

# **DISEÑO DE UN CONTROLADOR BASADO EN IA PARA UNA BANDA TRANSPORTADORA COMO ALTERNATIVA PARA EL MANEJO DE PAQUETERÍAS**

## **DESING OF AN AI-BASED CONTROLLER FOR CONVEYOR-BELT AS AN ALTERNATIVE TO PACKAGE DISTRIBUTION**

Jonathan Sánchez Romero, José Gustavo Leyva Retureta, Yazmín Rivera Peña, Fernando Aldana Franco

### **RESUMEN**

Los problemas de planificación en la producción incluyen estrategias contrastantes entre el almacenamiento, distribución de paquetería y la optimización de la producción. Para ello, la Industria 4.0 en combinación con la inteligencia artificial (IA) permiten controlar sistemas productivos de valor añadido al acelerar la integración de las tecnologías de fabricación y comunicación de información, uniendo la informática, la comunicación y el control. Así se diseñó un controlador inteligente que regula la velocidad angular de un motor de corriente directa. Se utilizaron como entradas el error y su derivada. Se compararon dos arquitecturas de red neuronal artificial. La primera fue entrenada mediante un algoritmo supervisado. La segunda fue optimizada mediante un algoritmo no supervisado construido con características mínimas e implementado en un entorno real. Ambas estructuras se probaron en una planta física para encontrar la que mejor regula la velocidad angular del motor con y sin perturbaciones externas. Se construyó un sistema basado en una tarjeta electrónica de desarrollo y un sensor infrarrojo como mecanismo de retroalimentación. El desempeño de la red neuronal no supervisada muestra mejor capacidad de adaptación al sistema.

**PALABRAS CLAVE:** inteligencia artificial, industria 4.0, control inteligente, redes neuronales artificiales, sistemas productivos

### **ABSTRACT**

Production planning problems include contrasting strategies between storage, parcel distribution and production optimization. Industry 4.0 in combination with Artificial Intelligence allow the control of value-added production systems by accelerating the integration of manufacturing and information communication technologies, linking information technology, communication, and control. Thus, an intelligent controller was designed to regulate the angular speed of a direct current motor. The error and its derivative were used as inputs. Two artificial neural network architectures were compared. The first ANN was trained using a supervised algorithm. The second was optimized using an unsupervised algorithm built with minimal features and implemented in a real environment. Both structures were tested in a physical plant to find the one that best regulates the angular speed of the motor with and without external disturbances. A system based on an electronic development system and an infrared sensor as a feedback mechanism was built. The performance of the unsupervised neural network shows a better capacity to adapt to the system.

**KEYWORDS:** artificial intelligence, industry 4.0, intelligent control, artificial neural networks, productive systems

## 1. INTRODUCCIÓN

La iniciativa Industria 4.0 aboga por la fabricación inteligente como la revolución industrial que conduce a la globalización y crecimiento económico (Lasi, Fettke, Kemper, Feld & Hoffmann, 2014), de tal forma que algunas industrias han comenzado la transformación de la era digital a la era inteligente, mediante el internet de las cosas (IoT), inteligencia artificial (IA) y *Big Data*. En este contexto, la fabricación tradicional en masa evoluciona a la oferta de fabricación a medida (CM) un paradigma de valor añadido para la fabricación inteligente (Modrak, Soltysova, Semanco & Sudhakara, 2019), ya que se refiere a productos y servicios personalizados.

Normalmente, las soluciones de inteligencia artificial se pueden aplicar a varios aspectos de fabricación inteligente. Los algoritmos de IA pueden ejecutar la fabricación de productos personalizados en una fábrica inteligente (Bahrin, Othman, Azli & Talib, 2016). El CM asistido por IA es para construir sistemas de fabricación soportados por computación cognitiva, detección del estado de la máquina, análisis de datos en tiempo real y toma de decisiones autónoma (Mircică, 2019). IA impregna a través de todos los eslabones de las cadenas de valor de CM, como el diseño, producción, gestión y servicio (El Hassani, El Mazgualdi & Masrour, 2019).

Basado en estos conocimientos de MC e IA, el enfoque de este artículo está en la implementación de la IA en la fábrica inteligente para CM que involucran arquitectura, equipos de bandas transportadoras, concretamente, se compararon dos arquitecturas de red neuronal artificial para encontrar la que mejor regula la velocidad angular del motor con y sin perturbaciones externas.

Para poder automatizar sistemas se emplea la ingeniería de control. Esta se caracteriza por los sistemas retroalimentados (Hernández-Olivares, 2018). Este tipo de sistemas utilizan la señal de salida y la comparan con un valor de referencia (ver figura 1). Esto genera una señal de error que sirve para, mediante un controlador, cambiar la respuesta de un dispositivo. El dispositivo a controlar se llama planta.

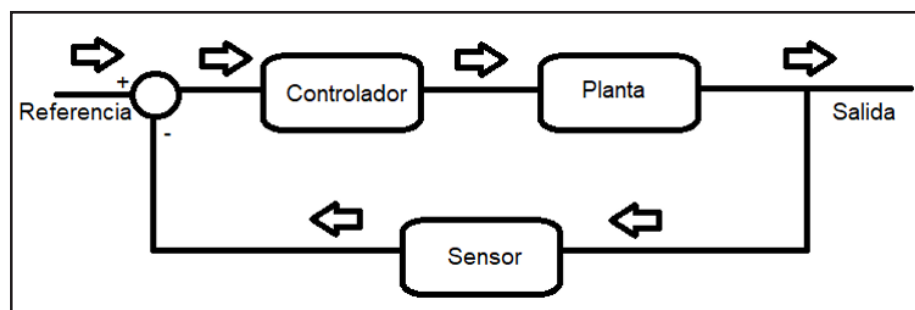


Figura 1. Diagrama de bloques de un sistema retroalimentado y su flujo de señales.  
Elaboración propia

En el proceso de sintonización de controladores, se requiere caracterizar la señal de entrada y la señal de salida. Por lo regular se emplean funciones matemáticas encarnadas en dispositivos electrónicos para procesar las señales dentro del sistema de control. Entre las herramientas utilizadas para modelar, analizar y diseñar este tipo de sistemas se encuentran las Ecuaciones Diferenciales, la transformada de Laplace y la función de transferencia.

Es justamente este último componente el que permite identificar características importantes del sistema. Una función de transferencia es una función matemática que representa el cociente entre la señal de salida y la señal de entrada (De Domenico, Rolland & Hochgreb, 2019). La función es algebraica y se representa en el plano complejo. Las raíces del polinomio numerador se llaman ceros, mientras que las raíces del polinomio denominador se llaman polos.

Cualquier sistema de lazo cerrado tiene un equivalente matemático en lazo abierto. Este equivalente se calcula mediante álgebra de bloques. Los polos del equivalente de lazo abierto determinan si el sistema tiene la capacidad de seguir la señal de referencia. Es decir, determina si el sistema puede operar de forma autónoma. Si el sistema no tiene esa capacidad, puede agregarse un controlador. El controlador más utilizado es el Proporcional-Integral-Derivativo (PID). Entre sus características más importantes se encuentra el uso de una ganancia proporcional para modificar la posición de los polos en el plano complejo, un tiempo de integración que contribuye a la disminución del error estado estable (diferencia entre la señal de referencia y el valor de salida) y la mejora en el tiempo de convergencia empleando la derivada del error (ver ec.1). El PID se alimenta fundamentalmente de la señal de error del sistema y se trata de un componente de bajo costo. Sin embargo, una de sus principales limitaciones es la incapacidad para procesar más de una señal de entrada o controlar más de una planta.

$$PID = k_p \left( 1 + \frac{1}{T_i S} + T_d S \right) \quad ec. 1$$

En los casos en donde la cantidad de entradas y/o salidas del sistema sean más de una, o que el dinamismo del sistema sea complicado de modelar, se pueden emplear controladores inteligentes (Neves, De Araújo Neto, Sales, Vasconcelos, & Brito, 2021). Estos son herramientas de inteligencia artificial incorporados al formalismo de la ingeniería de control (Luna, 2019). Entre dichas herramientas se encuentran la lógica difusa y las Redes Neuronales Artificiales (RNA).

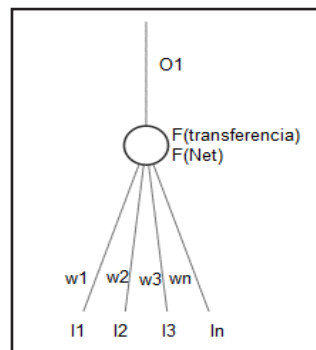


Figura 2. Modelo y componentes de una neurona artificial. Elaboración propia

Las RNA son modelos computacionales utilizados para la clasificación en ambientes altamente dinámicos (Van Gerven, & Bohte, 2017). Proviene del modelo de procesamiento de la información de una neurona natural. Una neurona artificial (ver figura 2) está compuesta por un conjunto de entradas (I) que son ponderadas por pesos sinápticos (W). Las entradas y sus ponderaciones son cuantificadas en una función neta (ver ec.2), que puede ser acompañada por un bias para el mejor ajuste de umbral de disparo.

$$Net = \sum_{j=0}^n I_j W_j \quad ec. 2$$

La función NET es convertida en un espacio matemático distinto mediante una función de transferencia (no es la misma función que aquella que modela una planta). Mediante una función de salida, se determina si la neurona está activa y en qué nivel de activación se encuentra, lo cual se refleja en la salida (O) de la neurona.

Para entrenar a una red neuronal existen dos opciones: aprendizaje supervisado y aprendizaje no supervisado. En el aprendizaje supervisado es necesario conocer cada estímulo a la red con su posible respuesta. Un ejemplo de este proceso es el algoritmo de retropropagación del error. Mientras que el no supervisado requiere de una respuesta aproximada a los diferentes estímulos presentados. Los algoritmos genéticos son buenos representantes de este tipo de aprendizaje.

Existen diferentes configuraciones posibles para una RNA usada como controlador. El proceso de diseño está influenciado por la experiencia del programador quien determina que modelo de procesamiento es mejor para cada problema. De tal manera que la selección de neuronas y capas, así como funciones de transferencia y salida dependen de la naturaleza del problema a resolver.

Lo que se puede establecer en forma más sencilla es que la información proveniente de los sensores se conecta a la capa de entrada. Mientras que la salida de la red neuronal se relaciona con la señal de control sobre la planta. El controlador basado en RNA más sencillo para sistemas retroalimentados emplea el error y su derivada como señales de entrada. Aunque se puede agregar cualquier tipo de entrada. La cantidad de señales de salida depende de los dispositivos a controlar. Inclusive permite sincronizar dispositivos mediante una red o mediante la comunicación entre diferentes redes u otro tipo de controladores.

El trabajo presentado en este artículo tiene la finalidad de presentar el diseño de un controlador basado en redes neuronales para ajustar automáticamente la velocidad angular de un motor de corriente directa. Este motor funciona como mecanismo propulsor de una banda transportadora. Se compararon dos neurocontroladores, el primero emplea un sistema de entrenamiento basado en aprendizaje supervisado. El segundo, emplea un mecanismo de ajuste en tiempo real basado en aprendizaje no supervisado.

## **2. DESARROLLO DE LA PROPUESTA**

Las bandas transportadoras son mecanismos que emplean un motor eléctrico como mecanismo de movimiento. En el caso de este trabajo se empleó un motor de corriente directa con un minirecutor a 5V como plataforma de pruebas. La velocidad objetivo del sistema fue de 60 RPM y fue representada por una entrada de tipo escalón.

Para medir la velocidad angular de salida del motor se empleó un sensor infrarrojo FC-51 y se contabilizó en revoluciones por minuto (RPM). Como complemento para procesar la señal de salida del motor se utilizó una tarjeta de desarrollo electrónico de la marca Arduino Mega 2560 (ver figura 3). Para complementar el sistema de prueba y como mecanismo de interfaz entre la tarjeta de desarrollo y el motor, se incorporó un driver L293D (Puente H). Este último dispositivo permite controlar motores de corriente directa mediante dispositivos digitales de la familia TTL, ya que compensa la baja corriente de salida de estos dispositivos. Para complementar el control se empleó el sistema PWM, que regula la duración del ciclo alto de una onda cuadrada. Si se reduce el tiempo en alto del ciclo, se reduce la velocidad del motor. Si aumenta el componente en alto nivel aumenta, la velocidad de giro también lo hace.

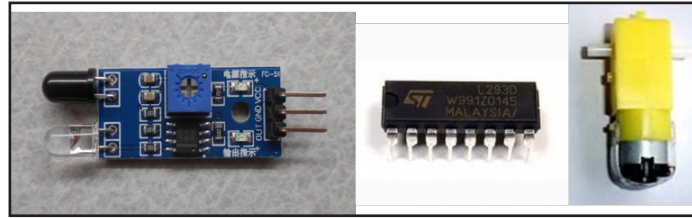


Figura 3. Componentes electrónicos empleados en el sistema de prueba. Elaboración propia

Se crearon dos grupos experimentales: controlador supervisado y controlador no supervisado. El primero representó a un conjunto de redes neuronales artificiales ajustadas por aprendizaje supervisado. El segundo correspondió a redes neuronales artificiales ajustadas por aprendizaje no supervisado. Cada grupo experimental estuvo constituido por 90 repeticiones del proceso de ajuste correspondiente.

Cada red neuronal ajustada en los dos grupos experimentales se probó y se determinó si cumplían con la condición de estabilidad (capacidad para seguir la señal de entrada). De la misma manera, se verificó si las redes ajustadas tenían un error en estado estable 0 con un margen de tolerancia del 2% respecto al valor objetivo. Las redes que resultaron estables y con error en estado estable 0 se probaron en cuatro escenarios. Los tres primeros iniciaron con una velocidad de giro de 40 RPM, 60 RPM y 80 RPM. El cuarto escenario implicó un aumento en la carga del sistema, colocando un objeto con un peso adicional al rotor del motor. En todos los casos se cuantificó la cualidad de estabilidad del sistema, además del tiempo de estabilización. Se calcularon los promedios de las repeticiones de cada grupo experimental para poder comparar el rendimiento de ambas técnicas.

Para el primer grupo experimental se empleó una red de 15 neuronas en la capa de entrada y 7 neuronas en la capa de salida (ver figura 4). La red estuvo constituida por 157 pesos, función de activación sigmoideal. La capa de entrada fue alimentada por el error y la derivada del error. Cada neurona en la red contenía un bias para facilitar el ajuste de disparo.

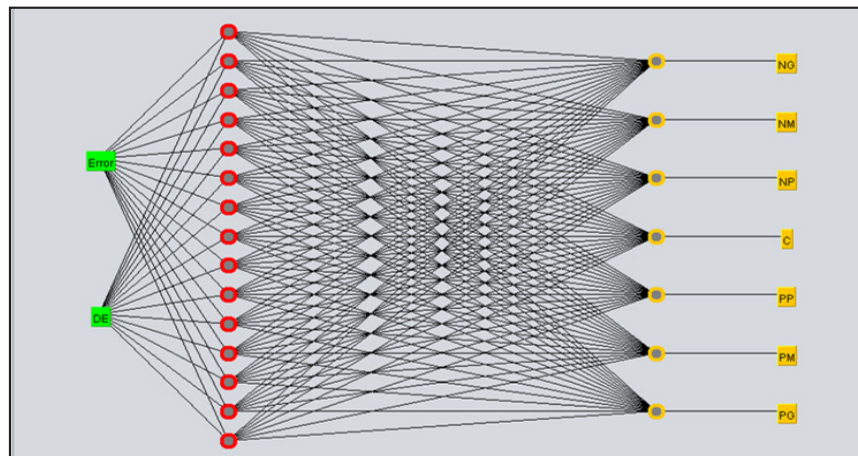


Figura 4. Red neuronal para aprendizaje supervisado. Elaboración propia

Para entrenar a la red neuronal se empleó el software Weka (Moreno ,2016). Para ellos se utilizó una base de datos constituida por 49 instancias (ver tabla 1). Esta base de datos estuvo formada por 7 valores nominales asociados con el error: negativo grande (NG), negativo medio (NM), negativo pequeño (NP), cero (C), positivo pequeño (PP), positivo medio (PM) y positivo grande (PG). Mismos 7 valores empleados para la derivada del error. Se utilizaron 7 valores de clase para la salida con las mismas 7 categorías de los datos de entrada. Estos valores de clase corresponden a las 7 acciones de control que se pueden ejecutar. Una vez entrenada la red se programó en la tarjeta Arduino. Cada salida activaba una reducción o aumento en el ciclo alto de trabajo del PWM. Las reglas de la base de datos fueron generadas a partir de la lógica del funcionamiento del motor. Fueron adaptadas a partir de modelos de conocimiento de trabajos previos (Cervantes, Mezura-Montes, y Guzmán-Gaspar, 2018).

| E/DE | NG | NM | NP | C  | PP | PM | PG |
|------|----|----|----|----|----|----|----|
| NG   | NG | NG | NG | NM | NP | NP | C  |
| NM   | NG | NG | NM | NP | NP | C  | PP |
| NP   | NG | NM | NP | NP | C  | PP | PP |
| C    | NM | NP | NP | C  | PP | PP | PM |
| PP   | NP | NP | C  | PP | PP | PM | PG |
| PM   | NP | C  | PP | PP | PM | PG | PG |
| PG   | C  | PP | PP | PM | PG | PG | PG |

Tabla 1. Datos para el aprendizaje supervisado.

La red neuronal artificial no supervisada constó de 3 capas (ver figura 5). La capa de entrada fue constituida por 4 neuronas, la capa intermedia por 3 y la capa de salida por 2. Cada neurona tuvo un bias para mejorar el ajuste de disparo. La red fue integrada por 35 pesos. Se utilizó una función de activación sigmoideal (ver ec.3). En la entrada, cada neurona recibía la señal de error y derivada del error. En la capa de salida, la primera regula la duración del ciclo en alto y la segunda del ciclo en bajo.

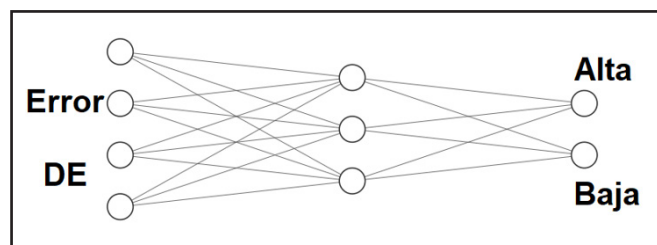


Figura 5. Red neuronal para aprendizaje no supervisado.  
Elaboración propia

$$F(\text{transferencia}) = \frac{1}{1 + e^{-net}} \quad \text{ec. 3}$$

Para su ajuste se empleó un algoritmo genético generacional (AGG). La configuración de este AGG fue reducida para que pudiera ser implementado en la tarjeta de desarrollo. Se utilizaron 6 individuos que fueron optimizados por 60 generaciones. Cada individuo fue probado en 3 trials de 30 steps, cada step de 10mS. El tiempo de prueba por cada individuo fue de 900mS. El tiempo de prueba esperado por todo el proceso evolutivo fue de 324S. Es importante señalar que, a este tiempo de prueba, se debe agregar el tiempo de procesamiento del algoritmo genético y sus operadores.

Se utilizó un operador de selección mediante torneo, seleccionando 2 padres de 4 candidatos. Los padres fueron cruzados mediante punto aleatorio. Se conservó hasta a un 20% de la población de la generación previa y se utilizó el mecanismo del elitismo para conservar al mejor individuo de cada generación. Se utilizó mutación del 2% al total de cromosomas de la próxima generación, exceptuando el mejor individuo.

La representación del cromosoma de los individuos fue binaria. Cada peso fue representado por 10 bits, 9 de los cuales corresponden a un número entre 0 y 1. El bit más significativo permitió establecer si el peso tenía valor positivo o negativo. Como función de aptitud se cuantificó el error acumulado durante todos los steps de prueba. Al error máximo esperado en cada step de prueba se le restó el error acumulado por cada individuo (ver ec.4). Cada proceso evolutivo corrió en el sistema de prueba en tiempo real en condiciones de operación normales.

$$f_{aptitud} = \sum_{n=0}^{90} (1 - error_n) \quad ec. 4$$

### 3. RESULTADOS

La propiedad más importante de un sistema de control es la estabilidad. Es decir, la capacidad de seguir la señal de referencia después de un tiempo de operación. En el caso de la RNA entrenada por aprendizaje supervisado (ver figura 6), 82 de las 90 redes resultaron estables. Esto representa el 91.11% con esta capacidad, mientras que 8 de las redes no tuvieron la capacidad de estabilización (8.88%). Por su parte, el aprendizaje supervisado produjo 75 de 90 (83.33%) RNAs con la propiedad de la estabilidad. Mientras que 15 (16.66%) de las redes entrenadas no alcanzaron la propiedad de estabilidad del sistema.

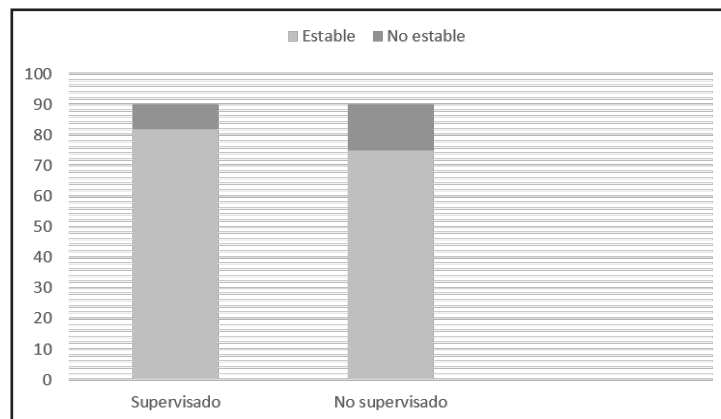


Figura 6. Capacidad de estabilidad en los neuro controladores entrenados. Elaboración propia

A partir de estos resultados se puede establecer que el aprendizaje supervisado arroja una mayor cantidad de controladores que producen estabilidad en el sistema. Pero esta capacidad no fue la única analizada. También se contabilizó la cantidad de controladores cuya salida alcanzó el valor de estabilidad deseado (60RPM) con una tolerancia del 2%. Es decir, se contabilizó la cantidad de redes que generaron un error en estado estable 0.

En cuanto al aprendizaje supervisado (ver figura 7), 60 redes (66.66%) obtuvieron un error en estado estable de 0. Del total de redes entrenadas, 30 (33.33%) no cumplieron con esta característica. Por su parte, en el aprendizaje no supervisado, la cantidad de redes que no tuvieron error en estado estable fue de 68 (75.55%). Mientras que 22 (24.44%) no cumplieron con esta característica.

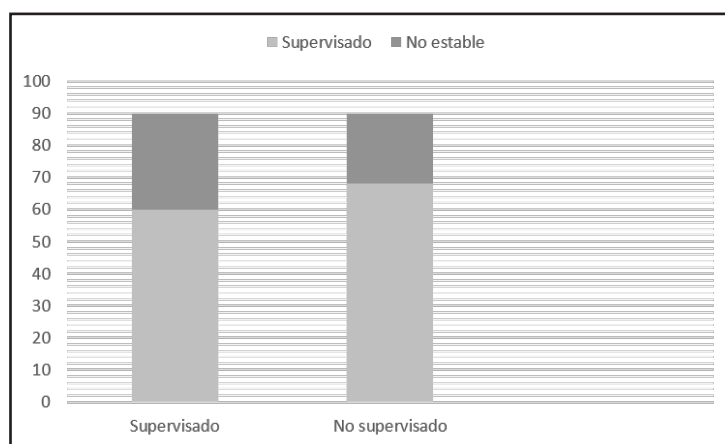


Figura 7. Capacidad de reducción del error en estado estable en los neuro controladores entrenados. Elaboración propia

Del total de controladores estables y sin error en estado estable, se contabilizó el tiempo de estabilización al 2%. El tiempo de estabilización se midió en segundos. Se calculó el promedio de los tiempos medidos en cada una de las cuatro pruebas. Así, la comparación de algoritmos permite establecer que el aprendizaje supervisado produce un tiempo de estabilización más elevado (ver tabla 2) con una prueba que arranca a 60RPM.

| Prueba             | Aprendizaje Supervisado | Aprendizaje no Supervisado |
|--------------------|-------------------------|----------------------------|
| 60 RPM             | 11.9 s.                 | 11.1 s.                    |
| 40 RPM             | 18.9 s.                 | 16.72 s.                   |
| 80 RPM             | 20.1 s.                 | 13.5 s.                    |
| 60 RPM (con carga) | 12.5 s.                 | 10.1 s.                    |

Tabla 2. Resumen de resultados

Cuando las velocidades de inicio son diferentes son diferentes a la velocidad objetivo, la superioridad de la respuesta del aprendizaje no supervisado es mayor. Lo mismo ocurre en la prueba con la carga (ver figura 8).

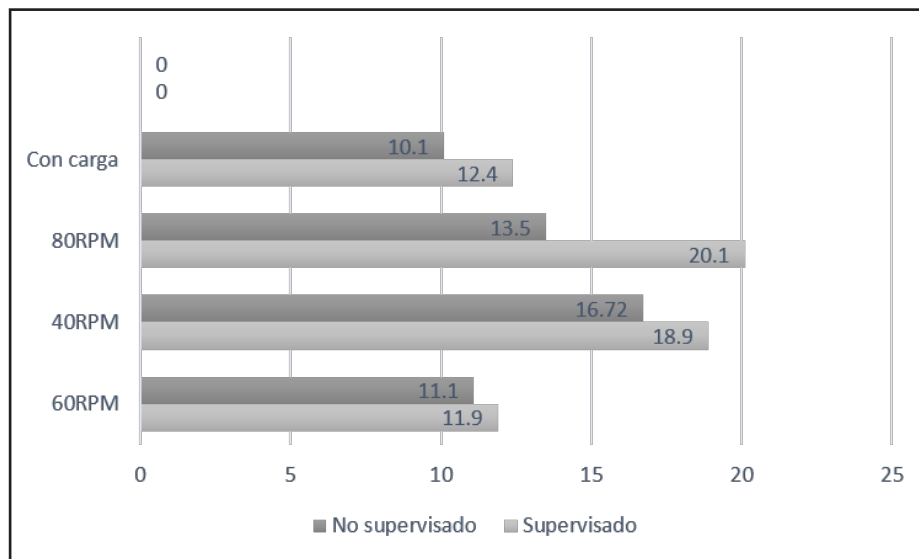


Figura 8. Promedio de los tiempos de estabilización de los controladores ante las cuatro pruebas realizadas. Elaboración propia

#### 4. DISCUSIÓN

La convergencia entre la IA y los dispositivos crea un nuevo campo de estudios y desarrollo de aplicaciones. En estos estudios, los esfuerzos son concentrados en el diseño de entrenamiento e inferencia, adaptando los modelos y acelerando el hardware para su uso con IA. Estos modelos aplicados a cualquier etapa de la fabricación inteligente, suponen una mejora óptima, tal es el caso de su aplicación a las bandas transportadoras, con las cuales se puede tener un control de inventarios, transporte y despacho de los productos terminados, dando como resultado una IA que permita seleccionar y ajustar las velocidades de transporte de acuerdo a las necesidades o pesos del producto, entre otros factores importantes.

Se construyó un sistema físico que emula el movimiento de una banda de transporte. Este dispositivo tuvo como componente fundamental un motor de corriente directa con un micro reductor a 5V. El sistema implementado es del tipo lazo cerrado. Para retroalimentar la señal de salida se utilizó un sensor infrarrojo (FC-51). Las señales fueron procesadas en una tarjeta Arduino.

Se entrenaron dos tipos de neuro controladores, empleando aprendizaje supervisado y no supervisado. Se realizaron pruebas de estabilidad, error en estado estable y tiempo de estabilización.

El análisis de los resultados permitió establecer que el controlador basado en RNA y entrenado en forma no supervisada fue el que mejor rendimiento presentó. Aunque una mayor cantidad de redes neuronales alcanzó la estabilidad

mediante el algoritmo supervisado, estas no cumplieron con las condiciones de estado estable necesarias para alcanzar la velocidad objetivo (60RPM). Este componente es fundamental en el sistema, ya que, de no alcanzar la velocidad de salida necesaria, se requiere de un componente adicional de control como una ganancia proporcional.

El aprendizaje no supervisado arrojó RNAs cuyo tiempo de convergencia (tiempo de estabilización) es menor en las pruebas realizadas. En estas pruebas se inició el sistema con una velocidad determinada, para conocer el tiempo en el que alcanzarían su velocidad objetivo en forma autónoma. Es importante señalar que los ajustes que se hacen durante el tiempo de prueba del sistema provienen de las salidas de la red neuronal que controla la planta.

Adicionalmente, el aprendizaje supervisado mostró capacidad ante perturbaciones externas. Lo que se observó en la prueba de tiempo de estabilización con carga. Al agregar peso adicional al rotor del motor, las condiciones de funcionamiento cambian. Sin embargo, el sistema no supervisado mantuvo su capacidad de estabilizar el sistema a la velocidad objetivo. En términos prácticos, esta prueba corresponde a los cambios de peso en la banda transportadora. De esta forma, las propiedades que se pueden deducir del aprendizaje no supervisado es que se trata de un sistema dinámico para el control de sistemas. Permite obtener sistemas estables, con error en estado estable mínimo y con capacidad para contender con el dinamismo del medio.

Otra cuestión que resulta importante es que este proceso se realizó en la planta en físico. Por lo que permite ajustar el rendimiento del controlador en tiempo real. Lo que, permite evitar problemas de Reality Gap (Masiero, 2016). Estos aparecen cuando un proceso evolutivo se produce en un ambiente simulado, que al ser probado en un ambiente físico genera diferencias de la respuesta del sistema. Esto ocurre como una consecuencia de las limitantes de los modelos computacionales en los simuladores.

El AGG empleado tiene una cantidad reducida de individuos y generaciones. Aunque existen pocos puntos de exploración en el espacio de soluciones potenciales, la solución suele converger a una configuración lo suficientemente adecuada para lograr el objetivo de control. Esta capacidad es totalmente compatible con la plataforma de procesamiento de la información, en donde no se utiliza la totalidad de los recursos computacionales en la solución del problema.

Este tipo de algoritmos pueden ser aplicados a diferentes plantas y tareas. Se debe procurar que los sistemas sean sometidos a todos los estímulos posibles en su etapa de entrenamiento. Particularmente aquellos estímulos más importantes. Por lo que no resulta importante determinar rangos de las variables de entrada ni establecer relaciones, ya que todo emerge del proceso de ajuste.

Además, los AGG, como cualquier sistema no supervisado, fomentan la emergencia de las soluciones desde la perspectiva del interior del sistema (Báez, 2019). Esto en otras ramas de la IA se le conoce como enfoque proximal (Pagliuca, Milano y Nolfi, 2020), que no siempre puede ser descrito directamente por el diseñador (Palacios-Leyva, Aldana-Franco, Lara-Guzmán & Montes-González, 2017).

Aplicados a un sistema de control de bandas transportadoras este tipo de sistemas permiten diseñar sistemas de control para cumplir con el objetivo deseado. Esta tarea será cumplida en condiciones normales y con perturbaciones externas. Además, permiten ajustar el controlador directamente en el modelo físico. Por lo que el entrenamiento puede realizarse

nuevamente cuando las condiciones del sistema cambian. Por ejemplo, cuando se cambia alguna pieza del sistema o algunas dimensiones de este son modificadas.

Este grado de adaptabilidad resulta adecuado para la perspectiva futura de la mayoría de las empresas. Particularmente es compatible con la industria 4.0, en donde se busca aumentar el grado de automatización de los sistemas y la coordinación de estos (Hozdić, 2015). Sobre todo, que estas soluciones sean adaptables a los diferentes entornos, sin la necesidad de modificar muchos componentes del sistema (Zawadzki & Żywicki, 2016).

## 5. CONCLUSIÓN

En este artículo se presenta un sistema de control retroalimentado basado en redes neuronales artificiales que funciona mediante el error y la derivada del error. La salida de la red controla la parte alta y baja de un ciclo de salida mediante la señal de PWM. El controlador es ajustado por un AGG mediante aprendizaje no supervisado. Este algoritmo se implementa en una tarjeta de desarrollo con pocos individuos y generaciones. La función de aptitud opera mediante la cuantificación del error. Los resultados presentados demuestran la capacidad para producir soluciones estables, con un grado de error en estado estable cercano a cero y con tiempos de respuesta adecuados para el problema planteado. Se puede resaltar la capacidad adaptativa de este sistema de aprendizaje no supervisado. También es destacable la perspectiva proximal para la solución del problema, misma que considera las condiciones internas y la auto-organización de la solución. Con lo cual se reduce la influencia del diseñador y se privilegia la experiencia del sistema en el entorno.

## 6. REFERENCIAS

- Báez, P. (2019). *Sistema de detección inteligente de personas atrapadas en derrumbes por sismos utilizando internet de las cosas* [Tesis de licenciatura, Universidad Veracruzana].
- Bahrin, M.A.K., Othman, M.F., Azli, N.N. & Talib, M. F. (2016). Industry 4.0: A review on industrial automation and robotic. *Jurnal Teknologi*, 78(6–13), 137–143.
- Cervantes, M. G., Mezura-Montes, E. & Guzmán-Gaspar, J. Y. (2018). Differential evolution based adaptation for the direct current motor velocity control parameters. *Mathematics and Computers in Simulation*, 122-141.
- De Domenico, F., Rolland, E. O. & Hochgreb, S. (2019). A generalised model for acoustic and entropic transfer function of nozzles with losses. *Journal of Sound and Vibration*, 440, 212-230.
- El Hassani, I., El Mazgualdi, C. & Masrour, T. (2019). Artificial intelligence and machine learning to predict and improve efficiency in manufacturing industry. arXiv e-prints, arXiv-1901.
- Hernández-Olivares, J. (2018). *Comparación de controladores inteligentes para un sistema de control retroalimentado para un motor de corriente alterna* [Tesis de licenciatura Universidad Veracruzana].
- Hozdić, E. (2015). Smart factory for industry 4.0: A review. *International Journal of Modern Manufacturing Technologies*, 7(1), 28-35.

- Lasi, H., Fettke, P., Kemper, H. G., Feld, T. y Hoffmann, M. (2014) Industry 4.0. *Inf. Syst. Eng.*, 6(4), 239–242.
- Luna, M. (2019). *Sistema Inteligente de Monitoreo para Condiciones Ambientales en Industria 4.0* [Tesis de licenciatura, Universidad Veracruzana].
- Masiero, S. (2016). The origins of failure: seeking the causes of design–reality gaps. *Information Technology for Development*, 22(3), 487-502.
- Mircică, N. (2019). Cyber-physical systems for cognitive Industrial Internet of Things: Sensory big data, smart mobile devices, and automated manufacturing processes. *Analysis and Metaphysics*, (18), 37-43.
- Modrak, V., Soltysova, Z., Semanco, P., y Sudhakara, P.R. (2019). Production scheduling and capacity utilization in terms of mass customized manufacturing. En *Advances in Manufacturing II*. Cham, Switzerland: Springer, 2019, 295–306.
- Moreno, N. F. (2016) *Introducción a la Minería de Datos con Weka: Aplicación a un Problema Económico* [Tesis de licenciatura, Universidad de Jaén].
- Neves, T. G., de Araújo Neto, A. P., Sales, F. A., Vasconcelos, L. G. S., y Brito, R. P. (2021). ANN-based intelligent control system for simultaneous feed disturbances rejection and product specification changes in extractive distillation process. *Separation and Purification Technology*, 259, 118104.
- Pagliuca, P., Milano, N., y Nolfi, S. (2020). Efficacy of modern neuro-evolutionary strategies for continuous control optimization. *Frontiers in Robotics and AI*, 7.
- Palacios-Leyva, R., Aldana-Franco, F., Lara-Guzmán, B., y Montes-González, F. (2017). The impact of population composition for cooperation emergence in evolutionary robotics. *International Journal of Combinatorial Optimization Problems and Informatics*, 8(3), 20-32.
- Van Gerven, M., y Bohte, S. (2017). Artificial neural networks as models of neural information processing. *Frontiers in Computational Neuroscience*, 11, 114.
- Żywicki, K., Zawadzki, P., & Górski, F. (2017, September). Virtual reality production training system in the scope of intelligent factory. *International Conference on Intelligent Systems in Production Engineering and Maintenance* (pp. 450-458). Springer, Cham.

## LOS AUTORES

**Sánchez Romero Jonathan**  
**Leyva Retureta José Gustavo**  
**Rivera Peña Yazmín**  
**Aldana Franco Fernando**

Facultad de Ingeniería Mecánica Eléctrica  
Universidad Veracruzana