

Design and construction of a low-cost robotic system for teaching and research^{*}

Julio C. Flores^{*} Alejandro Gutierrez-Giles^{**}

^{*} *Facultad de Ingeniería-UNAM, Av. Universidad 3000, Ciudad Universitaria, Coyoacan, CDMX, 04510, Mexico, (e-mail: julszero@gmail.com)*

^{**} *Centro de Estudios en Computación Avanzada-UNAM, Av. Universidad 3000, Ciudad Universitaria, Coyoacan, CDMX, 04510, Mexico, (e-mail: alejandro-giles@cecav.unam.mx).*

Resumen: The design and construction of a signal-acquisition and control platform for its use to control a robotic manipulator is proposed in this work. The system consists of an open-loop architecture robotic arm with DC motors and encoders, whose signals are acquired and processed using a standard PC. This platform can be potentially used for research and teaching purposes. The main contribution is its low-cost modular design with widely available components, which permits to scaling it for manipulators with several degrees of freedom. The performance of the designed system is tested by several experiments with different nonlinear controllers.

Keywords: Robots manipulators, Robotics technology, Mechatronic systems, Hardware/software co-design, Reliable measurement and actuation.

1. INTRODUCCIÓN

En los últimos años, la investigación de algoritmos de control avanzados ha tenido un desarrollo importante en la comunidad de control y robótica a nivel mundial. Particularmente, a partir de la década de 1980 se introdujeron controladores no lineales basados en las propiedades del modelo Lagrangiano (e.g. pasividad, parametrización lineal, cotas), en los trabajos seminales de Slotine (1985); Slotine and Li (1987); Ortega and Spong (1989). Desde entonces, una parte importante del trabajo de investigación reside en validar la teoría propuesta mediante experimentos confiables o simulaciones. Para muchos investigadores, las simulaciones han sido una herramienta valiosa para validar rápidamente la teoría propuesta dada la dificultad de implementación, el tiempo, o la falta del equipo necesario para realizar experimentos. No obstante, el uso de simulaciones puede llevar a una sobre simplificación del problema, creando una brecha entre la teoría y la práctica cuando se quiere llevar a un escenario real. Esta misma situación se presenta en la enseñanza de robótica industrial a nivel superior, particularmente en los cursos en los que se hace énfasis en el control basado en modelo. En ambos casos, un *benchmark* comúnmente utilizado es el robot planar de dos grados de libertad con articulaciones de revolución. El uso de este manipulador como *benchmark* se debe a que en él se pueden apreciar efectos no lineales como los pares debido a fuerzas centrífugas y de Coriolis y los pares debidos a la gravedad.

Recientemente se han desarrollado robots de bajo costo con fines educativos y de investigación. Por ejemplo, motivados por la pandemia de Covid-19 en Benitez et al. (2020) se propone un sistema basado en Arduino con un brazo manipulador impreso en PLA. Está enfocado totalmente a la educación, particularmente a la cinemática. De manera similar, en Elfasakhany et al. (2011) se propone un sistema basado en Arduino con un brazo antropomórfico de 3 grados de libertad. Este sistema también está enfocado al aprendizaje de cinemática directa e inversa. En Kim and Song (2013) se presenta un diseño de un robot de 3 grados de libertad con contrapesos de bajo costo controlado mediante servomotores. Los sistemas mencionados anteriormente no están diseñados para la implementación de controladores no lineales, por lo que tienen periodos de muestreo relativamente altos. Por otro lado, en Maldonado et al. (2018); Ramos et al. (2019) se propone un sistema de bajo costo que permite implementar controladores no lineales con un periodo de muestreo de 1 [ms] en un mecanismo de 1 grado de libertad. Sin embargo, la adquisición de señales está basada en la tarjeta *Sensaray 626*, lo que eleva el costo total del sistema y lo hace menos portátil.

En este trabajo se propone una plataforma de adquisición de señales y de control de bajo costo para robots manipuladores cuyos actuadores son motores eléctricos equipados con encoders para medir la posición articular. Se hace énfasis en la arquitectura abierta del sistema, en la que las señales de salida (posiciones articulares) y de entrada (voltajes en los motores) del robot son procesadas en una computadora personal (PC, por sus siglas en inglés) estándar. Lo anterior permite al sistema

^{*} Este trabajo fue desarrollado en el marco del proyecto DGAPA-UNAM número IN117820. El segundo autor agradece al Programa de Becas Postdoctorales DGAPA-UNAM.

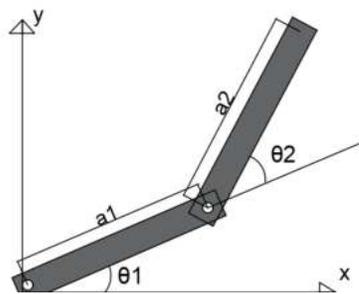


Figura 1. Manipulador de 2 grados de libertad.

ser flexible para realizar experimentos rápidamente, tanto en la programación de nuevos algoritmos como en la sintonización de los mismos, sin la necesidad de grabarlos primero en un microcontrolador (sistemas embebidos), con las limitaciones intrínsecas que esto implicaría. Adicionalmente, otra ventaja de procesar las señales en una computadora personal es que estas pueden ser guardadas en mayor cantidad, limitadas sólo por la memoria RAM de la PC, para ser procesadas posteriormente, e.g. para identificación de modelos, presentación en gráficas, entre otras. El diseño propuesto es del tipo modular, en el sentido de que se puede fácilmente escalar para robots manipuladores de cualquier número de grados de libertad. Asimismo, se consideró para el diseño emplear componentes ampliamente disponibles en el mercado y de bajo costo, pero con la calidad y prestaciones necesarias para obtener un equipo confiable y duradero. En el trabajo presente, se construyó un robot planar de dos grados de libertad con articulaciones de revolución dado que, como se mencionó anteriormente, es un *benchmark* comúnmente utilizado tanto en investigación como en docencia. No obstante, el mismo diseño se puede utilizar para robots de cualquier número de grados de libertad, limitados sólo por las conexiones físicas (puertos de entrada/salida de los microcontroladores utilizados). En la configuración presentada en la siguiente sección se podrían conectar hasta 26 motores.

2. MÉTODOS

A continuación se describen las partes que constituyen el sistema propuesto: la parte mecánica (manipulador) y la parte de adquisición y control.

El manipulador está constituido por dos eslabones: el primero es de dos barras paralelas de dimensiones 20x4x0.18 [cm], separadas 2 [cm] con el fin de sujetar a la segunda articulación; el segundo eslabón consta de una única barra de 30x4x0.18 [cm]. Ambos eslabones son de aleación de aluminio y se unen a sus respectivas articulaciones mediante coples de brida y tornillos para fijar los coples y las dos barras del primer eslabón (Figura 1).

Los actuadores articulares son dos motores de corriente directa de diferente potencia y dimensiones, con la misma relación de conversión de la caja de engranajes (103:1) y el mismo tipo de encoder óptico de cuadratura (West Instruments, 2022), con una resolución de 512 pulsos por revolución (PPR). Por lo anterior, la resolución de la posición articular máxima obtenible es de 0.0017 [deg/count],

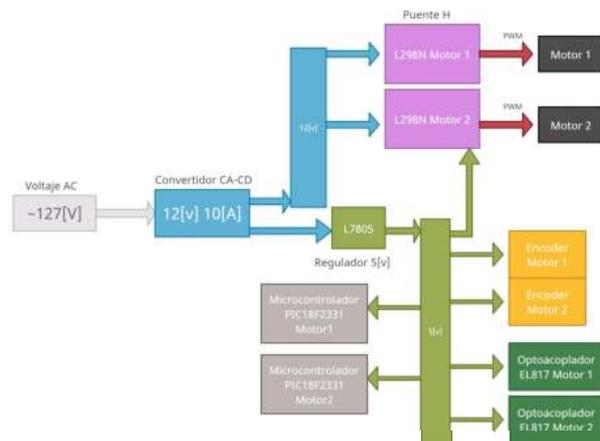


Figura 2. Niveles de tensión.

o 210944 cuentas por vuelta. Ambos motores son de la marca MAXON¹, con un voltaje nominal de 12 [V].

Para alimentar al sistema de adquisición y control se utilizaron varios niveles de tensión, como se muestra en la Figura 2. La alimentación de los motores (línea azul), se realiza mediante un convertidor AC/DC con el que se obtiene una tensión de 12 [V] a 10 [A] nominales. La alimentación para los microcontroladores PIC18F2331, los encoders y el sistema de protección por optoacoplamiento se obtiene utilizando un regulador de voltaje (línea verde) a 5 [V].

Los niveles y tipos de comunicación se representan en la Figura 3. La comunicación con la PC se realiza mediante el microcontrolador PIC18F4550. Este microcontrolador es de 8 bits, permite la comunicación por puerto USB con drivers del tipo *Human Interface Device (HID)* y trabaja a una frecuencia de 96 [MHz]. La velocidad de transferencia utilizada en este trabajo es del tipo *full-speed* que es capaz de transmitir hasta 64 [KB/s] con una tasa de transacción de 1 [ms]. Lo anterior implica que no se pueden obtener periodos de muestreo de menos de 1 [ms], aunque se transmiten simultáneamente 64 [bytes], de los que 2 son de control, por lo que se pueden transmitir hasta 62 [bytes/ms]. Cada par motor/encoder utiliza 2 [bytes] de entrada y 2 [bytes] de salida para recibir y transmitir la información de los encoders y de los motores a la PC, por lo que se podrían conectar hasta 31 pares motores/encoders con la misma tasa de transferencia. Esta es una limitación intrínseca de los microcontroladores que utilizan el protocolo USB full-speed. Sin embargo, actualmente hay muy pocos microcontroladores que emplean protocolos más rápidos e.g. *high-speed* disponibles en el mercado. Adicionalmente, estos últimos microcontroladores no emplean *drivers* estándar para ser conectados y reconocidos por la PC. Por lo anterior, el uso de microcontroladores más poderosos como los de la serie dsPIC o de 32 bits no incrementaría la velocidad de muestreo del sistema.

¹ Los motores de la marca MAXON son de alto desempeño y por lo tanto elevan el costo de la plataforma. Sin embargo, cualquier tipo de motor de corriente directa puede ser utilizado, dependiendo del desempeño requerido.

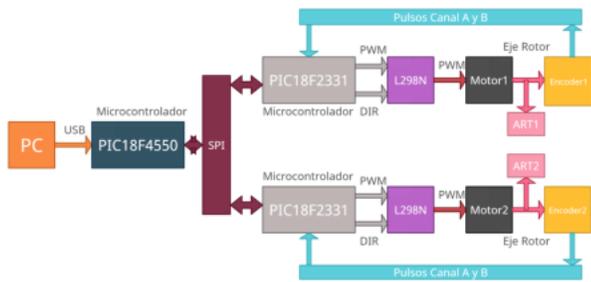


Figura 3. Esquema de comunicación, generación y adquisición.

El PIC18F4550 envía 16 bits de datos que especifican el ciclo de trabajo del PWM para cada motor y recibe otros 16 bits que representan las cuentas de cada encoder. Cada par de datos se transmiten (y reciben) a un microcontrolador PIC18F2331 por medio del protocolo *Serial Peripheral Interface (SPI)*. Cada microcontrolador PIC18F2331 trabaja a 20 [MHz] por lo que el envío y recepción de datos al PIC18F4550 tarda sólo 0.8 [μ s] por cada motor/encoder, lo que no representa ningún cuello de botella aún cuando se tuvieran los 31 motores mencionados anteriormente. Dado que por cada motor/encoder se necesita un pin entrada/salida de propósito general para ambos microcontroladores, el número máximo de motores/encoders está limitado por los pines de entrada salida disponibles en el PIC18F4550 al que se conectan el resto de los microcontroladores, por lo que el número máximo de motores/encoders con el sistema actual es de 26.

Cada PIC18F2331 tiene sólo dos funciones: llevar la cuenta de los pulsos de los encoders y generar la señal PWM hacia cada motor. Este microcontrolador fue elegido porque cuenta con el módulo *Quadrature Encoder Interface (QEI)* que realiza la cuenta de los pulsos de los encoders por hardware, permitiendo especificar el número de cuentas máximas y regresar las cuentas a cero, entre otras. También cuenta con un módulo generador de señales PWM de 14 bits, señales que son luego amplificadas mediante circuitos integrados L298N para finalmente ser enviadas a los actuadores. Dado que el contador máximo de la interfaz QEI es de 16 bits y el número máximo de cuentas por vuelta (210944) sobrepasa esta capacidad, se realizó una rutina de interrupción para poder realizar la cuenta máxima y luego se dividió entre 4 para enviarla utilizando una palabra de 16 bits, por lo que la resolución de cada encoder en realidad es de 52736 cuentas por vuelta o equivalentemente 0.006826 [deg/count]. El esquema de conexiones se muestra en la Figura 4.

Cada microcontrolador se programó solo para adquirir, generar y transmitir la infracción de los encoder a la computadora y de la computadora hacia los motores. La calibración, el acondicionamiento de las señales y la aplicación de las técnicas de control se realizó en la plataforma donde se diseñó la interfaz de usuario, desarrollada en *Visual Studio*, bajo el sistema operativo *Windows*. Esto le da la capacidad de ser flexible al usuario cuando desea implementar un nuevo algoritmo de control, sin la necesidad de reprogramar la tarjeta de adquisición. En general, el sistema tiene un periodo de muestreo de

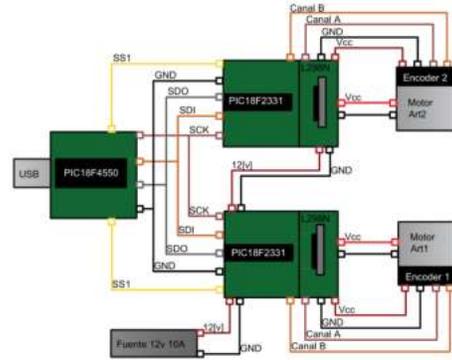


Figura 4. Esquemático de conexiones entre módulos.

2 [ms], considerando la adquisición de señales y el cálculo de las leyes de control/observación programadas en la PC. Este periodo de muestreo es adecuado para el robot presentado en este trabajo y para la mayoría de los robots manipuladores, cuando se utilizan controladores que no dependen críticamente del periodo de muestreo (e.g. control por modos deslizantes de primer orden). Para este último caso, una solución sería programar la ley de control embebida en el microcontrolador PIC18F4550, aunque perdiendo la flexibilidad y la rapidez de programación y sintonización de los controladores.

El costo de construcción de la parte mecánica de sistema (manipulador) se determinó con base en el material que se utilizó para dicha construcción, con un valor final de \$251.50 MXN. Por otro lado, el costo del sistema de adquisición, control y comunicación, es de \$1,978.09 MXN (en junio del 2022) sin considerar los actuadores, como se describe en la Tabla 1.

Cuadro 1. Costos de fabricación del Módulo de Adquisición, Control y Comunicación (precios en Pesos Mexicanos (MXN) a Junio del 2022).

Componente	Cantidad	Costo/u	Costo
PIC18F4550	1	\$ 410	\$ 410
PIC18F2331	2	\$ 393	\$ 786
Tarjeta de desarrollo JT45k50	1	\$ 77	\$ 77
Circuito L298N	2	\$ 51.72	\$ 103.45
Fuente de Poder YXD WISH	1	\$ 268.70	\$ 268.70
Material Electrónico	-	-	\$ 332.94
Total			\$ 1,978.09

Para definir la capacidad que entrega el sistema de adquisición, control y comunicación propuesto (USBServo v1.0), se realizó una comparativa con otras tarjetas disponibles en el mercado que son capaces de realizar la misma tarea. Esta comparación se presenta en la Tabla 2.

La tarjeta Compac RIO-9053 de National Instruments, esta tarjeta de adquisición solo consta del modulo principal que comunica con la computadora, para la lectura de encoder y la etapa de potencia hacia los motores se requieren de módulos individuales para cada uno de los motores a controlar, por ejemplo la tarjeta NI9505 que es un modulo de control de motores con 3 entradas para lectura de encoders y una salida a motor por PWM de

Cuadro 2. Costos y características de tarjetas de adquisición y control.

Módulo	Características	Costo
NI cRIO-9053	Velocidad 1.33Ghz, Conectividad Ethernet y USB, slots para módulos I/O Analógicos, I/O Digitales.	\$43,575
Q8-USB	I/O Analógicos, I/O Digitales, 8 Encoders canal A y B, 8 PWM a 2.5[v] con 16bits. 6 Encoders canal A y B, I/O Analógicas, I/O Digitales, interfaz de comunicación USB, CAN y Ethernet.	\$70,000
MG MACS	Mini- MACS	Costo por pedido
USBServoV1.0 (Propuesto)	Velocidad 20MHz, En- coder canal A y B, PWM a 12v con 16 bits, comunicación SPI y USB.	\$1,978.09

40kHz de frecuencia máxima, esto como consecuencia eleva el precio de todo el sistema de adquisición y control.

El Q8-USB es una tarjeta de adquisición y generación de señales de la marca Quanser, permite leer 8 señales de tipo encoder y genera señales PWM a 16 bits de resolución para controlar motores, adicionalmente requiere de una etapa de potencia ya que el voltaje y la corriente que proporciona no es capaz de alimentarlos. Además se comunica por medio de interfaz USB y SPI.

MG MiniMACS es un modelo de tarjeta de control de motores de la marca MAXON, permite controlar 3 ejes, es considerado el modelo de menor costo y no requiere de etapa, trabaja a 24 [V] y además cuenta con I/O digitales y analógicas que se pueden utilizar para otras aplicaciones, es capaz de comunicarse con un PC, PLC o interfaz mediante USB, Ethernet y CAN.

El modelo matemático del brazo robótico se obtuvo mediante la formulación de Euler-Lagrange como (Arteaga et al., 2021)

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{f}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \quad (1)$$

donde $\mathbf{H}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ es la matriz de inercia, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \in \mathbb{R}^n$ es el vector de fuerzas centrífugas y de Coriolis, $\mathbf{f}(\dot{\mathbf{q}}) \in \mathbb{R}^n$ son los pares debidos a la fricción, $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ son los pares de origen gravitatorio y $\boldsymbol{\tau} \in \mathbb{R}^n$ son los pares de entrada en los actuadores. Se realizó la siguiente parametrización lineal del modelo con la finalidad de obtener una identificación de los parámetros

$$\boldsymbol{\tau} = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\theta}, \quad (2)$$

donde $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \in \mathbb{R}^{n \times p}$ es el regresor y $\boldsymbol{\theta} \in \mathbb{R}^p$ es el vector de parámetros. Se agruparon las señales de entrada y del regresor obtenidas en cada muestreo como sigue

$$\bar{\boldsymbol{\tau}} = \begin{bmatrix} \tau(t_1) \\ \vdots \\ \tau(t_N) \end{bmatrix} = \begin{bmatrix} \mathbf{Y}(t_1) \\ \vdots \\ \mathbf{Y}(t_N) \end{bmatrix} \boldsymbol{\theta} = \bar{\mathbf{Y}}\boldsymbol{\theta}, \quad (3)$$



Figura 5. Plataforma Robot Manipulador.

donde t_1 es el primer muestreo y t_N es el último instante de muestreo. Para obtener una estimación de los parámetros se utilizó el método de mínimos cuadrados estándar, i.e.

$$\boldsymbol{\theta} = (\bar{\mathbf{Y}}^T \bar{\mathbf{Y}})^{-1} \bar{\mathbf{Y}}^T \bar{\boldsymbol{\tau}}. \quad (4)$$

El regresor $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ y el vector de parámetros $\boldsymbol{\theta}$ para el manipulador de dos grados de libertad, considerando fricción de Coulomb y fricción viscosa (Bona and Indri, 2005), i.e. $\mathbf{f}(\dot{\mathbf{q}}) = \mathbf{D}_s \text{sign}(\dot{\mathbf{q}}) + \mathbf{D}\dot{\mathbf{q}}$, tiene la forma

$$\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = \begin{bmatrix} \ddot{q}_1 & y_{12} & y_{13} & \dot{q}_1 & 0 & \text{sign}(\dot{q}_1) & 0 & c_1 & c_{12} \\ 0 & y_{22} & y_{23} & 0 & \dot{q}_2 & 0 & \text{sign}(\dot{q}_2) & 0 & c_{12} \end{bmatrix}, \quad (5)$$

donde $c_1 = \cos(q_1)$, $c_{12} = \cos(q_1 + q_2)$, $s_2 = \sin(q_2)$, $y_{12} = 2c_2\ddot{q}_1 + c_2\ddot{q}_2 - s_2\dot{q}_1\dot{q}_2 - s_2(\dot{q}_1 + \dot{q}_2)\dot{q}_2$, $y_{13} = y_{23} = \ddot{q}_1 + \ddot{q}_2$ y $y_{22} = c_2\dot{q}_1 + s_2\dot{q}_1^2$. Por otra parte, el vector de parámetros se define como

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \\ \theta_8 \\ \theta_9 \end{bmatrix} = \begin{bmatrix} m_1 l_{c1}^2 + m_2 a_1^2 + \frac{m_2}{12} a_2^2 \\ m_2 a_1 l_{c2} \\ m_2 l_{c2}^2 + \frac{m_2}{12} a_2^2 \\ c_{fv1} \\ c_{fv2} \\ c_{fs1} \\ c_{fs2} \\ (m_1 l_{c1} + m_2 a_1)g \\ m_2 l_{c2}g \end{bmatrix}, \quad (6)$$

donde m_1 y m_2 son las masas de los eslabones, a_1 y a_2 son las longitudes totales de los eslabones, l_{c1} y l_{c2} son las distancias a los centros de masa, c_{fv1} y c_{fv2} son los coeficientes de fricción viscosa, c_{fs1} y c_{fs2} son los coeficientes de fricción de Coulomb y g es la constante de aceleración gravitatoria.

3. EXPERIMENTOS

La plataforma construida en este trabajo se muestra en la Figura 5. La adquisición, procesamiento y envío de señales hacia el manipulador se realiza con un periodo de muestreo de 2 [ms], utilizando sólo un puerto USB. La programación se realizó en una computadora con el sistema operativo Windows, utilizando el entorno de desarrollo Visual Studio C/C++. Para la identificación

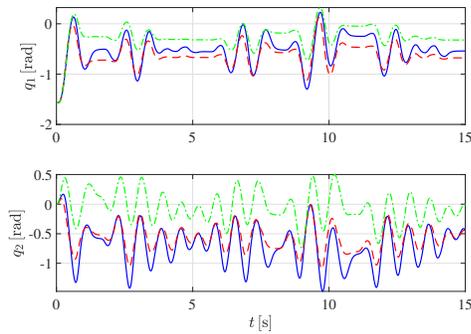


Figura 6. Posiciones articulares obtenidas con: señales medidas (—), parámetros nominales (---) y parámetros identificados (-.-).

de los parámetros, se utilizó una señal en lazo abierto para cada articulación como sigue

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} b_{11} + b_{12} \sin(2\pi t) + b_{13} \sin\left(\frac{2\pi t}{f_1}\right) + b_{14} \sin\left(\frac{2\pi t}{f_2}\right) \\ b_{21} + b_{22} \cos(2\pi t) + b_{23} \cos\left(\frac{2\pi t}{f_1}\right) + b_{24} \cos\left(\frac{2\pi t}{f_2}\right) \end{bmatrix}, \quad (7)$$

donde $b_{11} = 0.22, b_{12} = 0.08, b_{13} = 0.1, b_{14} = 0.12, b_{21} = 0.025, b_{22} = 0.016, b_{23} = 0.032, b_{24} = 0.048$ y $f_1 = 0.9, f_2 = 0.7$. Los parámetros identificados mediante el método de mínimos cuadrados (4) se muestran en la Tabla 3, junto a los valores nominales (pesados y medidos). Las posiciones articulares medidas, simuladas con

Cuadro 3. Tabla de comparativa entre parámetros estimados y calculados.

Parámetro	Unidades	Estimado	Nominal
θ_1	kgm ²	0.00559	0.003522
θ_2	kgm ²	0.00022	0.001252
θ_3	kgm ²	0.00148	0.001302
θ_4	kgm ² /(rads)	0.01467	---
θ_5	kgm ² /(rads)	0.00303	---
θ_6	Nm	0.05617	---
θ_7	Nm	0.00374	---
θ_8	Nm	0.20793	0.2479
θ_9	Nm	0.02807	0.0744

los parámetros nominales y simuladas con los parámetros identificados utilizando la misma entrada (7) se muestran en la Figura 6.

Para validar el funcionamiento y la flexibilidad de programación de la plataforma se programaron 6 controladores ampliamente conocidos en la literatura: el control lineal PID, Control Par Calculado, PD con compensación de gravedad, Control Basado en Pasividad, Control Adaptable y Control por Modos Deslizantes de primer orden. Para todos los controladores se realizaron experimentos de regulación, seguimiento de trayectorias articulares y seguimiento de trayectorias Cartesianas. En la Figura 7 se muestran las posiciones articulares obtenidas para el experimento de regulación. Este experimento es en general más agresivo que el seguimiento de trayectorias. Los errores articulares se muestran en la Figura 8, mientras que los pares de entrada correspondientes se muestran en la Figura 9.

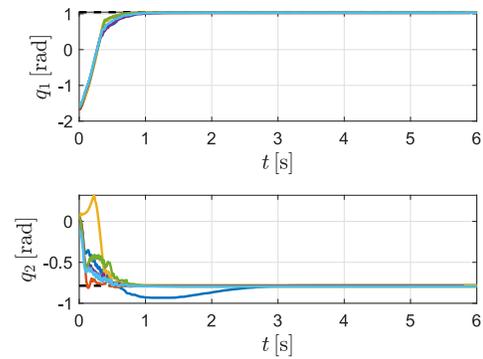


Figura 7. Regulación q_1 y q_2 : Deseada (—), PID (---), PD+G (-.-), Par Calculado (....), Pasivo (-.-.-), Adaptable (— — —), Modos Deslizantes (— — —).

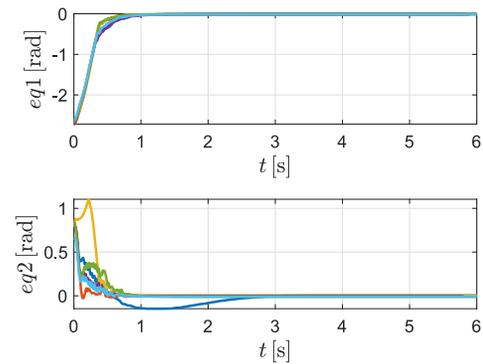


Figura 8. Error de Posición eq_1 y eq_2 : PID (---), PD+G (-.-), Par Calculado Identificado (....), Par Calculado Nominal (-.-.-), Pasivo (-.-.-.-), Adaptable (— — —), Modos Deslizantes (— — —).

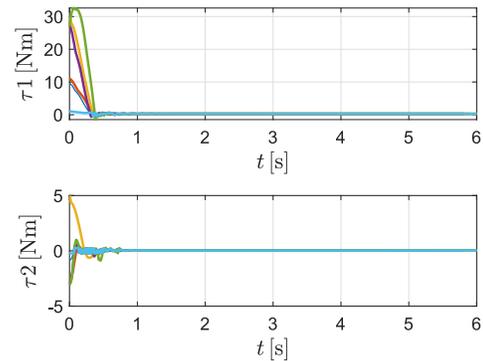


Figura 9. Pares articulares: PID (---), PD+G (-.-), Par Calculado Identificado (....), Par Calculado Nominal (-.-.-), Pasivo (-.-.-.-), Adaptable (— — —), Modos Deslizantes (— — —).

El seguimiento de trayectorias en coordenadas articulares se muestra en la Figura 10, mientras que los errores de seguimiento se muestran en la Figura 11. Por último, los pares articulares resultantes de cada controlador se muestran en la Figura 12.

4. CONCLUSIONES Y TRABAJO FUTURO

Se implementaron satisfactoriamente los algoritmos de control no lineales más conocidos en la literatura en la

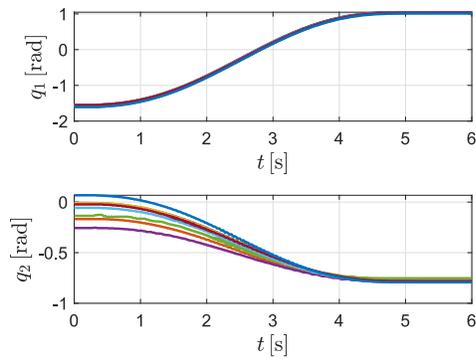


Figura 10. Trayectoria de q_1 y q_2 : Deseada (—), PID (—), PD+G (—), Par Calculado Identificado (—), Par Calculado Nominal (—), Pasivo (—), Adaptable (—), Modos Deslizantes (—).

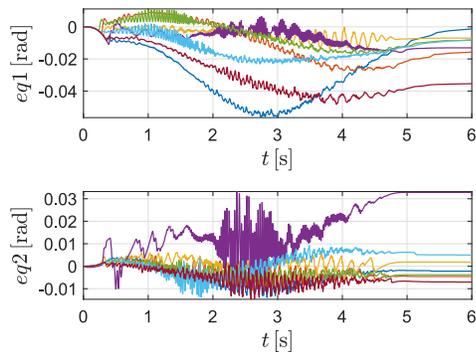


Figura 11. Error de Posición eq_1 y eq_2 : PID (—), PD+G (—), Par Calculado Identificado (—), Par Calculado Nominal (—), Pasivo (—), Adaptable (—), Modos Deslizantes (—).

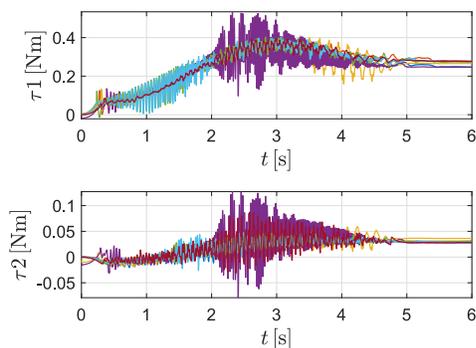


Figura 12. Par τ_1 y τ_2 : PID (—), PD+G (—), Par Calculado Identificado (—), Par Calculado Nominal (—), Pasivo (—), Adaptable (—), Modos Deslizantes (—).

plataforma con los resultados esperados. Dada la facilidad de implementación de estos controladores, es razonable asegurar que se puedan implementar nuevos controladores para investigación o que pueda ser utilizada por alumnos para implementar controladores ya reportados. Asimismo la plataforma es portátil, al requerir únicamente de una PC con puerto USB. El diseño modular y de bajo costo permitiría replicarla fácilmente o ampliarla para otro tipo de manipuladores. Al haber construido el brazo manipu-

lador, fue posible obtener los parámetros nominales los que, utilizándolos en el modelo, obtuvieron respuestas más parecidas a las señales obtenidas en la plataforma que las obtenidas mediante el modelo utilizando los parámetros identificados. Esto abre la posibilidad de mejorar las técnicas de identificación paramétrica y realizar la comparación con lo presentado en este trabajo.

Como trabajo futuro se propone diseñar un robot de mayor número de grados de libertad, aprovechando el diseño modular planteado. También se planean construir otros dos robots del tipo pendubot y acrobot. Para el caso del pendubot, se propone cambiar el segundo motor por un rodamiento para disminuir la fricción acoplado a un encoder. Para el caso del Acrobot, se propone modificar las características mecánicas del brazo, así como incrementar la potencia del segundo motor. Asimismo, se pretende implementar librerías para su programación directa utilizando Matlab. De la misma forma, se propone su integración para ser utilizado en aplicaciones bajo el sistema operativo robótico ROS.

REFERENCIAS

- Arteaga, M.A., Gutiérrez-Giles, A., and Pliego-Jiménez, J. (2021). *Local Stability and Ultimate Boundedness in the Control of Robot Manipulators*. Springer.
- Benitez, V.H., Symonds, R., and Elguezabal, D.E. (2020). Design of an affordable iot open-source robot arm for online teaching of robotics courses during the pandemic contingency. *HardwareX*, 8, e00158.
- Bona, B. and Indri, M. (2005). Friction compensation in robotics: an overview. In *Proceedings of the 44th IEEE Conference on Decision and Control*, 4360–4367. IEEE.
- Elfasakhany, A., Yanez, E., Baylon, K., Salgado, R., et al. (2011). Design and development of a competitive low-cost robot arm with four degrees of freedom. *Modern mechanical engineering*, 1(02), 47.
- Kim, H.S. and Song, J.B. (2013). Low-cost robot arm with 3-dof counterbalance mechanism. In *2013 IEEE International Conference on Robotics and Automation*, 4183–4188. IEEE.
- Maldonado, J., Lopez, K., Garrido, R., and Mondié, S. (2018). Implementing time-delay controllers on an educational motion control platform. In *2018 XX Congreso Mexicano de Robótica (COMRob)*, 1–6. IEEE.
- Ortega, R. and Spong, M.W. (1989). Adaptive motion control of rigid robots: A tutorial. *Automatica*, 25(6), 877–888.
- Ramos, J.J.M., Pineda, J.L.L., Moctezuma, R.A.G., and Zavala, J.G.C. (2019). A teaching methodology based on an educational experimental platform. *IEEE Latin America Transactions*, 17(08), 1363–1370.
- Slotine, J.J.E. (1985). The robust control of robot manipulators. *The International Journal of Robotics Research*, 4(2), 49–64.
- Slotine, J.J.E. and Li, W. (1987). On the adaptive control of robot manipulators. *The international journal of robotics research*, 6(3), 49–59.
- West Instruments (2022). *Manual de Aplicación de Encoders*.