

# Red neuronal artificial para la extracción de parámetros dinámicos de robots a partir de información incompleta de su movimiento

## Artificial neural network for the extraction of dynamic parameters of robots from incomplete information of their movement

Carlos Leopoldo Carreón Díaz de León<sup>1</sup> , Sergio Vergara Limon<sup>1</sup> , Juan Manuel Gonzalez-Calleros<sup>1</sup> , María Aurora Diozcora Vargas Treviño<sup>1</sup> 

<sup>1</sup>Benemérita Universidad Autónoma de Puebla, Puebla, México.

[carlos.carreond@alumno.buap.mx](mailto:carlos.carreond@alumno.buap.mx), [sergio.vergara@correo.buap.mx](mailto:sergio.vergara@correo.buap.mx), [juan.gonzalez@cs.buap.mx](mailto:juan.gonzalez@cs.buap.mx),  
[aurora.vargas@correo.buap.mx](mailto:aurora.vargas@correo.buap.mx)

(Recibido: 28 julio 2021; aceptado: 15 septiembre 2021; Publicado en Internet: 1 diciembre 2021)

**Resumen.** Las redes neuronales artificiales son útiles para procesar datos con información incompleta para obtener una salida deseada. En los sistemas de medición de robots manipuladores, solo se toman muestras cuantificadas de la posición y, por lo tanto, no se puede ejecutar en un tiempo razonable algoritmos deterministas para extraer los parámetros del robot. En el estado del arte, se abordan algoritmos de extracción de parámetros basados en la suposición de que no existe la cuantificación de las señales del movimiento del robot y que la primera y segunda derivada de la posición son muestreadas y no estimadas. En este trabajo, se propone un algoritmo basado en una red neuronal entrenada para extraer los parámetros de un determinado robot para reducir el tiempo de caracterización del robot, además, con la metodología propuesta se pueden extraer parámetros dinámicos del mismo tipo de robot con el que se ha entrenado la red neuronal.

**Palabras clave:** Red neuronal, Robots, Parámetros dinámicos.

**Abstract.** The artificial neural networks are suitable for processing incomplete data to achieve the desired output. The acquisition system of the manipulator robots takes quantified samples of the position; therefore, it is not possible to execute deterministic algorithms of parameter extraction in a reasonable time. State of the art describes algorithms based on the assumption that the motion signals are not quantified, and the first and second derivatives of the position are sampled instead of estimated. In this paper, a trained neural network-based extraction parameter algorithm for a determined robot is proposed to reduce the robot characterization time. Also, with the proposed methodology is possible to extract the parameters of the same kind of robot used for training the neural network.

**Keywords:** Neural network, Robots, Dynamic parameters.

**Tipo de artículo:** Artículo de investigación.

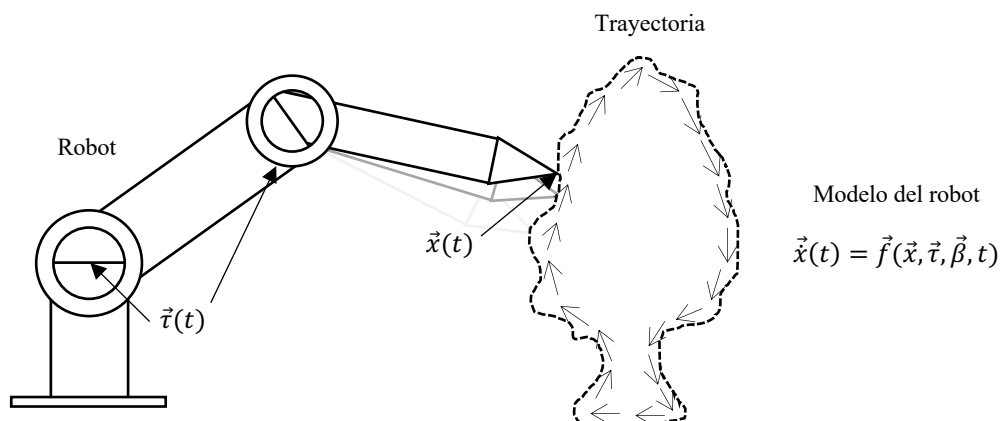
## 1 Introducción

Los robots manipuladores están descritos por ecuaciones diferenciales ordinarias conocidas como modelo dinámico (ver [Figura 1](#)). Para que el modelo dinámico esté asociado a un robot real, las constantes involucradas o parámetros dinámicos deben coincidir con las constantes físicas del robot real. El problema de medir estos parámetros es que el robot debe estar en movimiento ya que la mayoría están asociadas a la velocidad y aceleración del robot.

El procedimiento de extraer parámetros de un robot se denomina identificación paramétrica y consiste en medir los parámetros con las señales del movimiento y las señales de los actuadores. Debido a que el modelo de un robot está determinado por ecuaciones, la tendencia para identificar los parámetros dinámicos es diseñar algoritmos basados en estas ecuaciones.

Esto implica que para que el algoritmo de identificación funcione adecuadamente, se deben de tener muestras de todas las variables involucradas del modelo: posiciones, velocidades, y aceleraciones. Sin embargo, la tecnología de medición de movimiento de los robots es limitada y únicamente se puede medir la posición con un cuantificador. Las señales de velocidad y aceleración son estimadas con la posición.

En aplicaciones de control de robots, los parámetros dinámicos son la compensación del modelo del robot con el objetivo de obtener una trayectoria de movimiento deseada. Sin los parámetros dinámicos, el ajuste de sistemas de control es complicado y depende de la experiencia para encontrar las constantes del controlador que permitan tener una trayectoria deseada con el menor error posible.



**Figura 1.** Robot manipulador siguiendo una trayectoria resultante del modelo del robot.

Para optimizar el diseño de robots, se necesitan los parámetros del modelo con el fin de emular su comportamiento en una computadora que puede obtener una respuesta del robot en menor tiempo que el robot real. Por lo tanto, se propone una metodología basada en una red neuronal que pueda identificar parámetros de un determinado robot en un tiempo menor al que toma con algoritmos convencionales. Este trabajo está dividido en las siguientes secciones. La sección 2, aborda el estado del arte de algoritmos de identificación paramétrica. En la sección 3, están los preliminares matemáticos. La sección 4 presenta la metodología propuesta. En la sección 5, están los resultados obtenidos de la metodología propuesta. Finalmente, la sección 6 presenta las conclusiones del artículo.

## 2 Trabajos relacionados

En trabajos de identificación paramétrica se ha encontrado que los sensores de movimiento son considerados como muestreadores discretos analógicos (Ogata, 1996). Del espectro de frecuencia del modelo de sensor considerado se recupera la señal original debajo del ancho de banda del muestreo. Sin embargo, no se ha encontrado aún un trabajo de identificación paramétrica que considere los sensores como muestreadores cuantificados digitales. En la [Figura 2a](#), está la distribución de los actuadores que utilizan los robots en los trabajos de identificación paramétrica de robots manipuladores consultados.

Xiao et al. (2007) y Swevers et al. (2007) muestran los métodos de diferenciación numérica y por espectro respectivamente utilizados en la mayoría de los trabajos consultados para estimación de velocidad y aceleración. Los actuadores son considerados como sistemas de transferencia de energía muestreados analógicos. Sin embargo, al igual que los sensores, no son modelados como muestreadores asíncronos digitales.

Para el análisis de ruido de las señales de los robots se ha encontrado que se asume un modelo de ruido aditivo en las señales de posición y estimaciones de velocidad, aceleración, y torque. Chávez-Olivares et al. (2012) estudia la identificación de un robot manipulador y se considera que las estimaciones de velocidad y aceleración contienen ruido numérico. En la [Figura 2b](#) está la distribución del modelo de ruido aditivo usado para la identificación paramétrica.

En Lin et al. (2008) se implementa un filtro digital para un encoder de posición. Por otra parte, Huynh et al. (2020) consideran que el encoder de posición utilizado en su robot contiene ruido aditivo de distribución normal, aunque no se muestra la distribución de ruido medida experimentalmente.

El filtro de Kalman (Ding et al., 2014; Kalman, 1960; Soh & Wu, 2017) contiene un estudio detallado para sistemas dinámicos con ruido aditivo de distribución normal en sus señales, sin embargo, las señales de las variables de estado se consideran continuas discretas.

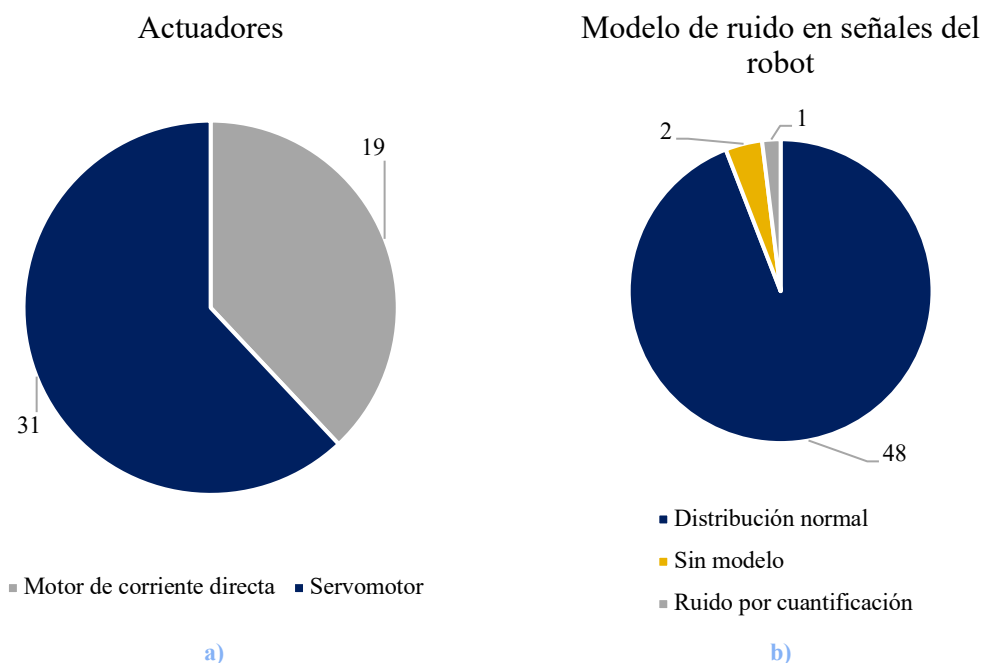


Figura 2. a) Distribución de tipos de actuadores, b) Tipo de modelo de ruido aditivo.

El modelo dinámico de los robots estudiados en los trabajos de identificación paramétrica es desarrollado por la metodología de Euler-Lagrange, Newton-Euler, y por aproximaciones lineales (Taylor, 2013). La metodología de Euler-Lagrange es presentada en forma resumida en la sección 3, donde se utiliza una notación vectorial. En la Figura 3a se aprecia la distribución de metodologías para los modelos dinámicos.

En los trabajos consultados, no se analiza previamente el comportamiento del modelo dinámico. Qin et al. (2010) estudia un robot de dos grados de libertad para identificación paramétrica el cual exhibe un comportamiento caótico en función de las condiciones iniciales. Los modelos de fricción utilizados en la mayoría de los trabajos son la fricción viscosa, fricción de Coulomb, y el modelo de la curva de Stribeck (You et al., 2012) (ver Figura 3b).

Los algoritmos de identificación paramétrica más utilizados son mínimos cuadrados (Pohlman & Leitner, 2003), máxima verosimilitud (Devore, 2011; Swevers et al., 1997), y redes neuronales (Berzal, 2018). Hu & Xiong (2018) utiliza una red neuronal multicapa para estimar el torque de un robot sin identificar los parámetros dinámicos (ver Figura 4). Wu & Jahanshahi (2019) utiliza una red neuronal convolucional para estimar el movimiento de tres sistemas dinámicos. Por su parte, Agand et al. (2017), utilizan una red neuronal recurrente para compensar el torque de un controlador PID en un helicóptero colocado en una base firme. Sus resultados de más de 100 segundos muestran que el helicóptero se estabiliza en la posición deseada.

Wang et al. (2020) utilizan una red neuronal *long short-term memory* (LSTM) en conjunto con el algoritmo de mínimos cuadrados para la identificación de parámetros dinámicos de un robot, sin embargo, se necesita de la optimización de una trayectoria de movimiento para que el robot identifique parámetros dinámicos. Liu et al. (2020) utilizan una red neuronal artificial en conjunto con el modelo dinámico de un robot para estimar el torque. Por su parte, Wu & Jahanshahi (2019) y Su et al. (2020), utilizan redes neuronales convolucionales para estimar el movimiento de robots manipuladores y dirigirlos con controles externos respectivamente.

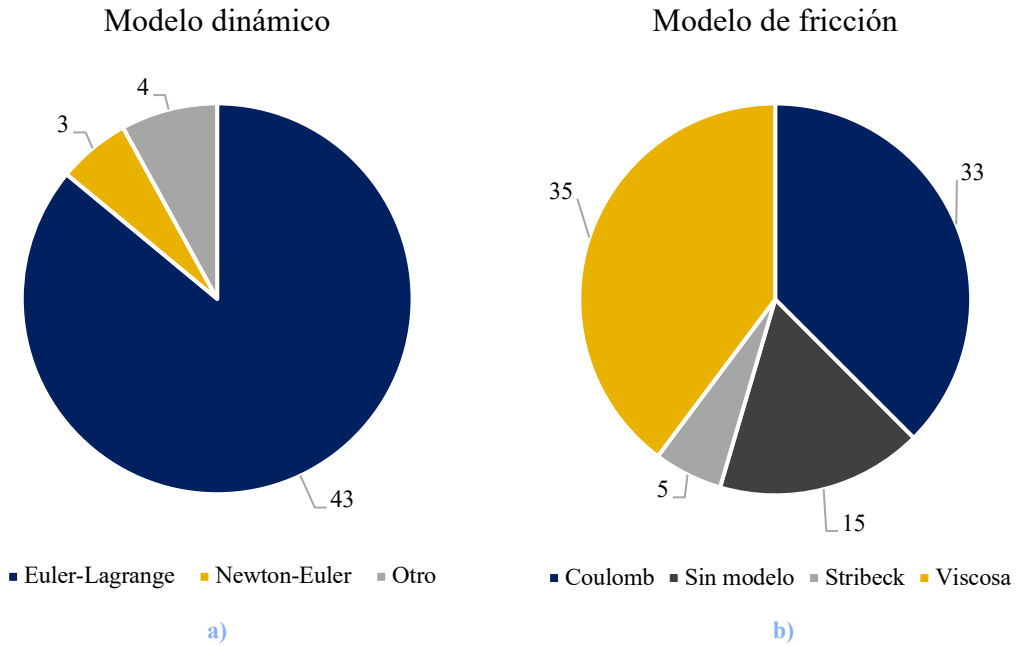


Figura 3. a) Distribución de metodologías utilizadas para determinar el modelo dinámico de un robot, b) Distribución de modelos de fricción utilizados.

#### Algoritmo de identificación paramétrica

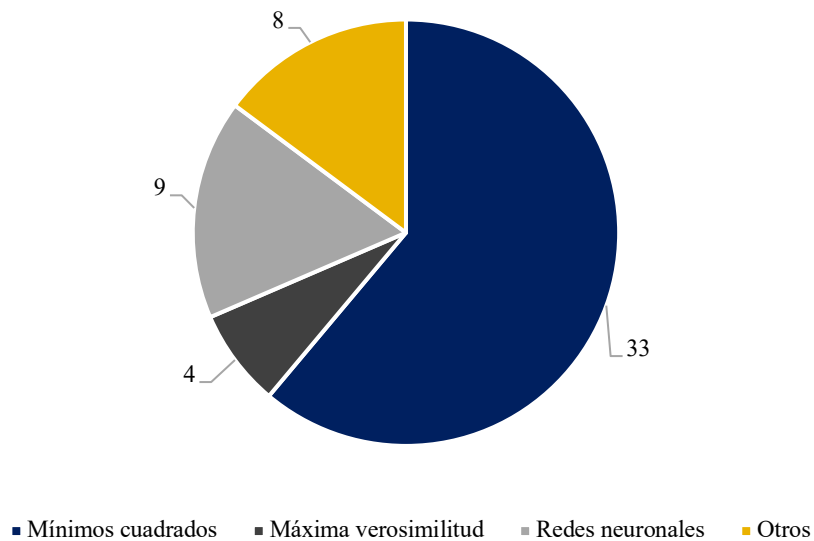


Figura 4. Distribución de algoritmos de identificación paramétrica en el estado del arte.

En la gran mayoría de trabajos consultados se utiliza la optimización de una trayectoria de movimiento con el fin de observar los parámetros dinámicos. La trayectoria se optimiza con algoritmos de inteligencia artificial, y las justificaciones halladas son relacionadas a reducir el ruido de las señales de posición, excitar los parámetros dinámicos, evitar la frecuencia natural del robot, y evitar diferenciación numérica (ver Figura 5).

### Trayectoria de movimiento del robot

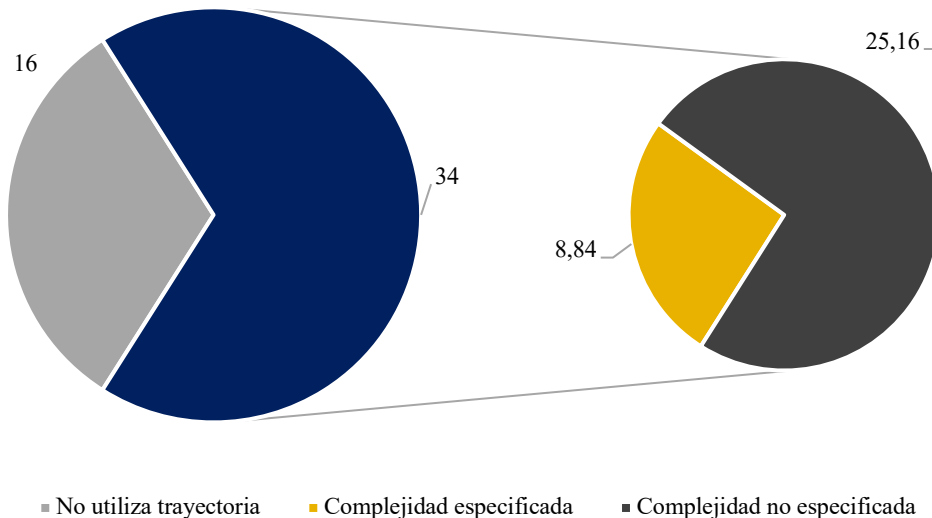


Figura 5. Distribución de la optimización de una trayectoria de movimiento definida y sus propiedades.

La validación de los parámetros dinámicos de los trabajos consultados nunca se realiza a bajas, medias, y altas velocidades. Tampoco se compara la aportación de cada parámetro dinámico al torque y su relación con los demás. En la Figura 6 se observa cómo se validan los parámetros dinámicos encontrados de los robots.

Las conclusiones del estado del arte son que no se ha hallado un algoritmo de estimación paramétrica con una red neuronal artificial que estime parámetros dinámicos sin la optimización de una trayectoria. Se ha encontrado que los modelos de señales considerados evitan mediciones completas del movimiento. Se da como aceptado y adecuado las estimaciones de velocidad y aceleración sin un análisis previo, además, son consideradas como mediciones directas de velocidad sin errores y continuas en los algoritmos de identificación. No se analiza el comportamiento de los robots descritos y las limitaciones de los sistemas de medición utilizados.

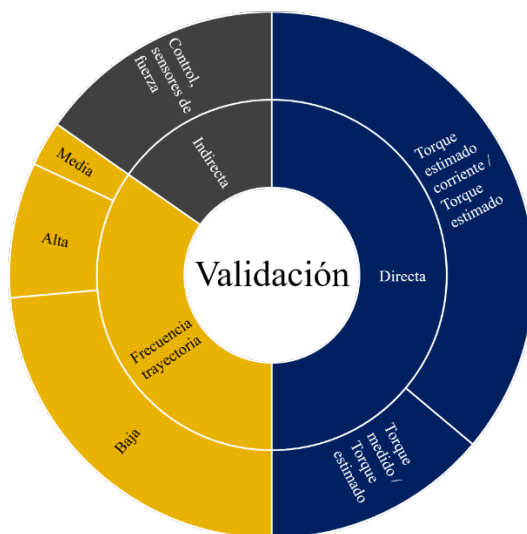


Figura 6. Características de la validación de los parámetros dinámicos del robot.

### 3 Preliminares

El modelo dinámico de un robot está determinado por un conjunto de ecuaciones diferenciales continuas. En la ecuación (1) el vector  $\vec{x}$  contiene las variables de estado, la función  $\vec{f}$  es el modelo dinámico,  $\vec{\beta}$  son los parámetros dinámicos,  $t$  es el tiempo, y el vector  $\vec{u}$  es la entrada de energía. En este modelo se considera que el vector de estados que contiene la posición y la velocidad de un robot pertenecen al conjunto de los números reales. Además, el vector de entrada pertenece al mismo conjunto.

$$\frac{d}{dt}\vec{x} = \vec{x}(t) = \vec{f}(\vec{x}, \vec{u}, \vec{\beta}, t) \in R^{2N} \quad (1)$$

Los parámetros dinámicos  $\vec{\beta}$  están asociadas a los parámetros de inercia, de fricción viscosa, de Coulomb, y a la gravedad. La ecuación (1) tiene una única solución para cada condición inicial  $\vec{x}_0$  por lo que es posible solucionar la ecuación por medio de integración:

$$\vec{x} = \int_{t_0}^t \vec{f}(\vec{x}, \vec{u}, \vec{\beta}, s) ds + \vec{x}_0 \quad (2)$$

Para identificar los parámetros dinámicos  $\vec{\beta}$ , se modifica la ecuación (2) para que las funciones involucradas en la función  $\vec{f}$  sean proporcionales a los parámetros. El modelo se reescribe como  $\vec{x}(t) = A(\vec{x}, \vec{u}, t)\vec{\beta}$  donde  $A$  es la matriz de observaciones. En implementaciones reales la medición del vector  $\vec{x}$  es parcial para algunos componentes y el resto es estimado con la serie de Taylor expandida hasta el primer término diferencial  $y_k$ , como se describe en la ecuación (3) donde  $O(h^2)$  son los términos que se ignoran para estimar la derivada utilizada aplicaciones reales de la ecuación (1).

$$\begin{aligned} \vec{y} &= A(\vec{x}, \vec{u}, t)\vec{\beta}, & \vec{x}(t) &= \vec{y} \\ \vec{x}_{k+1} &= \vec{x}_k + h\vec{y}_k + O(h^2) \end{aligned} \quad (3)$$

Considerando la cuantificación de las señales del vector  $\vec{x}$  que es utilizada por los sensores de posición de los robots:

$$y_q = r \text{floor}\left(\frac{y}{r}\right) \quad (4)$$

$$\text{floor}(z) = \mathbb{N}(z) \quad \forall |z| > 0, \quad \text{floor}(z) = 0 \Leftrightarrow z = 0$$

Donde  $r$  es la resolución y  $\mathbb{N}(z)$  son los valores enteros de  $z$ , se observa que el estimador de la ecuación (3) contiene un error mayor al presentado al ignorar los términos  $O(h^2) = \frac{h^2}{2!} \left(\frac{d}{dt}\vec{y}_k\right) + \dots + \frac{h^n}{n!} \left(\frac{d^{n-1}}{dt^{n-1}}\vec{y}_k\right) + \dots$  de la serie. En la Figura 7 se aprecia una señal senoidal que es cuantificada por la ecuación (4). La señal de color azul no es idéntica a la señal de color negro por lo tanto existe un error inducido que se ve reflejado en sus derivadas con respecto al tiempo. Si se deriva directamente la señal cuantificada  $\vec{x}_q$  de la ecuación (4) como se describe en la ecuación (5), el resultado es un tren de impulsos de frecuencia  $1/t_m$ , lo que ya no representa la información del movimiento esperada. En la Figura 8 se aprecia la derivada de la señal cuantificada de la Figura 7 donde los pulsos de color azul son modelados con la ecuación (5). La derivada de la señal senoidal es la función coseno, sin embargo, con el cuantificador se obtiene un tren de impulsos que no corresponden con la derivada real.

$$\frac{d}{dt}y_q = \lim_{\Delta t \rightarrow 0} \left[ \frac{r}{\Delta t} \sum_{m=0}^{\infty} \delta(t_m) \right] \quad (5)$$

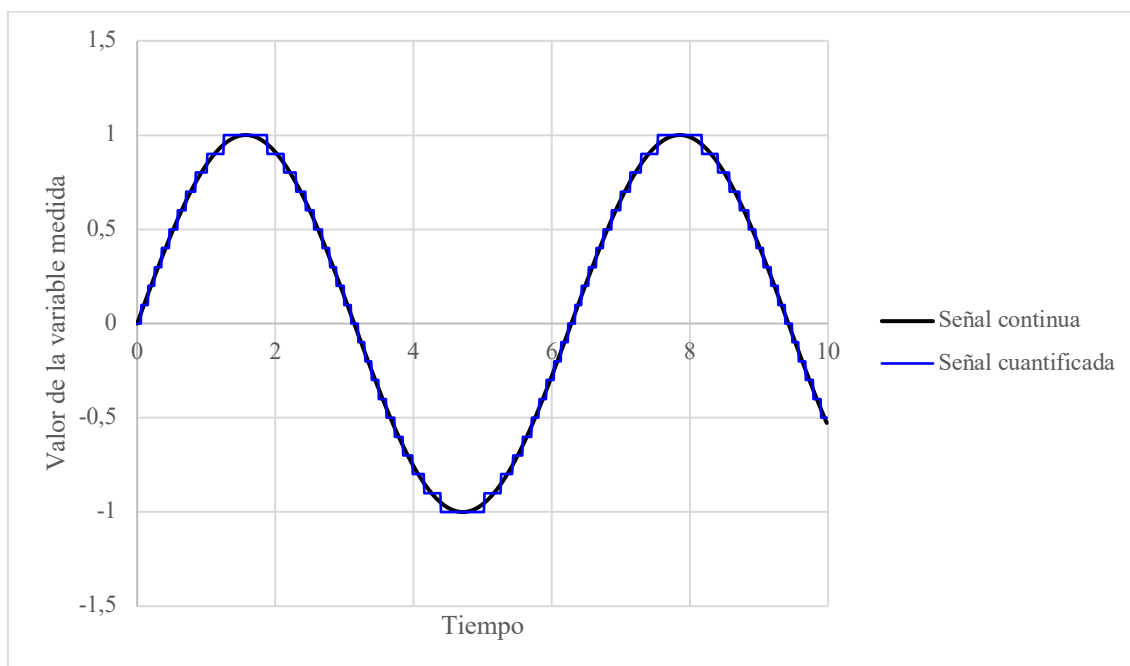


Figura 7. Señal cuantificada descrita por la ecuación (4).

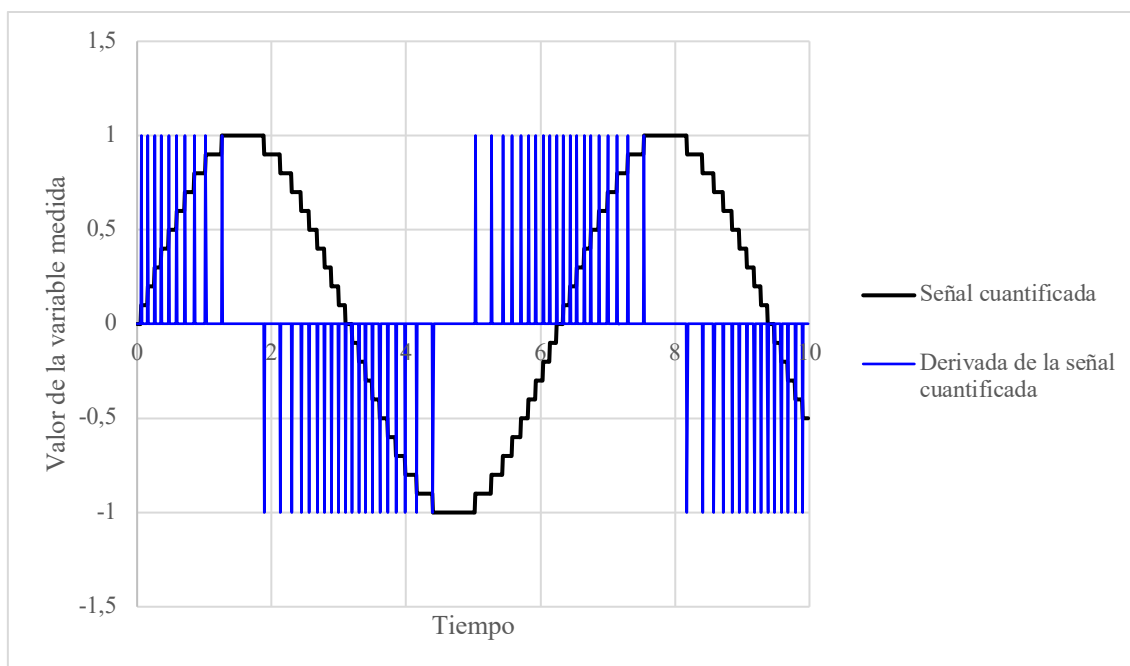


Figura 8. Derivada de la señal cuantificada de la ecuación (4).

En el estado del arte se encuentra la utilización de filtros pasa bajas que están sustentados en el teorema de Nyquist (Ogata, 1996). Sin embargo, es importante mencionar que este teorema únicamente está demostrado cuando las señales son solamente muestreadas en tiempo, pero no en amplitud con un cuantificador modelado por la ecuación (4). Por lo tanto, no se puede afirmar que las señales de las derivadas estimadas con la ecuación (3) utilizando las mediciones de la posición, corresponden a las señales del movimiento real del robot ya que se obtiene el tren de impulsos de la ecuación (5) para la velocidad y aceleración.

Debido a las limitaciones de medir las derivadas de la posición con la precisión suficiente para ser usadas en algoritmos de identificación deterministas, se propone utilizar un algoritmo con una red neuronal artificial que estime los parámetros dinámicos sin necesidad de estimaciones precisas de velocidad y aceleración. En la sección 4 se aborda la metodología propuesta para identificar los parámetros dinámicos de la ecuación (1).

#### 4 Metodología propuesta

Se propone utilizar un algoritmo de identificación con una red neuronal artificial que procese la información del movimiento del robot (solo la que proviene de mediciones de sensores y de los actuadores), y que devuelva los parámetros dinámicos del robot. En la Figura 9 se observa la metodología propuesta dividida en tres fases: diseño preliminar, desarrollo del algoritmo, e implementación.

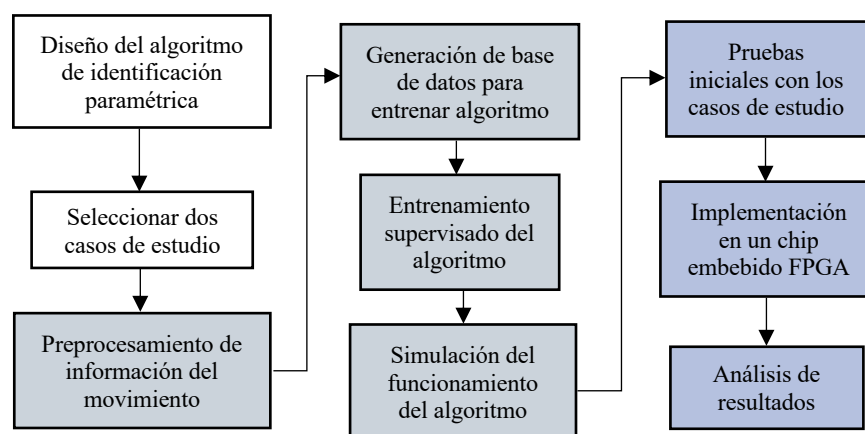


Figura 9. Metodología propuesta para identificar parámetros con una red neuronal.

En la primera fase se crea el diseño preliminar del algoritmo de identificación que incluye la arquitectura de la red neuronal artificial y los pasos preliminares para identificar los parámetros dinámicos. Se seleccionan dos robots manipuladores y sus modelos dinámicos que servirán para crear las bases de datos de simulación para entrenar el algoritmo propuesto. En la segunda fase se diseña el preprocesamiento de la información del movimiento del robot que incluye la forma en que es introducida al algoritmo de identificación.

Para la generación de la base de datos se utiliza la simulación del modelo dinámico de los dos robots manipuladores seleccionados. Debido a la cantidad de parámetros y variables involucradas en un modelo de un robot, la simulación se implementa en C++ donde la ejecución es rápida lo que permite generar miles de datos en poco tiempo. Para entrenar al algoritmo se utiliza el gradiente descendente del error donde la función de costo es la función cuadrática. La base de datos es dividida en dos partes; una para entrenar la red neuronal artificial del algoritmo y otra para validar el entrenamiento de la red. Para corroborar el funcionamiento del algoritmo se simula con datos generados de una simulación con parámetros dinámicos aleatorios.

La tercera fase consiste en realizar pruebas con los dos casos de estudio para observar el funcionamiento del algoritmo y retroalimentar el diseño de este. En esta fase si las pruebas son satisfactorias se implementará el algoritmo en un sistema embebido FPGA con un procesador tipo *soft-core*. En la sección 5 se muestra la primera versión del algoritmo propuesto junto con algunas funciones necesarias para identificar los parámetros dinámicos.



## 5 Resultados

En el [Algoritmo 1](#) se muestra la primera versión del algoritmo de identificación paramétrica. Los datos de entrada son el modelo dinámico, la posición del robot, y el torque aplicado donde  $k$  indica que son mediciones obtenidas por el muestreo de los sensores y actuadores. En la línea 2 se unen las mediciones con las estimaciones para conformar el conjunto  $Y_m$ . Con la función submuestreo de la línea 3 se reduce el número de muestras a una longitud  $n$ . En esta función se pretende que la longitud de entrada y de salida no sean múltiplos. Se debe submuestrear las señales a una cantidad  $n < n_k$  donde  $n_k$  es la longitud de los datos obtenidos.

En la línea 4 se inicializan los parámetros dinámicos  $p$  de longitud  $n_p$  en forma aleatoria con la función inicialización. Dentro de esta función se debe normalizar la inicialización para que no se obtengan parámetros iniciales negativos. En la línea 5 se procesa la información con la función preprocesamiento la cual devuelve en la variable  $M$  los datos preprocesados para ser introducidos a la red neuronal.

Algoritmo 1	
<b>Entrada:</b>	Señales de posición $\vec{q}(k)$ y torque $\vec{\tau}(k)$ del robot, Modelo dinámico $A(\vec{q}, \vec{\tau}, t)\vec{\beta}$ .
<b>Salida:</b>	Parámetros dinámicos del robot.
<b>1:</b>	$Y \leftarrow \vec{q}(k)$
<b>2:</b>	$Y_m \leftarrow Y \cup \vec{\tau}(k) \cup Y_{estimaciones}$
<b>3:</b>	$y_m \leftarrow \text{submuestreo}(Y_m, n)$
<b>4:</b>	$p \leftarrow \text{inicialización}(n_p)$
<b>5:</b>	$M \leftarrow \text{preprocesamiento}(y_m, p)$
<b>6:</b>	$v \leftarrow 0$
<b>7:</b>	<b>while</b> $v < 0.9$ <b>do</b>
<b>8:</b>	$p \leftarrow \text{red neuronal}(M, p) + p$
<b>9:</b>	$v \leftarrow \text{métrica}(p, \vec{q}, \vec{\tau}, \vec{f})$
<b>10:</b>	$M \leftarrow \text{preprocesamiento}(y_m, p)$
<b>11:</b>	<b>devolver</b> $p$

El ciclo *while* de la línea 7 ejecuta la red neuronal para hallar la cantidad que es necesaria para hallar los parámetros dinámicos, ejecuta la métrica de evaluación y genera un conjunto de datos nuevos en caso de que la métrica almacenada en  $v$  no satisfaga el criterio de  $v \geq 0.9$ . Finalmente, el algoritmo cuando se satisface la métrica de evaluación devuelve los parámetros dinámicos.

La red neuronal utilizada continua en fase de desarrollo con el fin de seleccionar la estructura que mejor se adapte para identificar los parámetros dinámicos. En la [Figura 10](#) se encuentra la metodología para entrenar la red neuronal artificial.

## 6 Conclusiones

La revisión del estado del arte ha mostrado las áreas de oportunidad encontradas para la identificación paramétrica de robots manipuladores. El área de oportunidad más importante es que los algoritmos convencionales son deterministas, pero trabajan con datos incompletos que producen demoras en el proceso. En los avances logrados está el diseño del algoritmo de identificación que no requiere de la optimización de una trayectoria de movimiento para el robot manipulador; solo es necesario utilizar una señal senoidal con un controlador proporcional derivativo.

Con la metodología propuesta se pretende poder identificar parámetros dinámicos del mismo tipo de robot con el que se ha entrenado el algoritmo para ser utilizados en procesos en serie como el ajuste de sistemas de control compensados que requieren de los parámetros dinámicos. El trabajo futuro es implementar el algoritmo propuesto y entrenar la red neuronal para los dos casos de estudio seleccionados. Para lograr este paso, es necesario que el modelo dinámico de los robots modele la respuesta de los robots reales con una precisión elevada.

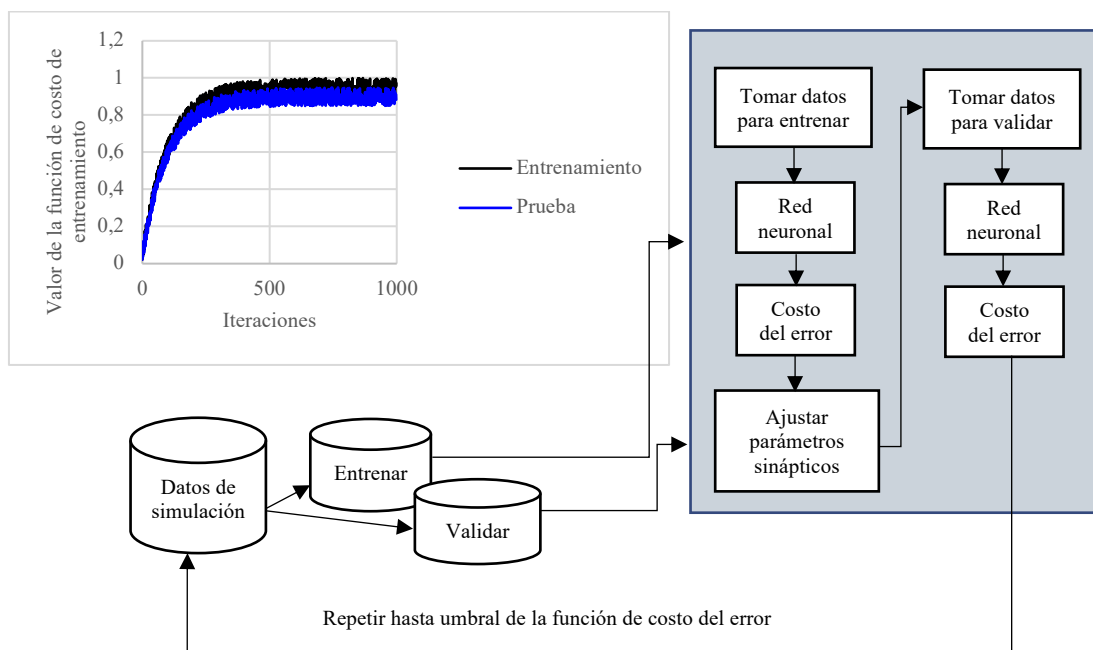


Figura 10. Metodología para entrenar la red neuronal con datos de simulación del modelo del robot.

## Declaración de conflicto de intereses

Los autores declaran no tener conflicto de intereses con respecto a la investigación, autoría o publicación de este artículo.

## Financiación

Este trabajo fue apoyado por el Consejo Nacional de Ciencia y Tecnología (CONACYT) del gobierno de México mediante las becas nacionales de manutención otorgadas a estudiantes que realizan un posgrado inscrito al Programa Nacional de Posgrados de Calidad (PNCP).

## Agradecimientos

Los autores de este trabajo agradecen a la Benemérita Universidad Autónoma de Puebla por el espacio creado para el programa de Doctorado en Ingeniería del Lenguaje y del Conocimiento (*Language & Knowledge Engineering*). Los autores igualmente agradecen a la Universidad Autónoma de Bucaramanga por la publicación de este trabajo en la Revista Colombiana de Computación.

## ORCID iD

Carlos Leopoldo Carreón Díaz de León  <https://orcid.org/0000-0002-9953-9561>

Sergio Vergara Limon  <https://orcid.org/0000-0002-5215-9262>

Juan Manuel Gonzalez-Calleros  <https://orcid.org/0000-0002-9661-3615>

María Aurora Diozcora Vargas Treviño  <https://orcid.org/0000-0001-7188-2782>

## Referencias

- Agand, P., Shoorehdeli, M. A., & Khaki-Sedigh, A. (2017). Adaptive recurrent neural network with Lyapunov stability learning rules for robot dynamic terms identification. *Engineering Applications of Artificial Intelligence*, 65. <https://doi.org/10.1016/j.engappai.2017.07.009>
- Berzal, F. (2018). *Redes Neuronales y Deep Learning*. Ikor.
- Chávez-Olivares, C., Reyes-Cortés, F., González-Galván, E., Mendoza-Gutierrez, M., & Bonilla-Gutierrez, I. (2012). Experimental Evaluation of Parameter Identification Schemes on an Anthropomorphic Direct Drive Robot. *International Journal of Advanced Robotic Systems*, 9(5). <https://doi.org/10.5772/52190>
- Devore, J. L. (2011). *Probability and Statistics for Engineering and the Sciences*. Cengage learning.
- Ding, Y., Zhao, B. Y., & Wu, B. (2014). Structural System Identification with Extended Kalman Filter and Orthogonal Decomposition of Excitation. *Mathematical Problems in Engineering*, 2014. <https://doi.org/10.1155/2014/987694>
- Hu, J., & Xiong, R. (2018). Contact Force Estimation for Robot Manipulator Using Semiparametric Model and Disturbance Kalman Filter. *IEEE Transactions on Industrial Electronics*, 65(4). <https://doi.org/10.1109/TIE.2017.2748056>
- Huynh, H. N., Assadi, H., Rivière-Lorphèvre, E., Verlinden, O., & Ahmadi, K. (2020). Modelling the dynamics of industrial robots for milling operations. *Robotics and Computer-Integrated Manufacturing*, 61. <https://doi.org/10.1016/j.rcim.2019.101852>
- Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1). <https://doi.org/10.1115/1.3662552>
- Lin, F.-J., Teng, L.-T., Chen, C.-Y., & Chang, C.-K. (2008). Robust RBFN Control for Linear Induction Motor Drive Using FPGA. *IEEE Transactions on Power Electronics*, 23(4). <https://doi.org/10.1109/TPEL.2008.924604>
- Liu, N., Li, L., Hao, B., Yang, L., Hu, T., Xue, T., Wang, S., & Shao, X. (2020). Semiparametric Deep Learning Manipulator Inverse Dynamics Modeling Method for Smart City and Industrial Applications. *Complexity*, 2020. <https://doi.org/10.1155/2020/9053715>
- Ogata, K. (1996). *Sistemas de control en tiempo discreto*. Pearson educacion.
- Pohlman, J. T., & Leitner, D. W. (2003). A Comparison of Ordinary Least Squares and Logistic Regression. *The Ohio Journal of Science*, 103(5), 118–125.
- Qin, Z., Baron, L., & Birglen, L. (2010). A new approach to the dynamic parameter identification of robotic manipulators. *Robotica*, 28(4). <https://doi.org/10.1017/S0263574709990233>
- Soh, J., & Wu, X. (2017). An FPGA-Based Unscented Kalman Filter for System-On-Chip Applications. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 64(4). <https://doi.org/10.1109/TCSII.2016.2565730>
- Su, H., Qi, W., Yang, C., Sandoval, J., Ferrigno, G., & Momi, E. de. (2020). Deep Neural Network Approach in Robot Tool Dynamics Identification for Bilateral Teleoperation. *IEEE Robotics and Automation Letters*, 5(2). <https://doi.org/10.1109/LRA.2020.2974445>
- Swevers, J., Ganseman, C., Tukul, D. B., de Schutter, J., & van Brussel, H. (1997). Optimal robot excitation and identification. *IEEE Transactions on Robotics and Automation*, 13(5). <https://doi.org/10.1109/70.631234>
- Swevers, J., Verdonck, W., & de Schutter, J. (2007). Dynamic Model Identification for Industrial Robots. *IEEE Control Systems*, 27(5). <https://doi.org/10.1109/MCS.2007.904659>
- Taylor, J. R. (2013). *Mecánica clásica*. Reverté.
- Wang, S., Shao, X., Yang, L., & Liu, N. (2020). Deep Learning Aided Dynamic Parameter Identification of 6-DOF Robot Manipulators. *IEEE Access*, 8. <https://doi.org/10.1109/ACCESS.2020.3012196>
- Wu, R.-T., & Jahanshahi, M. R. (2019). Deep Convolutional Neural Network for Structural Dynamic Response Estimation and System Identification. *Journal of Engineering Mechanics*, 145(1). [https://doi.org/10.1061/\(ASCE\)EM.1943-7889.0001556](https://doi.org/10.1061/(ASCE)EM.1943-7889.0001556)
- Xiao, W., Dunford, W. G., Palmer, P. R., & Capel, A. (2007). Application of Centered Differentiation and Steepest Descent to Maximum Power Point Tracking. *IEEE Transactions on Industrial Electronics*, 54(5). <https://doi.org/10.1109/TIE.2007.899922>
- You, B., Qiao, M., Xu, J., & Liu, D. (2012). Low-Speed Control of Heavy-Load Transfer Robot with Long Telescopic Boom Based on Stribeck Friction Model. *Mathematical Problems in Engineering*, 2012. <https://doi.org/10.1155/2012/432129>