

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

CONTROL DE UNA PLANTA PILOTO DE PRESIÓN PARA LA COMPARACIÓN DE CONTROL PID CONVENCIONAL CON REDES NEURONALES

Didier Arbey García Galvis

Darwin Antonio Soto Castro

Ingeniería Mecatrónica

Director del trabajo de grado
Orlando Zapata Cortés

INSTITUTO TECNOLÓGICO METROPOLITANO

01 de septiembre de 2019

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

RESUMEN

En este trabajo se desarrolló el control de una planta piloto de presión por medio de redes neuronales, usando la posibilidad de entrenamiento y aprendizaje que estas brindan, aumentando con esto las áreas de enseñanza que abarca el módulo, no solo a sistemas de control, sino también a asignaturas como sistemas inteligentes.

Con la implementación de esta planta piloto de presión se realizó la comparación entre ambas técnicas (control convencional vs control inteligente) en cuanto a eficiencia y utilidad.

Dicha comparación se lleva a cabo por medio de datos experimentales tomando en cuenta los resultados obtenidos con el controlador actualmente utilizado (controlador convencional) y el del controlador aquí propuesto (controlador inteligente). Se realizan una serie de perturbaciones una vez que el sistema se encuentre en su estado estable para así obtener los tiempos de respuesta de ambos sistemas y determinar cuál de ellos es más eficiente. Con esto se espera comprobar la eficiencia de técnicas de control inteligente como las redes neuronales aplicadas a este tipo de procesos, con respecto a las técnicas usadas comúnmente.

El control se realiza en la placa de desarrollo Arduino Mega en comunicación serial con Matlab para la visualización de las variables y las gráficas en tiempo real, de esta manera observar con mayor comodidad el comportamiento del sistema.

Palabras clave: Inteligencia artificial, RNA (Red Neuronal Artificial), Control inteligente, Control Convencional, Presión.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

RECONOCIMIENTOS

Se desea brindar agradecimientos a las personas que hicieron parte de la evolución de este proyecto, que nos brindaron su ayuda y conocimientos, a los compañeros egresados de la institución que realizaron el módulo didáctico para el control de la variable presión, que gracias a su producto de laboratorio surge la idea de implementar redes neuronales y comparar el control convencional con el control inteligente. Se hace mención especial a nuestro asesor de trabajo de grado Orlando Zapata Cortés y al Instituto Tecnológico Metropolitano, por prestarnos sus servicios en el laboratorio de fluidos con buena disponibilidad para trabajo independiente, y así llevar a cabo el desarrollo de este proyecto.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

ACRÓNIMOS

PWM - Modulación por ancho de pulso

XLSX - Hoja de cálculo

RNA- Red Neuronal Artificial

EMC- Error Medio Cuadrado

N- Coeficiente de aprendizaje

P - Control proporcional

PI - Control proporcional integral

PID - Control proporcional integral derivativo

K_p - Constante proporcional

K_i - Constante integral

K_d - Constante derivativa

K - Ganancia estática

MLP – Multi Layer Perceptron (Perceptrón multicapa)

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Tabla de contenido

1. INTRODUCCIÓN	7
1.1. Objetivos	8
1.1.1. Objetivo General	8
1.1.2. Objetivos Específicos	8
2. MARCO TEÓRICO	10
2.1 Instrumentación y Control	10
2.1.1. Dispositivos neumáticos	10
2.1.2. Compresor de aire	10
2.1.3. Tanques de aire	11
2.1.4. Sensor de presión	12
2.1.5. Válvula Antirretorno	12
2.1.6. Válvula Solenoide	12
2.1.7. Filtro De Aire	13
2.1.8. Sistemas De Control	13
2.1.9. Circuito De Control	14
2.1.10. Sistemas Embebidos	14
2.1.11. Comunicación Serial	15
2.1.12. Fuente De Alimentación	15
2.1.13. Módulo Didáctico Para El Control De La Variable Presión	15
2.1.14. Presión De Aire	16
2.2. Entorno de Programación	17
2.2.1. Matlab	17
2.2.2. Lenguaje de programación (Matlab)	17
2.2.3. Guide	17
2.3. Sistemas Inteligentes	17
2.3.1. Inteligencia artificial	17
2.3.2. Algoritmos genéticos	18
2.3.3. Red Neuronal Artificial	18
2.3.4. Aprendizaje De Una Red Neuronal	20

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

2.3.5. Entrenamiento De Una Red Neuronal.....	20
2.3.6. Pesos	20
2.3.7. Bias	20
2.3.8. Perceptrón Multicapa	20
2.3.9. Capas Ocultas	20
2.3.10. Coeficiente De Aprendizaje	21
2.3.11. Error Máximo Deseado.....	21
2.3.12. Feedforward.....	21
2.3.13. Backpropagation	21
2.3.14. Lógica difusa	21
3. ESTADO DEL ARTE	23
4. METODOLOGIA	32
5. RESULTADOS Y DISCUSIÓN.....	53
6. CONCLUSIONES, RECOMENDACIONES Y TRABAJO A FUTURO	60
6.1 Conclusiones	60
6.2 Recomendaciones	61
6.3 Trabajo futuro	61
Bibliografía	62

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

1. INTRODUCCIÓN

El control de una planta piloto de presión para la comparación de control PID convencional con redes neuronales, consiste en el desarrollo e implementación de técnicas de redes neuronales, al realizar el control de la variable presión por medio del controlador inteligente, se obtuvieron resultados que difieren del método de control convencional, incluyendo tiempos de respuesta en tiempo real, llegando a la conclusión de cuál es el método más eficiente, además de aportar a la investigación en el área de la inteligencia artificial.

Matlab es el programa principal en el cual se realizó la comunicación serial y una interfaz gráfica amigable, de fácil uso y acceso, proporcionando al educando una manera cómoda para ingresar variables y visualizar resultados en tiempo real.

El uso de los sistemas inteligentes es poco común, no es de conocimiento general para el público ya que es una ciencia que está en continuo desarrollo y se torna necesario tener una mejor profundización de este tema. Debido a esto se vuelve un desafío para los docentes el hecho de dar a conocer este tema sin crear confusiones a los estudiantes. De ahí parte la idea de exponer las técnicas de los sistemas inteligentes por medio del control del módulo didáctico de presión usando redes neuronales para comparar el control convencional y el control inteligente y la importancia de estos métodos para mejorar el aprendizaje de los estudiantes del Instituto Tecnológico Metropolitano y así facilitar el trabajo de los docentes al uso de estas tecnologías y crear experiencias nuevas para el avance de esta materia y afines. Por tal motivo este módulo está implementado para proporcionar un mejor aprendizaje a los estudiantes del Instituto Tecnológico Metropolitano, haciendo uso de las técnicas de los sistemas inteligentes, aportando con esto a las áreas de enseñanza que abarca el módulo. Se espera que con la comparación entre ambas técnicas (control convencional vs control

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

inteligente) y los resultados de estas, se pueda definir qué método es más eficiente y de mejor implementación.

También se realizó una interfaz virtual, en la que los educandos pueden ingresar de manera fácil todos los valores requeridos para el control del módulo por medio de las redes neuronales, y del mismo modo visualizar la respuesta frente a estos.

1.1. Objetivos

1.1.1. Objetivo General

- Controlar la planta piloto didáctica de presión implementando redes neuronales con el fin de realizar comparaciones en tiempo real con el control PID convencional actual utilizando datos experimentales.

1.1.2. Objetivos Específicos

- Identificar técnicas de control por redes neuronales aplicadas en procesos reales, investigando las metodologías actuales que están siendo estudiadas.
- Implementar control por medio de redes neuronales al módulo de presión, tomando en cuenta las variables a manipular y los datos recolectados.
- Validar con datos experimentales en tiempo real los resultados obtenidos con el control con la red neuronal frente a los resultados actuales del controlador PID para

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

confirmar si se presenta una disminución significativa del error y mejoras generales en el control automático actual de la planta.

- Fomentar el aprendizaje de las redes neuronales a las futuras generaciones, dejando preparado el módulo para su uso en las diferentes áreas del conocimiento que este abarca, incluyendo la documentación de la realización de este proyecto.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

2. MARCO TEÓRICO

En este capítulo se proporciona información de componentes de instrumentación y control, cómo estos brindan la posibilidad de medir, convertir, monitorear y registrar variables (en este caso presión), y luego, trasmitirla, evaluarla y controlarla.

Además, este capítulo cuenta con información necesaria de lo que son los algoritmos de control, enfocado en los sistemas inteligentes; cómo el lenguaje de programación permite el trabajo en conjunto y coordinado, a través de instrucciones posibles para llevar a cabo el control de la variable a medir (en este caso presión).

Es preciso afianzarse en los conceptos descritos a continuación, ya que son de vital importancia para entender el comportamiento del módulo y cómo estos influyen en su funcionamiento para el desarrollo de los capítulos posteriores.

2.1 Instrumentación y Control

2.1.1. Dispositivos neumáticos

Son aquellos dispositivos que emplea aire comprimido como energía para mover y hacer funcionar mecanismos. El aire al aplicarle una fuerza, se comprime, mantiene esta compresión y devolverá la energía acumulada cuando se le permita expandirse (Solé, 2007).

2.1.2. Compresor de aire

Los compresores son máquinas destinadas a elevar la presión de un cierto volumen de aire, admitido en condiciones atmosféricas hasta una determinada presión exigida en la ejecución de los trabajos realizados por el aire comprimido (Parker Hannifin Corporation, 2003).

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01



Figura 1. Compresor de aire utilizado para el módulo (Autoría propia).

2.1.3. Tanques de aire

Un tanque es un recipiente de diversos materiales, comúnmente de forma cilíndrica, son usados para almacenar el aire comprimido, enfriar el aire ayudando a la eliminación de condensado, compensar las fluctuaciones de presión en todo el sistema de distribución, estabilizar el flujo de aire, controlar las marchas de los compresores, entre otros (Parker Hannifin Corporation, 2003).



Figura 2. Tanques de aire utilizados para el módulo (Autoría propia).

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

2.1.4. Sensor de presión

Es un instrumento compuesto por un elemento detector de presión con el que se determina la presión real aplicada al sensor (utilizando distintos principios de funcionamiento) y otros componentes que convierten esta información en una señal de salida (HBM, 2019).



Figura 3. Sensor de presión utilizado para el módulo (Autoría propia).

2.1.5. Válvula Antirretorno

Tienen la misión de impedir el paso del aire en un sentido y dejarlo pasar en sentido opuesto. La obturación del paso puede lograrse con una bola, membrana, etc., impulsados por la propia presión de trabajo o bien con la ayuda complementaria de un muelle (Nicolás, 2009).



Figura 4. Válvula antirretorno utilizada para el módulo (Autoría propia).

2.1.6. Válvula Solenoide

son dispositivos diseñados para controlar el flujo (ON-OFF) de un fluido. Están diseñadas para poder utilizarse con agua, gas, aire, gas combustible, vapor entre otros. Estas válvulas pueden ser de dos hasta cinco vías. Pueden estar fabricadas en latón, acero inoxidable o PVC.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Dependiendo del fluido en el que se vayan a utilizar es el material de la válvula (Mattarollo, 2014).



Figura 5. Válvula solenoide utilizada para el módulo (Autoría propia).

2.1.7. Filtro De Aire

Este dispositivo elimina las impurezas micrónicas y ayuda en la remoción parcial de la humedad contenida en el aire comprimido (Parker Hannifin Corporation, 2003).



Figura 6. Filtro de aire utilizado para el módulo (Autoría propia).

2.1.8. Sistemas De Control

Un sistema de control se define como un conjunto de componentes que pueden regular su propia conducta o la de otro sistema con el fin de lograr un funcionamiento predeterminado,

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

de modo que se reduzcan las probabilidades de fallos y se obtengan los resultados buscados (Valenzuela, Alexandro, 2017).

2.1.9. Circuito De Control

es el componente fundamental de los sistemas de control industrial. Se refiere a todos los componentes físicos y funciones de control necesarios para ajustar automáticamente el valor de una variable de proceso medida para igualar el valor de un punto de ajuste deseado (Douglas, 2017).

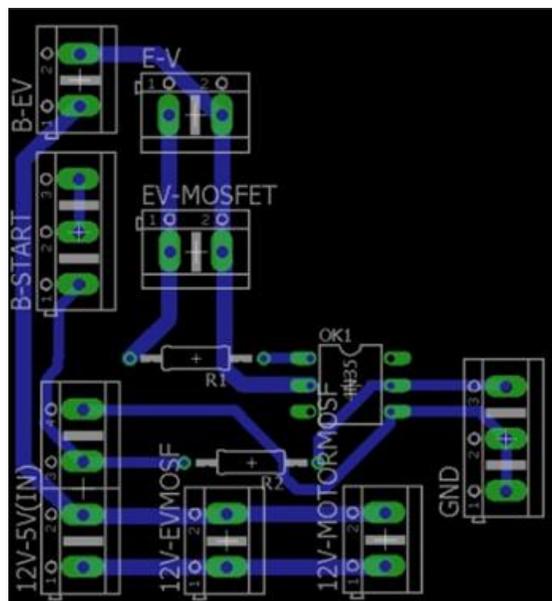


Figura 7. Esquema del circuito de control utilizado para el módulo (Autoría propia).

2.1.10. Sistemas Embebidos

Los sistemas embebidos a pesar de no ser muy nombrados están en muchas partes, desde vehículos hasta teléfonos celulares e incluso en algunos electrodomésticos comunes como refrigeradores y hornos microondas. Este no es más que un microprocesador que incluye interfaces de entrada/salida en el mismo chip. Normalmente estos sistemas poseen un interfaz externo para efectuar un monitoreo del estado y hacer un diagnóstico del sistema (Llinares, 2005).

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

2.1.11. Comunicación Serial

En telecomunicaciones y computación, la comunicación serial es el proceso de envío de datos de un bit por vez, secuencialmente, sobre un canal de comunicación o un bus de computadora. Contrasta con la comunicación paralela, donde todos los bits de cada símbolo (la más pequeña unidad de datos transmitida por vez) son enviados juntos (Alegsa, 2016).

2.1.12. Fuente De Alimentación

Es el dispositivo que se encarga de transformar la corriente alterna en corriente continua o directa; que es la que utilizan los dispositivos electrónicos tales como televisores y computadoras, suministrando los diferentes voltajes requeridos por los componentes, incluyendo usualmente protección frente a eventuales inconvenientes en el suministro eléctrico, como la sobretensión (Raffino, 2019).



Figura 8. Fuente de alimentación utilizada para el módulo (Autoría propia).

2.1.13. Módulo Didáctico Para El Control De La Variable Presión

Es una planta portátil, usando elementos a escala menor que realizan las mismas funciones de los elementos de una planta grande. En este módulo didáctico se pueden aplicar los mismos algoritmos de control usados en las plantas industriales.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

El módulo está diseñado para la realización de diversas prácticas enfocadas en el control, más específicamente el control de la presión en un tanque a escala.

Este dispositivo tiene la gran característica de ser modular, porque los estudiantes pueden modificar el hardware y el software a fines convenientes, ya que la información sensada por la placa de desarrollo puede ser llevada a cualquier software por medio de comunicación serial, y así usar el software con el cual se sienta más cómodo el educando. Matlab es el programa principal en el cual se realizó la comunicación serial y una interfaz gráfica amigable, de uso intuitivo y de fácil acceso, proporcionando al usuario una comodidad para ingresar variables y visualizar resultados en tiempo real.

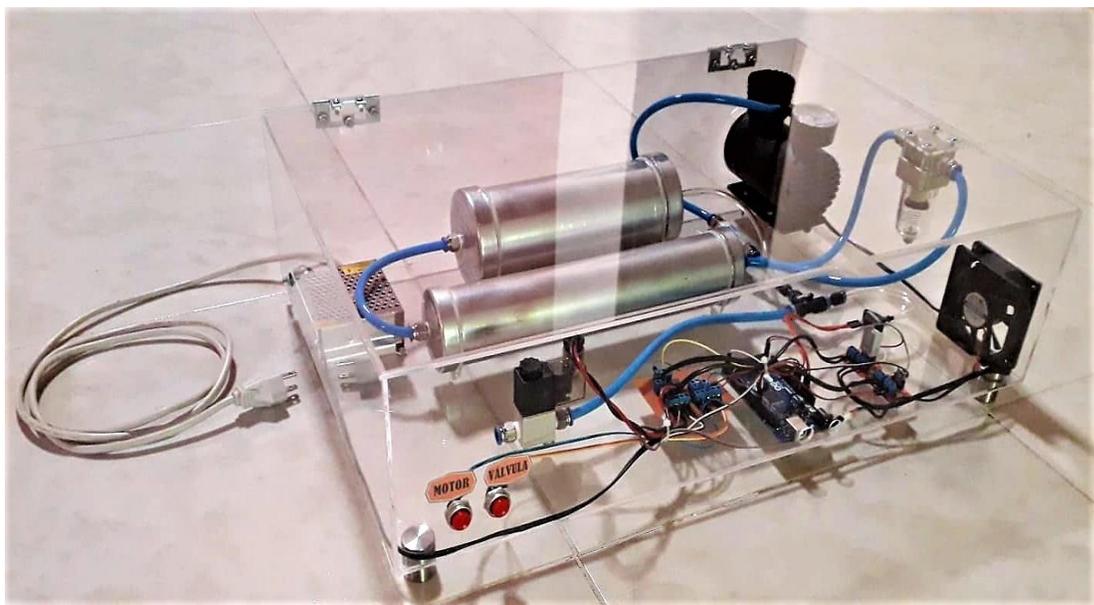


Figura 9. Modelo físico del módulo didáctico para el control de la variable presión (Autoría propia).

2.1.14. Presión De Aire

La presión es peso del aire por unidad de superficie, el cual disminuye con la altitud y se mide en libras por pulgada cuadrada (PSI), también en pascales (Pa) y en bar de presión (bar) (Parlow, 2008).

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

2.2. Entorno de Programación

2.2.1. Matlab

MATLAB (abreviatura de MATrix LABoratory, "laboratorio de matrices") es un sistema de cómputo numérico que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M). Está disponible para las plataformas Unix, Windows, Mac OS X y GNU/Linux (wikipedia, 2015).

2.2.2. Lenguaje de programación (Matlab)

MATLAB es un lenguaje de alto desempeño diseñado para realizar cálculos técnicos.

MATLAB integra el cálculo, la visualización y la programación en un ambiente fácil de utilizar donde los problemas y las soluciones se expresan en una notación matemática. Esto permite resolver muchos problemas computacionales, específicamente aquellos que involucren vectores y matrices, en un tiempo mucho menor al requerido para escribir un programa en un lenguaje escalar no interactivo tal como C o Fortran (Instituto Tecnológico De Ciudad Madero , 2002).

2.2.3. Guide

Es un entorno de programación visual disponible en Matlab para realizar y ejecutar programas que necesiten ingreso continuo de datos. Tiene las características básicas de todos los programas visuales como Visual Basic o Visual c++ (Guerrero, 2008).

2.3. Sistemas Inteligentes

2.3.1. Inteligencia artificial

La inteligencia artificial puede definirse como la capacidad de una máquina para aprender de su entorno o de una tarea específica y usar dicho aprendizaje para resolver problemas o llegar

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

a un objetivo por sí misma. En las últimas décadas, el campo de la inteligencia artificial se ha ido desarrollando y se han logrado grandes avances en diversos campos, especialmente en clasificación, reconocimiento de patrones, reconocimiento de voz, conducción autónoma, e incluso en el campo de la medicina. Existen múltiples técnicas de inteligencia artificial, entre ellas están los algoritmos genéticos, las redes neuronales y la lógica difusa.

2.3.2. Algoritmos genéticos

Los algoritmos genéticos son métodos basados en la genética de los seres vivos, la teoría de la evolución y la selección natural. En este se definen individuos, los cuales son posibles soluciones a un problema dado. La agrupación de estos individuos se denomina población. En cada población se le asigna un valor a cada individuo, el cual indica qué tan acertada es su solución al problema (fitness). Una vez obtenido el fitness de cada individuo de la población, se realiza una mezcla entre la información de los individuos (cruce) y se realizan modificaciones aleatorias a estos (mutación). El resultado de estas operaciones se almacena en una nueva población que pasará a ser la nueva generación. También es una práctica común seleccionar uno o más individuos con el mejor fitness y hacerlos parte de la siguiente generación, simulando así la supervivencia del más apto. Estas modificaciones se realizan en múltiples iteraciones (épocas) hasta que se encuentre al individuo óptimo que dé solución al problema, estando dentro de un margen de error permisible dado.

2.3.3. Red Neuronal Artificial

Para definir una red neuronal artificial (RNA), primero hay que definir la neurona artificial, la cual será la unidad base de una RNA.

Una neurona artificial (también llamada perceptrón simple) simula una neurona natural, teniendo conexiones de entrada y de salida, y procesando los datos de entrada antes de entregarlos en la salida. Esta neurona no es más que una función matemática, la cual realiza una suma ponderada de las entradas multiplicadas por un valor llamado peso, que indica la importancia de cada conexión de entrada, y otro valor llamado bias que nos otorga un desplazamiento del resultado obtenido. Esta suma se pasa entonces por una

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

función de activación, la cual agregará deformaciones no lineales a la salida, muy importante al trabajar con redes neuronales.

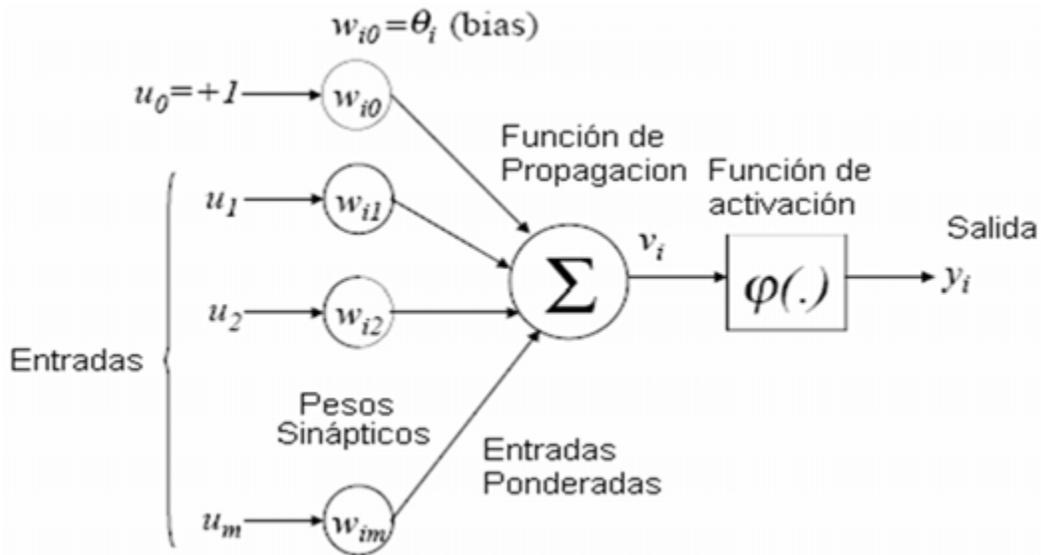


Figura 10. Neurona artificial (Diaz Rodríguez, Jorge, 2019).

El uso de una sola neurona tiene la limitación de que solo puede resolver problemas linealmente separables, y es aquí donde se hace necesario encadenar neuronas, creando así una red neuronal.

Una RNA es un paradigma de aprendizaje y procesamiento automático inspirado en el funcionamiento del sistema nervioso humano. Una red neuronal está compuesta por un conjunto de neuronas interconectadas entre sí mediante enlaces, donde cada neurona toma como como entradas las salidas las neuronas antecesoras, multiplica cada una de esas entradas por un peso y mediante una función de activación calculan una salida. esta salida es a su vez entrada de la neurona a la que precede.

La unión de todas estas neuronas interconectadas es lo que compone la red neuronal artificial (Calvo, 2017).

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

2.3.4. Aprendizaje De Una Red Neuronal

La red neuronal artificial al igual que las redes biológicas aprenden por repetición, y cuantos más datos se tengan para entrenar y más veces se entrene a la red mejores resultados se obtendrán (Calvo, 2017).

2.3.5. Entrenamiento De Una Red Neuronal

Entrenar una red neuronal artificial (RNA) es un proceso que modifica el valor de los pesos y bias asociados a cada neurona, con el fin de que la RNA pueda a partir de unos datos presentados en la entrada, generar una salida (Gomez, Henao, & Salazar, 2004).

2.3.6. Pesos

Los pesos son realmente la forma de aprender que tendrá la neurona. Estos pesos se irán modificando de cierta manera para ir adaptando el valor de la salida de tal forma que se minimice su error con respecto al resultado real que debería sacar la neurona artificial (Domínguez, 2019).

2.3.7. Bias

Algunos modelos de red empleados tienen una entrada extra, la cual es denominada “bias” cuyo único objetivo es lograr una convergencia más rápida de la red (Alberto Delgado, 1998).

2.3.8. Perceptrón Multicapa

El perceptrón multicapa es una red neuronal artificial (RNA) formada por múltiples capas, de tal manera que tiene capacidad para resolver problemas que no son linealmente separables, lo cual es la principal limitación del perceptrón (también llamado perceptrón simple). El perceptrón multicapa puede estar total o localmente conectado (Wikipedia. Redes de Neuronas Artificiales, 2012).

2.3.9. Capas Ocultas

Son aquellas formadas por neuronas cuyas entradas provienen de capas anteriores y cuyas salidas pasan a neuronas de capas posteriores (Wikipedia. Redes de Neuronas Artificiales, 2012)

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

2.3.10. Coeficiente De Aprendizaje

Es un valor numérico pequeño que se proporciona al momento de entrenar la red neuronal, con el fin de disminuir el error de cada neurona en pequeños pasos, ya que de hacerlo completamente podría ocasionar que la red se vuelva oscilatoria y nunca converja.

2.3.11. Error Máximo Deseado

Es el valor de error permisible que se le da a la red. Si el error de la red está por debajo de este valor, se dice que la red converge, y ha aprendido la tarea para la cual fue entrenada. Este valor depende en gran medida de la tarea que se quiere realizar y qué tan precisa debe ser la salida de la red.

2.3.12. Feedforward

Es un proceso que parte de las neuronas de la capa de entrada y termina en las neuronas de la capa de salida, cada neurona realiza una suma ponderada de todas las entradas de acuerdo a unos pesos, pasa el resultado por una función de activación y genera el resultado, que pasa a la siguiente capa. Este proceso se repite hasta la última capa de neuronas que nos devuelve el resultado de la red (Calvo, 2017).

2.3.13. Backpropagation

El backpropagation trata de determinar cuánto aporta cada una de las neuronas al error total de la red neuronal. El nombre de backpropagation viene de que el algoritmo busca el error correspondiente a cada neurona de manera iterativa desde la última capa hasta la primera (Calvo, 2017).

2.3.14. Lógica difusa

Las técnicas de lógica difusa buscan resolver problemas en los que tenemos valores que oscilan entre 2 extremos opuestos, permitiendo así manejar información imprecisa, en la que los datos tienen cierto grado de pertenencia a cada uno de los grupos definidos para la clasificación de un problema o variable.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Por citar algunos ejemplos de uso, la empresa japonesa Matsushita utiliza en sus lavadoras un sistema de control que determina automáticamente el ciclo de lavado según el tipo, cantidad de suciedad y tamaño de la colada. Los estabilizadores de imágenes en sus cámaras digitales incorporan reglas que eliminan las vibraciones involuntarias de la mano del operario, comparando la imagen actual con las imágenes anteriores de la memoria. En el ámbito de la automoción, Mitsubishi y General Motors emplean sistemas de transmisión automática y control de temperatura basados en lógica difusa (González, 2011).

Con los conceptos descritos anteriormente se espera concretar un mejor entendimiento de los componentes y el módulo a utilizar, como estos son influenciados por diferentes métodos de control, en este caso los sistemas inteligentes, dando a conocer una de sus técnicas como son las redes neuronales, debido a su gran capacidad de adaptación a diferentes tipos de problemáticas, la experiencia previa con el uso de éstas, y la facilidad de implementación de este tipo de técnica, además de ser una técnica con gran potencial que está causando una revolución al demostrar ser el futuro de la tecnología.

Con el fin de tener claro el uso de los sistemas inteligentes, se ve necesario resaltar la importancia del estado del arte, con su profundización, se proporciona información relevante de los avances y técnicas que se han propuesto para la elaboración del control por medio de redes neuronales.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

3. ESTADO DEL ARTE

Las redes neuronales artificiales surgen dentro del campo de la inteligencia artificial, simulando el comportamiento de una red neuronal biológica, con el fin de resolver problemas complejos que serían muy difíciles de resolver empleando algoritmos convencionales. (Gonzales, 2010). Hay muchos tipos diferentes de redes neuronales artificiales, los cuales son usados para diferentes aplicaciones dependiendo de su desarrollo. Estas redes son ampliamente utilizadas para tareas tales como:

- Clasificación de datos.
- Predicción de resultados.
- Detección de patrones.
- Obtención de modelos de la retina y del funcionamiento del cerebro.
- Evaluación de probabilidades.
- Explotación de bases de datos.
- Optimización.
- Reconocimiento de caracteres escritos.
- Previsión del clima.
- Identificación de falsificaciones.
- Visión artificial.
- Entre otros.

Sin embargo, en lo referente al campo de modelado de sistemas para la automatización y control (Robots automatizados, sensórica en general, controladores, etc.), las redes neuronales artificiales son relativamente de uso reciente, sin embargo, el uso de estas está incrementando en este tipo de sistemas debido a la eficiencia de los resultados que estas pueden generar, ahorrando así el uso de cálculos complejos con un mejor rendimiento

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

general. Actualmente, los sistemas de control por medio de redes neuronales artificiales pueden resumirse en 4 estructuras, estas son:

Control Supervisado: la red neuronal aprende un conjunto de entradas y las salidas deseadas para dar solución al problema tratado.

Control Inverso Directo: la red neuronal aprende de la retroalimentación de un sistema, con lo que, al obtener la señal, esta determina el control a realizar.

Retropropagación de Utilidad: esta estructura optimiza la función de transferencia del sistema, aunque requiere de un modelo del sistema a controlar.

Crítico Adaptativo: similar a la estructura de retropropagación de utilidad, pero sin necesidad de un modelo de la planta. (Matich, 2001)

Múltiples investigaciones pueden ser encontradas sobre diversos temas en cuanto a la aplicación de redes neuronales artificiales, y en cuanto al campo de los sistemas de control pueden mencionarse ejemplos como los siguientes:

Control de fallas para robots exploradores en marte: considerando la posibilidad de fallas en los actuadores de los robots exploradores en marte, y las inclemencias del clima en dicho planeta, se propuso un sistema de control por medio de redes neuronales artificiales para brindar una capacidad de detección de fallas autónoma, a su vez de la posibilidad de auto reconstrucción (Huang, Li, & Sun, 2019). La red neuronal utilizada para este propósito se muestra en la figura 11.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

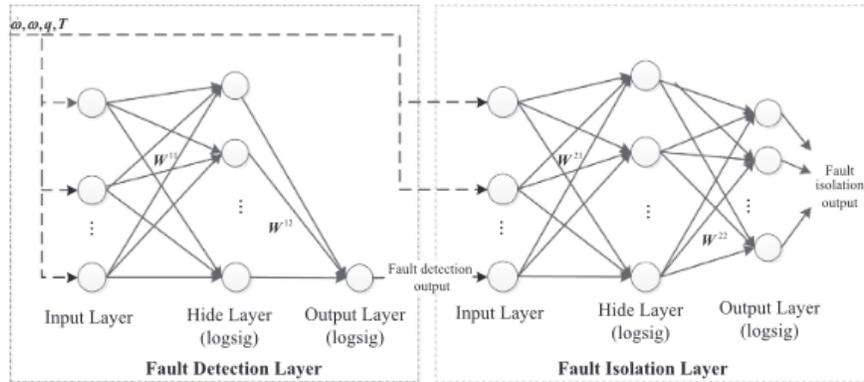


Figura 11. Red neuronal para robot explorador (Huang et al, 2019).

Adicionalmente, se incluye en este artículo el diagrama del controlador diseñado para dicho fin (figura 2), donde pueden apreciarse los puntos específicos en los que fueron usadas las redes neuronales y su interacción con el sistema.

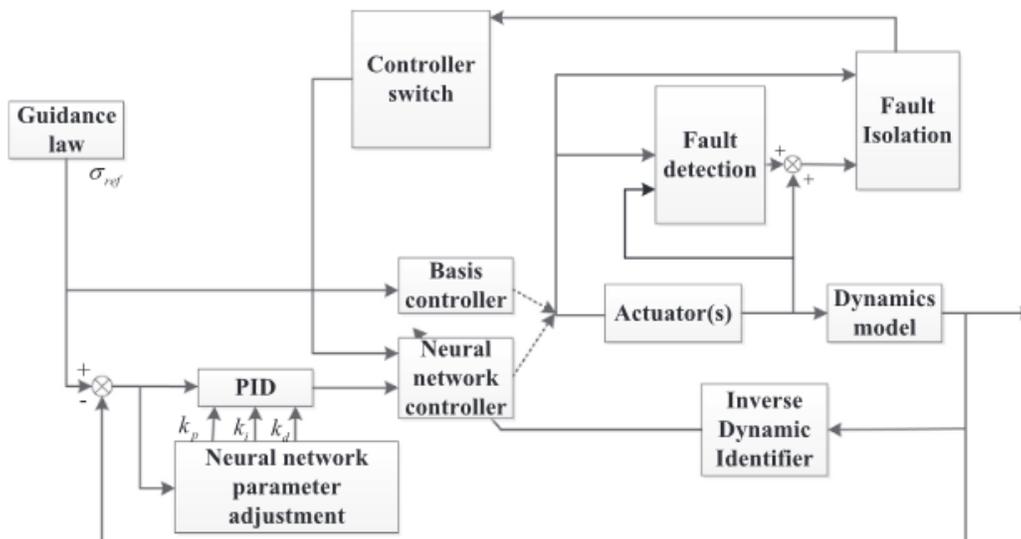


Figura 12. Controlador para robot explorador (Huang et al, 2019).

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Estos modelos fueron verificados por medio de simulaciones y se pudo concluir que estos métodos de control inteligente son perfectamente aplicables a problemas que requieren una alta confiabilidad como lo son las misiones de exploración espacial.

Red neuronal para controlar el transporte aéreo de una carga suspendida por un cable:

En este proyecto se pretende comprobar la efectividad del control por medio de redes neuronales llevando un registro de la trayectoria basado únicamente en la posición de la aeronave. Esto fue pensado debido a que últimamente se han estado empleando cuadricópteros (drones) para realizar entregas y transportar paquetes en situaciones de espacio reducido o con configuraciones estructurales complejas, obteniendo resultados muy prometedores (Baron Gomez & Ramos Velasco, 2017).

Al igual que en el artículo anterior, se incluye la red neuronal utilizada para dicho propósito y el controlador diseñado.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

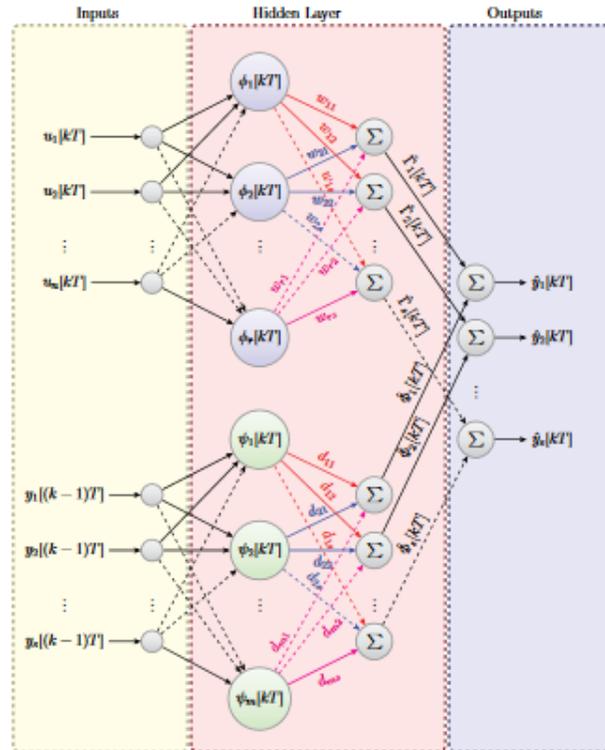


Figura 13. Red neuronal carga suspendida (Baron Gomez & Ramos Velasco, 2017).

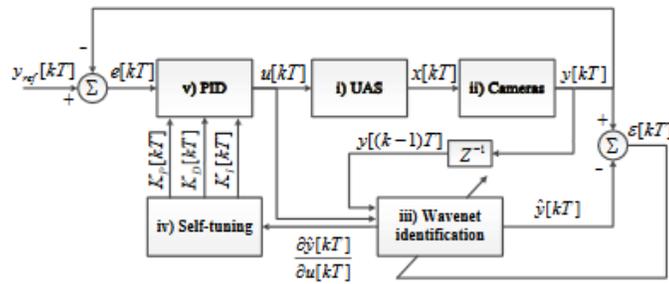


Figura 14. Controlador carga suspendida (Baron Gomez & Ramos Velasco, 2017).

La metodología propuesta fue evaluada por medio de un experimento en tiempo real, transportando una carga por medio de un cable usando un dron y obteniendo un resultado satisfactorio, demostrando así la aplicabilidad y efectividad del control diseñado. Los

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

siguientes retos en este campo serían el control en un entorno donde actúen múltiples agentes, o en casos donde varios dispositivos lleven una misma carga.

Control PID por redes neuronales para sistemas de control no lineal multivariable: Este artículo se centra en una problemática relacionada directamente al control por medio de redes neuronales artificiales, el cual es la obtención de parámetros iniciales óptimos para los pesos de las conexiones de cada neurona, lo cual tiene un alto impacto en el resultado final y en el rendimiento de toda la red, especialmente en un control multivariable. Este tipo de control siempre ha sido un reto debido a que la calibración de los parámetros usados en un PID convencional no suele arrojar buenos resultados cuando son enfrentados a un sistema no lineal. Es aquí donde se pueden explotar las características que las redes neuronales pueden ofrecer en conjunto con el control PID. Sin embargo, si se usa una selección aleatoria de los pesos de cada neurona en dicha red, la eficiencia en los sistemas multivariable puede verse comprometida, con lo que se ha propuesto determinar estos valores por medio de un algoritmo evolutivo. (Hu, Zhao, & Liang, 2012)

Se incluye un modelo de la red neuronal propuesta para dicho fin.

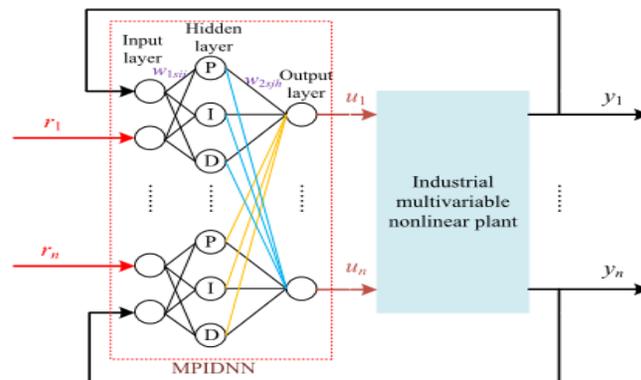


Figura 15. Red neuronal para control multivariable (Hu, Zhao, & Liang, 2012).

Estos modelos fueron comprobados por medio de simulación, y aunque resultan ser efectivos, estos resultados pueden ser mejorados ya que aún hay mucho campo de investigación en este ámbito.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Partiendo del hecho de que las redes neuronales pueden definirse como formas de computación o modelos matemáticos compuestos por un número de elementos conectados entre sí para procesar datos y brindar una salida esperada, es necesario tener claro este concepto ya que puede llegar a ser complicado debido al uso de algoritmos y fórmulas matemáticas que estas conllevan.

Por consiguiente, se plantean una serie de problemas que surgen cuando se trabajan con redes neuronales aplicadas a sistemas de control y su manejo en las variables.

Inicialmente, se debe analizar el dominio del problema y decidir a qué clase pertenece. Luego debe decidirse si una red neuronal es adecuada para resolver dicho problema. (Matich, 2001)

Según Matich, ante estas incógnitas, se llega a la llamada etapa preliminar. Concluida esta etapa, se abordan una serie de preguntas que han de ser respondidas para alcanzar el punto de partida frente a la propuesta a realizar con los sistemas de control de una red neuronal:

- **Origen de los datos.**

¿Qué datos son de vital importancia para el manejo del problema a tratar ya definido? ¿Qué variables son relevantes para abordar y manejar dicho problema? ¿De dónde pueden obtenerse los datos?

Para este proyecto en específico, se requiere analizar a profundidad el proceso, para obtener la función de transferencia de este y todas las variables que influyen en la problemática a tratar. Estos datos se adquirirán por medio de ensayos experimentales usando el módulo didáctico de la variable presión.

- **Preparación y codificación de los datos.**

¿Cómo preparar y codificar los datos?

Los datos obtenidos de los ensayos experimentales se organizarán en una hoja de cálculo para luego ser ingresado al software que se utilizará para codificar la red neuronal (Matlab).

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

- **Topología de la red (dependiendo de la preparación y codificación de los datos).**

¿Qué tipo de red debe escogerse? ¿Cuántas capas ocultas y con cuántas neuronas son necesarias el manejo de la posible solución al problema propuesto? ¿Qué regla de aprendizaje escoger? ¿Qué inicialización para los pesos?

La red que se diseñará será seleccionada inicialmente con una configuración de 2 capas ocultas con 20 neuronas en cada capa, y un coeficiente de aprendizaje de 0.3 y pesos aleatorios. Sin embargo, una vez implementada, se realizarán varias pruebas para determinar si un cambio en alguno de los parámetros es necesario para obtener mejores resultados. La estructura que se pretende implementar para este proyecto se muestra en la figura 15. Se propone usar la red neuronal como el sistema de control en general, y se explorará la posibilidad de intercambiar entre la red y el PID convencional. Dicha estructura podría ser cambiada si se descubre una mejor alternativa durante el proceso de desarrollo e implementación.

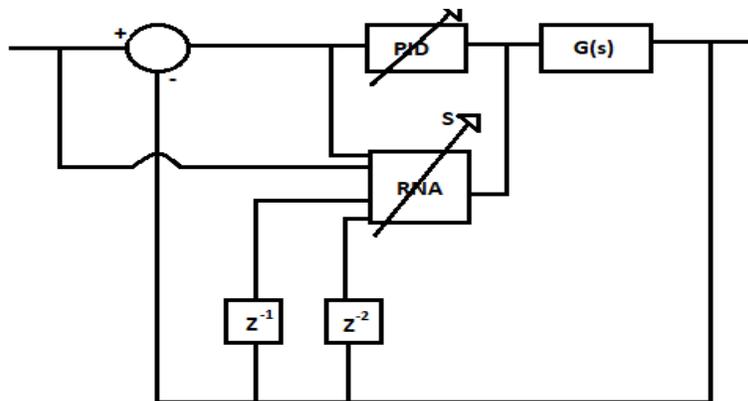


Figura 16. Esquema inicial propuesto (Autoría propia).

- **Decisiones concernientes al proceso de aprendizaje.**

Se entrenará la red neuronal artificial ingresando los valores obtenidos de los ensayos preliminares mencionados anteriormente, además de los valores esperados, realizando ciclos según sea necesario hasta que la red tenga un error permisible.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Ahora bien, después de haber brindado una solución a cada pregunta propuesta se plantea la comparación de un sistema de control automático inteligente utilizando técnicas de redes neuronales y sus ventajas frente al sistema convencional de control, el cual se espera encontrar un mejor tiempo de respuesta, mayor exactitud en función de los datos obtenidos frente a la posibilidad de obtener mejoras del error, mayor eficiencia en el control frente al control convencional, coste y utilidad.

Un control más rápido y eficiente podría significar un mayor índice de ganancias para las industrias en las que sea implementado, además de aportar una solución confiable y segura en procesos que requieran alta precisión, además de fomentar el uso y ampliar la documentación existente sobre este tipo de técnicas aplicadas al control de procesos.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

4. METODOLOGIA

Inicialmente se propuso un modelo de red neuronal fijo con una configuración de 2 capas ocultas con 20 neuronas en cada capa, 2 entradas, 2 salidas y un coeficiente de aprendizaje de 0.3 y pesos aleatorios. Sin embargo, después de analizar la naturaleza del módulo didáctico para ayudar a los estudiantes a entender conceptos de algunas de las asignaturas de la ingeniería mecatrónica, se decidió realizar una interfaz en la cual los estudiantes pudieran configurar sus propias redes neuronales, y así el módulo siguiera cumpliendo su objetivo inicial. Este cambio hizo más fácil realizar pruebas con distintas configuraciones, encontrando así que la más veloz era una red neuronal de 1 capa oculta de 13 neuronas, 2 entradas, 1 salida y un coeficiente de aprendizaje de 0,01 como se puede observar en la figura 17.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

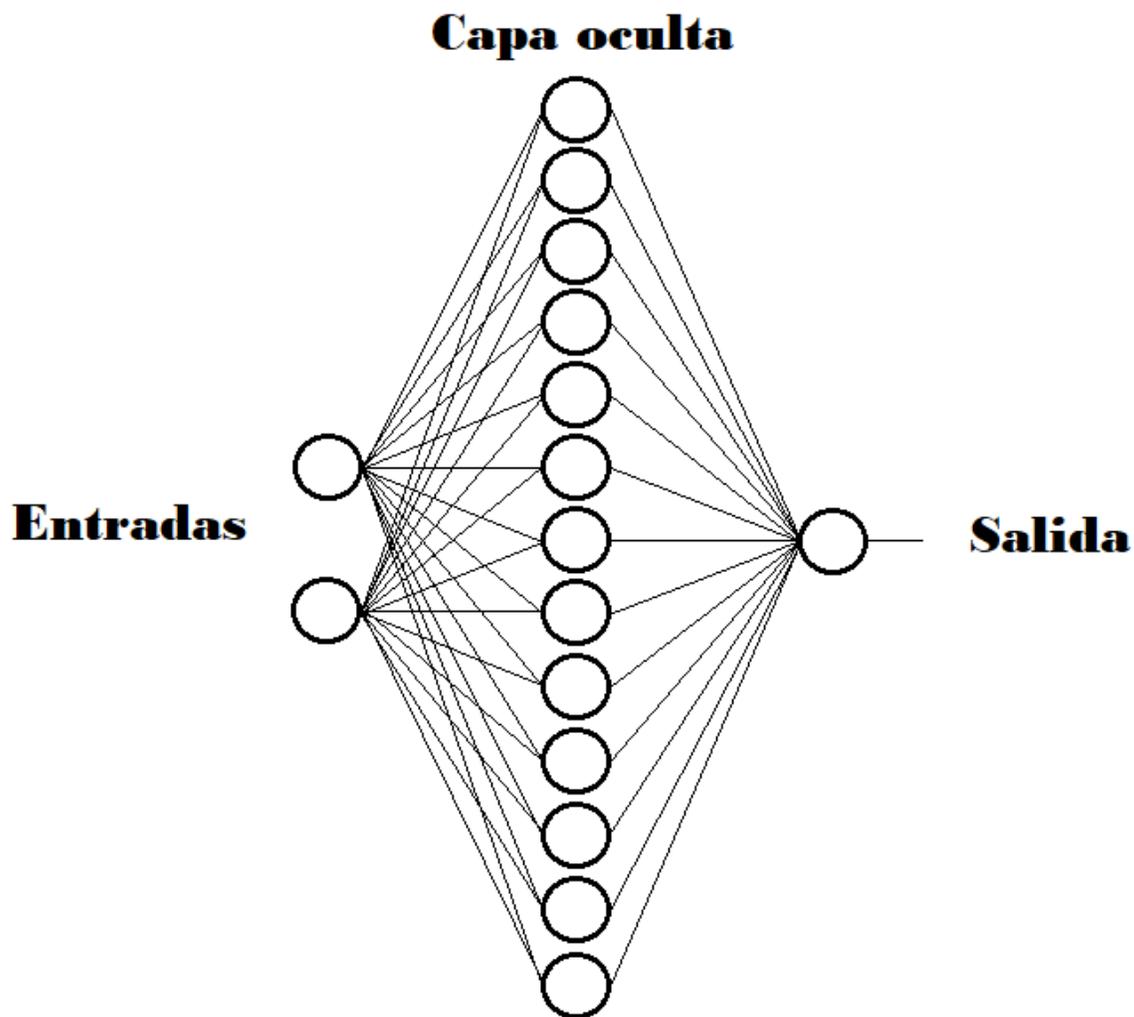


Figura 17. Configuración de red neuronal usada para el control neuronal (Autoría propia).
 Se realizó un acercamiento al módulo de presión con el programa que usa el control convencional, evaluando el estado de sus componentes y el funcionamiento del programa usado para realizar el control de este. Se encontró que el compresor presentaba fallas y era necesario reemplazarlo, por lo que se efectuó la compra de un nuevo compresor para garantizar que todo el sistema funcione de manera óptima. El resto de los componentes funcionaban con normalidad. Ningún cambio importante se hizo en cuanto a la estructura o funcionamiento del módulo, por lo que el modo de operación de este y su función de transferencia permanecieron inalterados.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Se realizó una investigación sobre redes neuronales por medio de documentos, páginas web y videos en plataformas online, para tener un mejor entendimiento del tema y empezar a definir los puntos más importantes para tener en cuenta al momento de implementar este proyecto; en dicha búsqueda se encontró información relevante con respecto a diferentes funciones de activación que podrían ser usadas, además del riesgo del overfitting. En cuanto a las funciones de activación, se probaron diversas funciones, entre ellas la logsig, la ReLu, la softplus, la hardlim y la tansig, eligiendo al final la logsig ya que era la función que llegaba a un bajo margen de error en menor tiempo para este proceso en específico con nuestros propios datos de entrada. El problema del overfitting puede definirse como un exceso de complejidad de la red neuronal que puede resultar contraproducente para esta, algo que tuvo que tomarse en cuenta debido a la diferencia en la naturaleza de los datos necesarios para controlar la válvula y el motor del módulo.

Posteriormente se desarrolló la red neuronal completamente desde cero y la base de datos con la cual sería entrenada usando Matlab, y el programa de Arduino para realizar una comunicación serial entre la interfaz de usuario y el dispositivo que maneja directamente los sensores y actuadores del módulo. El programa de Matlab consta de 4 partes:

Control Neuronal: Esta es la parte principal, donde se configura y se entrena la red neuronal. Esta se realizó usando arreglos de celdas, de modo que fuera posible guardar matrices de diferentes tamaños en una sola variable. Cada fila de la variable red es un tipo de dato diferente, como los pesos de cada capa, los bias, los errores, etc.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01



Figura 18. Diagrama de flujo del archivo Control_Neural (Autoría propia).

En este programa, el usuario tiene la posibilidad de cargar una configuración anterior, o iniciar el entrenamiento de la red desde cero, en cuyo caso se utilizan valores aleatorios para inicializar la red neuronal. Acto seguido se carga la base de datos y se entra a un ciclo en el que se realiza el feedforward y el backpropagation hasta que se alcanza el número de épocas indicado o hasta que el error sea menor o igual al mínimo ingresado por el usuario. Una vez finalizado este ciclo, se presenta la opción de guardar la configuración y se llama al programa de operación, el cual se comunica directamente con el Arduino.

Feedforward: En este archivo se realizó la etapa de feedforward de la red, donde se pasan todos los elementos de la base de datos por la red neuronal, obteniendo una salida.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

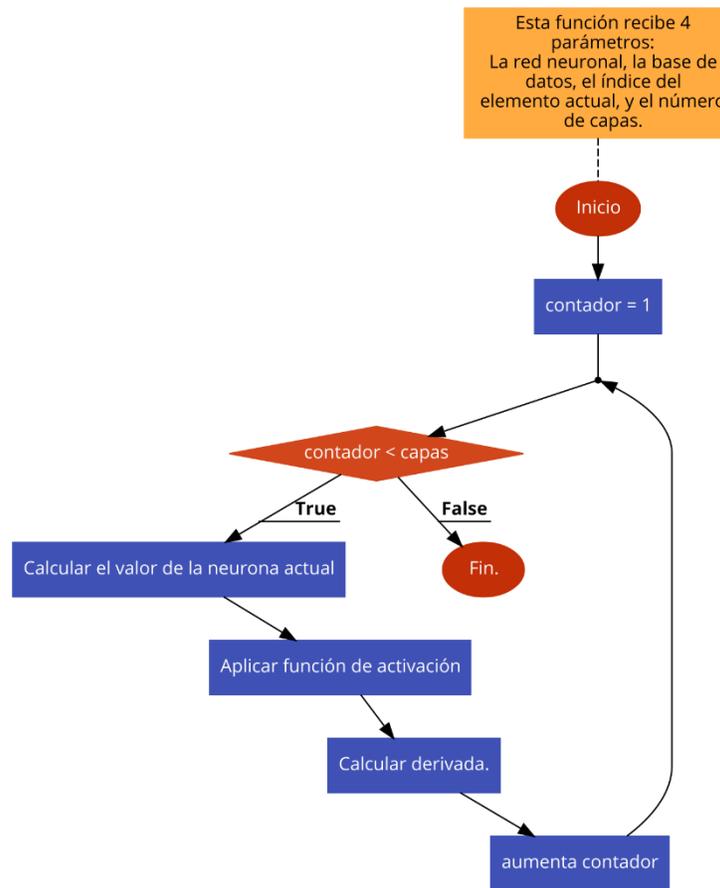


Figura 19. Diagrama de flujo de nuestra implementación del Feedforward (Autoría propia).

Backpropagation: El algoritmo de backpropagation que fue implementado para esta red fue el del descenso del gradiente, donde se busca minimizar el error calculando las derivadas parciales de la función de error o coste (Error medio cuadrado en este caso), en cuanto a los pesos de cada neurona, modificando estos últimos con el fin de disminuir el error en la salida.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

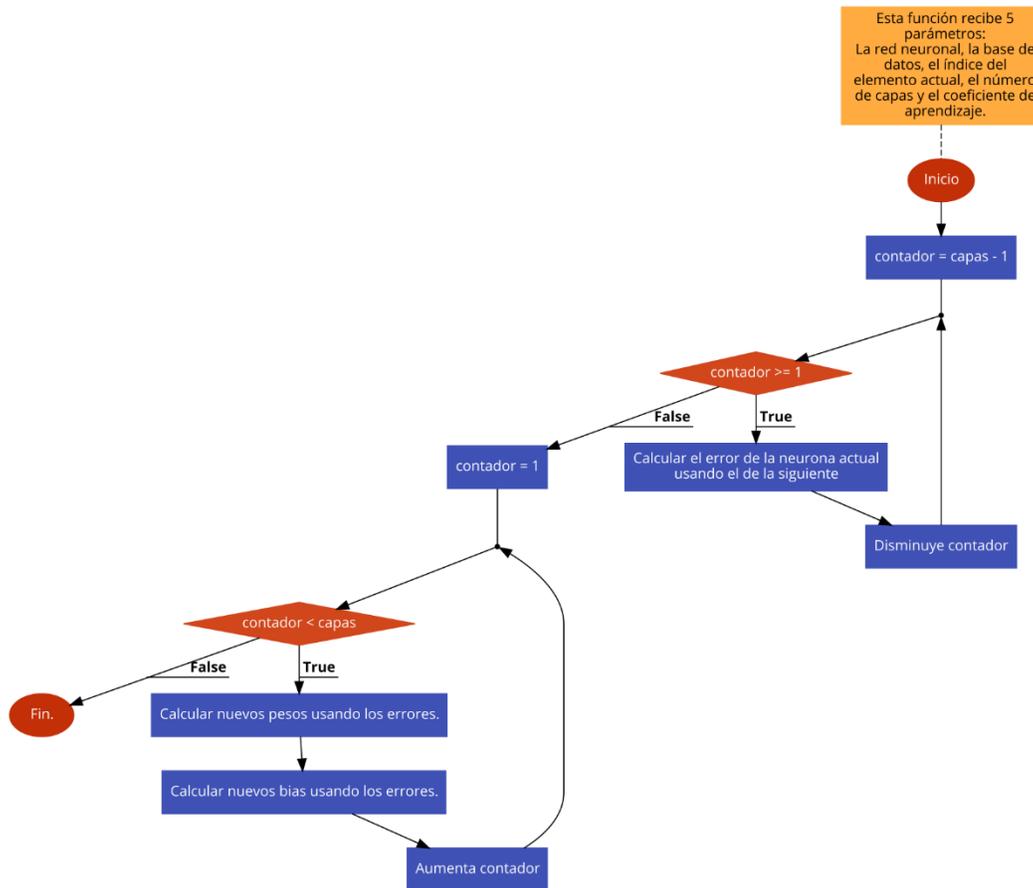


Figura 20. Diagrama de flujo de nuestra implementación del backpropagation (Autoría propia).

Operación: Esta parte del programa se encarga de realizar la comunicación entre el Arduino y Matlab por medio de comunicación serial una vez la red ha sido entrenada. Se trata de un ciclo en el que el Arduino le indica a Matlab el valor obtenido por el sensor de presión, el cual usa el programa de Matlab junto con el setpoint establecido como entradas de la red neuronal ya entrenada, y en base a la salida de la red, le indica al Arduino cómo debe proceder (activar o desactivar el motor, así como la válvula).

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

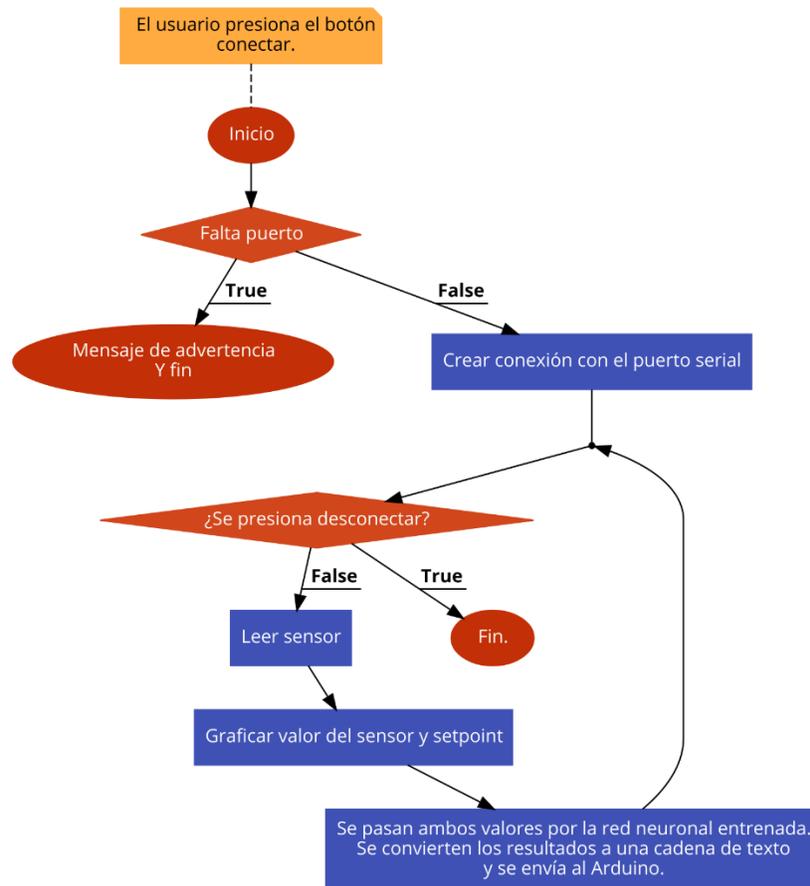


Figura 21. Diagrama de flujo del archivo operación (Autoría propia).

Se realizaron varias pruebas para verificar el aprendizaje de esta red y que los datos entregados por esta fueran coherentes incluso con parámetros que no había recibido en la etapa de entrenamiento, obteniendo valores más que aceptables y procediendo así a la etapa de implementación.

Una vez terminada la configuración de la red, se conectó esta con el módulo para llevar a cabo pruebas reales, y notando algunos problemas en cuanto a la comunicación del dispositivo Arduino con la interfaz de usuario, las cuales fueron solucionadas mediante código optimizando el método de recolección de datos enviados por el sensor al Arduino y de este a la interfaz de usuario.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Con esto, se pudo efectuar las pruebas necesarias y obtenerse los resultados que posteriormente serían comparados con el método de control convencional usado hasta el momento en el módulo.

A continuación, se explicará el funcionamiento del programa realizado, y se mostrará la interfaz de usuario.

Para ejecutar el programa, se cuenta con varias opciones, dependiendo de los elementos con los que cuente el usuario en su computador.

Si el usuario tiene Matlab instalado en su computador: Si se cuenta con Matlab versión 2015 o superior, basta con ejecutar el archivo “Control_Neuronal.exe” ubicado en la ruta “Control_Neuronal\Programa\Con Matlab”.

Si el usuario no tiene Matlab instalado en su computador: En este caso, debe dirigirse a la ruta “Control_Neuronal\Programa\Sin Matlab” donde encontrará 2 archivos:

Control_Neuronal_web.exe: Este archivo puede usarse para instalar el programa si se cuenta con conexión a internet, ya que se descargarán los recursos necesarios para su funcionamiento, haciendo el archivo más liviano y portable.

Control_Neuronal_mcr.exe: Este archivo es más pesado que el anterior, pero cuenta con la ventaja de no requerir internet para su instalación.

Para instalarlo solo es necesario oprimir la tecla “next” en cada pantalla que aparece, hasta que finalice la instalación.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Una vez abierto el programa, el usuario se encontrará con la pantalla inicial, en la cual se ingresan los parámetros para configurar la red neuronal deseada y se procede al entrenamiento de la red neuronal.

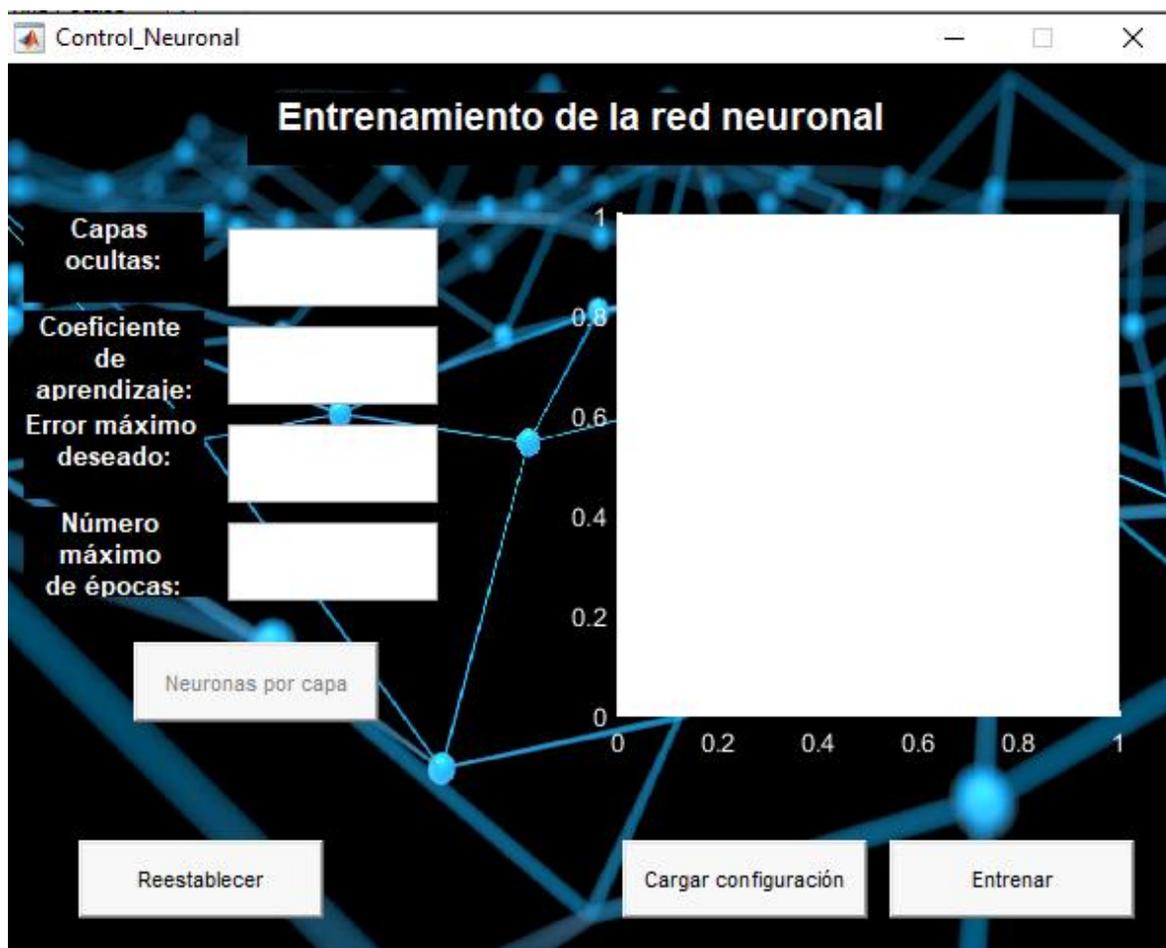


Figura 22. Pantalla inicial (Autoría propia).

Los parámetros que pueden definirse son:

Capas ocultas: Número de capas ocultas que el usuario desea para la red neuronal. Si se ingresa cero, el programa actuará como un perceptrón simple. Se recomienda el uso de pocas capas para evitar el problema del overfitting.

Coeficiente de aprendizaje: Valor que modificará la medida en que la red neuronal aprende. Se recomienda usar valores entre 0.01 y 0.1 para una mejor convergencia.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Error máximo deseado: Error permisible bajo el cual debe llegar a estar la red neuronal. Para el control del módulo se recomienda usar como valor 0.001, de modo que se obtenga un control muy preciso.

Número máximo de épocas: Número máximo de iteraciones que el programa realizará, se detendrá al llegar a este, aunque el error máximo no haya sido alcanzado. Se recomienda usar 100000.

Una vez ingresado el número de capas ocultas, se habilitará el botón “neuronas por capa” donde podrá indicarse cuántas neuronas se desea en cada capa. Este cuadro se repetirá si el usuario deja algún campo en blanco o ingresa algún valor no permitido (Como letras o números negativos). Se recomiendan valores inferiores a 200 para no sobrecargar los recursos computacionales.

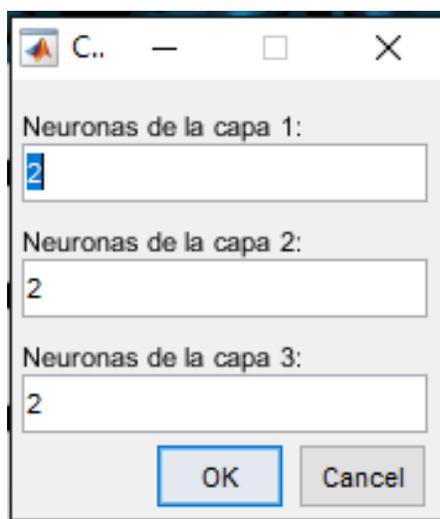


Figura 23. Selección de neuronas (Autoría propia).

Con todos los parámetros configurados, se puede pulsar el botón “entrenar”, el cual inmediatamente cambiará a “detener”, y en la gráfica podrá verse cómo varía el error con respecto al paso de las épocas.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

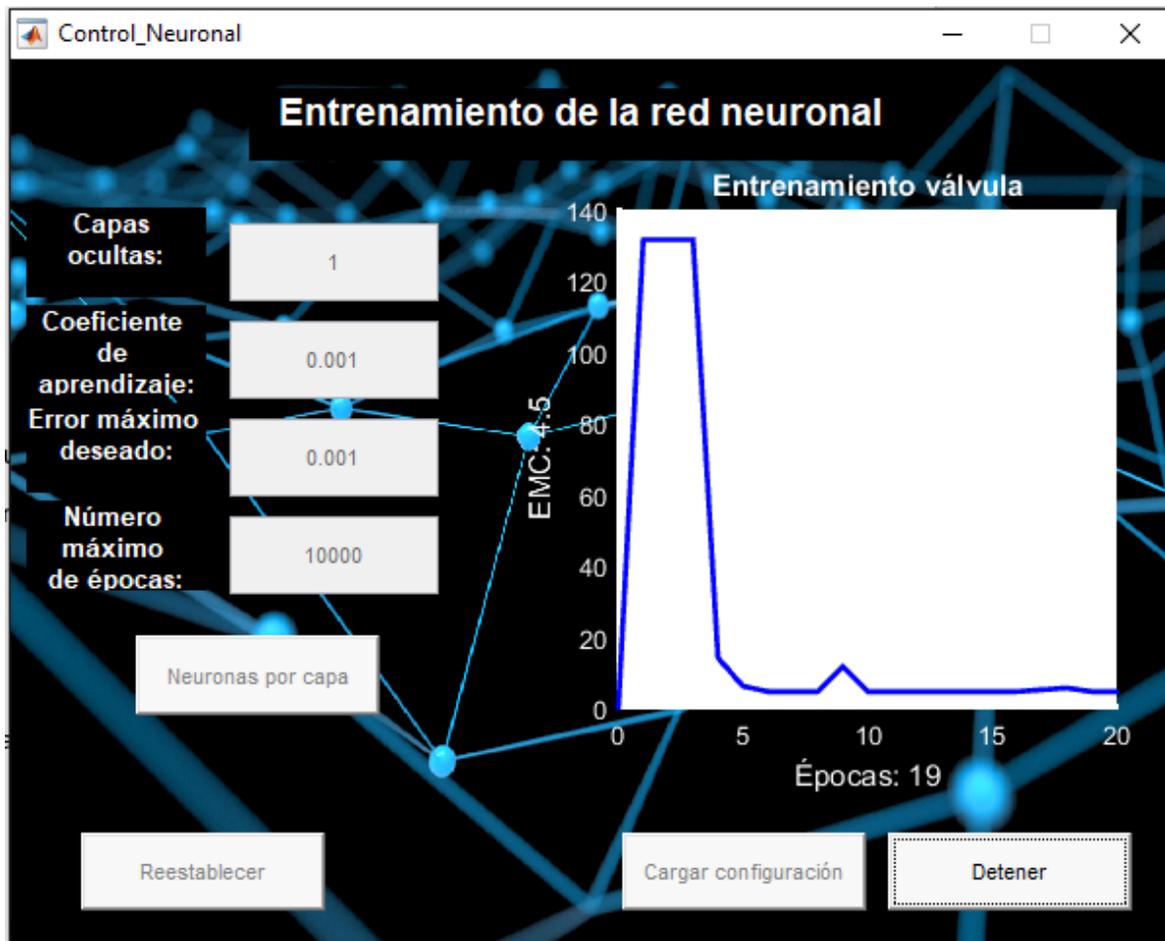


Figura 24. Vista de entrenamiento (Autoría propia).

Si se pulsa el botón detener, los parámetros del coeficiente de aprendizaje, error máximo deseado, y número máximo de épocas podrán ser modificados para continuar el entrenamiento desde donde se dejó, con estos nuevos valores, con lo cual pueden explorarse opciones como coeficientes de aprendizaje variables. Si se desea volver a empezar desde cero una red neuronal, solo debe pulsarse el botón de reestablecer, e ingresar nuevamente los valores deseados para configurar la nueva red.

La última opción que ofrece esta interfaz, es el botón de cargar configuración, con el cual puede cargarse al programa una red neuronal obtenida previamente, ya sea para continuar configurándola, o para trabajar directamente en ella.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

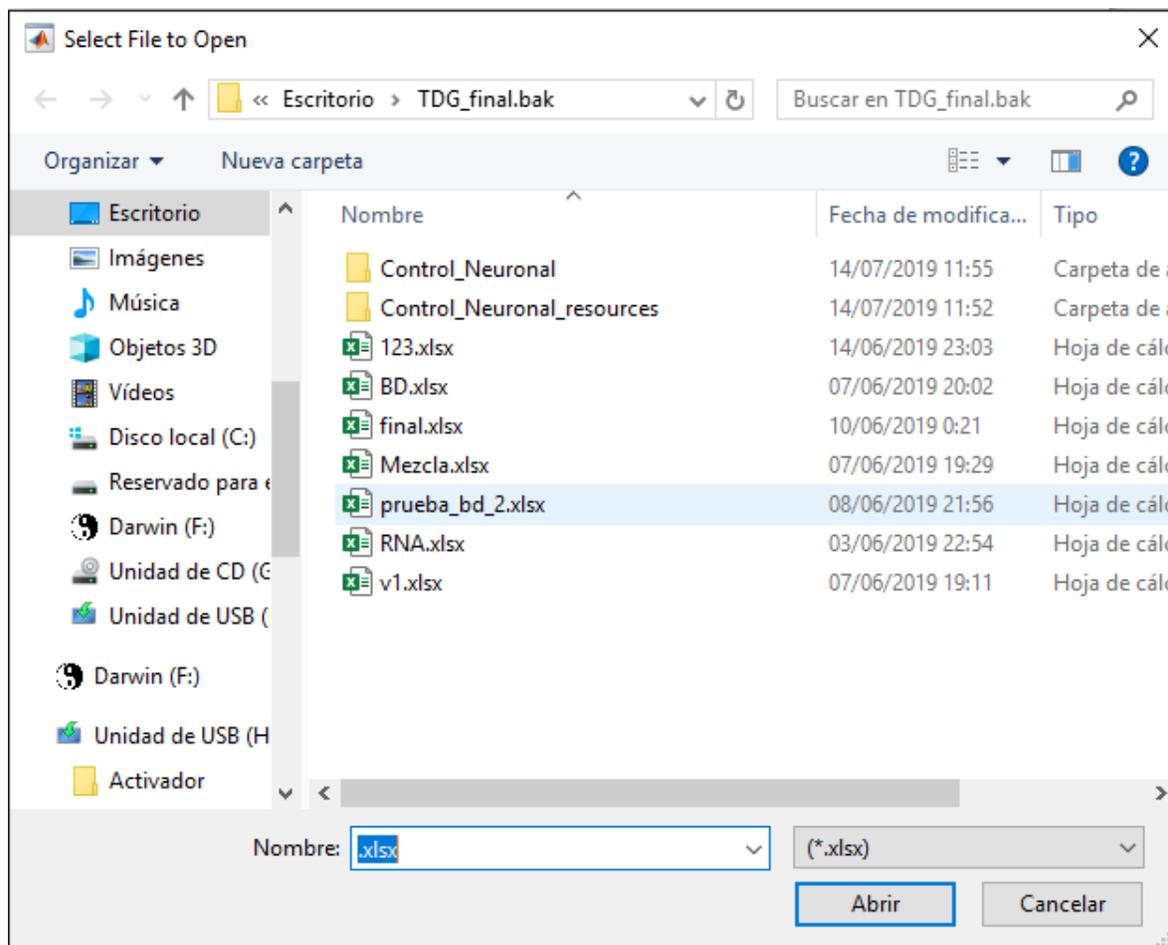


Figura 25. Pantalla de carga de archivo de configuración (Autoría propia).

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01



Figura 26. Vista del programa principal cargando la configuración seleccionada (Autoría propia).

Los valores del archivo de configuración se cargan en las variables `config` y `config2`, las cuales son usadas para llenar los arreglos de celdas `red` y `red2`, en los cuales cada fila corresponde a un tipo de dato específico, siendo la primera fila para el número de entradas y neuronas por capa, la segunda las matrices de pesos por capa, y la tercera los bias de las neuronas de cada capa. Estas variables pueden observarse en la siguiente figura:

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

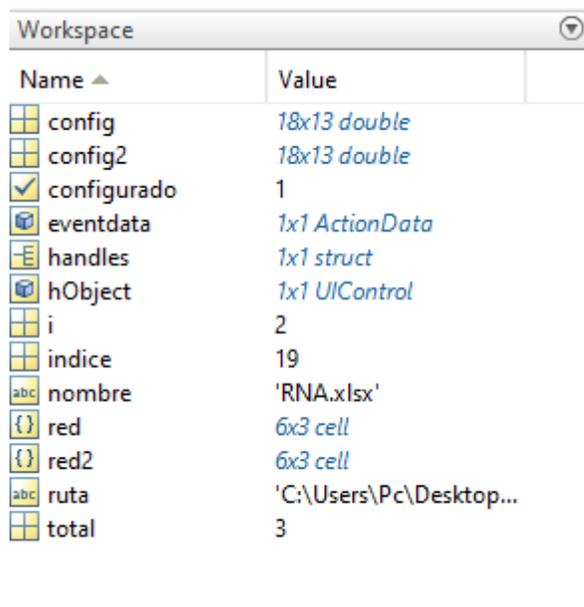


Figura 27. Workspace de Matlab después de la carga de archivos (Autoría propia).

Una vez finalizado el entrenamiento, el programa preguntará si se desea guardar la configuración para su posterior uso, y en caso de obtener una respuesta afirmativa, se abrirá un cuadro para elegir la ubicación y nombre para guardar la base de datos.

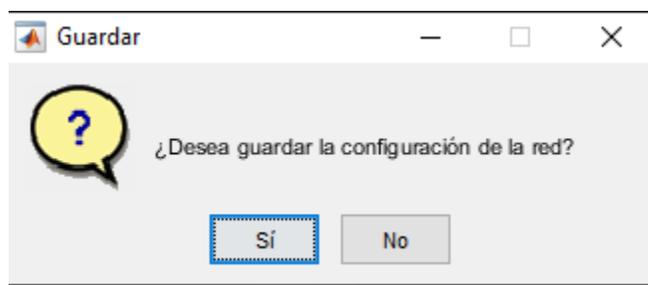


Figura 28. Mensaje de confirmación de guardado (Autoría propia).

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

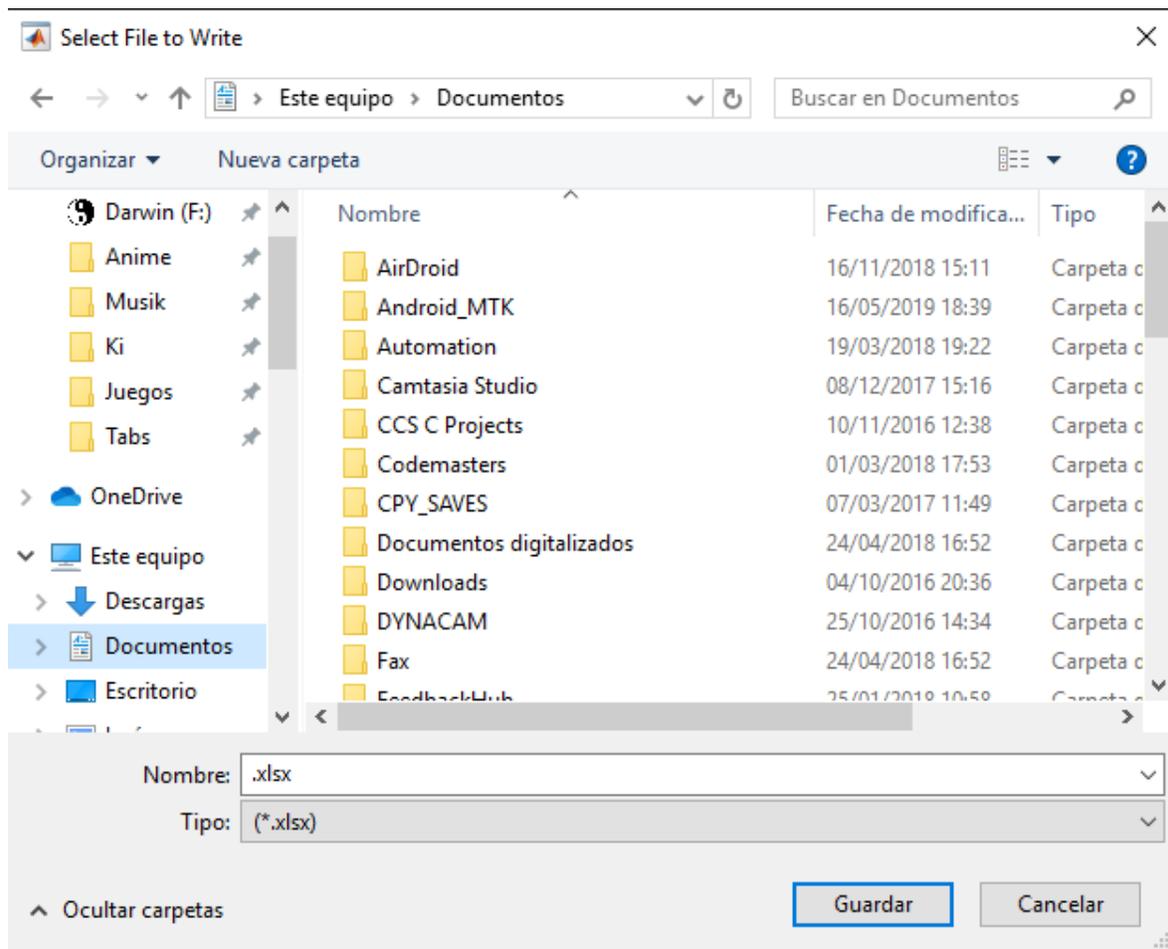


Figura 29. Pantalla de guardado de configuración (Autoría propia).

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01



Figura 30. Vista de la pantalla inicial guardando la configuración (Autoría propia).

2	13	1											
3,81228708	2,12306979	-0,14345656	0,37260452	3,36158708	-0,9117243	5,49558488	3,40044633	6,805778	6,92957103	6,2005226	-1,44556468	-1,69225712	
-3,67247517	-2,17700451	0,14232936	-0,8569451	-3,10387481	0,55799604	-4,83256529	-2,93734156	-6,19062168	-6,09371455	-5,33074988	1,45867154	0,87700834	
0,10425932													
0,01071889													
-0,24351526													
-0,10123381													
-0,11411431													
-0,32677066													
0,11352367													
-0,06307574													
0,26994931													
0,24988785													
0,16053076													
-0,38139616													
-0,3361939													
0,4800655	0,30720019	0,26595676	0,45187588	0,37854496	0,22734439	0,35379943	0,08967667	0,06999723	0,27433	-0,01017715	0,43806199	0,73401206	
-0,50317671													

Figura 31. Vista de los datos en el archivo de Excel donde se guarda la configuración de la red (Autoría propia).

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

En la figura 29 puede apreciarse el archivo de Excel una vez guardado. La primer fila corresponde al número de entradas y de neuronas por capa, en este caso 2 entradas, 13 neuronas en la capa oculta y 1 neurona de salida. Las siguientes filas corresponden a las matrices de pesos de cada capa, cuyas dimensiones dependerán de los valores de la primera fila de la base de datos, teniendo en este caso una matriz de $2 * 13$ (Entradas * Neuronas de la capa oculta) seguida de una matriz de $13 * 1$ (Neuronas de la capa oculta * salida). Al final se encuentran los vectores de Bias, dependiendo del número de neuronas en cada capa (Cada neurona tiene un Bias), teniendo en este caso un vector de 13 posiciones (Para la capa oculta), y uno de solo una posición al final del archivo (El Bias de la neurona de salida).

Este archivo de Excel contiene 2 hojas, la primera para la red de la válvula, y la segunda para la red del motor, las cuales siempre tendrán la misma estructura en cuanto a número de capas y neuronas.

Una vez terminado este proceso, se pasará automáticamente a la pantalla de operación, desde la cual se realizará el control del módulo con la red entrenada.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

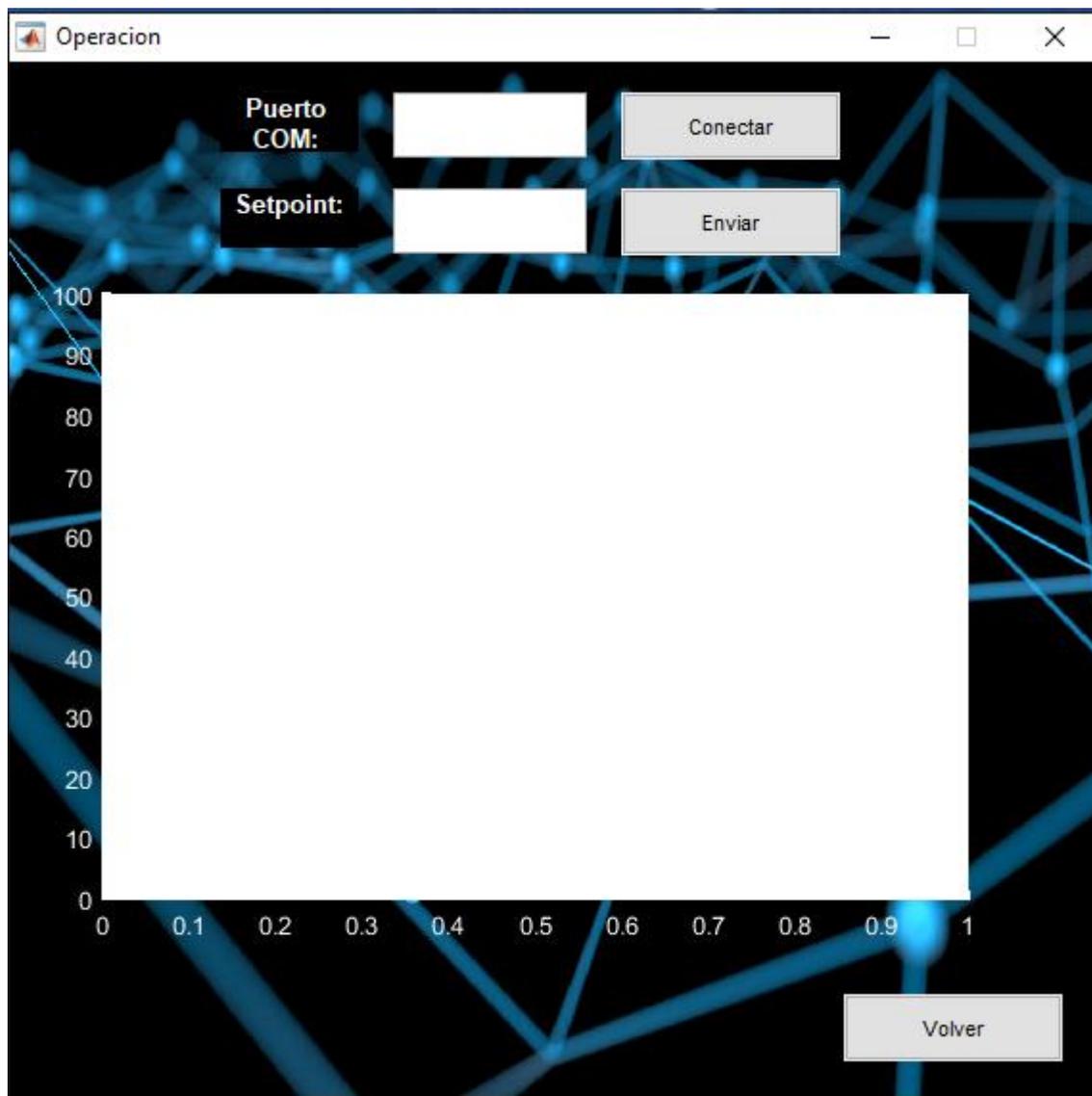


Figura 32. Pantalla de operación (Autoría propia).

Esta pantalla solo contiene 2 campos:

Puerto COM: Puerto en el cual está conectado el Arduino. Este puede verse en el administrador de dispositivos de Windows, en la sección de puertos COM, buscando el que tenga la palabra Arduino.

Una vez ingresado el puerto, se debe presionar el botón “conectar”, lo que iniciará la comunicación entre el Arduino y la interfaz de usuario, mostrando en tiempo real en la gráfica

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

el valor medido por el sensor de presión. El texto de dicho botón cambiará a “desconectar”, con el cual podrá terminarse la conexión con el módulo.

Setpoint: Porcentaje de presión a la que se desea llevar al sistema. (Si se desea por ejemplo un 10% se debe ingresar 10.)

Cada que se cambie el valor del setpoint debe presionarse el botón enviar para que la instrucción sea enviada al Arduino y el control sea realizado.

El botón “volver” sirve para volver a la pantalla de entrenamiento.

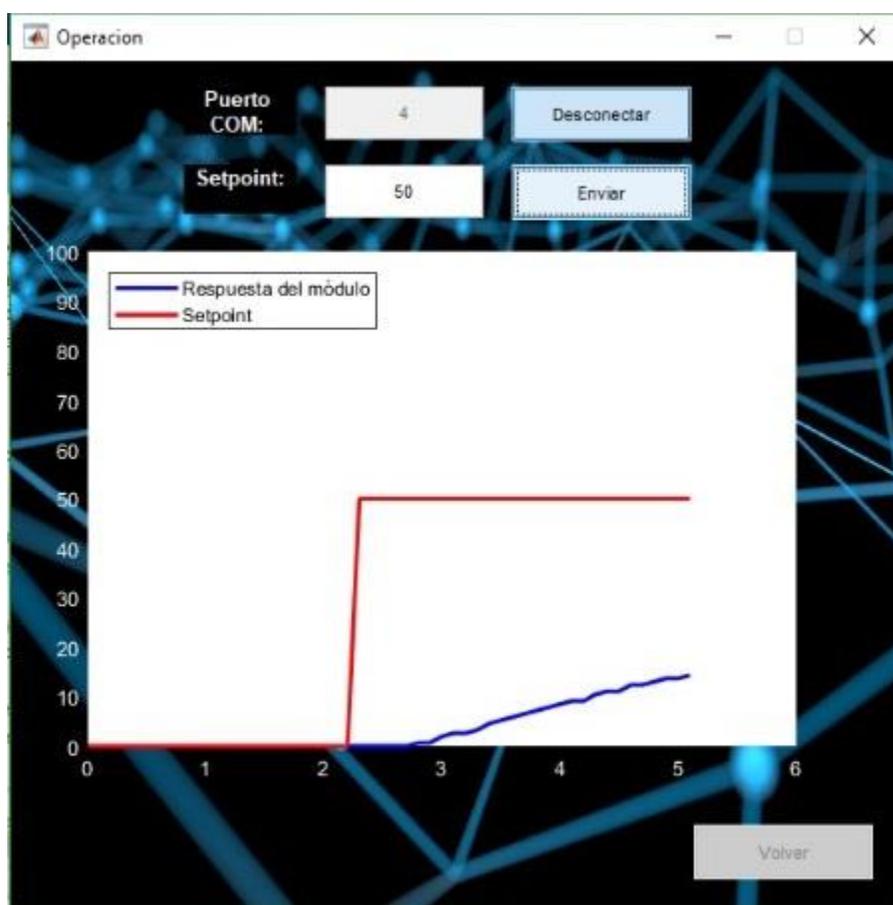


Figura 33. Pantalla de operación en comunicación con el módulo (Autoría propia).

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Mensajes de advertencia y error

El programa puede presentar diversos mensajes de error, ocasionados por un uso inadecuado del programa, y usados para prevenir el mal funcionamiento de este, indicando al usuario la acción que ocasionó dicho error. Estos mensajes son los siguientes:

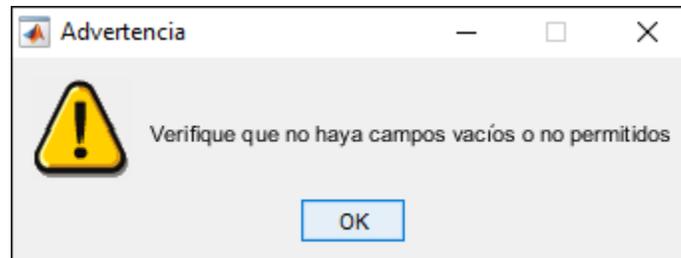


Figura 34. Mensaje de error de campos vacíos o no permitidos (Autoría propia).

Este error aparece en la pantalla inicial cuando el usuario deja algún campo vacío o ingresa algún valor no permitido (como letras o números negativos). Hay que tener en cuenta que solo se puede ingresar el número cero (0) en el campo de capas (si se desea que el programa actúe como un perceptrón simple). También aparece este mensaje si no se eligen las neuronas para las capas indicadas.

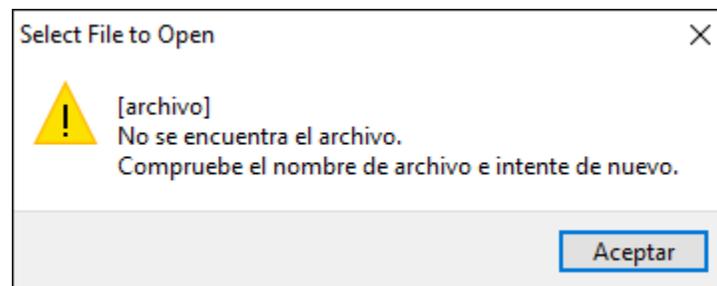


Figura 35. Mensaje de error de archivo inválido (Autoría propia).

Este mensaje aparece cuando se elige un nombre de archivo inválido al momento de tratar de cargar una configuración.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

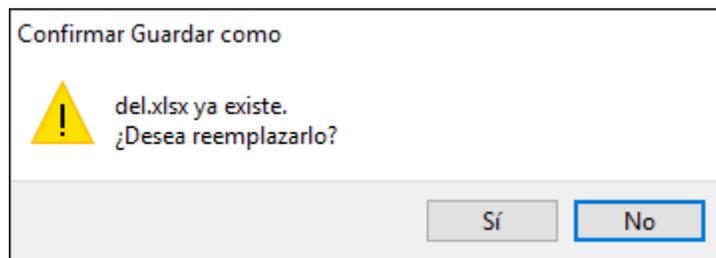


Figura 36. Mensaje de advertencia de sobreescritura de archivo existente (Autoría propia).

Este mensaje aparece cuando se intenta guardar una configuración con un nombre de archivo existente en la carpeta seleccionada. En este caso, el usuario puede elegir si reemplazar el archivo, usar otro nombre, o cancelar el guardado (Seleccionando “No” y después “Cancelar”).

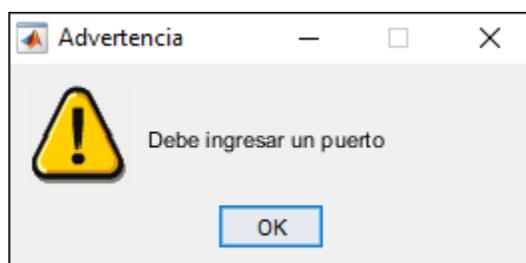


Figura 37. Mensaje de error de puerto no ingresado (Autoría propia).

Este mensaje aparece en la pantalla de control, cuando no se ingresa un puerto en el campo destinado para este fin y se oprime el botón “conectar”.

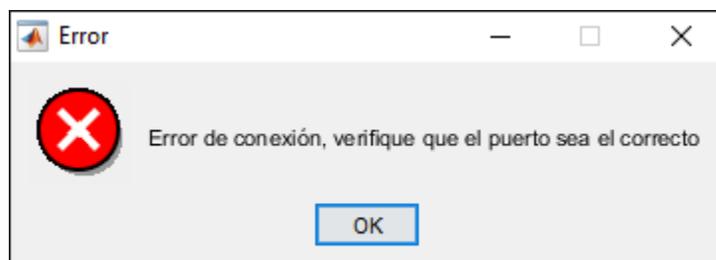


Figura 38. Mensaje de error de puerto equivocado (Autoría propia).

Este mensaje se muestra en pantalla cuando se ingresa un puerto equivocado en el campo de puerto.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

5. RESULTADOS Y DISCUSIÓN

El esquema de control con la red neuronal quedó de la siguiente manera:

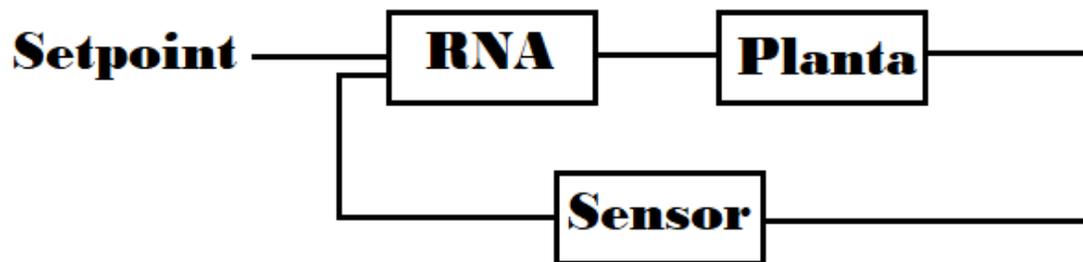


Figura 39. Esquema de control con la RNA (Autoría propia).

O de un modo físico:

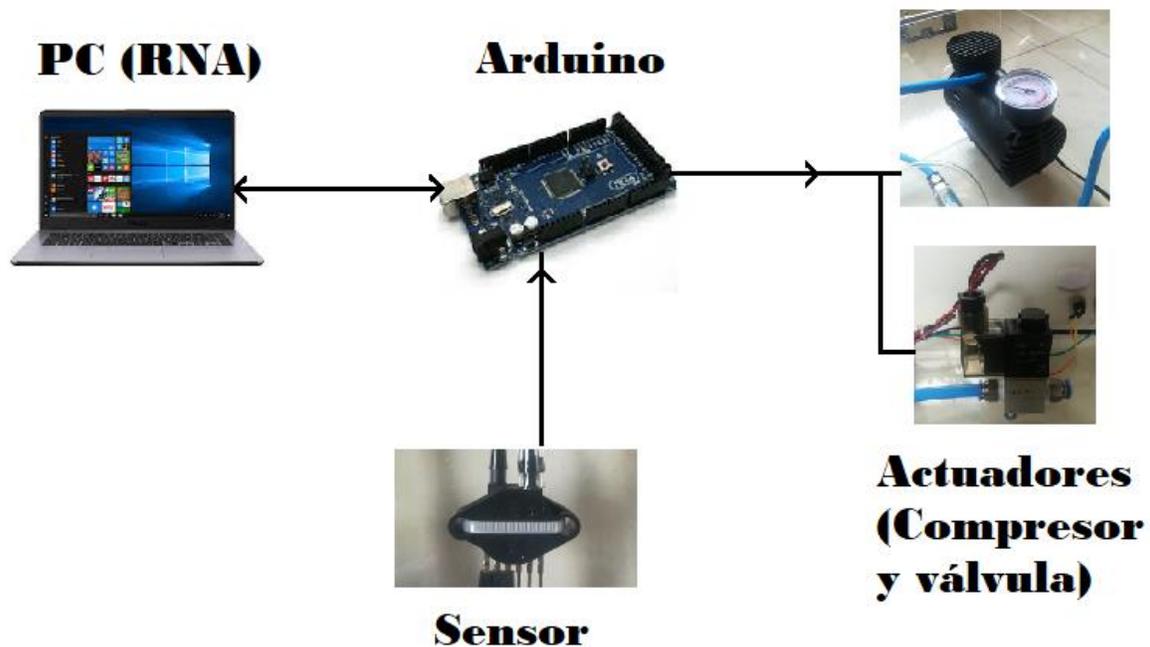
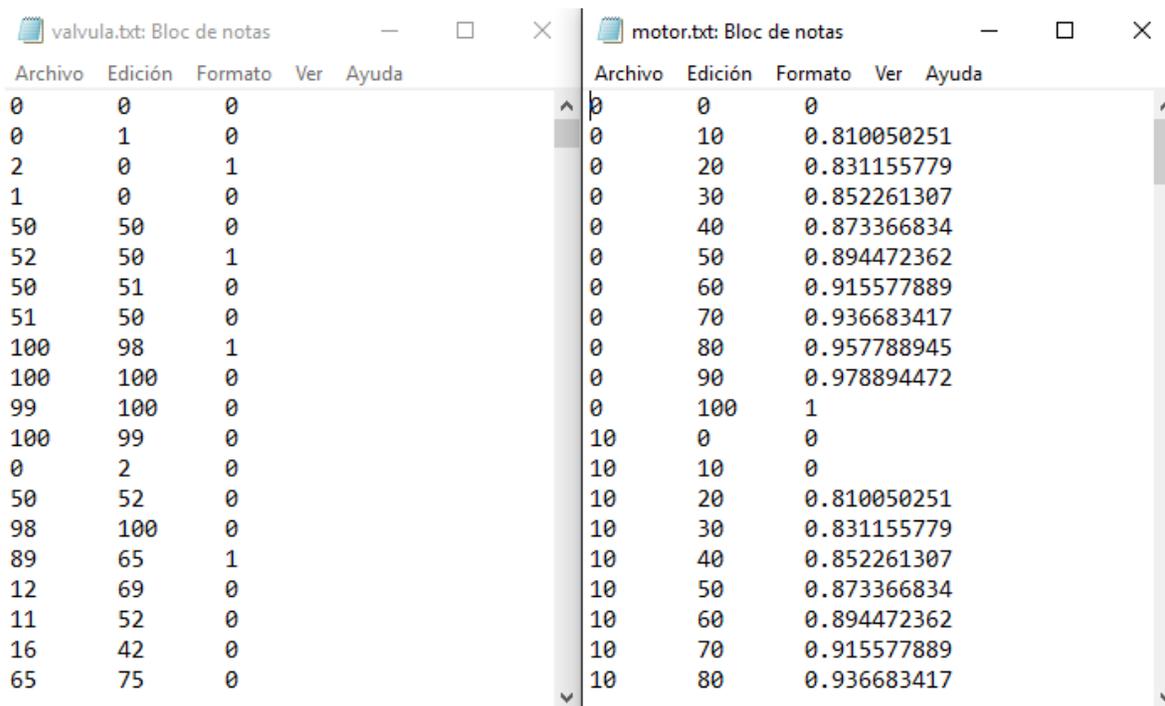


Figura 40. Esquema de comunicación entre los dispositivos (Autoría propia).

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

La red neuronal se entrenó usando las bases de datos motor.txt y válvula.txt, ambas cuentan con 500 filas de datos, siendo la primera columna el valor del sensor, la segunda el setpoint y la tercera el valor al que debería ponerse la válvula o el motor con esa combinación de datos entre 0 (0%) y 1 (100%).



valvula.txt: Bloc de notas				motor.txt: Bloc de notas			
Archivo	Edición	Formato	Ver Ayuda	Archivo	Edición	Formato	Ver Ayuda
0	0	0		0	0	0	
0	1	0		0	10	0.810050251	
2	0	1		0	20	0.831155779	
1	0	0		0	30	0.852261307	
50	50	0		0	40	0.873366834	
52	50	1		0	50	0.894472362	
50	51	0		0	60	0.915577889	
51	50	0		0	70	0.936683417	
100	98	1		0	80	0.957788945	
100	100	0		0	90	0.978894472	
99	100	0		0	100	1	
100	99	0		10	0	0	
0	2	0		10	10	0	
50	52	0		10	20	0.810050251	
98	100	0		10	30	0.831155779	
89	65	1		10	40	0.852261307	
12	69	0		10	50	0.873366834	
11	52	0		10	60	0.894472362	
16	42	0		10	70	0.915577889	
65	75	0		10	80	0.936683417	

Figura 41. Bases de datos usadas para entrenar la red (Autoría propia).

Estas bases de datos se usaron para entrenar la red con diferentes configuraciones, seleccionando al final la más rápida.

Después de obtener la red neuronal entrenada e implementarla al módulo, se realizaron diversas pruebas comparando el programa de control convencional con el de control inteligente, obteniendo los siguientes resultados:

Primero se ingresó a ambos programas un setpoint del 20%:

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

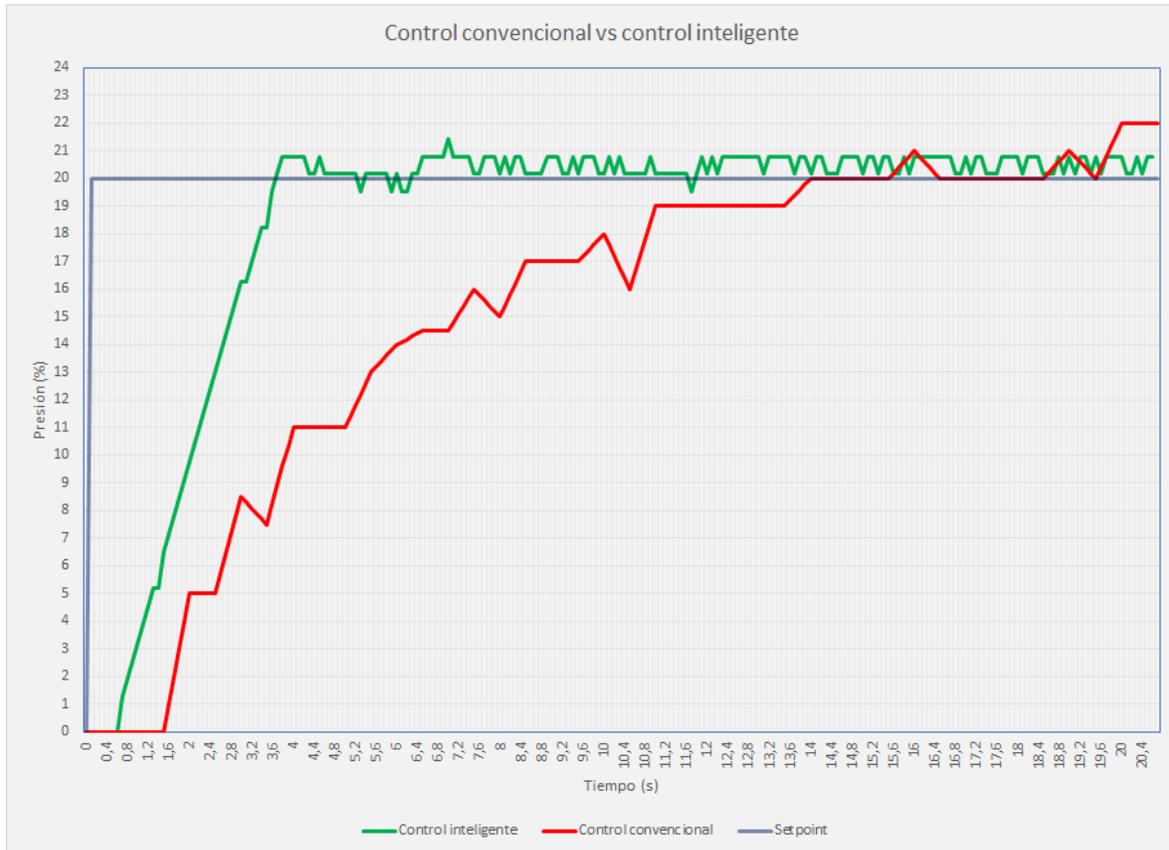


Figura 42. Resultado de la comparación entre ambas técnicas de control (Autoría propia).

Puede observarse que el control inteligente (en verde) presenta un mejor tiempo de respuesta que el control convencional (en rojo), además de llegar al punto indicado en un tiempo mucho menor (3.6 s vs 14 s).

Teniendo en cuenta esto, se realizaron otras comparaciones usando los programas implementados para cada tipo de control, la primera de ellas fue una escalera, aumentando 20% cada vez en ambos programas hasta llegar al 100%, y disminuyendo de nuevo 20% hasta llegar a cero:

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

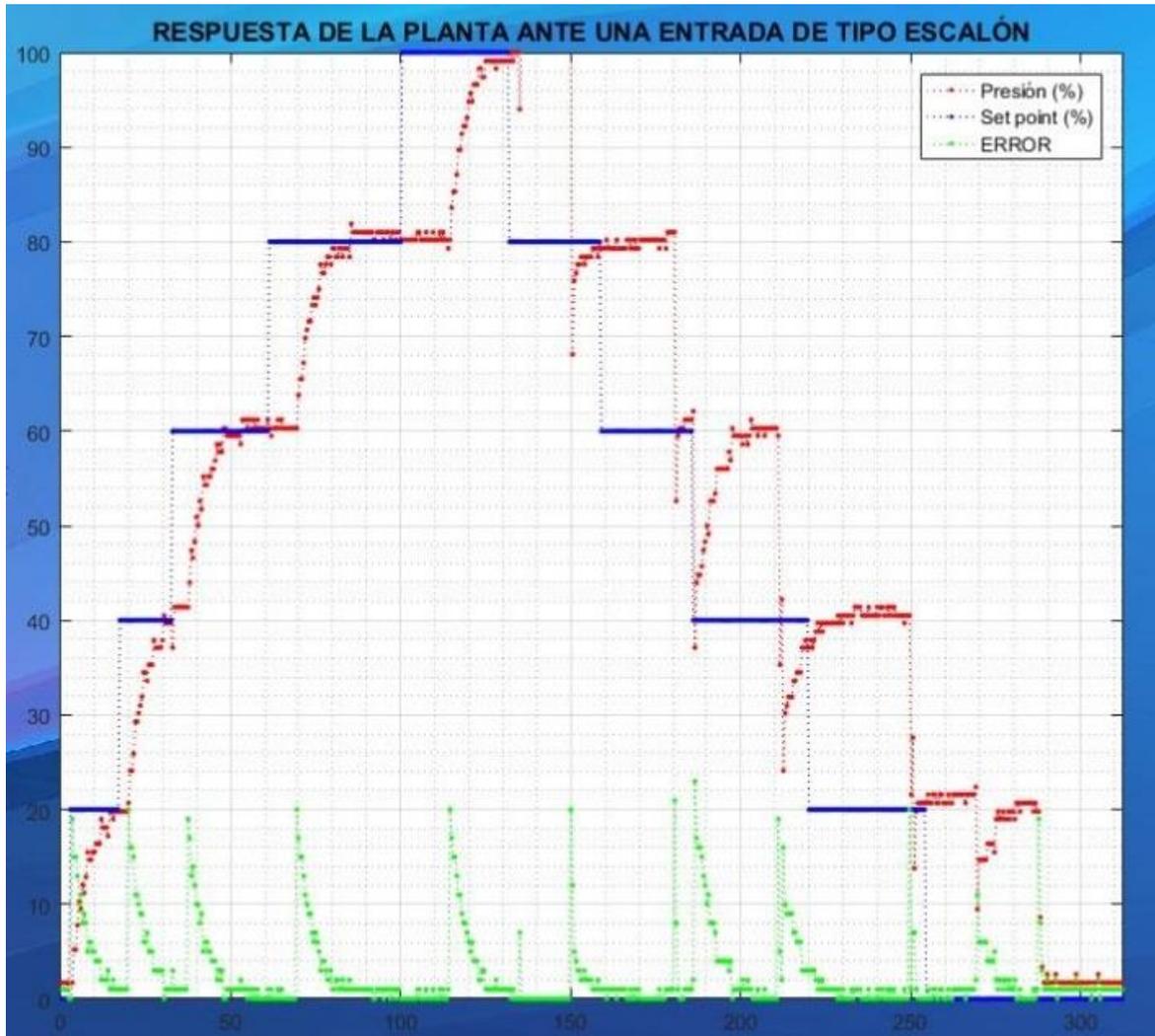


Figura 43. Respuesta del módulo a una serie de escalones del 20% con control convencional (Autoría propia).

Puede observarse que se iniciaron los escalones desde el principio (segundo cero) y se aumentaban cada que el programa llegaba al setpoint establecido, tardando en realizar todo el recorrido 290 segundos

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

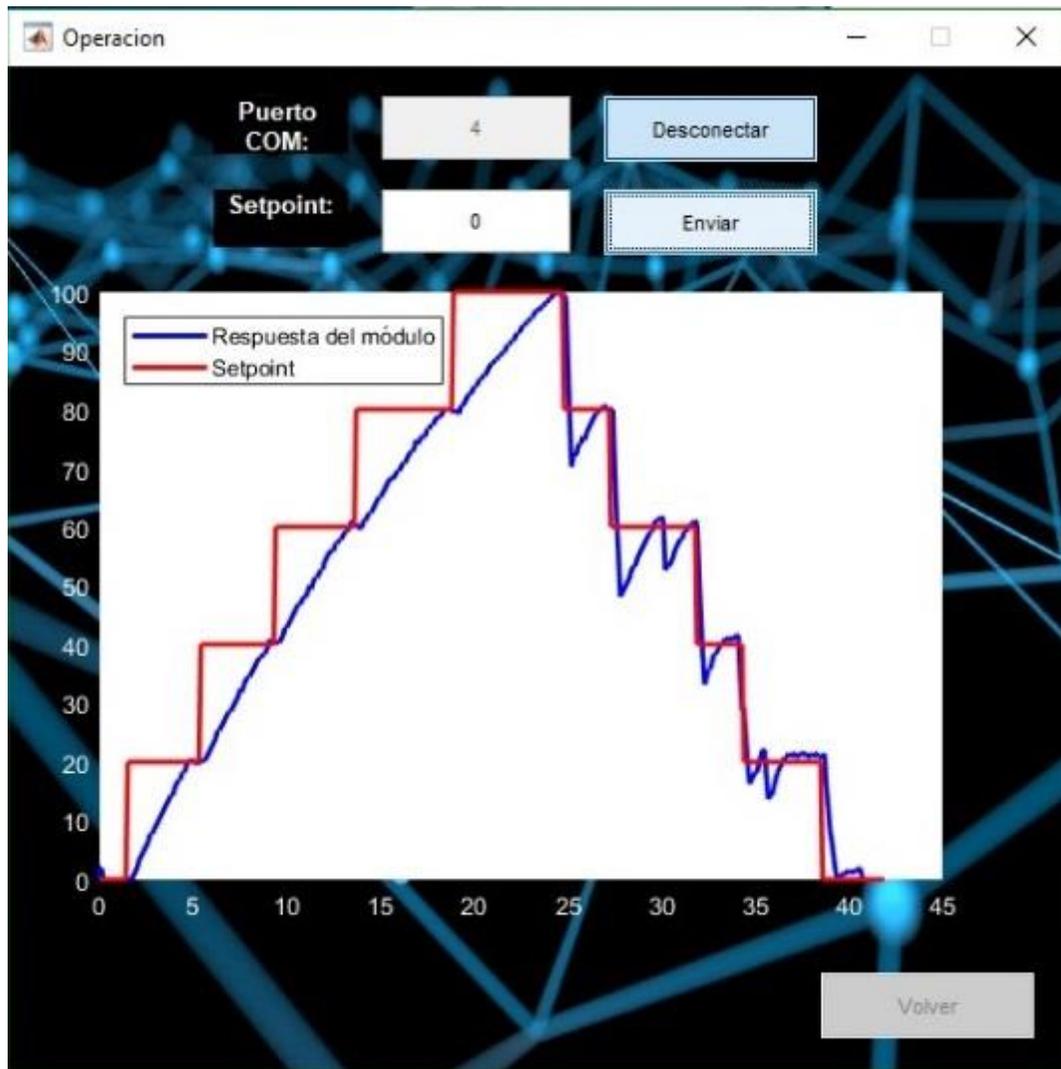


Figura 44. Respuesta del módulo a una serie de escalones del 20% con control inteligente (Autoría propia).

Con el control inteligente se inició a los 2 segundos, terminando de completar todo el proceso a los 40 segundos, obteniendo así un tiempo total de 38 segundos, siendo en esta prueba 7.6 veces más rápido que el control convencional.

Acto seguido, se realizó una prueba a ambos programas para observar su comportamiento ante una perturbación de la válvula una vez están en estado estable:

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

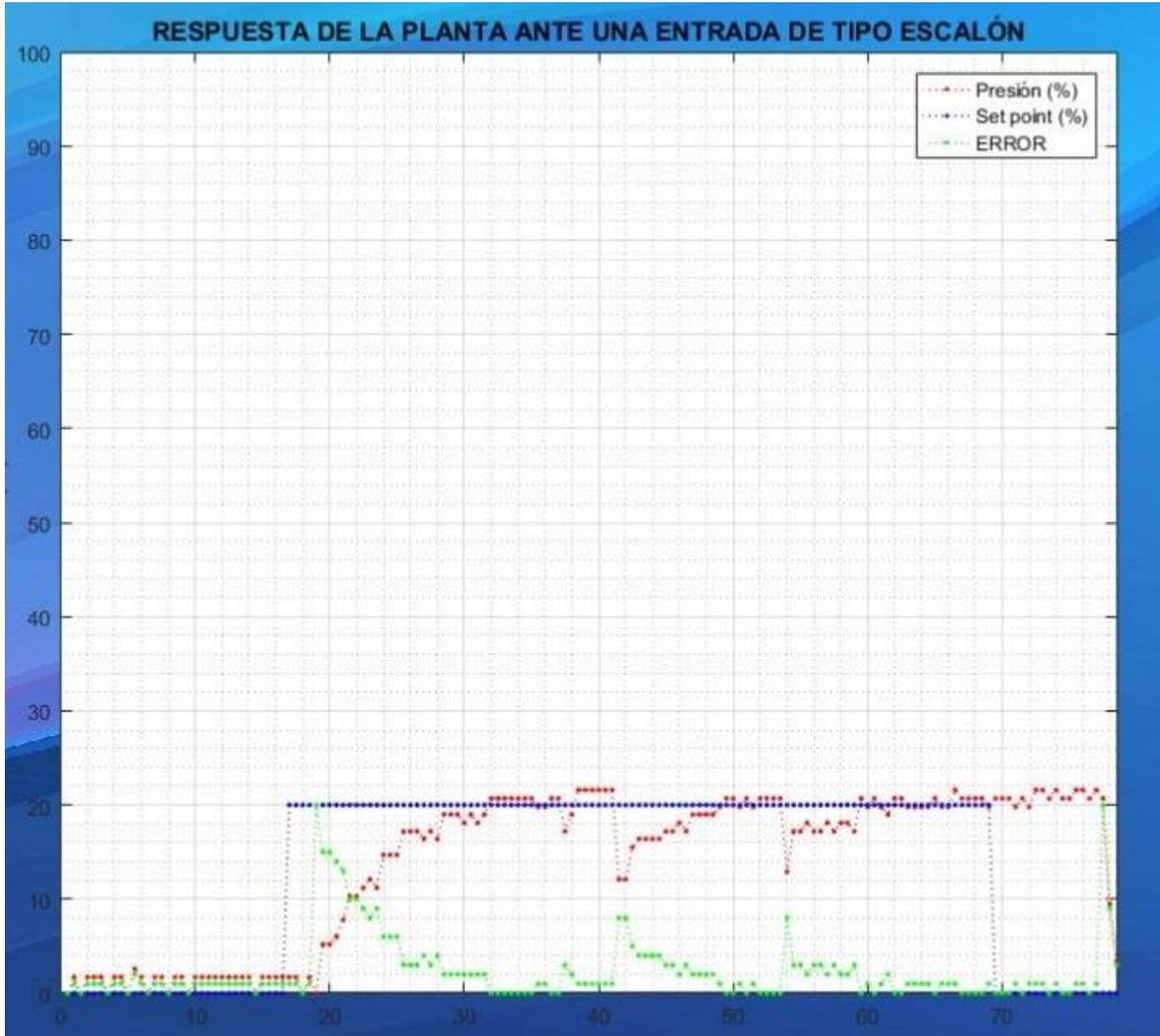


Figura 45. Respuesta del módulo a perturbaciones con control convencional (Autoría propia).

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

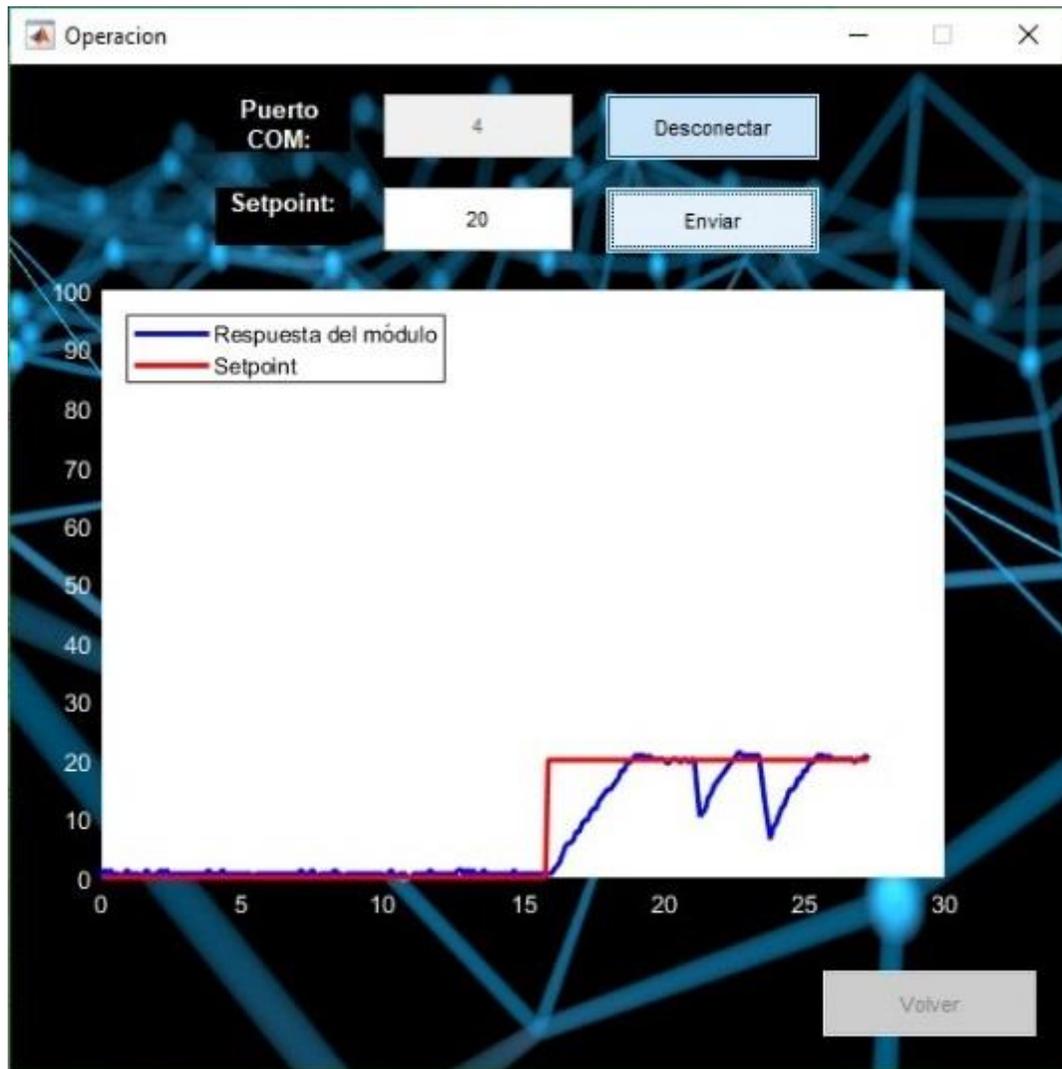


Figura 46. Respuesta del módulo a perturbaciones con control inteligente (Autoría propia).

Se puede observar que ambos programas manejan correctamente estas perturbaciones retornando rápidamente al nivel requerido, sin embargo, el programa de control neuronal logra su cometido en la mitad del tiempo de lo que le toma al programa de control convencional.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

6. CONCLUSIONES, RECOMENDACIONES Y TRABAJO A FUTURO

6.1 Conclusiones

- Se logró controlar por medio de redes neuronales el módulo didáctico de presión para comparar el control convencional con el control inteligente. Por consiguiente, se identificó que este tipo de técnicas son más eficientes que las usadas actualmente en el módulo.
- Se implementaron técnicas de control inteligentes aplicadas en procesos reales, realizando una investigación sobre las metodologías que están siendo estudiadas actualmente, garantizando que el manejo de estas técnicas es más competente, ya que de esta manera se obtienen los resultados esperados en un mejor tiempo.
- Se presentó una disminución significativa del error en los resultados obtenidos comparando los datos del control convencional contra los datos del control inteligente y su vez se evidencia un mejor tiempo de respuesta en este último.
- Teniendo en cuenta los resultados obtenidos, no es de extrañar que las redes neuronales sean el pilar de la llamada cuarta revolución industrial, demostrando ser útiles en múltiples campos, ofreciendo alta eficiencia y confiabilidad.

	<p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p>	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

6.2 Recomendaciones

- Se recomienda a todos los estudiantes y profesores leer previamente este informe para conocer la correcta operación de este módulo.
- Fomentar el aprendizaje de las redes neuronales y sus respectivas técnicas, partiendo en la asignatura de sistemas inteligentes y sus concernientes áreas afines.
- Se recomienda a todos los estudiantes tener ciertas nociones básicas acerca de las redes neuronales para su adecuado uso y mejor aprendizaje.

6.3 Trabajo futuro

- Implementación de otras técnicas de inteligencia artificial, como por ejemplo la lógica difusa, o una mezcla de técnicas como algoritmos genéticos con redes neuronales.
- Cambiar la válvula por una que pueda abrirse y cerrarse parcialmente, para poder llevar a cabo un control más preciso y eficiente.
- Aumentar las variables de la red, usando valores como los estados anteriores u otros que puedan influir en el proceso.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Bibliografía

- Alberto Delgado. (1998). *Redes Neuronales*. Obtenido de <http://disi.unal.edu.co/~lctorress/RedNeu/LiRna008.pdf>
- Alegsa. (2016). *Definición de Comunicación serial*. Obtenido de http://www.alegsa.com.ar/Dic/comunicacion_serial.php
- Baron Gomez, R., & Ramos Velasco, L. (2017). *Wavelet neural network PID controller for a UAS transporting a cable-suspended load*.
- Calvo, D. (2017). *Definición de red neuronal artificial*. Obtenido de <http://www.diegocalvo.es/definicion-de-red-neuronal/>
- Domínguez, C. R. (2019). *Ingeniería Informática*. Obtenido de <ftp://decsai.ugr.es/pub/usuarios/castro/Actividades/Redes-Neuronales/Faqs/faq.pdf>
- Douglas, C. (2017). Obtenido de <https://controlguru.com/the-components-of-a-control-loop/>
- Gomez, G. A., Henao, J. C., & Salazar, H. (2004). Entrenamiento De Una Red Neuronal Artificial Usando El Algoritmo Simulated Annealing. Obtenido de <https://dialnet.unirioja.es/descarga/articulo/4844874.pdf>
- Gonzales, L. G. (2010). *Control por medio de redes neuronales sistemas avanzados de control*.
- Guerrero, D. O. (2008). Manual de interfaz Grafica de Usuario en Matlab.
- HBM. (2019). Obtenido de <https://www.hbm.com/es/7646/que-es-un-sensor-de-presion/>
- Hu, B., Zhao, Z., & Liang, J. (2012). *Multi-loop nonlinear internal model controller design under nonlinear dynamic PLS framework using ARX-neural network model*.
- Huang, Y., Li, S., & Sun, J. (2019). *Mars entry fault-tolerant control via neural network and structure adaptive model inversion*.
- Instituto Tecnológico De Ciudad Madero . (2002). Matlab e Interfaces Gráficas .
- Llinares, A. N. (06 de 2005). *Sistemas Embebidos*. Obtenido de <http://server-die.alc.upv.es/asignaturas/paees/2005-06/a07%20-%20sistemas%20embebidos.pdf>
- Matich, D. J. (2001). *Redes neuronales conceptos basicos y aplicaciones*.
- Mattarollo, Y. (2014). Obtenido de <https://www.altecdust.com/blog/item/32-como-funcionan-las-electrovalvulas-o-valvulas-solenoides-de-uso-general>

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

Nicolás, A. S. (2009). *Neumática Práctica*.

Parker Hannifin Corporation. (2003). *Tecnología Neumática Industrial*.

Parker Hannifin Corporation. (2003). *Tecnología Neumática Industrial*.

Parlow, E. (2008). Obtenido de meteoblue Presion del Aire:
<https://content.meteoblue.com/es/meteoscool/el-tiempo/presion-del-aire>

Raffino, M. E. (24 de Febrero de 2019). *Fuente de Alimentacion*. Obtenido de
https://concepto.de/fuente-de-alimentacion/#Que_es_Fuente_de_Alimentacion

Valenzuela, Alexandro. (2017). *Sistemas De Control*. Recuperado el 16 de 06 de 2019, de
https://www.academia.edu/38597820/UNIDAD_2_SISTEMAS_DE_CONTROL_2.1.-INTRODUCCI%C3%93N_2.2.-CONTROL_ELECTROMECC%C3%81NICO_2.3.-CONTROL_ELECTR%C3%93NICO_2.4.-CONTROL_POR_ORDENADOR_2.1.-INTRODUCCI%C3%93N_1.1._Qu%C3%A9_es_el_control_autom%C3%A1tico

Solé, A. C. (2007). *Neumática e Hidráulica*.

wikipedia. (2015). *matlab lenguaje de programacion*. Obtenido de
<https://es.m.wikipedia.org/wiki/MATLAB>

Wikipedia. Redes de Neuronas Artificiales. (2012). *Perceptrón multicapa*.

Carlos, Juan & Jauregui, Jaimes & Eduardo, Oscar & Guerrero, Gualdrón & Diaz Rodriguez, Jorge. (2019). Desarrollo de un modelo de regresión con redes neuronales artificiales para estimar la resistencia rotórica de un motor de inducción.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	01
		Fecha	2019-09-01

FIRMA ESTUDIANTES Didier Garcia
Darwin Soto

FIRMA ASESOR OSWALDO YARATA CORRES

FECHA ENTREGA: 03/12/2019

FIRMA COMITÉ TRABAJO DE GRADO DE LA FACULTAD _____

RECHAZADO___ ACEPTADO___ ACEPTADO CON MODIFICACIONES___

ACTA NO. _____
FECHA ENTREGA: _____

FIRMA CONSEJO DE FACULTAD _____

ACTA NO. _____
FECHA ENTREGA: _____