



Facultad de Ingeniería en Electricidad y Computación

Diseño de protecciones para sistemas aislados de tierra usando algoritmo de
detección de fallas con IA

INGE-2824

Proyecto De Materia Integradora

Previo a la adquisición del Grado Académico de:

Ingeniero Eléctrico

Realizado por:

Isabella Angelina Santana Barzallo

Miguel Ángel Pérez Guerra

Guayaquil - Ecuador

Año: 2025

Dedicatoria

Este proyecto lo dedico a todas las personas que me apoyaron y no dejaron que decaiga frente a este nuevo reto que al inicio parecía imposible. Este no es el fin sino el comienzo de una nueva etapa.

Isabella Santana | Miguel Pérez

Agradecimientos

Mi más sincero agradecimiento a Dios, mis papás, Janina y Marcos, mis hermanos Marco y Geanella, en especial a Lily por acompañarme y enseñarme en esta etapa que existirán adversidades, pero me recuerdan que no estoy sola y ustedes estarán siempre para apoyarme. A los amigos que hice a lo largo de mi vida académica e hicieron divertida mi experiencia universitaria y a los maestros que tuve que me hicieron descubrir nuevas metas.

Ing. Isabella Santana

Con profunda gratitud, agradezco primero a Dios y a mis padres, pilares de mi vida. Su amor, sacrificio y enseñanzas sobre el esfuerzo son el cimiento de todo lo que logro. Les debo todo por haber creído en mí siempre.

Un especial agradecimiento para mis tíos Alfredo y José Guerra, por sus sabios consejos y apoyo invaluable.

A mis primos William, Clelia, Carol, Dennise, Kelly y Naomi Cuenca, por su preocupación hacia mí y su apoyo constante.

Reconozco también a mis maestros, en especial al Msc. Otto Alvarado y a mi tutor, el Msc. Iván Endara, por su guía inspiradora.

No olvido a quienes, aunque ya no están, sembraron en mí la semilla que hoy florece.

A todos, gracias infinitas.

Ing. Miguel Pérez

Declaración Expresa

Nosotros Isabella Angelina Santana Barzallo y Miguel Ángel Pérez Guerra acordamos y reconocemos que:

La titularidad de los derechos patrimoniales de derechos de autor del proyecto de graduación corresponderá al autor o autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores.

La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por mí/nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que me/nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de mi/nuestra innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique a los autores que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, 31 de mayo del 2025.

Isabella Angelina
Santana Barzallo

Miguel Ángel
Pérez Guerra

Evaluadores

Mgtr. Patricia Isabel Pasmay Bohórquez

Profesor de Materia

Msc. David Iván Endara Vélez

Tutor de proyecto

Resumen

Se plantea un esquema inteligente que, mediante aprendizaje automático, reconoce en milisegundos las fallas de tierra que suelen pasar inadvertidas en redes con neutro aislado; se persigue reducir el tiempo de despeje y aumentar la continuidad operativa, formulándose la hipótesis de que un clasificador liviano puede superar a las lógicas fijas sin añadir hardware costoso. Para comprobarla, se modeló en MATLAB-Simulink un sistema IEEE-14 barras; se simularon más de 30 000 escenarios con variaciones de carga y cortocircuitos aleatorios, y las corrientes primarias fueron escaladas a valores secundarios mediante relaciones típicas de transformadores de corriente. Las magnitudes RMS, picos y componentes de secuencia fueron extraídas y, con ellas, se entrenó un bosque aleatorio de 300 árboles; posteriormente el modelo fue validado con archivos inéditos. Se logró clasificar fallas monofásicas, bifásicas y trifásicas con un índice de confianza superior al 0,96 y una latencia de 16 a 25 ms, mientras los disparos indebidos se mantuvieron por debajo del 2 %. Se concluye que la propuesta ofrece una detección veloz, adaptable y económicamente ventajosa para redes críticas que operan sin neutro conectado.

Palabras Clave: Sistema aislado, Transitorios, Bosque aleatorio, Simulink, transformadores

Abstract

An intelligent scheme is proposed that uses machine learning to recognize ground faults in milliseconds, which often go unnoticed in isolated neutral networks. The aim is to reduce clearance time and increase operational continuity, based on the hypothesis that a lightweight classifier can outperform fixed logic without adding expensive hardware. To test this hypothesis, an IEEE 14-bus system was modeled in MATLAB-Simulink. More than 30,000 scenarios were simulated with load variations and random short circuits, and the primary currents were scaled to secondary values using typical current transformer ratios. The RMS magnitudes, peaks, and sequence components were extracted and used to train a random forest of 300 trees; the model was then validated with unpublished files. Single-phase, two-phase, and three-phase faults were classified with a confidence index greater than 0.96 and a latency of 16 to 25 ms, while false trips remained below 2%. It is concluded that the proposal offers fast, adaptable, and cost-effective detection for critical networks operating without a connected neutral.

Keywords: Isolated system, Transients, Random Forest, Simulink, Transformers

Índice general

Contenido	
Evaluadores-----	6
Resumen -----	7
<i>Abstract</i> -----	8
Índice general -----	9
Abreviaturas -----	11
Índice de figuras -----	12
Índice de Tablas-----	13
Capítulo 1 -----	14
1.1 Introducción-----	15
1.2 Descripción Del Problema-----	15
1.3 Justificación Del Problema -----	15
1.4 Objetivos	16
1.4.1 Objetivo General	16
1.4.2 Objetivos Específicos.....	16
1.5 Marco Teórico	17
1.5.1 Características De Las Señales De Falla En Sistemas De Potencia	17
1.5.2 Protecciones Tradicionales Y Esquemas Adaptativos.....	17
1.5.3 Dinámica Del Sistema Aislado.....	18
1.5.4 Aprendizaje Supervisado En Protecciones De Sistemas Con Neutro Aislado A Tierra.....	19
1.5.5 Comparativa De Modelos De IA Para La Detención De Fallas	20
1.5.6 Extracción De Características Y Construcción Del Dataset Supervisado	22
1.5.7 Árbol De Decisión (Random Forest).....	23
1.5.8 Criterios Matemáticos del Árbol De Decisión (Random Forest)	25
Capítulo 2 -----	27
2. Metodología -----	28
2.1 Diagrama De Flujo	28
2.2 Diseño Del Sistema De Simulación	28

2.3 Generación De La Base De Datos	31
2.3.1 Corrientes_(Normal/Load)(NumeroDeBarra)_(Falla).....	32
2.3.2 Datos_Corrientes_(Normal/Load)(NumeroDeBarra)_(Falla).....	32
2.4 Conversión De Corrientes Primarias A Valores Secundarios De Tc.....	33
2.5 Extracción De Características Y Construcción Del Dataset Supervisado	34
2.6 Entrenamiento	35
2.6.1 Registro Sin Falla.....	35
2.6.2 Registro Con Falla	37
2.6 Prueba.....	38
2.6.1 Archivos Con Fallas En Barras Distintas	38
2.6.2 Validación.....	39
2.6.3 Tiempo De Duración De La Falla	39
Capítulo 3 -----	43
3. Resultados -----	44
3.1 Tablas De Porcentajes de confianza.....	44
3.2 Gráficas De Corriente	45
3.3 Análisis Del Archivo Con Random Forest.....	47
3.4 Análisis Del Modelo En Comparación A Otra IA	48
3.5 Presupuesto.....	49
Capítulo 4 -----	52
4.1 Conclusiones y recomendaciones -----	53
4.1.1 Conclusiones	53
4.1.2 Recomendaciones	54
Referencias -----	55

Abreviaturas

AG	Falla de la Fase A a Tierra (ejemplo de nomenclatura de fallas)
AI / IA	Inteligencia Artificial
CNN	Redes Neuronales Convolucionales (Convolutional Neural Networks)
CSV	Valores Separados por Comas (Comma-Separated Values)
DFT	Transformada Discreta de Fourier (Discrete Fourier Transform)
DRL	Aprendizaje por Refuerzo Profundo (Deep Reinforcement Learning)
DSP	Procesador Digital de Señales (Digital Signal Processor)
DWT	Transformada Wavelet Discreta (Discrete Wavelet Transform)
FIR	Respuesta al Impulso Finito (Finite Impulse Response)
IEEE	Instituto de Ingenieros Eléctricos y Electrónicos
IT	Neutro Aislado a Tierra (del francés Isolé-Terre)
MLP	Perceptrón Multicapa (Multilayer Perceptron)
RA	Red Aislada (por ejemplo, generación distribuida)
RF	Bosques Aleatorios (Random Forest)
RMS	Valor Cuadrático Medio (Root Mean Square)
RNA	Redes Neuronales Artificiales
SVM	Máquinas de Vectores de Soporte (Support Vector Machine)
TAF	Tiempo Antes de la Falla
TC	Transformador de Corriente
TPF	Tiempo de Finalización de la Falla
TC	Transformador de Corriente
TPF	Tiempo de Finalización de la Falla
VLL	Voltaje entre Líneas (Line-to-Line Voltage)

Índice de figuras

FIGURA 1.1 DIAGRAMA DE UN SISTEMA DE TIERRA AISLADA Y CORRIENTE CAPACITIVA A CAUSA DE UNA FALLA A TIERRA.	17
FIGURA 1.2 RELÉ NUMÉRICO SEL-751A, USADO COMO PROTECCIÓN DE ALIMENTADOR (FEEDER) EN REDES DE MEDIA Y ALTA TENSIÓN.	18
FIGURA 1.3 COMPARACIÓN DE LA RESPUESTA ANTE UNA FALLA: GENERADOR CONVENCIONAL VS. INVERSOR RÁPIDO VS. INVERSOR ULTRA-RÁPIDO.	19
FIGURA 1.4 REPRESENTACIÓN VISUAL DEL ALGORITMO DE BOSQUE ALEATORIO Y SU MODO DE OPERACIÓN.	24
FIGURA 2.1 FLUJO GENERAL DEL PROCESO METODOLÓGICO IMPLEMENTADO PARA LA DETECCIÓN DE FALLAS EN EL SISTEMA AISLADO DE TIERRA.	28
FIGURA 2.2 IMAGEN REPRESENTATIVA DEL MODELO DE 14 BARRAS PROPUESTO POR LA IEEE.	29
FIGURA 2.3 BLOQUE THREE-PHASE FAULT USADO COMO CATALIZADOR DE FALLA PARA EL ESTUDIO.	30
FIGURA 2.4 REPRESENTACIÓN DEL SISTEMA THREE-PHASE FAULT USADO PARA UNA DETERMINAR LAS FALLAS EN EL SISTEMA.	31
FIGURA 2.5 COMANDO PARA IDENTIFICAR FALLA EN PRUEBA1.	35
FIGURA 2.6 RESULTADO DE PRUEBA1.	36
FIGURA 2.7 COMANDO PARA IDENTIFICAR FALLA EN PRUEBA2.	37
FIGURA 2.8 RESULTADO DE PRUEBA2.	37
FIGURA 2.9 OBTENCIÓN DE DATOS DE LA BARRA 9 Y 14.	38
FIGURA 2.10 RESULTADO DEL TIPO DE FALLA.	39
FIGURA 2.11 SIMULACIÓN DE UNA FALLA TRIFÁSICA Y EL PICK UP DE LA CORRIENTE, OBTENIDO POR MATLAB Y SIMULINK EN LA CARGA 2 DEL SISTEMA CONECTADA A LA BARRA 2.	40
FIGURA 2.12 DATOS OBTENIDOS POR LA IA, DE SU CORRIENTE DE FALLA (VALORES SECUNDARIOS) Y TIEMPO DE OCURRENCIA.	41
FIGURA 2.13 DIFERENCIAS DE LOS TIEMPOS DE DONDE INICIA LA FALLA Y EL TIEMPO EN EL CUAL LA IA DETECTA DICHA EVENTUALIDAD COMO FALLA, CON UN TIEMPO MENOR A LOS 10MS.	41
FIGURA 2.14 SEÑAL DE DISPARO QUE SIMULARÍA EL VALOR ENVIADO POR EL RELÉ HACIA UN INTERRUPTOR IDEAL QUE DESPEJARÍA LA FALLA.	42
FIGURA 3.1 GRÁFICAS DE CORRIENTES DEL ARCHIVO: CORRIENTES_IEEE14.CSV.	46
FIGURA 3.2 GRÁFICAS DE CORRIENTES DEL ARCHIVO: CORRIENTES_LOAD2.CSV.	46

Índice de Tablas

TABLA 1.1 COMPARATIVA DE LOS MODELOS EMPLEADOS PARA DETECCIÓN DE FALLAS EN SISTEMAS DE POTENCIA.....	21
TABLA 1.2 COMPARACIÓN DEL MÉTODO TRADICIONAL PARA LA PROTECCIÓN DE SISTEMAS EN EVENTOS DE FALLA VS EL MÉTODO PROPUESTO.	22
TABLA 2.1 PARÁMETROS DE POTENCIA EMPLEADOS PARA LA CARGA CONECTADA EN CADA BARRA, VALORES EN MEGA.	29
TABLA 2.2 VALORES SETEADOS EN LOS GENERADORES UBICADOS EN EL SISTEMA DE POTENCIA DE LA IEEE 14 BUS.	30
TABLA 2.3 VALORES DE FALLAS PRESENTE EN LOS CSV DE CORRIENTES.	32
TABLA 2.4 DATOS DE CADA SIMULACIÓN ACORDE A LA FALLA.	33
TABLA 2.5 VALORES DE LA SIMULACIÓN CON LA CORRIENTE REFLEJADA COMO SI PROVIENE DE UN TC.	33
TABLA 2.6 RELACIÓN DE LOS TRANSFORMADORES DE CORRIENTE (TC) EMPLEADOS EN EL ESTUDIO DE FALLAS, QUE CONECTAN A LA BARRA 2.	34
TABLA 2.7 MÉTRICAS USADAS COMO FEATURES DEL SISTEMA CON EL QUE SE ENTRENÓ LA IA.	34
TABLA 3.1 PRUEBA GENERALIZADA DE EVENTOS DE FALLA DE UN SISTEMA SIN CONSIDERAR OPERACIÓN NOMINAL.	444
TABLA 3.2 PRUEBA GENERALIZADA DE EVENTOS DE FALLA DE UN SISTEMA CONSIDERANDO OPERACIÓN NOMINAL.	44
TABLA 3.3 COMPARACIÓN DEL MODELO USADO CON OTRA IA	48
TABLA 3.4 PRESUPUESTO GENERAL.	50

Capítulo 1

1.1 Introducción

En los sistemas eléctricos con neutro aislado, la detección de fallas de una sola fase es un desafío debido a las bajas corrientes [1], un problema agravado por los sistemas que limitan la corriente. Para solucionar esto, se ha desarrollado un nuevo método de protección. Este sistema utiliza transformadores de corriente para medir las formas de onda en tiempo real. Luego, por medio de un algoritmo de inteligencia artificial. En donde el algoritmo extrae características clave de la curva mediante la toma de datos.

Con esta información, el modelo de IA buscará detectar fallas en un menor tiempo medido en milisegundos y enviar una orden de desconexión. Las simulaciones, realizadas por medio de software de un modelo modificado del ya conocido IEEE-14 barras.

1.2 Descripción Del Problema

En los sistemas con neutro aislado, una falla monofásica a tierra genera corrientes capacitivas de baja magnitud, pocos amperios [2], tal que los relés de sobre corriente tradicionales la detecten. Cuando el sistema de protecciones clásico no detecta el defecto, la aislación se degrada [3].

La tesis propone una solución ligera que aprovecha la señal ya disponible del transformador de corriente (TC), procesando cada ciclo por medio de IA, capaz de decidir de forma rápida si la forma de onda corresponde a una falla incipiente y, de ser así, enviar una orden de apertura al interruptor (CB). El enfoque se apoya en tres hechos: los TC suministran datos suficientes para extraer patrones sutiles; los relés digitales modernos pueden ejecutar algoritmos simples de IA si se limita el número de características; y validar el esquema en un modelo IEEE-14 modificado con registros locales haciendo así su adaptación a la realidad.

1.3 Justificación Del Problema

Los sistemas de protección tradicionales de sistemas aislados a tierra utilizan equipos como relés electromecánicos o digitales con configuraciones estáticas que no se adaptan al comportamiento dinámico del sistema, siendo esta rigidez operativa lo que limita su capacidad para detectar fallas en condiciones cambiantes, especialmente en redes con generación distribuida. Por eso, la integración de algoritmos basados en inteligencia artificial en esquemas de protección representa una alternativa adaptable pues aprenden del comportamiento histórico del sistema, mejoran la clasificación de eventos y reducen el tiempo de operación de los relés, incrementando así la seguridad y confiabilidad de la red [4]. Al tiempo, un sistema de protección que integre IA,

conectado a transformadores de corriente y a un interruptor automático, podría operar con menor infraestructura física ofreciendo mejor relación costo-beneficio al minimizar el tiempo de despeje de fallas, pudiendo evitar desconexiones innecesarias y reducir las pérdidas económicas asociadas a interrupciones no planificadas que inciden directamente en la calidad del servicio eléctrico y en la continuidad del suministro [5].

1.4 Objetivos

1.4.1 Objetivo General

Desarrollar un diseño de protección inteligente en sistemas eléctricos aislados a tierra con generación distribuida, mediante la integración de técnicas de inteligencia artificial, logrando la detección y mitigación de fallas eléctricas de manera oportuna y eficiente.

1.4.2 Objetivos Específicos

1. Generar una base de datos utilizando el sistema de barras IEEE de 14 barras para que el algoritmo aprenda patrones característicos de operación normal y en falla.
2. Simular múltiples escenarios de fallas eléctricas mediante programas de simulación computacional para el desarrollo de estrategias de protección.
3. Crear un algoritmo que detecte anomalías empleando técnicas procesamiento de datos recopilados para el desarrollo de un modelo que detecte eventos anómalos.

1.5 Marco Teórico

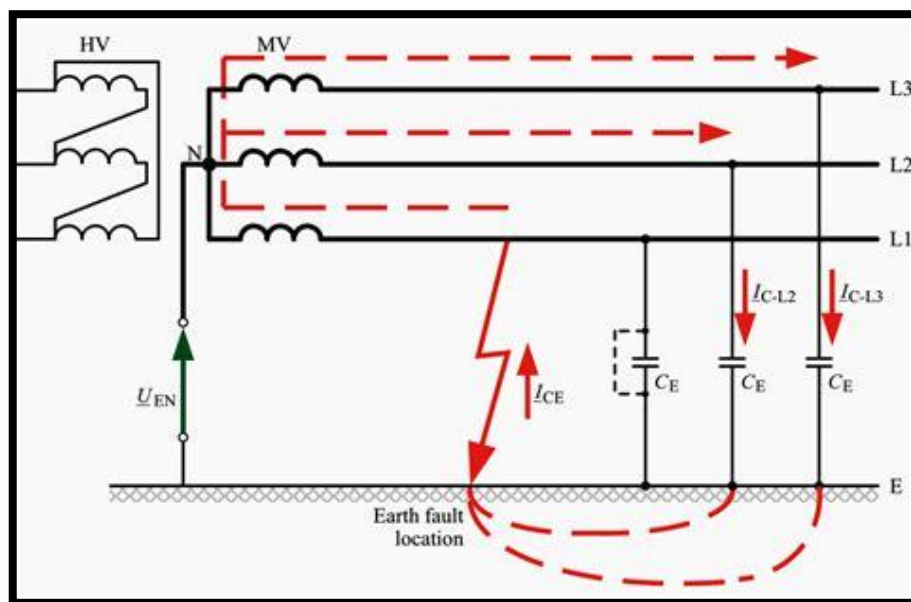
1.5.1 Características De Las Señales De Falla En Sistemas De Potencia

En redes con neutro sólidamente aterrizado y generadores síncronos, las fallas entre fases generan corrientes de cortocircuito de grandes magnitudes, que puede conllevar a caídas abruptas de tensión, en donde las fallas asimétricas inducen componentes de secuencia negativa y cero fácilmente detectables. En contraste, los esquemas en donde el neutro se encuentra aislado limitan su corriente a unos cuantos amperios donde pueden considerarse como valores normales [6].

Es así como la ausencia de una referencia a tierra desplaza los voltajes de fase: la fase en falla desciende a potencial de tierra, mientras las restantes se elevan hacia $V_{LL}\sqrt{3}$. La corriente resultante es la suma de las capacitancias distribuidas y suele estar por debajo de 10 A en redes de media tensión [7].

Figura 1.1

Diagrama de un sistema de tierra aislada y corriente capacitiva a causa de una falla a tierra.



1.5.2 Protecciones Tradicionales Y Esquemas Adaptativos

Los sistemas de distribución tradicionales coordinan relés temporizados de sobre corriente desde el extremo de alimentación a la carga. Estos dispositivos suponen flujos unidireccionales y corrientes de falla significativamente mayores que la carga. La red aislada (RA) puede llegar a modificar la direccionalidad y reduce la corriente de cortocircuito, provocando descoordinación: un relé próximo a la subestación puede captar corriente inversa procedente de la RA y disparar

incorrectamente. El fenómeno de asignación a ciegas que se produce cuando las corrientes no alcanzan los umbrales de disparo establecidos [8].

Figura 1.2

Relé numérico Sel-751A.



Nota: Usado como protección de alimentador (Feeder) en redes de media y alta tensión.

Los esquemas adaptativos introducen lógicas que ajustan umbrales y retardos conforme al estado de la red. Algunas soluciones centralizadas recopilan información de interruptores y flujos para actualizar parámetros de los relés en tiempo real, mientras otras incluyen inteligencia local que combina criterios de corriente, subtensión rápida y dirección. La estrategia dual corriente más tensión mejora la detección de faltas en redes aisladas y suprime falsas actuaciones al considerar el hundimiento de tensión como disparador adicional.

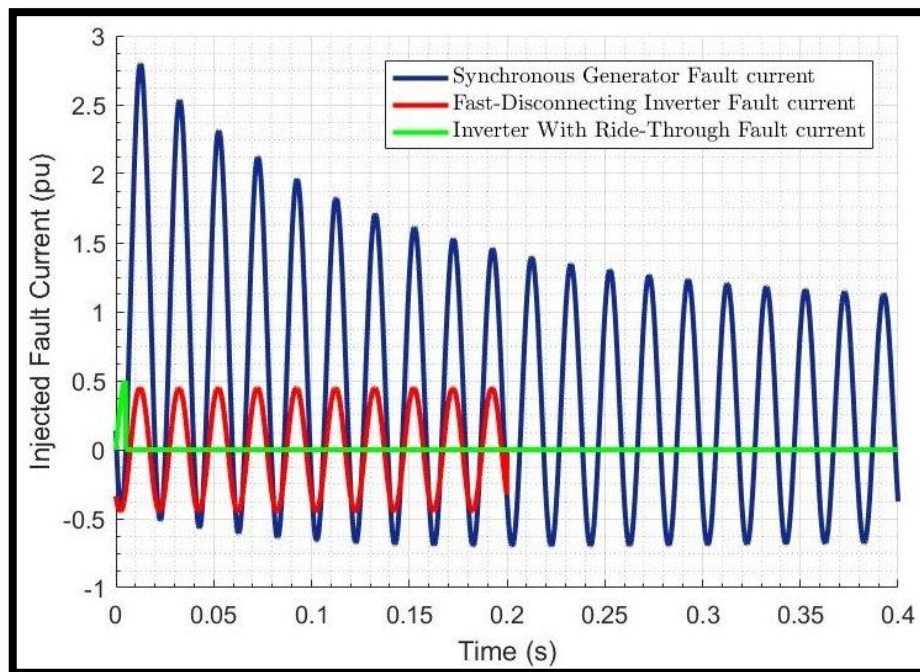
1.5.3 Dinámica Del Sistema Aislado

En redes eléctricas con neutro aislado, la detección de fallas a tierra de alta impedancia o transitorias es un reto clásico para la protección convencional. Los modelos de IA supervisados ya sean RNA o Árboles de Decisión (Random Forest) han surgido como herramientas prometedoras, pues pueden modelar relaciones no lineales complejas entre variables de señal (corriente de secuencia cero, tensión residual, ángulo de fase, transitorios, etc.). En particular, arquitecturas como el Perceptrón Multicapa (MLP) han demostrado buena capacidad para clasificar eventos eléctricos diversos: Distinguen fallas reales de conmutaciones de carga u otros transitorios benignos mediante vectores de características extraídos, por ejemplo, con transformadas wavelet o análisis espectral. Esta capacidad de aprendizaje no lineal y generalización convierte a las RNA o los RF en candidatas ideales para esquemas de protección adaptativa, incluso en entornos

industriales ruidosos con sensores inteligentes. Por ejemplo, estudios recientes integran la Transformada Wavelet Discreta (DWT) con redes neuronales para extraer rasgos tempo-espaciales relevantes de la señal y luego clasificar la falla [9].

Figura 1.3

Comparación de la respuesta ante una falla: generador convencional vs. inversor rápido vs. inversor ultrarrápido.



1.5.4 Aprendizaje Supervisado En Protecciones De Sistemas Con Neutro Aislado A Tierra

El aprendizaje supervisado es el marco más utilizado para entrenar modelos predictivos en sistemas de protección. Se basa en el uso de datos históricos, donde cada muestra está asociada a una salida deseada, como la activación o no de una función de protección. En sistemas con neutro aislado, este enfoque es crucial para clasificar fallas a tierra de alta impedancia, identificar la fase involucrada o incluso anticipar condiciones peligrosas antes de que se activen los elementos de corte.

Durante el entrenamiento, el modelo ajusta sus parámetros para minimizar una función de error entre la salida esperada y la salida real. Algoritmos como retro propagación del error (backpropagation) junto con descenso de gradiente permiten que modelos como redes neuronales, bosques aleatorios o Super Vector Machine (SVM) adquieran una capacidad de decisión confiable [10].

En el caso específico de sistemas aislados a tierra, el aprendizaje supervisado permite mejorar la sensibilidad sin sacrificar la selectividad, ya que puede aprender a distinguir patrones complejos de corriente y tensión que no se observan claramente con umbrales fijos.

1.5.5 Comparativa De Modelos De IA Para La Detención De Fallas

La identificación de fallas en sistemas eléctricos con aislación a tierra ha experimentado un avance notable debido a la inclusión de la IA en los últimos años. Se han incorporado múltiples modelos que ayuden a optimizar la exactitud, rapidez y fiabilidad de los sistemas de protección. Cada modelo posee distintas características que relacionan la capacidad de aprendizaje, complejidad computacional, interpretabilidad, factores que condicionan la viabilidad de su implementación en condiciones operacionales del día a día.

En la Tabla 1.1, se presenta una comparativa que evalúa los principales modelos de IA empleados en la clasificación y diagnóstico de fallas eléctricas, resaltando tanto las ventajas como las limitaciones.

Tabla 1.1

Comparativa de los modelos empleados para detección de fallas en sistemas de potencia.

Modelos de IA		
Enfoque	Ventaja	Desventaja
Árboles de decisión/ Bosques aleatorios	Fácil de interpretar, robustos y puede estimar importancia de variables, mejorando la precisión y evita sobreajuste [11].	Puede ser lento con grandes datasets.
Máquina de Vector de Soporte (SVM)	Alta precisión para clasificar fallas y buena discriminación con pocos datos.	El modelo resultante no es tan interpretable, requiero mucho tiempo y memoria [11].
Redes Neuronales (Perceptrón Multicapa-MLP)	Capaz de modelar relaciones no lineales en señales de voltaje y corriente. También, detecta patrones sutiles en mediciones [9].	Requieren una gran base de datos de fallas etiquetadas para entrenamiento. Sin embargo, posee diagnósticos reducidos.
Redes Neuronales Convolucionales (CNN)	Extrae automáticamente características relevantes y detecta eventos de falla.	Requieren gran dataset y cómputo. Mayor complejidad en implementación y afinación para tiempo real.
Aprendizaje por Refuerzo Profundo (DRL)	Aprende disparo sin datos etiquetados, optimiza estrategias de aislamiento y reconexión automática [12].	Entrenamiento complejo, impredecible fuera de escenarios simulados.
Transformada Wavelet (DWT/CWT)	Excelente preprocesamiento y permite localizar en tiempo-frecuencia las fallas a tierra.	Sensible a ruido y requiere selección cuidadosa de Wavelet

Tabla 1.2

Comparación del método tradicional para la protección de sistemas en eventos de falla vs el método propuesto.

Protección de sistemas aislado: método tradicional vs tradicional		
Criterio	Método Tradicional	Esquema con IA
Sensibilidad	Limitada por la baja magnitud de corriente de falla y como los umbrales suelen quedar por encima, la falla pasa inadvertida [9].	Alta sensibilidad, detecta fallas de alta impedancia incluso bajo niveles bajos de corriente residual.
Selectividad	Dificultad para identificar el alimentador exacto en fallas monofásicas.	Algoritmos de IA como redes neuronales permiten identificar con precisión la ubicación de la falla.
Rapidez	Retardo inherente por el uso de relés de sobretensión o detección por bobina de alta impedancia.	Alta velocidad de respuesta (en ms) gracias a clasificación en tiempo real de las señales de corriente por medio de los TC y su postprocesado.
Confiabilidad	Afectada por perturbaciones y condiciones de carga.	Algoritmos autoajustables permiten mantener rendimiento bajo condiciones variables.

1.5.6 Extracción De Características Y Construcción Del Dataset Supervisado

La extracción de características es el proceso de transformar datos brutos en información significativa que pueda ser empleada por modelos de IA [13]. Esto implica seleccionar o calcular atributos relevantes (llamados características o features) a partir de las señales o datos originales, de modo que se reduzca la complejidad de los datos sin perder la información esencial. Una buena extracción de características permite aislar la información más importante y descartar el ruido o datos redundantes, facilitando así el entrenamiento del modelo con mejores resultados. [13]

En el contexto de detección de fallas eléctricas (como en sistemas de potencia), la extracción de características presenta retos adicionales debido a la gran cantidad de información involucrada y a la variabilidad de las condiciones de falla. Se requiere analizar múltiples parámetros y señales, lo cual hace compleja la identificación de los atributos más relevantes [14]. Por ello, suele ser necesario aplicar criterios de ingeniería de características basados en conocimiento del dominio eléctrico, para resumir cada evento de falla en un conjunto de métricas representativas. Entre las métricas más comunes empleadas como características en señales eléctricas se encuentran, por ejemplo, el valor eficaz (RMS), el valor pico y el valor medio de la corriente o voltaje en el intervalo de análisis.

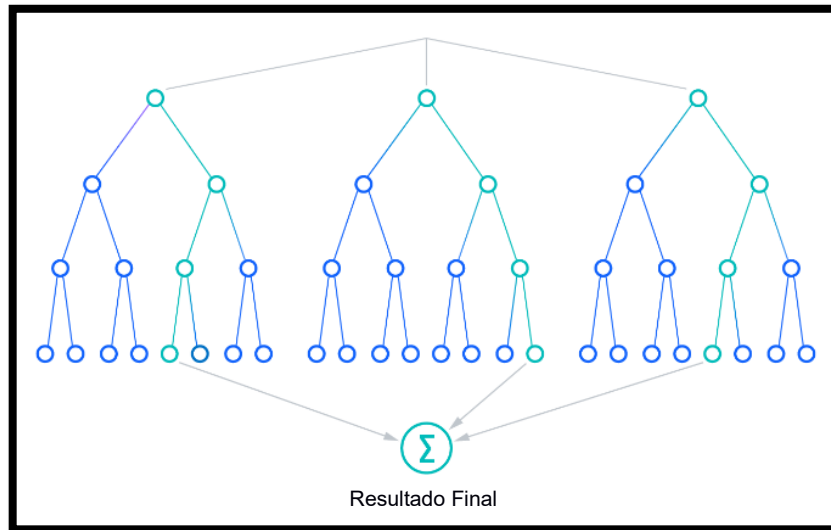
Una vez extraídas las características pertinentes, se procede a la construcción del dataset supervisado. En un aprendizaje supervisado típico, el conjunto de datos (dataset) se organiza como una tabla donde cada fila corresponde a un ejemplo o instancia (por ejemplo, una simulación de cierta condición de falla), con columnas para cada característica calculada y para la etiqueta o clase asignada [15]. Durante la preparación del dataset, es común realizar también tareas de preprocesamiento adicionales, como la normalización o escalado de las características (para que todas tengan rangos comparables) y la segmentación del dataset en subconjuntos de entrenamiento, validación y prueba [15].

1.5.7 Árbol De Decisión (Random Forest)

El árbol de decisión es un modelo de aprendizaje supervisado en forma de estructura jerárquica de decisiones, donde se realizan preguntas o comprobaciones lógicas sobre las características de entrada para subdividir iterativamente los datos [16]. Cada nodo interno del árbol representa una pregunta (por ejemplo, "¿La corriente RMS supera cierto umbral?") y las ramas corresponden a las posibles respuestas (ej. sí o no), dividiendo el conjunto de datos según dicha condición. Este proceso de particionado continúa hasta llegar a nodos hoja, que representan la decisión o clasificación final. Los árboles de decisión se entrenan eligiendo en cada paso la división que mejor separa los datos de acuerdo con alguna métrica de pureza (como la ganancia de información o el índice Gini), construyendo así un modelo que aprende reglas de decisión a partir de los datos etiquetados [16]. Su resultado es fácil de interpretar, dado que se visualiza como un conjunto de reglas if-then anidadas. Sin embargo, un solo árbol de decisión puede ser inestable y propenso al sobreajuste (overfitting), ya que tiende a ajustarse muy específicamente los datos de entrenamiento, perdiendo capacidad de generalización, para mitigar estos efectos Breiman y Cutler propusieron el algoritmo Random Forest (Bosque Aleatorio) [16].

Figura 1.4

Representación visual del algoritmo de bosque aleatorio y su modo de operación.



Nota: Obtenida del portal IBM.

La idea central es entrenar muchos árboles, cada uno ligeramente diferente, y combinar sus resultados para obtener una respuesta más robusta. En el método clásico de Random Forest, a cada árbol se le asigna un subconjunto aleatorio de instancias de entrenamiento (mediante Bootstrap sampling, es decir, muestreo con reemplazo) y, adicionalmente, en cada nodo de cada árbol se selecciona aleatoriamente solo un subconjunto de características para evaluar posibles divisiones [16]. Estas dos fuentes de aleatoriedad (en los datos y en las variables) aseguran que los árboles individuales sean poco correlacionados entre sí. Como consecuencia, cuando se combinan las predicciones de muchos árboles, los errores tienden a promediarse y cancelarse, logrando un resultado final más preciso y generalizable [17]. En problemas de clasificación, la combinación se realiza típicamente por votación mayoritaria: cada árbol emite un voto por la clase predicha, y la clase con más votos es la salida del bosque [17]. En problemas de regresión, se suele promediar el valor numérico estimado por cada árbol. Gracias a este esquema, el Random Forest aprovecha la fuerza conjunta de modelos débiles para conformar un modelo de alto rendimiento, capaz de manejar tanto tareas de clasificación como de regresión con buenos resultados.

1.5.8 Criterios Matemáticos del Árbol De Decisión (Random Forest)

Impureza Gini (criterio de clasificación): En un árbol de decisión de clasificación, la calidad de una división se mide con índices de impureza como el índice de Gini. Matemáticamente, la impureza de Gini de un nodo con “c” - clases se define como:

$$Gini = 1 - \sum_{k=1}^c p_k^2 \quad (1.1)$$

Donde:

- p_k : es la proporción de instancias pertenecientes a la clase “k” en ese nodo

Si se tiene un valor de $Gini = 0$ indica un nodo puro (todas las instancias son de la misma clase) mientras que valores mayores indican mezcla de clases (impureza)

El Error Cuadrático Medio (criterio de regresión), es donde la variable objetivo es continua, se utiliza un criterio basado en la variabilidad de los valores numéricos. El más común es la suma de cuadrados residuales (RSS), equivalente al error cuadrático, que calcula la dispersión de los valores respecto a la media del nodo.

$$RSS = \sum_{i \in R_j} (y_i - y_{R_j})^2 \quad (1.2)$$

Donde:

- $(y_i - y_{R_j})^2$: es la proporción de instancias pertenecientes a la clase “k” en ese nodo

Este criterio penaliza la heterogeneidad dentro del nodo (desviaciones grandes aumentan el RSS). Al construir el árbol de regresión, en cada posible división se calcula la suma de RSS de los nodos hijo, y se escoge la partición que minimiza el RSS total (es decir, la que produce los nodos más homogéneos en términos de y)

Una vez entrenado el bosque aleatorio, las predicciones de los múltiples árboles se combinan para dar el resultado final más robusto. En clasificación, esto se realiza mediante votación por mayoría: cada árbol emite un voto por la clase predicha, y la clase que recibe más votos entre los “T” árboles es la predicción final del Random Forest

$$YRF^{(x)} = \operatorname{argmax}_c \sum_{t=1}^T I(y^{(t)}(x) = c) \quad (1.3)$$

Donde:

- $y^{(t)}(x)$: es la clase predicha por el árbol “t” para la instancia “x”
- $I = C$: es la función indicadora (que vale 1 si el árbol “t” predice la clase “c”, y 0 en caso contrario)

En regresión, la combinación es mediante promedio: la predicción final $YRF^{(x)}$ es la media de las salidas numéricas de todos los árboles individuales, entonces:

$$YRF^{(x)} = \frac{1}{T} \sum_{t=1}^T (y^{(t)}(x)) \quad (1.4)$$

lo cual tiende a reducir la varianza del modelo combinado en comparación con cualquier árbol individual. En ambos casos (votación mayoritaria o promedio), este esquema de agregación es la clave que le da al Random Forest su mayor precisión y generalización respecto a un solo árbol, aprovechando la sabiduría colectiva de múltiples modelos débiles para producir una predicción final sólida.

Capítulo 2

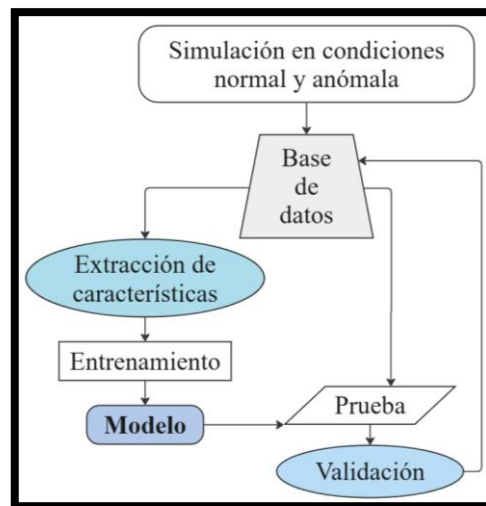
2. Metodología

2.1 Diagrama De Flujo

La Figura 2.1, describe el diagrama del flujo metodológico para la detección de fallas.

Figura 2.1

Flujo general del proceso metodológico implementado para la detección de fallas en el sistema aislado de tierra.



En la figura 2.1, se aprecian las etapas de simulación, procesamiento de datos, modelado con IA y validación que se emplearon para el desarrollo del algoritmo.

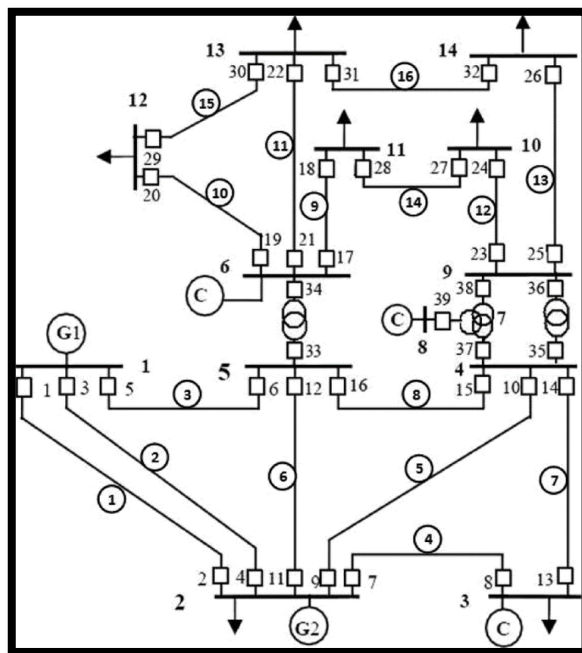
2.2 Diseño Del Sistema De Simulación

El sistema de potencia se modeló en MATLAB/Simulink usando Simscape Electrical. Se adoptó el caso de prueba estándar IEEE 14-barras desarrollado por Joshua Armah Dantuo, de la Universidad de Energía y Recursos Naturales de Ghana (14 barras, 5 generadores y 11 cargas), el sistema ampliamente conocido y utilizado en el análisis de sistemas de potencia.

A dicho modelo se incluyeron bloques de generador, líneas de transmisión y cargas para reproducir condiciones reales, y se acondicionó en base a nuestras necesidades de un sistema aislado; siendo así, se cambió su configuración en Y a una en delta en los transformadores elevadores; en las Tablas 2.1 y 2.2, se describen tanto los parámetros de carga como las configuraciones de los generadores.

Figura 2.2

Imagen representativa del modelo de 14 barras propuesto por la IEEE.



Nota: Obtenida de Research Gate.

Tabla 2.1

Parámetros de potencia empleados para la carga conectada en cada barra, valores en Mega.

LOAD/ BARRA	Potencia Activa (P) [MW]	Potencia Reactiva (QL) [MVAR]	Potencia Reactiva (QC) [MVAR]
Load 2	26.04	15.24	0
Load 3	113.04	22.86	0
Load 4	57.36	0	4.64
Load 5	9.12	1.92	0
Load 6	13.44	9	0
Load 9	35.4	19.92	0
Load 10	10.8	6.96	0
Load 11	4.2	2.16	0
Load 12	7.32	1.92	0
Load 13	16.2	6.96	0
Load 14	17.88	6	0

Tabla 2.2

Valores seteados en los generadores ubicados en el sistema de potencia de la IEEE 14 Bus.

Generador	Voltaje Fase (VLL)-kV	Angulo de Generación (δ°)	Resistencia del Gen (Ω)	Inductancia del Gen (H)
GEN 1	150,7723	37,7080	0,8929	0,0166
GEN 2	146,5967	-15,8323	0,8929	0,0166
GEN SINCRONO 1	140,9648	-34,4800	0,8929	0,0166
GEN SINCRONO 2	145,3426	-36,2022	0,8929	0,0166
GEN SINCRONO 3	149,6863	-36,1424	0,8929	0,0166

Nota: Datos extraídos de la simulación dada por Joshua Armah Dantuo, de la Universidad de Energía y Recursos Naturales de Ghana.

Para simular fallas a tierra, se usaron bloques de falla trifásica de Simulink, que permiten activar cierres programados entre fases y de fases a tierra. En cada simulación se programó una falla con una resistencia de tierra variable como se ve en la Figura 2.3. De forma aleatoria se generaron parámetros de simulación en cada ejecución: resistencia de puesta a tierra (R_g), resistencia entre líneas (R_{on}) y tiempo de inicio y fin de la falla (aleatorizados en rangos predefinidos), se variaron las cargas activas y reactivas dentro de límites para representar distintos escenarios operativos.

Figura 2.3

Bloque Three-Phase Fault usado como catalizador de falla para el estudio.

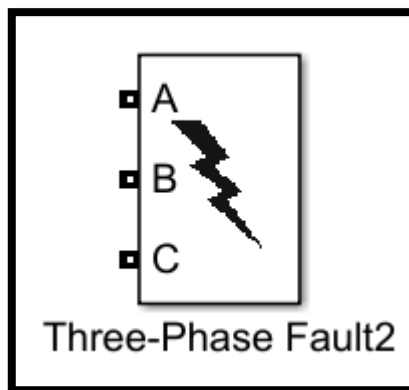
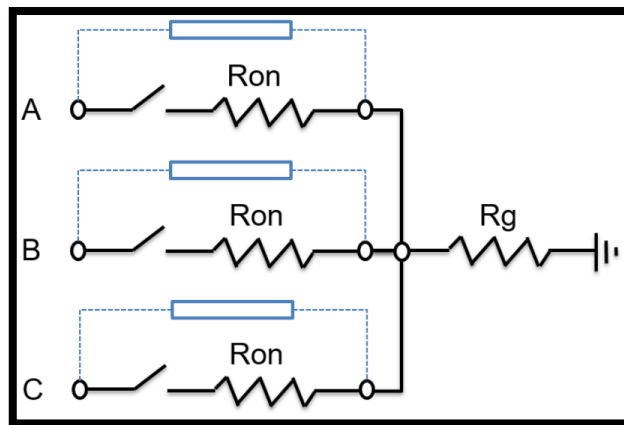


Figura 2.4

Representación del sistema Three-Phase Fault usado para una determinar las fallas en el sistema.



Durante la ejecución del modelo, se adquirieron las señales de corriente por fase en puntos clave: corriente de falla (I_{fault}), corrientes en líneas y generadores seleccionados (p. ej. IG2A, ILINE1_2A, ILINE2_3A, etc.). Estas señales se almacenaron en archivos CSV junto con la marca de tiempo. Paralelamente, se guardaron en otro archivo los parámetros de cada simulación (resistencia de tierra, tiempos de falla, valores de carga), para documentar las condiciones de cada caso. Este proceso se lleva a cabo para obtener la generación de características específicas (features) con el que se entrenó la IA. Cabe resaltar que la elección del entorno Simulink ofreció bloques especializados para fallas y mediciones en sistemas eléctricos, facilitando una simulación realista.

2.3 Generación De La Base De Datos

Para entrenar el modelo basado en inteligencia artificial (IA), se utilizó MATLAB con Simulink, tomando como base el modelo estándar libre IEEE 14 barras, adaptado con modificaciones para cumplir los objetivos del estudio. La generación de datos se llevó a cabo mediante simulaciones controladas por un script conectado a MATLAB, el cual permitía configurar diversas condiciones.

Se buscó una estructura clara para el operador y la IA en el manejo de archivos; con esto se generaron en formato CSV con la siguiente nomenclatura:

2.3.1 Corrientes_(Normal/Load)(NumeroDeBarra)_(Falla)

Esta convención facilita la interpretación de los eventos simulados. En donde el nombre del archivo indica si el sistema opera en condiciones normales (sin variaciones de carga) o con carga variable ("Load"), así mismo describe la barra en que se está simulando, además del tipo de falla registrada. Por ejemplo, "AG" representa una falla de la fase A a tierra; "AB", una falla entre fases; y combinaciones como "BGAC", una falla simultánea de fase B a tierra y entre fases A y C. Los datos detallados de este archivo se describen en la Tabla 2.3.

Tabla 2.3

Valores de fallas presente en los CSV de corrientes.

TIEMPO	Ifault_A	Ifault_B	Ifault_C	IG2A	IG2B	IG2C	Simulación
0	111.287	-36.901	-74.386	332.753	531.117	531.11	...	1
n	n

2.3.2 Datos_Corrientes_(Normal/Load)(NumeroDeBarra)_(Falla)

Este caso sigue la misma lógica del ejemplo anterior, donde la estructura del nombre del archivo es similar, sin embargo, con una diferenciación relacionada a los datos incluidos. Estos datos conforman la base principal utilizada para el entrenamiento del modelo de inteligencia artificial (IA). Cada archivo contiene información relevante como el número específico de la simulación, los valores de resistencia entre fases y resistencia a tierra. Es importante decir que, en caso de que una falla específica no involucre resistencia de fase o tierra, la simulación se realiza igualmente, aunque la IA omite dichos valores en su análisis gracias a su filtrado a través del nombre del archivo.

Además, el archivo contiene datos cruciales como el Tiempo Antes de la Falla (TAF) y el Tiempo de Finalización de la Falla (TPF). Así mismo, se registran valores de potencia aparente, activa y reactiva, los cuales se detallan en la Tabla 2.4.

Tabla 2.4

Datos de cada simulación acorde a la falla.

Simulación	R_Ground	Res_Row	taf	tpf	Load	Active	Reactive
1	409.214	9.14	0.07705	0.08631
n

Todo lo anteriormente descrito fue generado por medio de MATLAB. Posteriormente, se lleva a cabo un segundo proceso mediante un código desarrollado en Python, cuya función principal es transformar los valores primarios de corriente en valores secundarios. Para identificar claramente esta transformación en los archivos resultantes, se añade el prefijo "TC_" al nombre. Es importante decir que los valores secundarios obtenidos tras la conversión son los datos utilizados específicamente para el entrenamiento del modelo de inteligencia artificial (IA).

Tabla 2.5

Valores de la simulación con la corriente reflejada como si proviene de un TC.

TIEMPO	Ifault_A	Ifault_B	Ifault_C	IG2A	IG2B	IG2C	Simulación
0	2.782	-0.922	-1.859	-1.662	2.655	1
n	n

2.4 Conversión De Corrientes Primarias A Valores Secundarios De Tc

En realidad, las corrientes de la línea nunca se miden de forma directa, sino que pasan por un TC, que reducen el valor primario a un nivel convencional usado como protección o medición (típicamente 1/5 A secundarios). Por lo que, asemejando a dicha práctica, se desarrolló un script en Python que toma los archivos CSV generados por el Simulink y aplica las relaciones de transformación de cada lectura, en base a la Tabla 2.6.

Estos valores fueron elegidos basándonos en la corriente base del sistema y la carga de cada línea. El script divide cada medición primaria por la relación correspondiente, generando la señal secundaria esperada en el lado de protección. Tras la conversión, los nuevos datos se guardaron en archivos intermedios para su análisis posterior. Este paso permitió tener dos archivos unos con los datos de corriente reales y otro como si fueran obtenidos por un TC.

Tabla 2.6

Relación de los transformadores de corriente (Tc) empleados en el estudio de fallas, que conectan a la barra 2.

Medida	Relación CT (Ip/Is)
Corriente de falla (Ifault)	200:5
Corriente en generador G2 (IG2)	1000:5
Corriente en línea 1–2 (IL1_2)	800:5
Corrientes en líneas 2–4 y 2–5 (IL2_4, IL2_5)	700:5
Corriente en línea 2–3 (IL2_3)	500:5

2.5 Extracción De Características Y Construcción Del Dataset Supervisado

Con estos datos se agregan un valor que superan las 30,000 simulaciones entre todas las fallas que se pueden construir con las limitaciones técnicas de simulink. Entre las características calculadas figuraron el valor RMS, el valor pico (peak) y la media de cada señal en el intervalo de análisis. Además, se identificaron los instantes de inicio y fin de cada falla, obteniendo los cambios de la señal (por ejemplo, umbrales o diferencias sucesivas). De esta manera, cada simulación generó una característica única (“features”) con los valores RMS, pico y otros parámetros del segmento de falla, junto con etiquetas de salida como lo son: la magnitud de corriente de falla, el porcentaje de confianza y el tiempo de ocurrencia.

Como síntesis de las características empleadas, la Tabla 2.7 muestra las principales métricas extraídas de la señal de corriente.

Tabla 2.7

Métricas usadas como características (features) del sistema con el que se entrenó la IA.

Característica	Descripción/Formulación
<i>RMS</i>	Raíz cuadrática media: $\sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2}$
<i>Valor pico (Peak)</i>	Valor máximo de la señal ($\max_i x_i$)
<i>Valor medio (Media)</i>	Promedio aritmético: $\frac{1}{N} \sum_{i=1}^N x_i$.
<i>Duración de falla</i>	Tiempo entre detección de inicio y fin de la falla.

También se consideraron otras métricas estadísticas, (como desviación estándar o factor de forma), usadas en la IA. Sin embargo, para fines del modelo supervisado se enfatizaron los datos RMS, pico y media por su sencillez e interpretabilidad. El conjunto de datos resultante fue una tabla donde cada fila corresponde a una simulación con sus características calculadas y dos etiquetas: la magnitud de corriente máxima de falla y el instante de inicio de la misma.

2.6 Entrenamiento

2.6.1 Registro Sin Falla

En la figura 2.5 de la prueba 1 no se detectó sobrecorrientes, por lo que corresponde a una operación normal con variaciones de carga dentro de los límites admisibles.

Figura 2.5

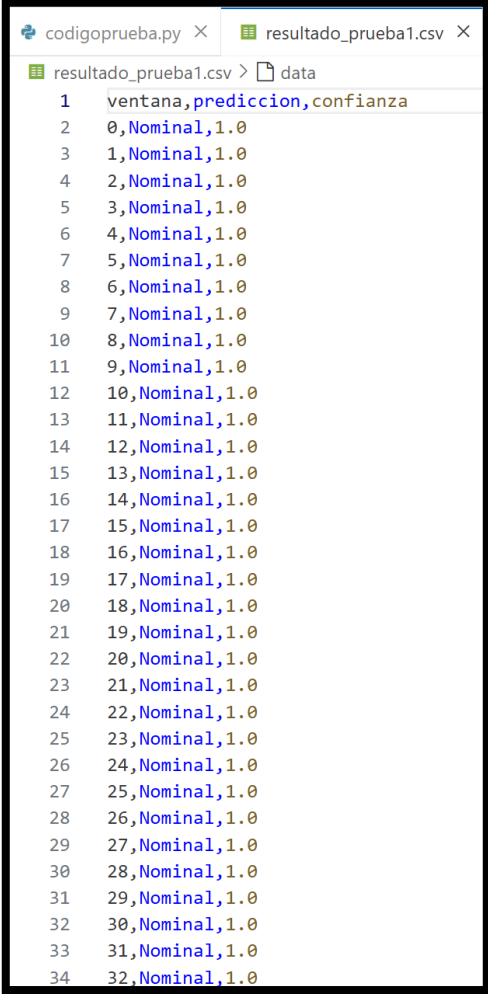
Comando para identificar falla en prueba1

```
PS C:\Users\IzzieSani\Desktop\TESIS> python codigoprueba.py predict prueba1.csv

Archivo: prueba1.csv
prediccion
Nominal      3183
Name: count, dtype: int64
Resultado detallado guardado en resultado prueba1.csv
```

Figura 2.6

Resultado de prueba1.



	ventana	prediccion	confianza
1	0	Nominal	1.0
2	1	Nominal	1.0
3	2	Nominal	1.0
4	3	Nominal	1.0
5	4	Nominal	1.0
6	5	Nominal	1.0
7	6	Nominal	1.0
8	7	Nominal	1.0
9	8	Nominal	1.0
10	9	Nominal	1.0
11	10	Nominal	1.0
12	11	Nominal	1.0
13	12	Nominal	1.0
14	13	Nominal	1.0
15	14	Nominal	1.0
16	15	Nominal	1.0
17	16	Nominal	1.0
18	17	Nominal	1.0
19	18	Nominal	1.0
20	19	Nominal	1.0
21	20	Nominal	1.0
22	21	Nominal	1.0
23	22	Nominal	1.0
24	23	Nominal	1.0
25	24	Nominal	1.0
26	25	Nominal	1.0
27	26	Nominal	1.0
28	27	Nominal	1.0
29	28	Nominal	1.0
30	29	Nominal	1.0
31	30	Nominal	1.0
32	31	Nominal	1.0
33	32	Nominal	1.0

Se observa en la Figura 2.6, el algoritmo evaluó 3183 ventanas de 20ms y les asignó etiquetas de “nominal” con un alto nivel de confianza, lo cual indica que las corrientes de fase no superaron en ninguna ventana el umbral de 3 veces la corriente nominal y tampoco mostraron componentes destacables en la secuencia 0, por lo que se encuentra dentro del rango de operación normal.

2.6.2 Registro Con Falla

A diferencia de la Figura 2.5, en la prueba 2 de la Figura 2.7, se observa que en cada ventana se superó los umbrales de sobrecorriente tanto en la secuencia 0 y 2 que caracterizan al cortocircuito. Cabe destacar que la predicción fue realizada por el algoritmo código_prueba.py.

Figura 2.7

Comando para identificar falla en prueba2.

```
PS C:\Users\IzzieSani\Desktop\TESIS> python codigoprueba.py predict prueba2.csv

Archivo: prueba2.csv
prediccion
Falla    3000
Name: count, dtype: int64
Resultado detallado guardado en resultado_prueba2.csv
```

Figura 2.8

Resultado de prueba2.

```
codigoprueba.py × resultado_prueba2.csv ×
resultado_prueba2.csv > data
1 ventana,prediccion,confianza
2 0,Falla,1.0
3 1,Falla,1.0
4 2,Falla,1.0
5 3,Falla,1.0
6 4,Falla,1.0
7 5,Falla,1.0
8 6,Falla,1.0
9 7,Falla,0.9966666666666667
10 8,Falla,1.0
11 9,Falla,0.98
12 10,Falla,0.9933333333333333
13 11,Falla,0.9966666666666667
14 12,Falla,0.9933333333333333
15 13,Falla,1.0
16 14,Falla,1.0
17 15,Falla,1.0
18 16,Falla,1.0
19 17,Falla,1.0
20 18,Falla,1.0
21 19,Falla,1.0
22 20,Falla,1.0
23 21,Falla,1.0
24 22,Falla,1.0
25 23,Falla,1.0
26 24,Falla,1.0
27 25,Falla,1.0
28 26,Falla,0.9966666666666667
29 27,Falla,1.0
30 28,Falla,1.0
31 29,Falla,1.0
32 30,Falla,1.0
33 31,Falla,1.0
34 32,Falla,1.0
```

La continuidad de la etiqueta de falla en todo el archivo indica que la falla persiste durante el intervalo analizado, además que refleja el grado de confianza varía entre 0.98 y 1 esto se debe a que existe un patrón de sobrecorrientes que se encuentra fuera de los rangos de operación normal.

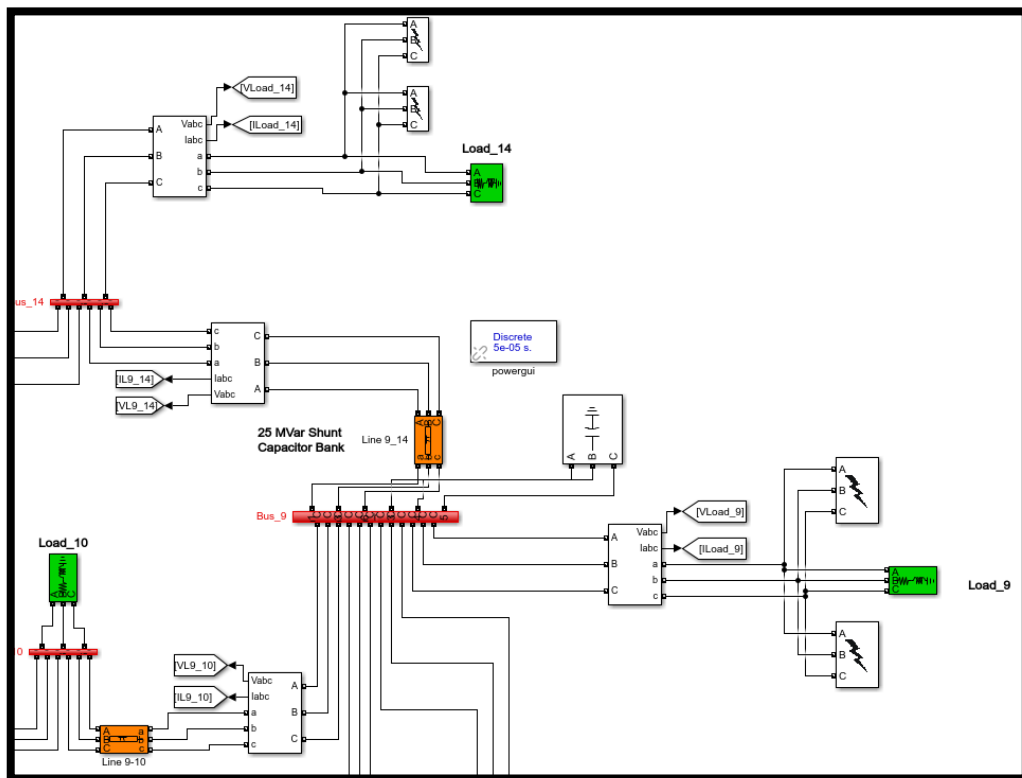
2.6 Prueba

2.6.1 Archivos Con Fallas En Barras Distintas

Se realizaron simulaciones en distintas barras, las cuales fueron utilizadas para afinar el modelo. En la Figura 2.9, fueron representadas las barras 9 y 14 para la obtención de datos, al estar estas alejadas de la generación principal serán útil para conocer su comportamiento y flujo. El total de barras empleadas para la simulación, tanto para el entrenamiento como para pequeñas pruebas, fueron las siguientes: Barra 2 (Principal), Barra 3, 4, 5, 6, 9 y 14.

Figura 2.9

Obtención de datos de la barra 9 y 14.



Nota: Simulación creada y subida en la página de Matlab, hecha por Joshua Armah Dantuo, de la Universidad de Energía y Recursos Naturales de Ghana.

2.6.2 Validación

En la Figura 2.10 se observó se clasificó como falla bifásica debido a que la fase b y fase c, que presentaron corrientes elevadas. Por tanto, es correcta la etiqueta proporcionada por el algoritmo entrenado, el cual colocó como falla con alta probabilidad, respaldando así el modelo. Por otro lado, el algoritmo determinó el tiempo de duración de falla, en menos de 2 ciclos.

Figura 2.10

Resultado del tipo de falla.



```
PS C:\Users\IzzieSani\Desktop\TESIS> & C:\Users\IzzieSani\AppData\Local\Programs\Python\Python313\python.exe c:/Users/IzzieSani/Desktop/TESIS/valida_falla.py
Iniciando análisis de fallas eléctricas...

Instrucciones:
✎ Escriba los nombres de archivo separados por espacio. Ejemplo:
  falla1.csv falla2.xlsx
✎ Para salir, simplemente presione Enter sin escribir nada.

Archivos a analizar: Corrientes_IEEE14_BC.csv

=====
📁 Archivo: Corrientes_IEEE14_BC.csv
🔍 Estado: Falla
⚡ Tipo de falla: Bifásica
🕒 Duración: 29870.0 ms, de 0.0500s a 29.9200s
📊 Corrientes: IA=53.68, IB=-4683.26, IC=4629.55
📊 Componentes: I0=0.01, I1=2688.51, I2=2688.51
📊 Ventanas con falla: 1049 / 3000
📄 Resultados guardados en: resultados_Corrientes_IEEE14_BC.csv

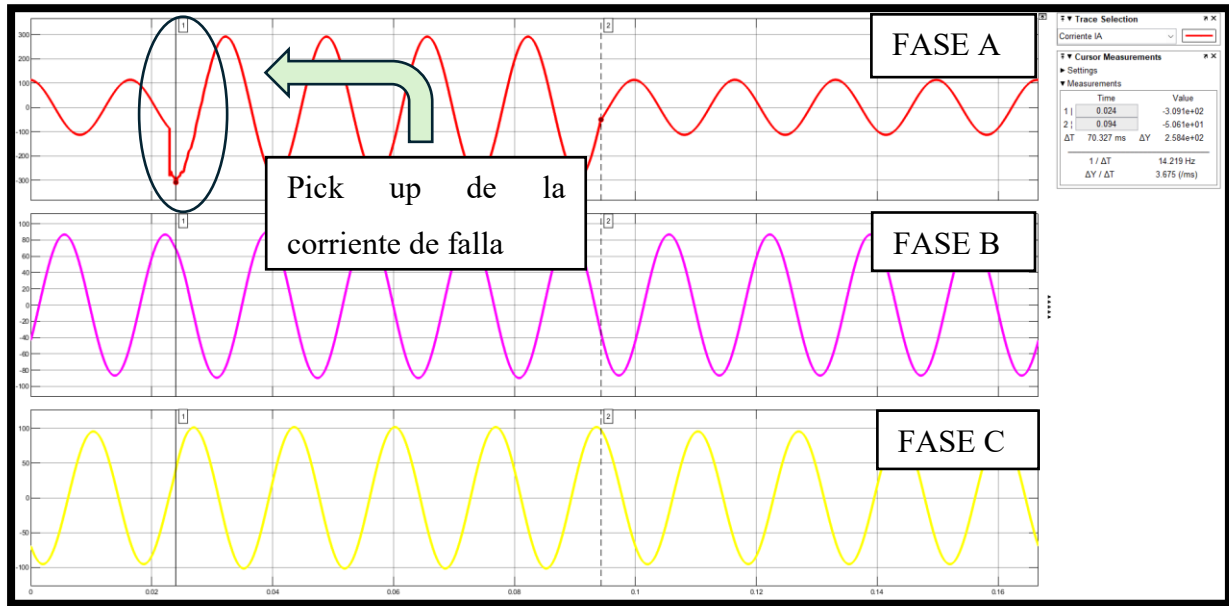
Presione Enter para salir...[ ]
```

2.6.3 Tiempo De Duración De La Falla

Ahora se muestra de forma gráfica como el sistema lograría despejar una falla. En la Figura 2.11, se tiene una curva, en la cual a los 52ms ocurre una falla entre fases (AG) que hace que su corriente se eleve a valores fuera de lo normal.

Figura 2.11

Simulación de una falla trifásica y el pick up de la corriente, obtenido por Matlab y Simulink en la carga 2 del sistema conectada a la barra 2.



Una vez generados los datos de corriente en valores primarios, se procede a procesarlos mediante el programa que simula el funcionamiento de un transformador de corriente (TC). Este paso previo convierte las magnitudes primarias en valores secundarios más manejables y adecuados para el entrenamiento de la IA.

Posteriormente, estos datos transformados se anexan al conjunto de entrenamiento, haciendo que la IA analice tanto la magnitud de la corriente como el instante temporal en que ocurre. Así se buscó que la identificación y clasificación de eventos, tal como se muestra en la Figura 2.12, detecten el valor de corriente censado y la estimación aproximada del tiempo en que ocurre dicha falla.

Figura 2.12

Datos obtenidos por la IA, de su corriente de falla (valores secundarios) y tiempo de ocurrencia.

```
>> TC_Corrientes_Load2N_AG.csv  
Tipo de falla estimada: ABCG  
Corriente de falla: -7.7263  
Tiempo de ocurrencia: 0.024
```

El tiempo estimado para la detección de una falla se encuentra en un rango de entre 1 y 1.5 ciclos de la onda de 60 Hz, lo que equivale aproximadamente a 16 a 25 milisegundos. Este es el lapso que inicia el evento hasta que se genera y transmite la señal de disparo. Tal como se muestra en Figura 2.14, el sistema busca minimizar el tiempo de respuesta, permitiendo una actuación casi instantánea ante este tipo de contingencias.

Figura 2.13

Diferencias de los tiempos de donde inicia la falla y el tiempo en el cual la IA detecta dicha eventualidad como falla, con un tiempo menor a los 10ms.

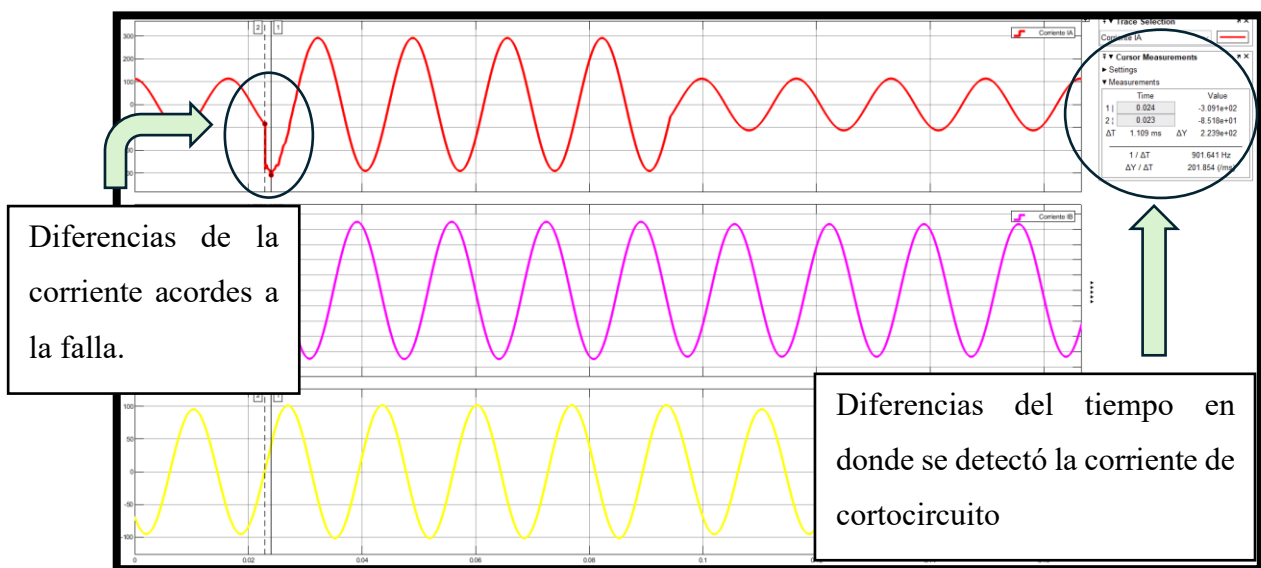
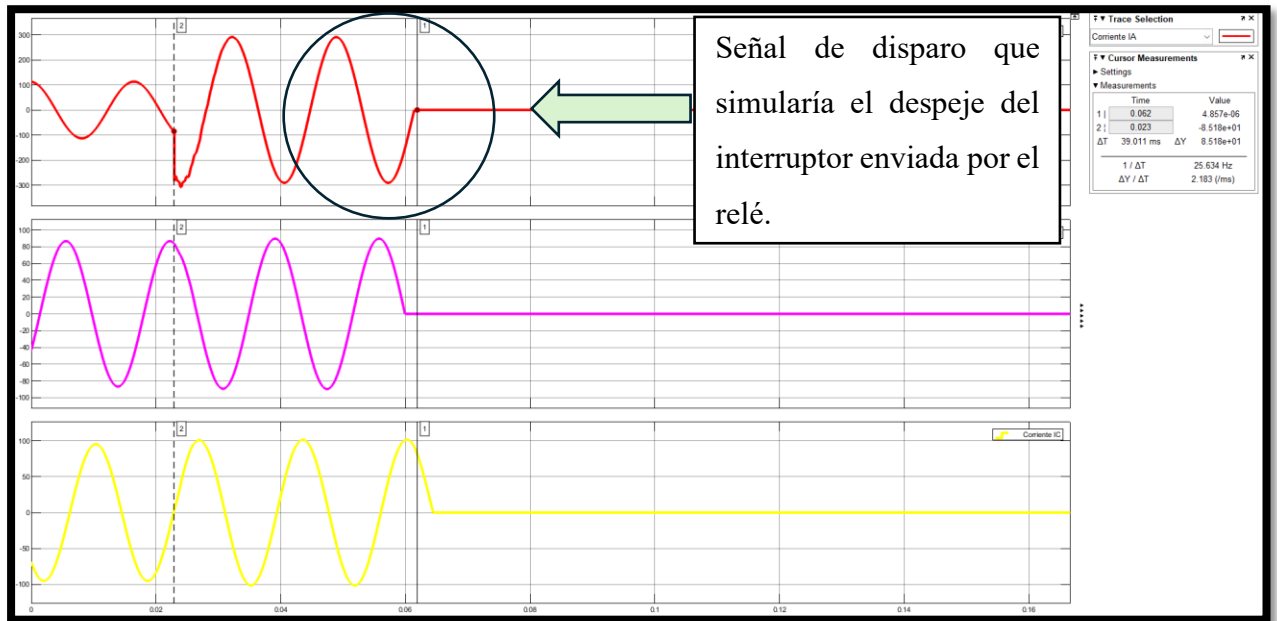


Figura 2.14

Señal de disparo que simularía el valor enviado por el relé hacia un interruptor ideal que despejaría la falla.



Capítulo 3

3. Resultados

3.1 Tablas De Porcentajes de confianza

Son los resultados de pruebas sobre eventos de fallos en un sistema eléctrico, expresados en porcentaje de confianza que el modelo logró clasificar. Estas métricas permiten medir la capacidad del algoritmo para distinguir entre los distintos aciertos al identificar un tipo de falla o estado de operación normal. La comparación entre ambas tablas permite identificar la solidez del modelo.

Tabla 3.1

Prueba con el archivo Corrientes_Load2_BG.

Clase	Ventanas	Conf_media	Conf_min	Conf_max
Monofásica	12655	0,964	0,339	1
Bifásica	3926	0,497	0,339	0,871
Trifásica	87	0,421	0,339	0,628

Para este caso, se colocó un archivo con el que ya fue entrenado el modelo, la cual fue una falla monofásica en la barra 2, para poder determinar el porcentaje de confianza del modelo, en el cual podemos observar en la Tabla 3.1 que el modelo tiene una elevada confianza media en fallas monofásicas con un 0.964, esto indica que posee una alta sensibilidad. La variabilidad entre los valores mínimos y máximos sugiere que, aunque existan momentos con baja confiabilidad, en la mayoría de los eventos la clasificación es precisa.

Por otro lado, las fallas bifásicas y trifásicas muestran valores promedios de confianza menores, esto significa que el modelo posee mayor dificultad para identificarlas de forma consistente.

Tabla 3.2

Prueba con el archivo Corrientes_Load14_AG.

Clase	Ventanas	Conf_media	Conf_min	Conf_max
Nominal	7148	0,782	0,28	1
Monofásica	8604	0,54	0,26	0,907
Bifásica	14107	0,41	0,263	0,753
Trifásica	4613	0,423	0,263	0,91

En el caso de la Tabla 3.2, se colocó un archivo nuevo para determinar el porcentaje de confianza del modelo frente a distintas condiciones de simulación que no se habían realizado anteriormente como lo fue la simulación de una falla monofásica en la barra 14. El cual obtuvimos que la confiabilidad fue de 0.54, El comportamiento evidencia que el modelo logra identificar las fallas monofásicas, pero con un nivel de confianza moderado y una alta dispersión en los resultados. La reducción de la confianza respecto a pruebas sin operación nominal se explica por el solapamiento entre condiciones nominales desbalanceadas y fallas reales, lo que obliga al clasificador a redistribuir su sensibilidad.

3.2 Gráficas De Corriente

Las siguientes figuras ilustran la huella temporal de las corrientes de falla registradas durante dos escenarios representativos del conjunto de pruebas. En cada gráfica se muestran, muestra a muestra, las amplitudes instantáneas de las fases A (azul), B (naranja) y C (verde) tal y como fueron obtenidas del sistema IEEE-14 barras después de añadirse la perturbación programada. La Figura 3.1, corresponde a un evento “AG” de alta energía, por lo que la rama de la fase A exhibe excursiones de cientos a miles de amperios, mientras que las fases restantes apenas se alteran. En contraste, la segunda figura representa un caso “Load2” con múltiples disturbios superpuestos: las tres fases presentan dispersión de picos de magnitud similar, lo que genera firmas menos diferenciadas y, en consecuencia, promedios de confianza más bajos —especialmente para las categorías bifásica y trifásica reportadas en la Tabla 3.2.

Figura 3.1

Gráficas de corrientes del Archivo: Corrientes_IEEE14.csv.

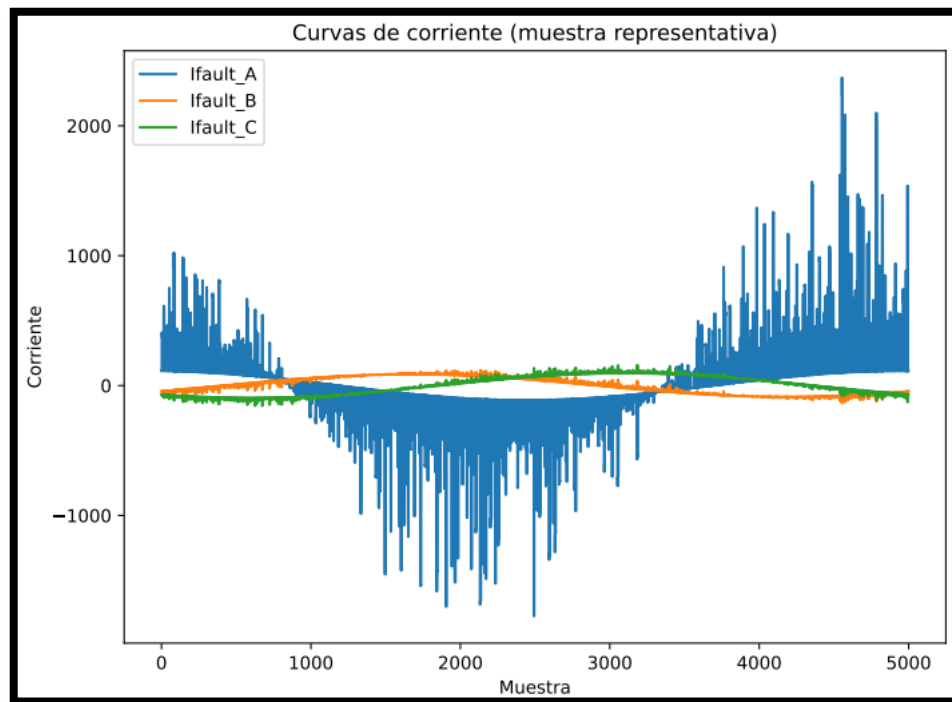
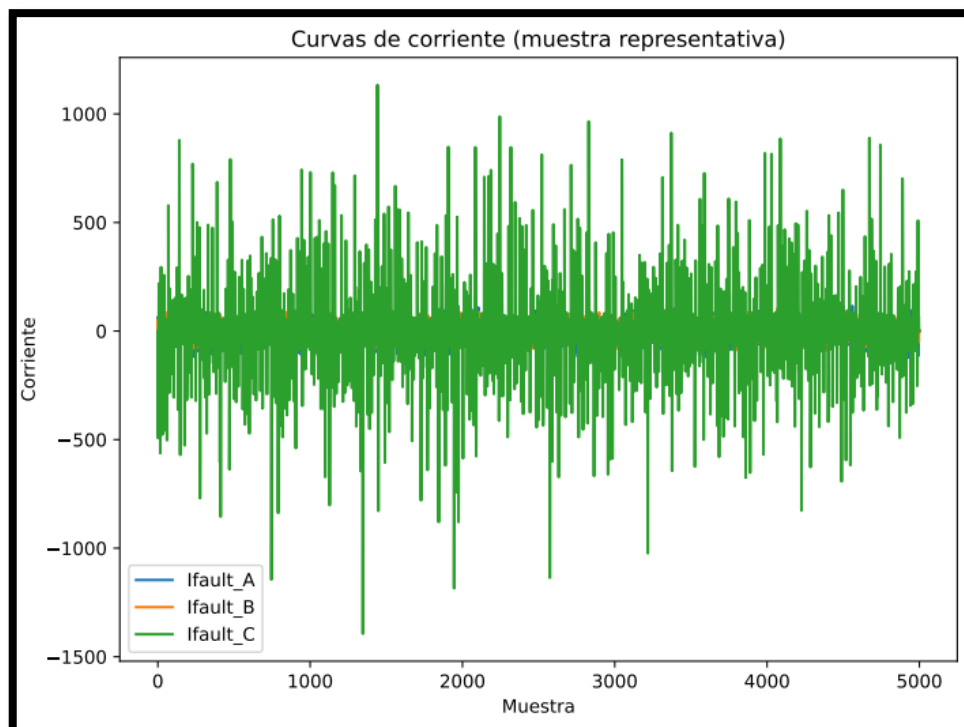


Figura 3.2

Gráficas de corrientes del Archivo: Corrientes_Load2.csv.



3.3 Análisis

En los informes generados fueron reflejados dos indicadores clave, el primero da la proporción de ventanas a las que la etiqueta dominante fue asignada; un 100 % de ventanas clasificadas como “Nominal” significó que indicios de sobrecorriente ni de secuencia cero anómala no fueron detectados. El segundo indicador, denominado confianza media, midió el grado de seguridad con que cada predicción fue emitida dentro de la clase ganadora, valores cercanos a uno implicaron firmas de corriente muy claras, mientras que medias inferiores a 0,6 revelaron que el patrón fue contaminado.

Conforme a la Tabla 3.1, correspondiente a la evaluación con el archivo Corrientes_Load2_BG que incorpora una falla monofásica en la barra 2 utilizada en el proceso de entrenamiento, el algoritmo alcanzó un índice de confiabilidad promedio del 96,4% en la clasificación de este evento. Este alto porcentaje refleja una sensibilidad y robustez elevadas en la identificación de fallas monofásicas a tierra, las normativas IEEE Std C37.230 e IEC 60909 reportan que entre el 65% y el 80% de las fallas en sistemas de potencia corresponden a esta categoría. Según la norma IEC 60255-151, los sistemas de protección deben garantizar detección rápida y segura de estas contingencias, requisito que el modelo cumple.

No obstante, el análisis de la misma tabla para fallas bifásicas y trifásicas indica una reducción significativa en los índices de confiabilidad, con valores promedio de 49,7% y 42,1% respectivamente. Esto evidencia limitaciones del modelo en la discriminación de fallas más complejas, atribuidas a un menor volumen de datos de entrenamiento representativos para estas categorías y a la similitud entre las firmas de corriente de estas fallas y perturbaciones nominales complejas. Conforme a la normativa IEEE C37.110 para la aplicación de transformadores de corriente, este comportamiento también podría estar influenciado por fenómenos de saturación y la presencia de contenido armónico en las corrientes durante eventos de cortocircuito.

En el análisis de la Tabla 3.2, relacionada con el archivo Corrientes_Load14_AG, se evaluó el modelo en un escenario no incluido en el conjunto de entrenamiento, permitiendo medir su capacidad de generalización. Si bien el clasificador identificó las fallas monofásicas, la confiabilidad promedio disminuyó a un 54%, con variabilidad marcada con valores mínimos del 26% y máximos cercanos al 91%. Esta disminución indica que, bajo condiciones nominales con cierto grado de desbalance, la superposición de corrientes normales y fallas a tierra induce incertidumbre en la clasificación. Esto recalca la necesidad de establecer umbrales de sensibilidad

ajustados al desbalance de carga, tal como recomiendan la norma IEEE 1159 sobre calidad de potencia y la IEC 60255 para evitar disparos intempestivos por corrientes residuales en condiciones normales. Las gráficas de corriente complementan estos hallazgos, en la Figura 3.1 muestra un evento tipo “AG” con una disrupción clara en la fase A, con magnitudes de corriente que van de cientos a miles de amperios, mientras que las fases B y C permanecen prácticamente inalteradas, en concordancia con la teoría clásica de fallas monofásicas a tierra y justificando la alta confiabilidad en su detección. En contraste, la Figura 3.2 presenta un escenario “Load2” con perturbaciones superpuestas y picos de corriente en las tres fases, generando patrones menos diferenciados que explican la disminución de la confiabilidad en la detección de fallas bifásicas y trifásicas. Esto enfatiza la necesidad de enriquecer el modelo con variables adicionales, tales como componentes de secuencia, fasores angulares y criterios temporales de persistencia, conforme a los lineamientos de la IEC 60255-121 sobre protecciones direccionales y de distancia.

Por ende, el análisis confirma que el modelo cumple eficazmente la función primaria de detección de fallas monofásicas, con índices de confiabilidad alineados con los requerimientos de las normas IEEE e IEC para contingencias críticas. No obstante, persisten desafíos en la clasificación de fallas bifásicas y trifásicas, sugiriendo la integración de estrategias de balance de datos, mejora en la extracción de características y validación bajo distintos escenarios de cortocircuito, para garantizar un desempeño uniforme y conforme a los estándares internacionales en protección de sistemas eléctricos.

3.4 Comparativa A Otra IA (SVM)

El sistema fue evaluado con un modelo distinto para identificar la base de conversión más eficiente, siendo el modelo de Máquinas de Vectores de Soporte (SVM) el seleccionado.

Tabla 3.3

Comparación del modelo usado con otra IA.

	Bosques Aleatorios	SVM
Cantidad de datos	Presentaron mejores valores de confianza a menor cantidad de datos.	El SVM necesita casi el doble de los datos que se usó en bosques aleatorios.

Ejecución	El tiempo de aprendizaje es mayor.	SVM su tiempo de aprendizaje es un 25% menor.
Uso de datos de apoyo	El sistema necesita una referencia externa de los datos a simular.	No es necesario esa cantidad de datos externos.
Confiabilidad	Su confianza llega al 80%, para los casos de estudios	La confiabilidad del sistema oscila un 40%.
Tiempos de detención de falla	Menores a los 2 ciclos de onda (<32 ms) y siempre detecta un tiempo	Sus tiempos llegaron a los 3 ciclos, no siempre determina un tiempo
Tipos de falla	Mayor acierto en el tipo de falla del sistema.	No determina el tipo de falla presente.

Acorde a la Tabla 3.3, para una rápida detección con una elevada robustez frente a ruido con una probabilidad confiable que trabaje con conjunto de datos limitados, el método de aprendizaje supervisado, bosques aleatorios, es la mejor opción. Por otro lado, SVM, es capaz de alcanzar un rendimiento similar en escenarios altamente controlados que dispongan de un ajuste exhaustivo y requiere mayor cantidad de datos.

3.5 Presupuesto

Dentro del presupuesto para el proyecto se estima un precio de \$4,185, los cuales serán desarrollados en la Tabla 3.4.

Tabla 3.4*Presupuesto general.*

Presupuesto					
Desarrollo, prueba y validación del algoritmo					
#	Concepto	Entregables concretos	Horas trabajadas (h)	Tarifa referencial	Sub-total
1	Simulación en Matlab	Diseño de fallas en sistema de 14 barras IEEE	35	22,5	787,5
2	Levantamiento y análisis de archivos	Matriz de requisitos, especificación de formato de datos y cronograma	15	22,5	337,5
3	Pre-procesamiento y limpieza de datasets	Manejo de valores nulos, normalización	15	22,5	337,5
4	Ingeniería de características	Cálculos de RMS, secuencias, ventanas deslizantes.	20	22,5	450
5	Implementación del clasificador (Random Forest)+pipeline CLI	Módulo rf_detector.py con entrenamiento, persistencia y modo predict	35	22,5	787,5
6	Generador de datos sintéticos para pruebas	Síntesis del código de fallas con cabeceras idénticas a plantilla	14	22,5	315
7	Pruebas unitarias y validación cruzada	Reporte con métricas de precisión, recall, matriz	30	22,5	675

8	Optimización y tuning (n árboles)	Busqueda automatizada de la red, exportación de hiper parámetros óptimos	10	22,5	225
9	Soporte de integración para transferencia de conocimiento	Sesión online y entrega de código y modelos	4	22,5	90
10	Contigencia	Cobertura de ajustes menores y retroalimentación de iteraciones	8	22,5	180
				Total	4185

Capítulo 4

4.1 Conclusiones y recomendaciones

4.1.1 Conclusiones

Luego de la ejecución secuencial de las fases planteadas en la propuesta, se logró las siguientes conclusiones que evidencian la eficacia del diseño y su aplicación en la industria para la protección de redes con neutro aislado integrando la detección inteligente de fallas mediante inteligencia artificial.

- La generación de la base de datos a partir del sistema de IEEE de 14 barras permitió estructurar un conjunto de registros eléctricos con patrones característicos de operación normal y de eventos de falla en distintas condiciones de carga y generación. La base se conformó de valores de corrientes de secuencia y corriente eficaz lo que permitió una mayor discriminación por parte del algoritmo. En la identificación de corrientes de falla y nominal se obtuvo una confianza por encima del 0.99 sin falsos positivos lo que evidencia la consistencia estadística del patrón de normalidad. Disponer de una base de datos estructurada facilitó el entrenamiento del algoritmo y se convirtió en un recurso vital para la implementación de protecciones predictivas en sistemas de generación distribuida.
- La simulación de múltiples escenarios de fallas mediante programas de análisis computacionales permitió validar estrategias de protección adaptadas a sistemas con neutro aislado. Se simuló sobrecorrientes monofásicas, bifásicas y trifásicas, así como fallas de distinta duración. El algoritmo detectó sobrecorriente en las fases B y C con valores que superaron el umbral de operación normal, clasificando el evento como falla a tierra con una probabilidad de 0,996. Solo dos ventanas en la transición de pre fallan y falla descendieron a valores entre 0,980 y 0,993 lo que corrobora la estabilidad incluso en intervalos de cambio abrupto.
- El desarrollo de algoritmo de detección basado en técnicas de procesamiento de datos, implementando mediante un modelo de clasificación de bosques aleatorios con 300 árboles demostró una precisión elevada en la identificación de fallas a tierra. En la prueba de detección de fallas de corta duración, el algoritmo no solo respondió en un tiempo mínimo, sino que también diferenció entre perturbaciones transitorias y fallas sostenidas, evitando disparos innecesarios. Esta capacidad de respuesta rápida

combinada con una alta selectividad y sensibilidad propone una metodología escalable facilitando la transición hacia esquemas de protección más inteligentes que van alineados a los requerimientos de continuidad y seguridad de operación para sistemas aislados a neutro.

4.1.2 Recomendaciones

Después de culminar, se establecen las siguientes recomendaciones que están orientadas en ampliar el alcance del presente trabajo y abordan aspectos como limitaciones de factibilidad que no fueron tratados en profundidad pero que resultan relevantes para la optimización y robustez de la protección en sistemas aislados de tierra mediante inteligencia artificial.

- Ampliar la base de datos por medio de registros reales de campo, si bien la base de datos ha demostrado ser representativa para el entrenamiento del algoritmo, su naturaleza sintética se ve limitada frente a las variaciones y perturbaciones presentes en un sistema real por lo que se recomienda recopilar datos que incluyan eventos transitorios, armónicos y condiciones de carga no línea para robustecer y minimizar posibles sesgos de simulación.
- Optimizar el modelo para reducir la carga computacional, pues el algoritmo basado en 300 árboles presentó un buen desempeño frente a precisión, sin embargo, su carga de procesamiento podría ser alta para dispositivos con recursos limitados por lo que una línea de trabajos futuros sería explorar técnicas para reducir el consumo de recursos y sea más sencillo su implementación.
- Integrar funciones de aprendizaje continuo, debió a que el modelo opera de forma estática tras el entrenamiento inicial, implementar la capacidad de actualizarse automáticamente con nuevos datos y patrones es clave para mantener su precisión a largo plazo. Se sugiere desarrollar una versión de aprendizaje incremental que permita recalibrar el algoritmo ante cambios en la red sin requerir un reentrenamiento completo lo que reduciría tiempo de ejecución.

REFERENCIAS

- [1] B. GmbH, «Electrical safety for high resistance grounded systems,» Bender Technical Article, 2021. [En línea]. Available: <https://www.bender.de/en/know-how/technology/high-resistance-grounding-system-hrg/>. [Último acceso: 1 Agosto 2025].
- [2] I. S. 242-2001, «IEEE Recommended Practice for Protection and Coordination of Industrial and Commercial Power Systems,» *IEEE XPLORE*, p. 710, 17 Diciembre 2001.
- [3] B. L. a. H. L. D. Yang, «High-Resistance Grounding Fault Detection and Line Selection in Resonant Grounding Distribution Network,» *MPDI*, vol. 11, p. 21, 28 Septiembre 2023.
- [4] I. P. T. Magazine, «Artificial Intelligence in Digital Relay Protection,» *Intelligent Power Today*, Septiembre 2023. [En línea]. Available: <https://www.intelligent-power-today.com/protective-relays/digital-or-numerical-relays/artificial-intelligence-in-digital-relay-protection#:~:text=With%20AI%2C%20digital%20relays%20can,equipment%20and%20the%20power%20network..> [Último acceso: 3 Agosto 2025].
- [5] U.S. Dept. of Energy Office of Electricity, «Smart Grid Investments Improve Grid Reliability, Resilience, and Storm Responses,» Noviembre 2014. [En línea]. Available: <https://www.energy.gov/oe/articles/smart-grid-investments-improve-grid-reliability-resilience-and-storm-responses-november>. [Último acceso: 7 Agosto 2025].
- [6] ABB, «Protection criteria for medium voltage networks,» 2010. [En línea]. Available: <https://search.abb.com/library/Download.aspx?DocumentID=1VCP000280#>. [Último acceso: 10 Agosto 2025].
- [7] Schweitzer Engineering Laboratories, «Review of Ground Fault Protection Methods for Medium,» Octubre 2007. [En línea]. Available: <https://selinc.com/api/download/2604>. [Último acceso: 12 Agosto 2025].
- [8] «Effects of Distributed Generation on Overcurrent Relay Coordination and an Adaptive Protection Scheme,» *IOP Science*, vol. 73, p. 11, 2017.
- [9] I. H. Abdulaziz Aljohani, «High-Impedance Fault Diagnosis: A Review,» *MPDI*, 5 Diciembre 2020.
- [10] D. R.-Q. y. C. A. C. Manuel Acevedo-Iles, «A Distributed Coordination Approach for Enhancing Protection System Adaptability in Active Distribution Networks,» *MPDI*, 30 Agosto 2024.

- [11] A. B. T. A. M. A. Ali Raza, «A Review of Fault Diagnosing Methods in Power Transmission Systems,» *MPDI*, 14 Febrero 2020.
- [12] D. K. M. B. L. X. Dongqi Wu, «Deep Reinforcement Learning-Based Robust Protection in DER-Rich Distribution Grids,» vol. 3, p. 10, 1 Junio 2021.
- [13] Daniella, «INNOVATIANA,» 25 11 2024. [En línea]. Available: <https://www.innovatiana.com/es/post/feature-extraction-in-ai#:~:text=La%20extracci%C3%B3n%20de%20caracter%C3%ADsticas%20es,tiempo%20mantener%20la%20informaci%C3%B3n%20esencial>. [Último acceso: 25 Agosto 2025].
- [14] M. J. A. C. M. G. Z. P. Julián Moreno, «Modelo Neuro Difuso para la Extracción de Características en Fallas Eléctricas dentro de Líneas de Transmisión,» 28 Febrero 2008. [En línea]. Available: <https://www.redalyc.org/pdf/1331/133114993002.pdf#:~:text=Resumen%E2%80%94El%C2%A0%20diagn%C3%B3stico%C2%A0%20autom%C3%A1tico%C2%A0%20de%C2%A0fa,En%C2%A0este%C2%A0art%C3%ADculo%C2%A0se%C2%A0propone%C2%A0el%C2%A0uso%C2%A0de%C2%A0un%C2%A0modelo%20que%C2%>. [Último acceso: 25 Agosto 2025].
- [15] C. S. Ivan Belcic, «IBM,» IBM, 2024. [En línea]. Available: <https://www.ibm.com/es-es/think/topics/supervised-learning#:~:text=Los%20datos%20etiquetados%20consisten%20en,y%20las%20etiquetas%20de%20datos>. [Último acceso: 28 Agosto 2025].
- [16] IBM Cloud Education, «IBM,» 2024. [En línea]. Available: <https://www.ibm.com/es-es/think/topics/random-forest>. [Último acceso: 27 Agosto 2025].
- [17] DataScientest, «DataScientest,» Enero 2025. [En línea]. Available: <https://datascientest.com/es/random-forest-bosque-aleatorio-definicion-y-funcionamiento>. [Último acceso: 30 Agosto 2025].
- [18] G. P. J. N. F. Y. Gong, «A New Ground Fault Detection Method in ThreeWire Distribution Systems to Reduce Fire Hazards,» 11 Septiembre 2024. [En línea]. Available: https://wprcarchives.org/wp-content/uploads/2025/04/Yanfeng-Gong_A-New-Ground-Fault-Detection-Method-in-Three-Wire-Distribution-Systems-to-Reduce-Fire-Hazards_2024.pdf.

- [19] P. S. S. M. D. M. V. Kustov, «Performance of Wavelet Based Compression in TI TMS320C6201 DSP,» [En línea]. Available: <https://www.ti.com/sc/docs/general/dsp/fest99/poster/rkustovsrini.pdf#:~:text=for%2016,I n%20our%20research%20Texas%20Instruments%E2%80%99>.