

Modelamiento de un horno de arco eléctrico utilizando una metodología basada en redes neuronales artificiales.

Cristian David Sosa Aguirre

Trabajo de grado presentado como requisito
parcial para optar al título de
Ingeniero Electricista

Director
Ph.D (c). Jesser James Marulanda Durango

UNIVERSIDAD TECNOLÓGICA DE PEREIRA
Facultad de Ingenierías
Programa de Ingeniería Eléctrica
Pereira, 2 de julio de 2019



Nota de Aceptación

Ph.D (c). Jesser James Marulanda Durango. Director

Jurado

Pereira, 2 de julio de 2019

Modelamiento de un horno de arco eléctrico utilizando una metodología basada en redes neuronales artificiales.

©Cristian David Sosa Aguirre

Pereira, 2 de julio de 2019

Programa de Ingeniería Eléctrica.

Universidad Tecnológica de Pereira

La Julita. Pereira(Colombia)

TEL: (+57)(6)3137122

www.utp.edu.co

Versión web disponible en: *http://recursosbiblioteca.utp.edu.co*

Agradecimientos

A Dios por guiarme en el camino, a mis padres: Elizabeth Aguirre Rivera y William Orlando Franco Cuervo, que gracias a su crianza, esfuerzo, apoyo y fe en mí pudimos lograr esta meta. A mi padre Oscar Eduardo Sosa por su apoyo y consejos. A mi hermano Juan Camilo Franco Aguirre por su apoyo incondicional.

A Mariana Tirado Vargas por su apoyo, amor y compañía.

A mi director de grado el Ingeniero Jesser James Marulanda por su guía, exigencia y aportes.

A mis compañeros de carrera, amigos y familia que me han apoyado y guiado en este proceso, y a la Universidad Tecnológica de Pereira por su formación académica y profesional.

Resumen

Las redes neuronales en las últimas décadas se han desarrollado ampliamente gracias a las diferentes campos de aplicación en que se encuentran y a los avances tecnológicos que permiten aplicarlas y reducir los tiempos de computo. Convirtiéndose en una herramienta potente para el modelamiento de cargas en el sistema eléctrico que presentan un comportamiento no lineal y estocástico ya que permite encontrar un modelo matemático de manera efectiva con el fin de poder realizar diferentes estudios y aplicaciones de ingeniería en estos sistemas de potencia eléctrica.

Para ello se hace fundamental definir la metodología, el tipo de red y la validación del método. En este trabajo después de realizar la investigación pertinente se decide combinar dos metodologías de dos artículos citados en la bibliografía para el modelamiento de un horno de arco eléctrico debido a la facilidad de implementación y los buenos resultados que presentan. El método consiste en tomar una muestra de datos de corriente y tensión eléctrica de una fase del horno eléctrico, posteriormente estos datos se normalizan y se definen en el modelo como entrada y salida respectivamente. Una vez que se tienen estos datos, se implementa la metodología basada en la red neuronal de funciones de base radial RBFNN del inglés (Radial Basis Function Neural Networks) la cual consiste en dividir en un 50 por ciento los datos para entrenamiento(training) de la red neuronal y el otro 50 por ciento de los datos para validación(recalling process). De esta manera se logra obtener el modelo matemático que caracteriza el comportamiento V-I del horno de arco eléctrico para un intervalo de tiempo medido.

Tabla de Contenido

1. Introducción	1
1.1. Planteamiento del problema	2
1.2. Justificación	3
1.3. Objetivos	3
1.3.1. General	3
1.3.2. Especificos	3
1.4. Estado del arte	4
1.5. Alcance	5
1.6. Estructura del trabajo de grado	6
2. Marco teórico	7
2.1. Horno de arco eléctrico	7
2.2. Redes Neuronales Artificiales	8
2.3. Red Neuronal Artificial de función de base radial	11
2.3.1. Obtención de los parámetros del modelo utilizando el método del gradiente descendiente	13
2.4. Método LUT	15
2.5. Algoritmo para la obtención de los parámetros iniciales	17
3. Metodología implementada	19
4. Resultados	21
5. Conclusiones	33

Capítulo 1

Introducción

Los seres humanos son capaces de resolver diferentes situaciones o problemas aplicando el conocimiento adquirido por medio de la experiencia. Este mismo principio se lleva a las redes neuronales artificiales, de tal forma que mediante datos históricos, la red es capaz de resolver problemas de acuerdo con la información previamente adquirida y también poder tomar decisiones para resolver nuevos tipos de problemas que pertenecen a la misma clase pero que no estaban explícitamente definidos en los datos históricos [1]. Es por este motivo que se convierte en una herramienta muy útil para modelar cargas no lineales que son variantes en el tiempo en los sistemas eléctricos [2], debido a la naturaleza de la carga no se puede encontrar un modelo circuital RLC o algún método lineal para modelar, siendo las redes neuronales artificiales una alternativa eficiente que permite lograr este propósito.

Una red neuronal artificial y la de interés de este trabajo de grado, la red neuronal de funciones de base radial (RBFNN) por sus siglas en inglés “Radial Basis Function Neuronal Network” de manera general se puede considerar como una “caja negra” que presenta unos datos de entrada, un procesamiento y datos de salida, el procesamiento es donde se encuentran las capas ocultas y en cada capa oculta se encuentran las “neuronas” y cada una de estas tiene su función que relaciona la entrada con la salida, el modelado consiste en encontrar los valores de los parámetros de las funciones que relacionan las entradas con las salidas, a partir de optimizar el error, que se define como la diferencia entre la salida deseada y la salida obtenida. [3].

1.1. Planteamiento del problema

Las redes neuronales artificiales se basan en el comportamiento de interacción y de aprendizaje que presenta el cerebro humano para la solución de problemas. Se utilizan para resolver aquellos problemas que no presentan solución computacional precisa o que requieren de estudios y/o algoritmos muy extensos para su solución.

En la actualidad se presentan investigaciones de aplicaciones de las redes neuronales para una gran variedad de campos del conocimiento entre estos se encuentran: ingeniería, medicina, sistemas, economía entre otros. Estas aplicaciones se pueden agrupar en subcategorías de investigación, algunas de estas son: Optimización, Predicción, Modelamiento y Procesamiento de información (datos, imágenes, etc.).

En el modelamiento se puede encontrar temáticas como la caracterización de la dinámica en la variabilidad cardiaca. En el procesamiento de información se encuentran investigaciones por ejemplo: detección de tumores cancerígenos utilizando imágenes, clasificación de determinados objetos(imágenes, sonidos, señales, etc.), reconocimiento de patrones entre otros. En la predicción se pueden encontrar temas por ejemplo: predicción de enfermedades degenerativas cardiacas, tendencias a corto y mediano plazo en bolsas de valores, y en la optimización temas de investigación como encontrar el punto óptimo de operación de un sistema eléctrico.

En la subárea de investigación de interés que es modelamiento, se encuentra una amplia gama de aplicaciones, esto es debido a que las redes neuronales no se limitan a un campo de aplicación del conocimiento. En específico de esta subárea de investigación, se analizan las aplicaciones de redes neuronales artificiales para modelar cargas no lineales en los sistemas eléctricos, siendo éste el nicho de investigación.

En el nicho de investigación y como finalidad del proyecto de grado, es encontrar el modelo matemático para un horno de arco eléctrico que presenta una alta no linealidad y es variante en el tiempo, por lo que los métodos tradicionales para encontrar el modelo no le aplican. Para ello se construirá la red neuronal artificial a través de un algoritmo computacional, basándose en algunos artículos donde implementan unas variaciones a las redes neuronales tradicionales y logran encontrar el modelo matemático para una carga no lineal.

1.2. Justificación

- La realización del proyecto de grado permitirá encontrar el modelo matemático para un horno de arco eléctrico con alta no linealidad, de esta manera se fortalece el modelamiento en cargas no lineales y abre paso a futuras investigaciones para mejorar o generar nuevas metodologías utilizando redes neuronales para este tipo de aplicación.
- Con el desarrollo del proyecto de grado se abre la posibilidad de investigar las distorsiones en la formas de onda (tensión y corriente eléctrica) del horno de arco eléctrico, fluctuaciones en la tensión eléctrica, el consumo de potencia y analizar las acciones correspondientes para optimizar su funcionamiento.
- La realización del proyecto de grado afianza el conocimiento de programación en Matlab ya que permite crear el algoritmo computacional para encontrar el modelo matemático del horno. También permite ser un punto de partida para realizar futuras investigaciones con el fin de desarrollar otros algoritmos más eficientes computacionalmente para optimizar el proceso ya sea en el mismo lenguaje de programación o en otro.
- Con la realización del trabajo de grado permite encontrar el modelo matemático del horno de arco eléctrico y esto brinda la posibilidad de realizar investigaciones para desarrollar un sistema de control y optimizar la operación de la carga.

1.3. Objetivos

1.3.1. General

Modelar un horno de arco eléctrico utilizando una metodología basada en redes neuronales artificiales para un periodo de tiempo medido.

1.3.2. Especificos

- Investigar y recolectar información acerca de las aplicaciones de las redes neuronales para modelar cargas no lineales que son variantes en el tiempo.
- Obtener las ecuaciones de recurrencia y la metodología de la red neuronal para obtener el modelo matemático.

- Investigar y obtener información sobre programación en Matlab.
- Crear el algoritmo computacional en Matlab para obtener el modelo matemático.
- Evaluar y validar los resultados obtenidos con los datos de muestra y documentar las conclusiones al respecto.

1.4. Estado del arte

Un modelo se utiliza como herramienta para el diseño de un proceso o para el diseño de su respectivo sistema de control. Existen 3 aspectos fundamentales que deben tenerse en cuenta a la hora de obtener un modelo:

- La formulación del modelo.
- La estimación de los parámetros.
- La validación del modelo.

Para el trabajo a realizar concierne la estimación de los parámetros alusivos al modelado utilizando una metodología basada en una red neuronal, para ello se analizará con sumo énfasis en la metodología expuesta en [2], debido a que describe una metodología que permite modelar cargas no lineales utilizando una red neuronal tradicional junto con el metodo denominado “Look Up Table” que le permite a la red neuronal artificial “grabar” el comportamiento dinamico que presenta la carga no lineal en un intervalo de tiempo medido. Sin embargo para tener una visión global de las redes neuronales, en [3],[4] y [1], explican de manera general y detallada, la definicion y funcionamiento de las redes neuronales basicas, así como sus características de aprendizaje y aplicaciones generales.

En este trabajo de grado su proposito es analizar aplicaciones de las redes neuronales artificiales para el modelado de cargas no lineales, por ello se estudia la red neuronal de función de base radial, debido a su facil interpretacion y aplicación, ademas de los resultados exitosos que ha presentado para modelar cargas no lineales variantes en el tiempo. En [5] se muestra la aplicación de una red neuronal de función de base radial para modelar una carga no lineal.

La convergencia de la red consiste en encontrar la relación entrada y salida de la carga no lineal minimizando el error que es la diferencia entre la salida deseada y la obtenida, utilizando funciones de base radial es decir; aquellas funciones que son funciones únicamente de la distancia, normalmente de la norma euclidiana, y dado que la convergencia del error se traduce en encontrar la convergencia de la red, en los artículos [6] y [7], cada uno presenta una metodología que mejora la red neuronal tradicional RBFNN para así minimizar el error y de esta manera encontrar la convergencia de la red.

Finalmente en los artículos [8] y [9] describen metodologías utilizando redes neuronales para modelar cargas no lineales variantes en el tiempo que servirán como complemento para seleccionar la metodología más acorde con el fin de cumplir el propósito de este trabajo de grado.

1.5. Alcance

Este trabajo busca documentar el procedimiento para encontrar el modelo matemático de una carga no lineal y estocástica del sistema eléctrico, en este caso un horno de arco eléctrico, basado en las metodologías propuestas de redes neuronales que se encuentran citados en la bibliografía, con esto se definirá la metodología a implementar, el concepto general de una red neuronal como también se define de manera general la red neuronal RBFNN, el método LUT y la ventaja de optimizar el error medio cuadrático para encontrar los mejores resultados, se muestran las ecuaciones de recurrencia que se utilizan para modelar y su respectiva demostración. Se presentan los resultados y sus conclusiones pertinentes.

También se busca que sea un documento en español para ser base para futuras aplicaciones y/o mejora de investigaciones para el modelamiento de cargas no lineales y estocásticas, como investigaciones de estudios de ingeniería para optimizar el rendimiento de su uso en el sistema eléctrico.

1.6. Estructura del trabajo de grado

El documento está organizado de la siguiente forma. En el capítulo 1 se presenta la introducción, planteamiento del problema, justificación con sus respectivos objetivos, general y específicos, estado del arte y el alcance de la investigación. Posteriormente, el capítulo 2 contiene el marco teórico que explicará las redes neuronales, la red RBFNN, las ecuaciones de recurrencia, la metodología LUT y la metodología para la asignación de las variables iniciales. El capítulo 3 es donde se describe la metodología implementada con la cual se construye el algoritmo para obtener y documentar los resultados correspondientes entre los valores estimados y medidos. En el capítulo 4 se muestran los resultados obtenidos por el algoritmo desarrollado, y por último. En el capítulo 5 se concluye sobre los resultados de la implementación de esta metodología para obtener el modelo matemático de este tipo de carga en el sistema eléctrico.

Capítulo 2

Marco teórico

2.1. Horno de arco eléctrico

El horno de arco eléctrico es ampliamente usado en la industria para la producción de acero, su uso en el sistema eléctrico representa un alto consumo de potencia activa y reactiva, generando distorsiones en la forma de onda de la tensión eléctrica e inyectando reactivos a la red [9]. Esto es debido a que su comportamiento en su característica V-I presenta un comportamiento no lineal y variante en el tiempo como se muestra en la siguiente figura.

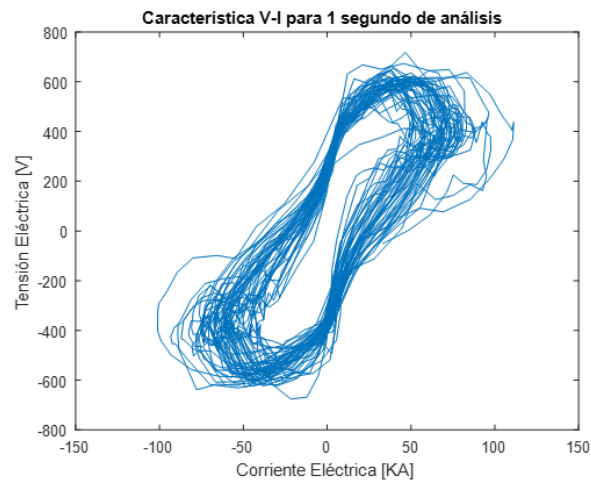


Figura 2.1: Característica V-I para 1 segundo de análisis.

La anterior figura corresponde a la medición de corriente y tensión eléctrica para un periodo de un segundo de tiempo de una de las fases del horno de arco eléctrico. Como se observa se presenta una alta no linealidad a su vez que tiene un comportamiento estocástico. Es por las anteriores razones se hace necesario obtener un modelo matemático que permita estimar su comportamiento lo más fiel posible, diferentes metodologías se han presentado para obtener el modelo, sin embargo las redes neuronales han sido una herramienta eficaz ya que permite modelar de manera más general [5] y no desde las características físicas del horno. Una vez que se obtiene el modelo matemático, se permite realizar las investigaciones pertinentes para aplicar las soluciones para corregir y/o mitigar las distorsiones en la formas de ondas y la inyección de reactivos.

2.2. Redes Neuronales Artificiales

El primer modelo simple de la neurona fue propuesto en 1943 por los investigadores Warren McCulloch y Walter Pitts, posteriormente en la década de los cincuenta y setenta B. Windrow y M. E Hoof lideraron las investigaciones. Ambos trabajaron en una maquina denominada Adaline de sus siglas en inglés (Adaptive Linear Element). Otro investigador pionero fue el psicólogo Frank Rosenblatt, que en 1959 construyó una maquina neuronal simple la cual llamó perceptrón, la cual tenía una matriz con 400 fotoceldas las cuales se conectaban de manera aleatoria a 512 unidades tipo neurona, su funcionamiento consistía en que cuando se representaba un patrón a las diferentes unidades sensoras, estas respondían enviando una señal hacia el banco de neuronas de tal forma que se indicaba la categoría del patrón. Logrando poder reconocer todas las letras del alfabeto. Sin embargo al final de la década de los 70 se demostró por Minsky y Papert que los perceptrones no eran capaces de resolver problemas simples como poder simular la función lógica XOR. Otros investigadores que se reconocen por trabajar en la década de los 70 fueron Teuvo Kohonen, Jim Anderson que trabajo con alternativas de semillas de coexionismo y Geoff Hinton que presento diferentes trabajos matemáticos y aplicación de redes neuronales [1].

En los siguientes años Stephen Grossberg y Gail Carpenter han propuesto un modelo de red neuronal llamado ART de las siglas en inglés (Adaptive Resonance Theory), y en 1986 Mc Clelland y Rumelhart publicaron un libro en dos volúmenes que se considera como el clásico en el área de las redes neuronales denominado Parallel Distributed Processing: Explorations in the Microstructure of Cognition, permitió ser el nuevo impulso

a la investigación en sistemas neuronales ya que mostro las ventajas y desventajas que presentan estos sistemas [1].

Las redes neuronales se basan en el proceso de solución que presenta el cerebro para resolver problemas. De la misma manera en que los seres humanos aplican el conocimiento adquirido por la experiencia para resolver nuevos problemas o situaciones, la red neuronal artificial se entrena con diferentes problemas y sus respectivas soluciones para construir un sistema capaz de dar una respuesta frente a una nueva problemática no contemplada con anterioridad. Se implementan en aquellos problemas que no tienen solución computacional precisa o que requieren de algoritmos bastantes extensos [1].

Analizando matemáticamente la red neuronal artificial esta se puede representar como una “caja negra” como se representa en la figura 2.2 [2], como observa en la figura se encuentra la entrada (X) que es un vector que contiene las (N) diferentes variables de información denominado la capa de entrada, estos datos de entrada pueden ser de cualquier naturaleza (corriente eléctrica, velocidad, temperatura entre otras). Luego en la capa oculta es donde se encuentra un numero finito de neuronas (L), cabe mencionar que existen redes multicapa, es decir; son redes que presentan varias capas y en cada capa se encuentra un número determinado de neuronas como se observa en la figura 2.3 [1]. Sin embargo este proyecto de grado se enfoca es en el tipo de red monocapa, es decir; redes que solo tienen neuronas en la capa oculta. A cada neurona se le asocia una “función de transferencia” que permite relacionar la respuesta que estas presentan frente a los datos de entrada. Finalmente en la capa de salida, es la respuesta del sistema, es un vector (Y) que contiene las (M) salidas del sistema.

Nótese que la salida se relaciona con las respuestas que entregan las neuronas y unos pesos (W) que indican el grado de “importancia” para una salida (Y_i).

De esta manera una red neuronal se puede representar como una caja negra, ya que permite pasar de un dominio de datos (X) a un dominio de datos (Y) para esto se debe encontrar la convergencia de la red, la cual es encontrar los valores de los pesos (W) [2]. Esto se realiza durante el entrenamiento, que es un algoritmo de actualización que se puede basar en el método del gradiente descendiente, en donde los pesos se actualizan por medio del uso de un conjunto ordenado de entradas y salidas deseadas junto con la comparación entre la salida deseada (medida) con respecto a la salida obtenida de la red neuronal para una entrada medida[1]. Otros métodos que se implementan para el

entrenamiento son el perceptrón, el algoritmo de mínimos cuadrados medios, el método de mínimos cuadrados recursivos entre otros.

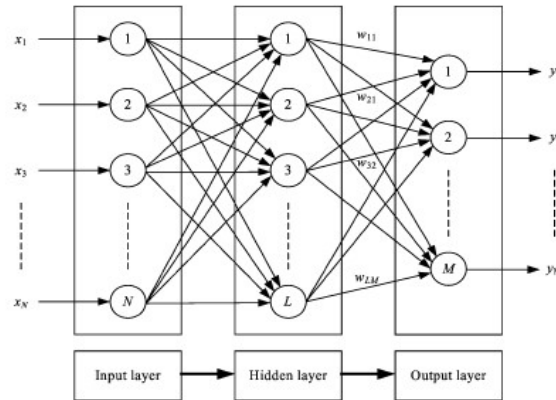


Figura 2.2: Estructura típica de una red neuronal artificial.

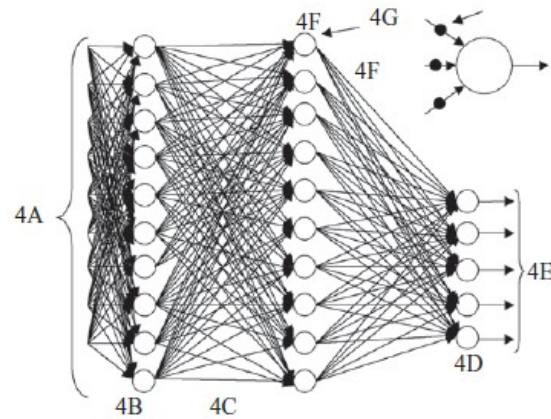


Figura 2.3: Estructura de una red neuronal artificial multicapa.

Como se observa en la figura anterior, se muestra un esquema de red neuronal multicapa, donde (4A) corresponde a la variable de entrada, posteriormente se encuentran las diferentes capas. (4B) capa de entrada donde se encuentran 10 neuronas, (4C) corresponde a la capa oculta donde también hay 10 neuronas, (4D) es la capa de salida y donde se encuentran las últimas 5 neuronas [1].

2.3. Red Neuronal Artificial de función de base radial

Las redes neuronales de función de base radial (RBFNN), de las siglas en inglés (Radial Basis Function Neuronal Network), son una estructura de redes neuronales que se han utilizado ampliamente debido a sus buenos resultados en teoría de aproximación y modelado para encontrar la relación entrada/salida de cargas eléctricas no lineales, ya que por su estructura se permite implementar fácilmente y tener buenos resultados [5], [9]. Su estructura es la de una red monocapa como se ve en la figura 2.2 Sin embargo muchas veces por simplicidad se toma el siguiente modelo como se muestra en la siguiente figura 2.4 [4].

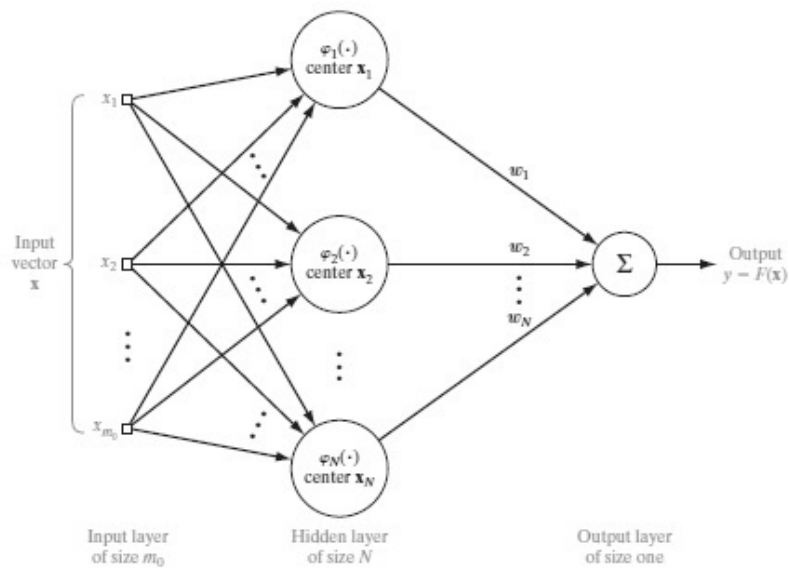


Figura 2.4: Estructura típica de una red RBFNN.

Su funcionamiento es como el que se explicó en la sección anterior, se diferencia es que en la capa de salida, normalmente se tiene una única respuesta, y que la función que presenta cada neurona es de base radial es decir; solamente depende de la distancia, normalmente la distancia es la norma euclidiana, de tal forma que en la capa oculta se utilizan son funciones Gaussianas ya que cumplen con dicha condición. [4].

De la figura 2.4 el modelo se define como se muestra en [4].

$$\Phi_j(X) = \Phi\|X - X_j\| \quad j = 1, 2, \dots, N \quad (2.1)$$

Donde:

j : Corresponde a la j -ésima neurona de las (N) neuronas que se encuentran en la capa oculta.

X : Corresponde al vector de entrada, desde X_1 hasta X_m .

X_j : Es el centro de cada neurona.

$\Phi_j(X)$: Es la respuesta de la función de base radial de cada neurona.

La función Gaussiana que se usa como función de base radial para cada neurona en la capa oculta es la siguiente:

$$\Phi_j(X) = \Phi\|X - X_j\| = \exp \frac{-\|X - X_j\|^2}{\sigma_j^2} \quad j = 1, 2, \dots, N \quad (2.2)$$

σ_j : Es la desviación estándar de cada neurona que indica la posición relativa del centro de la misma con respecto a los datos de entrada.

La combinación entre los pesos (W) y las respuestas de las neuronas en la capa oculta, como se muestra en la figura 2.4 permite encontrar la salida y de esta manera se puede encontrar la relación entrada/salida del modelo.

$$y = \sum_{j=1}^N W_j \cdot \Phi_j(X) \quad (2.3)$$

y : Corresponde a la salida estimada del modelo.

W_j : Corresponde al peso de cada neurona.

Para obtener el modelo se deben encontrar los parámetros del mismo que son (centros, desviaciones y los pesos) Distintos algoritmos se pueden implementar para encontrar estos parámetros y mejorar el rendimiento de la red [4], entre estos están el método de mínimos cuadrados, el gradiente descendiente, el algoritmo de mínimos cuadrados ortogonales [7] entre otros. En la siguiente subsección se demuestran la obtención de los parámetros utilizando el método del gradiente descendiente.

2.3.1. Obtención de los parámetros del modelo utilizando el método del gradiente descendiente

Primero se debe definir la diferencia $e(p)$, entre la respuesta deseada y la estimada para la iteración p -ésima.

$$e(p) = d(p) - y(p) \quad (2.4)$$

Donde:

$d(p)$: Es la respuesta deseada (medida) para la iteración p -ésima.

$y(p)$: Es la respuesta estimada para la iteración p -ésima.

La función de error a optimizar se define conveniente como en [2]. De esta manera se tiene una función cóncava por lo que permite encontrar una solución óptima para el error.

$$E(p) = \frac{1}{2} \cdot e(p)^2 \quad (2.5)$$

Finalmente organizando las ecuaciones 2.2, 2.3, 2.4 y 2.5 se obtiene.

$$E(p) = \frac{1}{2} \cdot \left(d(p) - \sum_{j=1}^N W_j(p) \cdot \exp \frac{-\|X(p) - X_j(p)\|^2}{\sigma_j(p)^2} \right)^2 \quad (2.6)$$

La ecuación anterior es la función de error que se extiende para cada iteración y a su vez para cada neurona. Los parámetros que no se conocen son los pesos, los centros y las desviaciones estándar, por lo que para obtenerlos se implementa el método del gradiente descendiente.

Las ecuaciones que se obtienen después de implementar el método gradiente descendiente son:

$$W_j(p+1) = W_j(p) - \lambda \cdot \frac{d(E(p))}{d(W_j(p))} \quad (2.7)$$

$$X_j(p+1) = X_j(p) - \lambda \cdot \frac{d(E(p))}{d(X_j(p))} \quad (2.8)$$

$$\sigma_j(p+1) = \sigma_j(p) - \lambda \cdot \frac{d(E(p))}{d(\sigma_j(p))} \quad (2.9)$$

Donde:

λ : Corresponde al paso de búsqueda para la siguiente iteración, se puede implementar un subproceso de optimización para encontrar el mejor valor, debido a la complejidad de la ecuación este valor se considera una constante.

Las ecuaciones de recurrencia 2.6, 2.7 y 2.8 se resuelven encontrando las derivadas parciales para la función del error. Estas son:

$$\frac{d(E(p))}{d(W_j(p))} = -e(p) \cdot \exp \frac{-\|X(p) - X_j(p)\|^2}{\sigma_j(p)^2} \quad (2.10)$$

$$\frac{d(E(p))}{d(C_j(p))} = -2 \cdot e(p) \cdot W_j(p) \cdot \Phi \|X(p) - X_j(p)\| \cdot \frac{(X(p) - X_j(p))}{\sigma_j(p)^2} \quad \text{Si } X(p) \neq X_j(p) \quad (2.11)$$

$$\frac{d(E(p))}{d(\sigma_j(p))} = -2 \cdot e(p) \cdot W_j(p) \cdot \Phi \|X(p) - X_j(p)\| \cdot \frac{\|X(p) - X_j(p)\|^2}{\sigma_j(p)^3} \quad \text{si } \sigma_j(p) \neq 0 \quad (2.12)$$

Finalmente las ecuaciones de recurrencia con las que se permite encontrar los parámetros para así obtener el modelo se definen a continuación:

$$W_j(p+1) = W_j(p) + \lambda \cdot e(p) \cdot \exp \frac{-\|X(p) - X_j(p)\|^2}{\sigma_j(p)^2} \quad (2.13)$$

$$X_j(p+1) = X_j(p) + 2 \cdot \lambda \cdot e(p) \cdot W_j(p) \cdot \Phi \|X(p) - X_j(p)\| \cdot \frac{(X(p) - X_j(p))}{\sigma_j(p)^2} \quad \text{Si } X(p) \neq X_j(p) \quad (2.14)$$

$$\sigma_j(p+1) = \sigma_j(p) + 2 \cdot \lambda \cdot e(p) \cdot W_j(p) \cdot \Phi \|X(p) - X_j(p)\| \cdot \frac{\|X(p) - X_j(p)\|^2}{\sigma_j(p)^3} \quad \text{si } \sigma_j(p) \neq 0 \quad (2.15)$$

Los parámetros iniciales (pesos, centros y desviaciones) estos se pueden definir de manera aleatoria al igual que el paso de búsqueda, sin embargo la convergencia de la red depende bastante del punto inicial (parámetros iniciales) ya que converge rápidamente alrededor de este punto, en [9] definen un algoritmo para obtener los parámetros iniciales con el objetivo de que el error medio cuadrático sea mínimo. Esta metodología se explicará en una sección posterior de este documento.

2.4. Método LUT

Las redes neuronales artificiales presentan buenos resultados para el modelado de cargas no lineales, sin embargo estas presentan cierto inconveniente cuando se tienen comportamientos estocásticos ya que al definir su “función de transferencia” esta es fija y no es capaz de seguir el comportamiento de esta naturaleza. Por ello se implementa la metodología LUT expuesta en [2]. Para este tipo de aplicación primero se redefine el modelo mostrado en la figura 2.5 por el siguiente[2]:

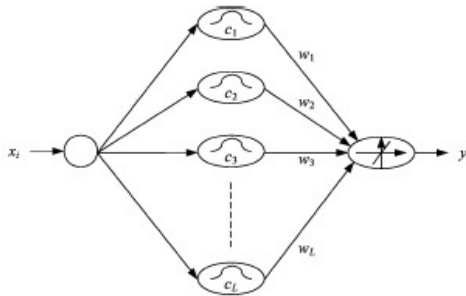


Figura 2.5: Modelo reestructurado de RBFNN.

De esta manera en vez de tener un modelo con múltiples entradas, a cada entrada de ese vector se le trata por separado.

La metodología LUT consiste en almacenar los parámetros (pesos, centros y desviaciones estándar) durante el proceso de aprendizaje.

Para cada una de las muestras del vector de entrada (X) como se muestra en la figura 2.3, utilizando el modelo redefinido en la figura 2.5 y utilizando las ecuaciones 2.13, 2.14 y 2.15 se puede construir 3 matrices una para los pesos, centros y desviaciones estándar, del tamaño MXN . Este proceso puede tomar tiempo de computo en la construcción de la LUT ya que depende del tamaño del vector de entrada y del número de neuronas, sin embargo cuando se complementa con el recalling process es computacionalmente eficiente debido a que se aplica directamente con los valores de las matrices para construir las funciones de transferencia instantáneas para cada muestra y así poder superar el inconveniente de función de transferencia fija [2].

El proceso de recalling process, se define con la siguiente figura [2]:

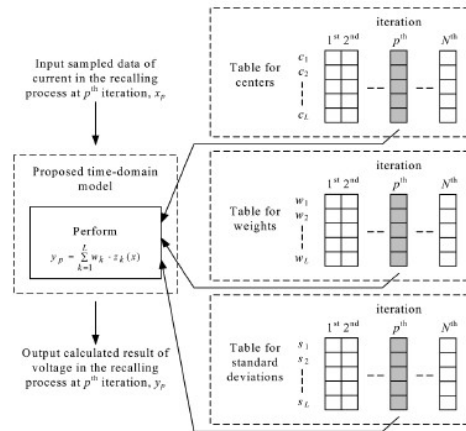


Figura 2.6: Recalling process con LUT.

Una vez que se tiene la matriz LUT construida, para cada muestra(iteración) se “llama” su correspondiente valor de peso, centro y desviación para cada una de las neuronas, al final con la sumatoria se obtiene la salida estimada para esa muestra (iteración).

2.5. Algoritmo para la obtención de los parámetros iniciales

Tener un buen punto de partida es decir; determinar los pesos, centros y desviaciones iniciales, permite aumentar la exactitud del modelo a costo de aumentar el tiempo de solución [9]. En vez de utilizar valores aleatorios para los parámetros iniciales, se utiliza el algoritmo descrito en [9] que permite no solo encontrar los parámetros iniciales sino también conducir la solución donde el error medio cuadrático sea mínimo. El algoritmo es el siguiente:

1. Se inicia el conteo de la iteración j , y se establece su valor máximo de iteraciones (J_{max}).
2. Se encuentra el mínimo (X_{min}) y máximo (X_{max}) de los valores de entrada del vector (X), que va desde (X_1) hasta (X_N) que son los respectivos datos medidos de entrada. Luego la suma de los valores absolutos de (X_{min}) y (X_{max}) se le denomina (X_{abs}). Por último, se define $(-10X_{abs}, 10X_{abs})$ como el rango permisible para los centros de todas las funciones de base radial de la ecuación 2.2.
3. De manera similar se encuentra el mínimo (Y_{min}) y máximo (Y_{max}) de los valores de salida del vector (Y), que va desde (Y_1) hasta (Y_M) que corresponden a los respectivos datos medidos de salida de entrenamiento. Luego la suma de los valores absolutos de (Y_{min}) y (Y_{max}) se le denomina (Y_{abs}). Por último, se define $(-0.01Y_{abs}, 0.01Y_{abs})$ como el rango permisible para los pesos de la ecuación 2.3.
4. Se divide cada rango de los centros y de los pesos por la cantidad de neuronas en la capa oculta (L), luego se asignan de manera equidistante de tal forma que entre un valor a otro, tengan el mismo delta de separación, así se tiene (L) valores como parámetros iniciales para los centros y los pesos de las (L) funciones de base radial.
5. Se define $(0.1X_{abs})$ como valor inicial para todas las desviaciones estándar de todas las neuronas de la capa oculta.
6. Luego se implementan las ecuaciones desde la 2.13 hasta la 2.15, junto con el método LUT y el recalling process (ver figura 2.6) para todos los datos de entrada (Vector de entrada) y así obtener los correspondientes datos de salida (Vector de salida). De esta manera con el vector de salida poder obtener el valor del error medio cuadrado (RMSE), de las siglas en ingles "Root Mean Square Error".

$$RMSE_j = \sqrt{\frac{\sum_{i=1}^N (d_i - y_i)^2}{N}} \quad (2.16)$$

Donde:

N : Es el valor del tamaño del vector de salida.

d_i : Es el valor de la i -ésima salida deseada (medida).

y_i : Es el valor de la i -ésima salida estimada.

7. Actualizar la desviación estándar para todas las neuronas de acuerdo con:

$$\sigma(j+1) = \sigma(j) + \sigma_0 \quad (2.17)$$

Revisar si la iteración j llegó al límite, si lo hizo se procede al siguiente paso, de lo contrario regresar a 6.

8. De las J_{max} iteraciones encontrar cual iteración se tiene el valor mínimo para el error medio cuadrado y con esto identificar los pesos, centros y desviaciones estándar asociados a este valor.

Nótese que con este algoritmo se cumple las condiciones para que sean validas las ecuaciones 2.14 y 2.15.

Capítulo 3

Metodología implementada

En este capítulo se especifica la metodología que se implementa para la creación del algoritmo en Matlab, que es el que permite encontrar el modelo matemático y los resultados para su respectivo análisis y conclusiones. La metodología es la siguiente:

Asignación de datos.

1. Se sigue el modelo definido en la figura 2.5, de esta manera se trata a cada dato medido de corriente y tensión como se define en el método LUT. También se define a (X) como variable de entrada para la corriente eléctrica y (Y) como la variable de salida para la tensión eléctrica.
2. Se define la ventana de tiempo de análisis ya que se tiene dos vectores cada uno con 33000 datos medidos de corriente y tensión eléctrica, el tiempo de muestreo corresponde a 2048 datos por segundo, aproximadamente 16 segundos de medición para la corriente y tensión eléctrica. Luego se obtienen los vectores de corriente y tensión eléctrica delimitados por el intervalo de tiempo de análisis, a su vez, de estos vectores se obtienen los valores máximos absolutos de corriente (X_{base}) y tensión (Y_{base}).
3. Luego se divide los dos vectores (X) y (Y) , creando cada uno dos vectores, de tal manera que se genera un vector (X_{train}) con el 50 por ciento de los datos para el proceso de aprendizaje y el otro vector (X_{val}) con el otro 50 por ciento de los datos para el proceso de validación (recalling process), de la misma se generan los vectores (Y_{train}) y (Y_{val}) correspondiente para los procesos de entrenamiento y validación respectivamente. Para la implementación de las redes neuronales, los

datos deben de estar normalizados, de tal manera que el rango permisible sea entre (-1,1) por lo que se dividen los vectores (X_{train}) y (X_{val}) por (X_{base}) y los vectores (Y_{train}) y (Y_{val}) por (Y_{base}).

Inicialización, entrenamiento y validación.

1. Se especifica el número total de neuronas para la capa oculta ($L=150$), También se especifica el valor del paso de búsqueda (λ) de las ecuaciones 2.13, 2.14 y 2.15. En la literatura a este valor se le conoce como factor de aprendizaje, de tal manera que no se tienen en cuenta el 2 que multiplica a este factor en las ecuaciones 2.14 y 2.15, y pueden tener valores diferentes para cada una de las ecuaciones, sin embargo en [2] este valor es de 0.01 para los 3 factores de aprendizaje. En este trabajo si se tuvo en cuenta el 2 que multiplica al factor de aprendizaje y se tomó de 0.01 para las 3 ecuaciones debido a que se encontraron mejores resultados.
2. Luego se implementa el algoritmo para la obtención de los parámetros iniciales, descrito en el capítulo anterior, para esto se utilizan los vectores (X_{train}) y (Y_{train}) que junto el método LUT se desarrolla el aprendizaje de la red. El número máximo de iteraciones (j) es de 200 por lo que entonces se van a evaluar 200 matrices para los pesos, 200 matrices para las desviaciones estándar y 200 matrices para los centros de las neuronas, estas matrices se validan de acuerdo con el recalling process (ver figura 2.6) con el vector (X_{val}) para así obtener 200 vectores de salida estimados ($Y_{estimado}$), con estos 200 vectores y el vector (Y_{val}) se utilizan para calcular el vector RMSE y posteriormente identificar la iteración en donde se presenta el menor error medio cuadrático. La diferencia que se implementa es que en vez de solamente obtener los parámetros iniciales, se almacena las matrices de los pesos, centros y desviación estándar como también el vector de salida estimada ($Y_{estimado}$) asociado al valor mínimo del error medio cuadrático. De esta manera se obtiene el modelo matemático para un intervalo de tiempo medido.
3. Una vez obtenido el modelo se procede a obtener los resultados y graficas correspondientes para realizar su posterior análisis.

Capítulo 4

Resultados

En este capítulo se exponen los resultados obtenidos una vez que se obtuvo el modelo matemático, se tienen 2 casos de análisis y en cada caso se tienen dos subcasos, cuando se hace la comparación del modelo con los mismos datos de entrenamiento, esto es que el recalling process se realiza con el vector (X_{train}) y cuando se hace la comparación con los datos de validación, cuando el recalling process se realiza con el vector (X_{val}).

1. Intervalo de análisis de (0-1 segundo).

A) Recalling process con los datos de entrenamiento.

De la metodología implementada en el capítulo 3, para la ventana de tiempo establecida, se tiene:

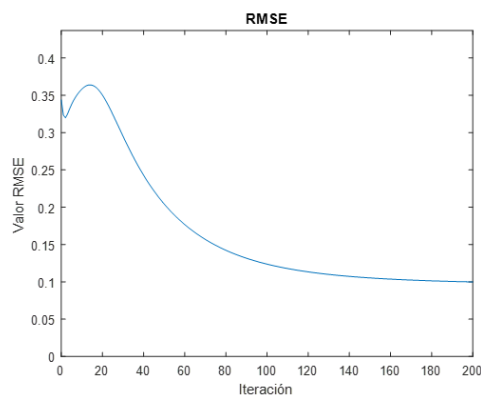


Figura 4.1: RMSE para recalling process con los datos de entrenamiento de (0-0.5s).

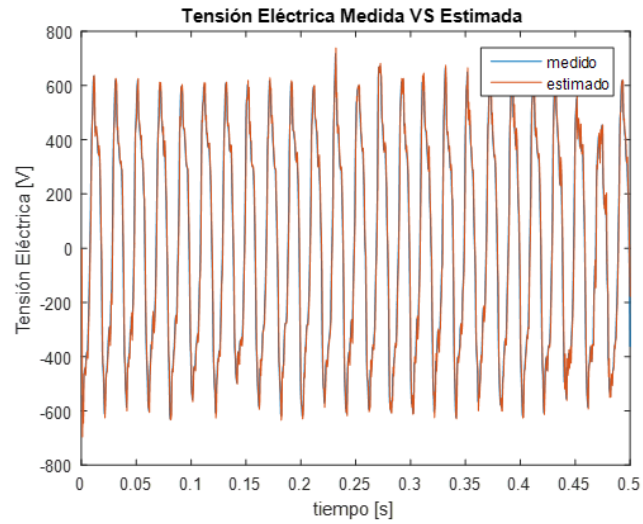


Figura 4.2: Comparación de la tensión eléctrica medida y estimada con los datos de entrenamiento (0-0.5s).

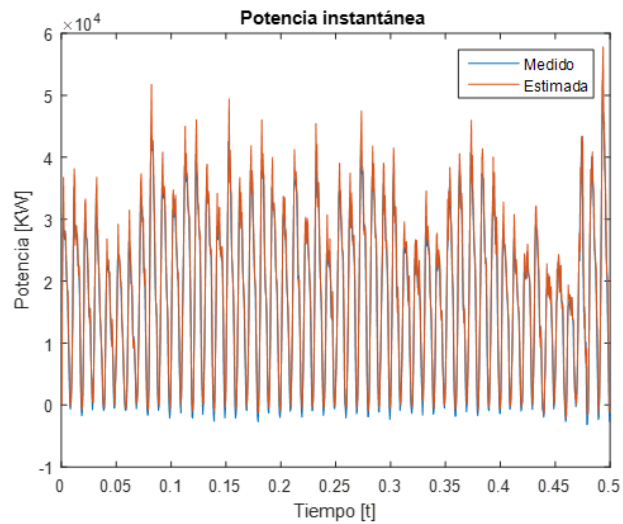


Figura 4.3: Comparación de la potencia instantánea eléctrica medida y estimada con los datos de entrenamiento (0-0.5s).

Como se observa en la figura 4.1, en las primeras iteraciones se tiene un aumento del error medio cuadrático sin embargo, este valor empieza a decaer hasta que tiende a llegar a un límite y luego se obtiene el modelo en donde el error medio cuadrático sea mínimo.

Se puede analizar que la tensión eléctrica estimada de la figura 4.2 es prácticamente la misma que la tensión eléctrica medida para los datos de entrenamiento, es decir que el modelo es capaz de estimar fielmente la forma de onda de la tensión eléctrica cuando se hace el recalling process con los datos de entrenamiento.

De la figura 4.3 se observa que el modelo es capaz de estimar adecuadamente la forma de onda de la potencia eléctrica instantánea para los datos de entrenamiento, sin embargo se observan unos sobre picos en la potencia estimada, más adelante se expondrá el valor de la potencia activa estimada para este periodo de tiempo y así poder comparar con la potencia activa medida y observar la diferencia que estos sobre picos generan.

B) Recalling process con los datos de validación.

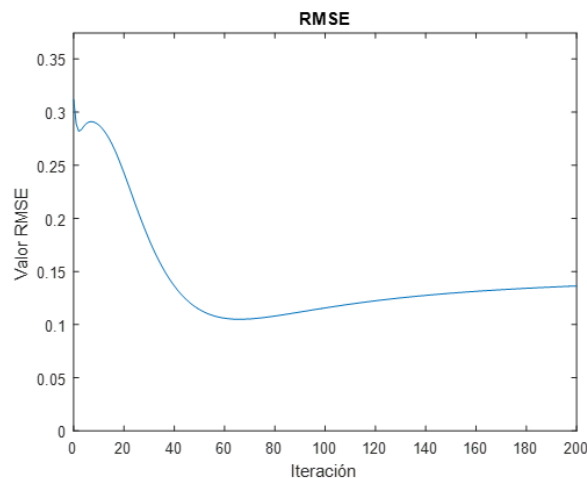


Figura 4.4: RMSE para recalling process con los datos de validación de (0.5-1s).

Se observa que en la figura 4.4 el error en las primeras iteraciones tiende a crecer, sin embargo empieza a disminuir hasta llegar al valor mínimo, posteriormente tiende a crecer de manera no tan significativa.

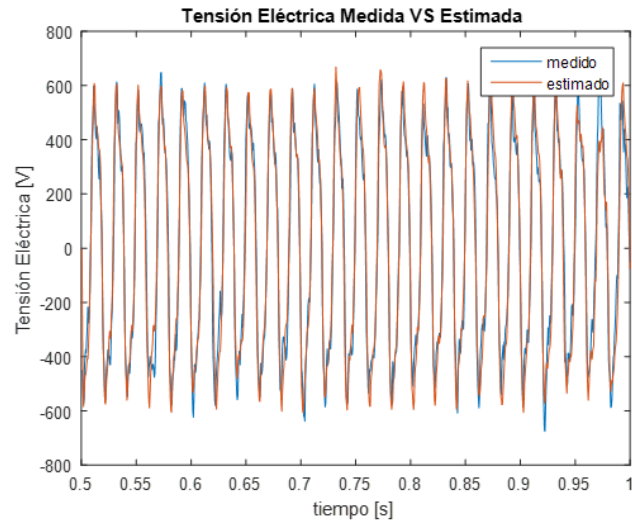


Figura 4.5: Comparación de la tensión eléctrica medida y estimada con los datos de validación (0.5-1s).

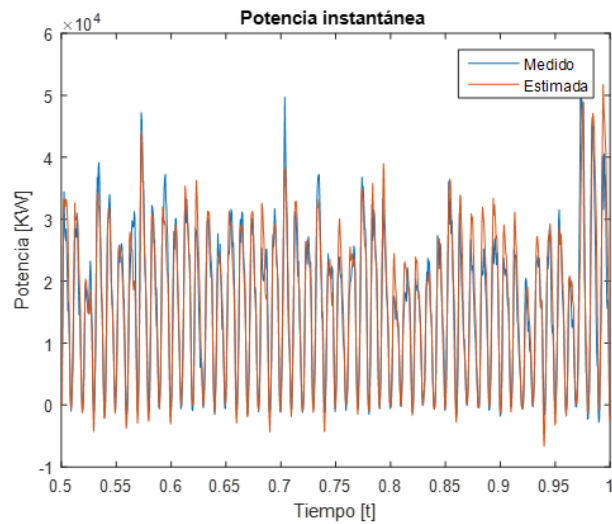


Figura 4.6: Comparación de la potencia instantánea eléctrica medida y estimada con los datos de validación (0.5-1s).

Se observa en la figura 4.5 que ya se encuentran diferencias significativas entre la tensión eléctrica estimada y la tensión eléctrica medida, sobre todo en los picos donde no es capaz de estimar adecuadamente la tensión eléctrica medida, esto se debe a que en los picos es donde se presentan cambios bruscos y el modelo no es capaz de representarlos correctamente. Posteriormente se evaluará el valor RMS de la tensión eléctrica estimada y medida para observar la diferencia producida.

El comportamiento de la figura 4.6 se esperaba ya que si la tensión eléctrica estimada no representaba fielmente a los datos medidos, la potencia instantánea que es el producto de la tensión y la corriente eléctrica tampoco lo haría, nuevamente es en los picos donde se observa mayor diferencia.

A continuación se muestra una comparación de la tensión RMS, potencia activa, y el valor RMSE tanto para el recalling process con los datos de entrenamiento y validación.

Tabla 4.1: Error medio cuadrático (RMSE) mínimo.

Recalling process	RMSE
Entrenamiento	0.0999
Validación	0.1051

Tabla 4.2: Comparación de la tensión RMS estimada y medida.

Recalling process	V(RMS) medido(V)	V(RMS) estimado(V)	Error relativo(%)
Entrenamiento	401.4656	403.4845	0.503
Validación	384.4564	394.3493	2.57

Tabla 4.3: Comparación de la potencia activa estimada y medida.

Recalling process	P(activa) medida (KW)	P(activa) estimada (KW)	Error relativo(%)
Entrenamiento	17130	18243	6.5
Validación	15798	16211	2,614

El error relativo se define como:

$$Er(\%) = \frac{|Variable(medida) - Variable(estimada)|}{Variable(medida)} \cdot 100 \quad (4.1)$$

Como se esperaba la tensión RMS para el recalling process con los datos de entrenamiento presenta un error relativo mucho menor que cuando se realiza el recalling process con los datos de validación, esto es debido a la mejor estimación del modelo con los datos de entrenamiento ya que a la red se le pide una estimación con los datos previamente entrenados, mientras que con los datos de validación a la red se le pide una respuesta frente a unos nuevos datos que son similares a los de entrenamiento pero no son iguales. Aun así se obtiene una apropiada estimación ya que el error relativo es menor al 3%.

De la tabla 4.3 se observa que se presenta un error relativo mayor en el recalling process con los datos de entrenamiento que con los datos de validación. Esto se debe a que en la figura 4.3 la potencia instantánea estimada, siempre se presentan sobre picos mientras que en la figura 4.6 la potencia instantánea estimada en algunos puntos de la forma de onda se presentan sobre picos y en otros sub picos con respecto a la potencia instantánea medida, entonces la potencia activa al ser el promedio de la potencia estimada se presenta un menor error ya que en promedio se equilibra mostrando un valor similar al de la potencia medida en el recalling process con los datos de validación.

2. Intervalo de análisis de (4-5 segundo).

A) Recalling process con los datos de entrenamiento.

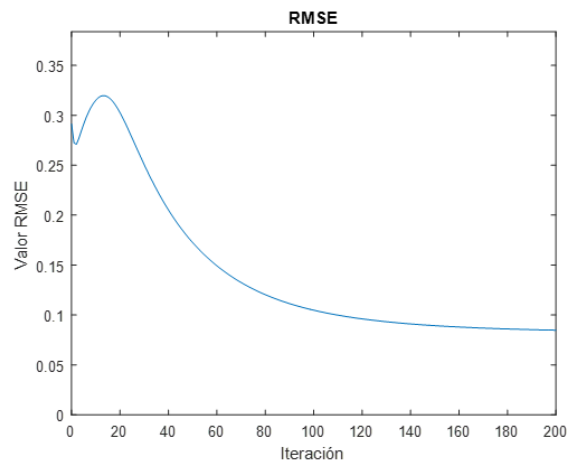


Figura 4.7: RMSE para recalling process con los datos de entrenamiento (4-4.5s).

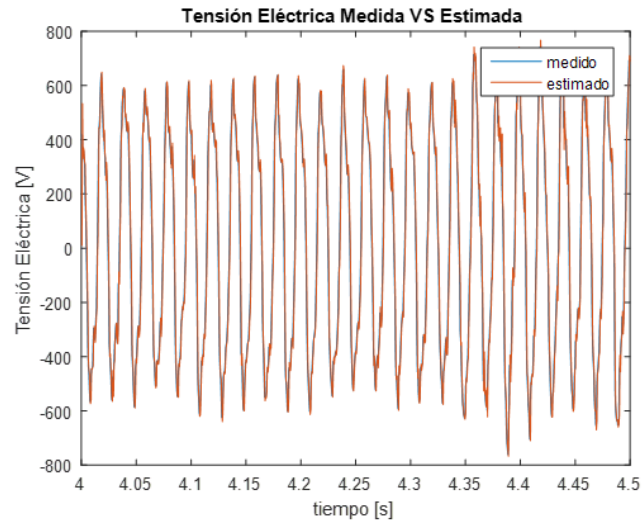


Figura 4.8: Comparación de la tensión eléctrica medida y estimada con los datos de entrenamiento (4-4.5s).

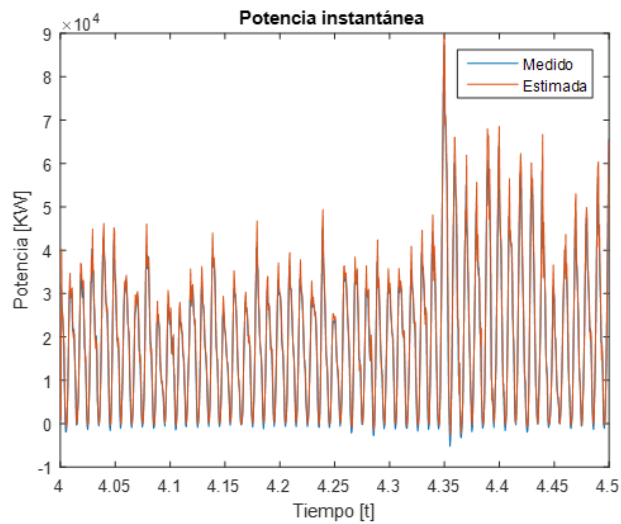


Figura 4.9: Comparación de la potencia instantánea eléctrica medida y estimada con los datos de entrenamiento (4-4.5s).

Las figuras 4.7, 4.8 y 4.9 presentan un comportamiento similar al de las figuras 4.1, 4.2 y 4.3 respectivamente.

Al analizar la figura 4.7 se observa que el error medio cuadrático nuevamente tiende a crecer para las primeras iteraciones y luego decrece hasta llegar un límite.

En la figura 4.8 la tensión eléctrica estimada nuevamente presenta una adecuada estimación de la tensión eléctrica medida puesto que las graficas a simple vista parecen ser la misma y la figura 4.9 la potencia eléctrica instantánea presenta una buena estimación sin embargo se presentan nuevamente sobre picos en su forma de onda. Aunque se tiene un intervalo de análisis diferente el comportamiento de las graficas es similar cuando se realiza el recalling process con los datos de entrenamiento.

B) Recalling process con los datos de validación.

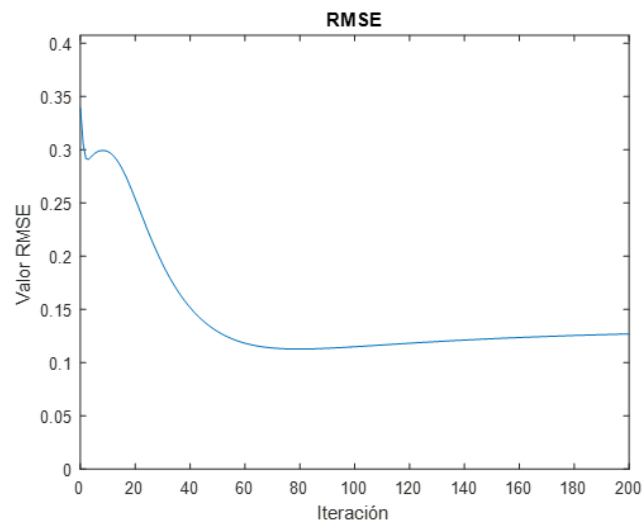


Figura 4.10: RMSE para recalling process con los datos de validación (4.5-5s).

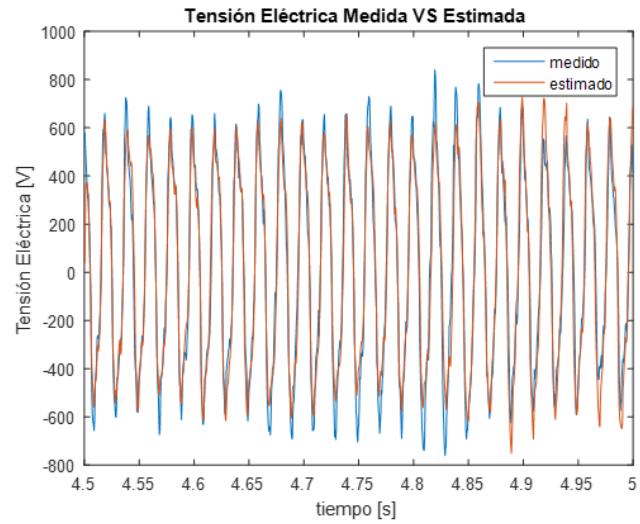


Figura 4.11: Comparación de la tensión eléctrica medida y estimada con los datos de validación (4.5-5s).

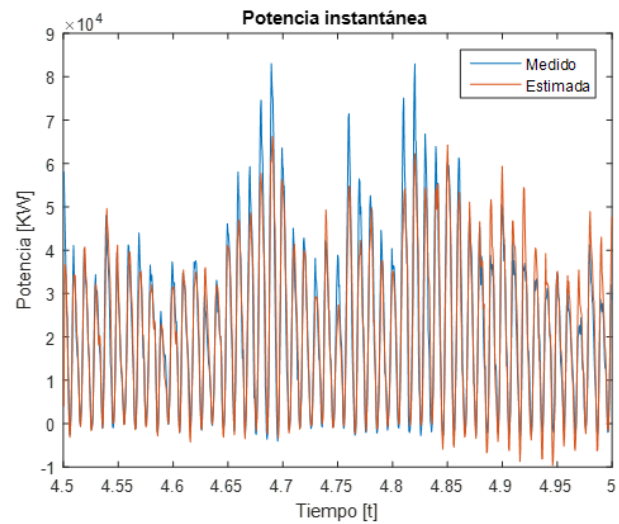


Figura 4.12: Comparación de la potencia instantánea eléctrica medida y estimada con los datos de validación (4.5-5s).

Las figuras 4.10, 4.11 y 4.12 también presentan un comportamiento similar a las figuras 4.4, 4.5 y 4.6 respectivamente, que corresponden al recalling process con los datos de validación. En la figura 4.10 se observa que de manera similar el error medio cuadrático (RMSE) decae, luego tiende a aumentar levemente y decrece hasta llegar a su valor mínimo, finalmente tiene un crecimiento poco significativo hasta que tiende a llegar un valor límite. De la figura 4.11 se observa nuevamente que la tensión eléctrica estimada presenta diferencias con la tensión eléctrica medida, de igual manera que en la figura 4.5 es en los picos de la forma de onda, donde debido a los cambios bruscos el modelo presenta inconvenientes para la estimación y en la figura 4.12 la forma de onda de la potencia eléctrica instantánea como en la figura 4.6 no se presenta una estimación exacta y esto se debe a que si la tensión eléctrica estimada presenta inconvenientes, la potencia eléctrica instantánea también los presenta.

En las siguientes tablas se expone los valores correspondientes de la tensión RMS, potencia activa, y el valor RMSE tanto para el recalling process con los datos de entrenamiento y validación en este intervalo de tiempo de análisis.

Tabla 4.4: Error medio cuadrático (RMSE) mínimo.

Recalling process	RMSE
Entrenamiento	0.0847
Validación	0.1128

Tabla 4.5: Comparación de la tensión RMS estimada y medida.

Recalling process	V(RMS) medido(V)	V(RMS) estimado(V)	Error relativo(%)
Entrenamiento	411.5099	413.1354	0.395
Validación	420.4006	409.2727	2.65

Tabla 4.6: Comparación de la potencia activa estimada y medida.

Recalling process	P(activa) medida (KW)	P(activa) estimada (KW)	Error relativo(%)
Entrenamiento	19045	20153	5.82
Validación	21687	21280	1.88

Al analizar las tablas 4.1,4.2,4.4 y 4.5 se observa que a medida que el valor RMSE es menor, el error relativo de la tensión RMS también lo es, esto es debido y como se explico anteriormente, es que el error medio cuadrático cuantifica que tan parecido es la tensión estimada a la tensión medida, ya sea en el recalling process con los datos de entrenamiento o validación, por lo que entre menor sea el valor RMSE significa que la tensión estimada tiene mayor similitud a la tensión medida por lo que al obtener los valores de las tensiones RMS estimada y medida serán mas parecidos y el error relativo entre estos será menor.

Lo anterior no se cumple exactamente para la potencia activa, al analizar las tablas 4.1,4.3,4.4 y 4.6 se observa que aunque en el recalling process con los datos de entrenamiento se presenta un valor RMSE menor y como se concluyó anteriormente es una mejor estimación de la tensión eléctrica y se esperaría que la potencia eléctrica instantánea presentara este comportamiento, sin embargo en las figuras 4.3 y 4.9, se presenta una buena estimación pero en sus formas de onda siempre se tienen sobre picos con respecto a la potencia instantánea medida por lo que esto afecta cuando se calculan las potencias activas ya que se presentan errores relativos en la tabla 4.4 y 4.6 por encima del 5%. Por otro lado al analizar las tablas para el recalling process con los datos de validación se observa que se presenta otro tipo de comportamiento, que aunque se tiene un valor mayor para el RMSE, el error relativo de la potencia activa es menor. En las tablas 4.4 y 4.6 se presentan valores relativos menores al 3% para la potencia activa, este comportamiento se muestra en las figuras 4.6 y 4.12, y como se explico anteriormente aunque se presenta una adecuada estimación para la potencia eléctrica instantánea, en los picos el modelo no es capaz de estimarlo adecuadamente, por lo que en algunas zonas se presentan sobre picos y en otros sub picos con respecto a la potencia instantánea medida y al calcular la potencia activa que es el promedio de la potencia instantánea se presenta un menor error ya que en promedio se equilibra mostrando un valor similar al de la potencia medida.

Capítulo 5

Conclusiones

En este documento se presenta un modelado basado en redes neuronales artificiales, que combina la red neuronal RBFNN y el método LUT, con esto se permite grabar los parámetros asociados a la red para cada muestra instantánea medida, debido a que la aplicación de la red se hace de manera secuencial permitiendo encontrar la relación entrada/salida sin necesidad de utilizar múltiples entradas lo que facilita la implementación. De esta manera la característica dinámica, el comportamiento de la forma de onda de la tensión eléctrica y su tendencia, se puede modelar adecuadamente.

Dos casos de análisis se expusieron para evaluar el método implementado, los resultados expuestos en el capítulo anterior indican un comportamiento consistente, ya que en ambos casos las gráficas presentan comportamientos similares y los resultados expuestos en las tablas de manera general presentan un error relativo menor al 3%, exceptuando cuando se analiza la potencia activa para el recalling process con los datos de entrenamiento, que se presentan errores por encima del 3% pero no superan el 7% por lo que modelar este tipo de carga con esta metodología es adecuada y efectiva ya que es una alternativa fácil de implementar para obtener el modelo matemático a partir de las mediciones de corriente y tensión eléctrica.

El metodo permite ser punto de partida para realizar diferentes estudios de ingeniería para mejorar y/o ajustar el rendimiento de la carga en las aplicaciones de tiempo real. También promover estudios que se centren en la mejora del método implementado, como se explicó en el documento, obtener un mejor punto de partida, encontrar un valor optimo para los factores de aprendizaje permitiría mejorar el modelo a partir de un algún preentrenamiento de los datos.

El método propuesto también se puede implementar para diferentes aplicaciones de aprendizaje y modelado de otras cargas no lineales de manera eficiente y simple.

Bibliografía

- [1] P. P. Cruz, *Inteligencia artificial con aplicaciones a la ingeniería*. Ciudad de México, México: Alfaomega Grupo Editor, S.A. de C.V, 2010.
- [2] G. W. Chang, C. Chen, and Y. Liu, “A neural-network-based method of modeling electric arc furnace load for power engineering study,” *IEEE*, vol. 25, no. 1, pp. 138–146, 2010.
- [3] J. S. R. Jang, C. T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing*. Englewood Cliffs, New Jersey: Prentice-Hall, 1997.
- [4] S. Haykin, *Neural Networks and Learning Machines*. Upper Saddle River, New Jersey: Pearson Education Inc, 2009.
- [5] A. R. Sadeghian and J. D. Lavers, “Application of radial basis function networks,” *IEEE*, vol. 6, pp. 3996–4001, 1999.
- [6] L. Yingwei, N. Sundararajan, and P. Saratchandran, “Performance evaluation of a sequential minimal radial basis function (rbf) neural network learning algorithm,” *IEEE*, vol. 9, no. 2, pp. 308–318, 1998.
- [7] S. Chen, C. F. N. Cowan, and P. M. Grant, “Orthogonal least squares learning algorithm for radial basis function networks,” *IEEE*, vol. 2, no. 2, pp. 302–309, 1991.
- [8] J. Zhang, G. G. Walter, Y. Miao, and W. N. W. Lee, “Wavelet neural networks for function learning,” *IEEE*, vol. 43, no. 6, pp. 1485–1497, 1995.
- [9] G. W. Chang, M. Shih, Y. Chen, and Y. Liang, “A hybrid wavelet transform and neural-network-based approach for modelling dynamic voltage-current characteristics of electric arc furnace,” *IEEE*, vol. 29, no. 2, pp. 815–824, 2014.