE ENTRENAMIENTO Y VALIDACIÓN

#### 3.8.1. ENTRENAMIENTO, VALIDACIÓN Y EVALUACIÓN

Hasta el momento se cuenta con un conjunto de muestras, de las cuales se quiere obtener dos modelos que permitan clasificar la dirección de ocurrencia de una falla, y el tipo de cortocircuito producido en una línea de transmisión. A su vez, se desea evaluar el desempeño de dichos modelos.

Comúnmente, cuando se realizan modelos predictivos, se debe contar con tres conjuntos de datos [35], como se indica en la Figura 3.7:

- *Entrenamiento*: Son los datos con los que se entrenan los modelos.
- *Validación*: Selecciona el mejor de los modelos entrenados.
- *Prueba o Evaluación*: Indica lo bien que clasifica el modelo con fallas que no ha visto durante su desarrollo, es decir que entrega el error real del modelo.



**Figura 3.7.** Conjuntos de Entrenamiento, Validación y Evaluación [35].

El conjunto total de muestras puede ser separado en las tres agrupaciones antes mencionadas; sin embargo, para este caso, se desea utilizar solo dos grupos: Entrenamiento y Validación, debido a que la Evaluación se realiza normalmente con señales reales tomadas en campo, o señales de fallas simuladas completamente diferentes a aquellas que ya conforman la base de datos. Además, sería redundante contar con Conjuntos de Validación y Evaluación del mismo tipo. En virtud de ello, se decidió tomar otras muestras, denominadas

*Señales Externas* para la evaluación del modelo. Consiste en un pequeño grupo de fallas producidas a diferentes longitudes de la base de datos ya existente (ver Sección 3.5). En otras palabras, las fallas que componen el grupo *Señales Externas* se generan en longitudes de fracción de kilómetro.

### 3.8.2. PORCENTAJES DE ENTRENAMIENTO Y VALIDACIÓN

La selección de los Conjuntos de Entrenamiento y Validación se realiza en base a porcentajes, cuyos valores típicos son de 60% y 40%, respectivamente. Otros valores también utilizados son 70% y 30%.

En consecuencia, se procedió con la extracción *aleatoria* del 60% del total de datos (matrices de entrada y vectores de salida), conformando así, el Conjunto de Entrenamiento. El 40% restante constituye el Conjunto de Validación.

En el Espacio de código 3.5 se muestra como ejemplo, la separación de los conjuntos para el *Modelo A*. Con este código se asegura que ninguna muestra utilizada en el entrenamiento se vuelva a repetir en la validación.

```
X = cD8_total; % Matriz total de entradas
Y = Y_total; % Matriz total de respuestas

idx60 = randsample(4800,4800*0.6); % Selección aleatoria

X60 = X(idx60,:); % 60% Matriz de entrada
Y60 = Y(idx60); % 60% Matriz de respuestas

X40 = X;
Y40 = Y;

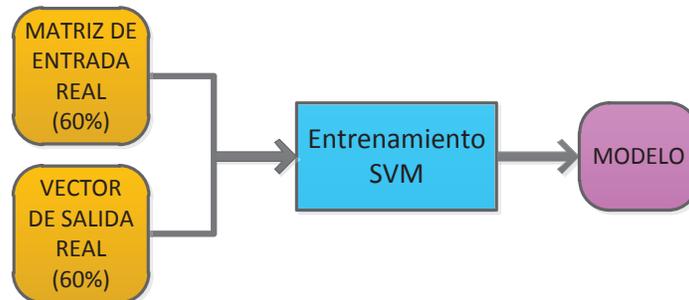
X40(idx60,:)=[]; % 40% restante - Matriz de entrada
Y40(idx60)=[]; % 40% restante - Matriz de respuestas
```

**Espacio de código 3.5.** Conjuntos de Entrenamiento (60%) y Validación (40%)  
[Impresión de pantalla de MATLAB].

## 3.9. ENTRENAMIENTO CON SVMs

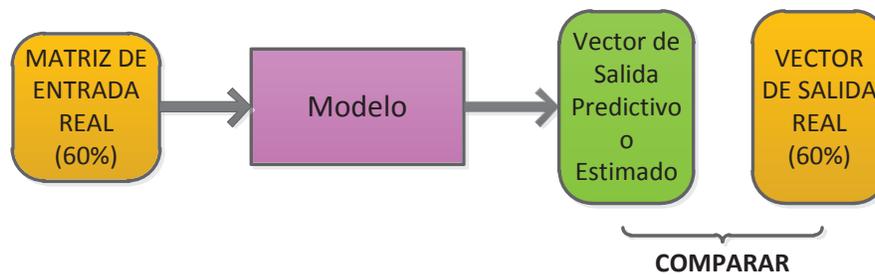
Considerando la Sección 3.7, se deben encontrar dos modelos de patrones, razón por la cual se ejecutaron dos entrenamientos independientes para extraer los Modelos A y B. El procedimiento es el mismo en ambos casos.

Las muestras del Conjunto de Entrenamiento (60%) fueron procesadas con la herramienta SVM, que se describe en la siguiente sección (3.9.1). La respuesta a este proceso es un modelo de patrones, como se puede apreciar en la Figura 3.8.



**Figura 3.8.** Proceso de Entrenamiento [Elaboración propia].

Con el propósito de constatar que el modelo es coherente, se realizó una verificación del mismo, comparando las respuestas correctas (vector de salida), con las respuestas obtenidas a partir del modelo mencionado. En la Figura 3.9 se presenta un gráfico que explica este concepto.



**Figura 3.9.** Comparación de vectores de salida en el Entrenamiento [Elaboración propia].

Se estableció que el criterio para considerar a un modelo como efectivo, es que tenga máximo 2 errores en la predicción de las salidas. Esto con el fin de construir un modelo extremadamente robusto que dé muy poca o ninguna flexibilidad a errores en la operación del relé.

### 3.9.1. COMANDO *fitcecoc*

En general, este comando se utiliza para ajustar modelos multiclase de SVMs u otros clasificadores [34].

### 3.9.1.1. Descripción del comando *fitcecoc*

La sentencia del comando es: `Mdl = fitcecoc (Xtraining, Ytraining)`.

El comando *fitcecoc* utiliza por defecto, modelos binarios de tipo SVMs y un diseño de codificación *uno contra uno*:  $K \times (K - 1)/2$ , donde  $K$  es el número de etiquetas de clase (niveles).

La salida `Mdl` es un modelo multiclase tipo ECOC, entrenado y completo. En el aprendizaje, este comando utiliza los predictores `Xtraining` y las etiquetas de clase `Ytraining`:

- **Xtraining** es una matriz de variables predictoras (matriz de entrada), que contiene específicamente valores numéricos de dimensión  $[N \times P]$ .
- **Ytraining** es el vector respuesta con  $N$  elementos de etiquetas de clase. Este parámetro puede ser una matriz categórica, vector lógico, vector numérico, o una matriz de celdas de cadenas de caracteres.

### 3.9.2. APLICACIÓN DEL COMANDO *fitcecoc*

Una vez establecido el Conjunto de Entrenamiento, se aplica el método SVM con el comando *fitcecoc* en la plataforma de MATLAB. En el Espacio de código 3.6, **X60** es la matriz de entrada y **Y60** es el vector de salida.

```
Mdl = fitcecoc(X60,Y60) %% Realiza el entrenamiento
```

**Espacio de código 3.6.** Entrenamiento del modelo  
[Impresión de pantalla de MATLAB].

La variable *Mdl* es aquella que contiene el clasificador o modelo entrenado. Es decir, que el *Modelo A* contiene los patrones capaces de determinar la dirección de una falla; y el *Modelo B*, los patrones que indican el tipo de falla ocurrida.

Luego, se verifica que el modelo *Mdl* cumpla con el criterio de no predecir erróneamente en más de 2 ocasiones. Para ello, se obtiene el vector de salida estimado **Ye** (con el comando indicado en el Espacio de código 3.7), y se compara con el vector de salida real.

```
Ye = predict(Mdl,X60); % Y estimada
```

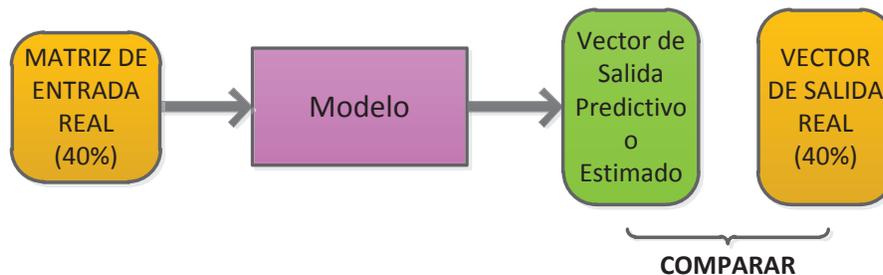
**Espacio de código 3.7.**  $Y_e$  obtenida con el Conjunto de Entrenamiento [Impresión de pantalla de MATLAB].

Sin embargo, no es suficiente verificar que el modelo es exitoso con los mismos datos que fueron entrenados. Se requiere un conjunto diferente, con muestras que no fueran utilizadas durante el entrenamiento. A este conjunto, se le conoce como Conjunto de Validación.

### 3.10. VALIDACIÓN

El Conjunto de Validación es aquel porcentaje de datos que no fueron utilizados para el entrenamiento; en este caso, es el 40%.

El modelo conseguido en la sección anterior, ahora debe pasar por un proceso de validación. Para esto, se toma el mencionado conjunto, y se realiza la predicción de salidas, pasándolo a través del modelo a evaluar, como se presenta en la Figura 3.10. Al comparar los dos vectores de salida, se evalúa el número de errores en los datos estimados.



**Figura 3.10.** Proceso de Validación [Elaboración propia].

Nuevamente, el criterio considerado para la validación es no exceder 2 errores en la predicción; caso contrario, se repite el proceso desde la Sección 3.8.2, referente a la selección aleatoria de los Conjuntos de Entrenamiento y Validación. Con esto se asegura que el modelo final sea el mejor resultado posible, ya que cumple con el criterio establecido en la validación. El comando para encontrar el vector de salida predictivo  $Y_e$ , se indica en el Espacio de código 3.8.

```
Ye = predict(Mdl,X40); % Y estimada (con 40%)
```

**Espacio de código 3.8.** Ye obtenida con el Conjunto de Validación [Impresión de pantalla de MATLAB].

### 3.10.1. COMANDO *plotconfusion*

La comparación entre el *vector de salida estimado* (predicciones realizadas por el modelo de clasificación) y el *vector de salida real*, se ejecutó mediante una herramienta denominada *Matriz de Confusión*, la cual permite observar con claridad el desempeño de un algoritmo mediante porcentajes de acierto y error. La matriz de confusión se emplea en el Aprendizaje Supervisado, dado que se necesita conocer a qué clase pertenecen los objetos que se evalúan.

El comando de MATLAB que devuelve el gráfico de una matriz de confusión es: `plotconfusion`. Esta función exige de dos parámetros de entrada: datos reales y datos estimados.

#### 3.10.1.1. Descripción del comando *plotconfusion*

Este comando se ejecuta a través de: `plotconfusion(targets, outputs)`, donde `targets` representa los datos reales; y `outputs`, los datos estimados.

En el gráfico de una matriz de confusión, las filas corresponden a la clase estimada (*outputs*); y las columnas, a la clase verdadera (*targets*). Las celdas diagonales de la matriz, informan cuántos ejemplos fueron clasificados correctamente por el modelo entrenado, es decir, presentan el porcentaje en que los datos verdaderos coinciden con los predichos. Las celdas fuera de la diagonal señalan en dónde se ubican los errores que cometió el clasificador.

La columna del extremo derecho de la gráfica, contiene los porcentajes de acierto y error de los datos que fueron clasificados en cierta clase *x*, mientras que la fila inferior de la gráfica muestra los porcentajes de acierto y error al clasificar elementos que pertenecen a una clase *x*.

La precisión general del modelo (considerando el total de datos validados) se puede apreciar en la celda inferior derecha de la matriz de confusión.

### 3.10.2. APLICACIÓN DEL COMANDO *plotconfusion*

Para comparar el vector de salidas estimadas (**Ye**) con el vector de respuestas reales (**Y40**), calcular el número de errores, y presentar en forma porcentual la eficacia del modelo, se recurrió a una matriz de confusión, con la sentencia indicada en el Espacio de código 3.9.

```
plotconfusion(Y40, Ye)
```

**Espacio de código 3.9.** Comando *plotconfusion* [Impresión de pantalla de MATLAB].

Tanto el vector **Ye** como el **Y40**, están constituidos por los siguientes valores:

- 1 y 2, para el *Modelo A*.
- 1 al 10, para el *Modelo B*.

Es importante señalar que, previamente, estos vectores deben ser transformados al formato que exige el comando *plotconfusion*. Para esto, cada vector debe convertirse en una matriz que contenga una cantidad de filas equivalente al mayor número del vector. El valor de cada elemento de este vector, será la posición de fila en la matriz que posea un 1, y el resto de elementos serán 0.

Para el caso del *Modelo A*, se observa que el vector de salida (parte superior de la Figura 3.11) tiene dos clases: el número 1 indica una falla en la línea anterior al relé, y el número 2 afirma que la falla ocurrió en la línea protegida. Después de la transformación, la matriz de salida (parte inferior de la Figura 3.11) queda formada solamente por 0 y 1. Cada fila representa una categoría.

1x1920 double										
	942	943	944	945	946	947	948	949	950	
1	1	1	1	1	2	2	2	2	2	^
2										

2x1920 double										
	942	943	944	945	946	947	948	949	950	
1	1	1	1	1	0	0	0	0	0	^
2	0	0	0	0	1	1	1	1	1	
3										

**Figura 3.11.** Parte del vector **Y40** del *Modelo A* [Impresión de pantalla de MATLAB].

De igual forma ocurre con el vector de salida del *Modelo B*, solo que en este caso existen diez categorías debido a los diez tipos de cortocircuitos.

Por ejemplo, en la Figura 3.12, cuando el vector tiene un valor de 4, se registra un 1 en la fila 4 de la matriz, y los demás elementos de la columna son 0.

Un 1 en la fila 4 hace referencia a una falla *Bifásica AB*; un 1 en la fila 5 de la matriz expresa una falla *Bifásica BC*. Un 1 en la fila 10 señala a una falla *Trifásica*. Estas equivalencias pueden visualizarse en la Tabla 3.4.

1x960 double									
	378	379	380	381	382	383	384	385	386
1	4	4	4	4	5	5	5	5	5
2									

10x960 double									
	378	379	380	381	382	383	384	385	386
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0
4	1	1	1	1	0	0	0	0	0
5	0	0	0	0	1	1	1	1	1
6	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0
11									

**Figura 3.12.** Parte del vector **Y40** del *Modelo B* [Impresión de pantalla de MATLAB].

En vista que el entrenamiento de datos tomó mucho tiempo en cada caso de simulación, aproximadamente de una hora (tiempo excesivo para este tipo de procedimiento), es recomendable implementar una técnica de transformación de datos, de manera que al consolidarlos, el proceso de extracción de patrones pueda ser más eficiente.

Esta nueva etapa, se ubica entre la *Formación de matrices de entrada y salida* y la *Selección de Conjuntos de Entrenamiento y Validación*, de acuerdo al diagrama de metodología (Figura 3.1).

### 3.11. ESTRATEGIAS DE TRANSFORMACIÓN DE DATOS

Existen varias estrategias de transformación de datos como: Eliminación de ruido, Construcción de Atributos, Agregación, Normalización, Discretización, etc. [24].

La técnica que se utiliza en este proceso de entrenamiento es la *Normalización*.

#### 3.11.1. NORMALIZACIÓN

Los atributos de un objeto (muestra o individuo) pueden ser expresados en diferentes unidades de medida (algunos valores podrían tener 5 cifras y otros 1 sola cifra), lo cual podrían afectar el análisis de los datos. En general, cuando un atributo se expresa en un rango más grande que otro, tenderá a producir un mayor efecto o *peso* sobre el resto. Para dar un peso igual a todos los atributos, los datos deben ser normalizados o estandarizados. Esto implica que deben ser transformados y ubicados dentro de un rango uniforme. Los más utilizados son:  $[-1, 1]$  y  $[0, 1]$ .

La normalización es útil para algoritmos de clasificación (como redes neuronales, SVMs) o para mediciones de distancia (como clasificación de vecinos más cercanos, *clustering*). Ayuda a acelerar la fase de aprendizaje, y evita que atributos con rangos inicialmente grandes compensen atributos con rangos más pequeños. Por ejemplo: ingresos versus datos binarios. Los ingresos podrían estar en el orden de los cientos o miles de dólares; mientras que los datos binarios solo contienen ceros (0) y unos (1). La normalización también es útil cuando no se tiene conocimiento previo de los datos [24].

Existen muchos métodos para la normalización de datos, en este caso se considera únicamente la *Normalización Min-Max*.

##### 3.11.1.1. Normalización Min-Max

La normalización Min-Max realiza una transformación lineal, preservando las relaciones entre los valores de los datos originales.

Se expresa con la siguiente ecuación [24]:

$$v'_i = \frac{v_i - \min_A}{\max_A - \min_A} (\text{new\_max}_A - \text{new\_min}_A) + \text{new\_min}_A \quad (3.1)$$

Donde:

- $A$ : Atributo.
- $\min_A$ : Valor mínimo del atributo  $A$ .
- $\max_A$ : Valor máximo del atributo  $A$ .
- $\text{new\_min}_A$ : Nuevo valor mínimo de  $A$ . Generalmente igual a -1, 0.
- $\text{new\_max}_A$ : Nuevo valor máximo de  $A$ . Generalmente igual a 1.
- $v_i$ : Valor original de  $A$ .
- $v'_i$ : Valor normalizado de  $v_i$  en el rango  $[\text{new\_min}_A, \text{new\_max}_A]$ .

### 3.11.2. COMANDO *mapminmax*

Procesa matrices mediante la normalización de valores mínimos y máximos de cada fila en el rango  $[YMIN, YMAX]$ .

#### 3.11.2.1. Descripción del comando *mapminmax*

$[Y, PS] = \text{mapminmax}(\mathbf{x}, YMIN, YMAX)$  toma como entrada la matriz  $\mathbf{x}$  de dimensión  $[N \times Q]$  y escala los datos por cada fila a un rango comprendido entre  $YMIN$  y  $YMAX$ , los cuales tienen un valor por defecto de -1 y 1 respectivamente, a menos que se les asigne un valor.

Este comando devuelve dos salidas, la matriz  $\mathbf{y}$ , de dimensión  $[N \times Q]$  (misma dimensión que  $\mathbf{x}$ ) y la variable  $PS$  que guarda las configuraciones del proceso, permitiendo un tratamiento coherente de nuevos valores.

#### 3.11.3. APLICACIÓN DEL COMANDO *mapminmax*

Para ampliar la forma de aplicación del comando *mapminmax*, se considera una matriz que ubica a los objetos (individuos) en filas, y a los atributos (variables) en columnas, conforme a la estructura de las matrices de entrada (Sección 3.7). En la Tabla 3.5 se presenta un ejemplo con ese formato.

**Tabla 3.5.** Matriz de 12 individuos con 6 atributos cada uno [Elaboración Propia]

		Características o Atributos					
		x1	x2	x3	x4	x5	x6
		Ingresos mensuales	¿Título 3er Nivel? 0 = No 1 = Sí	Estado Civil: 1 = Soltero; 2 = Casado; 3 = Unión libre; 4 = Divorciado; 5 = Viudo	Número de hijos	¿Vehículo? 0 = No 1 = Sí	Velocidad máx. de conducción (km/h)
Objetos o Individuos	< 20	100	0	1	0	0	0
	< 20	400	0	1	1	0	0
	20 - 30	1200	1	1	0	1	120
	20 - 30	850	1	2	1	0	0
	30 - 40	800	0	3	2	0	0
	30 - 40	1300	1	2	2	1	150
	40 - 50	1500	1	5	3	1	100
	40 - 50	1000	0	2	2	1	110
	50 - 60	1200	1	4	3	0	0
	50 - 60	600	0	5	2	0	0
	> 60	900	0	5	5	1	80
	> 60	600	0	2	4	0	0

En efecto, la Tabla 3.5 muestra que los datos de una misma columna tienen valores semejantes y comparables, no así los datos de cada fila, cuyos valores pertenecen a diferentes rangos. Por esta razón, la mejor opción para transformar los datos de una matriz, es normalizar *por variables*, que en este caso, se ubican en columnas.

Ya que el comando `mapminmax` de MATLAB realiza la normalización por filas, se ejecutó el Espacio de código 3.10 para obtener la nueva matriz de entrada ( $X$ ) normalizada por columnas.

```
[cD8_2f,PS1] = mapminmax(cD8_tot1');
X = cD8_2f'; %Matriz de entradas NORMALIZADA
```

**Espacio de código 3.10.** Normalización de la Matriz de entrada [Impresión de pantalla de MATLAB].

Adicionalmente, se optó por probar la normalización por filas, y la normalización de toda la matriz, con el objetivo de conocer cuán diferentes son los resultados, y verificar la forma correcta de transformar los datos con la estrategia de *Normalización*. Los resultados a estas pruebas se registran más adelante, en la Sección 4.5.1.

La normalización se aplica únicamente a las matrices de entrada, puesto que los vectores de salida contienen una sola columna y sus datos se encuentran en el mismo rango. Una vez que se hubieron transformado las matrices, se continuó con la metodología tal como indica la Figura 3.1; esto es, se procedió con la selección de los Conjuntos de Entrenamiento y Validación (Sección 3.8).

### **3.12. EVALUACIÓN O PRUEBA**

El proceso de evaluación de los modelos implica la creación de un nuevo y pequeño conjunto de muestras, obtenido de cortocircuitos simulados a longitudes de fracción de kilómetro, en la línea protegida y en la línea anterior a ésta. Es decir, que son señales totalmente diferentes a las consideradas hasta ahora. A estos datos se los denominó Conjunto de Evaluación, y fueron colocados en una carpeta de nombre *Señales Externas*, con los cuales se procedió a evaluar el modelo entrenado, comparando las predicciones con los datos reales.

El Conjunto de Evaluación está formado por 40 señales de falla. Esto quiere decir que se crearon 2 archivos para cada uno de los 20 casos posibles (10 tipos de falla en la línea protegida y 10 en la línea anterior).

### **3.13. ANÁLISIS DE RESULTADOS**

En esta sección, se tabularon y analizaron los resultados obtenidos en los procesos de normalización, entrenamiento, validación y evaluación.

Con objeto de conseguir el mínimo número de errores en la validación de los Modelos A y B, se efectuaron múltiples entrenamientos que dieron origen a los denominados *Modelos Predeterminados*, los cuales fueron almacenados posteriormente en una carpeta que lleva el mismo nombre. El análisis de resultados tiene su esencia justamente en el desempeño de los Modelos A y B Predeterminados. Mediante la aplicación de la técnica llamada *Matriz de Confusión*, se observó con claridad el porcentaje de acierto-error producido en la predicción de datos sobre el Conjunto de Validación. Estos resultados fueron un factor determinante para caracterizar si el prototipo es efectivo o no.

## CAPÍTULO 4

### APLICACIÓN DE LA METODOLOGÍA Y ANÁLISIS DE RESULTADOS

Este capítulo aborda secciones prácticas de la metodología descrita en el Capítulo 3, es decir, se dedica a presentar datos concretos que fueron aplicados durante el desarrollo del trabajo, por ejemplo se muestran las especificaciones de elementos que componen el sistema de prueba, los parámetros utilizados en la simulación de cortocircuitos, las dimensiones definidas en la base de datos, las matrices de entrada y salida. También se ocupa de tabular y analizar resultados, centrándose en la efectividad que presentan los Modelos A y B Predeterminados.

#### 4.1. IMPLEMENTACIÓN DEL SISTEMA DE PRUEBA

El caso de prueba es un sistema de potencia de 7 barras, que fue tomado de [36] y se muestra en la Figura 4.1. Como se puede observar, el sistema está compuesto por:

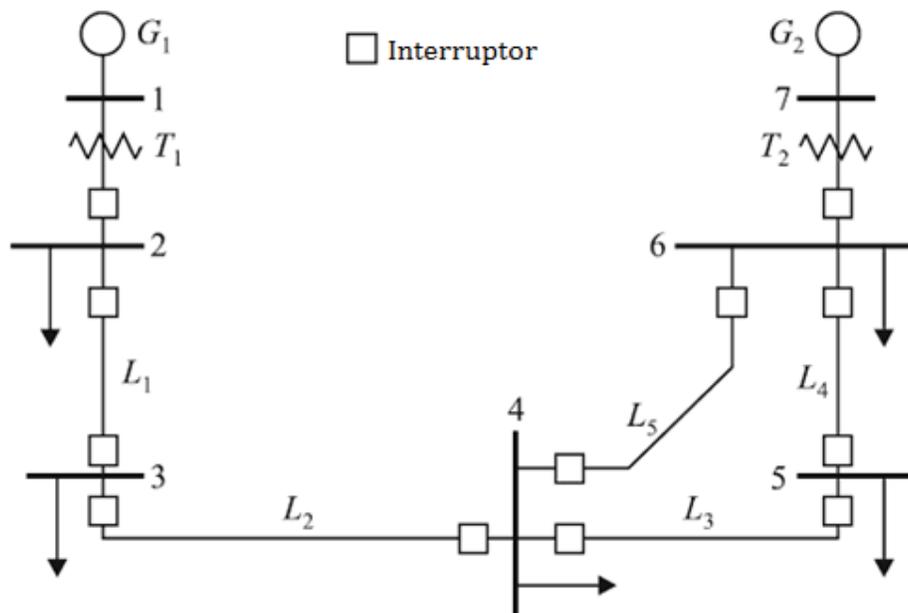


Figura 4.1. Sistema de prueba [36].

- 2 Generadores (G1, G2).
- 2 Transformadores (T1, T2).
- 5 Líneas de Transmisión (L1, L2, L3, L4, L5).
- 5 Cargas.

Es un sistema sencillo pero válido para demostrar la efectividad de los modelos creados. El programa computacional que se utilizó en la implementación de este sistema fue ATP. Todos los elementos que conforman el sistema de potencia (Figura 4.1), fueron añadidos en un nuevo archivo del programa ATP, presionando clic derecho en un espacio en blanco y escogiendo el elemento a ingresar en el menú contextual que aparece. Luego, se procedió a llenar la información de cada componente del sistema, de acuerdo a los datos presentados en las tablas de especificaciones técnicas. Este procedimiento se ejecutó para cada elemento y se muestra a continuación.

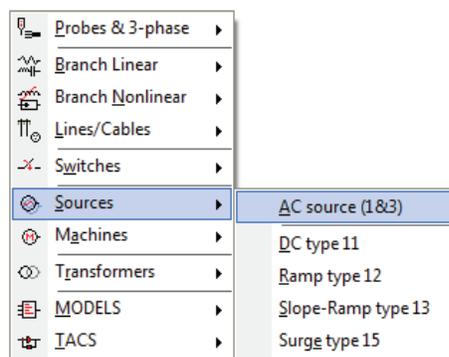
#### 4.1.1. GENERADORES

Las características de los generadores G1 y G2 se registran en la Tabla 4.1.

**Tabla 4.1.** Especificaciones técnicas de G1 y G2 [36].

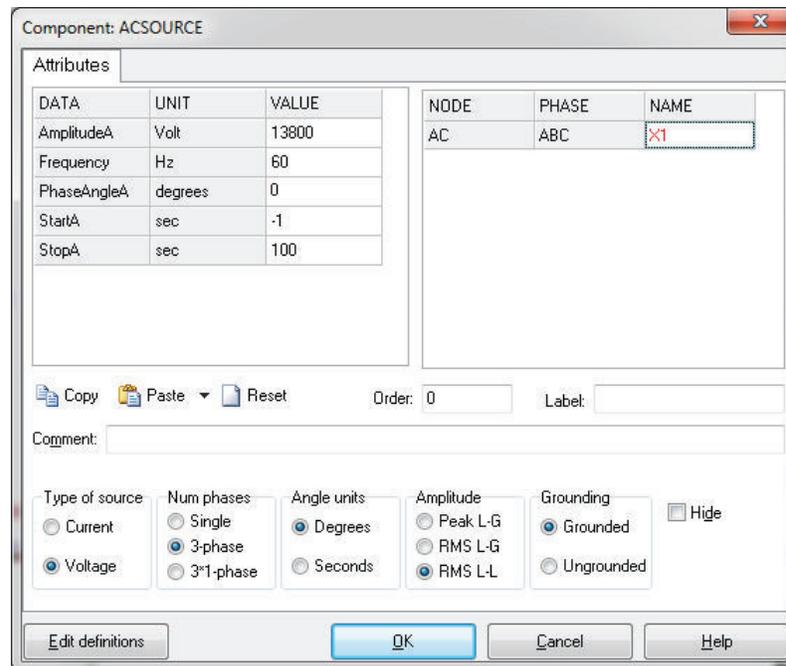
	Voltaje (kV)	Frecuencia (Hz)	Potencia (MVA)	Número de Fases	Puesta a tierra
<b>G1</b>	13,8	60	100	3	Sí
<b>G2</b>	15,0	60	200	3	Sí

Para ingresar este elemento, se elige *AC source (1&3)* de la opción *Sources* como se indica en la Figura 4.2.



**Figura 4.2.** Menú contextual para ingresar un Generador [Impresión de pantalla de ATP].

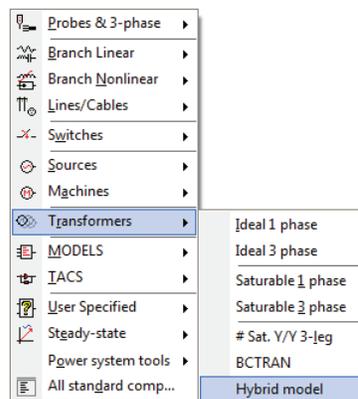
Después de ingresado el elemento, se hace *doble clic* o *clic derecho* sobre él, y se llenan los campos requeridos (Figura 4.3).



**Figura 4.3.** Características técnicas del Generador G1 [Impresión de pantalla de ATP].

#### 4.1.2. TRANSFORMADORES

En el menú contextual se opta por la opción *Transformers*, y luego *Hybrid model* (Figura 4.4). Las especificaciones técnicas de los transformadores T1 y T2 se registran en la Tabla 4.2. La Figura 4.5 es una captura de pantalla de las características de T1.



**Figura 4.4.** Menú contextual para ingresar un Transformador [Impresión de pantalla de ATP].

**Tabla 4.2.** Especificaciones técnicas de T1 y T2 [36].

	Potencia (MVA)	Voltaje MV / HV (kV)	Conexión MV / HV	Frecuencia (Hz)	Número de fases	Imp. X (p.u.)
T1	100	13,8 / 230	$\Delta / Y$	60	3	0,11
T2	200	15 / 230	$\Delta / Y$	60	3	0,11

Hybrid transformer :

Structure

Number of phases: 3

Number of windings: 2

Type of core: 3-leg stacked

Test frequency [Hz]: 60

Data based on: Ind. Res. Cap. Core

Design param.

Test report

Typical values

Ratings & connections

	Prim.	Sec.
L-L voltage [kV]	13.8	230
Power [MVA]	100	100
Connections	D	Y
Phase shifts	0	30
Node names	X2	X3

Winding sequence: core-inner-outer

Ext. neutral connections

Hide core nodes

Data

Inductance Resistance Capacitance Core

Data equal to Resistance's Test report  Zero sequence data available

**positive sequence**

	Imp. (%)	Pow. (MVA)	Loss (kW)
P-S	11	100	30

Order: 0 Label: Comment:  Hide

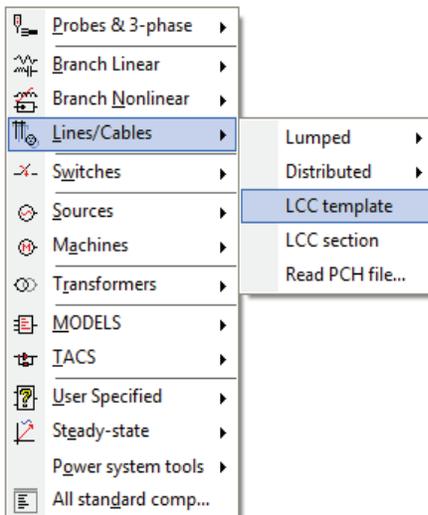
OK Cancel Import Export Edit defin. Help

**Figura 4.5.** Características técnicas del Transformador T1 [Impresión de pantalla de ATP].

#### 4.1.3. LÍNEAS DE TRANSMISIÓN

En el menú contextual se elige *Lines/Cables*, y luego *LCC template* (ver Figura 4.6). Las especificaciones técnicas de las líneas L1, L2, L3, L4, L5 se registran en la Tabla 4.3. Las cinco líneas del sistema están configuradas con el tipo *JMarti*, un modelo dependiente de la frecuencia con matriz de transformación constante. Además, es importante considerar que en las líneas L1 y L2 se realizaron las

simulaciones de cortocircuitos, y por este motivo, la implementación se elaboró de acuerdo al prototipo indicado en la Sección 3.2.3.1.



**Figura 4.6.** Menú contextual para ingresar una línea de transmisión [Impresión de pantalla de ATP].

**Tabla 4.3.** Especificaciones técnicas de L1, L2, L3, L4, L5 [36].

	Voltaje (kV)	Longitud (km)	Frecuencia (Hz)	Número de Fases
L1	230	240	60	3
L2	230	240	60	3
L3	230	40	60	3
L4	230	15	60	3
L5	230	50	60	3

Como ejemplo, en la Figura 4.7 se muestra una captura de pantalla de las características de L1. En este punto, es necesario explicar con mayor detalle el ingreso de variables en la pestaña *Model* y *Data*.

Considerando que se quiere implementar líneas de transmisión aéreas con el modelo *JMarti*, se presenta a continuación una breve descripción de los campos más importantes. Si se requiere mayor información sobre este elemento, se debe consultar el Manual indicado en la referencia [30].

Line/Cable Data: L1

Model Data Nodes

System type  
Name: L1  Template

Overhead Line  Transposed #Ph: 3

Auto bundling  
 Skin effect  
 Segmented ground  
 Real transf. matrix

Standard data  
Rho [ohm\*m] 100  
Freq. init [Hz] 0.01  
Length [km] 240  
 Set length in icon

Units  
 Metric  
 English

Model Type  
 Bergeron  
 PI  
 JMartí  
 Semlyen  
 Noda

Data  
Decades 7 Points/Dec 10  
Freq. matrix [Hz] 5000 Freq. SS [Hz] 60  
 Use default fitting

Comment: L1 Order: 0 Label:   Hide

OK Cancel Import Export Run ATP View Verify Edit defin. Help

---

Model Data Nodes

#	Ph.no.	Rin [cm]	Rout [cm]	Resis [ohm/km DC]	Horiz [m]	Vtower [m]	Vmid [m]	Separ [cm]	Alpha [deg]	NB
1	1	0.3615	1.2945	0.0795	-7.5	32.94	10.01	45	45	4
2	2	0.3615	1.2945	0.0795	0	32	9.07	45	45	4
3	3	0.3615	1.2945	0.0795	7.5	32.94	10.01	45	45	4
4	0	0	0.4575	5.222	-14.3	40	20.63	0	0	0
5	0	0.3776	0.85	0.3075	14.3	40	20.63	0	0	0

Add row Delete last row Insert row copy Move

OK Cancel Import Export Run ATP View Verify Edit defin. Help

Figura 4.7. Características técnicas de la línea de transmisión L1 [Impresión de pantalla de ATP].

#### 4.1.3.1. Pestaña Model

En el panel *System type*:

- *Transposed*: Si se marca el botón, la línea aérea se transpone.
- *Auto bundling*: Cuando se selecciona esta opción, se habilita la función de agrupación automática de LINE CONSTANTS.

- *Skin effect*: Si se pulsa este botón, se considera el efecto skin.
- *Metric / English*: Se puede elegir en qué sistema de unidades trabajar.
- *Segmented ground*: Cables de tierra segmentados. Si el botón está desactivado, se supone que los cables de tierra están continuamente conectados a tierra.
- *Real trans. matrix*: Si se selecciona la matriz de transformación se supone que es real, y pueden ser modificados por el usuario.

En el panel *Standard data*:

- *Rho [ohm×m]*: Resistividad del terreno en ohmios.
- *Freq init [Hz]*: El punto de menor frecuencia del ajuste de parámetros.
- *Length [km]*: Longitud de la línea aérea en kilómetros.
- *Set length in icon*: Mostrar la longitud de la línea en el ícono.

Dentro del panel *Model*, luego de seleccionar *JMarti*, en el subpanel *Type*, se requiere la siguiente información correspondiente al panel *Data*:

El modelo de línea *JMarti* se ubica en un rango de frecuencias que comienza desde el parámetro *Freq init* (en el panel *Standard data*) hasta un límite de frecuencia superior especificado por el número de *Decades* y el número de puntos de muestra por década (*Puntos/Dec*). El modelo también requiere:

- *Freq. Matriz*: Frecuencia donde se calcula la matriz de transformación.
- *Freq. SS*: Frecuencia de estado estacionario para el cálculo de la condición de estado estable.

#### 4.1.3.2. Pestaña *Data*

Describe la geometría de la línea de transmisión [30].

- *Ph.no.*: Número de fase a la que corresponde el conductor. 0 = para hilos de guardia.
- *Rin*: Radio interno del conductor (cm o pulgadas). Solo está disponible si se selecciona la casilla *Skin effect* en la pestaña *Model*.
- *Rout*: Radio externo del conductor (cm o pulgadas).

- *Resis*: Resistencia del conductor (ohm / unidad de longitud) en DC (con efecto skin activo), o resistencia de CA en *Freq. Init* (si no se selecciona el efecto skin).
- *Horiz*: Distancia horizontal (m o pie) desde el centro del haz de conductores hasta una línea de referencia seleccionable por el usuario.
- *Vtower*: Altura desde el suelo hasta el conductor o hasta el centro del haz de conductores (m o pie).
- *Vmid*: Altura del conductor o haz de conductores en la mitad del vano (m o pie).

Si se selecciona *Auto bundling* en el panel *System type* de la pestaña *Model*:

- *Separ*: Distancia entre conductores en un haz (cm o pulgada).
- *Alpha*: Posición angular de uno de los conductores en un haz, medida en sentido antihorario desde la línea horizontal.
- *NB*: Número de conductores en un haz.

#### 4.1.4. CARGAS

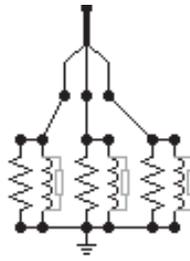
Las especificaciones técnicas para todas las cargas son las mismas. Estas se registran en la Tabla 4.4.

**Tabla 4.4.** Especificaciones técnicas de las cargas C1, C2, C3, C4, C5 [36].

	Voltaje (kV)	Potencia Activa P (MW)	Potencia Reactiva Q (MVA <sub>r</sub> )	Frecuencia (Hz)	Número de Fases
C1, C2, C3, C4, C5	230	50	30	60	3

La modelación de las cargas (Figura 4.8) se llevó a cabo partiendo de sus respectivas potencias P y Q, de las cuales se obtuvieron los valores de R y X por fase, ya que cada ramal de la carga trifásica está constituido por el paralelo R-X.

A continuación se muestra el proceso:



**Figura 4.8.** Modelación en paralelo de cargas trifásicas [Impresión de pantalla de ATP].

$$P = \frac{V^2}{R} \quad (4.1)$$

$$Q = \frac{V^2}{X} \quad (4.2)$$

De las ecuaciones (4.1) y (4.2) se despejaron  $R$  y  $X$ , respectivamente. Luego, se reemplazaron los valores de potencias y voltaje por fase.

Las potencias  $P = 50 \text{ MW}$  y  $Q = 30 \text{ MWAr}$  son valores trifásicos, y deben ser divididos para tres. El voltaje  $V = 230 \text{ kV}$  es un valor fase-fase, por lo cual se dividió para  $\sqrt{3}$ , obteniendo un valor fase-neutro.

*Cálculo de  $R$ :*

$$R = \frac{\left(\frac{V}{\sqrt{3}}\right)^2}{\frac{P}{3}} \quad (4.3)$$

$$R = \frac{V^2}{P} = \frac{(230 \text{ kV})^2}{50 \text{ MW}}$$

$$R = 1058 \Omega$$

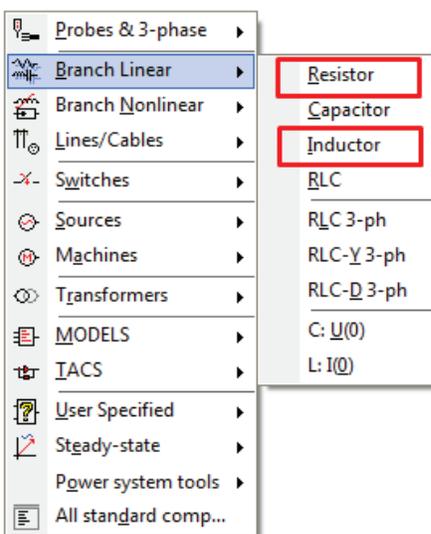
*Cálculo de  $X$ :*

$$X = \frac{\left(\frac{V}{\sqrt{3}}\right)^2}{\frac{P}{3}} \quad (4.4)$$

$$X = \frac{V^2}{Q} = \frac{230^2}{30}$$

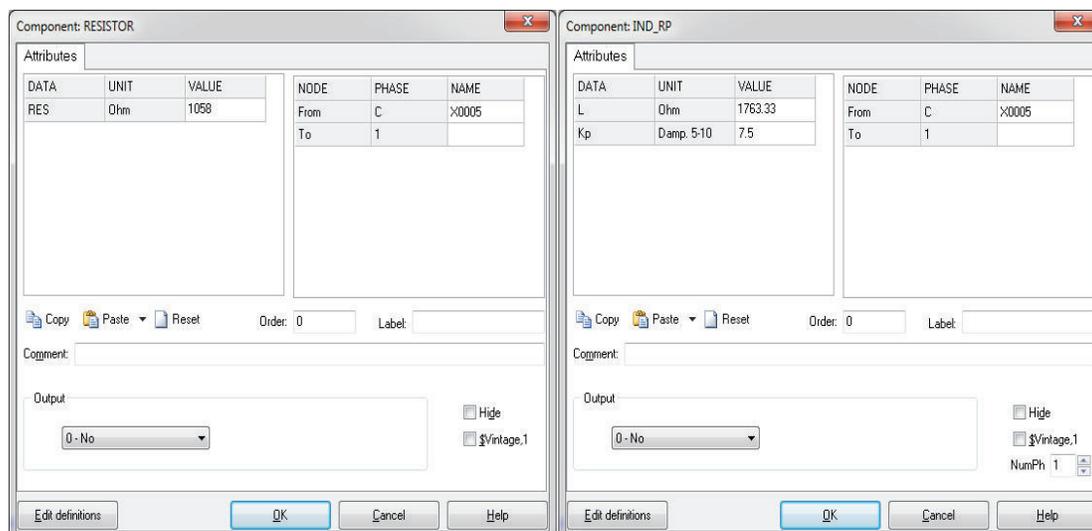
$$X = 1763,333 \Omega$$

Una vez conocidos los valores de  $R$  y  $X$  por fase, se ingresaron tres resistencias y tres reactancias para modelar la carga total, de acuerdo al esquema de la Figura 4.8. En el menú contextual se elige *Branch Linear*, seguido de *Resistor* o *Inductor* como se indica en la Figura 4.9.



**Figura 4.9.** Menú contextual para ingresar una  $R$  o  $X$  [Impresión de pantalla de ATP].

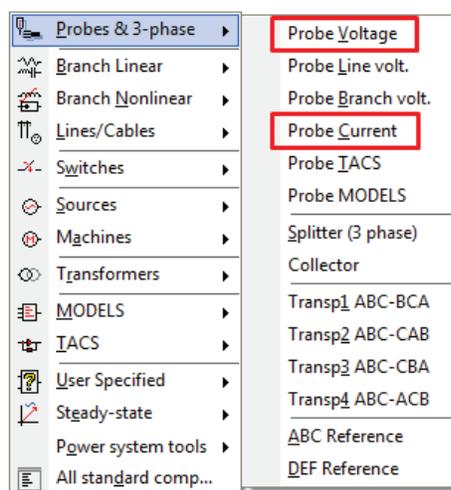
Como ejemplo, se muestra una captura de pantalla de las características de  $R$  y  $X$ , en la Figura 4.10.



**Figura 4.10.** Características de  $R$  y  $X$  de las cargas [Impresión de pantalla de ATP].

#### 4.1.5. EQUIPOS DE MEDICIÓN

Es necesario añadir medidores de corriente y voltaje para sensar las señales en diferentes puntos del sistema. En el menú contextual de la Figura 4.11, se escoge la opción *Probes & 3-phase*, y después *Probe Voltage* o *Probe Current*, dependiendo del medidor que se quiera instalar.



**Figura 4.11.** Menú contextual para ingresar medidores de voltaje o corriente [Impresión de pantalla de ATP].

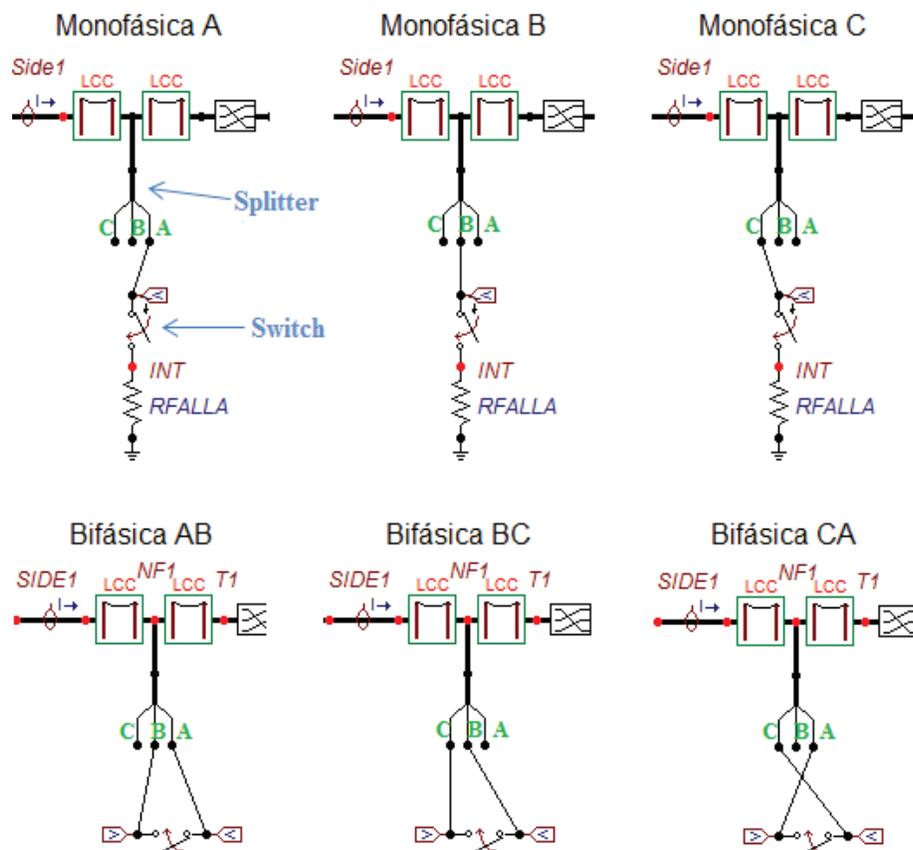
#### 4.2. ARCHIVOS BASE EN ATP

Los cuatro archivos base sobre los cuales se realizaron las simulaciones (de 10 tipos de cortocircuitos) llevan los siguientes nombres:

- i) *basecase1.atp* → fallas monofásicas: A, B y C.
- ii) *basecase2.atp* → fallas bifásicas: AB, BC y CA.
- iii) *basecase3.atp* → fallas bifásicas-tierra: AB-Gnd, BC-Gnd y CA-Gnd.
- iv) *basecase4.atp* → fallas trifásicas.

Para modificar el tipo de falla dentro del mismo archivo, basta con modificar la conexión del *switch* con el *splitter*. En la Figura 4.12 se puede observar las conexiones para fallas monofásicas y bifásicas.

Después de implementar cada caso, se ejecutó el archivo *.acp* en ATPDraw para obtener el archivo *.atp* requerido por MATLAB.



**Figura 4.12.** Conexiones del *switch* en ATP para fallas monofásicas y bifásicas [Impresiones de pantalla de ATP].

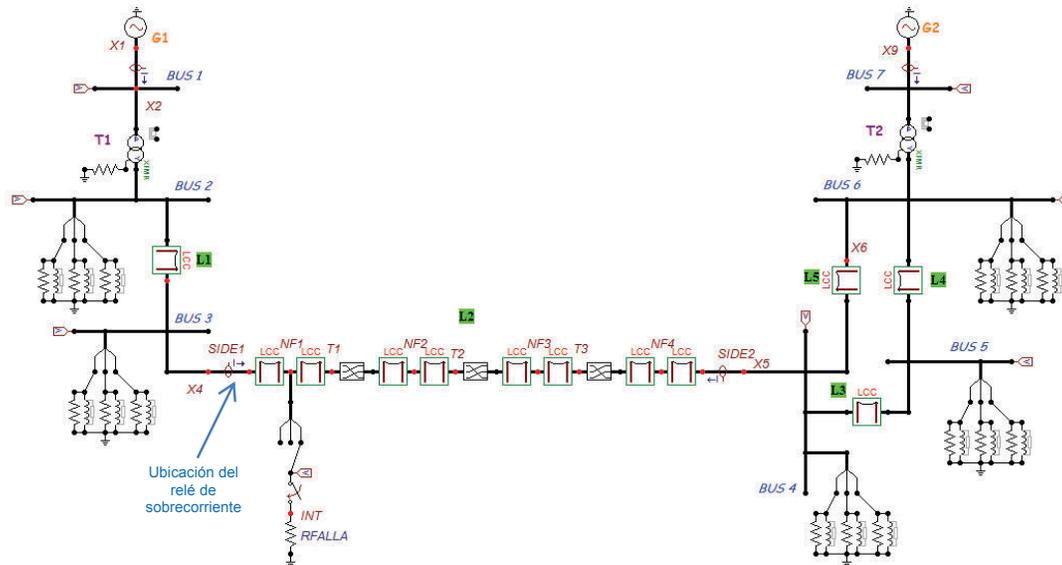
### 4.3. SIMULACIÓN DE CORTOCIRCUITOS VINCULANDO ATP Y MATLAB

La ubicación del relé de sobrecorriente se consideró en el extremo izquierdo de la línea L2 (ver Figura 4.1); por tanto, fue necesario simular cortocircuitos a lo largo de dos líneas de transmisión:

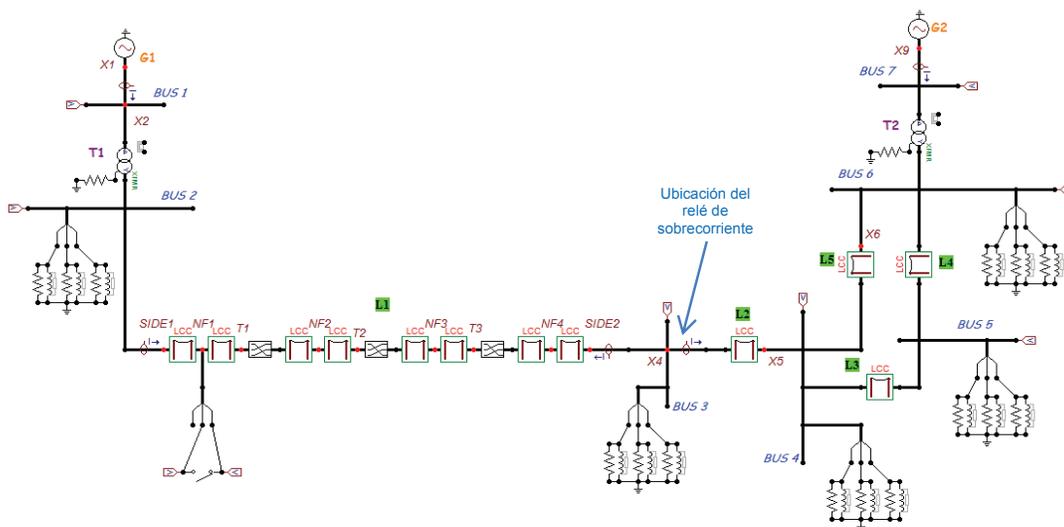
- *L2*: La línea protegida, ya que el relé debe detectar fallas *aguas abajo* (fallas en L2).
- *L1*: La línea anterior a L2. Si una falla ocurre en L1 (*aguas arriba*), la corriente tendrá una dirección contraria a la falla en L2; y el relé debe ser capaz de distinguir entre ambas señales.

El modelo de línea para simulación múltiple de fallas (presentado en la Sección 3.2.3.1) se ubicó primero en L2. El sistema completo implementado en ATP, se

muestra en la Figura 4.13. El mismo sistema de potencia, pero ubicando el prototipo de línea en L1, se presenta en la Figura 4.14.



**Figura 4.13.** Sistema de prueba en ATP para simulaciones de fallas en L2 (línea protegida) [Impresión de pantalla de ATP].



**Figura 4.14.** Sistema de prueba en ATP para simulaciones de fallas en L1 (línea anterior) [Impresión de pantalla de ATP].

Antes de empezar con las respectivas simulaciones, fue necesario inicializar algunos parámetros en el script *runatp matlab* de MATLAB. A continuación se presentan los más importantes: *Longitud de la línea* y *Tiempos de referencia*.

- **Longitud de la línea**

De acuerdo a la Tabla 4.3, la longitud de las líneas L1 y L2 es *240 km*, cuyo valor se registró en la función de simulación *runatpmatlab*.

- **Tiempos de referencia**

Los tiempos de referencia, conocidos como  $t_0$  dentro del script, fueron tomados de acuerdo el cruce por cero de los voltajes graficados en ATP.

En la Tabla 4.5 se muestran los respectivos valores para todos los tipos de falla.

**Tabla 4.5.** Tiempos de cruce por cero de la señal de voltaje para todas las simulaciones [Elaboración propia].

Simulaciones en línea	Voltaje Fase	Tiempo de cruce por cero [s]	Para Fallas
L1	A	0,016231	<ul style="list-style-type: none"> <li>• Monofásicas A</li> <li>• Bifásicas AB</li> <li>• Bifásicas AB-Gnd</li> <li>• Trifásicas</li> </ul>
	B	0,021806	<ul style="list-style-type: none"> <li>• Monofásicas B</li> <li>• Bifásicas BC</li> <li>• Bifásicas BC-Gnd</li> </ul>
	C	0,027350	<ul style="list-style-type: none"> <li>• Monofásicas C</li> <li>• Bifásicas CA</li> <li>• Bifásicas CA-Gnd</li> </ul>
L2	A	0,016098	<ul style="list-style-type: none"> <li>• Monofásicas A</li> <li>• Bifásicas AB</li> <li>• Bifásicas AB-Gnd</li> <li>• Trifásicas</li> </ul>
	B	0,021663	<ul style="list-style-type: none"> <li>• Monofásicas B</li> <li>• Bifásicas BC</li> <li>• Bifásicas BC-Gnd</li> </ul>
	C	0,027213	<ul style="list-style-type: none"> <li>• Monofásicas C</li> <li>• Bifásicas CA</li> <li>• Bifásicas CA-Gnd</li> </ul>

#### 4.3.1. ENTRADAS DE LA FUNCIÓN DE SIMULACIÓN “runatpmatlab”

Es importante especificar algunos valores constantes que se tomaron como argumentos de la función *runatpmatlab*. De acuerdo al formato de la sentencia `[verifica] = runatpmatlab(1,10,90,1,1)`, analizada en la Sección 3.4.1, se estableció que:

1. El primer argumento es el único que varía debido al tipo de falla.
2. El segundo argumento tiene un valor de 10. Para este campo se consideró únicamente una resistencia de falla de 10 ohmios, para todos los posibles casos. Además, para fallas bifásicas o trifásicas, donde no interviene la resistencia de falla, el programa coloca automáticamente un valor de 0, no importa qué valor se ingrese.
3. El tercer argumento, para todos los casos, tiene un valor de 90° (grados), indicando que la falla se produce a  $90^\circ = 0,004167$  segundos después del tiempo de referencia (cruce por cero).
4. El parámetro 4, para todos los casos, tiene un valor de 1, porque la primera falla ocurre al kilómetro del inicio de la línea.
5. El parámetro 5, para todos los casos, también tiene un valor de 1, ya que se simularon cortocircuitos en pasos de 1 km de longitud.

Es decir que, prácticamente cuatro de los cinco parámetros tienen valores constantes en las diferentes simulaciones.

#### 4.4. MATRICES DE ENTRADA Y SALIDA

Antes de presentar la estructura de las matrices de entrada y salida, conviene tener en cuenta la siguiente información que fue empleada durante las simulaciones de cortocircuitos, y la aplicación de la Transformada Wavelet Discreta en MATLAB:

- En cada simulación, se tomaron señales de corriente de las tres fases (*a*, *b*, *c*), cada una compuesta por 35266 atributos (o puntos muestreados).

- Se aplicó la DWT de nivel 8 a cada corriente de fase obtenida. Por tanto, los coeficientes de detalle cD8 quedaron constituidos por 144 valores.
- Los cD8 de fase *a*, *b* y *c* de una misma falla, se colocaron en la misma fila de la matriz. En total suman 432 valores.
- La línea L1, al igual que L2, tiene una extensión de 240 km. Se simularon fallas en cada kilómetro de longitud. Entonces, existen 240 elementos o individuos por cada tipo de falla.

Para la **Matriz A**:

- Se registraron 10 tipos de falla en L1 y 10 tipos en L2, en total 20. De esta manera, el número total de filas fue de 4800 ( $240 \times 20$ ).

Para la **Matriz B**:

- Se especificaron 10 tipos de falla en L2. Como resultado se consiguieron  $240 \times 10 = 2400$  filas.

El procedimiento completo de la formación de las matrices de entrada y salida se describió en la Sección 3.7.

En la Tabla 4.6, se puede apreciar que la dimensión de la **Matriz A** es de 4800 x 432. Esto quiere decir que consta de 4800 individuos con 432 atributos cada uno. Así mismo, la dimensión de la **Matriz B** es de 2400 x 432, y se puede observar en la Tabla 4.7.

Evidentemente, los vectores de salida: **Vector A** y **Vector B**, tienen el mismo número de filas que la **Matriz A** y **Matriz B**, respectivamente. Poseen una sola columna que contiene el número equivalente a su tipo de falla (ver Tabla 4.8).

Se puede apreciar que las matrices de entrada y salida cumplen con el formato establecido en la Sección 3.7. Cada individuo de las matrices pertenece a un espacio de 432 dimensiones. Por tanto, se considera apropiado trabajar con SVMs, debido a su capacidad de generalización en  $R^n$ .

**Tabla 4.6. Matriz A** para el sistema de prueba [Elaboración propia].

		Dim. [n x m]	1	2	...	143	144	145	146	...	287	288	289	290	...	431	432
Línea 1 (anterior)	1φ - A	1	cD8a1					cD8b1					cD8c1				
		⋮	cD8a1					cD8b1					cD8c1				
		240	cD8a1					cD8b1					cD8c1				
	1φ - B	241	cD8a2					cD8b2					cD8c2				
		⋮	cD8a2					cD8b2					cD8c2				
		480	cD8a2					cD8b2					cD8c2				
	1φ - C	481	cD8a3					cD8b3					cD8c3				
		⋮	cD8a3					cD8b3					cD8c3				
		720	cD8a3					cD8b3					cD8c3				
	2φ - AB	721	cD8a4					cD8b4					cD8c4				
		⋮	cD8a4					cD8b4					cD8c4				
		960	cD8a4					cD8b4					cD8c4				
	2φ - BC	961	cD8a5					cD8b5					cD8c5				
		⋮	cD8a5					cD8b5					cD8c5				
		1200	cD8a5					cD8b5					cD8c5				
	2φ - CA	1201	cD8a6					cD8b6					cD8c6				
		⋮	cD8a6					cD8b6					cD8c6				
		1440	cD8a6					cD8b6					cD8c6				
	2φ Gnd - AB	1441	cD8a7					cD8b7					cD8c7				
		⋮	cD8a7					cD8b7					cD8c7				
		1680	cD8a7					cD8b7					cD8c7				
	2φ Gnd - BC	1681	cD8a8					cD8b8					cD8c8				
		⋮	cD8a8					cD8b8					cD8c8				
		1920	cD8a8					cD8b8					cD8c8				
	2φ Gnd - CA	1921	cD8a9					cD8b9					cD8c9				
		⋮	cD8a9					cD8b9					cD8c9				
		2160	cD8a9					cD8b9					cD8c9				
	3φ	2161	cD8a10					cD8b10					cD8c10				
		⋮	cD8a10					cD8b10					cD8c10				
		2400	cD8a10					cD8b10					cD8c10				
Línea 2 (protegida)	1φ - A	2401	cD8a11					cD8b11					cD8c11				
		⋮	cD8a11					cD8b11					cD8c11				
		2640	cD8a11					cD8b11					cD8c11				
	1φ - B	2641	cD8a12					cD8b12					cD8c12				
		⋮	cD8a12					cD8b12					cD8c12				
		2880	cD8a12					cD8b12					cD8c12				
	1φ - C	2881	cD8a13					cD8b13					cD8c13				
		⋮	cD8a13					cD8b13					cD8c13				
		3120	cD8a13					cD8b13					cD8c13				
	2φ - AB	3121	cD8a14					cD8b14					cD8c14				
		⋮	cD8a14					cD8b14					cD8c14				
		3360	cD8a14					cD8b14					cD8c14				
	2φ - BC	3361	cD8a15					cD8b15					cD8c15				
		⋮	cD8a15					cD8b15					cD8c15				
		3600	cD8a15					cD8b15					cD8c15				
	2φ - CA	3601	cD8a16					cD8b16					cD8c16				
		⋮	cD8a16					cD8b16					cD8c16				
		3840	cD8a16					cD8b16					cD8c16				

2 $\phi$ Gnd - AB	3841	cD8a17	cD8b17	cD8c17
	⋮	cD8a17	cD8b17	cD8c17
	4080	cD8a17	cD8b17	cD8c17
2 $\phi$ Gnd - BC	4081	cD8a18	cD8b18	cD8c18
	⋮	cD8a18	cD8b18	cD8c18
	4320	cD8a18	cD8b18	cD8c18
2 $\phi$ Gnd - CA	4321	cD8a19	cD8b19	cD8c19
	⋮	cD8a19	cD8b19	cD8c19
	4560	cD8a19	cD8b19	cD8c19
3 $\phi$	4561	cD8a20	cD8b20	cD8c20
	⋮	cD8a20	cD8b20	cD8c20
	4800	cD8a20	cD8b20	cD8c20

**Tabla 4.7. Matriz B para el sistema de prueba [Elaboración propia].**

		Dim. [n x m]	1	2	...	143	144	145	146	...	287	288	289	290	...	431	432
Línea 2 (protegida)	1 $\phi$ - A	1	cD8a1					cD8b1					cD8c1				
		⋮	cD8a1					cD8b1					cD8c1				
		240	cD8a1					cD8b1					cD8c1				
	1 $\phi$ - B	241	cD8a2					cD8b2					cD8c2				
		⋮	cD8a2					cD8b2					cD8c2				
		480	cD8a2					cD8b2					cD8c2				
	1 $\phi$ - C	481	cD8a3					cD8b3					cD8c3				
		⋮	cD8a3					cD8b3					cD8c3				
		720	cD8a3					cD8b3					cD8c3				
	2 $\phi$ - AB	721	cD8a4					cD8b4					cD8c4				
		⋮	cD8a4					cD8b4					cD8c4				
		960	cD8a4					cD8b4					cD8c4				
	2 $\phi$ - BC	961	cD8a5					cD8b5					cD8c5				
		⋮	cD8a5					cD8b5					cD8c5				
		1200	cD8a5					cD8b5					cD8c5				
	2 $\phi$ - CA	1201	cD8a6					cD8b6					cD8c6				
		⋮	cD8a6					cD8b6					cD8c6				
		1440	cD8a6					cD8b6					cD8c6				
	2 $\phi$ Gnd - AB	1441	cD8a7					cD8b7					cD8c7				
		⋮	cD8a7					cD8b7					cD8c7				
		1680	cD8a7					cD8b7					cD8c7				
	2 $\phi$ Gnd - BC	1681	cD8a8					cD8b8					cD8c8				
		⋮	cD8a8					cD8b8					cD8c8				
		1920	cD8a8					cD8b8					cD8c8				
	2 $\phi$ Gnd - CA	1921	cD8a9					cD8b9					cD8c9				
		⋮	cD8a9					cD8b9					cD8c9				
		2160	cD8a9					cD8b9					cD8c9				
	3 $\phi$	2161	cD8a10					cD8b10					cD8c10				
		⋮	cD8a10					cD8b10					cD8c10				
		2400	cD8a10					cD8b10					cD8c10				

**Tabla 4.8.** Vectores de Salida A y B para el sistema de prueba  
[Elaboración propia].

**Vector A**

Dim. [n x m]	Falla en:	SIGNIFICADO
1	1	Línea anterior
2	1	Línea anterior
3	1	Línea anterior
⋮	1	Línea anterior
2398	1	Línea anterior
2399	1	Línea anterior
2400	1	Línea anterior
2401	2	Línea protegida
2402	2	Línea protegida
2403	2	Línea protegida
⋮	2	Línea protegida
4798	2	Línea protegida
4799	2	Línea protegida
4800	2	Línea protegida

**Vector B**

Dim. [n x m]	Tipo de falla:	SIGNIFICADO
1	1	Monofásica A
⋮	1	Monofásica A
240	1	Monofásica A
241	2	Monofásica B
⋮	2	Monofásica B
480	2	Monofásica B
481	3	Monofásica C
⋮	3	Monofásica C
720	3	Monofásica C
721	4	Bifásica AB
⋮	4	Bifásica AB
960	4	Bifásica AB
961	5	Bifásica BC
⋮	5	Bifásica BC
1200	5	Bifásica BC
1201	6	Bifásica CA
⋮	6	Bifásica CA
1440	6	Bifásica CA
1441	7	Bifásica AB - Tierra
⋮	7	Bifásica AB - Tierra
1680	7	Bifásica AB - Tierra
1681	8	Bifásica BC - Tierra
⋮	8	Bifásica BC - Tierra
1920	8	Bifásica BC - Tierra
1921	9	Bifásica CA - Tierra
⋮	9	Bifásica CA - Tierra
2160	9	Bifásica CA - Tierra
2161	10	Trifásica
⋮	10	Trifásica
2400	10	Trifásica

## 4.5. ANÁLISIS DE RESULTADOS

Esta sección presenta los resultados de la metodología, en los procesos de:

- Normalización.
- Entrenamiento.
- Validación.
- Evaluación.

Los datos fueron tabulados de forma que sean apreciables para el lector y permitan realizar un análisis objetivo de los mismos.

Los *Modelos A y B Predeterminados*, que son el objetivo principal de la metodología, se analizan en la Sección 4.5.3 (Validación).

### 4.5.1. NORMALIZACIÓN

A continuación, se presentan los resultados de los Modelos A y B, mediante porcentajes de validación obtenidos con la *Matriz de confusión* (Sección 3.10.1). Dichos modelos fueron entrenados con matrices de entrada donde se aplicó previamente la normalización por filas, por columnas, y de toda la matriz.

#### 4.5.1.1. Normalización de toda la matriz

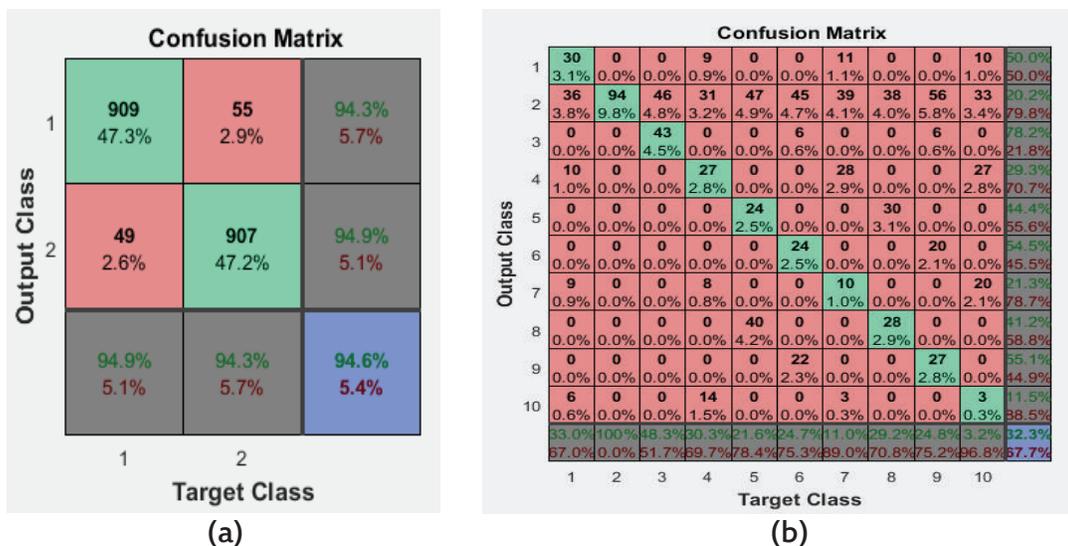
En la Tabla 4.9 se observa la efectividad de diez Modelos A y B, entrenados después del proceso de normalización de toda la matriz de entrada.

**Tabla 4.9.** Porcentajes de validación para Modelos A y B con normalización de toda la matriz de entrada [Elaboración propia].

No.	Modelo A (%)	Modelo B (%)
1	92,8	33,8
2	93,4	37,5
3	92,9	41,1
4	96,3	28,4
5	94,6	32,3
6	95,4	29,3
7	93,2	51,2
8	94,9	34,3
9	91,1	31,7
10	95,7	32,5

En la Figura 4.15 se muestran como ejemplo, las matrices de confusión del quinto par de modelos, resaltado en la Tabla 4.9.

Los porcentajes de validación del Modelo B son demasiado bajos, y aunque los porcentajes del Modelo A son superiores, los valores obtenidos en ambos casos no son suficientemente altos para calificar a los modelos como aceptables.



**Figura 4.15.** Matrices de confusión con normalización de toda la matriz de entrada. a) Para el Modelo A; b) Para el Modelo B [Impresión de pantalla de MATLAB].

#### 4.5.1.2. Normalización por filas

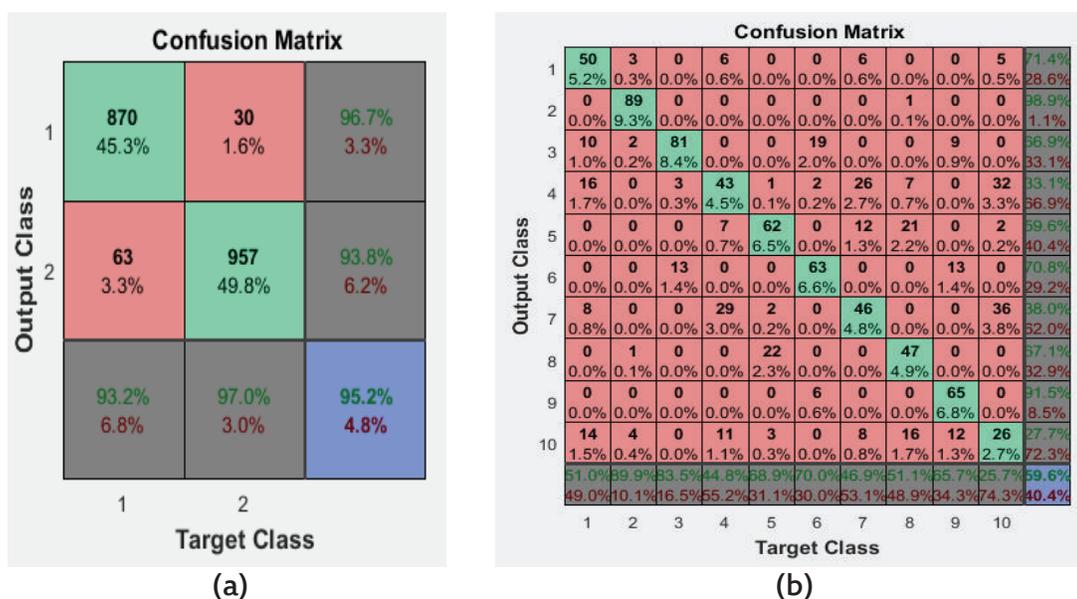
Luego de aplicar la normalización por filas en las matrices de entrada, se realizaron diez entrenamientos, cuyos modelos fueron validados y dieron como resultado los porcentajes de acierto de la Tabla 4.10.

En la Figura 4.16, se presentan las matrices de confusión de los primeros modelos entrenados.

Al igual que en la *Normalización de toda la matriz* (Sección 4.5.1.1), los porcentajes de validación obtenidos en la *Normalización por filas* para el Modelo A, son mayores que para el Modelo B, pero ninguno llega a un porcentaje de aciertos aceptable.

**Tabla 4.10.** Porcentaje de validación para Modelos A y B con normalización por filas en la matriz de entrada [Elaboración propia].

No.	Modelo A (%)	Modelo B (%)
1	95,2	59,6
2	95,4	58,4
3	94,2	52,9
4	98,7	61,3
5	95,7	51,8
6	93,6	55,2
7	96,2	60,6
8	94,3	53,7
9	97,1	48,5
10	91,9	54,5



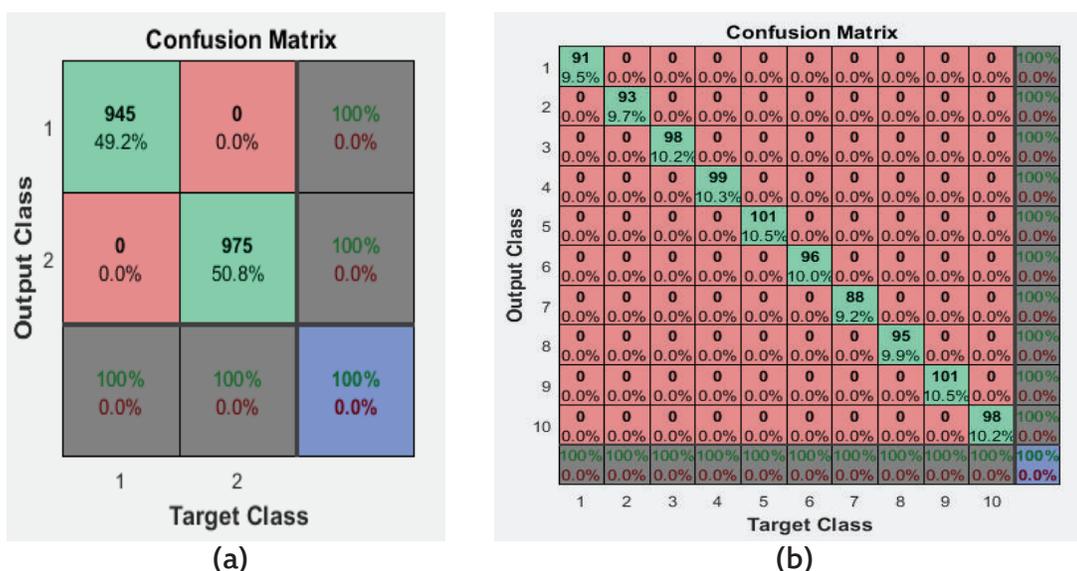
**Figura 4.16.** Matrices de confusión con normalización por filas en la matriz de entrada. a) Para el Modelo A; b) Para el Modelo B [Impresión de pantalla de MATLAB].

#### 4.5.1.3. Normalización por columnas

La Tabla 4.11 muestra porcentajes de validación obtenidos para diez entrenamientos del Modelo A y diez del Modelo B, con matrices de entrada normalizadas por columnas. En la Figura 4.17 se encuentran las matrices de confusión aplicadas a dichos modelos en el noveno entrenamiento.

**Tabla 4.11.** Porcentaje de validación para Modelos A y B con normalización por columnas en la matriz de entrada [Elaboración propia].

No.	Modelo A (%)	Modelo B (%)
1	100,0	100,0
2	100,0	100,0
3	100,0	99,7
4	100,0	99,5
5	99,7	100,0
6	100,0	99,2
7	100,0	100,0
8	100,0	99,5
9	100,0	100,0
10	99,8	100,0



**Figura 4.17.** Matrices de confusión con normalización por columnas en la matriz de entrada. a) Para el Modelo A; b) Para el Modelo B [Impresión de pantalla de MATLAB].

En este caso, se observa que los porcentajes de validación de ambos modelos son iguales al 100% o muy cercanos a este valor. En los casos de Normalización de toda la matriz y Normalización por filas, el porcentaje de acierto de los modelos se encontraba alrededor del 30% y 60%, valores muy por debajo de lo deseado.

Por tanto, se corrobora que la mejor opción para normalizar las matrices de entrada, es hacerlo por columnas, ya que las variables de cada individuo que forman las matrices de entrada, están ubicadas en columnas.

#### 4.5.2. RESULTADOS DE ENTRENAMIENTO

En la Sección 3.9 se explicó que cada modelo (A y B) debe evaluarse primero con el Conjunto de Entrenamiento, es decir con las mismas muestras que fueron entrenadas (60%), y luego con un conjunto diferente de datos. Los resultados para veinte entrenamientos se resumen en Tabla 4.12.

**Tabla 4.12.** Resultados de Entrenamiento [Elaboración propia].

No.	Modelo A (%)	Modelo B (%)
1	100,0	100,0
2	100,0	100,0
3	100,0	100,0
4	100,0	100,0
5	100,0	100,0
6	100,0	99,9
7	100,0	100,0
8	100,0	100,0
9	99,8	100,0
10	100,0	100,0
11	100,0	99,8
12	100,0	100,0
13	100,0	100,0
14	100,0	100,0
15	100,0	99,9
16	100,0	99,7
17	100,0	100,0
18	100,0	100,0
19	100,0	100,0
20	100,0	100,0

En casi todos los casos, los modelos fueron entrenados perfectamente. Los porcentajes de precisión son muy cercanos al 100%. Evidentemente, se puede decir que el proceso de entrenamiento se ha realizado con éxito.

#### 4.5.3. RESULTADOS DE VALIDACIÓN

Luego de verificar la efectividad de los Modelos A y B con el Conjunto de Entrenamiento, se utilizó el Conjunto de Validación, tal como se menciona en la Sección 3.10. Los resultados se encuentran en la Tabla 4.13

**Tabla 4.13.** Resultados de Validación [Elaboración propia].

No.	Modelo A (%)	Modelo B (%)
1	99,2	98,5
2	99,8	99,1
3	100,0	99,6
4	99,9	100,0
5	99,4	98,4
6	100,0	99,9
7	100,0	99,7
8	98,7	99,4
9	99,5	100,0
10	99,7	99,6
11	100,0	98,8
12	99,%	98,6
13	100,0	99,5
14	99,6	99,2
15	99,3	99,6
16	100,0	98,4
17	99,7	100,0
18	100,0	99,7
19	99,7	99,6
20	98,3	99,7

Los porcentajes de acierto presentados en la Tabla 4.13 son muy buenos. Se mantienen cercanos al 100%; sin embargo, se debe resaltar que son menores a los porcentajes obtenidos en los Resultados de Entrenamiento. De todas maneras, fue posible conseguir modelos cuya validación sea del 100% (valores resaltados en la Tabla 4.13). Para cada modelo, se tomó uno de estos ejemplos, y se los denominó *Modelos A y B Predeterminados*, que son el principal objeto de análisis de este capítulo y de todo el trabajo.

#### 4.5.3.1. Modelos A y B Predeterminados

Los Modelos A y B Predeterminados se analizaron con sus respectivas matrices de confusión, que se muestran en la Figura 4.18 y Figura 4.19, respectivamente.

El total de individuos utilizados para la validación del *Modelo A predeterminado* es 1920 (que corresponde al 40% del total 4800). Este valor equivale la suma de todos los elementos de la matriz de confusión. En la Figura 4.18, se suman únicamente los valores:  $945 + 975 = 1920$ , porque el resto de celdas son nulas.

	1	2	
1	945 49.2%	0 0.0%	100% 0.0%
2	0 0.0%	975 50.8%	100% 0.0%
	100% 0.0%	100% 0.0%	100% 0.0%
	1	2	
	Target Class		

Figura 4.18. Matriz de confusión del *Modelo A predeterminado* [Impresión de pantalla de MATLAB].

Del mismo modo, en la validación del *Modelo B predeterminado*, se tomó el 40% del total 2400, que equivale a 960 individuos. En la Figura 4.19, los valores que deben sumarse son:  $91 + 93 + 98 + 99 + 101 + 96 + 88 + 95 + 101 + 98 = 960$ . En ambos casos, se obtiene las sumas correctas.

	1	2	3	4	5	6	7	8	9	10	
1	91 9.5%	0 0.0%	100% 0.0%								
2	0 0.0%	93 9.7%	0 0.0%	100% 0.0%							
3	0 0.0%	0 0.0%	98 10.2%	0 0.0%	100% 0.0%						
4	0 0.0%	0 0.0%	0 0.0%	99 10.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	101 10.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
6	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	96 10.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
7	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	88 9.2%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
8	0 0.0%	95 9.9%	0 0.0%	0 0.0%	100% 0.0%						
9	0 0.0%	101 10.5%	0 0.0%	100% 0.0%							
10	0 0.0%	98 10.2%	100% 0.0%								
	100% 0.0%										
	1	2	3	4	5	6	7	8	9	10	
	Target Class										

Figura 4.19. Matriz de confusión del *Modelo B predeterminado* [Impresión de pantalla de MATLAB].

Como se observa, cada celda de una matriz de confusión muestra dos valores:

- El número de elementos clasificados en esa división (parte superior).
- El porcentaje correspondiente respecto al total de individuos (parte inferior).

Además, se puede apreciar que en la diagonal de la matriz de confusión (color verde) se ubican los elementos correctamente clasificados. Las celdas fuera de la diagonal (color rojo) contienen las clasificaciones erróneas.

La lectura de la Figura 4.18 informa que:

- 945 fallas en L1 se clasificaron correctamente. Esto corresponde al 49,2% del total de 1920 fallas.
- 975 fallas ocurridas en L2, fueron clasificadas de forma acertada y representan un 50,8%.
- No existen datos clasificados incorrectamente, es decir que ninguna falla producida en L1 se clasificó como falla en L2; y ninguna falla ocurrida en L2 fue clasificada en L1. En definitiva, el 100% de las predicciones fueron correctas; y el 0%, equivocadas.

De esta manera se confirma que el *Modelo A predeterminado* es totalmente útil para establecer la direccionalidad del relé de sobrecorriente.

En la Figura 4.19, sucede prácticamente lo mismo:

- No se registra ninguna clasificación errónea.
- Todas las predicciones fueron correctas, con la siguiente distribución:
  - 91 fallas Monofásica A.
  - 93 fallas Monofásica B.
  - 98 fallas Monofásica C.
  - 99 fallas Bifásica AB.
  - 101 fallas Bifásica BC.
  - 96 fallas Bifásica CA.
  - 88 fallas Bifásica AB – Tierra.
  - 95 fallas Bifásica BC – Tierra.
  - 101 fallas Bifásica CA – Tierra.
  - 98 fallas Trifásica.

Con esto queda demostrado que el *Modelo B predeterminado* es capaz de determinar el tipo de falla ocurrida en una línea de transmisión protegida por el relé de sobrecorriente.

De manera adicional y con el propósito de apreciar mejor la lectura de datos en una matriz de confusión con clasificaciones incorrectas, se analiza el ejemplo de la Figura 4.15(b), en el Anexo 4.

En conclusión, al contar con un 100% de predicciones acertadas en el Conjunto de Validación, los Modelos Predeterminados fueron calificados como excelentes/óptimos, cumpliendo con el objetivo principal del presente trabajo que consiste en la obtención de un modelo de patrones con la capacidad de determinar la dirección de ocurrencia de una falla en una línea de transmisión, y adicionalmente, se puede conocer el tipo de falla producida en la línea protegida. Estos modelos se muestran en la Figura 4.20 con todas las características que proporciona MATLAB.

Property	Value
BinaryY	2880x1 double
CodingName	'onevsone'
Y	2880x1 double
X	2880x432 single
RowsUsed	[]
W	2880x1 double
ModelParameters	1x1 ECOCParams
NumObservations	2880
PredictorNames	1x432 cell
CategoricalPredictors	[]
ResponseName	'Y'
ExpandedPredictorNames	1x432 cell
ClassNames	[1,2]
Prior	[0.5052 0.4948]
Cost	[0 1; 1 0]
ScoreTransform	'none'
BinaryLearners	1x1 cell
BinaryLoss	'hinge'
CodingMatrix	[-1; 1]
LearnerWeights	1.0000

Property	Value
BinaryY	1440x1 double
CodingName	'onevsone'
Y	1440x1 double
X	1440x432 single
RowsUsed	[]
W	1440x1 double
ModelParameters	1x1 ECOCParams
NumObservations	1440
PredictorNames	1x432 cell
CategoricalPredictors	[]
ResponseName	'Y'
ExpandedPredictorNames	1x432 cell
ClassNames	[1;2;3;4;5;6;7;8;9;10]
Prior	[0.1035 0.1021 0.0986 0.0979 0.0965 0.1035 0.1021 0.0986 0.0979 0.0965 0.1035]
Cost	10x10 double
ScoreTransform	'none'
BinaryLearners	45x1 cell
BinaryLoss	'hinge'
CodingMatrix	10x45 double
LearnerWeights	1x45 double

**Figura 4.20.** Modelos A y B Predeterminados en MATLAB [Impresión de pantalla de MATLAB].

#### 4.5.4. RESULTADOS DE EVALUACIÓN

El Conjunto de Evaluación está compuesto de 40 señales de falla. Como se mencionó en la Sección 3.12, existen 2 objetos por cada tipo de falla.

En la Tabla 4.14, el símbolo (✓) significa que la clasificación se realizó correctamente, y (---) indica una condición no aplicable.

**Tabla 4.14.** Resultados de Evaluación [Elaboración propia].

No.	Tipo de falla	Línea fallada	Modelo A	Modelo B
1	Monofásico A (1)	L1 (anterior a la protegida)	✓	---
2	Monofásico A (2)		✓	---
3	Monofásico B (1)		✓	---
4	Monofásico B (2)		✓	---
5	Monofásico C (1)		✓	---
6	Monofásico C (2)		✓	---
7	Bifásico A (1)		✓	---
8	Bifásico A (2)		✓	---
9	Bifásico B (1)		✓	---
10	Bifásico B (2)		✓	---
11	Bifásico C (1)		✓	---
12	Bifásico C (2)		✓	---
13	Bifásico – Tierra AB (1)		✓	---
14	Bifásico – Tierra AB (2)		✓	---
15	Bifásico – Tierra BC (1)		✓	---
16	Bifásico – Tierra BC (2)		✓	---
17	Bifásico – Tierra CA (1)		✓	---
18	Bifásico – Tierra CA (2)		✓	---
19	Trifásico (1)		✓	---
20	Trifásico (2)		✓	---
21	Monofásico A (1)	L2 (protegida)	✓	✓
22	Monofásico A (2)		✓	✓
23	Monofásico B (1)		✓	✓
24	Monofásico B (2)		✓	✓
25	Monofásico C (1)		✓	✓
26	Monofásico C (2)		✓	✓
27	Bifásico A (1)		✓	✓
28	Bifásico A (2)		✓	✓
29	Bifásico B (1)		✓	✓
30	Bifásico B (2)		✓	✓
31	Bifásico C (1)		✓	✓
32	Bifásico C (2)		✓	✓
33	Bifásico – Tierra AB (1)		✓	✓
34	Bifásico – Tierra AB (2)		✓	✓
35	Bifásico – Tierra BC (1)		✓	✓
36	Bifásico – Tierra BC (2)		✓	✓
37	Bifásico – Tierra CA (1)		✓	✓
38	Bifásico – Tierra CA (2)		✓	✓
39	Trifásico (1)		✓	✓
40	Trifásico (2)		✓	✓

Se puede observar que las predicciones en todo el Conjunto de Evaluación fueron correctas, lo cual vuelve a ratificar que los *Modelos Predeterminados* cumplen a cabalidad con la función de identificar la direccionalidad y el tipo de falla ocurrida en una línea de transmisión.

## CAPÍTULO 5

### CONCLUSIONES Y RECOMENDACIONES

#### 5.1. CONCLUSIONES

- ✱ Luego de una pertinente revisión bibliográfica en cuanto a las características fundamentales de los sistemas de protección, y a la base matemática de las herramientas utilizadas: Transformada Wavelet (WT) y Máquinas de Soporte Vectorial (SVMs), se obtuvo un enfoque claro sobre los principios y las ventajas de trabajar con cada una de ellas. Las SVMs, como método de entrenamiento en el aprendizaje supervisado, muestran una gran capacidad de generalización, ya que pueden proyectar la información inicial a un espacio de características de mayor dimensión, donde un problema de difícil clasificación puede convertirse en un caso linealmente separable. Por otro lado, la WT cuenta con la versatilidad necesaria para trabajar a distintas frecuencias con diferentes resoluciones de tiempo (análisis multi-resolución) en señales no estacionarias (como cortocircuitos). Se puede resaltar la efectividad de la WT como técnica de reducción de datos, debido a que sus coeficientes reducen el número de atributos, pero mantienen íntegras las características de los datos originales. Actualmente, se realizan investigaciones en torno a las aplicaciones de la WT en sistemas de potencia, en temas como: calidad de energía, protecciones eléctricas, estimación de la demanda, entre otras.
  
- ✱ Se implementó un sistema de potencia de 7 barras en el programa ATP, con una modelación apropiada para líneas de transmisión, de forma que sea posible simular numerosas fallas. Se realizó la simulación de diez tipos de cortocircuitos tanto en la línea protegida como en la línea anterior. Estas simulaciones fueron posibles gracias a la vinculación entre los programas ATP y MATLAB. Como resultado, se recopilaron archivos *.mat*, los cuales fueron organizados en varias carpetas, obteniendo la base de datos. El sistema implementado es sencillo, pero bastante práctico para el estudio,

análisis y determinación de la dirección de una falla, producida en una de las líneas de transmisión.

- ✧ La metodología propuesta se apoyó en la aplicación de la Transformada Wavelet a las corrientes de falla, obteniendo los denominados Coeficientes de Detalle. Estos coeficientes fueron ordenados adecuadamente en una matriz, la cual ingresó como un parámetro al método de entrenamiento SVMs. Como resultado se obtuvo exitosamente el modelo de patrones deseado (Modelo A), capaz de establecer la característica de direccionalidad en un relé de sobrecorriente sin la necesidad de un TP para la señal de referencia. Además, como un aporte adicional significativo, se construyó un segundo modelo (Modelo B) que tiene la capacidad de determinar el tipo de cortocircuito, siempre que éste ocurra en la línea protegida. Ambos modelos presentan un excelente rendimiento, después de varios entrenamientos configurados con el mínimo número de errores. En efecto, los modelos proporcionan una precisión del 100% en la predicción de la dirección y tipo de falla. La principal ventaja de esta metodología es que se requiere procesar únicamente señales de corriente, evitándose así la adquisición de transformadores de voltaje.
  
- ✧ La implementación de la metodología se sintetiza en una interfaz gráfica de usuario programada en MATLAB. Se diseñó un software sencillo y amigable que permite cargar datos de corrientes de falla, así como entrenar y probar los Modelos A y B. Una observación considerable del programa desarrollado es que tanto los archivos que contienen las señales de corriente, como las matrices de entrada y salida deben ubicarse en carpetas específicas, como se explica en la Sección A2.2, para que la interfaz del programa funcione correctamente.
  
- ✧ Los dos modelos de patrones (Modelo A y Modelo B), atravesaron prácticamente por tres procesos de validación. El primero se realizó con el Conjunto de Entrenamiento (Sección 3.9). En el segundo proceso, se utilizó el Conjunto de Validación propiamente dicho (Sección 3.10), compuesto

por datos que no fueron utilizados para el entrenamiento. En ambos casos, los modelos dan óptimos resultados, con un error presente entre 0 y 2% (Tabla 4.13). En realidad, sería suficiente calificar a los modelos con las dos validaciones anteriores; sin embargo, se realiza un tercer proceso de validación con el denominado Conjunto de Evaluación, el cual está formado por un pequeño grupo de señales externas, que son corrientes de falla simuladas a diferentes longitudes de los casos anteriores. Las predicciones sobre estas señales, no presentan ningún error. Con esto, se ratifica que los Modelos A y B, basados en la WT y SVMs, muestran una alta efectividad en su desempeño y son una excelente alternativa para conocer la dirección de una falla, y el tipo de la misma.

## 5.2. RECOMENDACIONES

- ✧ En la implementación del sistema de prueba, se utilizaron varios medidores de corriente y voltaje, en principio para familiarizarse con el programa ATP y verificar el comportamiento del sistema. Esto hizo que los archivos *.mat* obtenidos como resultado de las simulaciones, tengan un gran tamaño. Sin embargo, estos archivos pueden ser reducidos si se disminuye el número de equipos de medición en el diagrama de ATP/EMPT.
- ✧ La formación de una *matriz de entrada* debe estar bien organizada para un óptimo entrenamiento. En caso de definir erróneamente los individuos y sus atributos, podrían existir inconsistencias y la imposibilidad de entrenar un modelo que presente resultados aceptables.
- ✧ El entrenamiento de modelos se realizó primero con las matrices de entrada, tal cual se obtuvieron después de aplicar la Transformada Wavelet. Debido a que el tiempo de entrenamiento fue muy extenso, es recomendable aplicar la técnica de normalización de datos, con la cual se comprobó que el tiempo se redujo considerablemente.

- ✿ No existe un método específico o una regla general para establecer el nivel de la Transformada Wavelet que se debe aplicar a un conjunto de datos. No obstante, es conveniente considerar que, dependiendo del número de datos que se desee manejar, cada nivel de descomposición arroja coeficientes wavelet que son la mitad de los datos antecesores.
  
- ✿ En este trabajo de titulación se utilizó el método de las SVMs para el entrenamiento de datos. En trabajos posteriores, se podría emplear diferentes métodos supervisados de aprendizaje (Redes Neuronales, Mezclas Infinitas o cualquier otro), con el objetivo de comparar resultados y conocer la potencialidad y aplicación de cada uno.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] R. Mason, *The Art And Science Of Protective Relaying*, 3.<sup>a</sup> ed. New York, U.S.A.: John Wiley and Sons Ltd, 1967.
- [2] P. M. Anderson, *Power System Protection*. The Institute of Electrical and Electronics Engineers, Inc., New York, U.S.A.: Wiley-IEEE Press, 1999.
- [3] G. Berterreix y M. Bonet, «Transmisión de Datos por la Red Eléctrica (PLC) en Banda Angosta», Universidad Nacional de Comahue, Argentina, 2006.
- [4] L. Brand y J. Moncada, *Protecciones de Sistemas Eléctricos*. Santiago de Chile, 1976.
- [5] C. Romero y R. Stephens, «Protección de Sistemas de Potencia». Universidad de los Andes, Venezuela, 01-jun-2002.
- [6] «IEEE Recommended Practice for Protection and Coordination of Industrial and Commercial Power Systems (IEEE Buff Book)», *IEEE Std 242-2001 Revis. IEEE Std 242-1986 IEEE Buff Book*, pp. 1-710, dic. 2001.
- [7] *Network Protection and Automation Guide*. ALSTOM GRID, 2011.
- [8] S. Ramírez Castaño, *Protección de Sistemas Eléctricos*, 1.<sup>a</sup> ed. Universidad Nacional de Colombia, Manizales, 2003.
- [9] «IEEE Standard Preferred Voltage Ratings for Alternating-Current Electrical Systems and Equipment Operating at Voltages Above 230 kV Nominal», *IEEE Std 1312-1993*, pp. 1-2, mar. 1993.
- [10] W. A. Elmore, *Protective Relaying: Theory and Applications*, 2.<sup>a</sup> ed. New York, U.S.A.: CRC Press, 2003.
- [11] M. Bedoya y F. Cadavid, «Coordinación de Relés de Sobrecorriente en Sistemas de Distribución con penetración de Generación Distribuida», Universidad Tecnológica de Pereira, Pereira, 2015.
- [12] «IEEE Standard Inverse-Time Characteristic Equations for Overcurrent Relays», *IEEE Std C37112-1996*, p. i-, 1997.
- [13] «Protecciones eléctricas en MT», *Schneider Electric: PT-071*, may-2003.

- [14] «IEEE Standard Electrical Power System Device Function Numbers, Acronyms, and Contact Designations», *IEEE Std C372-2008 Revis. IEEE Std C372-1996*, pp. 1-48, oct. 2008.
- [15] R. Saavedra Cossio, «Protección de Sistemas Eléctricos». 2009.
- [16] S. H. Horowitz y A. G. Phadke, *Power System Relaying*, 4th Edition. U.S.A.: John Wiley and Sons Ltd, 2014.
- [17] E. Pérez Fernández, «Nuevo método de detección y análisis en tiempo real de eventos en la tensión de suministro de energía eléctrica empleando un modelo combinado wavelets-filtro de Kalman extendido», Tesis Doctoral, Universidad de Cantabria, Santander, España, 2006.
- [18] N. Atlas y G. Sheifali, «Wavelet Based Techniques for Speckle Noise Reduction in Ultrasound Images», *Journal of Engineering Research and Applications*, vol. 4, feb-2014.
- [19] R. D. Reyes De Luna, «Aplicación de la Transformada Wavelet en el análisis de Calidad de la Energía», Instituto Politécnico Nacional, México, 2009.
- [20] R. Shariatinasab, M. Akbari, y B. Rahmani, «Application of Wavelet Analysis in Power Systems», en *Advances in Wavelet Theory and Their Applications in Engineering, Physics and Technology*, Editado por Dumitru Baleanu., InTech, 2012.
- [21] J. Iglesias Lorenzo, «Diseño de un relé neuronal de protección para líneas aéreas de AT con procesamiento de señal mediante la transformada wavelet», Tesis Doctoral, Departamento de Ingeniería Eléctrica, Universidad Politécnica de Cataluña, España, 2003.
- [22] A. V. Oppenheim y R. W. Schaffer, *Discrete-time signal processing*. Prentice Hall, 1989.
- [23] H. N. Díaz y R. M. de Castro, «Aplicación de la Transformada Wavelet en los Sistemas Eléctricos de Potencia», Madrid, jul. 2002.
- [24] J. Han, J. Pei, y M. Kamber, *Data Mining: Concepts and Techniques*, 3.<sup>a</sup> ed. Waltham, MA 02451, USA: Elsevier, 2011.
- [25] «Aprendizaje Automático - Universidad de Stanford», *Coursera*. [En línea]. Disponible en: <https://es.coursera.org/learn/machine-learning>.

- [26] J. J. Mora Flórez, «Localización de Faltas en Sistemas de Distribución de Energía Eléctrica usando Métodos Basados en el Modelo y Métodos de Clasificación Basados en el Conocimiento», Tesis Doctoral, Campus Montilivi, Universidad de Girona, Girona, Cataluña, España, 2006.
- [27] G. Betancourt, «Las Máquinas de Soporte Vectorial (SVMs)», *Scientia Et Technica*, Abril-2005.
- [28] E. J. Carmona Suárez, «Tutorial sobre Máquinas de Vectores Soporte (SVM)», Madrid, España, 2013.
- [29] F. E. Pérez-Yauli y G. D. Guidi-Venerdini, «MATLAB Program for Systematic Simulation over a Transmission Line in Alternative Transients Program».
- [30] L. Prikler y H. K. Høidalen, «Manual de Usuario ATPDRAW version 5.6 for Windows», nov. 2009.
- [31] F. González, «Introducción a los Sistemas de Potencia», 2008.
- [32] A. R. Herrera, «Análisis de los efectos de la variación de los parámetros del modelo de línea, de carga y de fuente, en la localización de fallas en sistemas de distribución», Universidad Tecnológica de Pereira, Pereira, 2013.
- [33] «Glossary of electric system-related terms | Learning Center | About Vermont Electric Power Company | VELCO», *Vermont Electric Power Company*. [En línea]. Disponible en: <http://www.velco.com/about/learning-center/glossary>.
- [34] «MathWorks – Makers of MATLAB and Simulink». [En línea]. Disponible en: <http://www.mathworks.com/>.
- [35] «Entrenamiento, validación y prueba | WebMining». [En línea]. Disponible en: <http://www.webmining.cl/2011/07/entrenamiento-validacion-y-prueba/>.
- [36] P. Venkatesh, B. V. Manikandan, S. C. Raja, y A. Srinivasan, *Electrical Power Systems: Analysis, Security and Deregulation*. PHI Learning Pvt. Ltd., 2012.

# ANEXOS

# ANEXO 1. MÉTODOS DE APRENDIZAJE

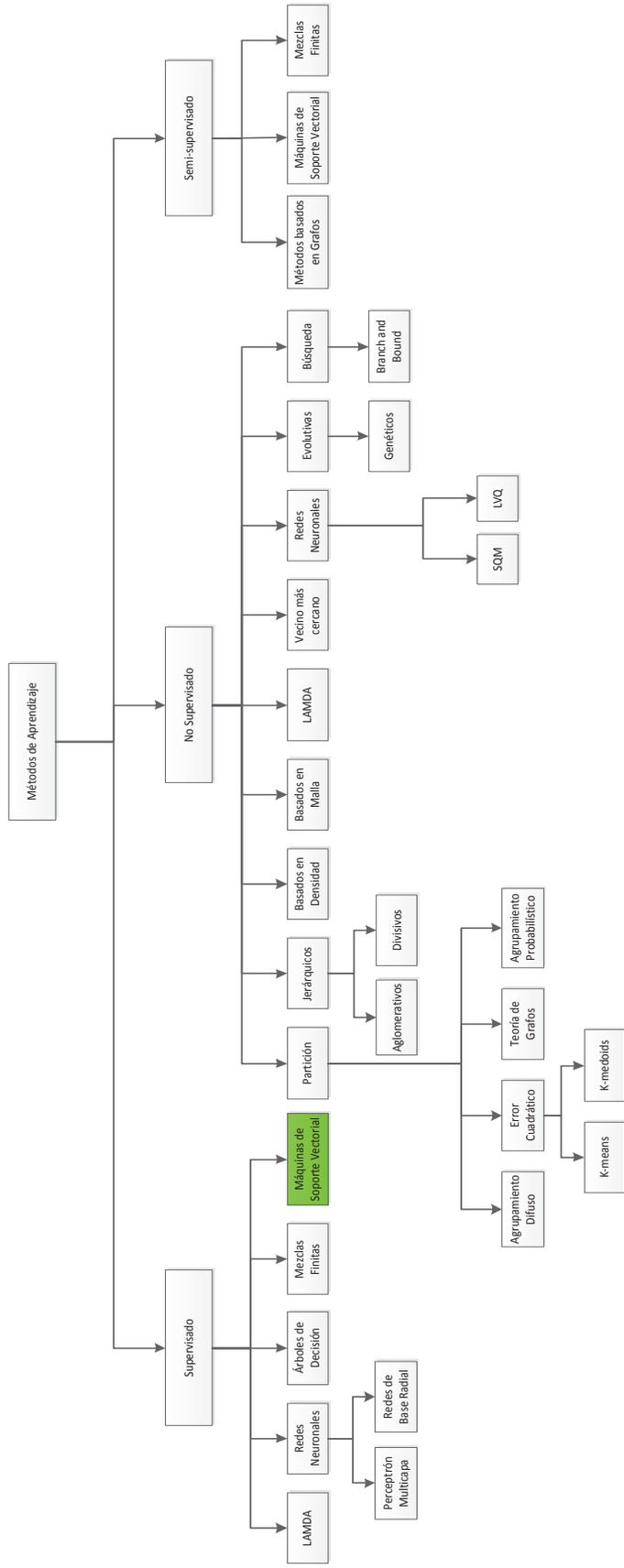


Figura A1.1. Métodos de Aprendizaje [26].

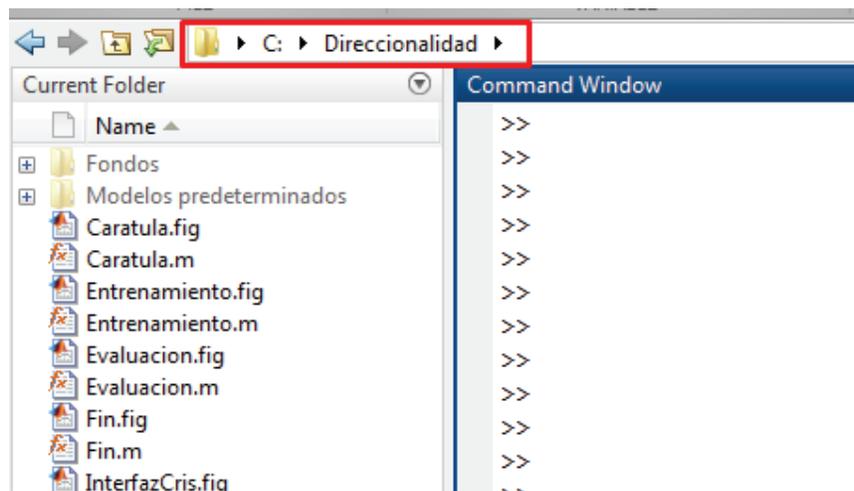
## ANEXO 2. MANUAL DE USUARIO - GUI

El programa diseñado se inicia considerando que las simulaciones de cortocircuitos fueron ejecutadas previamente (los archivos fueron almacenados en una base de datos).

El programa se desarrolló en la plataforma de MATLAB, por lo cual es necesario contar con este paquete instalado en la PC.

### A2.1. INSTALACIÓN EN PC

- Colocar en la unidad C del disco duro, la carpeta “Direccionalidad” con todos sus archivos y carpetas internas.
- Abrir el programa MATLAB.
- Mediante la barra del directorio actual, que generalmente aparece sobre las subventanas *Current Folder* y *Command Window* (ver Figura A2.1), actualizar la Carpeta de Trabajo (*Current Folder*) con el path *C:\Direccionalidad*. Esta carpeta es utilizada por MATLAB para buscar y ejecutar los ficheros.



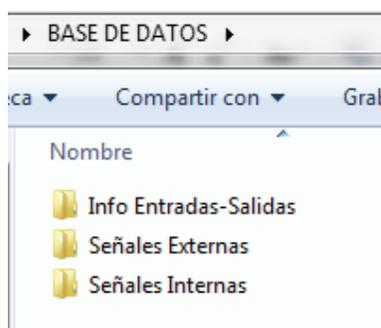
**Figura A2.1.** Ruta *C:\Direccionalidad* [Impresión de pantalla de MATLAB].

Se recomienda no modificar la posición, nombres ni contenido de las carpetas y archivos ubicados dentro de *Direccionalidad*. De otro modo, el programa tendrá errores en su ejecución.

## A2.2. ORGANIZACIÓN DE LA BASE DE DATOS

Además de la carpeta *Direccionalidad*, se requiere otra carpeta denominada *BASE DE DATOS*, donde se recopiló la información de todas las simulaciones. Puede ubicarse en cualquier dirección o disco de la PC, siempre y cuando cumpla con la siguiente estructura:

- Contiene tres subcarpetas: *Info Entradas-Salidas*, *Señales Externas* y *Señales Internas* (ver Figura A2.2).



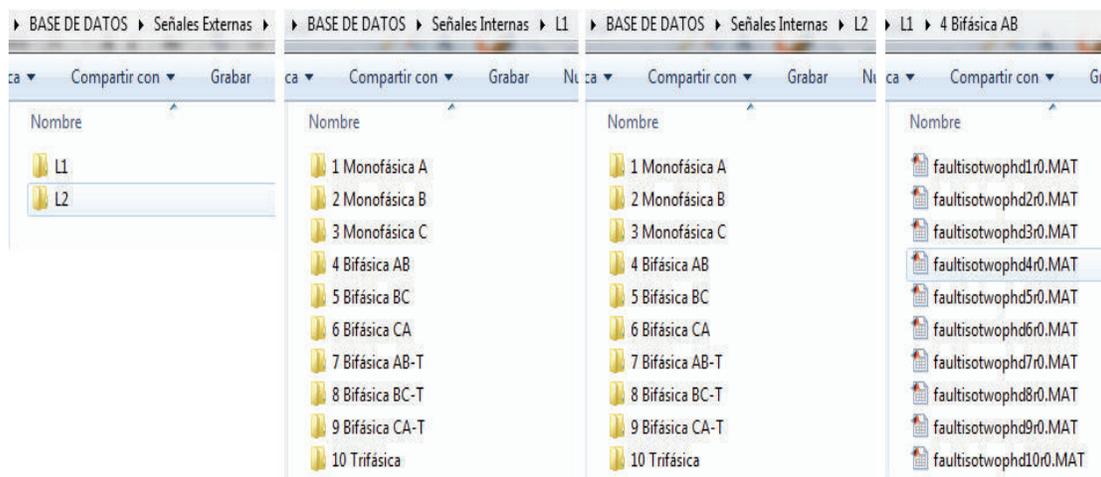
**Figura A2.2.** Contenido de carpeta *BASE DE DATOS* [Impresión de pantalla del Explorador de Windows].

- *Info Entradas-Salidas* contiene:
  - Coeficiente de Detalle para formar las **Matrices A y B**.
  - Vectores respuesta (o de salida) para ambos modelos.
- La estructura de la carpeta *Señales Externas* es la misma de la carpeta *Señales Internas*.

Cada una contiene las subcarpetas *L1* y *L2*, las cuales a su vez, contienen 10 subcarpetas conforme a los tipos de cortocircuitos: 1 *Monofásica A*, 2 *Monofásica B*, 3 *Monofásica C*, 4 *Bifásica AB*, 5 *Bifásica BC*, 6 *Bifásica CA*, 7 *Bifásica AB-T*, 8 *Bifásica BC-T*, 9 *Bifásica CA-T*, *Trifásica*.

En cada subcarpeta, se ubican 240 archivos *.mat*, que son el resultado de las simulaciones.

Esta organización se puede observar en la Figura A2.3



**Figura A2.3.** Estructura de las carpetas *Señales Externas* y *Señales Internas* [Impresión de pantalla del Explorador de Windows].

### A2.3. INICIO DEL PROGRAMA

La interfaz gráfica se compone de cinco ventanas, incluidas la presentación y el cierre de la misma.

Para iniciar con el programa, se debe escribir la palabra *Caratula* en el prompt (>>) de la ventana de comandos (*Command Window*) de MATLAB:

```
>> Caratula
```

Esta sentencia abrirá la primera ventana del programa, cuya descripción se presenta enseguida.

#### A2.3.1. Ventana CARÁTULA

Muestra únicamente la presentación del trabajo. Como página principal de la interfaz, introduce una bienvenida a la aplicación desarrollada. En la Figura A2.4 se observa que esta ventana consta de:

- Encabezado con logotipos de la Escuela Politécnica Nacional
- Imagen de fondo
- Título del trabajo de titulación
- Nombres de autor y director
- Botón “Continuar” para acceder a la siguiente ventana



**Figura A2.4.** Pantalla principal (Carátula) [Impresión de pantalla de MATLAB].

Se debe presionar el botón *CONTINUAR* para ir a la siguiente ventana de la interfaz: *Entrenamiento*.

### A2.3.2. Ventana ENTRENAMIENTO

Tiene como título *Entrenamiento de Datos* y permite al usuario dos opciones para obtener los Modelos A y B (ver Figura A2.5).

#### A2.3.2.1. Opción 1

A pulsar el botón *Modelos Predeterminados*, automáticamente se cargan los modelos guardados en la PC, los cuales fueron creados previamente realizando varios intentos para conseguir un error del 0%. Los modelos fueron almacenados en una carpeta que lleva el mismo nombre: *Modelos predeterminados*, y está ubicada dentro de la carpeta *Direccionalidad*. Para verificación del usuario, se imprime un mensaje con la opción seleccionada (ver Figura A2.6).

Se habilitan también los botones *Matriz de Confusión MODELO 1* y *Matriz de Confusión MODELO 2*, que permiten visualizar el desempeño de los modelos predeterminados. En la Figura A2.7 se pueden apreciar estas gráficas, con un 100% de aciertos en cada modelo.



Figura A2.5. Ventana de Entrenamiento de Datos [Impresión de pantalla de MATLAB].

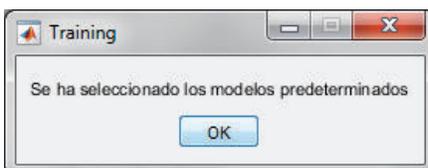


Figura A2.6. Pantalla Mensaje de comprobación - Opción 1 [Impresión de pantalla de MATLAB].

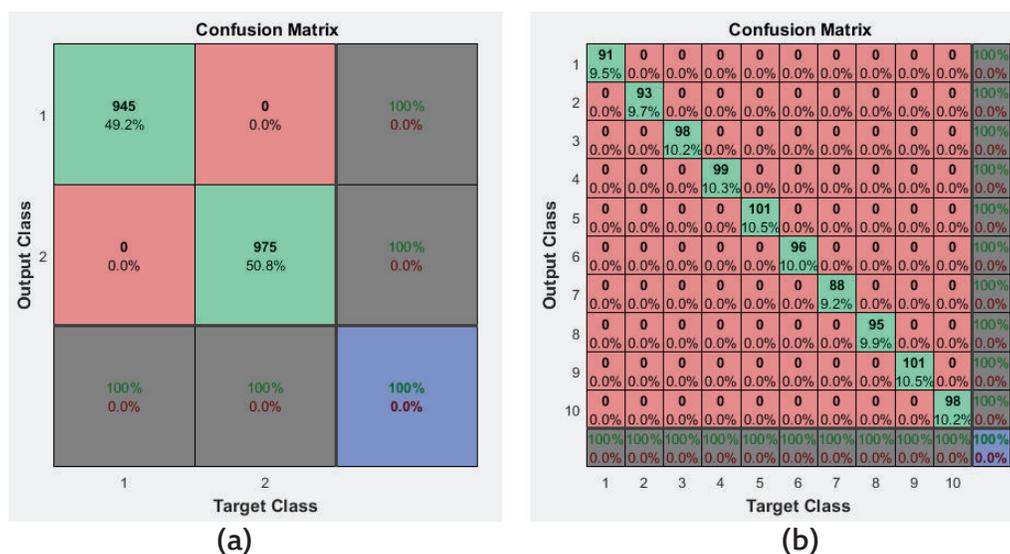


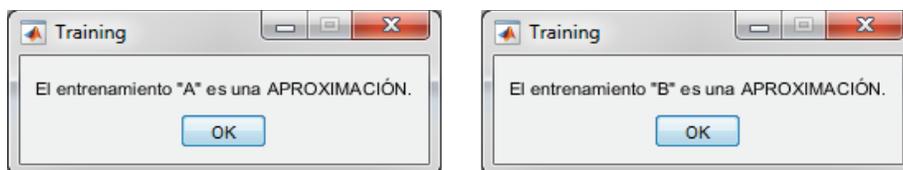
Figura A2.7. Matrices de Confusión de los Modelos Predeterminados. a) Para el Modelo A; b) Para el Modelo B [Impresión de pantalla de MATLAB].

### A2.3.2.2. Opción 2

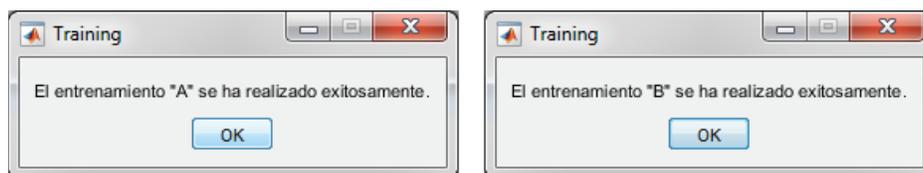
1. Pulsar el botón *Buscar* y seleccionar la carpeta *Info Entradas-Salidas* (ubicada dentro de la carpeta *BASE DE DATOS*). Dentro de la carpeta *Info Entradas-Salidas* se encuentran los Coeficientes de Detalle de todas las fallas simuladas, y los Vectores de Salida.
2. Presionar los botones *Crear Matriz A - Vector A* y *Crear Matriz B - Vector B*, para agrupar los coeficientes de detalle y formar las **Matrices A y B**, así como los **Vectores A y B**. En la parte inferior de cada botón, se visualizará la dimensión de las entradas y salidas.
3. Ingresar el porcentaje de *Entrenamiento*, y el de *Validación* será calculado automáticamente (los valores serán redondeados); caso contrario, los valores por defecto son de 60% y 40%, respectivamente.
4. En el casillero *Número máximo de errores*, escribir un número entero  $n$ . El valor por defecto es 2. Si por error, se ingresa un número con decimales, éste será redondeado.
5. Pulsar el botón *Entrenar* a fin de obtener los modelos deseados.

El proceso realizará máximo 100 entrenamientos para cada modelo. Este valor podría ser configurado únicamente en el código interno de la interfaz.

Si durante los 100 entrenamientos, no fue posible conseguir un modelo que presente máximo  $n$  errores en el Conjunto de Validación, aparece el mensaje de la Figura A2.8; caso contrario, el mensaje de la Figura A2.9.



**Figura A2.8.** Mensaje cuando Entrenamiento NO cumple con valores por defecto [Impresión de pantalla de MATLAB].



**Figura A2.9.** Mensaje cuando Entrenamiento SÍ cumple con valores por defecto [Impresión de pantalla de MATLAB].

6. Presionar los botones *Matriz de Confusión A* y *Matriz de Confusión B* para desplegar las respectivas matrices de confusión, y observar el desempeño de cada modelo.

Una vez que se consiguen los Modelos A y B, sea mediante la *Opción 1* u *Opción 2*, se habilita el botón *CONTINUAR* que permite acceder a la tercera ventana de la interfaz: *Validación*.

### A2.3.3. Ventana VALIDACIÓN

Esta ventana contiene un esquema del sistema de prueba donde se puede identificar la ubicación de las líneas *L1* y *L2*, así como la posición del relé de sobrecorriente. Existen cuatro paneles que permiten al usuario realizar la validación de los datos.

En la Figura A2.10 se ilustra la estructura completa de la ventana.

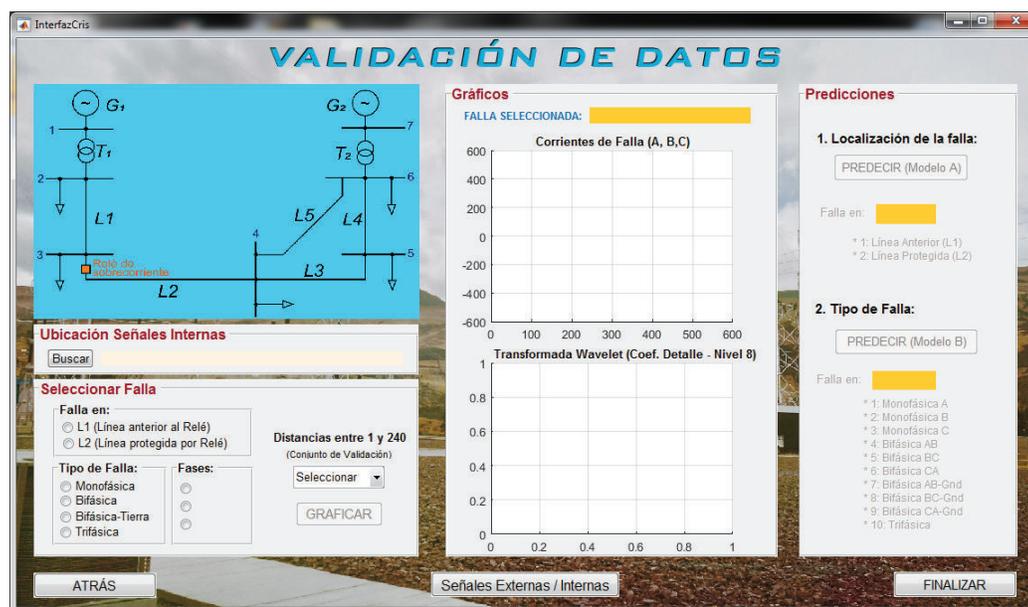


Figura A2.10. Ventana de Validación de Datos [Impresión de pantalla de MATLAB].

#### A2.3.3.1. Ubicación Señales Internas

Presionar el botón *Buscar* y seleccionar la carpeta *Señales Internas* (ubicada dentro de la carpeta *BASE DE DATOS*) que contiene todos los archivos de las fallas simuladas.

#### A2.3.3.2. *Seleccionar Falla*

Seleccionar las 4 variables que caracterizan a la falla:

- Línea en la cual se produce la falla (Recuadro *Falla en*).
- Tipo de falla (Recuadro *Tipo de Falla*).
- Las fases falladas (Recuadro *Fases*).
- La distancia a la cual ocurre la falla. (Menú desplegable *Distancias entre 1 y 240*). Contiene solo los datos del Conjunto de Validación.

Una vez especificada la falla que se quiere analizar, pulsar el botón GRAFICAR. Con esta acción se dibujan dos curvas en el siguiente panel.

#### A2.3.3.3. *Gráficos*

Este panel es básicamente informativo. Indica el tipo de falla seleccionada para el análisis, y presenta dos formas de onda:

- Corriente de Falla.
- Coeficiente de Detalle, nivel 8 de la DWT.

#### A2.3.3.4. *Predicciones*

Pulsar el botón *PREDECIR (Modelo A)* para conocer en qué línea se produjo la falla, y por tanto, la dirección de la misma.

Si la falla se ubica en la línea protegida por el relé de sobrecorriente, se habilita el botón *PREDECIR (Modelo B)*. Presionar dicho botón para conocer qué tipo de falla ocurrió en la línea de transmisión.

Cada predicción da como resultado un número entero, cuyo significado (acorde a la Tabla 4.8) se expone en la parte inferior del casillero de respuesta.

Estos resultados se comparan visualmente con la falla seleccionada para el análisis, con el objetivo de verificar que los modelos entrenados en la ventana de Entrenamiento son efectivos.

Un ejemplo de este proceso se ilustra en la Figura A2.11.

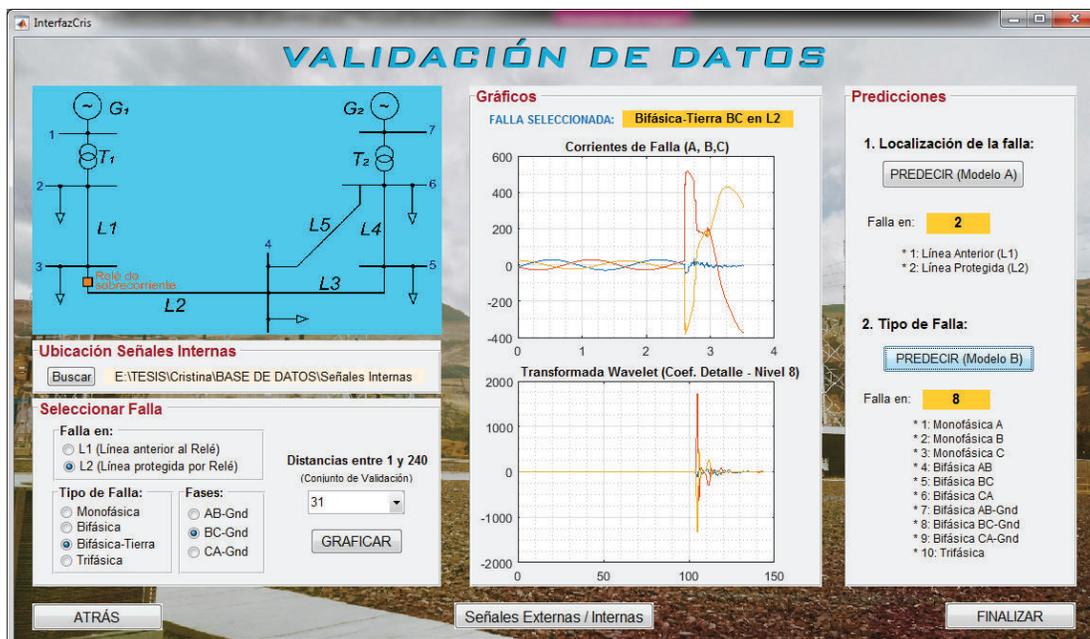


Figura A2.11. Ejemplo de Validación de modelos con una falla seleccionada [Impresión de pantalla de MATLAB].

La ventana Validación, también cuenta con un botón adicional denominado *Señales Externas/Internas*, que permite la apertura de la ventana *Evaluación*.

#### A2.3.4. Ventana EVALUACIÓN

Esta ventana contiene un menú con dos opciones (ver Figura A2.12), que se describen a continuación.

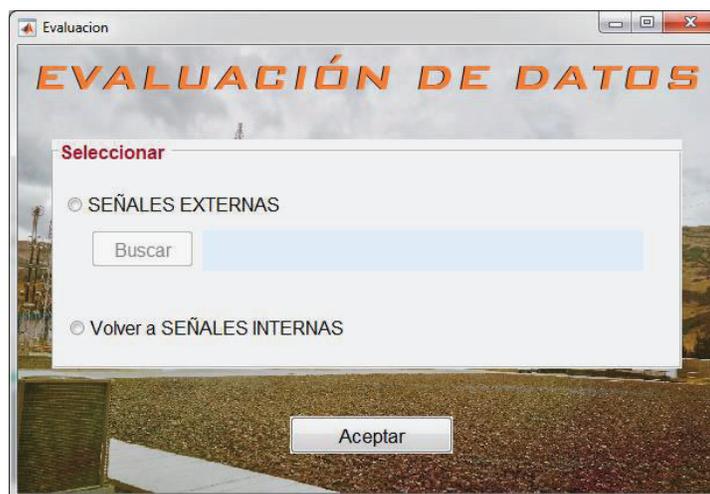


Figura A2.12. Ventana Evaluación [Impresión de pantalla de MATLAB].

#### A2.3.4.1. Señales Externas

- Seleccionar *SEÑALES EXTERNAS*, luego pulsar el botón *Buscar* y ubicar el archivo *.mat* de la falla que se quiere analizar. Los archivos *.mat* están ubicados en varias carpetas de acuerdo a la ubicación y tipo de falla. Todos estos archivos se encuentran dentro de la carpeta *Señales Externas*, que a su vez está dentro de la carpeta *BASE DE DATOS*.
- Presionar el botón *Aceptar*. Se retornará a la ventana Validación, donde el panel *Seleccionar Falla* habrá desaparecido, pero se dibujarán las formas de onda correspondientes en el panel *Gráficos*. Además, se podrá realizar la estimación de variables en el panel *Predicciones*, bajo los mismos criterios descritos en la Sección A2.3.3.4.

#### A2.3.4.2. Señales Internas

- Seleccionar la opción *Volver a SEÑALES INTERNAS*.
- Pulsar el botón *Aceptar*. Se retornará a la ventana Validación, reseteando todos los datos ingresados. El panel *Seleccionar Falla* estará nuevamente visible.

En la esquina inferior derecha de la ventana Validación, existe el botón *FINALIZAR*. Esta opción direcciona a la última ventana: *Fin*.

#### A2.3.5. Ventana FIN

Es una sencilla presentación final que muestra un texto de despedida (ver Figura A2.13). Esta ventana indica que el programa ha finalizado.



**Figura A2.13.** Ventana Fin [Impresión de pantalla de MATLAB].

### ANEXO 3. ANÁLISIS DE UNA MATRIZ DE CONFUSIÓN

Output Class	1	2	3	4	5	6	7	8	9	10	
1	30 3.1%	0 0.0%	0 0.0%	9 0.9%	0 0.0%	0 0.0%	11 1.1%	0 0.0%	0 0.0%	10 1.0%	50.0%
2	36 3.8%	94 9.8%	46 4.8%	31 3.2%	47 4.9%	45 4.7%	39 4.1%	38 4.0%	56 5.8%	33 3.4%	79.8%
3	0 0.0%	0 0.0%	43 4.5%	0 0.0%	0 0.0%	6 0.6%	0 0.0%	0 0.0%	6 0.6%	0 0.0%	78.2%
4	10 1.0%	0 0.0%	0 0.0%	27 2.8%	0 0.0%	0 0.0%	28 2.9%	0 0.0%	0 0.0%	27 2.8%	70.7%
5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	24 2.5%	0 0.0%	0 0.0%	30 3.1%	0 0.0%	0 0.0%	44.4%
6	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	24 2.5%	0 0.0%	0 0.0%	20 2.1%	0 0.0%	54.5%
7	9 0.9%	0 0.0%	0 0.0%	8 0.8%	0 0.0%	0 0.0%	10 1.0%	0 0.0%	0 0.0%	20 2.1%	78.7%
8	0 0.0%	0 0.0%	0 0.0%	0 0.0%	40 4.2%	0 0.0%	0 0.0%	28 2.9%	0 0.0%	0 0.0%	41.2%
9	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	22 2.3%	0 0.0%	0 0.0%	27 2.8%	0 0.0%	58.8%
10	6 0.6%	0 0.0%	0 0.0%	14 1.5%	0 0.0%	0 0.0%	3 0.3%	0 0.0%	0 0.0%	3 0.3%	65.1%
	33.0%	100%	48.3%	30.3%	21.6%	24.7%	11.0%	29.2%	24.8%	3.2%	32.3%
	67.0%	0.0%	51.7%	69.7%	78.4%	75.3%	89.0%	70.8%	75.2%	96.8%	67.7%
	1	2	3	4	5	6	7	8	9	10	
	Target Class										

Figura A3.1. Matriz de Confusión tomada como ejemplo.

De la Figura A3.1 se puede deducir que:

- En las celdas de color verde se observan  $30 + 94 + 43 + 27 + 24 + 24 + 10 + 28 + 27 + 3 = 310$  fallas clasificadas correctamente de la siguiente manera:
  - 30 como *Monofásicas A* (3,1% de las 960 fallas).
  - 94 como *Monofásicas B* (9,8% de las 960 fallas).
  - 43 como *Monofásicas C* (4,5% de las 960 fallas).
  - 27 como *Bifásicas AB* (2,8% de las 960 fallas).
  - 24 como *Bifásicas BC* (2,5% de las 960 fallas).
  - 24 como *Bifásicas CA* (2,5% de las 960 fallas).
  - 10 como *Bifásicas AB – Tierra* (1,0% de las 960 fallas).
  - 28 como *Bifásicas BC – Tierra* (2,9% de las 960 fallas).
  - 27 como *Bifásicas CA – Tierra* (2,8% de las 960 fallas).
  - 3 como *Trifásicas* (0,3% de las 960 fallas).

Analizando la matriz de izquierda a derecha, y de arriba hacia abajo, se identifican varias celdas de color rojo (clasificaciones erróneas). Por ejemplo:

- 9 fallas *Bifásicas AB*, 11 fallas *Bifásicas AB – Tierra*, y 10 fallas *Trifásicas* fueron clasificadas erróneamente como *Monofásicas A*.
- 36 fallas *Monofásicas A*, 46 fallas *Monofásicas C*, 31 fallas *Bifásicas AB*, 47 fallas *Bifásicas BC*, 45 fallas *Bifásicas CA*, 39 fallas *Bifásicas AB – Tierra*, 38 fallas *Bifásicas BC - Tierra*, 56 fallas *Bifásicas CA - Tierra*, y 33 fallas *Trifásicas* fueron clasificadas erróneamente como *Monofásicas B*.
- 6 fallas *Bifásicas CA*, y 6 fallas *Bifásicas CA* fueron clasificadas erróneamente como *Monofásicas C*.
- 10 fallas *Monofásicas A*, 28 fallas *Bifásicas AB – Tierra*, y 27 fallas *Trifásicas* fueron clasificadas erróneamente como *Bifásicas AB*.

De forma similar, se puede recorrer la matriz para conocer el número total de fallas que fueron mal clasificadas.

*Lectura de las celdas de color gris:* En la columna del extremo derecho, se analiza la primera y octava celda, y esto aplica para el resto de casillas.

- De las 60 predicciones realizadas como fallas *Monofásicas A*, el 50% son correctas y el 50% son erróneas.
- De las 68 predicciones realizadas como fallas *Bifásicas BC - Tierra*, el 41,2% son correctas y el 58,8% son erróneas.

A continuación, se describe la tercera y octava celda de la fila inferior de color gris. La forma de lectura aplica para cualquier celda de la fila:

- De los 89 casos de fallas *Monofásicas C*, el 48,3% fueron correctamente clasificadas como *Monofásicas C*, y el 51,7% se predijo incorrectamente.
- De los 96 casos de fallas *Bifásicas BC - Tierra*, el 29,2% fueron correctamente clasificadas como *Bifásicas BC - Tierra*, y el 70,8% se predijo incorrectamente.

En conclusión: el 32,3% de las predicciones son correctas, y el 67,7% son clasificaciones erróneas.