



Educación

Secretaría de Educación Pública



TECNOLÓGICO
NACIONAL DE MÉXICO®

cenidet[®]
Centro Nacional de Investigación
y Desarrollo Tecnológico

Tecnológico Nacional de México

**Centro Nacional de Investigación
y Desarrollo Tecnológico**

Tesis de Doctorado

**Reducción de distorsión armónica total en un
inversor multinivel mediante sistemas de inferencia
neuro-difusa adaptativa optimizados con algoritmo
de enjambre de partículas**

presentada por

M.C. Oscar Sánchez Vargas

como requisito para la obtención del grado de

**Doctor en Ciencias en Ingeniería
Electrónica**

Directora de tesis

Dra. Susana Estefany De León Aldaco

Codirector de tesis

Dr. Jesús Aguayo Alquicira

Cuernavaca, Morelos, México. Enero de 2026.



Centro Nacional de Investigación y Desarrollo tecnológico
Subdirección Académica

Cuernavaca Mor, 03/octubre/2025

Oficio No. SAC/233/2025

Asunto: Autorización de impresión de tesis

OSCAR SÁNCHEZ VARGAS
CANDIDATO AL GRADO DE DOCTOR EN CIENCIAS
EN INGENIERÍA ELECTRÓNICA
P R E S E N T E

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado **"REDUCCIÓN DE DISTORSIÓN ARMÓNICA TOTAL EN UN INVERSOR MULTINIVEL MEDIANTE SISTEMAS DE INFERENCIA NEURODIFUSA ADAPTATIVA OPTIMIZADOS CON ALGORITMO DE ENJAMBRE DE PARTÍCULAS"**; ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

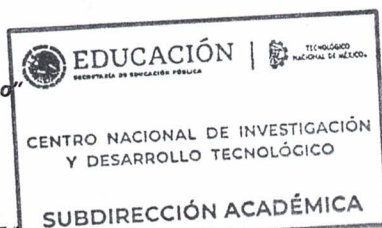
Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

ATENTAMENTE

Excelencia en Educación Tecnológica®

"Conocimiento y Tecnología al Servicio de México"

CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO



c.c.p. Departamento de Ingeniería Electrónica
Departamento de Servicios Escolares

CMAZ/lmz



2025
Año de
La Mujer
Indígena

Interior Internado Palmira S/N, Col. Palmira,
C. P. 62490, Cuernavaca, Morelos Tel. 01 (777) 3627770, ext. 4104,
e-mail: acad_cenidet@tecnm.mx tecnm.mx | cenidet.tecnm.mx

cenidet
Centro Nacional de Investigación
y Desarrollo Tecnológico



 Centro Nacional de Investigación y Desarrollo Tecnológico	ACEPTACIÓN DE IMPRESIÓN DEL DOCUMENTO DE TESIS DOCTORAL	Código: CENIDET-AC-006-D20
	Referencia a la Norma ISO 9001:2008 7.1, 7.2.1, 7.5.1, 7.6, 8.1, 8.2.4	Revisión: 0
	Página 1 de 1	

Cuernavaca, Mor., a 02 de octubre de 2025

Dr. Carlos Manuel Astorga Zaragoza
Subdirector Académico
Presente

At'n: Dr. Juan Reyes Reyes
Presidente del Claustro
Doctoral del Departamento
De Ingeniería Electrónica

Los abajo firmantes, miembros del Comité Tutorial del estudiante OSCAR SANCHEZ VARGAS manifiestan que después de haber revisado el documento de tesis titulado "REDUCCIÓN DE DISTORSIÓN ARMÓNICA TOTAL EN UN INVERSOR MULTINIVEL MEDIANTE SISTEMAS DE INFERENCIA NEURODIFUSA ADAPTATIVA OPTIMIZADOS CON ALGORITMO DE ENJAMBRE DE PARTÍCULAS", realizado bajo la dirección de la Dra. Susana Estefany De León Aldaco y la codirección del Dr. Jesús Aguayo Alquicira, el trabajo se ACEPTA para proceder a su impresión.

ATENTAMENTE



Dra. Susana Estefany De León Aldaco
Directora de Tesis / CENIDET



Dr. Jesús Aguayo Alquicira
Codirector de Tesis / CENIDET



Dr. Mario Ponce Silva
Miembro del Comité Tutorial / CENIDET



Dr. Jesús Darío Mina Antonio
Miembro del Comité Tutorial / CENIDET



Dr. Jarniel García Morales
Miembro del comité Tutorial / CENIDET



Dr. Adolfo Rafael López Núñez
I.T. Superior de Irapuato

c.c.p: Mtra. Verónica Sotelo Boyas/ jefa del Departamento de Servicios Escolares.
c.c.p: Dr. Jarniel García Morales / jefe del Departamento de Ingeniería Electrónica.
c.c.p: Expediente.

Agradecimientos

A mi asesora, la **Dra. Susana E. De León Aldaco**, por su tiempo, paciencia, conocimiento y, en especial, por esos necesarios “jalones de oreja” que me ayudaron a mantenerme enfocado durante el desarrollo de esta tesis. Agradezco profundamente su disposición para compartir su experiencia científica y por brindarme siempre su apoyo cuando más lo necesitaba.

A mi co-asesor, el **Dr. Jesús Aguayo Alquicira**, por su acompañamiento constante, por sus clases, su guía académica y el soporte técnico que resultó esencial para concretar este trabajo.

A los distinguidos miembros del jurado: el **Dr. Mario Ponce Silva**, el **Dr. Jesús Darío Mina**, el **Dr. Jarniel García Morales** y el **Dr. Luis Gerardo Vela Valdés** (†), por sus valiosas observaciones, tiempo y dedicación en la revisión de esta tesis.

A **Karla**, por su amable apoyo en los trámites de titulación y a todo el personal académico y administrativo del CENIDET por su compromiso y asistencia durante toda mi estancia en esta institución.

A mis compañeros y amigos del Doctorado —**Rafa, Kenia, Omar, Carmen, Alex, Beto, Juanito, Cachorro, Oswi, Dryden, Leo** — por compartir conmigo no solo el camino académico, sino también experiencias que hicieron de esta etapa una de las mejores de mi vida. Gracias por las risas, el apoyo y la compañía.

A mi padre **Óscar Sánchez Dorantes** y, especialmente, a mi madre **Patricia Vargas Suárez**, por su amor incondicional, su dedicación y su constante preocupación por mi bienestar. Son y serán siempre pilares fundamentales en mi vida, y su confianza en mí ha sido clave para continuar.

A mi hermana **Jimena** y a mi cuñado **Josa**, por su apoyo incondicional y sus palabras de aliento, aun cuando los temas que enfrentaba no eran fáciles de comprender. Su compañía fue un gran respaldo.

A mi sobrino **Nicolás**, por regalarme tantas sonrisas con su alegría y energía contagiosa.

A mi amigo **Daniel Arámbula**, por animarme a emprender este camino en la maestría. Su invitación y motivación fueron determinantes para que viviera esta magnífica etapa.

Al **Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET)**, por brindarme la oportunidad de cursar la Maestría en Ciencias y por el entorno académico y humano que me permitió crecer.

Al **Secretaría de Ciencia, Humanidades, Tecnología e Innovación (Secihti)**, por el apoyo económico otorgado para la realización de mis estudios de posgrado.

Contenido

Agradecimientos.....	i
Tabla de contenido	ii
Lista de Figuras.....	iv
Lista de Tablas.....	v
Acrónimos.....	vi
Resumen.....	viii
Abstract	ix
1. Introducción.....	1
1.1 Planteamiento del problema	3
1.2 Objetivo general	6
1.3 Objetivos específicos.....	6
1.4 Organización de la Tesis	6
2. Marco Teórico	8
2.1 Inteligencia Artificial	8
2.2 Redes Neurales Artificiales	9
Elementos de una RNA	11
2.3 Adaptive Neuro-Fuzzy Inference System (ANFIS)	12
Entrenamiento de las ANFIS	15
2.4 Algoritmo de Optimización por Enjambre de Partículas (PSO)	15
Parámetros de optimización del PSO.....	16
2.5 Combinación de ANFIS y PSO para la optimización de los MLI.	18
3. Estado del arte	21
3.1 Antecedentes: Aplicaciones ANFIS.....	25
3.2 Metodología de revisión sistemática de la literatura.....	29
3.2.1 Selección de artículos de revistas y ponencias de conferencias.....	30
3.2.2 Características importantes de los inversores.....	34

3.3	Discusión del Estado del Arte	37
3.4	Perspectivas futuras y conclusiones del Estado del Arte.....	45
	Perspectivas futuras	45
	Conclusiones del Estado del Arte	46
4.	Resultados	48
4.1	Propuesta.....	48
4.2	Metodología	49
4.3	Caso de estudio	52
4.4	Entrenamiento de la Red.....	54
4.5	Simulación MATLAB del caso de estudio.....	59
4.6	Discusión	62
5.	Conclusiones y trabajos futuros	67
5.1	Conclusiones.....	67
5.2	Trabajos Futuros.....	68
5.3	Productos académicos derivados del trabajo de tesis	68
6.	Referencias.....	I
	Anexo A Programa principal ANIFS.....	X
	Anexo B Código Matlab Funciones.....	XIX

Lista de Figuras

Figura 1.1 Funcionamiento de un SAI en línea [1].	2
Figura 1.2 Forma de onda escalonada de un MLI de 7 niveles.	3
Figura 1.3 Forma de onda escalonada con voltajes diferentes en las fuentes.	5
Figura 2.1 Eventos importantes de las Redes Neuronales.	9
Figura 2.2 Estructura de una RNA de dos salidas.	11
Figura 2.3 Diagramas de un sistema difuso.	12
Figura 2.4 Evaluación del sistema de inferencia difusa.	13
Figura 2.5 Estructura de una red ANFIS con dos entradas y una salida.	14
Figura 2.6 Diagrama de flujo PSO.	16
Figura 2.7 Diagrama de flujo de la combinación de ANFIS y PSO.	19
Figura 3.1 Avances en electrónica de potencia, centrados en la Fase 3 [27].	21
Figura 3.2 Clasificación de sistemas neuro difusos.	24
Figura 3.3 Metodología de revisión sistemática de la literatura.	29
Figura 3.4 Artículos encontrados sobre el uso de redes ANFIS en inversores multinivel.	30
Figura 3.5 Distribución de los métodos de entrenamiento utilizados para ANFIS.	33
Figura 3.6 Distribución de las publicaciones por tipo de cargas.	34
Figura 3.7 Distribución de las topologías utilizadas en el universo de publicaciones.	35
Figura 3.8 Distribución de las publicaciones del número de fases.	36
Figura 3.9 Distribución de las publicaciones en función del número de niveles.	36
Figura 3.10 Distribución de publicaciones Años vs. Algoritmos.	38
Figura 3.11 Distribución de publicaciones Aplicación frente a algoritmos.	39
Figura 4.1 Experimento 1 etapa de entrenamiento. a) THD vs. ángulos b) Error vs. datos de entrenamiento c) Cantidad de datos vs. error.	55
Figura 4.2 Visualización en 3D de los ángulos de conmutación obtenidos.	57
Figura 4.3 Etapa de prueba de los gráficos de error.	57
Figura 4.4 THD obtenida a la salida de la red.	59
Figura 4.5 CHBMLI de siete niveles en MATLAB Simulink.	60
Figura 4.6 Voltaje y corriente del CHBMLI en MATLAB Simulink.	61
Figura 4.7 Contenido armónico y THD del voltaje del experimento 1.	61
Figura 4.8 Comparación de %THD: Entrenamiento vs. Simulación.	63

Lista de Tablas

Tabla 2.1 Parámetros de control del PSO.	17
Tabla 3.1 Revisiones de la literatura anteriores sobre redes ANFIS o sistemas neuro difusos	25
Tabla 3.2 Documentos encontrados sobre ANFIS aplicado a MLI.....	31
Tabla 3.3 Datos obtenidos para el entrenamiento y el porcentaje de THD.	40
Tabla 3.4 ANFIS aplicado a inversores.....	44
Tabla 4.1 Porcentaje y voltajes importantes del SAI para el caso de estudio.....	52
Tabla 4.2 Porcentajes y voltajes importantes del caso de estudio.....	53
Tabla 4.3 Parámetros del sistema FIS.....	533
Tabla 4.4 Ángulos y THD obtenidos en la red ANFIS.	58
Tabla 4.5 %THD entrenamiento vs. %THD simulación.....	62
Tabla 4.6 Comparación de % THD Voltaje en la literatura.....	64
Tabla 4.7 Comparación de parámetros relevantes de ANFIS en la literatura.	644

Acrónimos

ABC - *Artificial Bee Colony* (Colonia de Abejas Artificial)
ALO - *Ant Lion Optimizer*
ALO – *Ant Lion Optimizer* (Optimizador León Hormiga)
ANFIS – *Adaptive Network-Based Fuzzy Inference System* (Sistema de Inferencia Neuro Difusa Adaptativa)
BP – *Back Propagation* (Retropropagación)
C1, C2 – Coeficientes de aceleración cognitiva y social
CEM – *Construction Engineering and Management* (Ingeniería y Gestión de la Construcción)
CHBMLI – *Cascaded H-Bridge Multilevel Inverter* (Inversor Multinivel en Cascada Puente H)
CC – Corriente Continua
DM-EFuNN – *Dynamic/Evolving Fuzzy Neural Network*
EFuNN – *Evolving Fuzzy Neural Network*
EPO – *Emperor Penguin Optimizer* (Algoritmo del pingüino emperador)
FALCON – *Fuzzy Adaptive Learning Control Network*
FINEST – *Fuzzy Inference and Neural Network in Fuzzy Inference Software Toolkit*
FIS – *Fuzzy Inference System* (Sistema de Inferencia Difusa)
FUN – *Fuzzy Net*
FV – Fotovoltaica
GA – *Genetic Algorithm* (Algoritmo Genético)
GARIC – *Generalized Approximate Reasoning Based Intelligent Control*
GD – *Gradient Descent* (Descenso de Gradiente)
HC – *Hill Climbing* (Escalada de Colina)
HL – *Hidden Layers* (Capas Ocultas)
IA – Inteligencia Artificial
IGBTs – *Insulated Gate Bipolar Transistors* (Transistores Bipolares de Puerta Aislada)
LSM – *Least Squares Method* (Método de Mínimos Cuadrados)
MCU – *Microcontroller Unit* (Unidad Microcontroladora)
Mi – Module index (Índice de Modulación)
MLI – *Multilevel Inverter* (Inversor Multinivel)
MOSFETs – *Metal-Oxide-Semiconductor Field-Effect Transistors*
MPPT – *Maximum Power Point Tracking* (Seguimiento del Punto de Máxima Potencia)
MR – Micro-Red
NEFCON – *Neuronal Fuzzy Controller*
NFN – *Neural Fuzzy Network*
NF – *Neuro-Fuzzy*
NR – *Newton–Raphson*
OMTHD – *Optimal Minimization of Total Harmonic Distortion*

PIC – *Peripheral Interface Controller*
PSO – *Particle Swarm Optimization* (Optimización por Enjambre de Partículas)
PWM – *Pulse Width Modulation* (Modulación por Ancho de Pulso)
RCGA – *Real-Coded Genetic Algorithm*
REN – Energías Renovables
RNA – *Redes Neuronales Artificiales*
RMSE – *Root Mean Square Error* (Error Cuadrático Medio)
SAI – *Sistema de Alimentación Ininterrumpida* (UPS)
SAPF – *Shunt Active Power Filter* (Filtro Activo Paralelo)
SHE – *Selective Harmonic Elimination* (Eliminación Selectiva de Armónicos)
SIF – *Sistema de Inferencia Fuzzy*
SM – Sin Mención
SONFIN – *Self-Constructing Neural Fuzzy Inference Network*
SSA – *Salp Swarm Algorithm*
SS – Solo Simulación
STATCOM – *Static Synchronous Compensator*
VRMS – *Root Mean Square Voltage* (Voltaje media cuadrática)
W – Coeficiente de Inercia

Resumen

Este trabajo trata de la optimización de los ángulos de disparo en un inversor multinivel en cascada (CHBMLI) para reducir la Distorsión Armónica Total (THD, Total Harmonic Distortion). Para ello, se propone una red de Sistema de Inferencia Neuro Difusa Adaptativa (ANFIS) combinada con el algoritmo de optimización por enjambre de partículas (PSO). El estudio se centra en la variación de los niveles de tensión de entrada derivados de un banco de baterías LiFePO_4 utilizado en un Sistema de Alimentación Ininterrumpida (SAI) para analizar su impacto sobre los ángulos de conmutación y la THD.

Se diseñó un conjunto de 35 casos experimentales utilizando un diseño de experimentos para entrenar la red ANFIS, considerando diferentes escenarios de carga de baterías. Posteriormente, el modelo entrenado se validó con datos de prueba y se comparó con los resultados obtenidos mediante simulaciones en Simulink. Los resultados muestran que la combinación de ANFIS y PSO determina eficientemente los ángulos de conmutación que minimizan la THD en el inversor, adaptándose dinámicamente a las variaciones del voltaje de entrada. Se observó una alta correlación entre los valores predichos por el modelo y los obtenidos en la simulación, validando así la eficacia del enfoque propuesto. Además, se verificó que el uso de ANFIS mejora la precisión en la selección de ángulos frente a los métodos de optimización convencionales.

Las variaciones observadas en los resultados, con $\pm 0.1\%$, demuestran que el modelo ANFIS con PSO puede manejar la incertidumbre y ajustar los ángulos de conmutación de forma eficaz, incluso en escenarios complejos. Sin embargo, se identificó que algunos experimentos muestran ligeras discrepancias en comparación con los valores entrenados, lo que puede atribuirse a la precisión decimal utilizada en los dispositivos físicos, como IGBTs o MOSFETs, que limita la precisión de conmutación.

El análisis comparativo también pone de relieve el rendimiento del método propuesto en relación con otros estudios. Aunque los niveles de THD pueden variar debido al rango de análisis de armónicos y a la presencia de filtros, el método consigue resultados comparables sin necesidad de complejas etapas de filtrado, alcanzando valores de THD entre el 10.39% y el 10.97% (para 7 niveles). Esto supone una ventaja significativa en cuanto a simplicidad de diseño y rentabilidad para sistemas SAI y otras aplicaciones de potencia.

En conclusión, la combinación de ANFIS con PSO resulta ser una alternativa para optimizar los ángulos de conmutación en inversores multinivel. Reducen la THD y mejoran el rendimiento de los sistemas SAI basados en baterías de iones de litio. La metodología propuesta puede extenderse a otros esquemas de inversores y condiciones de operación, permitiendo futuras investigaciones en electrónica de potencia e inteligencia artificial aplicada.

Abstract

This paper deals with optimizing the firing angle in a cascaded multilevel inverter (CHBMLI) to reduce the total harmonic distortion (THD). For this purpose, an adaptive inference network based on fuzzy systems (ANFIS) is combined with the particle swarm optimization (PSO) algorithm. The study focuses on the variation of input voltage levels derived from a LiFePO₄ battery bank used in an uninterruptible power supply (UPS) to analyze its impact on switching angles and THD.

A set of 35 experimental cases was designed using a design of experiments to train the ANFIS network, considering different battery charging scenarios. Subsequently, the trained model was validated with test data and compared with results obtained through simulations in Simulink.

The results show that the combination of ANFIS and PSO efficiently determines the switching angles that minimize the THD in the inverter, adapting dynamically to the input voltage variations. A high correlation was observed between the values predicted by the model and those obtained in the simulation, thus validating the effectiveness of the proposed approach. Furthermore, it was verified that using ANFIS improves the accuracy of angle selection compared to conventional optimization methods.

The variations observed in the results, with a precision of $\pm 0.1\%$, demonstrate that the ANFIS model with PSO can effectively handle uncertainty and adjust the switching angles, even in complex scenarios. However, it was identified that some experiments show slight discrepancies compared to the trained values, which can be attributed to the decimal precision used in the physical devices, such as IGBTs or MOSFETs, which limits the switching accuracy.

The comparative analysis also highlights the proposed method's performance relative to other studies. Although THD levels may vary due to the range of harmonic analysis and the presence of filters, the method achieves comparable results without requiring complex filtering stages, reaching THD values between 10.39% and 10.97% (for 7 levels). This is a significant advantage in design simplicity and cost-effectiveness for UPS systems and other power applications.

In conclusion, the combination of ANFIS and PSO proves to be an alternative for optimizing switching angles in multilevel inverters. They reduce THD and improve the performance of lithium-ion battery-based UPS systems. The proposed methodology can be extended to other inverter schemes and operating conditions, allowing future research in power electronics and applied artificial intelligence.

1. Introducción

Los SAI han alcanzado un nivel de madurez que les permite suministrar energía limpia e ininterrumpida a cargas sensibles, incluso en condiciones variables de la red eléctrica. Los SAI suelen suministrar un voltaje de salida sinusoidal regulada con baja THD y un alto factor de potencia de entrada, independientemente de los cambios en el voltaje de la red, protegiendo equipos sensibles de daños y evitando la pérdida de datos o fallos inesperados [1, 2].

Esto es crucial para aplicaciones que no pueden tolerar interrupciones o variaciones en la calidad de la energía, como sistemas de comunicación, centros de datos y equipos médicos. En entornos industriales, los SAI permiten una transición suave a generadores de emergencia u otras fuentes de reserva sin interrumpir las operaciones, lo cual es crucial para evitar tiempos de inactividad en sistemas que dependen de un funcionamiento continuo.

Los sistemas SAI se clasifican en tres categorías principales: SAI fuera de línea, SAI interactivo y SAI en línea. Cada configuración tiene características específicas que la hacen adecuada para diferentes aplicaciones [1, 3].

1. **SAI Fuera de Línea (Offline SAI):** En esta configuración, el SAI permanece inactivo mientras la red eléctrica está operativa y solo entra en funcionamiento cuando se detecta una interrupción en el suministro. Son más simples y económicos, pero ofrecen una menor protección contra problemas de calidad de la energía.
2. **SAI Interactivos de Línea:** Estos sistemas funcionan de manera similar a los SAI fuera de línea, pero incluyen un transformador de voltaje que regula la tensión de salida, incluso cuando la red eléctrica presenta variaciones. Ofrecen una mejor regulación de la tensión, aunque todavía dependen en gran medida de la red eléctrica para su funcionamiento.
3. **SAI en Línea (Online SAI):** A diferencia de los dos anteriores, los SAI en línea operan de manera continua, proporcionando una tensión de salida constante sin interrupciones, al tiempo que aíslan por completo las cargas sensibles de las fluctuaciones de la red eléctrica.

Los SAI en línea, que proporcionan una protección continua y completa contra las fluctuaciones de la red eléctrica, son los más adecuados para aplicaciones críticas, como centros de datos, sistemas médicos y sistemas de comunicaciones, donde es fundamental una calidad de la energía estable y precisa. Aunque las consecuencias de una mala calidad de la energía no siempre son evidentes a primera vista, pueden tener costes ocultos importantes que afectan a la productividad, la vida útil de los equipos y la rentabilidad general [4].

A pesar de sus ventajas, los SAI en línea se enfrentan a ciertos retos, como la gestión de transitorios, los altos niveles de THD y el ruido, que pueden afectar negativamente a la calidad de la energía suministrada. Para mitigar estos problemas, los Inversores Multinivel (MLI), en concreto los CHBMLI, se han posicionado como una solución eficaz debido a su capacidad para reducir la THD y mejorar el espectro armónico [1].

Sin embargo, la aplicación de técnicas de modulación, como la Modulación por Ancho de Pulsos (PWM), es un factor clave para optimizar aún más el rendimiento

del SAI en línea, ya que permite un control más preciso de los interruptores del inversor.

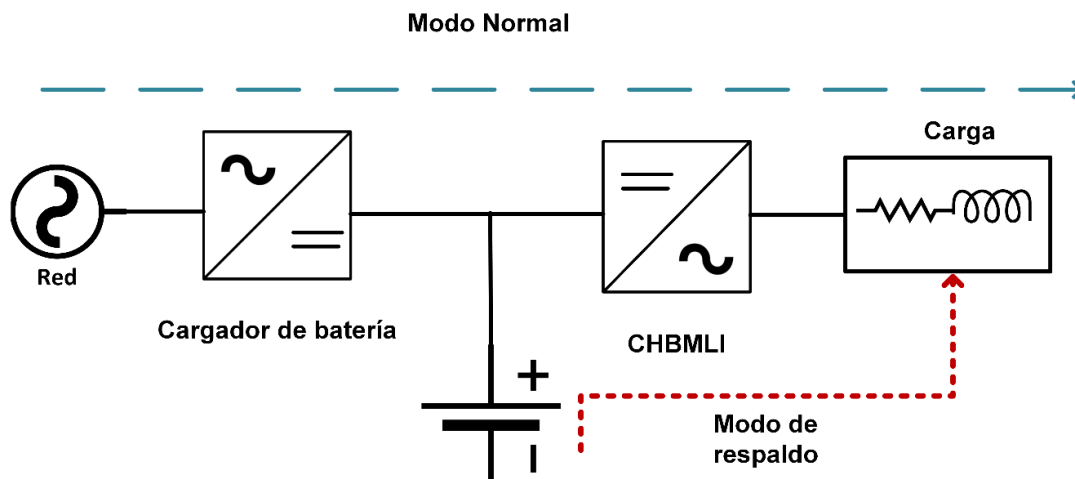


Figura 1.1 Funcionamiento de un SAI en línea [1].

La Figura 1.1 muestra el funcionamiento de un SAI en línea, los componentes principales y el flujo de energía dentro del sistema. Este diagrama muestra cómo las técnicas CHBMLI y de modulación son cruciales para mejorar la calidad del suministro de energía eléctrica.

En la generación de energía, existe una creciente preocupación por mejorar tanto la eficiencia como la calidad de la conversión. Los CHBMLI destacan por su eficacia en aplicaciones de media y alta potencia y su capacidad para reducir el tamaño de los filtros y mejorar la calidad de la alimentación, lo que los convierte en la opción ideal para integrarlos en los SAI modernos [5]. Sin embargo, uno de los principales retos sigue siendo la gestión de las variaciones de voltaje, que pueden aumentar significativamente la THD y afectar la calidad de la energía suministrada.

Aunque los CHBMLI tienen muchas ventajas, los MLI se han convertido en general en la tecnología de transformación de energía más utilizada para muchas aplicaciones [6]. Aunque los MLI ofrecen ventajas significativas, su aplicación presenta complicaciones, y los investigadores están explorando nuevas estrategias para mejorar su rendimiento [7]. Una de estas estrategias es el uso de la Inteligencia Artificial (IA), que se ha destacado como una poderosa herramienta para optimizar el rendimiento de los MLI.

La IA, aunque tiene múltiples definiciones debido a la vaguedad del término "inteligencia", puede describirse como la ciencia y la ingeniería de máquinas capaces de comprender su entorno y tomar decisiones con conocimiento de causa [8]. En pocas palabras, la IA trata de aproximarse al pensamiento humano para resolver problemas de forma eficiente y adaptativa. Dentro del campo de la IA destacan tres ramas principales: la lógica difusa, los algoritmos genéticos y las Redes Neuronales Artificiales (RNA). Las RNA intentan reproducir el proceso de resolución de problemas del cerebro [9, 10].

Las RNA, al igual que el cerebro humano, requieren un proceso de entrenamiento en el que se las alimenta con conocimientos previos para resolver problemas concretos. A pesar de su éxito, las RNA se enfrentan a problemas de

interpretabilidad y adaptabilidad en sistemas variables [9, 10]. Para hacer frente a estas limitaciones, ANFIS se desarrolló como un modelo híbrido que combina la capacidad de aprendizaje de las redes neuronales con la interpretabilidad de la lógica difusa, ofreciendo flexibilidad y claridad en escenarios complejos para la resolución de problemas igual de complejos. Esta combinación hace que ANFIS sea especialmente adecuado para aplicaciones que exigen ajustes precisos y adaptabilidad, como la optimización de inversores multinivel.

Tanto las RNA como los ANFIS requieren entrenamiento, que normalmente se consigue mediante métodos iterativos, aunque en los últimos años se ha observado un cambio hacia el uso de métodos metaheurísticos para mejorar la convergencia y la calidad de las soluciones. Los métodos metaheurísticos son algoritmos diseñados para explorar eficazmente el espacio de búsqueda e identificar soluciones óptimas a problemas complejos. A diferencia de las técnicas clásicas, como Newton-Raphson (NR) o Gradiente Descendiente (GD), los algoritmos metaheurísticos imitan procesos naturales o sociales, lo que les confiere una mayor flexibilidad y adaptabilidad a diversos escenarios [11, 12].

Aunque la integración de métodos metaheurísticos en el entrenamiento de redes ha mostrado beneficios potenciales, su aplicación sigue siendo limitada. Pocos estudios han demostrado mejoras con estos métodos en el entrenamiento de ANFIS, lo que deja este campo en gran medida inexplorado.

Aunque las redes ANFIS han encontrado amplias aplicaciones en diversas áreas, su uso en electrónica de potencia, en particular para optimizar inversores multinivel, sigue siendo limitado. La mayoría de los estudios se centran en ANFIS para optimizar sistemas eléctricos y motores eléctricos, y sólo unos pocos abordan directamente su aplicación en inversores y electrónica de potencia [13, 14].

1.1 Planteamiento del problema

La modulación de un inversor multinivel determina las relaciones entre la THD, la magnitud de la tensión y la frecuencia de salida. La forma de onda escalonada de la tensión de salida del MLI se genera mediante la solución de las ecuaciones no lineales para obtener los ángulos de conmutación de los dispositivos semiconductores, como se muestra en la Figura 1.2.

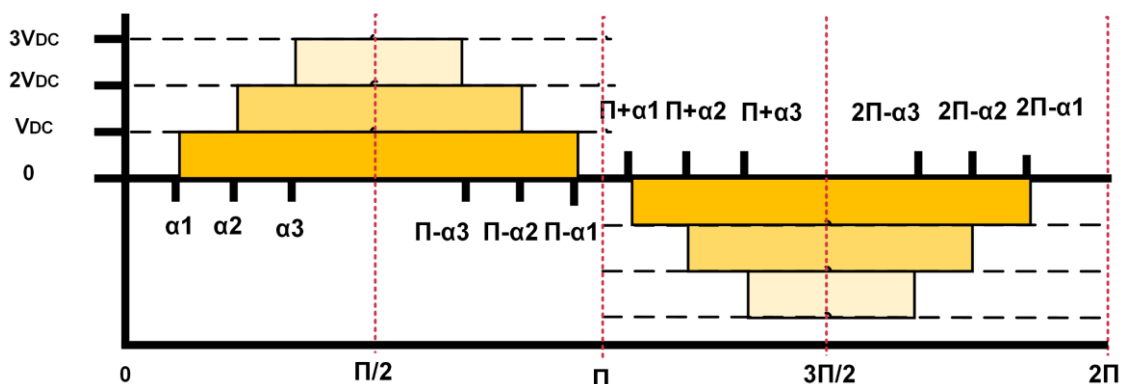


Figura 1.2 Forma de onda escalonada de un MLI de 7 niveles.

Para reducir la THD en un CHBMLI, se comienza examinando la forma de onda escalonada del voltaje de salida utilizando la serie de Fourier, y expresándola en una ecuación con simetría de cuarto de onda:

$$V_{out}(\theta) = \sum_{n=1}^{\infty} b_n \sin(n\theta) \quad (1)$$

Donde:

$n = 1$ hasta el infinito y solo impares y b_n es dado por:

$$b_n = \frac{4V_{dc}}{n\pi} \sum_{N=1}^S (V_N \cos(n\alpha_N)) \quad (2)$$

Donde:

n : 1 hasta el infinito y solo impares.

N : número de ángulos de conmutación por cuarto de ciclo.

α : ángulo de conmutación.

S : número de fuentes de CD.

El nivel de armónicos en una forma de onda de voltaje de salida es crucial para determinar su calidad. La norma IEEE 519 [15] recomienda utilizar hasta 50 armónicos en el cálculo de THD, ya que esto permite una buena evaluación del contenido armónico del sistema. Por tanto, la fórmula utilizada para calcular la THD se limita a este valor. La ecuación (3) presenta la función objetivo que esboza la fórmula de la THD:

$$THD = \frac{\sqrt{\sum_{n=3}^{49} V_n^2}}{V_1} * 100 \quad (3)$$

Donde:

V_n : Representa el valor de la amplitud de los armónicos (solo impares).

V_1 : Representa el valor de la amplitud de la señal fundamental.

Los ángulos deben cumplir las siguientes restricciones:

$$0 \leq \alpha_1 \leq \alpha_2 \leq \dots \leq \alpha_N \leq \frac{\pi}{2} \quad (4)$$

Donde:

N = número de ángulos de conmutación por cuarto de ciclo.

α_1, α_2 y α_N = son los ángulos de conmutación.

Un problema surge cuando varía el voltaje de entrada en alguna de las fuentes del CHBMLI. Los ángulos previamente calculados deben ajustarse, lo que resulta en una THD más alta y deficiente. Esta variación provoca un cambio en la forma de onda escalonada, como se muestra en la Figura 1.3.

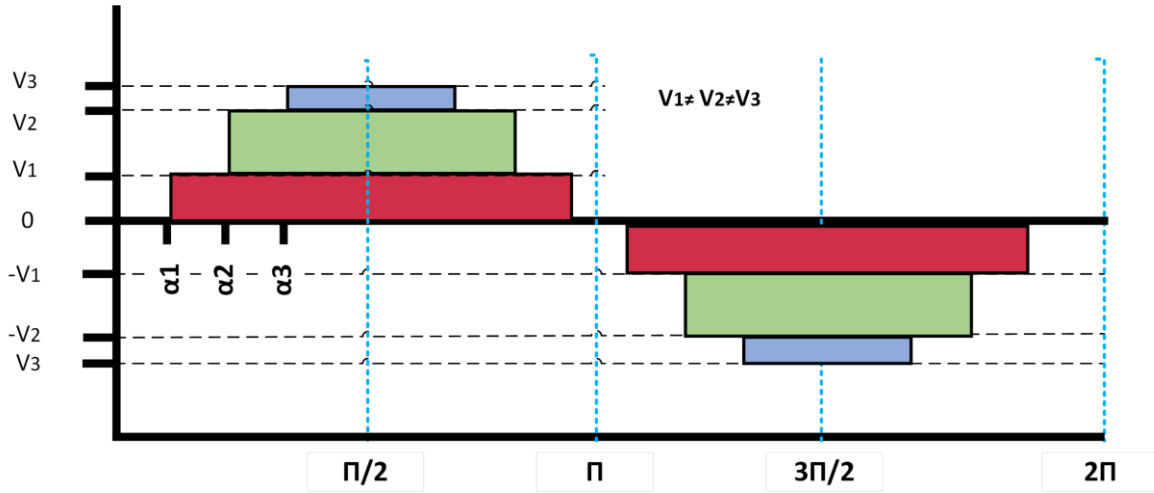


Figura 1.3 Forma de onda escalonada con voltajes diferentes en las fuentes.

La comparación entre las Figuras 1.2 y 1.3 muestra que, ante variaciones en los voltajes de las fuentes, la forma de onda escalonada cambia significativamente. Por lo tanto, el cálculo de la THD debe adaptarse para considerar estas variaciones. La ecuación (3) se reescribe para incluir los voltajes de las fuentes de entrada, de modo que se pueda determinar la THD en diferentes escenarios y condiciones de voltaje:

$$THD = \sqrt{\frac{\sum_{n=impares}^{49} \left[\frac{V_1 \cos(\alpha_1 * n) + V_2 \cos(\alpha_2 * n) + \dots V_s \cos(\alpha_N * n)}{n} \right]^2}{(V_1 \cos(\alpha_1) + V_2 \cos(\alpha_2) + \dots V_s \cos(\alpha_N))}} * 100 \quad (5)$$

Donde:

$V_1 \neq V_2 \neq V_3$ son fuentes de voltaje de entrada y variantes.

El cálculo preciso de los ángulos de conmutación en inversores multinivel es un desafío complejo debido a la variabilidad de las fuentes de entrada. Cada cambio en los voltajes de las fuentes altera las condiciones de operación, lo que genera múltiples escenarios que deben considerarse. Resolver en tiempo real las ecuaciones de eliminación selectiva de armónicos (SHE) se vuelve impráctico con métodos clásicos y algoritmos metaheurísticos, ya que estos requieren procesos iterativos intensivos y no garantizan una solución factible en cada caso.

Como alternativa, el método SHE-PWM programado calcula previamente los ángulos para distintos valores de voltaje y los almacena en tablas de consulta. Sin embargo, este enfoque tiene limitaciones significativas: demanda una gran cantidad de memoria y no permite la adaptación dinámica ante variaciones en tiempo real, lo que reduce su eficacia en aplicaciones donde las condiciones del sistema cambian constantemente.

Para superar estas restricciones, se han explorado soluciones basadas en IA, que ofrecen un control más flexible y adaptativo. Técnicas como la lógica difusa y las redes ANFIS han demostrado su eficacia en aplicaciones de electrónica de potencia, lo que permite gestionar sistemas no lineales con alta precisión. Estas herramientas ofrecen un control flexible y han tenido éxito en la gestión de sistemas

no lineales, gracias a su capacidad para interpolar y extrapolar datos con precisión una vez que han sido entrenadas adecuadamente [16].

1.2 Objetivo general

Desarrollar y simular un método de modulación para un inversor multinivel basado en la integración de una red ANFIS optimizada mediante PSO, con el propósito de minimizar la THD a través de la obtención de ángulos de conmutación óptimos para diferentes variaciones de voltaje.

1.3 Objetivos específicos

- a) Analizar el estado del arte sobre la problemática a abordar e identificación de áreas de oportunidad.
- b) Definir el caso de estudio.
- c) Simular el caso de estudio en un entorno de software y analizar los resultados.
- d) Realizar un análisis comparativo y discusión.

1.4 Organización de la Tesis

La tesis está organizada en una estructura que permite abordar de manera lógica y sistemática los objetivos planteados, así como contextualizar los métodos y resultados obtenidos. A continuación, se describe brevemente el contenido de cada capítulo:

Capítulo 1: Introducción

Se presentan el contexto y la motivación del estudio, seguidos del planteamiento del problema que se busca resolver. Posteriormente, se establecen los objetivos generales y específicos de la investigación. Finalmente, se describe la estructura del documento para guiar al lector a través de los diferentes capítulos.

Capítulo 2: Marco Teórico

Se introduce el concepto de inteligencia artificial y su aplicación en sistemas de potencia. Luego, se detallan las RNA y ANFIS, explicando su funcionamiento y su entrenamiento. También se aborda el algoritmo de PSO y su combinación con ANFIS para optimizar inversores multinivel.

Capítulo 3: Estado del Arte

Se revisan los antecedentes y aplicaciones previas del sistema ANFIS en el campo de los inversores multinivel. Se detalla la metodología de la revisión sistemática de la literatura, incluyendo la selección de artículos científicos y ponencias relevantes. Además, se discuten características importantes de los inversores y se plantea la propuesta de solución basada en los hallazgos de la revisión.

Capítulo 4: Resultados

Se describe la metodología utilizada para el desarrollo del trabajo, incluyendo el diseño del caso de estudio, la implementación del modelo ANFIS-PSO y los criterios de evaluación empleados para medir el desempeño del sistema. Después se presentan los resultados del entrenamiento de la red ANFIS, así como la simulación del caso de estudio en MATLAB. También se incluye un análisis de los resultados y una discusión sobre el impacto de la optimización en la reducción de la THD.

Capítulo 5: Conclusiones y trabajos futuros.

Se resumen los principales hallazgos del estudio, destacando las ventajas y limitaciones de la metodología propuesta. Además, se plantean posibles líneas de investigación futura para mejorar el rendimiento del sistema. Se detallan los artículos científicos, ponencias y demás productos académicos generados a partir de la investigación realizada.

Capítulo 6: Referencias

Se listan las fuentes bibliográficas utilizadas a lo largo del documento, siguiendo las normas de citación establecidas.

2. Marco Teórico

2.1 Inteligencia Artificial

La Inteligencia Artificial (IA) tiene diversas definiciones, ya que la palabra inteligencia es un término vago. Por lo que varios autores proponen diferentes definiciones de inteligencia; una definición es: la IA es una rama de las ciencias computacionales que se ocupa de los símbolos y métodos no algorítmicos para la solución de problemas [9]. Otra definición sería: la ciencia e ingeniería de las máquinas que actúan de manera que puedan comprender el entorno, capturar el significado de las cosas y darles un sentido [8]. A grandes rasgos, la IA trata de aproximarse al comportamiento y pensamiento humano para la solución de problemas.

La IA se puede dividir en tres ramas que serían:

1. Lógica difusa: es una rama de la IA que permite analizar información del mundo real en una escala de valores falso y verdadero. A diferencia de la lógica clásica, donde un valor solo puede ser 0 (falso) o 1 (verdadero), la lógica difusa maneja conceptos vagos, como caliente o frío, seco o húmedo. En la lógica difusa no se toman valores únicos o absolutos, se toman valores entre el 0 y 1, lo que permite representar grados de pertenencia, por esto las opciones que se pueden tener son infinitas, si no se acota adecuadamente el sistema [8, 9].
2. Algoritmos genéticos: Son técnicas de búsqueda iterativa inspiradas en la selección natural de Darwin. Los algoritmos genéticos no buscan modelar la evolución biológica sino crear estrategias de optimización. El concepto se basa en la generación de poblaciones de individuos mediante la reproducción de los padres.
3. Redes Neuronales Artificiales: Tratan de reproducir el proceso de solución de problemas del cerebro [9, 10]. Al igual que un ser humano requiere experiencia para tomar decisiones, las RNA necesitan un proceso de aprendizaje basado en datos previos, conocido como entrenamiento.

El entrenamiento de las RNA se puede clasificar en diferentes categorías, siendo las más relevantes los métodos clásicos y los métodos metaheurísticos.

- Métodos clásicos: utilizan enfoques matemáticos y estadísticos bien definidos para optimizar parámetros. Algunos ejemplos incluyen el método de NR y el GD, los cuales buscan soluciones óptimas mediante cálculos deterministas.
- Métodos metaheurísticos: son algoritmos que exploran de manera eficiente el espacio de búsqueda utilizando diferentes estrategias. El nombre metaheurístico deriva del sufijo en griego “meta” que significa “más allá” o “superior” y la palabra “heurístico” que significa “encontrar” [11]. Los métodos metaheurísticos no suelen ser tan rigurosos como los métodos clásicos. Los

metaheurísticos imitan ciertas características de la naturaleza, las cuales se pueden basar en la selección natural o adaptación social [11]. Dependiendo de su complejidad, pueden variar desde búsquedas locales simples hasta técnicas avanzadas de aprendizaje.

Las RNA han demostrado ser herramientas poderosas en diversas aplicaciones, desde el reconocimiento de patrones hasta la optimización de sistemas complejos. Gracias a su capacidad de aprendizaje y generalización, continúan siendo un pilar fundamental en el desarrollo de nuevas tecnologías dentro de la inteligencia artificial.

2.2 Redes Neurales Artificiales

Las RNA son modelos computacionales inspirados en el funcionamiento del cerebro humano. Al igual que un ser humano, una RNA aprende a resolver problemas a partir de experiencias previas, utilizando conjuntos de datos para ajustar sus parámetros internos y mejorar su capacidad de toma de decisiones y clasificación [9, 17].

Las RNA son particularmente útiles para resolver problemas en los que no existe una solución computacional exacta o en aquellos que requieren algoritmos demasiado complejos, como el reconocimiento de imágenes o la optimización de sistemas dinámicos. En la Figura 2.1 se presenta una línea de tiempo con los eventos más importantes en el desarrollo de las redes neuronales, basada en la información de [8, 9].

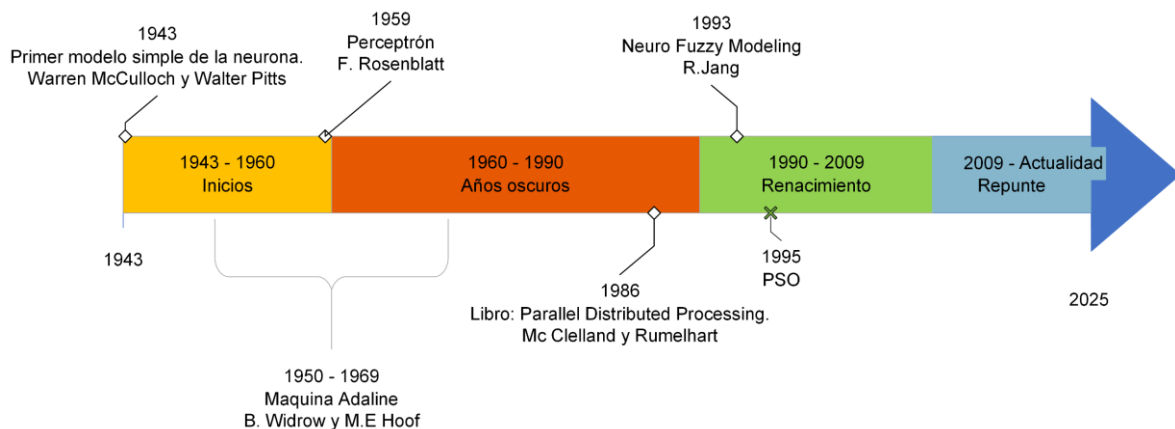


Figura 2.1 Eventos importantes de las Redes Neurales.

El desarrollo de las redes neuronales ha pasado por varias etapas clave, desde los primeros modelos teóricos en 1943 hasta su consolidación en la actualidad. Durante los primeros años, investigadores como McCulloch y Pitts establecieron las bases matemáticas para modelar neuronas artificiales. Sin embargo, las limitaciones computacionales y la falta de avances teóricos llevaron a un período de estancamiento entre 1960 y 1990, conocido como los "años oscuros". Este escenario cambió en 1986 con la publicación del libro Parallel Distributed

Processing de McClelland y Rumelhart, que estableció un marco más sólido para el desarrollo de redes neuronales eficientes. Gracias a estos avances y al progreso en la capacidad de cómputo, surgieron nuevos enfoques híbridos, como los modelos neuro-difusos (por ejemplo, ANFIS), que combinan la flexibilidad de las redes neuronales con la capacidad de razonamiento de los sistemas difusos. Asimismo, la introducción de algoritmos de optimización, como PSO en 1995, permitió mejorar el ajuste de los parámetros en las redes neuronales, ampliando su aplicabilidad en distintos campos. Estos avances sentaron las bases para el resurgimiento de las redes neuronales y su crecimiento exponencial a partir de 2009, consolidándolas como herramientas esenciales para la resolución de problemas complejos.

El renacimiento de las redes neuronales no solo permitió su aplicación en diversas áreas, sino que también resaltó sus múltiples ventajas frente a otros enfoques computacionales. Gracias a su capacidad de aprendizaje, adaptación y procesamiento de datos complejos, las RNA se han convertido en una herramienta poderosa para resolver problemas que los métodos tradicionales resultaban ineficaces.

Ventajas de las RNA

Las RNA tienen diversas ventajas que las hacen atractivas para múltiples aplicaciones, entre ellas:

- Pueden desarrollar soluciones a través de un proceso de aprendizaje sin necesidad de programación explícita.
- No requieren un conocimiento detallado de las ecuaciones matemáticas subyacentes, solo es necesario contar con datos representativos del problema.
- Son capaces de resolver problemas no lineales de manera eficiente [9, 18].
- Son robustas ante fallas parciales, ya que pueden seguir funcionando incluso si algunos elementos de procesamiento dejan de operar.
- El aprendizaje es mediante muestras.
- Son ampliamente utilizadas en clasificación, asociación y reconocimiento de patrones [19].

A pesar de sus múltiples ventajas, las redes neuronales también presentan algunas desventajas:

- Requieren entrenamiento específico para cada problema, lo que implica ajustar parámetros y estructuras.
- Determinar la arquitectura adecuada suele realizarse mediante múltiples pruebas.
- El entrenamiento es largo y demanda un alto uso de recursos computacionales.
- Necesitan grandes volúmenes de datos para lograr una buena generalización.
- Modificar una red ya entrenada puede ser complejo, especialmente si se requiere añadir nueva información sin afectar el conocimiento previo.

Principales topologías de redes neurales

Existen dos topologías que son más usadas que son:

- A. **Redes de propagación hacia adelante (*feed-forward*):** Consisten en un flujo de información estrictamente directo de las entradas a las salidas, sin ninguna conexión de retroalimentación.
- B. **Redes recurrentes:** Estas redes presentan conexiones de retroalimentación, lo que da lugar a un proceso evolutivo que conduce a un estado estacionario en el que los estados de activación de las neuronas permanecen constantes [9, 20].

Elementos de una RNA

La RNA consta de un conjunto de elementos de procesamiento (también llamados neuronas) conectados entre sí y entre los que se envían información a través de conexiones en una o varias capas. Las capas vecinas están totalmente interconectadas por “pesos” o valores.

Las neuronas de la capa de entrada reciben información del entorno exterior y las transmiten a las neuronas de la capa oculta sin realizar ningún cálculo.

Las capas situadas entre la de entrada y la de salida se denominan capas ocultas o “HL” (siglas en inglés de *Hidden Layers*) y pueden contener un gran número de unidades de procesamiento ocultas.

Finalmente, las neuronas de la capa de salida producen las predicciones al mundo exterior. Cualquier neurona que no pertenezca a la capa de entrada, calcula primero una combinación lineal de las salidas de las neuronas de la capa anterior, más un sesgo [21].

En la Figura 2.2 se observa un esquema básico de una RNA donde marca las diferentes capas y componentes que tiene un RNA *feed-forward*:

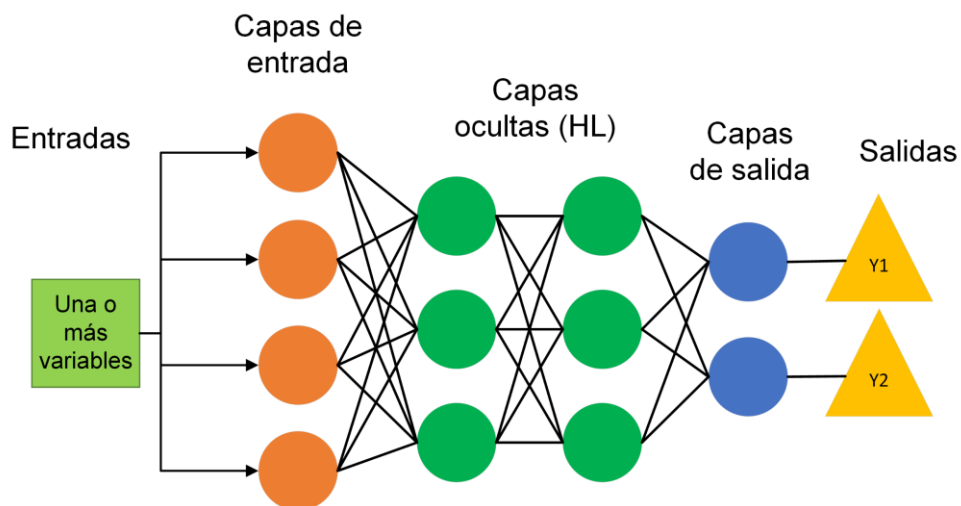


Figura 2.2 Estructura de una RNA de dos salidas.

2.3 Adaptive Neuro-Fuzzy Inference System (ANFIS)

Las redes ANFIS son una combinación de lógica difusa y redes neuronales, que utilizan distintos métodos para ajustar los parámetros. R. Jang propuso esta red en su tesis sobre modelización neuro difusa [22]. ANFIS puede modelar sistemas complejos, ya que incorpora el estilo de razonamiento humano de los sistemas difusos mediante el uso de conjuntos difusos con un modelo lingüístico consistente en un conjunto de reglas difusas, como se puede observar en la Figura 2.3 donde se aprecia el funcionamiento de un sistema difuso.

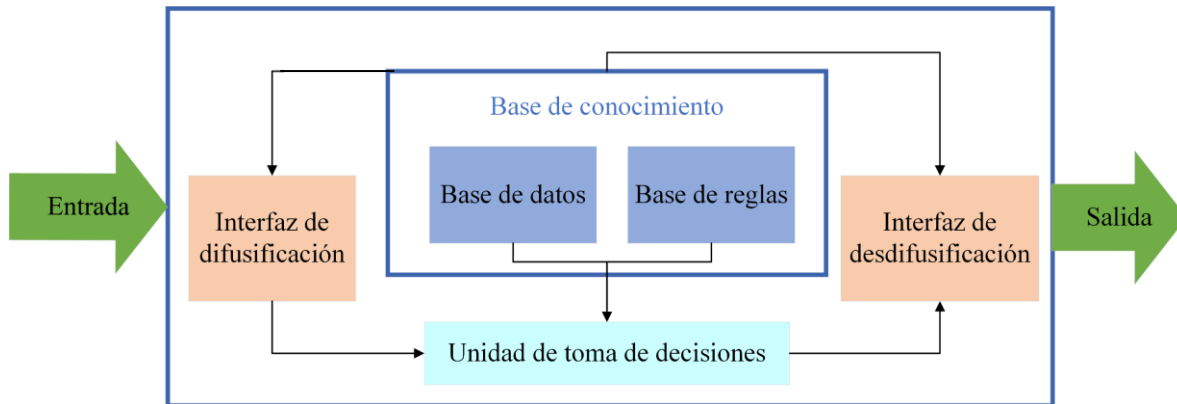


Figura 2.3 Diagramas de un sistema difuso

Este sistema está compuesto de cuatro partes principales que serían:

1. **Interfaz de difusificación:** La difusificación convierte los datos de entrada en valores lingüísticos que son las etiquetas de las funciones de pertenencia o conjuntos difusos [9].
2. **Base de conocimientos:** La base de conocimientos contienen toda la información de la aplicación, que se va a controlar y las metas del controlador. La base de conocimientos consta de una base de datos y una base de reglas. La base de datos proporciona las definiciones para el establecimiento de reglas y la manipulación de datos difusos. Mientras que la base de reglas especifica las acciones del objetivo de control mediante un conjunto de reglas lingüísticas. Dicho de otra manera, la base de reglas contiene las reglas que proporcionaría un experto. La base de reglas contiene un conjunto de reglas *IF-THEN* [9, 10].
3. **Unidad de toma de decisiones (inferencias):** Trata de simular la lógica que utilizan las personas para la toma de decisiones, con base en conceptos difusos y en la inferencia de acciones de control, empleando implicaciones y las reglas establecidas en la base de conocimientos. En otras palabras, una vez que todas las variables de entrada monitoreadas se transforman en sus respectivas variables lingüísticas (difusificación), el motor de inferencia evalúa el conjunto de reglas *IF-THEN* (dado en la base de reglas) y así se obtiene un resultado que es de nuevo un valor lingüístico para la variable lingüística [10].

4. Interfaz de desdifusificación: Se encarga del mapeo a escala que convierte el rango de valores de las variables de salida a sus universos de discurso correspondientes [9].

El principal punto fuerte de los modelos ANFIS es que son aproximadores universales que pueden utilizar reglas IF-THEN [21].

En este trabajo se utiliza el modelo difuso Takagi-Sugeno, ya que es el más adecuado para problemas de salida numérica continua, como la estimación de ángulos de conmutación en inversores multinivel. A diferencia del modelo de Mamdani, que requiere un proceso adicional de desdifusificación, el modelo Sugeno emplea funciones lineales en la salida, lo que lo hace más eficiente computacionalmente y más fácil de integrar con técnicas de entrenamiento basadas en datos, como PSO. Además, el entorno de MATLAB y la implementación estándar de ANFIS están basados en este tipo de sistema, como se describe en [22] y [9], lo cual refuerza su elección como la opción más robusta para el control adaptativo en electrónica de potencia.

El modelo ANFIS aplica el sistema difuso Takagi-Sugeno para asignar las entradas al espacio de salida, que se expresa del siguiente modo para dos entradas:

Regla 1: IF “x” es A1 y “y” es B1, THEN $f_1 = p_1x + q_1y + r_1$.

Regla 2: IF “x” es A2 y “y” es B2, THEN $f_2 = p_2x + q_2y + r_2$.

Donde p_1 , p_2 , q_1 , q_2 , r_1 y r_2 son parámetros lineales y A1, A2, B1 y B2 son parámetros no lineales.

La arquitectura de un modelo ANFIS con dos entradas se muestra en la Figura 2.4.

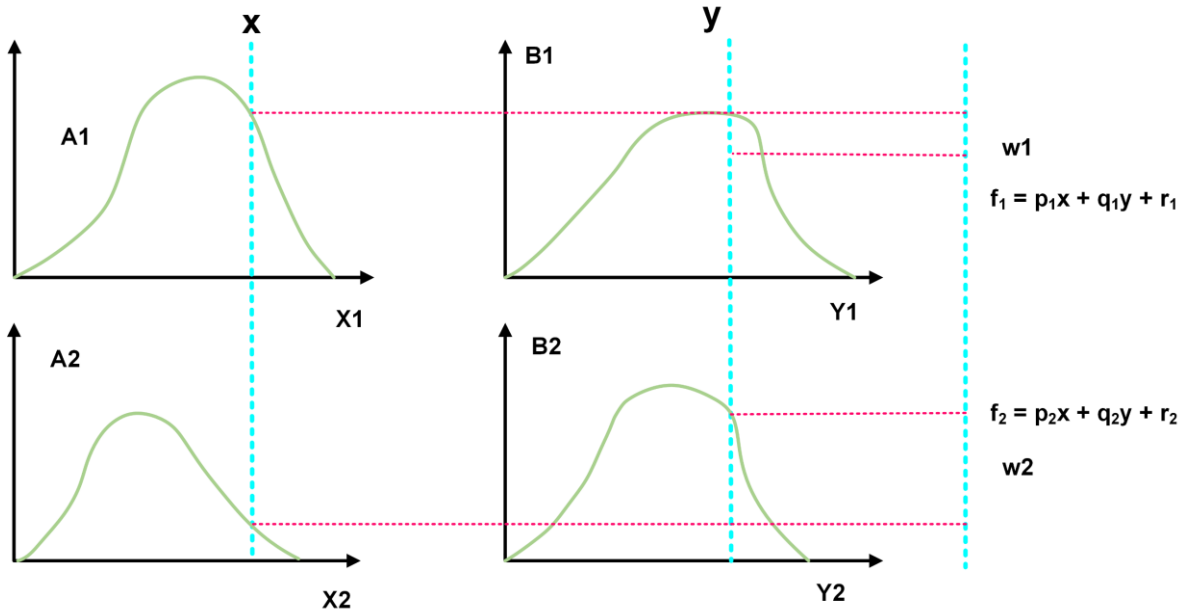


Figura 2.4 Evaluación del sistema de inferencia difusa.

En este contexto, la ecuación final del sistema de inferencia difusa de dos reglas ANFIS, que combina los resultados de ambas reglas utilizando sus valores de relevancia (w_1 y w_2), se presenta en la ecuación (6):

$$f = \frac{w_1 f_1 + w_2 f_2}{w_1 + w_2} = \bar{w}_1 f_1 + \bar{w}_2 f_2 \quad (6)$$

Donde $\bar{w}_1$ y $\bar{w}_2$ representan los pesos normalizados. Esta ecuación describe la salida final del sistema ANFIS, que es una combinación ponderada de las salidas de cada regla, integrando los valores de pertenencia de forma adaptativa para ajustar la precisión del modelo.

Las redes ANFIS, a diferencia de las redes neuronales comunes, se componen de 5 capas, como se muestra en la siguiente figura:

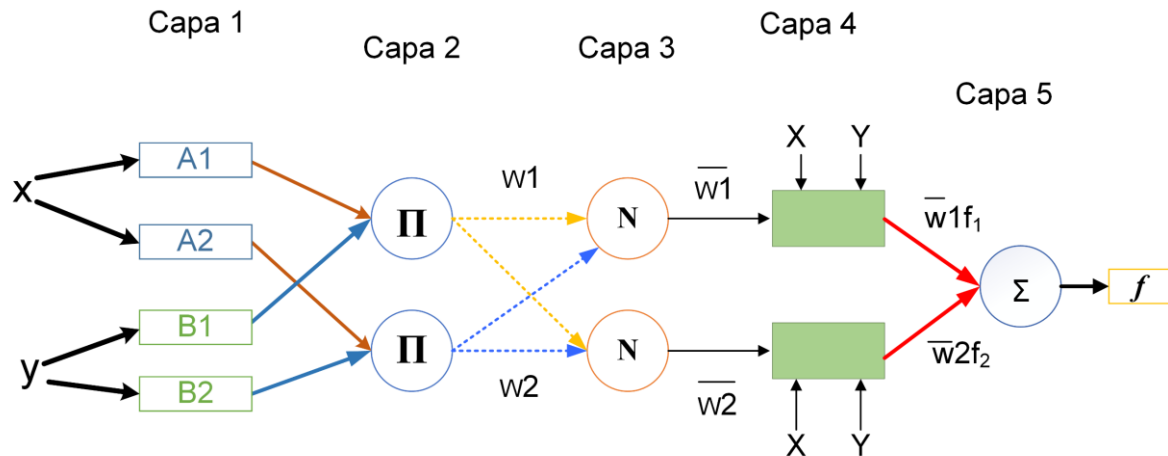


Figura 2.5 Estructura de una red ANFIS con dos entradas y una salida.

La estructura ANFIS se muestra en la Figura 2.5. Estas capas se explican como sigue:

Capa 1: La Capa de Fuzzificación transforma los datos de entrada en valores lingüísticos. Los parámetros, conocidos como parámetros de premisa, determinan la forma de la función de pertenencia [19, 20].

Capa 2: La Capa de reglas realiza la operación lógica “y” calculando el producto entre las funciones de pertenencia obtenidas en la capa de fuzzificación. Las salidas de la capa de producto sirven como funciones de peso de entrada para el siguiente nodo [19, 21].

Capa 3: La Capa de normalización calcula las fuerzas de activación asociadas a cada regla. Aquí, los pesos de entrada se normalizan para ejecutar la operación “y”. Esta capa se denomina N [19].

Capa 4: La Capa de de-fuzzificación ayuda a derivar las funciones de membresía de salida utilizando las reglas difusas previamente establecidas [20].

Capa 5: Capa de suma - Esta capa calcula la salida sumando las salidas de cada regla en la capa de de-fuzzificación [20].

Entrenamiento de las ANFIS

El entrenamiento de las redes ANFIS es importante para obtener resultados eficaces. El entrenamiento consiste en configurar y ajustar los parámetros de la red para que las entradas produzcan las salidas deseadas reforzando las conexiones. Para mejorar el entrenamiento de la red, pueden ajustarse pesos previamente conocidos. Otro método consiste en utilizar técnicas de retroalimentación y patrones de aprendizaje para cambiar los pesos hasta encontrar los adecuados [9, 18, 21]. El aprendizaje puede dividirse en dos formas:

1. Supervisado o asociativo: deben introducirse entradas que correspondan a salidas específicas (puede hacerlo un agente externo o el propio sistema).
2. No supervisado o auto organizado: debe encontrar características estadísticas entre agrupaciones de patrones de entrada.

En ANFIS, se utilizan dos métodos de entrenamiento: uno utiliza una técnica de optimización para entrenar todos los parámetros, mientras que el otro implica un enfoque de aprendizaje híbrido. En este enfoque, la parte de las suposiciones se entrena utilizando un método de optimización, y la parte de las consecuencias se entrena utilizando otro. Este enfoque tiene ventajas como evitar los mínimos locales y mejorar el rendimiento global. Los enfoques de aprendizaje híbrido que combinan métodos derivados y heurísticos se utilizan mucho en la industria [19, 20].

Una publicación reciente sobre el estado del arte de las redes ANFIS aplicadas a la electrónica de potencia, en particular en los inversores multinivel [23], destaca el limitado número de trabajos que utilizan métodos metaheurísticos y la escasez de métodos combinados con ANFIS. Los híbridos metaheurísticos con ANFIS llevan unos diez años en la literatura. Así pues, se trata de una oportunidad importante para demostrar la eficacia del entrenamiento de ANFIS con PSO.

2.4 Algoritmo de Optimización por Enjambre de Partículas (PSO)

Es una técnica de optimización metaheurística introducida por Kennedy y Eberhart en 1995. Se inspira en el comportamiento inteligente, social y de intercambio de experiencias de las bandadas de pájaros o los bancos de peces. El algoritmo PSO encuentra soluciones óptimas utilizando un conjunto de partículas voladoras con velocidades que se ajustan dinámicamente en función de su rendimiento histórico, de las partículas vecinas y, sobre todo, del "líder" en el espacio de búsqueda. Cada partícula mira en una dirección específica y luego, comunicándose entre sí, identifica la partícula en la mejor ubicación [24, 25].

La Figura 2.6 muestra el diagrama de flujo básico del algoritmo PSO.

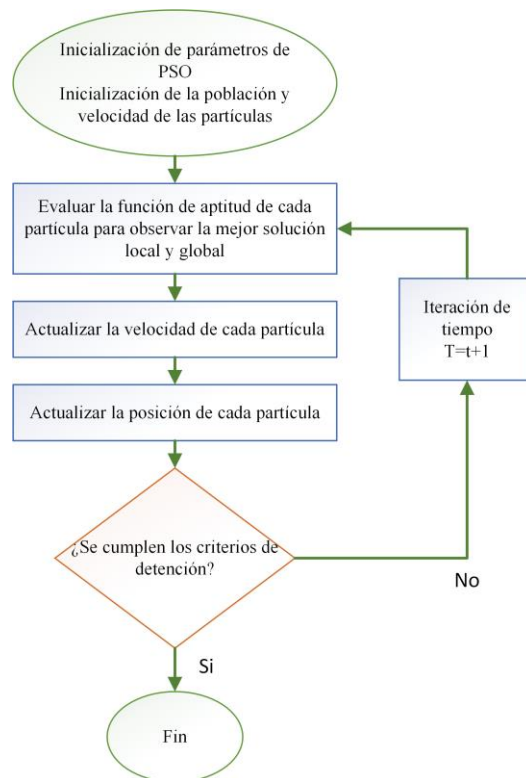


Figura 2.6 Diagrama de flujo PSO.

Parámetros de optimización del PSO

La configuración adecuada de los parámetros del algoritmo PSO es crucial para maximizar su eficiencia y eficacia en la búsqueda de soluciones óptimas. En el algoritmo PSO, varios parámetros influyen en el comportamiento del enjambre y en la calidad de la solución:

- **Coeficiente de inercia (w):** Este parámetro regula la influencia de la velocidad anterior de la partícula al calcular el nuevo valor de velocidad. Tiene un impacto directo en la capacidad de barrido y la velocidad de convergencia. Un valor de inercia adecuado permite al PSO lograr un buen barrido inicial mientras ajusta gradualmente la aproximación hacia la solución óptima.
- **Coeficientes de aceleración cognitiva y social ($c1$, $c2$):** Estos coeficientes determinan el grado de influencia ejercido por el conocimiento individual y el conocimiento colectivo sobre cada partícula. El coeficiente $c1$ regula la influencia de la mejor posición alcanzada por la partícula para guiar su nueva dirección, y el coeficiente $c2$ regula la influencia del líder del grupo en el cálculo de la dirección de búsqueda de la partícula.
- **Número de partículas y generaciones:** Como en otros algoritmos metaheurísticos basados en poblaciones, el número de partículas y generaciones afecta a la precisión y eficacia de la búsqueda. Un mayor número de partículas y generaciones puede mejorar la búsqueda de la solución óptima, pero también aumenta el coste computacional. Para evitar procesos

innecesarios, el número de generaciones se limitó después de que el algoritmo lograra resultados satisfactorios en un número reducido de iteraciones.

Si alguno de estos parámetros cambia, el desempeño del algoritmo puede variar significativamente. Por ejemplo, aumentar NP o GEN podría mejorar la calidad de las soluciones al explorar más el espacio de búsqueda, pero también incrementa el tiempo de cálculo. Por otro lado, modificar c_1 y c_2 puede causar que el enjambre se concentre demasiado en soluciones individuales (mayor c_1) o en la solución global (mayor c_2), lo cual puede llevar a una convergencia prematura. Asimismo, una mala configuración del coeficiente de inercia (w) puede resultar en oscilaciones o en estancamiento en óptimos locales. Por tanto, los valores seleccionados buscan un equilibrio entre exploración, explotación y costo computacional.

La Tabla 2.1 presenta los valores finales de cada parámetro. Estos valores fueron tomados directamente de [26], en donde demostraron ser efectivos para la minimización de la THD y el cumplimiento de la función objetivo, teniendo en cuenta los siguientes factores:

1. **Influencia en la velocidad de convergencia:** la configuración favoreció una convergencia rápida hacia soluciones de alta calidad, evitando las oscilaciones y el estancamiento en los óptimos locales.
2. **Capacidad de exploración:** El enjambre trató de explorar el espacio de búsqueda de forma inteligente, evitando la convergencia prematura hacia soluciones subóptimas.
3. **Equilibrio entre exploración y explotación:** La configuración final ofrece un equilibrio adecuado entre la exploración de nuevas áreas del espacio de búsqueda y la explotación de las mejores soluciones.

Tabla 2.1 Parámetros de control del PSO.

Nombre	Símbolo	Cantidad
Población (número de partículas)	NP	120
Generaciones	GEN	20
Coeficientes de aceleración cognitiva y social	C_1 y C_2	2
Coeficiente de inercia	$w_{\max}$	0.9
	$w_{\min}$	0.4

Una vez establecidos estos parámetros óptimos, se integraron en el proceso de optimización ANFIS, logrando una sinergia que mejora la precisión de la predicción del ángulo de conmutación y la minimización de la THD en sistemas de inversores multinivel.

2.5 Combinación de ANFIS y PSO para la optimización de los MLI.

La combinación de ANFIS y PSO es una estrategia innovadora para mejorar la generación de ángulos de conmutación en inversores multinivel. Aprovechando los puntos fuertes de ambas técnicas, se consiguen mejoras en la formación y en la minimización de THD, superando las limitaciones de los métodos tradicionales.

Las principales ventajas de esta combinación son:

- **Adaptabilidad y aprendizaje de ANFIS:** ANFIS tiene la capacidad de justarse a datos complejos y aprender patrones en la dinámica del sistema. Al ser entrenado con conjuntos de datos específicos, puede modelar eficazmente la relación entre las entradas (voltajes de las fuentes) y las salidas (ángulos de conmutación), adaptándose incluso a variaciones de voltaje.
- **Optimización global de PSO:** inspirado en el comportamiento de enjambres, busca soluciones óptimas en un espacio de búsqueda amplio. Su integración con ANFIS permite afinar los ajustes obtenidos en el entrenamiento, mejorando la precisión de los ángulos de conmutación y reduciendo la THD de manera más eficiente.

La combinación de ANFIS y PSO facilita la generación de ángulos de conmutación óptimos para distintos escenarios operativos, asegurando una menor THD en el MLI. La Figura 2.7 muestra el proceso entre ambas técnicas para el CHBMLI. Mientras que la Figura 2.8 muestra un diagrama de flujo de las etapas clave de manera ilustrativa.

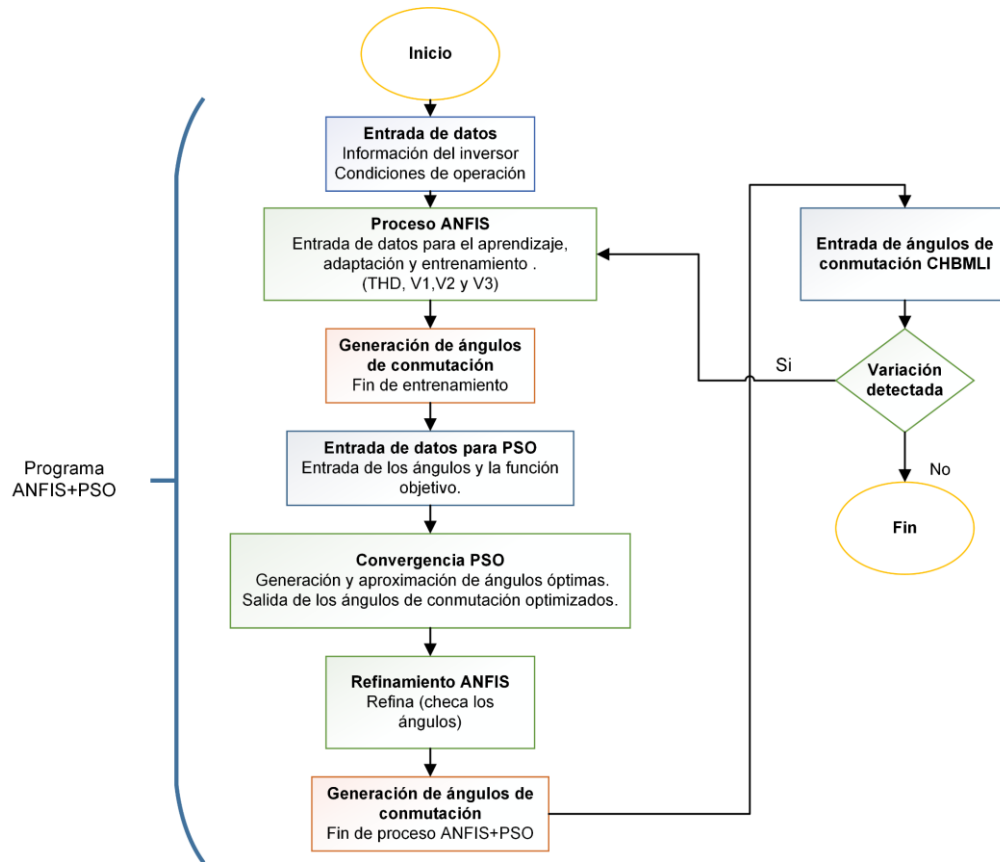


Figura 2.7 Diagrama de flujo de la combinación de ANFIS y PSO.

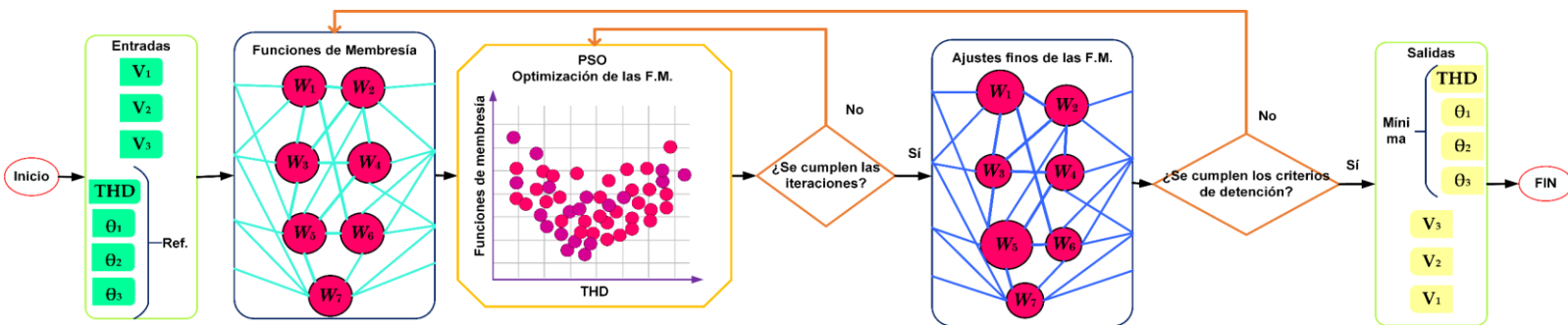


Figura 2.8 Diagrama de flujo del proceso de optimización.

El procedimiento general sigue las siguientes etapas, como se detalla en la Figura 2.8:

1. **Entrada de datos:** Se proporciona información del inversor y sus condiciones de operación, considerando variaciones en los voltajes de las fuentes de DC.
2. **Proceso ANFIS:** La red ANFIS se estructura con tres entradas (voltajes de las fuentes V1, V2 y V3) y tres salidas (ángulos de conmutación θ_1 , θ_2 y θ_3). Se definen

funciones de membresía para cada entrada y un conjunto de reglas difusas que modelan la relación entre las entradas y salidas.

3. **Optimización de PSO:** Se inicializa la población, donde cada partícula representa un conjunto de parámetros de ANFIS, incluyendo las funciones de membresía y reglas difusas. Después, se evalúa el desempeño de cada partícula calculando la THD, y se actualizan las posiciones y velocidades hasta alcanzar la convergencia en un número máximo de generaciones. Este proceso mejora continuamente las funciones de membresía y las reglas difusas.

4. **Refinamiento de ANFIS:** ANFIS ajusta sus parámetros con base en los datos optimizados por PSO, lo que mejora la calidad de las soluciones y el desempeño del entrenamiento.

Este enfoque busca demostrar la eficacia de la combinación ANFIS+PSO en la obtención de resultados óptimos en distintos escenarios. Además, permite evaluar el proceso de entrenamiento y la eficiencia de ambas técnicas en conjunto, superando las limitaciones de los métodos tradicionales y proporcionando soluciones más precisas y eficientes para la inversión multinivel.

3. Estado del arte

La historia de la electrónica de potencia se remonta a hace más de 100 años. Con el desarrollo de los semiconductores de potencia a lo largo de 65 años, esta tecnología ha evolucionado desde la búsqueda de las topologías básicas, técnicas de control y eficiencia hasta mejoras tecnológicas orientadas a las aplicaciones, como el envasado, la gestión térmica y la modularidad [27]. La Figura 3.1 muestra las fases de la electrónica de potencia.

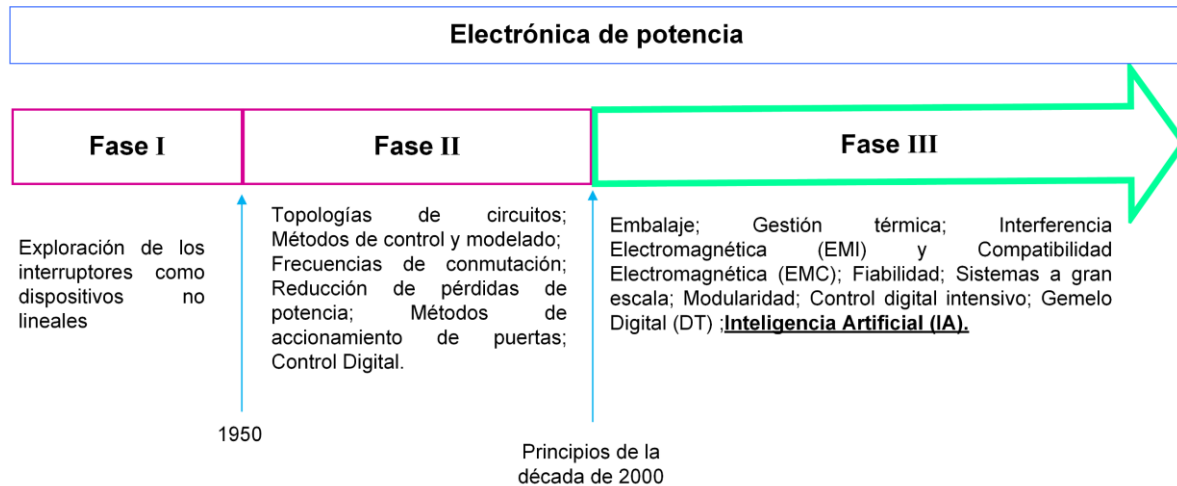


Figura 3.1 Avances en electrónica de potencia, centrados en la Fase 3 [27].

Desde la invención del rectificador de arco de mercurio hasta los avances en semiconductores de potencia y Tecnologías de la Información y la Comunicación (TIC), la electrónica de potencia ha evolucionado para permitir una integración más eficaz de las fuentes de energía renovables en la red eléctrica.

Las tecnologías de electrónica de potencia han revolucionado la conversión y el control de la energía eléctrica, especialmente tras la invención de los tiristores en la década de 1950. A lo largo de los años, se han desarrollado diversas topologías y técnicas de control para lograr una mayor eficiencia y rendimiento en los sistemas de generación de energías renovables [27].

La tecnología multinivel se ha revelado como una solución eficaz para hacer frente a las limitaciones de tensión en los dispositivos de potencia, lo que resulta especialmente relevante para los sistemas de energías renovables (REN) a gran escala. Además, el control de los convertidores de potencia se ha convertido en un componente crucial para adaptarse a las distintas demandas y maximizar la eficiencia de los sistemas REN.

Los avances en la IA, incluidos los sistemas expertos, la lógica difusa, las RNA y los algoritmos genéticos (o computación evolutiva), han ampliado las capacidades de la electrónica de potencia [28]. En la era actual de la tecnología de redes

inteligentes, la electrónica de potencia está a la vanguardia de la ingeniería energética de alta tecnología.

En los últimos años, las técnicas de IA han revolucionado la electrónica de potencia al permitir estrategias de control más sofisticadas y adaptativas. El uso de la IA en los controladores ha mejorado drásticamente la fiabilidad y estabilidad de los sistemas de generación de energía renovable, mejorando la eficiencia y la capacidad de respuesta a los cambios en las condiciones de funcionamiento.

Sin embargo, la integración de sistemas de generación de energías renovables basados en la electrónica de potencia también presenta retos, sobre todo en lo que se refiere a la calidad de la energía y los problemas asociados a las interferencias armónicas. Los armónicos elevados generados por los convertidores y su interacción pueden provocar resonancias e incapacidades en la red eléctrica [27].

Las técnicas de IA han ganado gran popularidad en los últimos años debido a su capacidad de aprender y razonar, lo que ofrece importantes ventajas en diversos sectores, como la clasificación de imágenes, el reconocimiento de voz y los coches autónomos. El campo de la electrónica de potencia también se ha beneficiado de los avances en IA, presentando un inmenso potencial [29].

En la electrónica de potencia se enfrenta a retos específicos y presenta características únicas, como la necesidad de una alta velocidad de sintonización en el control y una mayor sensibilidad en la supervisión del estado para la detección del envejecimiento. La aplicación de técnicas de IA ha repercutido notablemente en los accionamientos de motores y los sistemas de energías renovables por sus flexibles opciones de control. Los algoritmos de IA destacan en el manejo de sistemas no lineales, ya que pueden interpolar y extrapolar con precisión datos aleatorios una vez entrenados adecuadamente [16].

En los últimos años se ha producido un aumento significativo del uso y desarrollo de la IA, sobre todo en el ámbito del control. Las publicaciones y actividades de investigación relacionadas con la IA han experimentado un notable auge. Además, el diseño y el mantenimiento de aplicaciones de IA han experimentado un crecimiento significativo desde 2007, y estas tendencias se han acentuado aún más en los últimos siete años [29].

En la literatura se han propuesto varias técnicas de IA, como las redes neuronales de lógica difusa, los algoritmos neuro difusos o la optimización metaheurística [19]. Una técnica de IA que ha despertado especial interés entre los investigadores son los sistemas neuro difusos, debido a la creciente necesidad de sistemas inteligentes para resolver problemas complejos del mundo real. Los primeros estudios sobre sistemas neuro difusos se remontan a principios de los años 90, con Jang, Lin y Lee en 1991, Berenji en 1992 y Nauck a partir de 1993, entre otros [30].

Los algoritmos neuro difusos combinan las redes neuronales artificiales y la lógica difusa, combinando el estilo de razonamiento humano de los sistemas difusos con las capacidades de aprendizaje y la estructura de las redes neuronales [19, 31].

Sin embargo, utilizar estas dos IA de forma aislada tiene limitaciones y dificultades que pueden superarse combinándolas [30]. Las distintas combinaciones de estas

técnicas pueden dividirse en sistemas neuro difusos cooperativos, sistemas neuro difusos concurrentes y sistemas neuro difusos fusionados o híbridos [30, 31].

1. **Sistemas Neuro difusos Cooperativos:** El sistema neuro difuso cooperativo es un preprocesador en el que el mecanismo de aprendizaje de las RNA determina las funciones de pertenencia del Sistema de Inferencia Difusa (en inglés Fuzzy Inference System o FIS) o las reglas difusas a partir de los datos de entrenamiento. Una vez determinados los parámetros del FIS, la RNA pasa a un segundo plano. El enfoque basado en reglas suele establecerse mediante una técnica de agrupación, como los mapas autoorganizados o los algoritmos de agrupación difusa. Las funciones de pertenencia suelen aproximarse mediante redes neuronales basadas en los datos de entrenamiento [32].
2. **Sistemas Neuro difusos concurrentes:** La RNA ayuda continuamente al FIS a determinar los parámetros necesarios, como el preprocesamiento o post procesamiento de las salidas. A veces, los resultados producidos por los FIS pueden no ser directamente relevantes para la tarea en cuestión. En tales casos, la RNA puede ser una herramienta de post procesamiento para refinar los resultados del SIF [30, 32].
3. **Sistemas Neuro difusos fusionados o híbridos:** utilizan algoritmos de aprendizaje de RNA para determinar los parámetros de los NFIS, compartiendo estructuras de datos y representaciones del conocimiento. Sin embargo, aplicar un algoritmo de aprendizaje a un sistema difuso requiere funciones diferentes, o el algoritmo de aprendizaje neuronal estándar puede no funcionar [32].

Existen varias formas de desarrollar sistemas neuro difusos fusionados, y múltiples investigadores han definido sus modelos particulares. Estos modelos son similares en esencia, pero presentan diferencias básicas.

Los sistemas neuro difusos pueden representarse mediante redes neuronales que realizan funciones lógicas, aunque esto no es necesario para aplicar un algoritmo de aprendizaje en un sistema difuso. Sin embargo, el uso de redes neuronales para la representación es más conveniente, ya que permite visualizar el flujo de datos dentro del sistema y las señales de error utilizadas para actualizar sus parámetros. La ventaja adicional es que permite comparar los distintos modelos y visualizar sus diferencias estructurales. Existen varias arquitecturas neuro difusas, como, por ejemplo:

- Fuzzy Adaptive Learning Control Network (FALCON) [33].
- Adaptive Network-based Fuzzy Inference System (ANFIS) [22].
- Generalized Approximate Reasoning based Intelligence Control (GARIC) [34].
- Neuronal Fuzzy Controller (NEFCON) [35].

- Fuzzy Inference and Neural Network in Fuzzy Inference Software (FINEST) [36].
- Fuzzy Net (FUN) [37].
- Self-Constructing Neural Fuzzy Inference Network (SONFIN) [38].
- Fuzzy Neural Network (NFN) [39].
- Dynamic/Evolving Fuzzy Neural Network (EFuNN and dmEFuNN) [40].

Entre los sistemas neuro difuso fusionados, el más destacado es ANFIS. En cuanto al Error Cuadrático Medio (RMSE), los modelos Neuro Difusos o Neuro-Fuzzy en inglés (NF) que utilizan el FIS de Takagi Sugeno obtienen mejores resultados que el FIS de Mamdani, a pesar de su elevado coste computacional. En [32], realizaron una comparativa del rendimiento de los sistemas neuro difuso. El entrenamiento se realizó utilizando 500 conjuntos de datos, y los modelos NF se probaron en otros 500 conjuntos de datos. Demostraron que ANFIS era el sistema con mejor rendimiento en términos de RMSE. La Figura 3.2 muestra la clasificación del sistema neuro difuso.

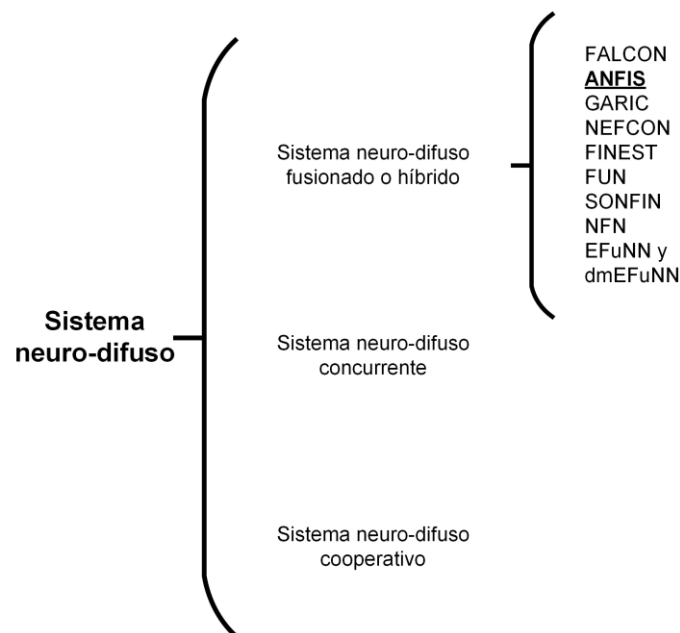


Figura 3.2 Clasificación de sistemas neuro difusos.

Para que los sistemas neuro difusos se consideren altamente inteligentes, deben cumplir unos requisitos específicos. Entre ellos se encuentran la capacidad de almacenamiento y recuperación para un aprendizaje rápido, la adaptación de nuevas características como entradas, salidas, nodos y conexiones para una adaptabilidad en línea, el logro de una tasa de error global baja y la minimización del coste computacional [32].

La adquisición y el preprocesamiento de los datos de entrenamiento también desempeñan un papel importante en el éxito de los sistemas neuro difusos. ANFIS ha demostrado cumplir muchas de estas características, lo que lo convierte en una

opción popular para los investigadores que trabajan en problemas de regresión, modelado, predicción y control.

Aunque ANFIS se ha aplicado ampliamente en múltiples áreas de la ingeniería, pocos estudios se han centrado en su aplicación en electrónica de potencia, especialmente en inversores, y en la reducción de la THD. Este estudio aborda este vacío revisando las aplicaciones recientes de las redes ANFIS en inversores multinivel. Los objetivos de este estado del arte son:

- 1) Investigar aplicaciones recientes de ANFIS aplicadas en inversores multinivel.
- 2) Observar el comportamiento y los diferentes algoritmos de entrenamiento que utilizan varios investigadores.
- 3) Explorar tendencias de investigación prometedoras para las aplicaciones de redes ANFIS en inversores multinivel.

Este estudio pretende identificar los problemas y retos a los que se enfrentan los investigadores a la hora de implementar redes ANFIS en un inversor multinivel y proporcionar una base para futuras investigaciones sobre ANFIS en electrónica de potencia.

3.1 Antecedentes: Aplicaciones ANFIS

Para comenzar la investigación, se realizó una búsqueda de revisiones sistemáticas, encuestas y estados del arte centrados en el uso de redes ANFIS, también conocidas como redes neuro difusas, aplicadas en diferentes áreas. Los artículos encontrados se clasificaron en la Tabla 3.1.

Tabla 3.1 Revisiones de la literatura anteriores sobre redes ANFIS o sistemas neuro difusos.

Ref.	Año	Título del artículo	Área	Periodos de revisión	Artículos revisados
[41]	2014	"Applications of fuzzy neuro systems: A brief review and future outline"	Diferentes temas/aplicaciones	2002-2012	187
[42]	2015	"Survey on Neuro-Fuzzy Systems and their Applications in Technical Diagnostics and Measurement"	Diagnóstico y Medición	1996-2015	27
[43]	2016	"A Review of Training Methods of ANFIS for Applications in Business and Economics"	Negocios y Economía.	2009-2014	27
[44]	2016	"ANFIS based four quadrant chopper control of separately excited DC motor: A literature review"	Motores	1993-2015	31

[13]	2018	“Recent advances in Neuro-fuzzy system: A survey”	Diferentes temas/aplicaciones	2000-2017	293
[45]	2018	“Maximum power point tracking using Hill Climbing and ANFIS techniques for PV applications: A review and a novel hybrid approach”	Aplicaciones fotovoltaicas (FV)	2000-2018	166
[19]	2018	“Adaptive network-based fuzzy inference system (ANFIS) training approaches: a comprehensive survey”	Diferentes temas/aplicaciones	1993-2016	278
[46]	2019	“Review of an ANFIS Methodology-Based Stock Market Prediction System”	Un sistema de previsión bursátil	1994-2013	44
[14]	2020	“Fuzzy neural networks and neuro-fuzzy networks: A review the main techniques and applications used in the literature”	Diferentes temas/aplicaciones	1984-2019	422
[47]	2021	“ANFIS modeling of biodiesels physical and engine characteristics: A review”	Motores con biodiésel	2012-2020	86
[48]	2022	Applications of fuzzy hybrid techniques in construction engineering and management research	Ingeniería de la construcción y gestión de la investigación	2004-2021	255
[49]	2022	Survey on adaptative neural fuzzy inference system (ANFIS) architecture applied to photovoltaic systems	Aplicaciones fotovoltaicas (FV)	2000-2021	179
[23]	2023	Adaptive network-based fuzzy inference system (ANFIS) applied to inverters: a survey	Aplicado a inversores multinivel	2010-2023	59

Una revisión realizada por los autores en [41] fue la primera en analizar exhaustivamente 187 artículos publicados entre 2002 y 2012. La revisión clasificó las aplicaciones en diez grandes campos: Modelización de Estudiantes, Sistemas Médicos, Sistemas Económicos, Control de Tráfico, Procesamiento de Imágenes y Extracción de Características, Predicción y Pronóstico, Fabricación y Modelización de Sistemas, Previsión y Predicción, Sistemas Eléctricos y Electrónicos, y Ciencias Sociales y Mejora de NFS.

En [42], los autores realizaron un análisis exhaustivo de 27 estudios publicados entre 1996 y 2015. Los estudios se centraron en clasificar las aplicaciones en dos áreas clave: diagnóstico técnico y medición. El artículo también ofrece una visión concisa de la historia de los sistemas neuro difusos y destaca las seis estructuras distintivas más cruciales que poseen los sistemas neuro difusos.

En [43], los autores revisaron 27 artículos sobre aplicaciones de ANFIS en economía y empresa publicados entre 2009 y 2014. En este artículo, se centraron más en los métodos de optimización y entrenamiento de ANFIS utilizados para resolver problemas empresariales y económicos. Esta revisión, por primera vez, se centra por completo en las redes ANFIS.

En [44], el autor recopiló 31 artículos relacionados con aplicaciones de motores de CC excitados por separado y su reciente aplicación a ANFIS entre 1993-2015. En este artículo, el autor presentó una revisión bibliográfica sobre los diferentes métodos utilizados para controlar la velocidad de un accionamiento de motor de CC. El autor concluyó que los controladores ANFIS son más robustos y eficientes que otras técnicas.

En [13], los autores revisaron 293 artículos entre 2000 y 2017. Este artículo clasificó los trabajos en tres campos: entrenamiento basado en algoritmos, método difuso y estructura. Estos autores proporcionaron comentarios y observaciones de los sistemas neuro difusos en función de algunos parámetros como la precisión, la complejidad, la clasificación y la capacidad de regresión, el tiempo de convergencia, entre otros. Llegaron a la conclusión de que comparar las distintas arquitecturas y evaluar su rendimiento requiere mucho trabajo.

En [45], los autores recopilaron 166 artículos relacionados con aplicaciones fotovoltaicas (FV) entre 2000-2018. Los autores se centraron en técnicas de seguimiento del MPP offline y online. Además, se revisaron técnicas híbridas, demostrando sus ventajas e inconvenientes. Finalmente, los autores desarrollaron una técnica híbrida de seguimiento del punto de máxima potencia (MPPT) basada en ANFIS Hill Climbing (HC).

En [19], los autores revisaron 278 artículos sobre ANFIS entre 1993-2016. En este trabajo, los autores se centraron más en los métodos de optimización en el entrenamiento de ANFIS utilizados en diferentes áreas de la ingeniería. Los autores clasificaron sus referencias en función del algoritmo que utilizaron para entrenar los parámetros de primacía y consecuencia.

En [46], los autores recopilaron 44 trabajos sobre aplicaciones a sistemas de predicción bursátil entre 1994 y 2013. En este trabajo se evaluaron los paradigmas más utilizados en la predicción bursátil, como la lógica difusa y las redes neuronales. A continuación, los autores desarrollaron un ANFIS para este tipo de aplicaciones.

En [14], los autores revisaron 422 artículos desde 1984 hasta 2019. Este artículo ofrece una vasta historia de los modelos inteligentes y los divide en varios campos importantes, como la salud, la economía, la industria y las finanzas. Presenta las principales funcionalidades de las arquitecturas, los métodos de actuación y la minería de datos. También ofrece una organización de sus principales métodos de entrenamiento.

En [47], los autores revisaron 86 artículos de 2012 a 2020. Se centraron en la búsqueda de documentos ANFIS para medir y analizar con precisión el rendimiento del motor y las emisiones de escape del motor cuando se utiliza biodiésel como combustible alternativo. Además, los autores proporcionaron una comparación entre diferentes métodos y ANFIS. Llegaron a la conclusión de que la predicción de las redes ANFIS es superior a la de las redes neuronales. Si se combinan PSO o el GA, se obtienen resultados mucho más optimizados.

En [48], los autores revisaron 255 artículos sobre técnicas híbridas difusas aplicadas a la Ingeniería y Gestión de la Construcción (CEM) entre 2004 y 2021. Los autores establecieron una lista de criterios de selección para elegir una técnica híbrida difusa adecuada para resolver un problema CEM específico.

En [49], los autores revisaron 179 artículos entre 2000 y 2021. Los autores se centraron en la búsqueda de trabajos sobre ANFIS aplicados en sistemas fotovoltaicos. Los autores los clasificaron en seis categorías principales diferentes, que serían:

Predicción de la irradiancia solar; Estimación de la potencia de salida FV; Identificación de parámetros para el dimensionamiento del sistema FV; MPPT; Control del inversor; y Diagnóstico de fallos del sistema FV.

Los autores analizaron a fondo los trabajos categorizados, incluyendo una comparación dentro de cada categoría. Los autores compararon ANFIS con otras técnicas para evaluar sus puntos fuertes y débiles. Aunque se dedica una categoría de estudio a los inversores, son pocos los trabajos dedicados exclusivamente a este tema. Cabe destacar que los autores compararon otros métodos de IA en su investigación, lo que significa que los trabajos obtenidos no se basan exclusivamente en ANFIS.

Los estudios de la revisión bibliográfica proporcionan diversas aplicaciones de las redes ANFIS en diferentes campos, como la economía, salud, motores, sistemas fotovoltaicos y construcción. Sin embargo, al analizar a fondo estas investigaciones, se identificó que la mayoría se enfocan en aplicaciones generales o temáticas específicas, sin abordar de forma concreta la aplicación de ANFIS en inversores multinivel.

Aunque se han realizado varias revisiones y estudios sistemáticos, solo [19], [41], [13], y [14] mencionan algunos artículos centrados en la electrónica de potencia. Incluso aquellos estudios relacionados con sistemas fotovoltaicos, como [45] y [49], presentan una categoría dedicada al control de inversores, pero no se enfocan específicamente en el uso de redes ANFIS para disminuir la THD como técnica principal.

Con tan pocos trabajos enfocados a electrónica de potencia, se requiere una revisión exhaustiva y especializada en el uso de redes ANFIS aplicadas a inversores multinivel. Este enfoque permite identificar tendencias específicas, brechas de conocimiento y metodologías recurrentes en esta aplicación específica, proporcionando una base sólida para el desarrollo de investigaciones futuras. La

Tabla 3.1, por tanto, pone en evidencia que, hasta la fecha, solo el trabajo presentado en [23] ha abordado directamente esta temática, posicionándose como el único estudio enfocado exclusivamente en ANFIS aplicado a inversores multinivel.

3.2 Metodología de revisión sistemática de la literatura

Para llevar a cabo esta revisión se siguieron cuatro pasos principales: investigación de antecedentes, revisión sistemática de la literatura, análisis de los resultados y conclusiones.

La metodología para la revisión sistemática incluyó criterios de identificación, selección, elegibilidad e inclusión que permitieron seleccionar las publicaciones de las diversas revistas y bases de datos relevantes consultadas. La Figura 3.3 muestra un esquema de la metodología aplicada

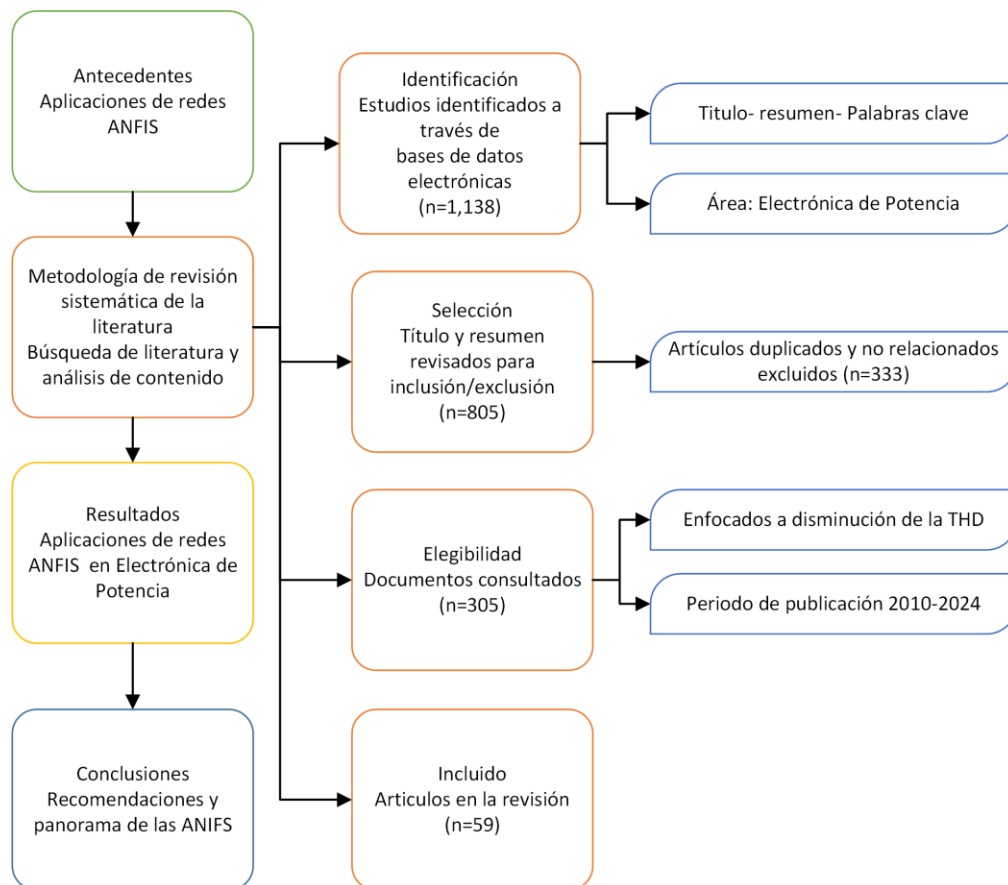


Figura 3.3 Metodología de revisión sistemática de la literatura.

3.2.1 Selección de artículos de revistas y ponencias de conferencias

En esta sección se expone la metodología empleada para buscar y cubrir artículos relevantes sobre aplicaciones de redes ANFIS en inversores multinivel. Para ello, en la etapa de identificación, se aplicaron criterios de selección que incluían palabras clave específicas, como: “Adaptive Neuro Fuzzy Inference System”, “Multilevel Inverter”, “THD”, “Harmonic Elimination”, “Optimization”, “Power Converter”, y “Power Electronics” en el título, resumen o palabras clave de artículos, revistas y actas de congresos en el área de la electrónica de potencia. Para la fase de selección, sólo se tuvieron en cuenta las publicaciones periódicas o regulares en lengua inglesa y con un proceso de revisión por pares. Se descartaron los artículos duplicados y no relacionados.

En cuanto a la elegibilidad de las publicaciones, se consideraron criterios de inclusión las publicaciones centradas en la reducción de THD en convertidores multinivel utilizando ANFIS. El periodo de publicación considerado fue de 2010 a 2024.

Finalmente, se identificaron 59 artículos, agrupados principalmente en 16 conocidas revistas de electrónica e informática/software. En la Figura 3.4 se muestra un histograma de las publicaciones relacionadas con redes ANFIS aplicadas a inversores multinivel.

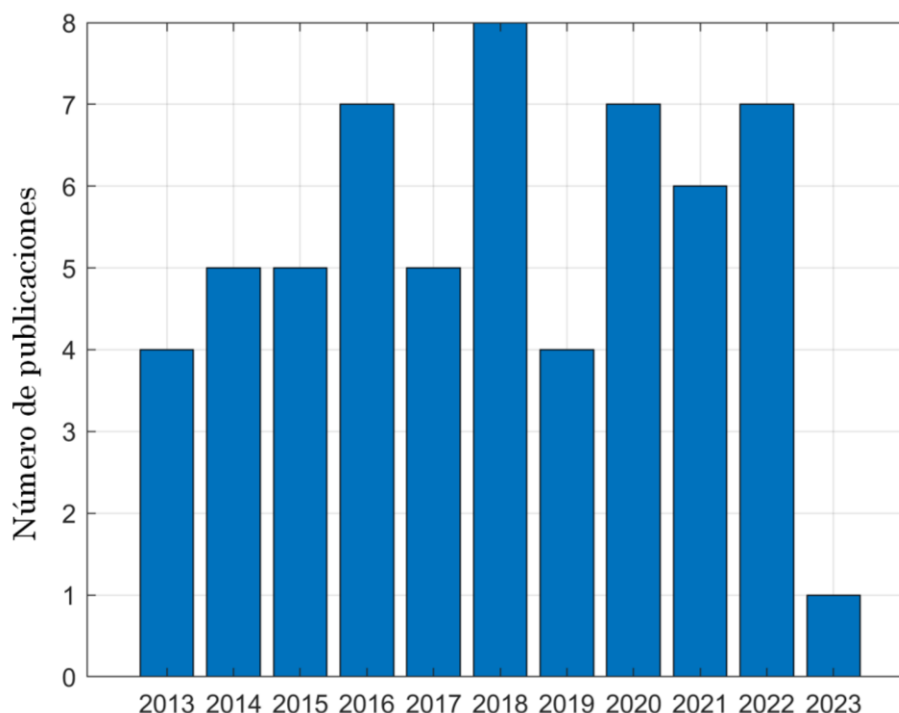


Figura 3.4 Artículos encontrados sobre el uso de redes ANFIS en inversores multinivel.

La Tabla 3.2 presenta los trabajos encontrados sobre ANFIS aplicados a MLI.

Tabla 3.2 Documentos encontrados sobre ANFIS aplicado a MLI.

Algoritmo de entrenamiento	Ref.	Año	Aplicación / Aspectos destacados	Plataforma
Gradiente	[50]	2013	FV/ 3er armónico eliminado.	PIC MCU
	[51]	2013	MR/control de las corrientes de fase.	SS
	[52]	2013	Controlador FV/ "Mi" y ángulos de encendido.	SS
	[53]	2013	Filtro activo de potencia en derivación (SAPF). Controlador de voltaje.	SS
	[54]	2014	FV/Controlador de MPPT.	SS
	[55]	2014	Aplicación de ANFIS/Controlador de voltaje y corriente.	PIC MCU
	[56]	2015	FV/Controlador de corriente.	FPGA
	[57]	2015	Implementación de ANFIS/Controlador de corriente (Control de las corrientes de fase).	SS
	[58]	2015	FV/Controlador de voltaje.	SS
	[59]	2015	WECS/Controlador de DFIG.	SS
	[60]	2016	Controlador de voltaje, variación Mi y carga.	SS
	[61]	2016	Controlador de voltaje.	SS
	[62]	2016	FV/Controlador de voltaje.	SS
	[63]	2016	WECS DFIG/Controlador vectorial.	SS
	[64]	2017	Controlador de voltaje / Eliminación de los armónicos 3, 5 y 7.	SS
	[65]	2017	MI /Los armónicos 5°, 7°, 11° y 13° se eliminan.	dsPIC
	[66]	2017	MI/Controlador de voltaje. Se centra más en el MPPT y el convertidor. Gradiente + LSM	DSP
	[67]	2018	MI/Controlador de velocidad.	SS
	[68]	2018	FV/Controlador de voltaje.	SS
	[69]	2018	PQ/Controlador de corriente.	SS
	[70]	2018	MI/Controlador de Torque.	SS
	[71]	2019	Aplicación de ANFIS /Controlador de voltaje y corriente.	SM
	[72]	2019	CHBMLI-SHAF/Control de la corriente de histéresis.	SS
	[73]	2019	FV/Controlador de voltaje.	SS
	[74]	2020	Implementación ANFIS/Variación de voltaje.	SM
	[75]	2020	SAPF/Controlador de corriente.	SS
	[76]	2020	FV/Controlador de corriente y SAPF.	FPGA
	[77]	2021	Inversor PEC9 /Variación de voltaje.	DSP
	[78]	2021	FV-Microred/Controlador de voltaje y corriente.	SM
	[79]	2021	Implementación ANFIS/Variación de voltaje y RMS.	OPAL-RT DSP
	[80]	2022	OMTHD/Entrenamiento ANFIS y variación de Mi.	SM
	[81]	2022	FV/Controlador de voltaje.	SS

Mínimos cuadrados	[82]	2022	Red, nueva topología/Controlador de desplazamiento de nivel.	SS
	[83]	2022	MI/Controlador de Torque.	SS
	[84]	2022	MR/Controlador de voltaje. Gradiente + LSM	SS
	[85]	2023	Rendimiento de ANFIS/Controlador de corriente.	SM
	[86]	2014	Controlador de voltaje/Variación de voltaje y carga.	SS
	[87]	2019	FV/Controlador MPPT ANFIS.	DSP
	[88]	2021	MI/Control de velocidad.	SM
	[89]	2022	SAPF/Controlador de voltaje. LSM + BP	SS
NR	[90]	2014	FV/Controlador MPPT, 5º armónico eliminado.	SM
GA	[91]	2015	MI/Controlador de velocidad. Comparación entre GA, BAT y PSO.	SM
	[92]	2016	STATCOM/ Controlador de voltaje.	SS
	[93]	2021	Variación de “Mi” y voltaje/ 5º y 7º armónicos eliminados. GA + NR	SM
RCGA	[16]	2018	Eliminación de los armónicos 5º, 7º, 11º y 13º/ Variación de Mi	SS
PSO	[94]	2014	FV/ Controlador MPPT con PSO–ANFIS. PSO + LSM	PIC MCU
	[95]	2016	MI/5º, 7º, 11º y 13º armónicos eliminados.	DSP FPGA
ABC	[96]	2017	MR/5º y 7º armónicos eliminados.	MCU
ALO	[97]	2018	Reducción de THD y Variación de voltaje/ Eliminación de 5º, 7º, 11º y 13º armónicos.	SS
FUZZY	[98]	2018	MI/OMTHD FUZZY + Resilient BP	SS
SSA	[99]	2018	WECS/Controlador DFIG	SS
	[100]	2020	MI/Controlador de velocidad y voltaje.	SM
PI	[101]	2020	SAPF/Controlador de voltaje.	DSP
EPO	[102]	2020	FV/Controlador de voltaje.	SS
HLM	[103]	2021	UPFC/Controlador de voltaje.	SS

Sin mención	[104]	2016	FV/Controlador de voltaje.	SS
	[105]	2017	FV/Controlador de corriente.	SS
	[106]	2020	MI/Controlador de voltaje y torque.	SM
	[107]	2022	MR/Controlador de voltaje y corriente.	OPAL-RT DSP

Las siglas utilizadas en la Tabla 3.2 son las siguientes: Solo Simulación (SS), Sin Mención (SM), Micro-Red (MR), Peripheral Interface Controller (PIC), Microcontrollers (MCU), Genetic Algorithm (GA), Real Coded Genetic Algorithm (RCGA), Salp Swarm Optimization Algorithm (SSA), Emperor Penguin Optimizer (EPO), Artificial Bee Colony (ABC), Ant Lion Optimizer (ALO), Least Square Method (LSM), Hebbian Learning Model (HLM), Unified Power Flow Controller (UPFC), Back Propagation (BP), Optimal Minimization of the Total Harmonic Distortion (OMTHD). Al final de la Tabla 3.2, la línea denominada "Sin mención" agrupa los elementos que no proporcionan información sobre el algoritmo de entrenamiento utilizado.

La Tabla 3.2 proporciona información importante sobre el año de publicación, el algoritmo de entrenamiento ANFIS utilizado, el área de aplicación, los armónicos eliminados y la plataforma utilizada. Este estudio del estado del arte abarcó revistas de varias bibliotecas en línea y bases de datos de editoriales de renombre, como Elsevier, IEEE Transactions y Taylor & Francis Group, donde se identificaron tanto investigaciones puramente simuladas como estudios experimentales que reportan las plataformas utilizadas.

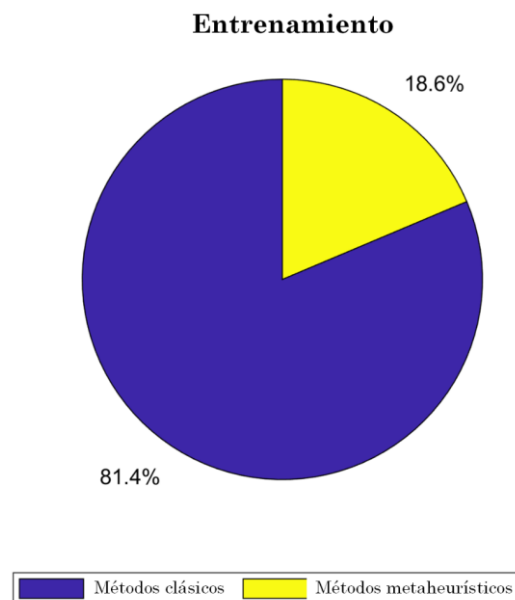


Figura 3.5 Distribución de los métodos de entrenamiento utilizados para ANFIS.

Cabe señalar que el método de entrenamiento de ANFIS utilizado en estos estudios desempeña un papel crucial, ya que la mayoría de los autores optan por métodos clásicos como los algoritmos de GD, NR y mínimos cuadrados. Sin embargo,

algunos autores también han experimentado con métodos metaheurísticos que simulan la selección natural o la adaptación social para explorar el espacio de búsqueda utilizando diversas estrategias de forma eficaz [98]. Los métodos metaheurísticos incluyen PSO, ALO, ABC, SSA y GA y sus variantes. Aunque el procedimiento realizado por los métodos metaheurísticos suele ser menos riguroso que el de los métodos clásicos, ofrecen un enfoque alternativo para el entrenamiento de ANFIS. En la Figura 3.5 se presenta un gráfico que ilustra la distribución de estos algoritmos; como puede observarse, el 81,4% de las publicaciones utilizaron métodos clásicos, mientras que alrededor del 18,6% emplearon métodos metaheurísticos.

En electrónica de potencia, ANFIS ha sido ampliamente empleado por los autores para los paneles fotovoltaicos y las redes, y algunos investigadores dan fe de su eficacia en los inversores multinivel.

3.2.2 Características importantes de los inversores

Esta sección muestra algunos gráficos que destacan los tipos de inversores y sus características utilizados en los trabajos encontrados.

El objetivo de esta sección es ofrecer una visión general del contexto tecnológico y académico en el que se enmarca el estudio, mostrando cómo los diferentes tipos de inversores y sus características han sido abordados en la literatura reciente.

Por ello, los gráficos y clasificaciones presentados no pretenden sustentar una elección técnica, sino ilustrar las tendencias, enfoques y aplicaciones más comunes reportadas en investigaciones previas, con el fin de contextualizar el desarrollo del presente trabajo dentro del panorama actual de los inversores multinivel.

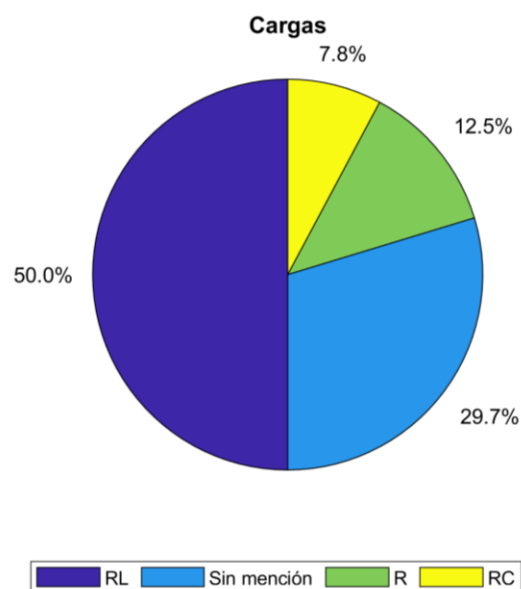


Figura 3.6 Distribución de las publicaciones por tipo de cargas.

La Figura 3.6 presenta un análisis detallado de la distribución de los distintos tipos de carga conectados a la salida del inversor. Los resultados indican que prevalece el tipo de carga RL de las publicaciones, que constituye el 50% de los casos estudiados. Además, el 29.7% de las investigaciones requerían detalles específicos sobre el tipo de carga utilizado en sus análisis. En cuanto a las configuraciones de carga, el 12.5% de las publicaciones emplearon una carga de tipo R, mientras que el 7.8% optaron por otras configuraciones, principalmente cargas RC. Estos resultados ponen de relieve la importancia de los tipos de carga RL y R en las investigaciones examinadas, aunque otras configuraciones, incluidas las cargas RC, también desempeñan un papel notable.

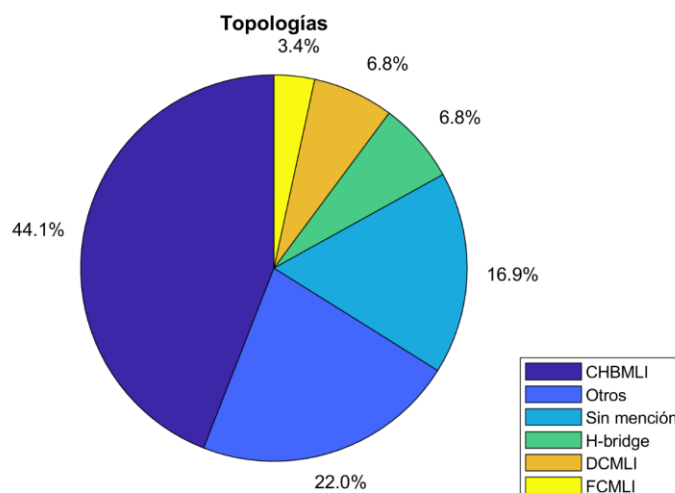


Figura 3.7 Distribución de las topologías utilizadas en el universo de publicaciones.

La Figura 3.7 presenta un análisis detallado de las preferencias en las topologías de inversores utilizadas en las distintas publicaciones. En este contexto, cabe destacar que el 44.1% de los trabajos optaron por la topología del inversor multinivel de puente en H en cascada (CHBMLI). Un porcentaje del 22% correspondió a otras topologías que no pertenecían a las convencionales, como el inversor Z, STATCOM, SAPF o versiones modificadas de estas. Además, el 16.9% de los estudios no proporcionaba información específica sobre las topologías utilizadas. Por otro lado, el 6.8% de los estudios optó por implementar inversores de puente H convencionales.

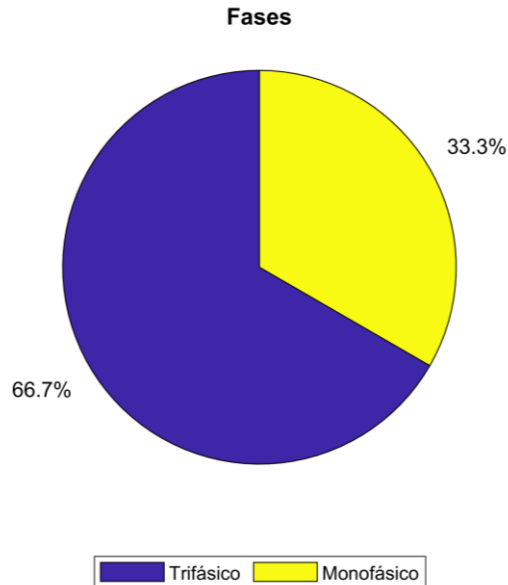


Figura 3.8 Distribución de las publicaciones del número de fases.

La Figura 3.8 muestra que el 66.7% de las publicaciones utilizaban un inversor trifásico y el 33.3% un inversor monofásico.

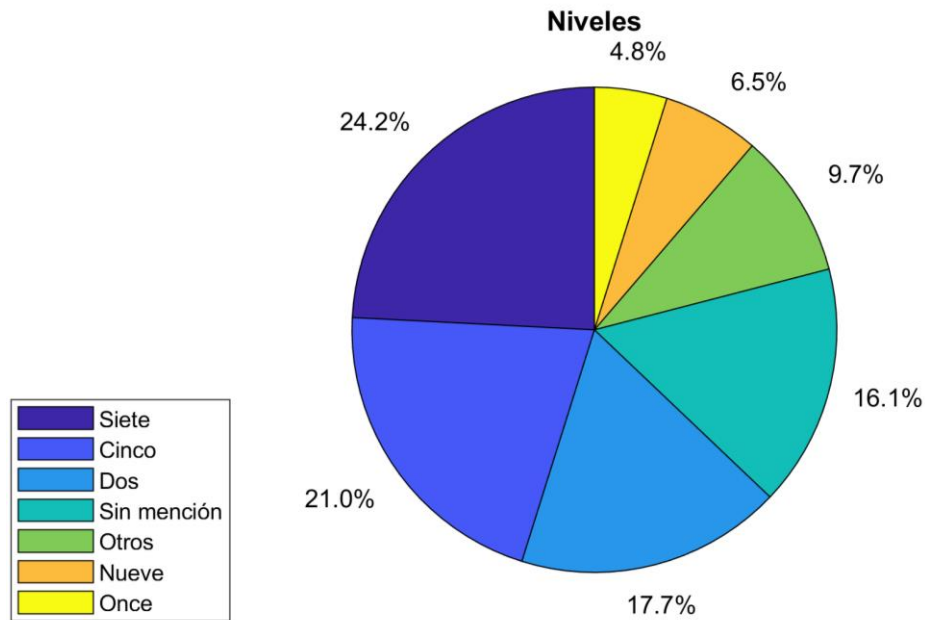


Figura 3.9 Distribución de las publicaciones en función del número de niveles.

La Figura 3.9 muestra un análisis detallado de las preferencias en cuanto a los niveles de inversores utilizados en las publicaciones. En este sentido, el 24.2% de los trabajos utilizó inversores multinivel de siete niveles, mientras que el 21% optó por inversores de cinco niveles. Del mismo modo, el 17.7% utilizó inversores de dos niveles. Aunque los inversores de dos niveles, más conocidos como inversores

convencionales, no se clasifican estrictamente como multinivel, se tuvieron en cuenta en el análisis.

Un segmento de especial interés corresponde al 16.1% de los trabajos que no proporcionan información explícita sobre el número de niveles de inversor empleados en su investigación. Además, destaca una sección en el gráfico etiquetada como: Otros, que constituye el 9.7%. Esta categoría alude a niveles distintos de los mencionados anteriormente (siete, cinco y dos). Es importante señalar que, aunque esta sección incluye niveles como nueve y once, el uso de niveles superiores a once es menos frecuente. Por ello, se decidió agrupar estos casos bajo la etiqueta “Otros”.

3.3 Discusión del Estado del Arte

La metodología de búsqueda de artículos sobre este tema constó de cuatro pasos: antecedentes, método, resultados y discusión. Tras filtrar los artículos irrelevantes y clasificar el resto, se obtuvo un total de 59 entre los años 2013 y 2023. Cabe destacar que en torno a 2010 y 2013 se aplicó por primera vez ANFIS a inversores multinivel centrados en la reducción de THD. Según la Figura 3.4, el número de artículos que aplican ANFIS a inversores multinivel ha aumentado. Los artículos más publicados fueron los de 2018, seguidos de los de 2016, 2020 y 2022, lo que indica que los inversores despiertan cada vez más interés en la investigación de ANFIS.

La Tabla 3.2 pretende ofrecer más información sobre los artículos encontrados y sus características, así como destacar el tipo de algoritmo utilizado para el entrenamiento de ANFIS. La mayoría de los autores utilizaron métodos clásicos, concretamente el método del gradiente descendente, mientras que sólo unos pocos emplearon técnicas metaheurísticas. La mayoría de los autores utilizaron métodos clásicos, concretamente el método del gradiente descendente, mientras que solo unos pocos emplearon técnicas metaheurísticas. Sin embargo, en 2014, algunos autores empezaron a aplicar la metaheurística en sus investigaciones.

En la Figura 3.10, se ha creado un gráfico para ofrecer una visión más clara del comportamiento de los métodos empleados para el entrenamiento de ANFIS, mostrando el paso del tiempo (2013 a 2023) y el número de métodos empleados, ya sean clásicos o metaheurísticos.

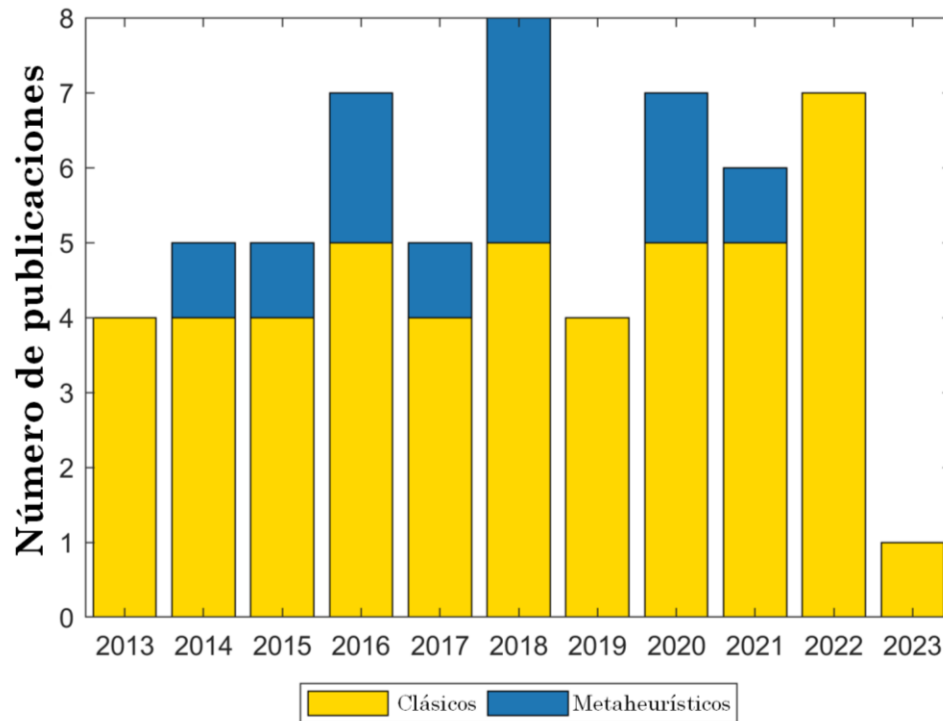


Figura 3.10 Distribución de publicaciones Años vs. Algoritmos.

Observando la Figura 3.10, es evidente que los algoritmos clásicos (indicados por el color amarillo) se han utilizado predominantemente a lo largo de los años. Sin embargo, los algoritmos metaheurísticos (indicados por el color azul) debutaron en 2014 y ganaron importancia en 2018. Por lo tanto, cabe destacar que, mientras que muchos investigadores siguen prefiriendo los algoritmos clásicos, los algoritmos metaheurísticos solo se han empleado recientemente para el entrenamiento de ANFIS.

Los investigadores han utilizado ANFIS en diversas aplicaciones, como paneles solares, motores de inducción, sistemas eólicos y microrredes. Algunos autores han realizado experimentos alterando las tensiones de entrada, el índice de modulación (Mi) o las cargas a la salida para evaluar el rendimiento y el comportamiento de ANFIS. Muchos autores han encontrado que ANFIS es altamente eficiente en cuanto a tiempo de respuesta y aproximación en tiempo real del valor esperado que necesitan sus respectivos sistemas.

En este estudio, las aplicaciones de ANFIS tienen como objetivo principal evaluar la eficacia y viabilidad de las redes ANFIS en inversores. La Figura 3.11 muestra un gráfico de correlación entre la aplicación y los algoritmos de entrenamiento.

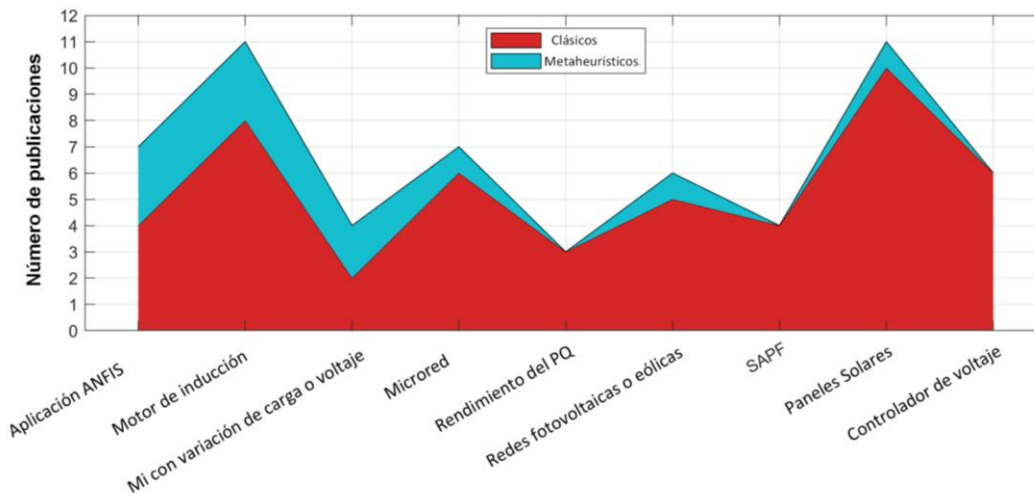


Figura 3.11 Distribución de publicaciones: Aplicación frente a algoritmos.

En la Figura 3.11, se presentan nueve categorías de aplicación donde se ha implementado ANFIS en electrónica de potencia. El gráfico permite observar la cantidad de publicaciones según el tipo de técnica utilizada: métodos clásicos o metaheurísticos. Se destaca que la mayoría de las investigaciones emplean métodos clásicos, mientras que los algoritmos metaheurísticos han sido menos frecuentes y, en su mayoría, aplicados solo en fases de entrenamiento. Solo en algunos casos, como en motores de inducción, se han aplicado metaheurísticos en entornos reales. Esta tendencia evidencia una brecha en la implementación práctica de técnicas avanzadas como PSO en combinación con ANFIS, lo que representa una oportunidad para futuras investigaciones.

Tras analizar y exponer los diversos aspectos significativos de la aplicación de ANFIS en inversores, el contenido del artículo se centra en destacar las características específicas relacionadas con dichos inversores. Esto abarca aspectos como las cargas conectadas a la salida, las topologías empleadas, el número de fases del inversor y el número de niveles que han utilizado los investigadores. El objetivo principal es orientar a los lectores hacia las áreas de innovación y exploración predominantes en el campo de los inversores.

Cabe destacar que la mayoría de los investigadores optaron por emplear inversores trifásicos en sus estudios. Este enfoque refleja la importancia y la aplicabilidad de los inversores trifásicos en diversas aplicaciones.

En cuanto al número de niveles, una proporción significativa de investigadores optó por implementar inversores de siete y cinco niveles. Además, existe una categoría denominada otros, que engloba niveles superiores a quince. Cabe señalar que algunos autores omitieron mencionar el número de niveles que utilizaron en sus investigaciones.

En resumen, esta exhaustiva exploración pretende ofrecer una visión detallada de las tendencias y preferencias predominantes en materia de inversores. También

pone de relieve las posibles áreas de desarrollo y la futura ampliación de conocimientos en este campo.

La Tabla 3.3 muestra los porcentajes de THD obtenidos en las referencias y los datos importantes para el entrenamiento de las redes ANFIS utilizadas por los investigadores.

Tabla 3.3 Datos obtenidos para el entrenamiento y el porcentaje de THD.

Ref.	Características ANFIS/ Destacados
[50]	La red ANFIS entrenada tiene un error de entrenamiento insignificante de 0.00066062. Tercer armónico eliminado.
[51]	THD en las corrientes obtenidas experimentalmente: L-A=2.9%, L-B=2.7%, L-C=2.7%.
[52]	THD obtenida en la simulación: P&O MPPT =4.97%, IC MPPT=3.48%.
[53]	Funciones de membresía =7. THD obtenida en simulación a $\alpha=90^\circ$: Sin SAF= 24.94%, PI=2.18%, FUZZY=1.08, ANFIS=0.48%.
[86]	THD obtenida en la simulación NFC $\approx$ 11.77%, ANFIS $\approx$ 9.16%.
[90]	MI=0.95. No muestra % de reducción de THD, se centra en MPPT.
[54]	MI=0.95. No muestra % de reducción de THD, se centra en MPPT.
[55]	25 reglas difusas. THD obtenida experimentalmente: Controlador PI= 4.78%, FUZZY= 1.93%, ANFIS=1%.
[94]	Datos recogidos=4000, Tolerancia de errores=0.001, Épocas=2500. THD obtenida en la simulación: PSO-ANFIS=29.13%, P&O-ANFIS= 47.54%, RNA=59.16%. Resultado de la alimentación de CC y CA. Análisis de las condiciones de sombra y no sombra
[56]	THD obtenida en la simulación: RNA=5.29%, FUZZY=1.48%, ANFIS=1.17%. THD obtenida experimentalmente: RNA=8.7%, FUZZY=15.6%, ANFIS=1.32%.
[57]	THD en corriente obtenida en simulación 2.7%
[58]	THD en corriente obtenida en simulación: Normal (Sin controlador) =2.7%, Red Neuronal= 5.29%, FUZZY=1.48%, ANFIS=1.17%.
[59]	Épocas=40, Funciones de membresía =49, Datos de entrenamiento=10,000, Datos de prueba=10,000. THD obtenida en la simulación=23.89%.
[91]	Se centra en el comportamiento y el rendimiento de ANFIS con metaheurísticas para la velocidad, sin tener en cuenta los armónicos.
[60]	THD obtenida en la simulación: RNA=14.57%, NR=11.17%, PSO=15.37%, NF=11.77%, ANFIS=9.16%.
[95]	20 épocas, el 75% de los datos registrados se utilizaron para el entrenamiento, el 15% para las pruebas y el 15% para la validación del modelo ANFIS. THD obtenida en la simulación: utilizando Mi=0.8, NR= 5.55%. Con Mi=0.9, PSO=4.79%. THD obtenida experimentalmente: utilizando Mi=0.8, NR= 4.8%. Con Mi=0.9, PSO=3.7%.
[61]	THD obtenida en la simulación: THD _v = 1.36%, THD _i = 0.78%.
[62]	Tasa de aprendizaje 0,02. THD obtenida en la simulación=18.21%.
[63]	THD obtenida en la simulación 4.84%

[104]	Frecuencia variable PWM. THD obtenida en la simulación: PI=7.67%, FUZZY=5.03%, ANFIS=3.3.62%, PID-PSO=2.68% APOD*
[92]	Redes ANFIS= 3, Épocas=100, Funciones de membrecía =15. Muestra el comportamiento y las prestaciones de las variaciones de tiempo y voltaje.
[96]	30 épocas. THD obtenida en la simulación: Mi=1.08, THD _{Phase} =11.7%, THD _{Line} =7.7%. THD obtenida experimentalmente: Mi=1.08, THD _{Phase} =10.95%, THD _{Line} =9.12%.
[64]	Se eliminan los armónicos: 3 ^a , 5 ^a y 7 ^a
[65]	Simulación THD=1.4%, Experimental THD= 1.2%
[105]	THD obtenida en simulación utilizando PWM =457.22%, SPWM =13.31%, Neuro Fuzzy con Rate-Based Regulation-SVPWM=0.71%.
[97]	THD obtenida en la simulación: ALO -RNA=12.95%, GA -RNA=14.91%, PSO-ANFIS=12.95%, ALO-ANFIS=14.91%
[98]	Tres neuronas de entrada, diez neuronas ocultas y siete neuronas de salida. Tasa de aprendizaje 0.95. Número de iteraciones 200. THD obtenida en la simulación: PID=15.32%, ANFIS=14.56%, PID+ANFIS=13.12%, DRSS=12.10%, RDNN=9.26%
[16]	Número de nodos=204, Número de parámetros lineales=50, Número de parámetros no lineales=200, Número total de parámetros=250, Número de pares de datos de entrenamiento=241, Número de pares de datos de control=241, Número de reglas difusas= 50. THD obtenida en la simulación: THD _{Line} =4.26%, THD _{Phase} =13.09%.
[67]	THD obtenida en la simulación: PID=6.34%, ANFIS=5.82%, Sin control=7.42%.
[68]	THD obtenida en la simulación Generación FV: FUZZY=2.13%, ANFIS=1.82%. No generación FV: FUZZY=0.39%, ANFIS=0.37%.
[69]	THD obtenida en la simulación: Equilibrado no lineal, Pi=29.73%, FUZZY=29.74%, ANFIS=29.76%. No lineal desequilibrada, Pi=20.65%, FUZZY=20.64%, ANFIS=18.01%
[70]	25 reglas difusas. THD obtenida en la simulación: 5-niveles MLI, PI=28.51%, FUZZY=19.71%, ANFIS=15.41%. 7-niveles, PI=22.88%, FUZZY=16.20%, ANFIS=12.68%, 9-niveles, PI=15.74%, FUZZY=13.01%, ANFIS=11.92%
[99]	Funciones de membrecía =6. Comprobación del método DVR de voltaje del estátor basado en TSBC-QMLI y ANFISC-MOSAA
[87]	Se proponen cinco conjuntos ANFIS para las variables de entrada/salida. La red se entrena durante 30 épocas, y el error de trayectoria es del 0%. THD obtenida en la simulación Red=0.06%. THD obtenida experimentalmente Red=2.5%.
[71]	Reglas difusas=20. THD obtenida en la simulación PI= 9.02%, ANFIS=5.01
[72]	THD obtenida en la simulación. Sin filtro=47.94%, RNA=2.53%, ANFIS=2.24%
[73]	THD obtenida en la simulación. Alta generación FV, Carga=1.4, Fuzzy=3.19%, ANFIS=3.02%, Sin generación FV, Carga=1.4 Fuzzy=1.4%, ANFIS=1.16%
[74]	Estructura del controlador ANFIS con una base de reglas de 5 x 5 para la entrada y la salida. THD obtenida en la simulación Corriente SV=3.18%-3.23% -3.30%. Corriente NSSV= 3.58%-3.66%-3.97%
[75]	THD obtenida en la simulación: Línea de corriente en sistema de carga no lineal= 0.49%
[76]	THD obtenida en la simulación:

	Carga R Sin filtro=25.05%, ANFIS=2.92%, Widrow–Hoff (WF) ADELIN=1.81%, Modified Widrow–Hoff (MWH) ADELIN=1.75%. Carga RL Sin filtro=24.77%, ANFIS=2.48%, WH ADELIN=1.86%, MWH ADELIN=1.75%. Carga RC Sin filtro=39.51%, ANFIS=3.32%, WH ADELIN=2.28%, MWH ADELIN=2.20%. THD obtenida experimentalmente: ANFIS=4.78%, WH ADELIN=3.89%, MWH ADELIN=3.21%.
[101]	Reglas difusas=9. THD obtenida en la simulación: Sin control=44.55%, BSMC=3.83%, ANFIS=2.96%
[102]	THD obtenida en la simulación: PI=18.19%, PID=16.11%, ANFIS=0.78%
[100]	Centrarse en la velocidad y el par motor. Comparación entre GOA, SSA, GORNNN, y SSA-ANFIS
[106]	Muestra la velocidad del par, el rotor y el rendimiento del estator utilizando FUZZY, ANFIS, NARMC, y los métodos AVRC.
[77]	THD obtenida experimentalmente: Voltaje=2.2%, Corriente=1.9%.
[78]	THD obtenida en la simulación: Red PI=1.57%, Carga=1.57%. Red FUZZY=1.57%, Carga=1.57%. Red ANFIS=1.58%, Carga=1.58%
[93]	Se calcularon 675 muestras, 540 se utilizaron para el entrenamiento y 135 para evaluar el sistema. THD obtenida en la simulación=8.1% Mi=0.8. THD obtenida experimentalmente= 7.6% Mi=0.8.
[79]	1000 muestras de datos de entrenamiento y 49 reglas difusas. 1000 puntos durante el periodo de entrenamiento. THD obtenida en la simulación: Sin control= 4.97%, FUZZY= 4.15%, ANFIS=3.77%. THD obtenida experimentalmente: Sin control=5.03%, FUZZY=4.26%, ANFIS=3.98%
[88]	Funciones de miembro=7, Función de activación=tansing, Iteraciones =1000, Tolerancia de error 0.01. Se centra en observar el comportamiento. THD obtenida en la simulación: 2-niveles=5.78% ;5–niveles=4.04%
[103]	Reglas difusas= 25. Se centra en el rendimiento energético comparando ANFIS con PI, FUZZY y RNA.
[80]	Se obtuvo un total de 340 datos para el valor THD de la corriente de salida. 272 datos, de los cuales se utilizaron en el entrenamiento y 68 de ellos en la clasificación de prueba. THD en la corriente obtenida en la simulación=5.46%
[81]	Iteraciones 3000, capas de entrada 2, capas ocultas 10, capas de salida 1, tasa de aprendizaje 0.05. THD obtenida en la simulación: PI =8.34%, FLC =5.65%, RNA= 8.18%, ANFIS= 4.04%
[82]	Usando LS-PWM. THD obtenida en la simulación: PI THD _i = 5.22% THD _v =0.01%. FIS THD _i = 2.92% THD _v =0.01%. ANFIS THD _i = 2.78% THD _v =0.01%. Usando PS-PWM, PI THD _i = 2.16% THD _v =0.01%
[83]	70% para el conjunto de datos de entrenamiento y 30% para las pruebas. 171 ms de promedio total de entrenamiento 7,8 ms promedio de tiempo de predicción. THD obtenida en la simulación: FUZZY= 14.32%, ANFIS=11.73%, Convolutional Long Short-Term Memory-direct torque control (ConvLSTMDTC)= 6.61%
[89]	Tipo de función de membresía= Gauss, Número de épocas=30, Nodos totales=131, Parámetros lineales(no.) = 49, Parámetros no lineales(no.) =42, Número total de parámetros=91, Pares de datos de entrenamiento= 5288, Reglas difusas= 49. THD

	obtenida en la simulación RL1 THD _{ig} =1%, THD _{il} =27.12%. RC THD _{ig} =1.05%, THD _{il} =30.18%. RL2 THD _{ig} =0.46%, THD _{il} =18.61%
[107]	ANFIS se utiliza para el controlador del inversor. Se centra en la gestión de la energía y la reducción de costes.
[84]	Épocas=300, Muestras de entrenamiento=2365. Muestra las velocidades del rotor, el par y otras características del motor de inducción.
[85]	THD obtenida en la simulación= Sliding Mode Controller based Synchronous Reference Frame (SRF)=3.34%, ANFIS-SRF = 1.56%

Como ya se ha mencionado, algunos investigadores se limitaron a realizar simulaciones o pruebas experimentales. Por lo tanto, este análisis presenta una visión general de los trabajos analizados.

La Tabla 3.3 revela que algunos investigadores no incluyeron datos importantes o la información necesaria para el entrenamiento de ANFIS, y sólo un pequeño grupo proporcionó detalles sobre el número de neuronas o muestras necesarias para un entrenamiento eficaz. Como ya se ha mencionado, el éxito de la red está estrechamente ligado a un entrenamiento adecuado. Aunque algunos investigadores mencionaron el número de muestras utilizadas en el entrenamiento, no está claro si eran suficientes para lograr un rendimiento óptimo de ANFIS. De ahí que sea crucial comprender el proceso de entrenamiento de ANFIS para garantizar su eficacia.

La Tabla 3.3 presenta el %THD que redujeron los investigadores. La tabla muestra que algunos investigadores se centraron en reducir el %THD de tensión, otros en el %THD de corriente, otros en el rendimiento MPPT y otros en los controladores. En general, las redes ANFIS superaron a otras formas de inteligencia artificial en cuanto a eficacia y rendimiento en la reducción del %THD. La siguiente tabla sintetiza información clave, proporcionando una visión clara de las tendencias actuales en la aplicación de ANFIS a sistemas de inversores.

La Tabla 3.4 presenta un resumen comparativo de diversos estudios que han aplicado redes ANFIS en inversores multinivel, destacando sus configuraciones, métodos de entrenamiento y resultados de THD obtenidos. Esta tabla, de carácter resumido, permite identificar no solo las principales estrategias utilizadas, sino también la escasez de investigaciones enfocadas en la implementación de ANFIS en inversores, y aún más en su aplicación dentro de sistemas SAI. Su propósito es ofrecer una referencia comparativa que evidencie las ventajas de ANFIS frente a otros algoritmos y las oportunidades de investigación existentes en este campo.

Tabla 3.4 ANFIS aplicado a inversores.

Ref.	Aplicación	Enfoque	Algoritmos y THD
[94]	Fotovoltaico (PV)	MPPT controlador Minimización de THD	PSO-ANFIS=29.13%, P&O-ANFIS=47.54%, RNA=59.16%
[56]	PV	Controlador de corriente Minimización de THD	RNA=5.29%, FUZZY=1.48%, ANFIS=1.17%
[60]	Controlador de voltaje	Minimización de THD	RNA=14.57%, NR=11.17%, PSO=15.37%, NF=11.77%, ANFIS=9.16%
[95]	Motor	Controlador de velocidad Minimización de THD	NR= 5.55%, PSO=4.79%
[104]	PV	Variador de frecuencia PWM	PI=7.67%, FUZZY=5.03%, ANFIS=3.62%, PID-PSO=2.68%
[97]	Reducción de THD	Controlador de voltaje Minimización de THD	ALO-RNA=12.95%, GA-RNA=14.91%, PSO-ANFIS=12.95%, ALO-ANFIS=14.91%
[72]	SHAF	Controlador de histéresis Minimización de THD	Sin filtro=47.94%, RNA=2.53%, ANFIS=2.24%
[73]	PV	Controlador de voltaje Minimización de THD	Fuzzy=3.19%, ANFIS=3.02%
[76]	PV	Controlador de corriente Minimización de THD	Sin filtro=39.51%, ANFIS=3.32%, WH ADELINE=2.28%, MWH ADELINE=2.20% (RC Load)
[101]	SAPF	Controlador de voltaje Minimización de THD	Sin control=44.55%, BSMC=3.83%, ANFIS-PI=2.96%
[102]	PV	Controlador de voltaje Minimización de THD	PI=18.19%, PID=16.11%, ANFIS-EPO=0.78%
[77]	Inversor PEC9	Variación de voltaje Minimización de THD	ANFIS=2.2%
[78]	PV	Controlador de corriente y voltaje Minimización de THD	PI=1.57%, FUZZY=1.57%, ANFIS=1.58%
[93]	Controlador de voltaje	Minimización de THD	GA+ANFIS=8.1%
[79]	ANFIS implementación	Controlador de voltaje y RMS Minimización de THD	Sin control= 4.97%, FUZZY= 4.15%, ANFIS=3.77%
[88]	Motor	Controlador de velocidad Minimización de THD	ANFIS + Mínimos cuadrados. 2-niveles=5.78%, 5 –Niveles =4.04%
[81]	PV	Controlador de voltaje Minimización de THD	PI=8.34%, FLC=5.65%, RNA=8.18%, ANFIS=4.04%
[82]	Red	Minimización de THD usando LS-PWM	PI=0.01%. FIS=0.01%. ANFIS=0.01%
[83]	Motor	Controlador de torque Minimización de THD	FUZZY=14.32%, ANFIS=11.73%, ConvLSTMDTC=6.61%
[85]	ANFIS rendimiento	Minimización de THD con controlador de modo deslizante	Synchronous Reference Frame (SRF)=3.34%, ANFIS-SRF = 1.56%
[108]	PV Microred	Controlador de voltaje Minimización de THD	Ausencia de controlador=1.25% FLC=4.72% ALO-NN=1.80% PSO-ANFIS=1.25%

En la Tabla 3.4, aunque las técnicas ANFIS y de optimización metaheurística se han aplicado en diversas áreas de la electrónica de potencia, existe poca bibliografía específica sobre su uso en sistemas SAI, ya que la mayoría de las aplicaciones se

centran en sistemas fotovoltaicos. Además, muchos estudios emplean ANFIS para el control de voltaje u otros tipos de controladores, mientras que pocos se centran en la minimización efectiva de la THD. Sin embargo, ANFIS suele obtener mejores resultados que otros métodos (como las RNA o la lógica difusa).

En particular, las referencias [94, 97, 101, 102] utilizaron ANFIS combinado con métodos de entrenamiento distintos a gradiente descendiente, que es el método básico más empleado. Referencias [94, 97] destacan por emplear la combinación de ANFIS con PSO, obteniendo los mejores resultados en reducción de THD. Esto indica una notable eficiencia y sugiere un área prometedora para futuras investigaciones. Además, es importante señalar que los autores emplearon topologías y niveles de inversor diferentes, lo que dificulta la estandarización o la comparación directa de las mejoras de THD. Esto subraya la necesidad de realizar más estudios centrados en estos sistemas, y el presente trabajo trata precisamente de llenar ese vacío.

Un reciente artículo de la revisión del estado del arte [23] destaca que, si bien algunos artículos se centran en ANFIS aplicados a inversores multinivel, pocos investigadores han utilizado métodos metaheurísticos para entrenar redes ANFIS. Unos pocos trabajos combinan ANFIS con métodos metaheurísticos, como PSO, GA y ABC. Aunque estos estudios son relativamente escasos, han demostrado ser prometedores en el proceso de entrenamiento y optimización. Esta combinación representa una novedad en este campo y sugiere un potencial sin explotar en la integración de ANFIS con métodos metaheurísticos [85, 107].

3.4 Perspectivas futuras y conclusiones del Estado del Arte

Perspectivas futuras

El análisis de la información recogida en los artículos estudiados permite vislumbrar el potencial de ANFIS para reducir el contenido armónico en los inversores multinivel con el fin de mejorar la calidad y la eficiencia de la energía. Como perspectivas de las aplicaciones de ANFIS, cabe mencionar las siguientes:

La herramienta ANFIS permite desarrollar nuevas estrategias avanzadas de modulación para reducir el contenido armónico de la forma de onda de salida en los inversores multinivel. Estas estrategias de modulación avanzadas podrían adaptarse a las fluctuaciones de la carga y la dinámica de la red en tiempo real, mejorando la calidad de la energía.

Por otra parte, la optimización de la secuencia de conmutación de los dispositivos semiconductores de potencia en los convertidores multinivel en cascada podría realizarse utilizando ANFIS y proporcionar un medio para reducir las pérdidas de conmutación de los dispositivos semiconductores mejorando el rendimiento y la eficiencia del sistema.

La compensación activa del contenido armónico es otro uso potencial de ANFIS para mitigar eficazmente los armónicos indeseables en la red eléctrica, garantizando que la calidad de la energía cumpla las normas más estrictas establecidas por los organismos reguladores.

Es deseable incorporar el uso de ANFIS en el diseño de sistemas de energías renovables. La energía eléctrica generada con fuentes renovables presenta fluctuaciones o intermitencia debido a la aleatoriedad de las condiciones meteorológicas. Dichos sistemas emplearían técnicas ANFIS para garantizar un mejor aprovechamiento de la energía eléctrica disponible, mejorando el MPPT y el rendimiento del sistema.

En resumen, el uso de ANFIS para reducir la THD en inversores multinivel tiene un gran potencial. Esta tecnología mejora la calidad de la energía, reduce las pérdidas de energía y se adapta con facilidad a los cambios en las aplicaciones de electrónica de potencia. La continuación de la investigación y el desarrollo tecnológico serán fundamentales para aprovechar todo el potencial de esta tecnología en este campo.

Conclusiones del Estado del Arte

Esta revisión estudia los métodos ANFIS y sus aplicaciones en inversores, centrándose en la electrónica de potencia. ANFIS se ha convertido en una técnica de IA muy popular debido a sus ventajas específicas. A pesar del uso generalizado de los métodos ANFIS en diversos campos de la ingeniería, nunca se ha realizado una revisión sistemática para investigar las aplicaciones de ANFIS en inversores multinivel.

Por lo tanto, este documento tiene como objetivo abordar esta brecha mediante el estudio de las aplicaciones ANFIS en las publicaciones de investigación de electrónica de potencia, centrándose en los inversores multinivel desde 2013 hasta 2023. Se ha realizado un análisis exhaustivo de tipo “survey”, revisiones sistemáticas y revisiones de ANFIS. Esta búsqueda tenía como objetivo determinar si existían artículos centrados en la electrónica de potencia o en los inversores. Por lo tanto, se revisaron y clasificaron los trabajos.

Se observó que el entrenamiento de ANFIS mediante algoritmos metaheurísticos aún no está muy extendido entre los investigadores, y que solo se han implementado unos pocos algoritmos (como PSO, GA, ABC, SSA y ALO). Por lo tanto, se pueden explorar nuevos algoritmos metaheurísticos o variaciones de algoritmos existentes para el entrenamiento de ANFIS.

En el campo de la electrónica de potencia, ANFIS presenta posibilidades en varias áreas. Los resultados de esta revisión sugieren la viabilidad de investigar configuraciones de carga alternativas y explorar otras topologías y niveles de inversores.

En cuanto a los métodos de implementación, cabe destacar la diversidad de enfoques. Algunos autores optaron exclusivamente por las simulaciones, mientras que otros realizaron investigaciones experimentales, aportando datos exhaustivos sobre los componentes utilizados. Aunque no hay una preferencia dominante por los dispositivos, destacan elementos notables como los OPAL, el Spartan de Xilinx y el microcontrolador 16f877A.

Los investigadores han adoptado distintos enfoques para reducir la THD. Algunos se concentraron en reducir la THD de tensión, otros en la THD de corriente, otros exploraron el MPPT y otros se centraron en los controladores. En general, es evidente que las redes ANFIS superan a otras formas de inteligencia artificial en términos de eficacia y rendimiento para la reducción de THD.

Asimismo, el documento destaca que varios investigadores omitieron información crucial para el proceso de formación, como el número de neuronas o de muestras; estos datos son necesarios para una formación ANFIS eficaz. Por tanto, se reafirma la necesidad de conocer a fondo el proceso de entrenamiento de ANFIS para garantizar su eficacia.

El estudio identifica el potencial de investigación adicional explorando varias configuraciones de carga, topologías y niveles, guiando a los investigadores hacia la creación de soluciones innovadoras en el ámbito de los inversores multinivel mediante la implementación de ANFIS.

4. Resultados

4.1 Propuesta

En sistemas con inversores multinivel, el cálculo de los ángulos de conmutación se vuelve complejo cuando existen variaciones en las fuentes de voltaje. Los métodos clásicos o incluso los metaheurísticos presentan limitaciones importantes, ya que no son capaces de generar ángulos óptimos en tiempo real ni adaptarse eficientemente a las condiciones cambiantes del sistema. Además, las estrategias tradicionales suelen requerir el almacenamiento de tablas extensas o cálculos computacionalmente costosos, lo que dificulta su implementación práctica.

Una posible solución es utilizar una estrategia híbrida, que combine un algoritmo metaheurístico y una red neuro difusa ANFIS. La propuesta consiste en aprovechar la capacidad de aprendizaje de la ANFIS y la eficacia de búsqueda del algoritmo PSO, para mejorar la precisión en la generación de ángulos de conmutación y, con ello, la calidad de la señal de salida del inversor y mejorar la precisión en la generación de ángulos de conmutación.

En esta propuesta, los ángulos de conmutación se determinan previamente fuera de línea utilizando un algoritmo PSO, que busca encontrar la configuración óptima para minimizar los armónicos. Estos ángulos de conmutación se utilizan para entrenar la red ANFIS, la cual se encarga de calcular y ajustar los ángulos en tiempo real, dependiendo del voltaje de entrada. Cuando se varían las fuentes de entrada en el CHBMLI, la red ANFIS actúa como un controlador inteligente que optimiza dinámicamente los ángulos de conmutación, permitiendo una respuesta precisa y eficiente.

La Figura 4.1 ilustra el funcionamiento general de la propuesta, mostrando la relación entre el proceso de ANFIS cuando hay variación del voltaje de las baterías. La efectividad de esta técnica se evaluará mediante pruebas en un inversor multinivel. Durante estas pruebas, se analizarán el voltaje y los armónicos resultantes. Estos análisis proporcionarán información crucial sobre la capacidad de la técnica para reducir eficientemente los armónicos y mejorar la calidad de la señal de salida del inversor. Con esta base, en la siguiente sección se detalla la metodología empleada para desarrollar y evaluar la propuesta.

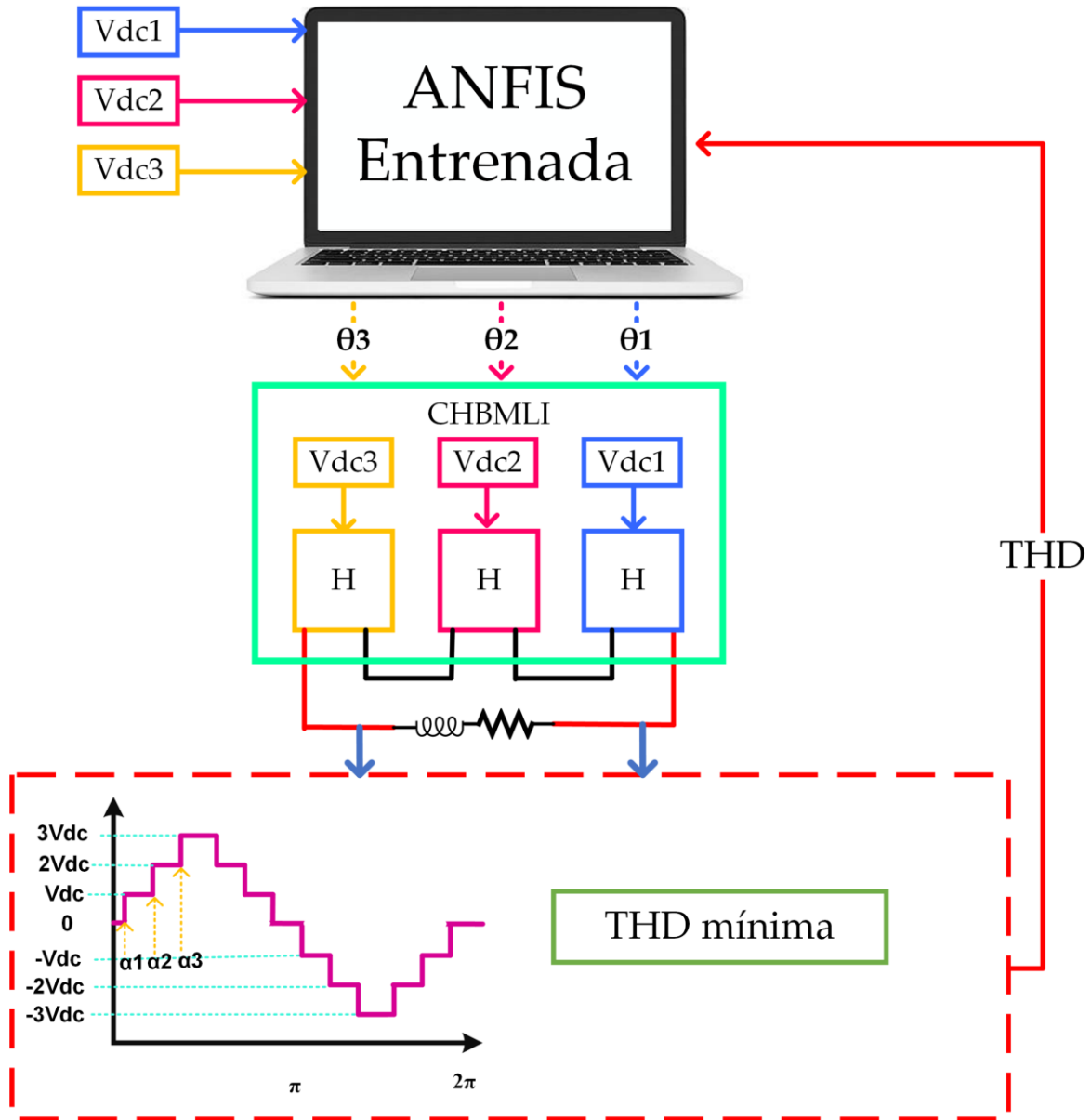


Figura 4.1 Funcionamiento de la propuesta.

4.2 Metodología

Este trabajo de tesis se desarrolló en seis etapas, siguiendo un enfoque metodológico estructurado para la optimización de los ángulos de conmutación en un CHBMLI mediante el uso de redes ANFIS y PSO. A continuación, se describen las etapas del proceso:

1. **Análisis del estado del arte:** Se lleva a cabo una revisión bibliográfica sobre el uso de inteligencia artificial en inversores, incluyendo conceptos básicos

de redes neuronales y metaheurísticas. Esto permite identificar avances y oportunidades en la aplicación de ANFIS a inversores multinivel.

2. **Identificación de áreas de oportunidad y definición del caso de estudio:** Se determinan los aspectos críticos en la regulación de ángulos de conmutación y se define el caso de estudio basado en un CHBMLI de 7 niveles con un banco de baterías LiFePO₄ como fuente de alimentación.
3. **Aprendizaje y programación de algoritmos:** Se estudian los principios de PSO y ANFIS, y se desarrolla el código de integración de ambas técnicas. En esta etapa se abordan los desafíos de implementación y refinamiento del modelo híbrido ANFIS+PSO.
4. **Desarrollo de la función objetivo y entrenamiento del modelo:** Se diseña la función objetivo para minimizar la THD de voltaje, considerando diferentes variaciones de voltajes de entrada. Se entrenan los modelos ANFIS+PSO y se realizan ajustes para mejorar su desempeño.
5. **Simulaciones en Simulink:** Se implementa el modelo del inversor multinivel en Simulink. Se realizan pruebas para evaluar el comportamiento del inversor con diferentes voltajes y se analizan los resultados.
6. **Análisis de resultados y conclusiones:** Se evalúa el rendimiento del modelo en términos de reducción de THD de voltaje, comparando los resultados obtenidos con otros enfoques y estudios previos. Se identifican las ventajas y limitaciones de la metodología propuesta y se plantean futuras líneas de investigación.

A continuación, se presenta el diagrama del proceso metodológico, con las actividades primarias y secundarias que se llevaron a cabo:

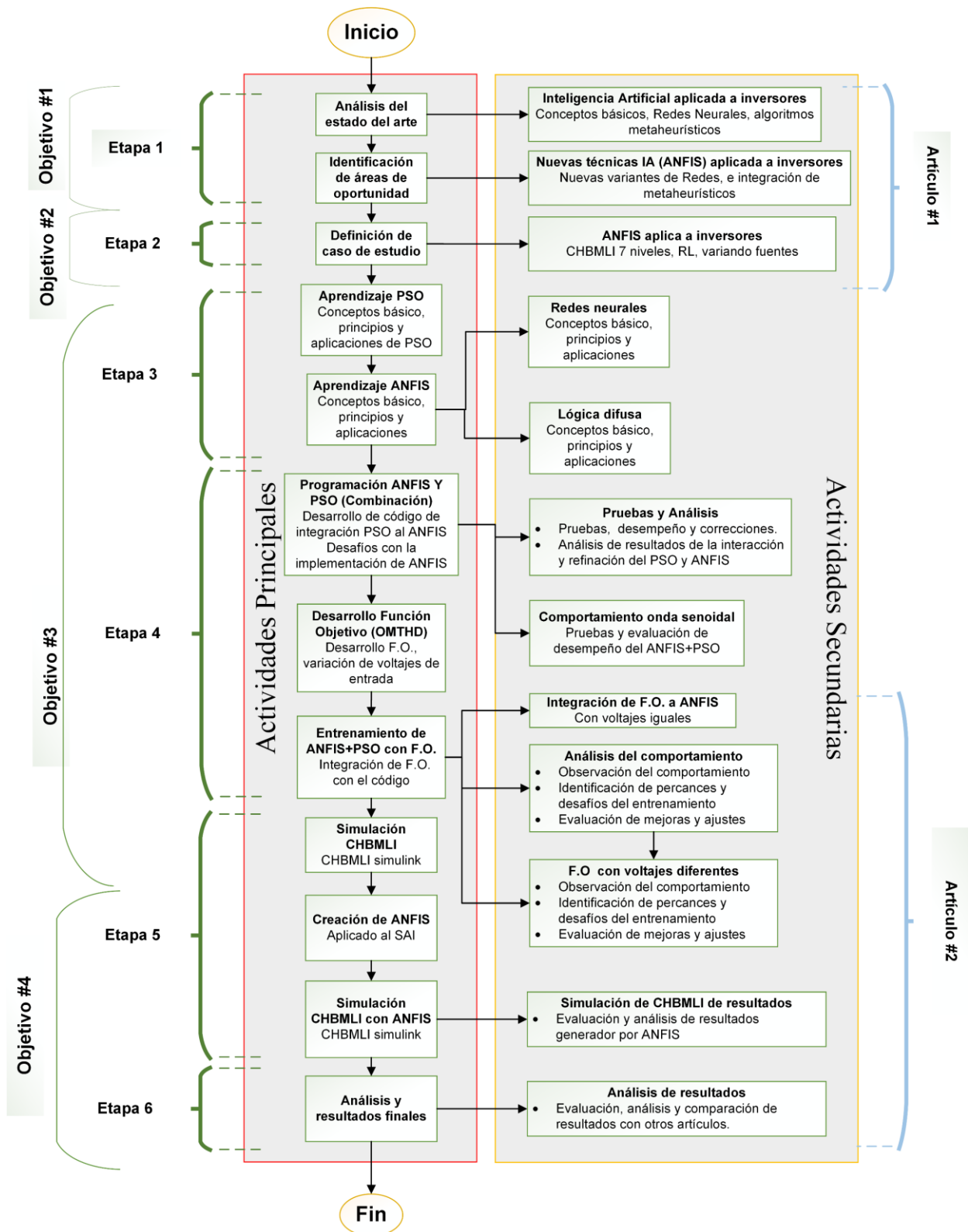


Figura 4.2 Diagrama de la metodología empleada.

La metodología presentada establece un enfoque estructurado para la optimización de los ángulos de disparo en un inversor multinivel en cascada. Mediante la combinación de ANFIS y PSO, se desarrolló un modelo capaz de adaptarse a variaciones en los voltajes de entrada, asegurando una minimización efectiva de la distorsión armónica total. En la siguiente sección, se presenta el caso de estudio utilizado para validar la efectividad del enfoque propuesto.

4.3 Caso de estudio

El caso de estudio se trata de un SAI en línea que utiliza un inversor multinivel en cascada de siete niveles con una carga RL. Este tipo de inversor requiere tres fuentes de alimentación independientes que, en este caso, son tres paquetes de baterías. A medida que las baterías se descarguen, darán distintos niveles de tensión. Se utilizaron baterías LifePo4 S17 (62 V) para obtener una tensión de salida de 131 VRMS o 186 V de pico, emulando la carga de un ordenador en aplicaciones críticas, como bases de datos bancarias u hospitalarias. Se consideraron cinco porcentajes de carga crítica para entrenar la red ANFIS basándose en los voltajes observados durante la descarga de las baterías, como se muestra en la Tabla 4.1.

Tabla 4.1 Porcentaje y voltajes importantes del SAI para el caso de estudio.

Porcentaje de carga %	Voltaje de la celda	Voltaje de la batería	Voltaje de salida del inversor	Voltaje RMS
100	3.65	62.05	186.15	131.62
95	3.38	57.46	172.38	121.9
50	3.276	55.692	167.076	118.14
15	3.15	53.55	160.65	113.6
5	2.851	48.467	145.401	102.81

Estos voltajes son importantes porque el rendimiento de las baterías LifePo4 muestra una tendencia significativa al descargarse. Al 95% de carga, el voltaje en las celdas sufre un cambio aparente; después de este punto, la descarga de la batería es constante y muestra poca variación. Por lo tanto, se decidió observar la carga del 50% para que la red ANFIS pudiera comprender mejor el comportamiento en este rango estable. Sin embargo, a partir del 15% de carga, se observa un cambio drástico en la tensión. Los puntos críticos en los que la batería muestra un comportamiento más estable se encuentran entre el 95% y el 15% de carga. Al 5%, la batería está prácticamente vacía. Aun así, para que la red ANFIS se entrene correctamente y prediga la posible THD en este escenario, se incluyó este porcentaje de la carga en el análisis.

Considerando estos cinco puntos críticos en la descarga de la batería, se pueden obtener múltiples combinaciones y escenarios posibles para entrenar la red ANFIS. Por este motivo, se diseñaron experimentos para identificar los escenarios más relevantes. Es importante tener en cuenta que, debido al tipo de modulación utilizada, los paquetes de baterías no se descargan de forma equilibrada. Este consumo de energía debe tenerse en cuenta a la hora de diseñar los experimentos para capturar todos los escenarios posibles. Por último, tras incorporar todas estas variables, el diseño de los experimentos reveló que se necesitan 35 experimentos

para entrenar la red correctamente. La Tabla 4.2 muestra los 35 experimentos y los distintos voltajes para el entrenamiento posterior.

Tabla 4.2 Porcentajes y voltajes importantes del caso de estudio.

Experimento	Voltaje de las fuentes de entrada			Voltaje de salida	Voltaje RMS
	V ₁	V ₂	V ₃		
1	62.05	62.05	62.05	186.15	131.63
2	57.46	62.05	62.05	181.56	128.38
3	55.692	62.05	62.05	179.8	127.13
4	53.55	62.05	62.05	177.65	125.61
5	48.467	62.05	62.05	172.56	122.02
6	57.46	57.46	62.05	176.97	125.13
7	55.692	57.46	62.05	175.20	123.88
8	53.55	57.46	62.05	173.06	122.37
9	48.467	57.46	62.05	167.98	118.77
10	55.692	55.692	62.05	173.43	122.63
11	53.55	55.692	62.05	171.29	121.12
12	48.467	55.692	62.05	166.20	117.52
13	53.55	53.55	62.05	169.15	119.60
14	48.467	53.55	62.05	164.06	116.01
15	48.467	48.467	62.05	158.98	112.42
16	57.46	57.46	57.46	172.38	121.89
17	55.692	57.46	57.46	170.61	120.64
18	53.55	57.46	57.46	168.47	119.12
19	48.467	57.46	57.46	163.38	115.53
20	55.692	55.692	57.46	168.84	119.39
21	53.55	55.692	57.46	166.70	117.87
22	48.467	55.692	57.46	161.62	114.28
23	53.55	53.55	57.46	164.56	116.36
24	48.467	53.55	57.46	159.47	112.76
25	48.467	48.467	57.46	154.39	109.17
26	55.692	55.692	55.692	167.07	118.14
27	53.55	55.692	55.692	164.93	116.62
28	48.467	55.692	55.692	159.85	113.03
29	53.55	53.55	55.692	162.79	115.11
30	48.467	53.55	55.692	157.71	111.51
31	48.467	48.467	55.692	152.62	107.92
32	53.55	53.55	53.55	160.65	113.59
33	48.467	53.55	53.55	155.56	110.00
34	48.467	48.467	53.55	150.48	106.41
35	48.467	48.467	48.467	145.40	102.81

A partir de estos datos, se desarrolló una red ANFIS en combinación con el algoritmo PSO en MATLAB. El proceso incluyó la programación inicial de la red ANFIS, seguida de la integración del PSO y el diseño de una función objetivo destinada a reducir la THD.

4.4 Entrenamiento de la Red

El código MATLAB resultante se diseñó para entrenar la red ANFIS con PSO utilizando los ángulos de conmutación y las respectivas medidas de THD. Esta configuración permitió observar detalladamente el comportamiento durante el proceso de entrenamiento, facilitando la evaluación de su rendimiento y precisión. Durante el desarrollo se realizaron continuas iteraciones y ajustes para optimizar la convergencia de la red y mejorar su capacidad predictiva. El proceso consistió en ajustar los parámetros de la función objetivo para garantizar un rendimiento óptimo durante el entrenamiento.

La red ANFIS implementada se estructuró con tres entradas (θ_1 , θ_2 y θ_3), correspondientes, y una salida, que representa a la THD. Se utilizaron tres funciones de pertenencia gaussianas por cada entrada, generando un total de 27 reglas difusas mediante el método de partición por cuadrícula (grid partitioning).

A continuación, se resumen los principales parámetros del sistema ANFIS entrenado:

Tabla 4.3 Parámetros del sistema FIS.

Parámetro	Valor	Descripción
Nombre del FIS	sugeno31	Nombre del archivo o estructura del modelo FIS
Tipo de FIS	Takagi-Sugeno	Tipo de sistema difuso utilizado (orden cero o uno)
Número de entradas	3	Corresponden a los tres ángulos de conmutación en grados
Número de salidas	1	Resultado del modelo: la estimación del THD (%)
Funciones de membresía por entrada	4 (tipo gaussmf)	Cada entrada usa 4 funciones de membresía de tipo gaussiana
Rango de entrada 1 (in1)	[8.5450, 8.7890]	Ángulo de conmutación 1
Rango de entrada 2 (in2)	[27.8430, 28.0000]	Ángulo de conmutación 2
Rango de entrada 3 (in3)	[49.8090, 50.0000]	Ángulo de conmutación 3
Funciones de membresía de salida	4 (tipo linear)	La salida utiliza funciones lineales (Sugeno de orden 1)
Rango de salida (out1)	[10.4326, 10.4372]	Rango de valores de THD estimados por el modelo (%)
Método AND	prod	Producto para evaluar la conjunción de reglas
Método OR	probor	Probabilistic OR para la disyunción difusa
Método de implicación	prod	Producto como método de implicación
Método de agregación	sum	Suma de los resultados de reglas activadas
Método de desdifusificación	wtaver	Media ponderada usada para convertir la salida difusa a un valor numérico
Número total de reglas difusas	4	Número de reglas generadas durante el entrenamiento

La Tabla 4.3 muestra los principales parámetros proporcionados por el sistema difuso generado en MATLAB (FIS). En ella se describen los valores clave de configuración y se explica el propósito de cada uno dentro de la red ANFIS entrenada.

Es importante señalar que los rangos de entrada y salida presentados corresponden al primer experimento. Estos valores no son relativos ni generalizados, sino que reflejan una configuración específica, optimizada mediante el algoritmo PSO para minimizar la THD.

Además, se trata de un sistema tipo Takagi-Sugeno de primer orden, caracterizado por funciones de pertenencia gaussianas en las entradas y funciones lineales en la salida.

Una vez definido y optimizado el sistema FIS, se procedió al entrenamiento de la red ANFIS, el cual consta de dos etapas: la etapa de entrenamiento y la etapa de prueba. En la etapa de entrenamiento, la red se entrenó con 35 casos prácticos para aprender y determinar los ángulos de conmutación que minimizan la THD en el inversor.

La Figura 4.3 muestra el proceso de convergencia y la distribución de ángulos durante el entrenamiento, concretamente para el experimento 1.

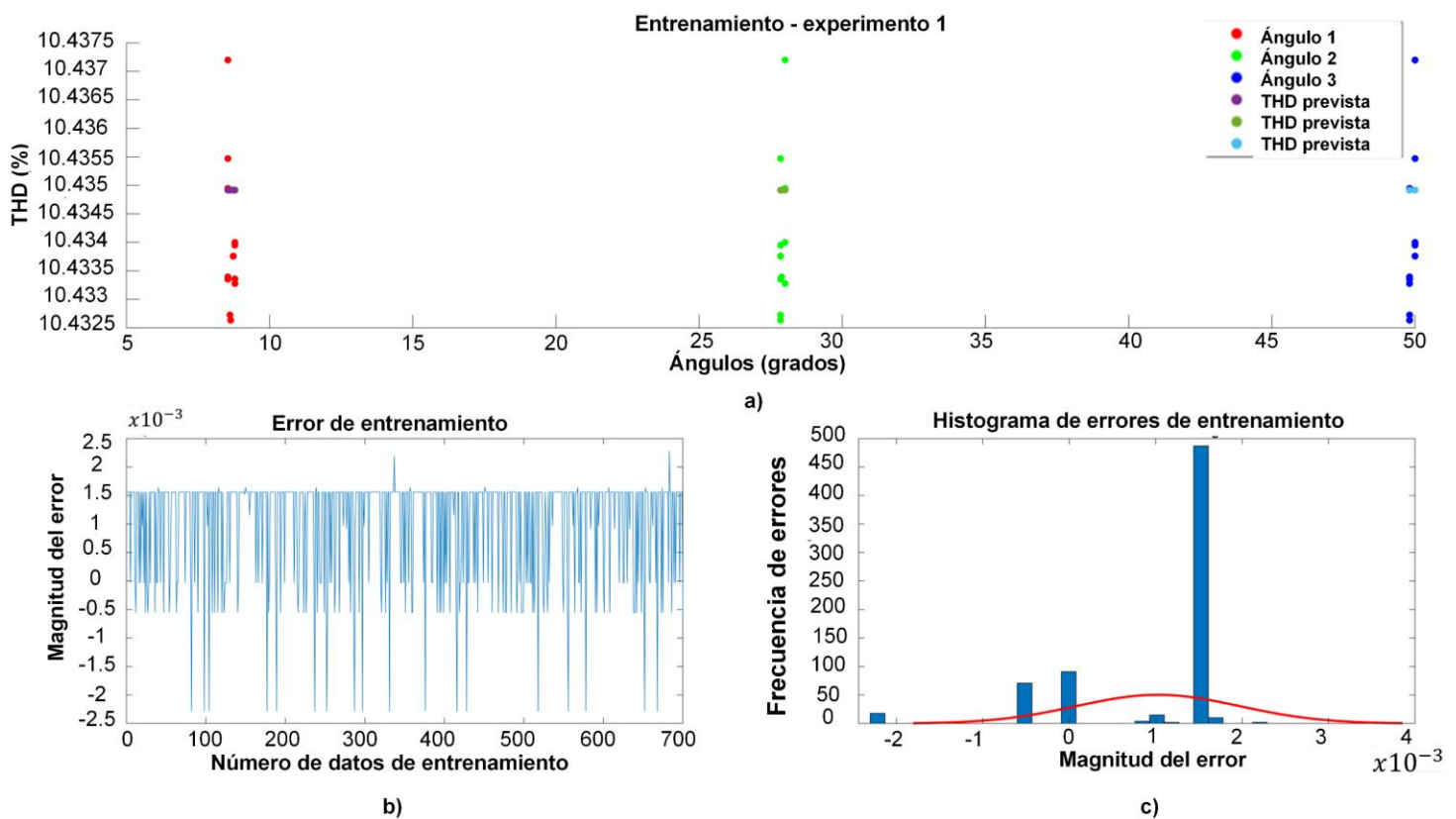


Figura 4.3 Experimento 1 etapa de entrenamiento. a) THD vs. ángulos b) Error vs. datos de entrenamiento c) Cantidad de datos vs. error.

La Figura 4.3 muestra cómo convergen los ángulos de conmutación a valores específicos y cómo se correlacionan con la THD. Para este caso práctico, se utilizaron 1000 muestras (conjuntos de ángulos) en estas pruebas, distribuyendo el 70% (700 muestras) para el entrenamiento de la red y el 30% (300 muestras) para las pruebas. Este número de muestras es suficiente para que la red converja eficazmente, ya que proporciona datos adecuados para que el modelo analice y aprenda las relaciones entre ángulos y THD.

Con un conjunto de entrenamiento de 700 muestras, la red puede observar patrones y lograr la convergencia, lo que se traduce en una mejora del rendimiento. Esta generación de ángulos se realizó de forma aleatoria, cumpliendo las restricciones establecidas. Sin embargo, también ayudó al sistema a orientar mejor a la PSO para encontrar mejores ángulos y obtener errores próximos a cero.

La Figura 4.3 muestra varios gráficos. En el gráfico a) se muestra cómo los ángulos de conmutación convergen a valores específicos: el primer ángulo converge a aproximadamente 8° , el segundo a entre 27° y 28° , y el tercero a entre 49° y 50° . Estos ángulos convergen a una THD mínima del 10.43%, acercándose así a los resultados descritos en la bibliografía para un MLI de siete niveles. En el gráfico b), que muestra los errores de entrenamiento, situados en la esquina inferior izquierda, con la ayuda y guía del PSO, las variaciones disminuyen, obteniéndose desviaciones de ± 0.002 .

Esto muestra una mejora significativa en la convergencia y una aproximación más cercana al mínimo global. En el gráfico c), que es de distribución gaussiana (esquina inferior derecha), se puede ver cómo los errores tienden a acercarse a 0. Sin embargo, muchos de los errores caen dentro de $+0.0015$, y son errores mínimos que no afectan al sistema y consiguen converger al resultado. Aunque se observa que hay errores que están fuera de la campana de Gauss, se considera que estos errores no tienen un impacto crítico en el rendimiento global.

En la segunda etapa, denominada etapa de prueba, se validaron los resultados obtenidos en el entrenamiento. Esto permitió evaluar la capacidad de la red para ajustar los ángulos de conmutación, optimizando la calidad de la señal de salida incluso cuando se producen variaciones en la tensión de entrada. La siguiente figura muestra los resultados de la etapa de prueba de datos.

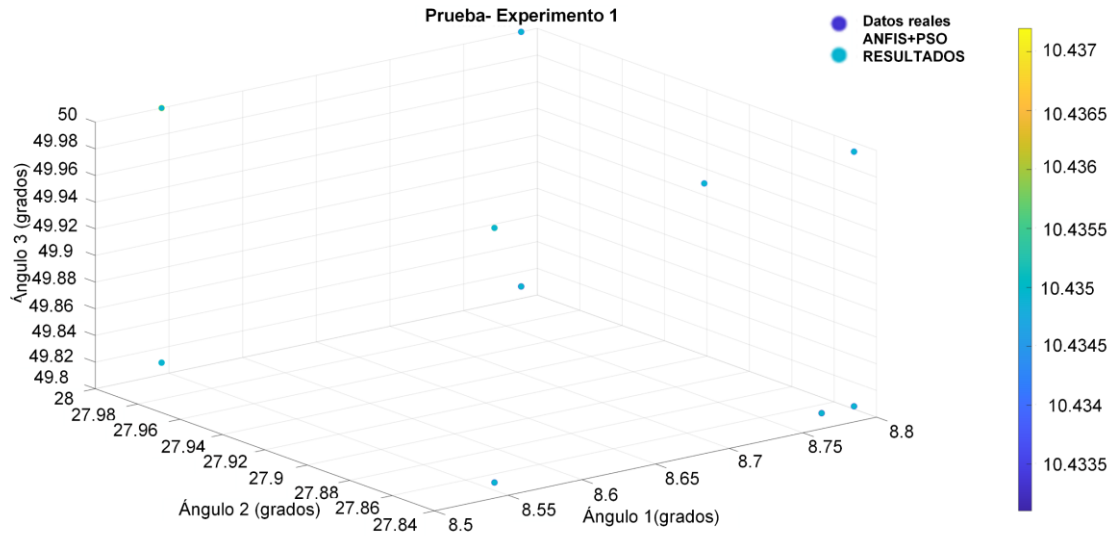


Figura 4.4 Visualización en 3D de los ángulos de conmutación obtenidos.

La Figura 4.4 muestra los tres ángulos de conmutación obtenidos en la etapa de prueba. En el gráfico tridimensional se muestran los datos reales junto con los resultados del ANFIS con PSO. Como puede observarse, los ángulos no muestran variaciones significativas, lo que indica que convergen con precisión a los valores esperados, con errores cercanos a ± 0.002 . En la parte derecha de la figura, se muestra un espectro de colores que representa la THD obtenida por los conjuntos de ángulos. Las tonalidades, concentradas en una gama de azules, confirman que las desviaciones son mínimas y que los ángulos calculados permiten aproximarse a la THD mínima esperada, como se ha mencionado anteriormente. Estas pequeñas variaciones de color reflejan desviaciones de entre 0.0015 y 0.002, lo que dificulta la diferenciación entre tonos en el espectro de colores. Para observar el comportamiento de las desviaciones y errores en la etapa de prueba, se muestran los gráficos de la Figura 4.5.

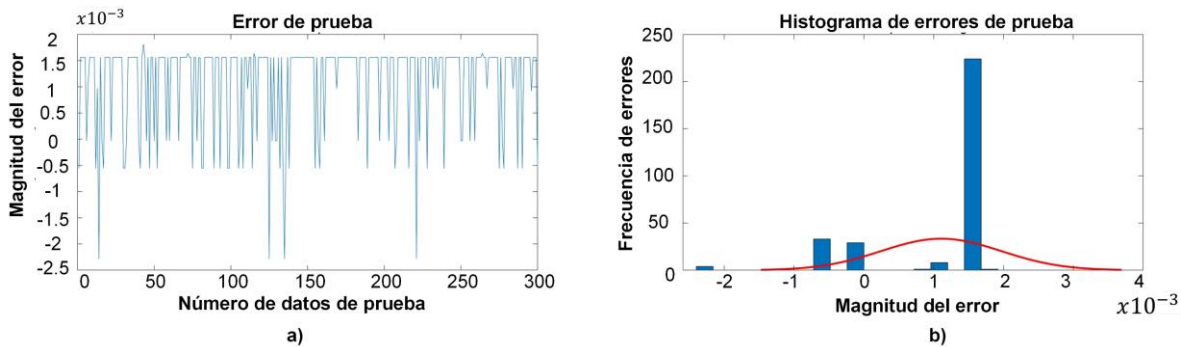


Figura 4.5 Etapa de prueba de los gráficos de error.

En la Figura 4.5 se utilizaron 300 datos de prueba. En el gráfico a), persisten desviaciones de ± 0.002 , lo que muestra un buen rendimiento, como ya se ha mencionado. Pasando al gráfico gaussiano b), la distribución de errores en muchos de ellos cae dentro de 0.0015, lo que muestra un comportamiento mejor que en la etapa de entrenamiento.

Tabla 4.4 Ángulos y THD obtenidos en la red ANFIS.

Experimento	THD% Voltaje	Ángulos		
		Áng. 1	Áng. 2	Áng. 3
1	10.43	8.69°	27.89°	49.81°
2	10.49	7.870°	27.30°	49.56°
3	10.52	7.67°	27.09°	49.53°
4	10.58	7.47°	26.81°	49.53°
5	10.76	7.06°	25.69°	49.69°
6	10.52	8.14°	27.37°	49.39°
7	10.55	7.88°	27.15°	49.33°
8	10.60	7.63°	26.87°	49.28°
9	10.78	7.19°	25.79°	49.31°
10	10.58	7.97°	27.17°	49.26
11	10.63	7.71°	26.89°	49.19
12	10.80	7.25°	25.83°	49.16°
13	10.66	7.8°	26.91°	49.08°
14	10.84	7.31°	25.88°	48.97°
15	10.97	7.49°	25.91°	48.44°
16	10.43	8.69°	27.89°	49.81°
17	10.45	8.31°	27.62°	49.67°
18	10.48	7.92°	27.34°	49.57°
19	10.62	7.37°	26.62°	49.55°
20	10.46	8.45°	27.67°	49.62°
21	10.49	8.03°	27.37°	49.5°
22	10.62	7.43°	26.65°	49.43°
23	10.51	8.18°	27.41°	49.42°
24	10.64	7.51°	26.69°	49.29°
25	10.73	7.72°	26.74°	48.98°
26	10.43	8.69°	27.89°	49.81°
27	10.46	8.22°	27.56°	49.64°
28	10.57	7.51°	26.87°	49.53°
29	10.47	8.39°	27.61°	49.58°
30	10.58	7.59°	26.91°	49.4°
31	10.65	7.84°	26.97°	49.12°
32	10.43	8.69°	27.89°	49.81°
33	10.51	7.72°	27.15°	49.54°
34	10.56	8.01°	27.22°	49.29°
35	10.43	8.69°	27.89°	49.81°

La Tabla 4.4 muestra los ángulos de conmutación por la red ANFIS junto con sus respectivas THD mínimas encontradas de THD asociados para cada uno de los 35 experimentos evaluados. Al observar los datos, se aprecia que el primer ángulo se encuentra generalmente entre 7° y 8°, el segundo entre 25° y 27.8°, y el tercero en

su mayoría alrededor de 49° , con ligeras variaciones. Esto refleja que, aunque las diferencias entre experimentos pueden parecer mínimas, la elección exacta del conjunto de ángulos es esencial. Incluso una pequeña desviación puede provocar un aumento significativo en el valor de THD.

La THD mínima encontrada en los experimentos ronda el 10.40%, mientras que el experimento 15 tiene una THD ligeramente superior a los demás (10.9%). Esto se puede apreciar mejor en la Figura 4.6, que muestra visualmente los resultados de THD obtenidos por la red ANFIS en los distintos experimentos, permitiendo evaluar tanto la calidad del entrenamiento como el comportamiento de convergencia.

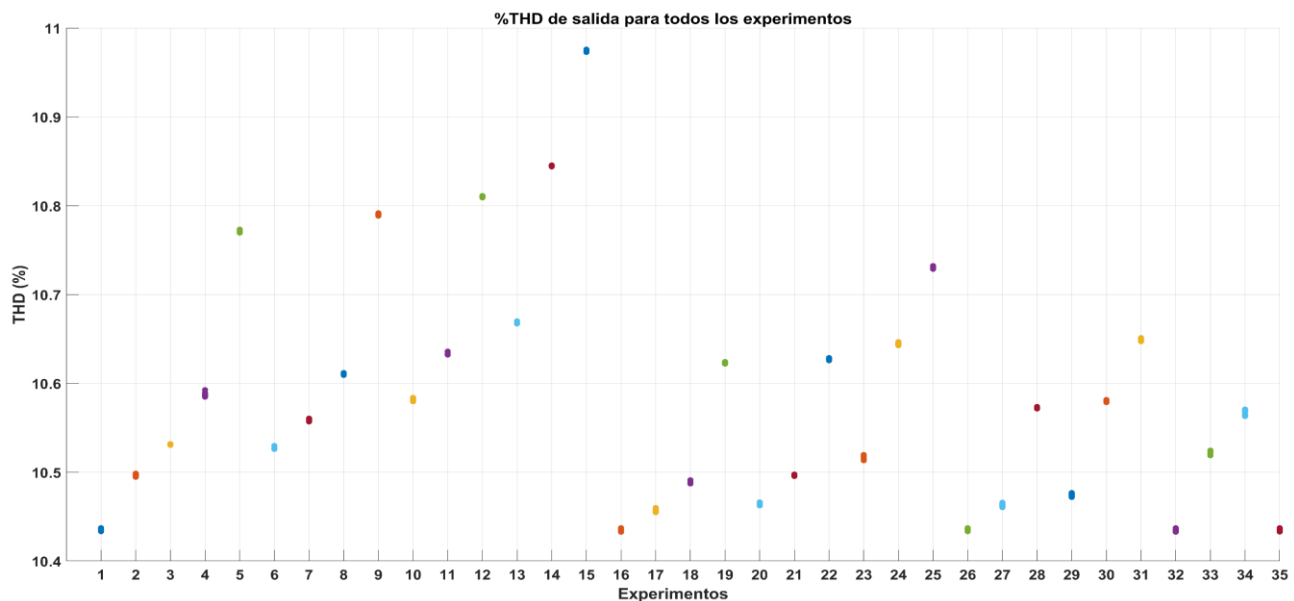


Figura 4.6 THD obtenida a la salida de la red.

La Figura 4.6 muestra los datos en diferentes colores y la THD obtenida del entrenamiento. Si bien los valores son coherentes con los registrados en la Tabla 4.4, algunos puntos presentan una mayor dispersión. Esto indica que en ciertos casos la red necesitó más iteraciones para alcanzar el mínimo deseado de THD. Esto no es necesariamente negativo; sólo refleja que, dependiendo del entrenamiento y los ajustes realizados, la iteración y los ajustes pueden mejorarse aún más. En general, los errores observados en la red ANFIS son de $\pm 0,002$, lo que confirma que los ángulos y los valores de THD obtenidos son coherentes, con variaciones mínimas entre simulaciones. Esto demuestra que la red ANFIS se ha entrenado y ajustado eficazmente.

4.5 Simulación MATLAB del caso de estudio.

Una vez finalizado el entrenamiento de la red ANFIS, el siguiente paso fue desarrollar la simulación del CHBMLI de 7 niveles en MATLAB Simulink. El objetivo

de esta simulación era comparar los resultados de THD obtenidos durante el entrenamiento de la red con los valores simulados en un inversor, utilizando los ángulos calculados por la red ANFIS para cada caso de estudio. La Figura 4.7 muestra el modelo CHBMLI de 7 niveles implementado en Simulink

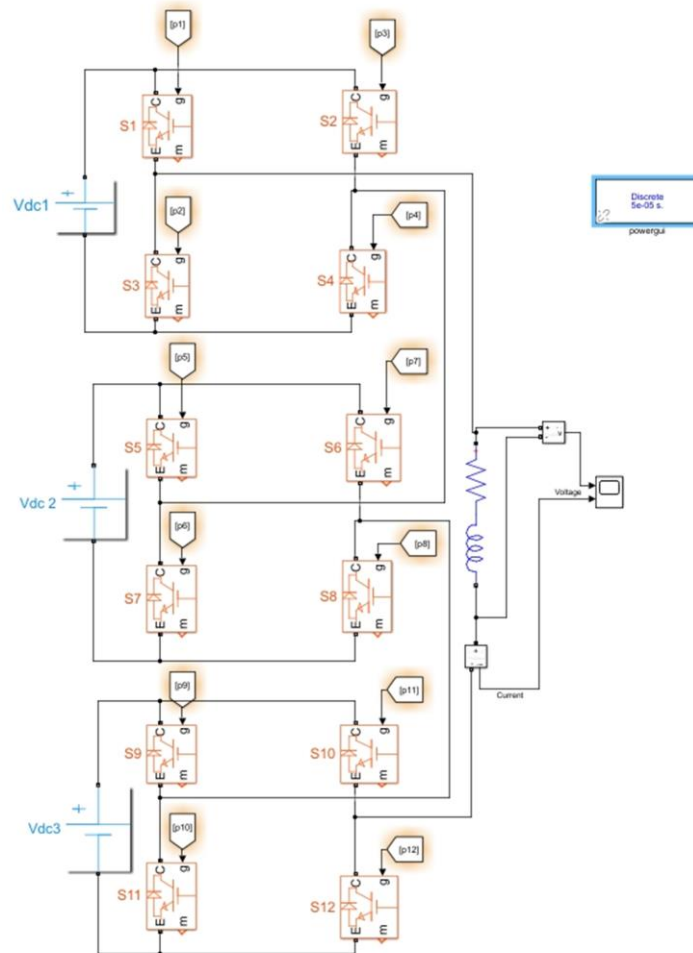


Figura 4.7 CHBMLI de siete niveles en MATLAB Simulink.

Las fuentes de voltaje y los ángulos de conmutación se variaron para la simulación, utilizando el experimento 1 para configurar y observar la salida del inversor, como se detalla en la figura siguiente.

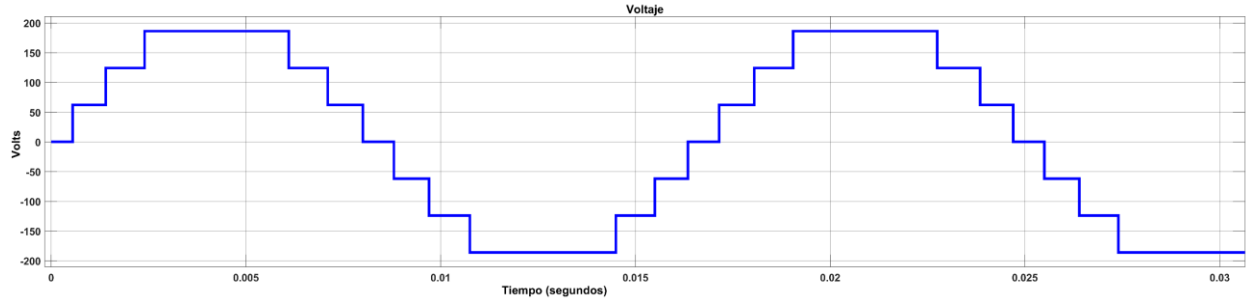


Figura 4.8 Voltaje y corriente del CHBMLI en MATLAB Simulink.

La Figura 4.8 muestra una salida de tensión de 186 V y una corriente de 1.86 A, con una carga conectada de 100Ω y $100 \mu\text{H}$. Esto confirma que el inversor produce los niveles de tensión y corriente esperados, lo que valida el correcto funcionamiento del CHBMLI en las condiciones simuladas. Posteriormente, se utilizó la herramienta de análisis FFT Powergui de Simulink para analizar el contenido armónico y la THD de la señal generada, como se muestra en la Figura 4.9.

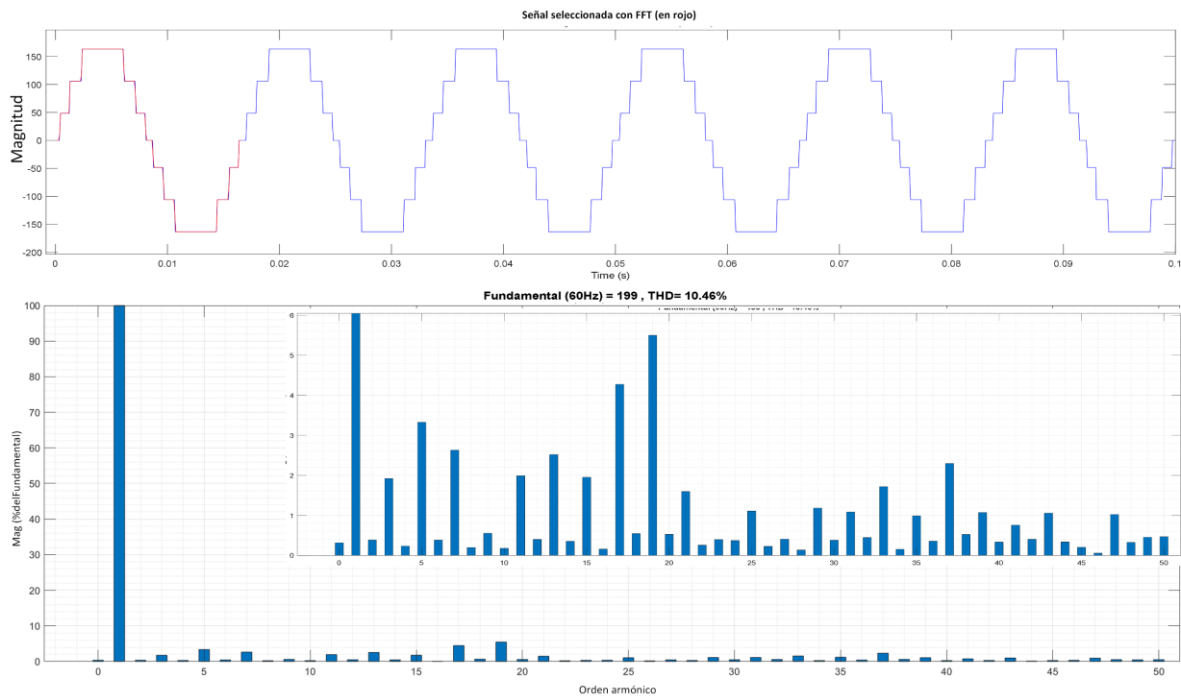


Figura 4.9 Contenido armónico y THD del voltaje del experimento 1.

La Figura 4.9 muestra el contenido armónico, la THD del voltaje resultante y su ampliación, ya que visualmente no se aprecia con claridad el porcentaje de cada componente. La THD total obtenida fue del 10.46%, lo que representa una diferencia de sólo el 0.03% en comparación con la THD calculada por la red ANFIS, lo que demuestra la precisión de la red en la predicción de los ángulos de conmutación. Cabe señalar que las diferencias mínimas observadas pueden deberse a la aproximación utilizada en la simulación, en la que sólo se consideraron dos decimales para los ángulos de conmutación. En un entorno realista, esta

simplificación es habitual, ya que componentes como los IGBT o los MOSFET no pueden conmutar con una precisión superior a dos decimales.

En cuanto al espectro mostrado, se observa la presencia de algunas componentes armónicas pares, lo cual podría parecer inusual, ya que teóricamente solo deberían existir armónicos impares en sistemas simétricos de conmutación. Sin embargo, esto se debe a que el análisis realizado mediante el Fast Fourier Transform (FFT) del toolbox de MATLAB no corresponde a un análisis de Fourier continuo, sino a una aproximación discreta e iterativa.

Este procedimiento numérico puede generar pequeños componentes pares, producto de los redondeos y la discretización de la señal. Es importante destacar que estos armónicos pares presentan magnitudes menores al 1%, por lo que son prácticamente despreciables y no afectan el desempeño global del inversor. En consecuencia, la presencia de estos componentes no implica un error en la modulación ni en el cálculo armónico, y los resultados obtenidos pueden considerarse altamente fiables y representativos del comportamiento real del sistema.

4.6 Discusión

Por último, se simuló los demás experimentos, obteniendo las THD correspondientes y presentando los resultados en la Tabla 4.5.

Tabla 4.5 %THD entrenamiento vs. %THD simulación.

Experiment o	THD% entrenamiento	THD% simulación	Experimento	THD% entrenamiento	THD% simulación
1	10.43	10.46	19	10.62	10.52
2	10.49	10.47	20	10.46	10.51
3	10.52	10.52	21	10.49	10.44
4	10.58	10.5	22	10.62	10.61
5	10.76	10.76	23	10.51	10.49
6	10.52	10.6	24	10.64	10.55
7	10.55	10.46	25	10.73	10.77
8	10.60	10.41	26	10.43	10.46
9	10.78	10.73	27	10.46	10.48
10	10.58	10.53	28	10.57	10.5
11	10.63	10.54	29	10.47	10.41
12	10.81	10.78	30	10.58	10.5
13	10.66	10.64	31	10.65	10.59
14	10.84	10.71	32	10.43	10.46
15	10.97	10.87	33	10.51	10.54
16	10.43	10.46	34	10.56	10.49
17	10.45	10.59	35	10.43	10.46
18	10.48	10.39			

La Tabla 4.5 muestra que, en la mayoría de los casos, la THD obtenida en el entrenamiento y en la simulación varía ligeramente, en general dentro de un margen del 0.05%. En algunos experimentos, la THD coincide con la obtenida durante el

entrenamiento; en otros, supera este margen de variación. Sin embargo, la mayor desviación observada no supera el 0.2%, lo que sigue siendo aceptable, pero debería considerarse para un análisis más detallado. Hay que señalar que esta desviación se produce sólo en unos pocos experimentos.

Un aspecto interesante es que, en algunos experimentos, la THD obtenida en la simulación es inferior a la calculada durante el entrenamiento. Esto sugiere que el sistema minimiza eficazmente la THD en determinadas condiciones de funcionamiento. Sin embargo, también se identifican casos en los que la THD simulada es ligeramente superior a la calculada, aunque, como ya se ha mencionado, estas diferencias se encuentran dentro de un rango aceptable de $\pm 0.2\%$.

La Tabla 4.5 se ha representado gráficamente para visualizar mejor estas variaciones, lo que da como resultado la Figura 4.10.

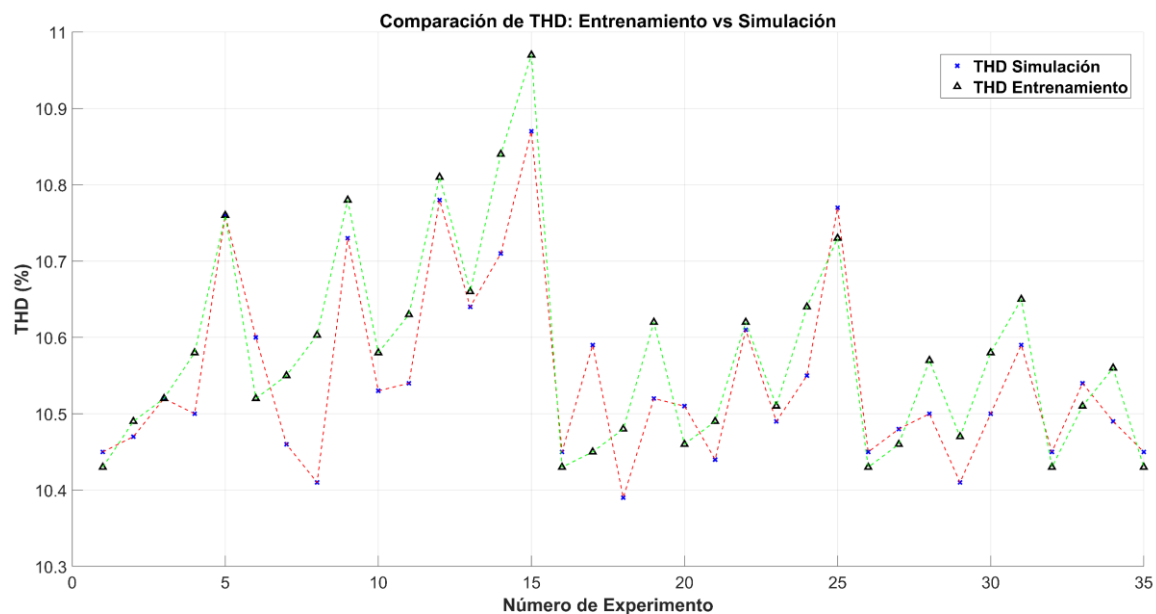


Figura 4.10 Comparación de %THD: Entrenamiento vs. Simulación.

La Figura 4.10 muestra el eje x que representa el número de experimentos y el eje y que muestra el porcentaje de THD obtenido. Como se ha indicado anteriormente, en la mayoría de los casos, las diferencias entre la THD calculada y la THD obtenida en la simulación son mínimas. Sólo en unos pocos casos se observa una variación mayor, siendo el Experimento 8 el que presenta la mayor diferencia, cercana al 0.2%.

Además, la desviación estándar entre el %THD en el entrenamiento y el %THD en la simulación es de 0.06. Este bajo valor de desviación estándar confirma que el proceso de optimización ha minimizado eficazmente la THD en la mayoría de los experimentos. Además, el gráfico nos permite identificar qué experimentos muestran una THD simulada inferior a la calculada, lo que indica una mejora en el rendimiento del sistema, y cuáles muestran una THD ligeramente superior. Aun así, es importante destacar que estas diferencias son pequeñas y no superan el 0.2%. Esto indica que el proceso de optimización ha encontrado efectivamente la THD mínima en la mayoría de los experimentos seleccionados.

Sin embargo, en todos los casos, los valores de THD simulados superan ligeramente el 8% de THD definido por la norma IEEE 519. La reducción efectiva de la THD en la etapa de conmutación permite diseñar filtros más pequeños y menos complejos, optimizando así el sistema.

Este enfoque propuesto demuestra que ANFIS con PSO permite alcanzar bajos niveles de THD en condiciones de simulación y facilita una implementación más eficiente en entornos industriales con fluctuaciones de voltaje. Los ángulos de conmutación optimizados generados por ANFIS con PSO reducen con éxito el contenido armónico. Este enfoque simplifica los requisitos de diseño al minimizar el tamaño y la complejidad de la etapa de filtrado, lo que resulta ventajoso para los sistemas SAI que se enfrentan a entradas de tensión variable. Además, las capacidades de adaptación de ANFIS, mejoradas por la potencia de optimización de PSO, permiten ajustar los ángulos de conmutación a medida que varía el voltaje, garantizando que el inversor mantenga constantemente una salida de alta calidad sin necesidad de tablas de consulta.

Para demostrar la eficacia de la red propuesta, la Tabla 4.6 presenta un resumen comparativo de los valores de THD de tensión alcanzados en diferentes estudios que utilizan inversores multinivel descritos en la bibliografía. Cabe señalar que la búsqueda de trabajos que utilicen inversores multinivel de 7 niveles en cascada con ANFIS combinados con algunas técnicas metaheurísticas es compleja. Por este motivo, la tabla incluye varios estudios relevantes para ilustrar las variaciones en los resultados y las aplicaciones descritas en la bibliografía. Además, cabe destacar que no existen estudios de casos totalmente comparables debido a la diversidad de aplicaciones de ANFIS en contextos específicos.

Tabla 4.6 Comparación de % THD de voltaje en la literatura.

Método	Ref.	Niveles	Características	%THD obtenida
Metaheurístico	[94]	5	No se menciona ángulos, 15 armónicos, sin filtro	PSO-ANFIS=29.13%
	[108]	5	No se menciona ángulos, 14 armónicos, con filtro	PSO-ANFIS=1.25%
	Técnica propuesta	7	Tres ángulos, 50 armónicos, sin filtro	ANFIS-PSO rango=10.39% a 10.87%
	[93]	7	Siete ángulos, 50 armónicos, sin filtro	GA-ANFIS=8.1%
	[97]	9	Cuatro ángulos, 15 armónicos, sin filtro	PSO-ANFIS=12.95% ALO-ANFIS=14.91%
	[102]	27	13 ángulos, 15 armónicos, sin filtro	ANFIS-EPO=0.78%
Clásico	[88]	2 5	No se menciona ángulos ni armónicos, con filtro	ANFIS-Least squares. 2-levels=5.78% 5 -levels=4.04%
	[60]	7	Tres ángulos, 20 armónicos, sin filtro	ANFIS=9.16%
	[78]	9	No se menciona ángulos, 20 armónicos, con filtro	ANFIS=1.58%
	[104]	15	No se menciona ángulos, 150 armónicos, con filtro	ANFIS=3.62%
	[81]	15	No se menciona ángulos ni filtro, 20 armónicos	ANFIS=4.04%

[79]	31	No se menciona ángulos ni filtro, 20 armónicos	ANFIS=3.77%
------	----	--	-------------

La Tabla 4.6 compara las THD de tensión obtenidas en la literatura con las obtenidas en este estudio. Los valores más altos de THD se encuentran en [94], con un inversor de 5 niveles, y en [97], con 9 niveles, lo que destaca la influencia del número de niveles en la reducción de THD.

En las referencias [60, 93], utilizan inversores de 7 niveles, similares al método propuesto. Aunque sus resultados muestran una THD ligeramente menor (entre 1 y 2% de diferencia), esto se debe a ciertos factores específicos. Por ejemplo, [60] sólo analiza hasta el armónico 20, mientras que este estudio considera hasta el armónico 49, cumpliendo con la norma IEEE 519. Un análisis de armónicos más bajo tiende a producir un valor THD más bajo.

En la referencia [93], el estudio emplea siete ángulos de conmutación, lo que disminuye un 2% de la THD, pero al aumentar el número de ángulos de conmutación, también incrementan la cantidad de ecuaciones a resolver, y tiempo en la búsqueda de estos ángulos, por lo que no es benéfico tener más ángulos si la reducción del 2% en la THD se puede eliminar con un filtro.

Las referencias restantes, con valores de THD entre el 0.78% y el 4%, las diferencias se explican por el uso de filtros o el empleo de inversores de 9 hasta 31 niveles. Aunque estos diseños mejoran la salida, también elevan la complejidad del control.

Cabe mencionar que los estudios con menor THD utilizan principalmente ANFIS con métodos clásicos (Gradiente Descendiente).

Otro aspecto importante a examinar es cómo fueron configuradas las redes ANFIS en los trabajos revisados. La Tabla 4.7 presenta los parámetros clave como el número de reglas difusas, la cantidad de muestras de entrenamiento y el número de épocas utilizadas. Estos elementos permiten dimensionar el esfuerzo computacional y el nivel de complejidad asumido en cada propuesta.

Tabla 4.7 Comparación de parámetros relevantes de ANFIS en la literatura.

Ref.	Reglas difusas	Épocas	Muestras o datos	Otros datos relevantes
[16]	50	SM	241 entrenam. / 241 control	250 parámetros totales (50 lineales, 200 no lineales)
[50]	SM	SM	SM	Error de entrenamiento = 0.00066062
[53]	7	SM	SM	SM
[55]	25	SM	SM	SM
[59]	49	40	10,000 entrenam. / 10,000 prueba	SM
[70]	25	SM	SM	SM
[71]	20	SM	SM	SM
[74]	SM	SM	SM	Una base de reglas de 5 x 5 para la entrada y la salida
[79]	49	SM	1000 muestras	SM
[80]	SM	SM	340 datos (272 entrenam., 68 prueba)	SM

[81]	SM	3000	SM	2 entradas, 10 ocultas, 1 salida, tasa de aprendizaje = 0.05
[83]	SM	SM	70% entrenamiento / 30% prueba	171 ms entrenamiento promedio, 7.8 ms predicción
[84]	SM	300	2365 muestras	SM
[87]	SM	30	SM	5 conjuntos ANFIS, error de trayectoria = 0%
[88]	7	1000	SM	Función de activación=tansing, Tolerancia de error 0.01
[89]	49	30	5288 muestras	131 nodos, 91 parámetros totales (49 lineales, 42 no lineales), Función de membresía: Gauss.
[92]	15	100	SM	3 redes ANFIS
[93]	SM	SM	675 (540 entrenamiento, 135 prueba)	SM
[94]	SM	2500	4000 muestras	Tolerancia de error = 0.001
[95]	SM	20	75% entrenam. / 15% prueba / 15% validación	SM
[96]	SM	30	SM	SM
[98]	SM	200	SM	3 neuronas entrada, 10 ocultas, 7 salida, tasa de aprendizaje = 0.95
[101]	9	SM	SM	SM
[103]	25	SM	SM	SM
Técnica propuesta	27	400	35 casos (1000 muestras por caso distribuidas 70% entrenam., 30% prueba)	3 entradas, 1 salidas, red entrenada con PSO, error menor a 0.015. Takagi-Sugeno (1er orden). Funciones de membresía por entrada 4 (tipo gaussmf)

Como se observa en la Tabla 4.7, algunos trabajos reportan información completa sobre los aspectos clave del modelo ANFIS, como el número de reglas difusas, las épocas de entrenamiento o la cantidad y distribución de datos utilizados. Sin embargo, muchas referencias solo mencionan uno o dos de estos aspectos, lo que dificulta la comparación, recopilación y mejora de la red.

El número de reglas, por ejemplo, varía ampliamente: algunos trabajos tienen pocas reglas (7 o 9 reglas), mientras que otros llegan a utilizar hasta 50. Lo mismo ocurre con las épocas de entrenamiento, que varían desde valores bajos (20 o 30) hasta cifras considerablemente altas (2500 o incluso 3000). En cuanto a los datos utilizados en el entrenamiento, algunos autores trabajan con 200 o 300 datos, mientras que otros emplean más de 10.000. Además, varios artículos no especifican cómo distribuyeron los datos de entrenamiento y prueba, lo que limita la posibilidad de comparar resultados de manera objetiva.

Es importante señalar que las referencias en negritas corresponden a los trabajos previamente analizados en la Tabla 4.6, los cuales están específicamente enfocados en inversores. Este hecho resalta que, además de haber poca información sobre los parámetros usados, también son escasos los estudios aplicados a inversores.

Frente a este panorama, la propuesta de este trabajo destaca por mantener una configuración compacta y bien definida: se utilizaron 27 reglas derivadas de tres entradas con tres funciones de membresía cada una, entrenadas a partir de un conjunto específico de 35 casos seleccionados mediante diseño de experimentos. El entrenamiento se optimizó con el algoritmo PSO, lo que permitió lograr un error de predicción menor que 0.015, sin necesidad de grandes volúmenes de datos ni de redes complejas.

En futuros trabajos se podrían explorar inversores con mayor número de niveles o que incorporen pequeños filtros, lo que permitiría reducir aún más la THD y evaluar la mejora que se podría obtener. Además de poder analizar la optimización o ajuste de los parámetros de la red ANFIS propuesta, ya sea modificando el número de funciones de membresía, reglas difusas o épocas de entrenamiento, con el fin de mejorar aún más la precisión de los ángulos obtenidos y la generalización del modelo ante otros escenarios de variación de voltaje.

5. Conclusiones y trabajos futuros

5.1 Conclusiones

Este trabajo presentó un enfoque basado en la combinación de técnicas de IA, concretamente ANFIS optimizado mediante el algoritmo PSO, con el objetivo de minimizar la THD de tensión en un CHBMLI de siete niveles. Los resultados muestran la efectividad de este enfoque en la optimización de los ángulos de conmutación, consiguiendo una THD reducida en comparación con los métodos convencionales.

Durante los experimentos, se observó que la combinación de ANFIS y PSO no sólo proporciona una THD de tensión cercana a los valores calculados durante el entrenamiento, sino que, en algunos casos, supera las expectativas, obteniendo valores de THD inferiores en las simulaciones. Esto destaca la eficacia de los algoritmos metaheurísticos frente a los métodos tradicionales, a menudo basados en aproximaciones analíticas o de descenso de gradiente.

Las variaciones observadas en los resultados, con $\pm 0.1\%$, demuestran que el modelo puede manejar la incertidumbre y ajustar los ángulos de conmutación con eficacia, incluso en escenarios complejos. Sin embargo, se ha identificado que algunos experimentos muestran una ligera discrepancia en comparación con los valores entrenados, lo que puede atribuirse a la precisión decimal utilizada en los dispositivos físicos, como los IGBT o los MOSFET, que limita la precisión de conmutación.

El análisis comparativo también subraya el rendimiento del método propuesto en relación con otros estudios. Aunque los niveles de THD pueden variar debido al rango del análisis armónico y los filtros, el método propuesto logra resultados comparables sin etapas de filtrado complejas, obteniendo una THD del 10%. Esto supone una ventaja significativa en cuanto a simplicidad de diseño y rentabilidad para sistemas SAI y otras aplicaciones de potencia o aquellos alimentados por energías renovables.

Los resultados obtenidos en este estudio son alentadores y sugieren que este enfoque puede aplicarse a otras áreas de la electrónica de potencia. Los trabajos futuros podrían explorar inversores de nivel superior o filtrado adicional para reducir aún más la THD, reforzando la aplicabilidad de este método en sistemas de energías renovables y aplicaciones de alta precisión.

5.2 Trabajos Futuros

Como parte del trabajo futuro de esta investigación, se contempla la implementación funcional de la red ANFIS entrenada dentro del entorno de simulación en Simulink. Aunque en este estudio se desarrolló el entrenamiento y la validación de la red ANFIS utilizando datos optimizados por el algoritmo PSO, aún no se ha realizado su integración directa al modelo del inversor multinivel en Simulink.

Este proceso, que aquí se ha denominado “interfaz”, hace referencia a la conexión operativa entre la red ANFIS entrenada y el modelo del inversor, de tal manera que la red pueda recibir, como entrada, los valores de voltaje de las fuentes en tiempo real durante la simulación, y generar como salida los ángulos de conmutación correspondientes. Esta integración permitirá validar el funcionamiento de la red ANFIS bajo condiciones dinámicas, emulando el comportamiento real del sistema ante variaciones en los niveles de voltaje.

Una vez lograda esta conexión funcional, se podrán realizar simulaciones avanzadas en las que se verifique la adaptabilidad y precisión del modelo ante diferentes escenarios operativos. Posteriormente, como una fase siguiente, se planea la implementación física del sistema completo, utilizando un prototipo real del CHBMLI. Esta etapa permitirá validar experimentalmente los resultados obtenidos en simulación y analizar el comportamiento del sistema en condiciones reales.

También se propone como línea de investigación futura comparar el desempeño de la red ANFIS con otras técnicas de inteligencia artificial o algoritmos de optimización, tales como redes neuronales profundas, lógica difusa clásica o variantes híbridas con algoritmos como GA o ED.

Finalmente, se propone mejorar la arquitectura de la red ANFIS utilizada, ya sea mediante un mayor número de reglas, funciones de membresía adaptativas o nuevas estrategias de entrenamiento, con el objetivo de reducir aún más la THD y ampliar su aplicabilidad en sistemas que demanden alta precisión y eficiencia energética.

5.3 Productos académicos derivados del trabajo de tesis

Artículos de revista:

Como resultado de este trabajo de investigación, se desarrollaron productos académicos que contribuyen al avance del conocimiento en el área de la optimización de inversores multinivel.

1. Artículo publicado en la IEEE Transactions on Power Electronics (Volume: 39, Issue: 1, Enero 2024). DOI: [10.1109/TPEL.2023.3327014](https://doi.org/10.1109/TPEL.2023.3327014). Con el nombre: Adaptive Network-Based Fuzzy Inference System (ANFIS) Applied to Inverters: A Survey. Con un factor de impacto de 6.5, citescore de 15.3 y es Q1.
2. Artículo publicado en Electronics **2024**, 13(22), 4456; <https://doi.org/10.3390/electronics13224456> con el nombre: ANFIS-PSO-Based Optimization for THD Reduction in Cascaded Multilevel Inverter UPS Systems. Con un factor de impacto 2.6, citescore de 6.1 y es Q2.

Conferencias:

1. Participación en la Escuela Doctoral Electrónica de Potencia en Redes Inteligentes 2025, a la cual se realizaron los días 3, 4, 5 y 6 de junio de 2025, en el Instituto Tecnológico de Metropolitano - Medellín, Antioquia. Con el trabajo: Reducción de THD en un inversor multinivel mediante ANFIS optimizado con PSO. Link del evento: <https://sites.google.com/view/escueladoctoral2025edep-sg/inicio?authuser=0>.
2. Participación en el XVII Congreso Mexicano de Inteligencia Artificial (COMIA), en la Universidad Panamericana, Ciudad de México, México, en los días del 13 al 16 de mayo de 2025. Con el trabajo: Intelligent Optimization of Multilevel Inverter Switching Angles with ANFIS and PSO: A Simulation-Based Approach. Link del evento: <https://smia.mx/comia/2025/>.

6. Referencias

- [1] M. Aamir, K. Ahmed Kalwar, and S. Mekhilef, "Review: Uninterruptible Power Supply (UPS) system," *Renewable and Sustainable Energy Reviews*, vol. 58, pp. 1395-1410, 2016. <https://doi.org/10.1016/j.rser.2015.12.335>.
- [2] S. Keyan and X. Dehong, "Chapter 15 - Operation and control of uninterruptible power supply system," vol. 4, pp. 457-509, 2024. <https://doi.org/10.1016/B978-0-323-85622-5.00019-5>.
- [3] A. Nasiri and S. A. Hamidi, "19 - Uninterruptible Power Supplies," in *Power Electronics Handbook (Fourth Edition)*, M. H. Rashid, Ed.: Butterworth-Heinemann, 2018, pp. 641-657, <https://doi.org/10.1016/B978-0-12-811407-0.00021-0>.
- [4] M. P. Godoy, V. A. Uberti, A. d. R. Abaide, G. D. Guidali, L. R. Prade, and A. L. Keller, "Identifying and reducing harmonic distortion in an industrial uninterruptible power supply system," in *2020 6th International Conference on Electric Power and Energy Conversion Systems (EPECS)*, 2020, pp. 34-39, <https://doi.org/10.1109/EPECS48981.2020.9304969>.
- [5] H. Katir, A. Abouloifa, K. Noussi, I. Lachkar, and F. Giri, "Cascaded H-Bridge Inverters for UPS Applications: Adaptive Backstepping Control and Formal Stability Analysis," *IEEE Control Systems Letters*, vol. 6, pp. 145-150, 2022. <https://doi.org/10.1109/LCSYS.2021.3051875>.
- [6] E. Kabalci, "Multilevel Inverters Introduction and Emergent Topologies," in *Multilevel Inverters*, E. Kabalci, Ed., 2021, pp. 1-27, 10.1016/C2019-0-04051-4.
- [7] S. Choudhury, M. Bajaj, T. Dash, S. Kamel, and F. Jurado, "Multilevel Inverter: A Survey on Classical and Advanced Topologies, Control Schemes, Applications to Power System and Future Prospects," *Energies*, vol. 14, no. 18, 2021. <https://doi.org/10.3390/en14185773>.
- [8] J. H. S. Azuela and F. R. Cortés, *Inteligencia artificial aplicada a Robótica y Automatización*. México: ALFAOMEGAGRUPOEDITOR,S.A.deC.V., 2020, p. 382
- [9] P. Ponce, *INTELIGENCIA ARTIFICIAL - Con Aplicaciones a la Ingeniería*. Alfaomega, 2010, p. 376
- [10] P. Vas, *Artificial-Intelligence-Based Electrical Machines and Drives Application of Fuzzy, Neural, Fuzzy-neural, and Genetic-Algorithm-based Techniques*. Oxford Science Publications 1999 <https://doi.org/10.1093/oso/9780198593973.001.0001>.
- [11] C. Blum, A. Roli, and E. Alba, "An Introduction to Metaheuristic Techniques," in *Parallel Metaheuristics A New Class of Algorithms* A. Y. Zomaya, Ed.: A JOHN WILEY & SONS, INC., PUBLICATION, 2005 https://doi.org/10.1002/0471739383.ch1?urlappend=%3Futm_source%3Dresearchgate.net%26utm_medium%3Darticle.
- [12] Z. Shuai, B. Frede, and W. Huai, "Chapter 8 - Artificial intelligence-assisted data-driven control of power electronics systems," vol. 4, pp. 219-239, 2024. <https://doi.org/10.1016/B978-0-323-85622-5.00005-5>.
- [13] K. V. Shihabudheen and G. N. Pillai, "Recent advances in neuro-fuzzy system: A survey," *Knowledge-Based Systems*, vol. 152, pp. 136-162, 2018. <https://doi.org/10.1016/j.knosys.2018.04.014>.

- [14] P. V. de Campos Souza, "Fuzzy neural networks and neuro-fuzzy networks: A review the main techniques and applications used in the literature," *Applied Soft Computing*, vol. 92, 2020. <https://doi.org/10.1016/j.asoc.2020.106275>.
- [15] *IEEE Recommended Practice and Requirements for Harmonic Control in Electric Power Systems* 2022.
- [16] I. A. Adeyemo, J. A. Ojo, and D. O. Badajide, "Artificial Intelligence Approach to Real-Time Selective Harmonic Elimination in Voltage Source Multilevel Inverter," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 8, no. 4, pp. 7-13, November 2018. <https://www.ijsce.org/wp-content/uploads/papers/v8i4/D3160118418.pdf>.
- [17] M.YRafiq, G.Bugmann, and D.JEasterbrook, "Neural network design for engineering applications.pdf," *Computers & Structures*, vol. 79, no. 17, pp. 1541-1552, July 2001. [https://doi.org/10.1016/S0045-7949\(01\)00039-6](https://doi.org/10.1016/S0045-7949(01)00039-6).
- [18] L. V. Fausett, *Fundamentals of Neural Networks_ Architectures, Algorithms, and Applications*. Prentice-Hall, 1994, <http://103.203.175.90:81/fdScript/RootOfEBooks/E%20Book%20collection%20-%202025/CSE%20%20IT%20AIDS%20ML/Fundamentals%20of%20Neural%20Networks.pdf>.
- [19] D. Karaboga and E. Kaya, "Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey," *Artificial Intelligence Review*, vol. 52, no. 4, pp. 2263-2293, 2018. <https://doi.org/10.1007/s10462-017-9610-2>.
- [20] V. S. Jigajinni and V. Upendranath, "Comparison of ANFIS and ANN Techniques in the Simulation of a Typical Aircraft Fuel System Health Management," *International Journal of Artificial Intelligence & Applications*, vol. 9, no. 4, pp. 43-55, 2018.10.5121/ijaia.2018.9404, <https://ssrn.com/abstract=3280193>.
- [21] J. Sobhani, M. Najimi, A. R. Pourkhorshidi, and T. Parhizkar, "Prediction of the compressive strength of no-slump concrete: A comparative study of regression, neural network and ANFIS models," *Construction and Building Materials*, vol. 24, no. 5, pp. 709-718, 2010. <https://doi.org/10.1016/j.conbuildmat.2009.10.037>.
- [22] J.-S. R. Jang, "ANFIS : Adaptive-Network-Based Fuzzy Inference System," *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETIC*, vol. 23, pp. 665-685, June 1993. <https://doi.org/10.1109/21.256541>.
- [23] O. S. Vargas, S. E. D. L. Aldaco, J. A. Alquicira, L. G. Vela-Valdés, and A. R. L. Núñez, "Adaptive Network-Based Fuzzy Inference System (ANFIS) Applied to Inverters: A Survey," *IEEE Transactions on Power Electronics*, vol. 39, no. 1, pp. 869-884, 2024. <https://doi.org/10.1109/TPEL.2023.3327014>.
- [24] H. Ahmed and J. Glasgow, "Swarm Intelligence: Concepts, Models and Applications," School of Computing Queen's University Kingston, Ontario, Canada K7L3N62012, <https://doi.org/10.13140/2.1.1320.2568>.
- [25] E. M. Montes, "Paradigmas emergentes en algoritmos bio-inspirados," in *Inteligencia Aritificial*, Alfaomega, Ed.: Alfaomega, 2006, pp. 504-533.
- [26] K. Y. Gómez Díaz, S. E. de León Aldaco, J. Aguayo Alquicira, and L. G. Vela Valdés, "THD Minimization in a Seven-Level Multilevel Inverter Using the TLBO Algorithm," vol. 4, no. 3, pp. 1761-1786, 2023. <https://doi.org/10.3390/eng4030100>.
- [27] F. Blaabjerg, Y. Yang, K. A. Kim, and J. Rodriguez, "Power Electronics Technology for Large-Scale Renewable Energy Generation," *Proceedings of the IEEE*, vol. 111, no. 4, pp. 335-355, 2023. <https://doi.org/10.1109/JPROC.2023.3253165>.

- [28] B. K. Bose, "Power Electronics, Smart Grid, and Renewable Energy Systems," *Proceedings of the IEEE*, vol. 105, no. 11, pp. 2011-2018, 2017. <https://doi.org/10.1109/JPROC.2017.2745621>.
- [29] S. Zhao, F. Blaabjerg, and H. Wang, "An Overview of Artificial Intelligence Applications for Power Electronics," *IEEE Transactions on Power Electronics*, vol. 36, no. 4, pp. 4633-4658, 2021. <https://doi.org/10.1109/TPEL.2020.3024914>.
- [30] J. VIEIRA, F. M. DIAS, and A. MOTA, "Neuro-Fuzzy Systems: A Survey," presented at the 5th WSEAS NNA International Conference on Neural Networks and Applications, Udine, Italia, 2004. https://www.researchgate.net/publication/310047437_Neuro-Fuzzy_Systems_A_Survey
- [31] A. Abraham, "Neuro Fuzzy Systems: State-of-the-Art Modeling Techniques," presented at the Connectionist Models of Neurons, Learning Processes, and Artificial Intelligence. IWANN 2001, Lecture Notes in Computer Science, Berlin, Heidelberg, 2001. https://doi.org/10.1007/3-540-45720-8_30
- [32] C. Grosan and A. Abraham, *Intelligent Systems A Modern Approach*, 1 ed. (Intelligent Systems Reference Library No17). Springer, 2011, <https://link.springer.com/book/10.1007/978-3-642-21004-4>.
- [33] C.-T. Lin and C. S. G. Lee, "Neural Network based Fuzzy Logic Control and Decision System," presented at the IEEE Transactions on computers, 1991. <https://doi.org/10.1109/12.106218>
- [34] H. R. Berenji and P. Khedkar, "Learning and Tuning Fuzzy Logic Controllers through Reinforcements," presented at the IEEE TRANSACTIONS ON NEURAL NETWORKS, 1992. <https://doi.org/10.1109/72.159061>
- [35] D. Nauck and R. Kruse, "Neuro-Fuzzy Systems for Function Approximation," presented at the 4th International Workshop Fuzzy-Neuro Systems, 1997. [https://doi.org/10.1016/S0165-0114\(98\)00169-9](https://doi.org/10.1016/S0165-0114(98)00169-9)
- [36] T. S, O. T, and A. T, "Deep combination of Fuzzy Inference and Neural Network in Fuzzy Inference software- FINEST," *Fuzzy Sets and Systems* 82, pp. 151-160, 1996. Deep combination of Fuzzy Inference and Neural Network in Fuzzy Inference software- FINEST.
- [37] S. SM, T.-G. NN, and V. SJ, "FUN: Optimization of Fuzzy Rule Based Systems Using Neural Networks," presented at the Proceedings of IEEE Conference on Neural Networks, San Francisco, March 1993, <https://doi.org/10.1109/ICNN.1993.298575>
- [38] J. C. Feng and L. C. Teng, "An Online Self Constructing Neural Fuzzy Inference Network and its Applications," *IEEE Transactions on Fuzzy Systems*, vol. 6, No.1, pp. 12-32, 1998. <https://doi.org/10.1109/91.660805>.
- [39] M. Figueiredo and F. Gomide, "Design of Fuzzy Systems Using Neuro-Fuzzy Networks," *IEEE Transactions on Neural Networks*, vol. 10, pp. 815-827, July 1999. <https://doi.org/10.1109/72.774229>.
- [40] K. N and Q. Song, "Dynamic Evolving Fuzzy Neural Networks with 'm-out-of-n' Activation Nodes for On-line Adaptive Systems," in "Technical Report TR99/04," University of Otago, Department of information science 1999, chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://kedri.aut.ac.nz/_data/assets/pdf_file/0018/23337/kas-tr99-04.pdf.
- [41] S. Kar, S. Das, and P. K. Ghosh, "Applications of neuro fuzzy systems: A brief review and future outline," *Applied Soft Computing*, vol. 15, pp. 243-259, 2014. <https://doi.org/10.1016/j.asoc.2013.10.014>.

- [42] Z. J. Viharos and K. B. Kis, "Survey on Neuro-Fuzzy systems and their applications in technical diagnostics and measurement," *Measurement*, vol. 67, pp. 126-136, 2015. <https://doi.org/10.1016/j.measurement.2015.02.001>.
- [43] M. N. M. Salleh and K. Hussain, "A Review of Training Methods of ANFIS for Applications in Business and Economics," *International Journal of u- and e- Service, Science and Technology*, vol. 9, no. 7, pp. 165-172, 2016. <https://doi.org/10.14257/ijunesst.2016.9.7.17>.
- [44] Stuti, M. K. Singh, and S. Srivastava, "ANFIS based four quadrant chopper control of separately excited DC motor: A literature review," presented at the International Conference on Control, Computing, Communication and Materials (ICCCCM), Allahbad, India, 2016. <https://doi.org/10.1109/ICCCCM.2016.7918266>
- [45] M. Lasheen and M. Abdel-Salam, "Maximum power point tracking using Hill Climbing and ANFIS techniques for PV applications: A review and a novel hybrid approach," *Energy Conversion and Management*, vol. 171, pp. 1002-1019, 2018. <https://doi.org/10.1016/j.enconman.2018.06.003>.
- [46] M. Alghieth, "Review of an ANFIS Methodology-Based StockMarket Prediction System," presented at the The Eighteenth International Conference on Networks, Valencia, Spain, March 24 - 28, 2019. https://personales.upv.es/thinkmind/dl/conferences/icn/icn_2019/icn_2019_5_10_300_06.pdf
- [47] K. Janardhana *et al.*, "ANFIS modeling of biodiesels' physical and engine characteristics: A review," *Heat Transfer*, vol. 50, no. 8, pp. 8052-8079, 2021. <https://doi.org/10.1002/hjt.22266>.
- [48] P. H. D. Nguyen and A. Robinson Fayek, "Applications of fuzzy hybrid techniques in construction engineering and management research," *Automation in Construction*, vol. 134, 2022. <https://doi.org/10.1016/j.autcon.2021.104064>.
- [49] M. I. S. Guerra, F. M. U. de Araújo, J. T. de Carvalho Neto, and R. G. Vieira, "Survey on adaptative neural fuzzy inference system (ANFIS) architecture applied to photovoltaic systems," *Energy Systems*, 2022. <https://doi.org/10.1007/s12667-022-00513-8>.
- [50] T. R. Sumithira and A. N. Kumar, "Elimination of Harmonics in Multilevel Inverters Connected to Solar Photovoltaic Systems Using ANFIS: An Experimental Case Study," *Journal of Applied Research and Technology*, vol. 11, pp. 124-132, 2013. [https://doi.org/10.1016/S1665-6423\(13\)71521-9](https://doi.org/10.1016/S1665-6423(13)71521-9)
- <https://jart.icat.unam.mx/index.php/jart/article/view/336>.
- [51] M. Singh and A. Chandra, "Real-Time Implementation of ANFIS Control for Renewable Interfacing Inverter in 3P4W Distribution Network," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 1, pp. 121-128, 2013. <https://doi.org/10.1109/TIE.2012.2186103>.
- [52] S. L. Shimi;, T. Thakur;, J. Kumar;, S. Chatterji;, and D. Karanjkar, "MPPT based solar powered cascade multilevel inverter," presented at the International Conference on Microelectronics, Communication and Renewable Energy, Kanjirapally, India, 2013. <https://doi.org/10.1109/AICERA-ICMiCR.2013.6576041>
- [53] N. Karthik, M. S. Kalavathi, and S. A. Gafoor, "Shunt active power filter based on cascaded multilevel converters using ANFIS," presented at the Third International Conference on Computational Intelligence and Information Technology (CIIT 2013),, Mumbai, India, 18-19 Oct., 2013. <https://doi.org/10.1049/cp.2013.2645>

- [54] Jyothirmayi.C.J and Nasar.A, "Step modulated multilevel inverter incorporated upon ANFIS based intelligent PV MPPT," presented at the International Conference on Magnetics, Machines & Drives(AICERA/iCMMD), Kottayam, India, 24-26 July, 2014. <https://doi.org/10.1109/AICERA.2014.6908278>
- [55] K.Esaki Shenbaga Loga, SP Umayal, and M. Chandrachoodan, "Comparative Analysis of Shunt Active Compensators for Non Linear Loads using PI, FUZZY & ANFIS Controllers," presented at the International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), Kanyakumari, India, 10-11 July, 2014. <https://doi.org/10.1109/ICCICCT.2014.6992941>
- [56] P. Karuppusamy, A. M. Natarajan, and K. N. Vijeyakumar, "An Adaptive Neuro-Fuzzy Model to Multilevel Inverter for Grid Connected Photovoltaic System," *Journal of Circuits, Systems and Computers*, vol. 24, no. 05, 2015.https://doi.org/10.1142/S0218126615500668?urlappend=%3Futm_source%3Dresearchgate.net%26utm_medium%3Darticle.
- [57] S. Sarada, L. Baya Reddy, and K. Balaji, "Real-Time Implementation of Multi Level Inverter for 3P4W Distribution Network using ANFIS Control," *Indian Journal of Science and Technology*, vol. 8, no. 30, 2015.<https://dx.doi.org/10.17485/ijst/2015/v8i30/78761>.
- [58] T. Logeswaran, A. Senthikumar, and P. Karuppusamy, "Adaptive Neuro-Fuzzy Model for Grid-Connected Photovoltaic System," *International Journal of Fuzzy Systems*, vol. 17, no. 4, pp. 585-594, 2015.<https://doi.org/10.1007/s40815-015-0078-4>.
- [59] A. Dida and D. Benattous, "Modeling and Control of DFIG Through Back-to-Back Five Levels Converters Based on Neuro-Fuzzy Controller," *Journal of Control, Automation and Electrical Systems*, vol. 26, no. 5, pp. 506-520, 2015.<https://doi.org/10.1007/s40313-015-0190-6>.
- [60] G. Nageswara Rao, K. Chandra Sekhar, and P. Sangameswararaju, "An Effective Technique for Reducing Total Harmonics Distortion of Multilevel Inverter," *Journal of Intelligent Systems*, vol. 27, no. 3, pp. 433-446, 2018.https://doi.org/10.1515/jisys-2016-0210?urlappend=%3Futm_source%3Dresearchgate.net%26utm_medium%3Darticle.
- [61] R. M. Patil, M. S. Nagaraj, and P. S. Venkataramu, "Novel Design of a Neuro-Fuzzy (ANFIS) Controller to Improve the Power Dynamics for Minimization of Harmonics Using a Hybrid Scheme " *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 5, no. 9, pp. 7456-7468, September 2016.10.15662/IJAREEIE.2016.0509046
https://www.ijareeie.com/upload/2016/september/46_Novel.pdf.
- [62] T. P. Kumar and B. N. Kartheek, "A Neuro-Fuzzy Controller for Multilevel Renewable Energy System," *Indian Journal of Science and Technology*, vol. 9, no. 12, 2016.<https://doi.org/10.1109/ICEEOT.2016.7755491>.
- [63] G. D. A. Kumar, "Performance improvement of grid connected DFIG fed by three level diode clamped MLI using vector control," presented at the IEEE Region 10 Conference (TENCON), Singapore, 22-25 November, 2016. <https://doi.org/10.1109/TENCON.2016.7848063>
- [64] G. N. Rao, "Selective Harmonic Elimination Technique for a Cascaded Multilevel Inverter Using Intelligent Systems," *International Journal of Research in Information Technology*, vol. 1, no. 1, p. 7, 2017.

- [65] S. Veerakumar,, S. Sathishkumar,, and K. Kannan, "Novel harmonic elimination technique for cascaded H-bridge inverter using sampled reference frame," presented at the International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 18-19 July, 2017. <https://doi.org/10.1109/ICCMC.2017.8282626>
- [66] B. Pakkiraiah and G. Durga Sukumar, "Enhanced Performance of an Asynchronous Motor Drive with a New Modified Adaptive Neuro-Fuzzy Inference System-Based MPPT Controller in Interfacing with dSPACE DS-1104," *International Journal of Fuzzy Systems*, vol. 19, no. 6, pp. 1950-1965, 2017. <https://doi.org/10.1007/s40815-016-0287-5>.
- [67] J. S. Vinodhini and R. S. R. Babu, "High voltage bidirectional cascaded H bridge 21 level multilevel inverter with regenerative braking for locomotive by comparing with (PID & ANFIS) and without controller," presented at the Fourth International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), Chennai, India, 27-28 Feb. 2018, 2018. <https://doi.org/10.1109/AEEICB.2018.8480997>
- [68] T. L. Lakshmi; and M. G. Naik, "Voltage Regulation Comparison of Novel Photo Voltaic Multilevel Inverter Topology during Nonlinear PV Generation," presented at the IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, India, 18-19 May, 2018. <https://doi.org/10.1109/RTEICT42901.2018.9012367>
- [69] R. S. Rani;; C. S. Rao;; and M. V. Kumar, "Performance of PQ theory based Seven- level Cascaded Shunt Active Power Filter using PI, Fuzzy and ANFIS controllers for Harmonic Reduction," presented at the International Conference on Recent Trends in Electrical, Control and Communication (RTECC). , Malaysia, Malaysia, 20-22 March, 2018. <https://doi.org/10.1109/RTECC.2018.8625679>
- [70] P. Rajasekaran and V. J. Senthilkumar, "Comparison of Various Strategies in DTFC Based Multilevel NPC Inverter Fed Induction Drive for Torque Ripple Reduction," *Transactions on Electrical and Electronic Materials*, vol. 19, no. 4, pp. 290-303, 2018. <https://doi.org/10.1007/s42341-018-0039-3>.
- [71] "Adaptive Neural Fuzzy Inference System Controller for Seven-Level Packed U-Cell Inverter," presented at the 45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal, 14-17 October 2019. <https://doi.org/10.1109/IECON.2019.8927684>
- [72] S. R. Das, P. K. Ray, A. Mohanty, and H. Das, "Performance evaluation of multilevel inverter based hybrid active filter using soft computing techniques," *Evolutionary Intelligence*, vol. 14, no. 2, pp. 345-355, 2019. <https://doi.org/10.1007/s12065-019-00217-6>.
- [73] T. Lova Lakshmi and M. Gopichand Naik, "Soft-Computing Techniques for Voltage Regulation of Grid-Tied Novel PV Inverter at Different Case Scenarios," in *Soft Computing and Signal Processing*(Advances in Intelligent Systems and Computing, 2019, pp. 181-190, https://doi.org/10.1007/978-981-13-3393-4_19.
- [74] M. Kalarathi and N. Rathina Prabha, "Multilevel Parallel Active Power Filter Based on ANFIS Controller for Harmonic Alleviation," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 45, no. 2, pp. 633-644, 2020. <https://doi.org/10.1007/s40998-020-00384-8>.

- [75] G. Goswami and P. K. Goswami, "ANFIS Supervised PID Controlled SAPF for Harmonic Current Compensation at Nonlinear Loads," *IETE Journal of Research*, vol. 68, no. 5, pp. 3585-3596, 2020. <https://doi.org/10.1080/03772063.2020.1770134>.
- [76] S. R. Das, P. K. Ray, A. K. Mishra, and A. Mohanty, "Performance of PV integrated multilevel inverter for PQ enhancement," *International Journal of Electronics*, vol. 108, no. 6, pp. 945-982, 2020. <https://doi.org/10.1080/00207217.2020.1818848>.
- [77] M. Mehra, M. Babaie, A. Zafari, and K. Al-Haddad, "Passivity ANFIS-Based Control for an Intelligent Compact Multilevel Converter," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5141-5151, 2021. <https://doi.org/10.1109/TII.2021.3049313>.
- [78] M. Rupesh and D. T. S. Vishwanath, "Fuzzy and ANFIS Controllers to Improve the Power Quality of Grid Connected PV System with Cascaded Multilevel Inverter," *International Journal of Electrical and Electronics Research*, vol. 9, no. 4, pp. 89-95, 2021. <https://ijeer.forexjournal.co.in/papers-pdf/ijeer-090401.pdf>.
- [79] K. Muralikumar and P. Ponnambalam, "Comparison of Fuzzy and ANFIS Controllers for Asymmetrical 31-Level Cascaded Inverter With Super Imposed Carrier PWM Technique," *IEEE Access*, vol. 9, pp. 82630-82646, 2021. <https://doi.org/10.1109/ACCESS.2021.3086674>.
- [80] T. Atar, S. Balci, and A. Kayabaşı, "The analysis of three level inverter circuit with regard to current harmonic distortion by using ANFIS," *Journal of Energy Systems*, pp. 143-152, 2022. <https://doi.org/10.30521/jes.951487>.
- [81] Goggi Narasimha, Gummadavelli Naveen, Kavali Shireesha, Kollam Kartheek, and B. P. Kumar, "ANFIS BASED POWER QUALITY IMPROVEMENT IN CASCADED H-BRIDGE MULTILEVEL INVERTER," *International Research Journal of Modernization in Engineering Technology and Science*, vol. 04, no. 06, p. 10, 2022. https://www.irjmets.com/uploadedfiles/paper//issue_6_june_2022/26004/final/f_in_irjmets1655283304.pdf.
- [82] Y. G. Loana, I. G. Biru, C. S. Reddy, and K. Iqbal, "Design and Analysis of Front Side Modular Multilevel Converter for Smart Transformer in 15kV Arba Minch Distribution Network Using Diverse Controllers and Multicarrier Modulation," *Journal of Engineering*, vol. 2022, pp. 1-14, 2022. <https://doi.org/10.1155/2022/7678241>.
- [83] S. Potturi *et al.*, "Direct Torque Control of Induction Motor Using ConvLSTM Based on Gaussian Pillbox Surface," *Mathematical Problems in Engineering*, vol. 2022, pp. 1-21, 2022. <https://doi.org/10.1155/2022/4408271>.
- [84] S. A. Abdalla, S. S. Abdullah, and A. M. Kassem, "Performance enhancement and power management strategy of an autonomous hybrid fuel cell/wind power system based on adaptive neuro fuzzy inference system," *Ain Shams Engineering Journal*, vol. 13, no. 4, 2022. <https://doi.org/10.1016/j.asej.2021.101655>.
- [85] K. Chenchireddy and V. Jegathesan, "ANFIS Based Reduce Device Count DSTATCOM," *Journal of Applied Science and Engineering*, vol. 26, no. 11, pp. 1657-1666, 2023. <http://jase.tku.edu.tw/articles/jase-202311-26-11-0014.pdf>.
- [86] G. Nageswara Rao, P. Sangameswara Raju, and K. Chandra Sekhar, "Harmonic elimination of cascaded H-bridge multilevel inverter based active power filter controlled by intelligent techniques," *International Journal of Electrical Power & Energy Systems*, vol. 61, pp. 56-63, 2014. <https://doi.org/10.1016/j.ijepes.2014.02.023>.
- [87] S. Benhalima, A. Chandra, and M. Rezkallah, "Real-time experimental implementation of an LMS-adaline-based ANFIS controller to drive PV interfacing power system," *IET*

- Renewable Power Generation*, vol. 13, no. 7, pp. 1142-1152, 2019.<https://doi.org/10.1049/iet-rpg.2018.5245>.
- [88] J. B. Banu, J. Jeyashanthi, and A. T. Ansari, "DTC-IM drive using adaptive neuro fuzzy inference strategy with multilevel inverter," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 10, pp. 4799-4821, 2021.10.1007/s12652-021-03244-3.
- [89] Chaimaa Taghzaoui *et al.*, "Advanced Control of Single-Phase Shunt Active Power Filter Based on Flying Capacitor Multicell Converter," presented at the 14th IFAC Workshop on Adaptive and Learning Control Systems ALCOS 2022, Casablanca, Morocco, June 29 – July 01, 2022. <https://doi.org/10.1016/j.ifacol.2022.07.288>.
- [90] Jyothirmayi.C.J and Nasar.A, "A Real Time Algorithm Based Cascade Multilevel Inverter with Step Modulation Integrated Upon ANFIS Based Solar MPPT," presented at the International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), Kanyakumari, India, 10-11 July 2014, 2014. <https://doi.org/10.1109/ICCICCT.2014.6993180>
- [91] K. Premkumar and B. V. Manikandan, "Speed control of Brushless DC motor using bat algorithm optimized Adaptive Neuro-Fuzzy Inference System," *Applied Soft Computing*, vol. 32, pp. 403-419, 2015.<https://doi.org/10.1016/j.asoc.2015.04.014>.
- [92] N. K. Saxena and A. Kumar, "Reactive power control in decentralized hybrid power system with STATCOM using GA, ANN and ANFIS methods," *International Journal of Electrical Power & Energy Systems*, vol. 83, pp. 175-187, 2016.<https://doi.org/10.1016/j.ijepes.2016.04.009>.
- [93] S. Y. Nikouei, B. M. Dehkordi, and M. Niroomand, "A Genetic-Based Hybrid Algorithm Harmonic Minimization Method for Cascaded Multilevel Inverters with ANFIS Implementation," *Mathematical Problems in Engineering*, vol. 2021, p. 15, 2021.<https://doi.org/10.1155/2021/6642317>.
- [94] M. Muthuramalingam and P. S. Manoharan, "Comparative analysis of distributed MPPT controllers for partially shaded stand alone photovoltaic systems," *Energy Conversion and Management*, vol. 86, pp. 286-299, 2014.<https://doi.org/10.1016/j.enconman.2014.05.044>.
- [95] S. Sudha Letha, T. Thakur, and J. Kumar, "Harmonic elimination of a photo-voltaic based cascaded H-bridge multilevel inverter using PSO (particle swarm optimization) for induction motor drive," *Energy*, vol. 107, pp. 335-346, 2016.<https://doi.org/10.1016/j.energy.2016.04.033>.
- [96] H. R. Baghaee, M. Mirsalim, G. B. Gharehpetian, H. A. Talebi, and A. Niknam-Kumle, "A Hybrid ANFIS/ABC-based Online Selective Harmonic Elimination Switching Pattern for Cascaded Multi-level Inverters of Microgrids," *IEEE Transactions on Industrial Electronics*, pp. 1-1, 2017 2017.<https://doi.org/10.1109/TIE.2017.2694403>.
- [97] R. Francis and D. Meganathan, "An Improved ANFIS with Aid of ALO Technique for THD Minimization of Multilevel Inverters," *Journal of Circuits, Systems and Computers*, vol. 27, no. 12, 2018.https://doi.org/10.1142/S0218126618501931?urlappend=%3Futm_source%3Dresearchgate.net%26utm_medium%3Darticle.
- [98] S. Sivaranjani and R. Rajeswari, "Internet of Things Based Industrial Automation Using Brushless DC Motor Application with Resilient Directed Neural Network Control FED

- Virtual Z-Source Multilevel Inverter Topology," *Wireless Personal Communications*, vol. 102, no. 4, pp. 3239-3254, 2018.<https://doi.org/10.1007/s11277-018-5365-6>.
- [99] A. Darvish Falehi and M. Rafiee, "Optimal control of novel fuel cell-based DVR using ANFISC-MOSSA to increase FRT capability of DFIG-wind turbine," *Soft Computing*, vol. 23, no. 15, pp. 6633-6655, 2018.<https://doi.org/10.1007/s00500-018-3312-9>.
- [100] V. M. Jape, H. M. Suryawanshi, and J. P. Modak, "Flywheel replacement of IM with convenient power electronic circuitry: SSA-ANFIS," *International Journal of Electronics*, vol. 107, no. 7, pp. 1146-1165, 2020.<https://doi.org/10.1080/00207217.2020.1726482>.
- [101] T. Toumi, A. Allali, A. Meftouhi, O. Abdelkhalek, A. Benabdelkader, and M. Denai, "Robust control of series active power filters for power quality enhancement in distribution grids: Simulation and experimental validation," *ISA Trans*, vol. 107, pp. 350-359, Dec 2020.<https://doi.org/10.1016/j.isatra.2020.07.024>.
- [102] S. P. Bihari and P. K. Sadhu, "Design analysis of high level inverter with EANFIS controller for grid connected PV system," *Analog Integrated Circuits and Signal Processing*, vol. 103, no. 3, pp. 411-424, 2020.<https://doi.org/10.1007/s10470-019-01578-9>.
- [103] J. Tokachichu and T. R. D. Gaddam, "Performance analysis of a transmission line connected with UPFC designed with three level cascaded H bridge inverter with generalized SVM technique using PI, FUZZY LOGIC, ANN and ANFIS controllers," *Materials Today: Proceedings*, vol. 51, pp. 1243-1251, 2022.<https://doi.org/10.1016/j.matpr.2021.07.338>.
- [104] P. Hemachandu and V. C. V. Reddy, "A Critical Evaluation of Advanced Multi-Carrier Modulation Scheme for 15-Level Inverter via PSO-PID Controller," *Procedia Technology*, vol. 23, pp. 240-247, 2016.<https://doi.org/10.1016/j.protcy.2016.03.023>.
- [105] B. Ponnappalli and P. Natarajan, "Harmonic Level Minimization Using Neuro-Fuzzy-Based SVPWM," *IETE Journal of Research*, vol. 64, no. 2, pp. 247-262, 2017.<https://doi.org/10.1080/03772063.2017.1351324>.
- [106] P. Thamizhazhagan and S. Sutha, "Adaptive vector control reference strategy based speed and torque control of Permanent Magnet Synchronous Motor," *Microprocessors and Microsystems*, vol. 74, 2020.<https://doi.org/10.1016/j.micpro.2020.103007>.
- [107] T. N. Prasad et al., "Power management in hybrid ANFIS PID based AC-DC microgrids with EHO based cost optimized droop control strategy," *Energy Reports*, vol. 8, pp. 15081-15094, 2022.<https://doi.org/10.1016/j.egyr.2022.11.014>.
- [108] M. D. Patil and P. B. M., "Performance Enhancement of Multi-Level Inverters with PSO-ANFIS Computational Techniques," *Journal of Electrical Systems* vol. 20, 2024.<https://journal.esrgroups.org/jes/article/view/5626/4018>.

Anexo A Programa principal ANIFS

```
clc;
clear;
close all;
warning('off', 'all');
% Definir la función objetivo (THD)
funObjetivo = @thd_71;
% Número de casos
numCasos = 35% #35 numero de casos con caso critico;
%% Generar datos de entrenamiento para caso
datosEntrenamiento = cell(numCasos, 1);
numMuestras = 1000;

%%Casos
for caso = 1:numCasos
    % Voltajes para cada caso
    if caso == 1
        V1 = 62.05;
        V2 = 62.05;
        V3 = 62.05;
    elseif caso == 2
        V1 = 57.46;
        V2 = 62.05;
        V3 = 62.05;
    elseif caso == 3
        V1 = 55.692;
        V2 = 62.05;
        V3 = 62.05;
    elseif caso == 4
        V1 = 53.55;
        V2 = 62.05;
        V3 = 62.05;
    elseif caso == 5
        V1 = 48.467;
        V2 = 62.05;
        V3 = 62.05;
    elseif caso == 6
        V1 = 57.46;
        V2 = 57.46;
        V3 = 62.05;
    elseif caso == 7
        V1 = 55.692;
        V2 = 57.46;
        V3 = 62.05;
    elseif caso == 8
        V1 = 53.55;
        V2 = 57.46;
        V3 = 62.05;
    elseif caso == 9
        V1 = 48.467;
        V2 = 57.46;
        V3 = 62.05;
    elseif caso == 10
        V1 = 55.692;
```

```

        V2 = 55.692;
        V3 = 62.05;
elseif caso == 11
    V1 = 53.55;
    V2 = 55.692;
    V3 = 62.05;
elseif caso == 12
    V1 = 48.467;
    V2 = 55.692;
    V3 = 62.05;
elseif caso == 13
    V1 = 53.55;
    V2 = 53.55;
    V3 = 62.05;
elseif caso == 14
    V1 = 48.467;
    V2 = 53.55;
    V3 = 62.05;
elseif caso == 15
    V1 = 48.467;
    V2 = 48.467;
    V3 = 62.05;
elseif caso == 16
    V1 = 57.46;
    V2 = 57.46;
    V3 = 57.46;
elseif caso == 17
    V1 = 55.692;
    V2 = 57.46;
    V3 = 57.46;
elseif caso == 18
    V1 = 53.55;
    V2 = 57.46;
    V3 = 57.46;
elseif caso == 19
    V1 = 48.467;
    V2 = 57.46;
    V3 = 57.46;
elseif caso == 20
    V1 = 55.692;
    V2 = 55.692;
    V3 = 57.46;
elseif caso == 21
    V1 = 53.55;
    V2 = 55.692;
    V3 = 57.46;
elseif caso == 22
    V1 = 48.467;
    V2 = 55.692;
    V3 = 57.46;
elseif caso == 23
    V1 = 53.55;
    V2 = 53.55;
    V3 = 57.46;
elseif caso == 24
    V1 = 48.467;
    V2 = 53.55;

```

```

        V3 = 57.46;
    elseif caso == 25
        V1 = 48.467;
        V2 = 48.467;
        V3 = 57.46;
    elseif caso == 26
        V1 = 55.692;
        V2 = 55.692;
        V3 = 55.692;
    elseif caso == 27
        V1 = 53.55;
        V2 = 55.692;
        V3 = 55.692;
    elseif caso == 28
        V1 = 48.467;
        V2 = 55.692;
        V3 = 55.692;
    elseif caso == 29
        V1 = 53.55;
        V2 = 53.55;
        V3 = 55.692;
    elseif caso == 30
        V1 = 48.467;
        V2 = 53.55;
        V3 = 55.692;
    elseif caso == 31
        V1 = 48.467;
        V2 = 48.467;
        V3 = 55.692;
    elseif caso == 32
        V1 = 53.55;
        V2 = 53.55;
        V3 = 53.55;
    elseif caso == 33
        V1 = 48.467;
        V2 = 53.55;
        V3 = 53.55;
    elseif caso == 34
        V1 = 48.467;
        V2 = 48.467;
        V3 = 53.55;
    elseif caso == 35
        V1 = 48.467;
        V2 = 48.467;
        V3 = 48.467;
    end

    angulos = zeros(numMuestras, 3);

    for i = 1:numMuestras
        % Ajustar los límites y valores de los ángulos según el caso
        if caso == 1 || caso == 16 || caso == 26 || caso == 32 || caso == 35
            ang1 = max(8.545, min(randn() * 8.6928 - 0.2, 8.789));
            ang2 = max(27.843, min(randn() * 27.896 - 0.1, 28));
            ang3 = max(49.809, min(randn() * 49.8166 - 0.1, 50));
        end
    end

```

```

elseif caso == 2
    ang1 = max(7.78, min(randn() * 7.87 - 0.03, 8.0));
    ang2 = max(27.12, min(randn() * 27.3 - 0.03, 27.51));
    ang3 = max(49.48, min(randn() * 49.56 - 0.01, 49.62));

    elseif caso == 3
    ang1 = max(7.48, min(randn() * 7.67 - 0.01, 7.89));
    ang2 = max(26.92, min(randn() * 27.09 - 0.02, 27.25));
    ang3 = max(49.34, min(randn() * 49.53 - 0.02, 49.76));

elseif caso == 4
    ang1 = max(7.299, min(randn() * 7.4755 - 0.0312, 7.604));
    ang2 = max(26.63, min(randn() * 26.816 - 0.02, 27));
    ang3 = max(49.42, min(randn() * 49.69289 - 0.05, 49.85));

elseif caso == 5
    ang1 = max(6.897, min(randn() * 7.06 - 0.121, 7.202));
    ang2 = max(25.495, min(randn() * 25.697 - 0.013, 25.802));
    ang3 = max(49.39, min(randn() * 49.58 - 0.04, 49.701));

elseif caso == 6
    ang1 = max(7.98, min(randn() * 8.14 - 0.04, 8.325));
    ang2 = max(27.156, min(randn() * 27.37 - 0.03, 27.57));
    ang3 = max(49.225, min(randn() * 49.39 - 0.02, 49.52));

elseif caso == 7
    ang1 = max(7.68, min(randn() * 7.88 - 0.03, 7.98));
    ang2 = max(27.021, min(randn() * 27.152 - 0.02, 27.227));
    ang3 = max(49.198, min(randn() * 49.33 - 0.02, 49.51));

elseif caso == 8
    ang1 = max(7.48, min(randn() * 7.63 - 0.03, 7.81));
    ang2 = max(26.690, min(randn() * 26.87 - 0.02, 27));
    ang3 = max(49.17, min(randn() * 49.288 - 0.02, 49.35));

elseif caso == 9
    ang1 = max(7, min(randn() * 7.198 - 0.03, 7.314));
    ang2 = max(25.590, min(randn() * 25.799 - 0.02, 25.885));
    ang3 = max(49.278, min(randn() * 49.311 - 0.02, 49.45));

elseif caso == 10
    ang1 = max(7.78, min(randn() * 7.97 - 0.03, 8.10));
    ang2 = max(26.980, min(randn() * 27.17 - 0.02, 27.32));
    ang3 = max(49.178, min(randn() * 49.26 - 0.02, 49.435));

elseif caso == 11
    ang1 = max(7.58, min(randn() * 7.71 - 0.03, 7.891));
    ang2 = max(26.690, min(randn() * 26.896 - 0.02, 27));
    ang3 = max(48.987, min(randn() * 49.191 - 0.02, 49.35));

elseif caso == 12
    ang1 = max(7.129, min(randn() * 7.25 - 0.03, 7.41));
    ang2 = max(25.690, min(randn() * 25.83 - 0.034, 26));
    ang3 = max(49.05, min(randn() * 49.163 - 0.021, 49.33));

```

```

elseif caso == 13
    ang1 = max(7.68, min(randn() * 7.802 - 0.03, 8));
    ang2 = max(26.823, min(randn() * 26.91 - 0.02, 27.087));
    ang3 = max(48.987, min(randn() * 49.088 - 0.02, 49.245));

elseif caso == 14
    ang1 = max(7.1975, min(randn() * 7.31 - 0.034, 7.412));
    ang2 = max(25.690, min(randn() * 25.88 - 0.024, 25.97));
    ang3 = max(48.778, min(randn() * 48.97 - 0.02, 49.084));

elseif caso == 15
    ang1 = max(7.318, min(randn() * 7.49 - 0.03, 7.61));
    ang2 = max(25.790, min(randn() * 25.91 - 0.02, 26.092));
    ang3 = max(48.278, min(randn() * 48.44 - 0.02, 48.65));

elseif caso == 17
    ang1 = max(8.1928, min(randn() * 8.31 - 0.036, 8.461));
    ang2 = max(27.490, min(randn() * 27.621 - 0.021, 27.842));
    ang3 = max(49.578, min(randn() * 49.671 - 0.019, 49.835));

elseif caso == 18
    ang1 = max(7.883, min(randn() * 7.921 - 0.01, 8.081));
    ang2 = max(27.190, min(randn() * 27.34 - 0.02, 27.52));
    ang3 = max(49.378, min(randn() * 49.57 - 0.02, 49.78));

elseif caso == 19
    ang1 = max(7.28, min(randn() * 7.37 - 0.03, 7.51));
    ang2 = max(26.490, min(randn() * 26.62 - 0.02, 26.82));
    ang3 = max(49.378, min(randn() * 49.55 - 0.02, 49.652));

elseif caso == 20
    ang1 = max(8.38, min(randn() * 8.452 - 0.03, 8.613));
    ang2 = max(27.496, min(randn() * 27.67 - 0.02, 27.772));
    ang3 = max(49.578, min(randn() * 49.62 - 0.02, 49.75));

elseif caso == 21
    ang1 = max(7.939, min(randn() * 8.03 - 0.03, 8.141));
    ang2 = max(27.190, min(randn() * 27.37 - 0.02, 27.52));
    ang3 = max(49.378, min(randn() * 49.500 - 0.02, 49.65));

elseif caso == 22
    ang1 = max(7.28, min(randn() * 7.43 - 0.0156, 7.51));
    ang2 = max(26.497, min(randn() * 26.650 - 0.02, 26.72));
    ang3 = max(49.278, min(randn() * 49.43 - 0.02, 49.65));

elseif caso == 23
    ang1 = max(7.98, min(randn() * 8.18 - 0.03, 8.31));
    ang2 = max(27.390, min(randn() * 27.412 - 0.02, 27.62));
    ang3 = max(49.278, min(randn() * 49.42 - 0.02, 49.55));

elseif caso == 24
    ang1 = max(7.38, min(randn() * 7.51 - 0.03, 7.61));
    ang2 = max(26.490, min(randn() * 26.69 - 0.02, 26.82));
    ang3 = max(48.998, min(randn() * 49.29 - 0.02, 49.45));

```

```

elseif caso == 25
    ang1 = max(7.58, min(randn() * 7.72 - 0.03, 7.91));
    ang2 = max(26.590, min(randn() * 26.74 - 0.02, 26.97));
    ang3 = max(48.778, min(randn() * 48.98 - 0.02, 49.082));

elseif caso == 27
    ang1 = max(8.09, min(randn() * 8.22 - 0.03, 8.41));
    ang2 = max(27.390, min(randn() * 27.56 - 0.02, 27.72));
    ang3 = max(49.478, min(randn() * 49.64 - 0.02, 49.85));

elseif caso == 28
    ang1 = max(7.38, min(randn() * 7.51 - 0.0189, 7.61));
    ang2 = max(26.690, min(randn() * 26.87 - 0.02, 27.00));
    ang3 = max(49.378, min(randn() * 49.53 - 0.02, 49.65));

elseif caso == 29
    ang1 = max(8.243, min(randn() * 8.39 - 0.015, 8.51));
    ang2 = max(27.490, min(randn() * 27.616 - 0.016, 27.82));
    ang3 = max(49.478, min(randn() * 49.58 - 0.02, 49.71));

elseif caso == 30
    ang1 = max(7.48, min(randn() * 7.59 - 0.03, 7.68));
    ang2 = max(26.793, min(randn() * 26.91 - 0.02, 27.092));
    ang3 = max(49.278, min(randn() * 49.400 - 0.02, 49.545));

elseif caso == 31
    ang1 = max(7.68, min(randn() * 7.84 - 0.024, 8.00));
    ang2 = max(26.790, min(randn() * 26.97 - 0.02, 27.192));
    ang3 = max(48.978, min(randn() * 49.12 - 0.017, 49.25));

elseif caso == 33
    ang1 = max(7.64, min(randn() * 7.72 - 0.012, 7.911));
    ang2 = max(26.990, min(randn() * 27.15 - 0.02, 27.28));
    ang3 = max(49.39, min(randn() * 49.54 - 0.02, 49.712));

elseif caso == 34
    ang1 = max(7.968, min(randn() * 8.01 - 0.01, 8.21));
    ang2 = max(27.0957, min(randn() * 27.22 - 0.02, 27.341));
    ang3 = max(48.99, min(randn() * 49.29 - 0.02, 49.33));

end

% Almacenar los ángulos en la matriz
angulos(i, :) = [ang1, ang2, ang3];
end

THD = zeros(numMuestras, 1);
for i = 1:numMuestras
    THD(i) = funObjetivo(angulos(i, :), V1, V2, V3);
end

% Dividir datos en conjuntos de entrenamiento y prueba
proporcionEntrenamiento = 0.7;
numEntrenamiento = round(proporcionEntrenamiento * numMuestras);
% Almacenar datos de entrenamiento y prueba por caso

```

```

    datosEntrenamiento{caso}= [angulos(1:numEntrenamiento, :),
    THD(1:numEntrenamiento)];
    datosPrueba{caso}= [angulos(numEntrenamiento+1:end, :),
    THD(numEntrenamiento+1:end)];

    %datosEntrenamiento{caso} = [angulos, THD];
    %% Crear el FIS
    fis = genfis3(datosEntrenamiento{caso}(:, 1:3),
datosEntrenamiento{caso}(:, 4));
    % Definir los datos de entrenamiento
    data.traininput = datosEntrenamiento{caso}(:, 1:3);
    data.trainoutput = datosEntrenamiento{caso}(:, 4);
    % Entrenar ANFIS+PSO
    data.V1 = V1;
    data.V2 = V2;
    data.V3 = V3;
    bestfis = TrainUsingPso(fis, data);
    % **Asignar el modelo entrenado a la variable anfisModel**
    anfisModel = bestfis;
    % Evaluar el FIS entrenado en datos de prueba para el caso actual
    testanfis_pso = evalfis(datosPrueba{caso}(:, 1:3), bestfis);
    angulos_anfis_pso = evalfis(datosPrueba{caso}(:, 1:3), bestfis);

    %% Gráficas para entrenamiento
    figEntrenamiento = figure;
    subplot(2, 2, [1 2]);
    % Crear un vector de colores para los ángulos
    colores = {'r', 'g', 'b'};

    for i = 1:3
        scatter(datosEntrenamiento{caso}(:,
i),datosEntrenamiento{caso}(:, 4), 'filled', ...
'DisplayName', sprintf('Ángulo %d', i), 'MarkerFaceColor',
colores{i});
        hold on;
    end
    % Mostrar los resultados ANFIS+PSO
    scatter(data.traininput, evalfis(data.traininput, bestfis),
'filled','DisplayName', 'THD pronosticada'); %Checar aqui por los datos
generados
    title(sprintf('Entrenamiento - Caso %d', caso));
    legend('show');
    xlabel('Ángulos'); % Etiqueta del eje x
    ylabel('THD'); % Etiqueta del eje y

    subplot(2, 2, 3);
    plot(evalfis(datosEntrenamiento{caso}(:, 1:3), bestfis) -
datosEntrenamiento{caso}(:, 4));
    title('Error del entrenamiento');

    subplot(2, 2, 4);
    histfit(evalfis(datosEntrenamiento{caso}(:, 1:3), bestfis) -
datosEntrenamiento{caso}(:, 4));
    title('Error del entrenamiento - Histograma');

    %% Gráficas para prueba

```

```

    figPrueba3d = figure;
    scatter3(datosPrueba{caso}(:, 1), datosPrueba{caso}(:, 2),
datosPrueba{caso}(:, 3), ...
    60, datosPrueba{caso}(:, 4), 'filled', 'DisplayName', 'Datos reales');
    hold on;
    scatter3(datosPrueba{caso}(:, 1), datosPrueba{caso}(:,
2), datosPrueba{caso}(:, 3), ...
    50, testanfis_pso, 'filled', 'DisplayName', 'Resultados ANFIS+PSO');
    title(sprintf('Prueba - Caso %d', caso));
    legend('show');
    xlabel('Ángulo 1'); % Etiqueta del eje x
    ylabel('Ángulo 2'); % Etiqueta del eje y
    zlabel('Ángulo 3'); % Etiqueta del eje z
    colorbar; %Checar esto

figPrueba = figure;
%Figura 2d de la grafica de los angulos
% Definimos un conjunto de colores para cada ángulo
colores = {'b', 'r', 'g'};
% Graficamos los datos reales de cada ángulo
for i = 1:3
    scatter(datosPrueba{caso}(:, i), datosPrueba{caso}(:, 4), ...
[], colores{i}, 'filled', 'DisplayName', sprintf('Ángulo %d', i));
    hold on
end
legend('show');
xlabel('Ángulos'); % Etiqueta del eje x
ylabel('THD'); % Etiqueta del eje y

%Graficas del error
subplot(2, 2, 3);
plot(testanfis_pso - datosPrueba{caso}(:, 4));
title('Error de la prueba');

subplot(2, 2, 4);
histfit(testanfis_pso - datosPrueba{caso}(:, 4));
title('Error de la prueba - Histograma');
end
% Concatenar datos de entrenamiento y prueba para todos los casos
datosEntrenamientoTotal = cat(1, datosEntrenamiento{:});
datosPruebaTotal = cat(1, datosPrueba{:});

% Obtener el número total de muestras y el número de casos
numMuestrasTotal = size(datosPruebaTotal, 1);
numCasos = numel(datosPrueba);

% Definir colores para cada caso
colores = lines(numCasos);

% Graficar THD de prueba de salida para todos los casos en un diagrama de
dispersión
figure;
hold on;
for caso = 1:numCasos
    % Obtener las muestras correspondientes al caso actual
    muestrasCaso = datosPrueba{caso}(:, 4);

```

```

    % Calcular el rango de índices para las muestras del caso actual
    rangoIndices = ((caso - 1) * numMuestrasTotal / numCasos) + 1 : (caso
* numMuestrasTotal / numCasos);

    % Graficar las muestras del caso actual con un color diferente
    scatter(repelem(caso, length(muestrasCaso)), muestrasCaso, 'filled',
'MarkerFaceColor', colores(caso, :));
end
hold off;

xlabel('Caso');
ylabel('THD');
title('THD de Prueba de Salida para Todos los Casos');
xticks(1:numCasos);
xticklabels({'Caso 1', 'Caso 2', 'Caso 3', 'Caso 4', 'Caso 5', 'Caso 6',
'Caso 7', 'Caso 8', 'Caso 9', 'Caso 10'...
, 'Caso 11', 'Caso 12', 'Caso 13', 'Caso 14', 'Caso 15', 'Caso 16',
'Caso 17', 'Caso 18', 'Caso 19', 'Caso 20'...
, 'Caso 21', 'Caso 22', 'Caso 23', 'Caso 24', 'Caso 25', 'Caso 26',
'Caso 27', 'Caso 28', 'Caso 29', 'Caso 30'...
, 'Caso 31', 'Caso 32', 'Caso 33', 'Caso 34', 'Caso 35'}); %
Etiquetas de los casos en el eje x
grid on;

% Entrenamiento de ANFIS
%anfisModel = anfis(trainingData, options);
%%figure;
%%plotfis(anfis_model);
disp(anfisModel);
%% Guardar el modelo ANFIS entrenado en un archivo .fis
%%writeFIS(anfisModel, 'modeloEntrenado.fis');

```

Anexo B Código Matlab Funciones

Costfis.m

```
function [z, out] = costfis(p, fis, data, V1, V2, V3)
    % Función para evaluar el rendimiento de un FIS con parámetros dados.
    fis = SetFisParams(fis, p); % Establece los parámetros del FIS según
    los valores en 'ans'.
    x = data.traininput; % Datos de entrada de entrenamiento
    t = data.trainoutput; % Salidas reales correspondientes
    y = evalfis(fis, x); % Evalúa el FIS en los datos de entrada x para
    obtener las salidas estimadas y.
    % Puedes utilizar otras métricas de rendimiento según sea necesario
    [THD, ansr] = thd_71(p,V1,V2,V3);
    % Combina THD y ansr según tus objetivos
    z = THD + ansr;
    out.fis = fis;
    out.THd = THD;
    out.ansr = ansr;
```

GetFisParams.m

```
function p=GetFisParams(fis)
% Función para extraer los parámetros de las funciones de membresía de un
FIS.
p = []; % Inicializa un vector vacío para almacenar los parámetros de las
funciones de membresía.
ninput = numel(fis.input); % Calcula el número de variables de entrada en
el FIS.
% Bucle para las variables de entrada
for i = 1:ninput
    nMF=numel(fis.input(i).mf); % Calcula el número de funciones de
    membresía en la variable de entrada.
    for j=1:nMF
        p=[p fis.input(i).mf(j).params]; % Agrega los parámetros al
        vector p.
    end
end
noutput=numel(fis.output); % Calcula el número de variables de salida en
el FIS.
% Bucle para las variables de salida
for i=1:noutput
    nMF=numel(fis.output(i).mf); % Calcula el número de funciones de
    membresía en la variable de salida.
    for j=1:nMF
        p=[p fis.output(i).mf(j).params]; % Agrega los parámetros al
        vector p.
    end
end
end
```

SetFisParams.m

```
function fis=SetFisParams(fis,p)
% Función para extraer los parámetros de las funciones de membresía de un
FIS.
```

```

ninput = numel(fis.input); % Calcula el número de variables de entrada en
el FIS.

% Bucle para las variables de entrada
for i = 1:ninput
    nMf = numel(fis.input(i).mf);
    for j = 1:nMf
        k=numel(fis.input(i).mf(j).params);
        fis.input(i).mf(j).params=p(1:k);
        p(1:k)=[];
    end
end
noutput = numel(fis.output); % Calcula el número de variables de salida
en el FIS.
% Bucle para las variables de salida
for i = 1:noutput
    nMf = numel(fis.output(i).mf); % Calcula el número de funciones de
membresía en la variable de salida.
    for j = 1:nMf
        k=numel(fis.output(i).mf(j).params);
        fis.output(i).mf(j).params=p(1:k);
        p(1:k)=[];
    end
end
end
end

```

TrainUsingPSO.m

```

function bestfis=TrainUsingPso(fis,data)
p0=GetFisParams(fis);
%%obtener los voltajes de cada caso
V1 = data.V1;
V2 = data.V2;
V3 = data.V3;
Problem.costfunction = @(x) costfis(x, fis, data, V1, V2, V3);
Problem.nvar=numel(p0);
alpha=0.1;
Problem.varmin = -10^alpha * ones(1, Problem.nvar);
Problem.varmax = 10^alpha * ones(1, Problem.nvar);
%%PSO Parametros
Params.maxit=20;
Params.npop=120;
%% Run PSO
results=RunPso(Problem,Params);
p=results.Globalbest.position.*p0;
bestfis=SetFisParams(fis,p);
end

```

RunPso.m

```

function results=RunPso(Problem,Params)
%% Definición del problema
costfunction=Problem.costfunction;
nvar=Problem.nvar;
varsize=[1 nvar];
varmin=Problem.varmin;
varmax=Problem.varmax;

```

```

%%Parametros PSO
maxit=Params.maxit;
npop=Params.npop;
w=0.9;
wdamp=0.4;
c1=2;
c2=2;
%% inicialización
empty_particle.position=[];
empty_particle.velocity=[];
empty_particle.cost=[];
empty_particle.best.position=[];
empty_particle.best.cost=[];
Globalbest.position=[];
Globalbest.cost=inf;
particle= repmat(empty_particle, npop, 1);
for i=1:npop
    if i>1
        particle(i).position = unifrnd(varmin, varmax, varsize);
    else
        particle(i).position = ones(varsize);
    end
    particle(i).cost = costfunction(particle(i).position);
    particle(i).velocity = zeros(varsize);
    particle(i).best.position = particle(i).position;
    particle(i).best.cost = particle(i).cost;
    if particle(i).cost < Globalbest.cost
        Globalbest.position = particle(i).position;
        Globalbest.cost = particle(i).cost;
    end
end
bestcost = zeros(maxit, 1);
%%mainloop
for it=1:maxit

    for i=1:npop
        particle(i).velocity = w * particle(i).velocity ...
            + c1 * rand(varsize) .* (particle(i).best.position -
particle(i).position) ...
            + c2 * rand(varsize) .* (Globalbest.position -
particle(i).position);
        % Actualizar posición
        particle(i).position = particle(i).position +
particle(i).velocity;
        % Evaluar
        particle(i).cost = costfunction(particle(i).position);
        % Actualizar mejor posición personal
        if particle(i).cost < particle(i).best.cost
            particle(i).best.position = particle(i).position;
            particle(i).best.cost = particle(i).cost;
            % Actualizar mejor posición global
            if particle(i).best.cost < Globalbest.cost
                Globalbest = particle(i).best;
            end
        end
    end
    bestcost(it) = Globalbest.cost;
end

```

```

        disp(['Iteración ' num2str(it) ': Mejor costo '
num2str(bestcost(it))]);
        w = w * wdamp;
    end
    %%results
    results.Globalbest=Globalbest;
    results.bestcost=bestcost;
    figure;
    plot(bestcost);
end

%%% Función Objetivo THD/CMLI 7NIVELES
function [THD, ansr] = thd_7l(x, V1, V2, V3)
sum=0;
ansr=0;
for n = 3:2:49
    term = (V1 * cos(n * pi * x(1) / 180) + V2 * cos(n * pi * x(2) / 180)
+ V3 * cos(n * pi * x(3) / 180)) / n;
    sum = sum + term^2;
end
denominator = V1 * cos(pi * x(1) / 180) + V2 * cos(pi * x(2) / 180) + V3
* cos(pi * x(3) / 180);
THD = (sqrt(sum) / denominator)*100;
disp('Resultado THD:');
disp(THD);
% Verificaciones de restricciones
if x(1) < 0 || x(2) < 0 || x(3) < 0 || x(1) > 90 || x(2) > 90 || x(3)
> 90
    ansr = ansr + 200; % Penalización más severa
end
if x(1) < x(2) && x(2) < x(3) && x(1) < x(3)
else
    ansr = ansr + 400; % Penalización más severa
end
disp('Resultado final de ansr:');
disp(ansr);
end

```