

PÉNDULO INVERTIDO CONTROLADO CON TÉCNICAS DE TIEMPO REAL: DESARROLLO DEL CONTROLADOR VÍA CAMPO DE DATOS

Hinojosa, Mauricio

mfhinojosa@utn.edu.ec

Carrera de Ingeniería en Mecatrónica, Universidad Técnica del Norte, Ibarra, Ecuador

Resumen—El presente artículo consiste del péndulo invertido El controlador implementado para el sistema fue desarrollado mediante la técnica de espacio de estados a partir del modelo matemático y simulado en Matlab obteniendo sus ganancias, las mismas que sirven para modificar el comportamiento del sistema. El sistema de control está dividido en dos, una parte de monitoreo de datos o HMI y otra de adquisición de datos y control.

Índice de Términos—Péndulo, Planta, Modelo Matemático, Diseño Mecánico, Control Distribuido, HMI.

I. INTRODUCCIÓN

El desarrollo de controladores es una constante que va de mano con el crecimiento económico y tecnológico de los pueblos y sus industrias; impulsando avances en los procesos industriales con el propósito de fomentar competitividad para los países en materia tecnológica [2].

Al desarrollar controladores que funcionen de la manera más adecuada en sistemas lineales y no lineales representa en si un reto, en especial desde el punto de vista académico, al tratar de acoplar la utilidad de los controladores, con la seguridad que deben presentar, y lo confiables que deben ser a la hora de su implementación [2].

Existe una simple razón por la que el “Péndulo” es tan utilizado en investigación, es su dinámica no lineal, que permite comprender el comportamiento de sistemas mas complejos desde el punto de vista de su linealidad y dinámica, como son los sectores de transporte, telecomunicaciones, aeronáutica; con lo que los controladores son fácilmente aplicables a estos sistemas [2].

En el presente trabajo se realizará la implementación de un controlador con técnicas de tiempo real sobre arduinos a través de una red de campo (basada en protocolos de comunicación CAN multi-nodo); donde los datos de monitoreo así como el controlador serán mostrados en la HMI desarrollada para este propósito .

II. METODOLOGÍA

En este capítulo se define y sintetiza el conjunto de fases que se debe seguir durante el desarrollo del sistema de péndulo invertido, en primera instancia se parte de la investigación teórica, se establecen los requisitos del sistema de control; se desarrolla la red de comunicaciones, en este caso una red CAN de 4 nodos y por último la HMI.

Función de Transferencia.

El controlador está basado en un sistema de equilibrio en bajo o mejor llamado Down-position o crank-down; y cuya función de transferencia obtenida es:

$$\ddot{x} = \frac{F - ml(\ddot{\theta} \cos \theta + \dot{\theta}^2 \sin \theta) - f_c \dot{x}}{(M + m)} \quad (3.1)$$

$$\ddot{\theta} = \frac{mgl \sin \theta - ml\ddot{x} \cos \theta - f_p \dot{\theta}}{(I_p + ml^2)} \quad (3.2)$$

Punto de equilibrio del sistema

La ecuación 3.1 muestra la representación del sistema de una forma no-lineal. Equilibrio está definido como un punto (o una curva) en donde todos los estados permanecen sin cambio, sus derivadas son ceros (0). Después de que 3.1 haya sido reemplazada en 3.2 y viceversa aparece un nuevo sistema de ecuaciones:

$$\beta \ddot{x} = -ml \cos \theta (mgl \sin \theta - f_c \dot{\theta}) = 0 \quad (3.6)$$

$$\beta \ddot{\theta} = (M + m)(mgl \sin \theta) = 0 \quad (3.7)$$

Linealización en posición bajo

El sistema linealizado en espacio de estados esta descrito por:

$$\begin{cases} \dot{X} = A_{down}X + B_{down}U \\ Y = CX + DU \end{cases}$$

Donde A,B,C,D son matrices y vectores definidos por espacio de estados y son:

$$A_{down} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{(I_p+ml^2)f_c}{\alpha} & -\frac{m^2l^2g}{\alpha} & -\frac{mlf_p}{\alpha} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{mlf_c}{\alpha} & -\frac{(M+m)mlg}{\alpha} & -\frac{(M+m)f_p}{\alpha} \end{bmatrix}$$

$$B_{down} = \begin{bmatrix} 0 \\ \frac{(I_p+ml^2)}{\alpha} \\ 0 \\ \frac{ml}{\alpha} \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad D=0 \quad U=F$$

Matriz numérica con parámetros del sistema

Tomando los siguientes parámetros:

- $M = 0.972 \text{ Kg}$
- $m = 0.144 \text{ Kg}$
- $L = 0.38 \text{ m}$
- $I_p = 0.02627 \text{ Kg/rad}$
- $f_p = 0.01 \text{ Ns/rad}$
- $f_c = 1.1 \text{ Ns/m}$

Cuadro 3.2: Constantes del sistema

Las matrices obtenidas son las siguientes:

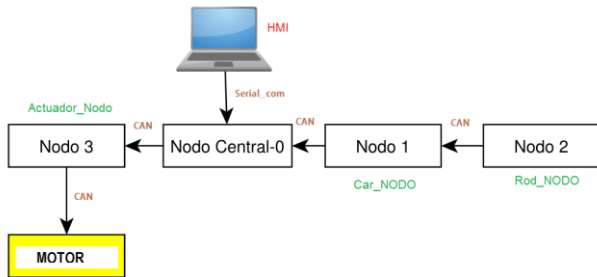
$$A_{down} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -06498 & -0,5930 & -0,011 \\ 0 & 0 & 0 & 1 \\ 0 & -1,21 & -12,08 & -0,2253 \end{bmatrix}$$

$$B_{down} = \begin{bmatrix} 0 \\ 4,04 \\ 0 \\ 1,10 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad D=0 \quad U=F$$

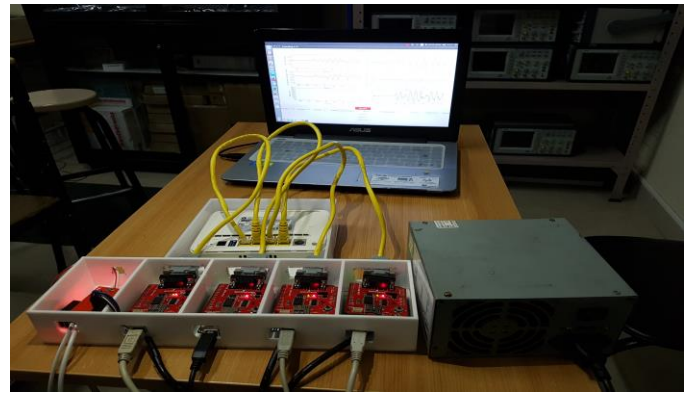
Ganancias obtenidas

- Estas son las ganancias obtenidas a partir del algoritmo de Matlab: $G1 = 59.86$ (Ganancia de la posición del carro).
- $G2 = 17.9$ (Ganancia de la velocidad del carro).
- $G3 = -3.17$ (Ganancia de la posición del péndulo).
- $G4 = -10.713$ (Ganancia de la velocidad del péndulo).
- $G5 = 0.9345$ (Ganancia del error integral).

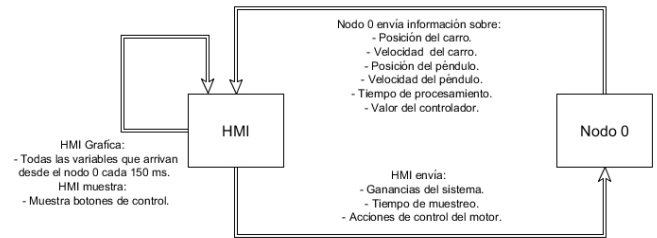
Can Bus Shield



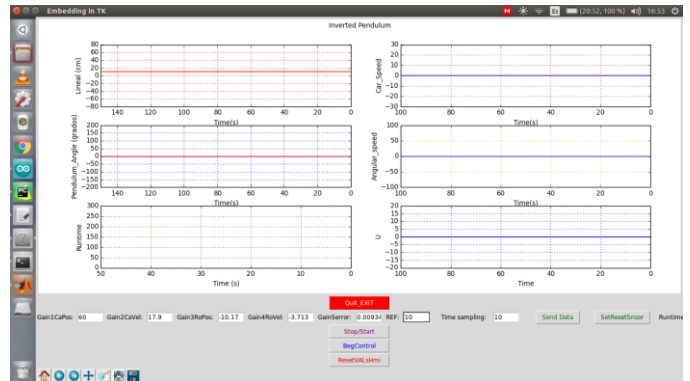
Red física CAN e interconexión con la HMI



Máquina de estado de la HMI



Ventana HMI terminada



Trámas de datos enviados desde HMI hacia el nodocentral

```

([command,direccion , ord(value0[0]), ord(value0[1]), ord(value0[2]), ord(value0[3]), ord(value1[0]), ord(value1[1]), ord(value1[2]), ord(value1[3]), ord(value2[0]), ord(value2[1]), ord(value2[2]), ord(value2[3]), ord(value3[0]), ord(value3[1]), ord(value3[2]), ord(value3[3]), ord(value4[0]), ord(value4[1]), ord(value4[2]), ord(value4[3]), ord(value5[0]), ord(value5[1]), ord(value5[2]), ord(value5[3]), ord(value6[0]), ord(value6[1]), ord(value6[2]), ord(value6[3]), ord(value7[0]), ord(value7[1]), ord(value7[2]), ord(value7[3]), ord(value8[0]), ord(value8[1]), ord(value8[2]), ord(value8[3]), ord(value9[0]), ord(value9[1]), ord(value9[2]), ord(value9[3]),0x0A])
    
```

III. RESULTADOS

PruebadeadquisicióndedatosconelsensorHEDM

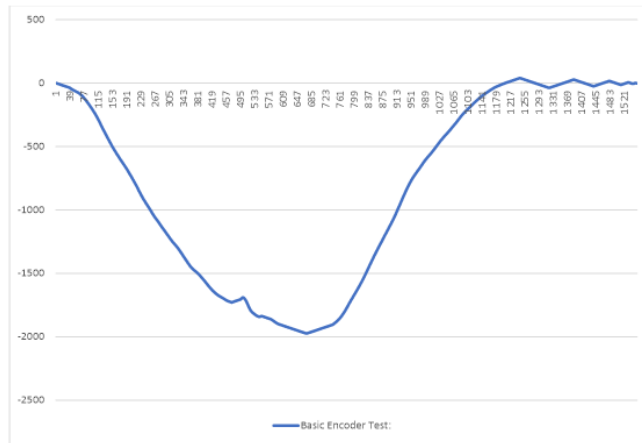
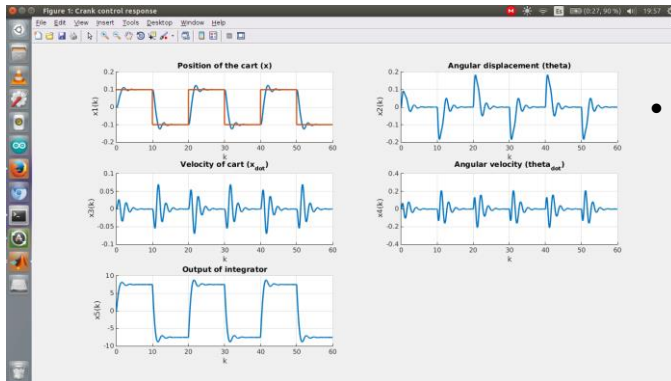


Gráfico obtenido de la simulación



Análisis de los resultados

Los resultados obtenidos por Matlab muestran que el sistema es controlable, con una itinerancia de software de 2149; donde se encontraron los polos del sistema así como sus ganancias, las cuáles fueron puestas en un sistema de realimentación cerrado y simulado por Matlab

Prueba del sistema real con las ganancias obtenidas

Prueba del sistema real con las ganancias obtenidas en el capítulo anterior en donde fueron modificadas para reducir el tiempo de estabilización en estado estable, dandola siguiente tabla

Prueba	G1CarPos	G2CarVel	G3RosPos	G4RosVel	G5error	Tiempo Estabilizacion(seg)
1	59.86	17.9	10.17	-3.713	0.98	Inestable
2	59.86	17.9	-10.17	-3.713	0.15	20
3	59.86	17.9	-40.17	-3.713	0.15	14
4	59.86	17.9	-40.17	-10.713	0.15	13
5	59.86	17.9	-40.17	-1.713	0.15	15
6	59.86	17.9	-40.17	-5.713	0.15	15
7	59.86	17.9	-40.17	-7.713	0.15	14
8	59.86	17.9	-40.17	-20.713	0.15	11
9	59.86	17.9	-40.17	-10.713	0.1	14
10	59.86	17.9	-40.17	-3.713	0.08	12
11	59.86	17.9	-3.17	-3.713	0.1	20
12	59.86	17.9	-40.17	-3.713	0.25	Inestable

Análisis de los ganancias modificadas en el sistema real

Los datos de la tabla 4.1 muestra que las ganancias más críticas son G3RosPos y G5error; tomando la primera valores menores a 0 el sistema se vuelve inestable y para la segunda con valores mayores a 1.5 sucede lo mismo. El mejor tiempo de estabilización se obtuvo con G3RosPos=-40.17, G2CarVel=-20.713 y G5error =0.15 con un tiempo de estabilización de 11 seg.

IV. CONCLUSIONES

- Los documentos recopilados referentes al control por red mostraban únicamente el envío y recepción de varios tipos de datos a través de una red sin que estos documentos estén relacionados al diseño de controladores; por otra parte, cuando se encontraba documentación relacionada al desarrollo de controladores para péndulos, era dentro de un mismo microprocesador con el que se adquiría, procesaba, calculaba y se tomaba una acción de control. Por tanto en base a que no se encontró información completa para desarrollar este trabajo se tomaron en cuenta 4 fundamentos ampliamente desarrollados en otras tesis y son: adquisición y procesamiento de datos, redes CAN, HMI's y diseño de controladores.
- Los requerimientos necesarios para el sistema fueron tomados de documentos en los que se realizaba el control en sistemas de péndulos invertidos pero la información relacionada a la implementación de controladores en una red de campo de datos era prácticamente nula, por lo que la misma fue especialmente diseñada y desarrollada para la planta.
- La Fig. 4.4 es resultado de introducir los parámetros y constantes de la planta, que están especificados en la sección 3.2.2, en el código desarrollado en Matlab para obtener el controlador diseñado bajo la técnica de espacio de estados. Como se muestra en esta simulación, el control cumple con la condición de estabilizar el sistema; el proceso antes mencionado se puede ver claramente en la gráfica que
- representa la posición del carrito, a la que se le sometió una entrada escalón (línea naranja) y se produjo una respuesta en la posición simulada del carrito (línea azul). Básicamente, la respuesta tiene una variación que sobrepasa el 10% de la entrada escalón

(aceptable en términos de diseño de controladores teóricos). Las ganancias obtenidas se utilizaron de base en el sistema real. En la construcción del sistema se puso especial cuidado en sectores críticos como la riel y la forma de las llantas que son primordiales en el desplazamiento del péndulo y un error en su manufactura daría como resultado una planta diferente al modelo matemático calculado.

- La adquisición, monitoreo y control de datos de la planta se muestra en la Fig. 3.13. La interfaz tiene dos partes; la parte superior, desarrollada para la adquisición y monitoreo de datos, representa las gráficas de las variables del sistema y está seccionado en 6 siendo: la posición del carro y su velocidad (superior izquierda y superior derecha); la posición del péndulo y su derivada (medial izquierda y medial derecha); tiempo de procesamiento (línea amarilla) y el controlador (línea azul, posicionada en la parte inferior derecha de la sección de gráficas). La parte de control de la propia HMI y de la red, está en la parte inferior de la ventana, con la que se puede modificar las ganancias del controlador del sistema, resetear sensores, apagar motor, parar y reiniciar el sistema y la comunicación, entre otros. Con estas dos secciones de la interfaz queda comprobado que se cubren los aspectos planteados en el diseño; además de la práctica utilidad que brinda para manejar el sistema.
- Después de haber implementado la red, HMI y el controlador para el sistema, las pruebas fueron realizadas y se pudieron obtener varios valores donde las ganancias estabilizan la planta. La tabla 4.1 muestra que después de 12 pruebas en las que se variaron 3 ganancias (G3Rospos: ganancia de la posición del péndulo; G4RosVel: ganancia de la velocidad del péndulo y G5error: ganancia del error integral), se pudo demostrar que sólo en dos pruebas con valores de G5error distintas (el primero con el que se empezó la prueba y cuyo valor fue tomado de la simulación y el segundo con valor randómico de 0.25, correspondiente a la prueba 12), la planta se volvía inestable; mientras que para el resto de ganancias modificadas la variación en tiempo de estabilización tiene una diferencia de 9 segundos entre los valor máximo y mínimo de la tabla; a lo que podemos concluir que el diseño del controlador esta dentro de los parámetros aceptables de funcionamiento (10 de 12 pruebas).

V. RECOMENDACIONES

- El correcto funcionamiento de la planta con Arduino Uno muestra la efectividad en cuanto a trabajo de procesamiento que tienen estos shields y control que proveen para con los sensores; sin embargo se recomienda el uso y desarrollo en Arduino Due o Mega.
- Habría que cambiar los shields de comunicación CAN si se usan algún otro tipo de shield de control, ya que los que están montados en la red son exclusivos para los Arduinos Uno o en su defecto manufacturar un shield propio. Se recomienda variar la altura del péndulo al momento de implementar un controlador y verificar su funcionamiento con esta variación.
- El sistema podría responder de manera más o menos estable, dependiendo de las ganancias que sean calculadas a posteriori para el mismo controlador o para controladores que sean implementados con diferentes técnicas de diseño de controladores. Se recomienda cambiar el eje que conecta a los 2 péndulos por uno solido debido a que puede llegar a deformarse debido al contacto constante al que está sujeto y no por el peso de los péndulos en sí.
- Para posteriores trabajos se deberá tomar en cuenta los tiempos de propagación dentro del sistema de comunicación CAN, aquellos que afectan al sistema de control de la planta.
- Realizar las modificaciones del sistema de control dentro de los nodos para conseguir la estabilización del sistema en alto.
- Se podría utilizar la HMI para trabajos no tan rigurosos, aún que la proyección es utilizarlo como punto de convergencia para intercambiar controladores sin tener que desmontar totalmente la red.

REFERENCIAS

- [1] A. García Osés, Diseño de una red CAN bus con Arduino. Universidad Pública de Navarra, 2015, trabajo Fin de Grado en Ingeniería en Tecnologías Industriales.
- [2] O. Montoya, J. Valenzuela, and D. Buitrago, "Balanceo y estabilización del péndulo invertido empleando redes neuronales artificiales y un regulador lineal óptimo con criterio cuadrático (LQR)," *Scientia et Technica*, vol. 14, mar 2013, universidad Tecnológica de Pereira, Pereira, Colombia.
- [3] F. Ortega, Modelado y control del péndulo invertido sobre carro mediante sistemas híbridos. Universidad de Sevilla, Escuela Superior Técnica de Ingenieros, 2012.
- [4] A. Hernández, M. Legaspi, and J. Peláez, Control inteligente del péndulo invertido. Universidad Complutense de Madrid, 2012.
- [5] K. Ogata, Ingeniería de Control Moderna 5ª ED. Prentice-Hall, 2010.
- [6] Text Analysis International Inc, "Integrated development environments for natural language processing," 2011.
- [7] S. Kavitha and S. Sindhu, "Comparison of integrated development environment (ide) debugging tools: Eclipse vs netbeans," *International Research Journal of Engineering and Technology (IRJET)*, vol. 02, jul 2015, universidad Tecnológica de Pereira, Pereira, Colombia.
- [8] P. Carbonnelle, "Topide top ide index (recuperado de:

- <http://pypl.github.io/IDE.html>),” 2015, online; accessed 31 August 2016.
- [9] S. Lochhaas and M. Moore, Open Source Software Libraries. B sides, Journal of the University of Iowa School of library and information science, 2010.
- [10] R. Arango, A. Navarro, and J. Padilla, “Sistemas open hardware y open source aplicados a la enseñanza de la electrónica,” Revista de Investigaciones- Universidad del Quindo, mar 2014, universidad del Quindol.
- [11] J. Rodríguez, L. Arana, A. Rabasa Dolado, and O. Martínez Bonastre, Introducción a la programación. Teoría y práctica. Editorial Club Universitario, 2013.
- [12] R. González, Python para todos. Creative Commons Reconocimiento 2.5 España., 2014, España.
- [13] R. Mall, Real-Time Systems Theory and Practice. Dorling Kinderley (India) Pvt. Ltd., licensee of Pearson Education in South Asia, 2012, department of Computer Science and Engineering Indian Institute of Technology Kharagpur India.
- [14] M. Ortiz López, Sistemas en Tiempo Real. Universidad de Córdoba, 2014, Área de Arquitectura y Tecnología de Computadores Departamento de Arquitectura de Computadores, Electrónica y Tecnología Electrónica.
- [15] H. Fernández, OPEN HARDWARE. Universidad Católica, 2012.
- [16] L. Russell, A. Steele, and R. Goubran, “Low-cost, rapid prototyping of imu and pressure monitoring system using an open source hardware design,” IEEE Advancing Technology for Humanity, jul 2012, DOI: 0.1109/I2MTC.2012.6229719.
- [17] A. Lazalde, J. Torres, and D. Vila-Vinas, “Hardware libre (v1.2),” Buen Conocer - FLOK Society documento de política pública, jan 2015, proyecto realizado bajo convenio con el Ministerio Coordinador del Conocimiento y Talento Humano, la Secretaría Nacional de Educación Superior, Ciencia, Tecnología e Innovación y el Instituto de Altos Estudios Nacionales del Ecuador (IAEN).
- [18] A. Montalvo, Inverted pendulum controlled via real-time techniques: Plant development and modeling. UTN-Ecuador, 2017, facultad de Ingeniería en Ciencias Aplicadas.

BIOGRAFÍA DE AUTOR

Autor. – Mauricio Fernando Hinojosa Rea. Nació el 11 de Julio de 1992 en el cantón Otavalo. Realizó sus estudios secundarios en el colegio Santa Juana de Chantal obteniendo el título de Físico Matemático. Actualmente es egresado de la carrera de Ingeniería Mecatrónica de la Universidad Técnica del Norte de la ciudad de Ibarra.

Área de interés: Automatización y control industrial, programación microcontroladores, sistemas de control electrónica y robótica.