

Identificación de articulación blanda para brazo robótico mediante redes neuronales

Andrea Toribio, Carlos Relaño, Concepción A. Monje, Santiago Martínez de la Casa, Carlos Balaguer
RoboticsLab en la Universidad Carlos III de Madrid, Avda de la Universidad 30, Leganés, 28911 Madrid, España
email: crelano@ing.uc3m.es

Resumen

La robótica blanda es una rama con gran potencial en el campo de la robótica actual. Presenta grandes ventajas frente a la antigua perspectiva rígida. Sin embargo, su desarrollo se ve limitado por la complejidad del modelado, identificación y control de estos sistemas, debido entre otros factores a su no linealidad. Es en este contexto, donde el uso de redes neuronales, capaces de adaptarse al comportamiento de sistemas muy variados independientemente del conocimiento disponible de los mismos, adquiere relevancia. En el presente trabajo se analiza la identificación de una articulación robótica blanda mediante redes neuronales, comparando los resultados frente a los obtenidos a través de la identificación mediante funciones de transferencia.

Palabras clave: Redes neuronales, robótica blanda, identificación de sistemas, inteligencia artificial, aprendizaje automático.

1. Introducción

La robótica blanda es un área de investigación que está ganando importancia y reconocimiento debido a las posibilidades que ofrece frente a la robótica tradicional de eslabones rígidos, motores y engranajes. Entre las principales ventajas se encuentran el aumento de los grados de libertad, su modularidad y escalabilidad, así como su diseño que permite reconfiguraciones [3] y ofrece soluciones a los desafíos actuales en robótica.

El uso de nuevos materiales y la inspiración en organismos biológicos aporta la flexibilidad y adaptabilidad necesaria para operar en espacios o situaciones donde otros robots no tienen un fácil acceso. Por ejemplo, en la inspección de tuberías [19], en medios acuáticos o terrestres en los que el terreno es irregular [12] o en el interior del cuerpo humano para la realización de cirugías mínimamente invasivas [8]. Por otra parte, la robótica blanda supone un avance en los robots humanoides, permitiendo que estos se asemejen cada vez más a su equivalente biológico. Además, se consigue una interacción

entre robot y humano más segura gracias al uso de materiales blandos. En [11] se evalúa una plataforma de robótica blanda que funciona mediante un sistema de inflado y utiliza sensores de presión blandos para detectar colisiones. El trabajo desarrollado en [15], expone el diseño y validación de un cuello blando para el robot humanoide TEO del Robotics Lab [13], el cual consiste en una plataforma muy similar a la que se va a estudiar en este trabajo.

También existen dificultades en la robótica blanda como su cinemática y control, lo que causa que la precisión y fiabilidad de estos robots se encuentren por debajo de las proporcionadas por sus semejantes rígidos. Estas desventajas tienen mayoritariamente su origen en la gran no linealidad del sistema a raíz de sus características físicas. Debido a esto, uno de los enfoques que más se utiliza es la realización de suposiciones y simplificaciones que permiten linealizar el sistema para que pueda ser controlado de manera más sencilla. Estas características causan que pueda ser complicada la identificación de sistemas para obtener el modelo matemático del sistema dinámico que se está estudiando. Tener un modelo ajustado es esencial en el diseño de controladores que puedan modificar el comportamiento del sistema para que éste se ajuste a una referencia deseada. Sin embargo, la identificación de sistemas también juega un importante papel en la optimización, supervisión y diagnóstico de procesos [4], así como en la predicción y simulación del sistema real [17].

2. Estado del arte

Uno de los métodos más utilizados en identificación de sistemas consiste en determinar los parámetros de una función de transferencia que relacione la salida y la entrada del sistema. Sin embargo, este método sólo puede ser utilizado con sistemas lineales e invariantes en el tiempo (LTI o linear time-invariant systems), y con sistemas que solo tienen una entrada y una salida (SISO o single input single output systems) [10]. Por otro lado, se encuentran los modelos de espacio de estados que sí pueden modelar sistemas no lineales, variantes en el tiempo e incluso con varias entra-

das y salidas. Sin embargo, su principal desventaja es la ineficiencia computacional que presentan para resolver problemas donde se tienen que estimar parámetros de alta dimensionalidad [10].

Una de las técnicas utilizadas actualmente en robótica blanda para identificación es usar Machine Learning. Este es otro enfoque para obtener la identificación o modelo del sistema como se expone en [21], y donde a partir de datos recogidos del sistema, se pueden emplear métodos como las redes neuronales para entrenarlas y obtener un modelo no lineal más ajustado al sistema. Este enfoque, de hecho, está relacionado con otro de los desafíos de la robótica actual identificados en [20] que es la implementación de la inteligencia artificial.

M. Giorelli et al. realizan en [6] y [5] la implementación de una red neuronal de tipo feedforward (FNN) para la resolución de la cinemática inversa de un robot blando con forma de tentáculo que se caracteriza por una curvatura no constante. Este manipulador, similar al que se estudia en este trabajo, también es dirigido mediante cables. Información más detallada sobre la arquitectura de la red y los parámetros utilizados se puede encontrar en [7].

Finalmente, tras comparar y analizar diferentes tipos de identificaciones, en este artículo se aborda la implementación de una red del tipo feedforward multicapa para la identificación de un brazo robótico blando actuado por cable. Estos resultados serán comparados con los obtenidos de la identificación realizada a través de función de transferencia.

3. Material y método

3.1. Descripción de la plataforma robótica blanda a identificar

El modelo que se pretende identificar consiste en una articulación robótica blanda. Se trata de un prototipo físico realizado en Robotics Lab [16], y a continuación, se va a detallar y explicar su diseño y funcionamiento.

Se trata de una articulación blanda conformada por una plataforma rígida, elementos de fijación, tres motores que controlan los tendones que actúan sobre el sistema y el cuerpo flexible de la articulación blanda, que es el elemento que realiza la flexión y ha sido elaborado a partir del material NinjaFlex, utilizando la impresora Creatobot600 pro. En la figura 1 se muestra un esquema del prototipo, identificándose en él las diferentes partes mencionadas. Gracias a esta configuración, es posible situar el extremo de la articulación en

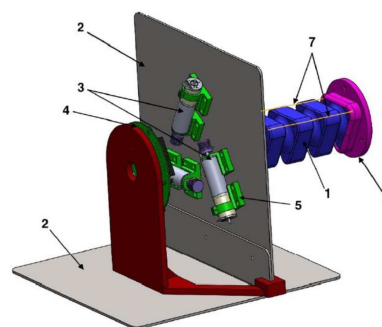


Figura 1: Esquema de la plataforma robótica con sus elementos: articulación blanda (1), placa metálica (2), motores (3), dispositivos electrónicos (4), soportes y elementos de fijación (5), base de la articulación (6) y tendones (7).

el espacio tridimensional, teniendo dos grados de libertad: cabeceo (pitch) y guiñada (yaw), como se observa en la figura 2.

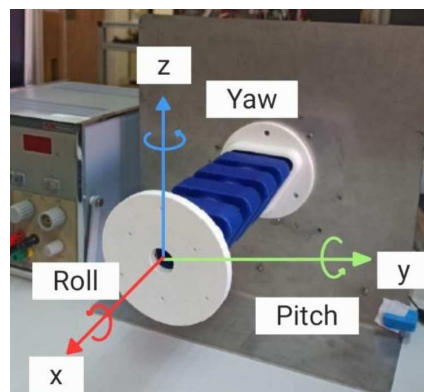


Figura 2: Representación de la articulación blanda a identificar y los ángulos de flexión.

Los tres motores utilizados para la actuación son Maxon EC- max 22. Para controlar su posición se utiliza el servo-variador inteligente Technodoft's Intelligent Drives iPOS 4808 MX, el cual envía y recibe señales del ordenador a través de busCAN. Para tomar los datos de los ángulos de pitch y yaw se utiliza el sensor 3DM-GX5-10. Este sensor inercial está situado en el extremo de la articulación y su hoja de especificaciones se puede consultar en [1]. Los datos se toman con una frecuencia de 50 Hz.

La geometría de la articulación blanda consiste en varios prismas triangulares unidos entre sí, dejando una apertura cilíndrica en el interior de la articulación. Además de esta sección, en los vértices de cada prisma hay un orificio por donde se introducen los tendones correspondientes. En las figuras 3 y 4 se muestra la geometría de la arti-

culación desde diferentes vistas, así como la numeración de los orificios que se relaciona con la numeración de los tendones y motores, siendo el motor 1 (m1) el que se encuentra en el eje Y con dirección negativa, y el resto espaciado a 120° en sentido contrario a las agujas del reloj.

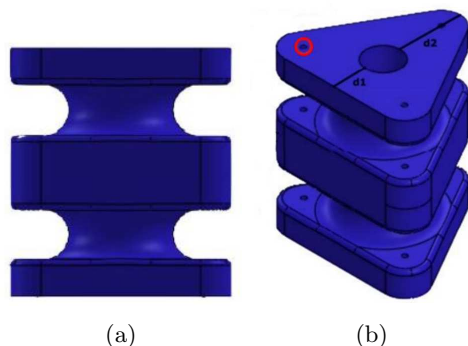


Figura 3: Vista frontal (a) y en perspectiva (b) de la articulación, apareciendo rodeado de color rojo uno de los orificios por donde pasan los tendones.

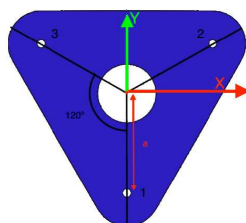


Figura 4: Vista superior de la articulación, donde se muestra la distribución de los tendones y motores numerados.

Finalmente nos encontramos ante un sistema formado por tres entradas, que son las longitudes de los tendones, y dos salidas, que son el ángulo pitch y yaw del extremo del brazo respecto a la base fija.

3.2. Obtención de los datos

Como se ha mencionado previamente, la identificación de un robot blando, como es este caso, puede resultar compleja debido a que se trata de un sistema no lineal. Una aproximación sería utilizar el método de modelado por función de transferencia, obteniendo un sistema lineal simplificado. Sin embargo, el sistema que se va a estudiar no sólo es no lineal, sino que también es un sistema MIMO (múltiples entradas y salidas).

Para abordar este problema es necesario realizar primero un desacoplado de los ángulos de pitch y yaw, de manera que se consigan dos sistemas SISO (sólo una entrada y una salida) más sencillos. Este enfoque, también utilizado en el control del cuello blando predecesor de esta articulación, demuestra tener buenos resultados [18] [14].

Al aplicar el método de desacople descrito en [14], se obtienen dos valores de referencia, alpha y beta, ecuaciones (1-3). Estos valores se relacionan con la variación de longitud del tendón, L_i , para una salida en una dirección de pitch y una salida de yaw respecto a las entradas alpha y beta, respectivamente.

$$L_1 = \frac{\alpha}{1,5} \tag{1}$$

$$L_2 = \frac{\beta}{1,732} - \frac{\alpha}{3} \tag{2}$$

$$L_3 = -\frac{\beta}{1,732} - \frac{\alpha}{3} \tag{3}$$

También se observa que para realizar un movimiento en la dirección de flexión pitch, el valor de alpha se relaciona con los tres motores. Sin embargo para realizar un movimiento en la dirección de flexión yaw, el valor de L_1 no se ve involucrado debido a que por la posición del motor 1 no genera flexión en esa dirección.

Teniendo en cuenta que la longitud de los tendones cuando estos están totalmente en reposo es $l_0 = 0,2m$ y que el radio del cilindro sobre el que se enrollan los cables es $R = 9,3mm$, se puede calcular la posición angular Ω a partir de la longitud del tendón actual L_i aplicando la ecuación (4) [16].

$$\Omega = \frac{(L_o - L_i)}{R} = \frac{0,2 - L_i}{9,3 \cdot 10^{-3}} \tag{4}$$

El procedimiento para obtener los datos consiste en ir variando los valores de alpha y beta, de 10 en 10, e introducir los valores L_i en el sistema para observar a qué ángulos de salida corresponden. Como límites inferior y superior de alpha y beta se tienen -50 y 50 para evitar que la plataforma sufra algún daño por un ángulo excesivo, y se añade ruido a la señal de referencia, de entre ± 5 . Para cada pareja de valores de alpha y beta, se parte de la posición de reposo, se manda una señal escalón para los valores de referencia deseados a la articulación y se vuelve a la posición inicial. Así, durante el primer segundo los valores de referencia son igual a 0 grados, desde el segundo 1 y durante 5 segundos se dan los valores de referencia correspondientes, y desde el segundo 6 hasta el segundo 11 aproximadamente, se vuelven a poner las referencias a 0, tiempo durante el que la articulación se vuelve a estabilizar en la posición inicial.

3.3. Identificación mediante función de transferencia

Para la identificación se proporcionan unas entradas de referencia donde una de ellas siempre tenga el valor de 0 y se toma como salida sólo el ángulo que es distinto de cero, siendo un total de 21 test. Con estos datos, se utiliza el programa MATLAB para estimar cada una de las funciones de transferencia sin ningún cero y con dos polos, por medio de la función tfest, y se obtiene una función de transferencia para cada pareja de valores alpha y beta.

Las respuestas ante entrada escalón para cada función de transferencia obtenida se pueden apreciar en las figuras 5 y 6. Estas gráficas tienen como eje horizontal el tiempo transcurrido, y como eje vertical el ángulo correspondiente teóricamente alcanzado. Se puede observar la no linealidad del sistema especialmente en pitch, donde los valores positivos y negativos de alpha alcanzan posiciones diferentes.

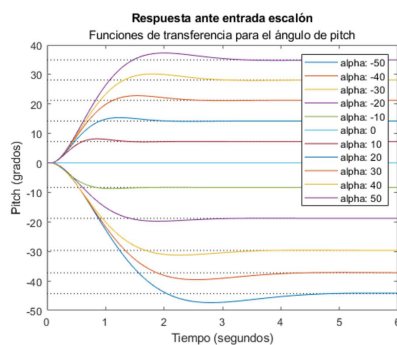


Figura 5: Respuesta de 11 funciones de transferencia ante entrada escalón alpha del sistema modelado según las funciones de transferencia para el ángulo de pitch, donde alpha varía y beta es constante igual a 0.

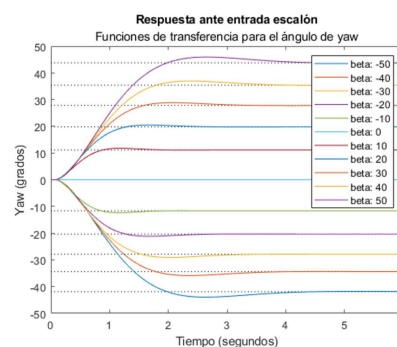


Figura 6: Respuesta de 11 funciones de transferencia ante entrada escalón beta del sistema modelado según las funciones de transferencia para el ángulo de yaw, donde beta varía y alpha es constante igual a 0.

Finalmente, se elabora una media de las funciones obtenidas, resultando en una función de transferencia para el ángulo de pitch y otra para el ángulo de yaw. Estas se muestran en la ecuación (5) y la ecuación (6), respectivamente. Se elige usar solo estos 21 test de las 121 posibles combinaciones debido a que se obtiene una mejor identificación.

$$TF \text{ media alpha} = \frac{6,5717}{s^2 + 3,4875s + 8,4294} \quad (5)$$

$$TF \text{ media beta} = \frac{7,0101}{s^2 + 3,4919s + 6,8441} \quad (6)$$

3.4. Identificación mediante redes neuronales

Se decide trabajar con redes feed-forward multicapa, considerando en principio una sola capa oculta con un número finito de neuronas [9]. Por tanto, una vez elegida la técnica de Machine Learning y el algoritmo que se va a utilizar, es necesario seleccionar la herramienta de software más adecuada. En este caso, se han considerado las ventajas que ofrece MATLAB en el procesamiento de datos y la aplicación Neural Net Fitting. Esta aplicación facilita todo el proceso de implementación de redes neuronales: desde la importación, procesamiento y división de los datos en los conjuntos de entrenamiento, test y validación, hasta la evaluación de los resultados y la generación de scripts, funciones y modelos de MATLAB y Simulink para su posterior uso. Además, permite elegir parámetros de la red como el número de neuronas de la capa oculta y el algoritmo de entrenamiento [2].

El procedimiento de toma de datos es idéntico al utilizado para la identificación por funciones de transferencia, ya explicado en la sección. En este caso se usaran todos los test, y la división de los datos por defecto será aleatoria y supone un 70 %, un 15 % y otro 15 % para los datos de entrenamiento, validación y test, respectivamente. El algoritmo que se usará para el entrenamiento es Levenberg-Marquardt Backpropagation y tras experimentación previa se decide usar 30 neuronas en la capa oculta, que es un número adecuado y suficiente para el problema analizado.

Los resultados obtenidos son una serie de valores de pitch y yaw en el tiempo. Para este caso, la red neuronal necesita que la entrada se un valor único, por lo que se elige el valor en el instante 3.5 segundos, debido a que se considera que ha pasado tiempo suficiente para llegar a la referencia.

Las redes que se realizarán son las siguientes: una red para identificar pitch y yaw a partir de los

datos de posición de los motores y una red para identificar la posición de los motores a partir de los datos de pitch y yaw, como se observa en la tabla 1.

Cuadro 1: Entradas y salidas de las redes neuronales para la identificación.

Tipo	Entradas	Salidas
Directa	Posición m1	Pitch y Yaw
	Posición m2	
	Posición m3	
Inversa	Pitch y Yaw	Posición m1 Posición m2 Posición m3

4. Resultados

Se realiza de nuevo la obtención de datos para realizar una medición y comparación de las identificaciones obtenidas. Siguiendo el procedimiento descrito anteriormente se obtienen los resultados para valores de alpha y beta desde -50 a 50 en saltos de 5, sin ruido, por lo que existen nuevas combinaciones que no han sido usadas en el entrenamiento ni en la validación. Estos datos se compararán en ambas identificaciones.

4.1. Resultados de identificación mediante redes neuronales

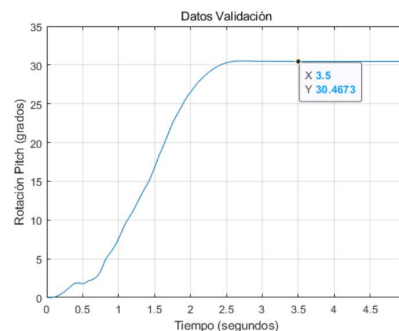
Del mismo modo que en el entrenamiento, se van a comparar los resultados en el segundo 3.5, como se comentó anteriormente, figura 7(a), para la comprobación de los 441 valores de alpha y beta. Se denomina caso a cada combinación de estos valores y se representan todos los valores de beta para un valor de alpha constante, se muestra un ejemplo de la disposición en la figura 7 (b).

Tras combinar los 441 casos, se muestran los resultados de la red neuronal directa en las figuras 8 y 9. El mismo procedimiento es seguido para la red neuronal inversa, figuras 10, 11 y 12.

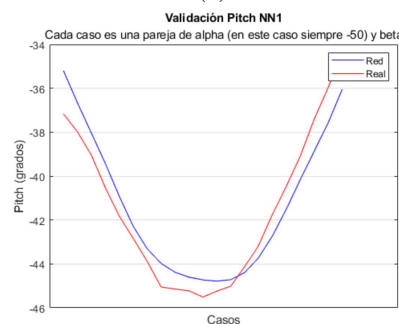
Finalmente se muestran los resultados comparados con la identificación realizada por funciones de transferencia, figuras 13 y 14.

4.2. Resultados de ajuste de las identificaciones obtenidas

Con el fin de relacionar los errores obtenidos con la dimensión que se está midiendo, se utilizan los errores normalizados. En este caso, se ha realizado esta normalización mediante la división de los errores ya obtenidos por la media de los valores reales. Así, se han obtenido la normalización



(a)



(b)

Figura 7: (a) Valor de pitch en el segundo 3.5, en un caso alpha =50 y beta =50. (b) Representación de lo valores de pitch en el segundo 3.5, en los casos alpha=-50 y la respuesta de la red neuronal directa.

de la raíz cuadrada del error cuadrático medio o NRMSE, y el error normalizado absoluto medio o NMAE (ambos expresados en porcentajes). Las ecuaciones utilizadas para su cálculo se muestran en la ecuación 7 y la ecuación 8.

$$NRMSE = 100 \% \cdot \frac{\sqrt{\frac{1}{N} \sum_{i=1}^n (y_i - \hat{y}_i)^2}}{\bar{y}} \tag{7}$$

$$NMAE = 100 \% \cdot \frac{\frac{1}{N} \sum_{i=1}^n |y_i - \hat{y}_i|}{\bar{y}} \tag{8}$$

Los errores obtenidos de las identificaciones usando el método de ajuste NRMSE para la red neuronal directa han sido de 16,9451 % y 7,4802 % para pitch y yaw, lo que significa en un ajuste del 83,0549 % y 92,5198 % respectivamente; y para la función de transferencia media 18.1103 % y 22.4517 %, lo que significa en un ajuste del 81,8897 % y 77,5483 % para pitch y yaw, respectivamente.

Los errores obtenidos de las identificaciones usando el método de ajuste NMAE para la red neuronal directa han sido de 14,6525 % y 5,6512 %,



Figura 8: Resultados de la red neuronal directa para pitch frente a los valores reales.

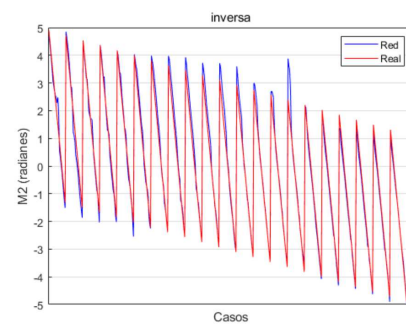


Figura 11: Resultados de la red neuronal inversa para el motor 2 frente a los valores reales.

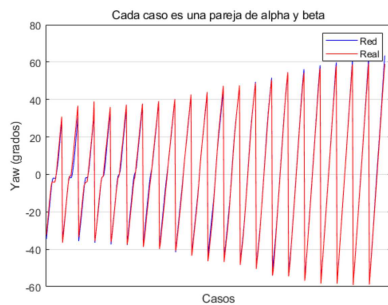


Figura 9: Resultados de la red neuronal directa para yaw frente a los valores reales.

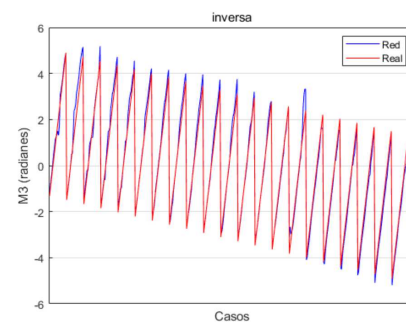


Figura 12: Resultados de la red neuronal inversa para el motor 3 frente a los valores reales.

lo que significa en un ajuste del 85,3475 % y 94,3488 % para pitch y yaw; y para la función de transferencia media 14.4262 % y 28.8976 %, lo que significa en un ajuste del 85,5738 % y 71,1024 % para pitch y yaw, respectivamente.

Los errores obtenidos de las identificaciones usando el método de ajuste NRMSE para la red neuronal inversa han sido de 22,8934 % y 17,5148 % y 16,9855 %, lo que significa en un ajuste del 77,1066 %, 82,4852 % y 83,0145 % para m1, m2 y m3; y usando el método de ajuste NMAE 17,8348 % 12,3473 % y 12,8757 % lo que significa en un ajuste del 82,1652 %, 87,6527 %, y 87,1243 % para m1, m2 y m3, respectivamente.

Los resultados de la función de transferencia son

lineales y no tienen en cuenta la influencia que tienen entre sí los ángulos de pitch y yaw, ajustándose, por ello, peor al sistema. Además, se puede observar en ellas que los errores respecto a los resultados del sistema real son mayores que los obtenidos para las redes neuronales. Se puede observar que los resultados presentan una mejoría clara en el caso de yaw. Para el caso de pitch, aunque el error no es mucho menor que el obtenido con la función de transferencia media de alpha, se ha podido observar, según las figuras citadas, que la red se ajusta mejor a la forma de la señal, por lo que los resultados también han sido exitosos.

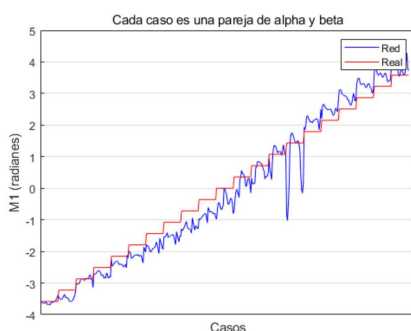


Figura 10: Resultados de la red neuronal inversa para el motor 1 frente a los valores reales.

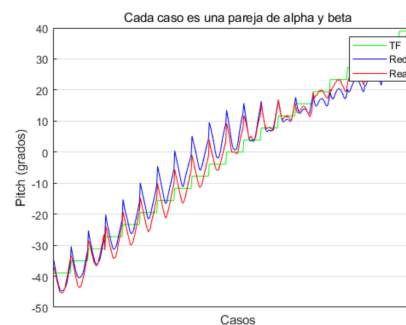


Figura 13: Resultados de la red neuronal directa para pitch frente a los valores reales y la función de transferencia media para alpha.

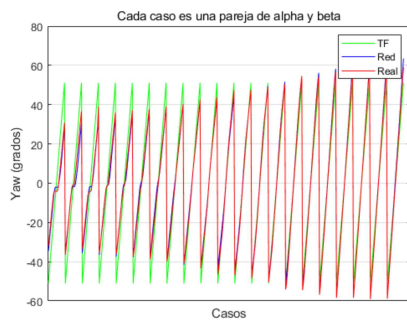


Figura 14: Resultados de la red neuronal directa para yaw frente a los valores reales y la función de transferencia media para beta.

5. Conclusiones

Se ha obtenido una identificación de la articulación mediante funciones de transferencia y redes neuronales, en este último caso tanto para la cinemática directa como la inversa. Tras validarse en la plataforma, se ha comprobado que la identificación directa por redes neuronales se ajusta mejor al sistema que la identificación mediante funciones de transferencia, obteniéndose en todas ellas un porcentaje de ajuste superior al 75 %.

Se ha demostrado que las neuronales son una alternativa al modelado de articulaciones blandas, generalmente no lineales, aun no teniendo demasiada información sobre la cinemática de las mismas. Además, la identificación por redes neuronales se puede considerar más sencilla al no requerir de la combinación de diferentes funciones de transferencia y puede trabajar con el sistema MIMO sin necesidad de desacoplarlo en varios sistemas SISO. Por otro lado, se comporta mejor frente a la no linealidad y no simetría del sistema, especialmente presente en el caso del ángulo de pitch. Por otro lado, estas redes neuronales solo presentan la cinemática del sistema en el punto de interés, mientras que las funciones de transferencia muestran mucha más información de la dinámica del sistema.

Las líneas futuras de trabajo que se plantean son la mejora de la identificación del sistema y la implementación de la red neuronal para el diseño de un controlador. Además, podrían recogerse más datos para el entrenamiento de las redes, explorando más posiciones de la articulación. Adicionalmente, podrían probarse otras configuraciones, tipos de redes o, incluso, redes de Deep Learning.

Agradecimientos

La investigación que ha conducido a estos resultados ha recibido financiación del proyecto SOFIA: Articulación blanda inteligente con capacidades de reconfiguración y modularidad para plataformas

robóticas, con referencia PID2020-13194GB-I00, financiado por el Ministerio de Economía, Industria y Competitividad, y del proyecto Desarrollo de articulaciones blandas para aplicaciones robóticas, con referencia IND2020/IND-1739, financiado por la Comunidad Autónoma de Madrid (CAM) (Departamento de Educación e Investigación). La investigación que ha conducido a estos resultados ha recibido financiación de RoboCity2030-DIH-CM, Madrid Robotics Digital Innovation Hub, S2018/NMT-4331, financiado por "Programas de Actividades I+D en la Comunidad de Madrid" cofinanciado por los Fondos Sociales Europeos (FSE) de la UE.

English summary

Soft joint identification for a robotic arm using neural networks

Abstract

Soft robotics is a branch with great potential in the field of robotics today. It has great advantages over the old rigid perspective. However, its development is limited by the complexity of modelling, identification and control of these systems, due, among other factors, to their non-linearity. It is in this context that the use of neural networks, capable of adapting to a wide variety of systems, regardless of the knowledge available about them, is relevant. This paper analyses the identification of a soft robotic joint by means of neural networks, comparing the results with those obtained by means of transfer function identification.

Keywords: neural networks, soft robotics, system identification, artificial intelligence, machine learning.

Referencias

- [1] LORD Microstrain 3DM-GX5-10. <https://bit.ly/3Nv0rwv>. [Online; accessed 23-July-2022].
- [2] Neural Net Fitting. <https://bit.ly/3xtxKvF>. [Online; accessed 23-July-2022].
- [3] Soft intelligent articulation with reconfiguration and modularity capabilities for robotic platforms. <https://bit.ly/3Nk4gXn>. [Online; accessed 23-July-2022].
- [4] Javier Sedano Franco and José Ramón Villar Flecha. Introducción a la identificación de sistemas. *Técnica Industria 256l*, 2005.

- [5] Michele Giorelli, Federico Renda, M Calisti, A Arienti, G Ferri, and C Laschi. Learning the inverse kinetics of an octopus-like manipulator in three-dimensional space. *Bioinspiration & biomimetics*, 10(3):035006, 2015.
- [6] Michele Giorelli, Federico Renda, Marcello Calisti, Andrea Arienti, Gabriele Ferri, and Cecilia Laschi. Neural network and jacobian method for solving the inverse statics of a cable-driven soft arm with nonconstant curvature. *IEEE Transactions on Robotics*, 31(4):823–834, 2015.
- [7] Michele Giorelli, Federico Renda, Gabriele Ferri, and Cecilia Laschi. A feed forward neural network for solving the inverse kinetics of non-constant curvature soft manipulators driven by cables. In *Dynamic Systems and Control Conference*, volume 56147, page V003T38A001. American Society of Mechanical Engineers, 2013.
- [8] Quanwei Hao, Zixing Li, Hao Yan, Guanjie Li, and Baiquan Su. A natural orifice soft robot with novel driven method for minimally invasive surgery (mis). In *2017 2nd Asia-Pacific Conference on Intelligent Robot Systems (ACIRS)*, pages 289–294. IEEE, 2017.
- [9] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. *Neural networks*, 3(5):551–560, 1990.
- [10] Karel J Keesman and Karel J Keesman. *System identification: an introduction*, volume 2. Springer, 2011.
- [11] Taekyoung Kim, Sohee John Yoon, and Yong-Lae Park. Soft inflatable sensing modules for safe and interactive robots. *IEEE Robotics and Automation Letters*, 3(4):3216–3223, 2018.
- [12] Aman Malhotra, Vivek Jagannath, Prabhat Kumar, Sahil Sanil, J Vighneswar, Advay S Pethakar, and M Sangeetha. Design, fabrication & control of 4-arm soft robot for terrestrial and underwater locomotion. In *2021 8th International Conference on Computing for Sustainable Global Development (INDIA-Com)*, pages 685–689. IEEE, 2021.
- [13] Santiago Martínez, Concepción Alicia Monje, Alberto Jardón, Paolo Pierro, Carlos Balaguer, and Delia Munoz. Teo: Full-size humanoid robot design powered by a fuel cell system. *Cybernetics and Systems*, 43(3):163–180, 2012.
- [14] Jorge Muñoz, Francesco Piqué, Concepción A. Monje, and Egidio Falotico. Robust fractional-order control using a decoupled pitch and roll actuation strategy for the i-support soft robot. *Mathematics*, 9(7):702, 2021.
- [15] Luis Nagua, C Monje, J Muñoz Yañez-Barnuevo, and Carlos Balaguer. Design and performance validation of a cable-driven soft robotic neck. In *Proc. Actas de las Jornadas Nacionales de Robtica*, pages 1–5, 2018.
- [16] Luis Nagua, Carlos Relaño, Concepción A Monje, and Carlos Balaguer. A new approach of soft joint based on a cable-driven parallel mechanism for robotic applications. *Mathematics*, 9(13):1468, 2021.
- [17] Oliver Nelles. *Nonlinear system identification: from classical approaches to neural networks, fuzzy models, and gaussian processes*. Springer Nature, 2020.
- [18] Fernando Quevedo, Jorge Muñoz Yañez-Barnuevo, Juan A Castano, Concepción A Monje, and Carlos Balaguer. Model identification of a soft robotic neck. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8640–8645. IEEE, 2020.
- [19] Yavuz Sapmaz, Savas Dilibal, Orhan Yilmaz, and Ali Riza Sapmaz. Nickel-titanium sma springs actuated bioinspired soft robot for pipeline inspections. In *2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pages 1–5. IEEE, 2021.
- [20] Guang-Zhong Yang, Jim Bellingham, Pierre E Dupont, Peer Fischer, Luciano Floridi, Robert Full, Neil Jacobstein, V Kumar, M McNutt, R Merrifield, et al. The grand challenges of science robotics. *science robotics*, 3 (14), eaar7650, 2018.
- [21] Gang Zheng. A simple strategy for the control of soft robot. In *2020 39th Chinese Control Conference (CCC)*, pages 1751–1755. IEEE, 2020.



© 2022 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution CC BY-NC-SA 4.0 license (<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>).