



Universidad Nacional de Córdoba
Facultad de Ciencias Exactas, Físicas y Naturales

Sistemas de Control 2
Trabajo Práctico N°2

Docente:
Dr. Ing. Julián Pucheta

Autor:
Esteban Samuel Suarez

Índice

1. Desarrollo	2
1.1. Caso de Estudio 1	2
1.1.1. Inciso 1	4
1.1.2. Desafíos enfrentados y lecciones aprendidas	9
1.1.3. Identificación de logros	9
1.2. Caso de Estudio 2	10
1.2.1. Inciso 2	10
1.3. Caso de Estudio 3	12
1.3.1. Inciso 3	13
1.3.2. Desafíos enfrentados y lecciones aprendidas	18
1.3.3. Identificación de logros	18
2. Conclusiones	19
3. Bibliografía	20
4. Sitios Web	20
5. Repositorio GitHub	20

1. Desarrollo

1.1. Caso de Estudio 1

En el presente caso de estudio se analiza un motor de corriente continua sometido a un torque de carga no nulo. Su comportamiento puede representarse mediante un modelo equivalente en forma de circuito, tal como se ilustra en la figura que se presenta a continuación.

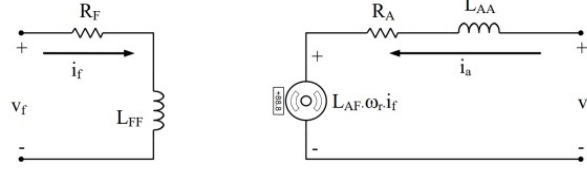


Figura 1: Modelo circuital equivalente

A partir del modelado del motor, es posible obtener las ecuaciones que describen su comportamiento dinámico:

$$\frac{di_a}{dt} = -\frac{R_A}{L_{AA}} \cdot i_a - \frac{K_m}{L_{AA}} \cdot \omega_r + \frac{1}{L_{AA}} \cdot V_a \quad (1)$$

$$\frac{d\omega_r}{dt} = \frac{K_i}{J} \cdot i_a - \frac{B_m}{J} \cdot \omega_r - \frac{1}{J} \cdot T_L \quad (2)$$

$$\frac{d\theta_t}{dt} = \omega_r \quad (3)$$

A partir del conjunto de ecuaciones diferenciales planteadas, se observa la presencia de tres integradores en el sistema, lo cual justifica la definición inicial de tres variables de estado:

$$x_1 = i_a$$

$$x_2 = \omega_r$$

$$x_3 = \theta_t$$

Cabe destacar que, según la forma en que se definan las variables de salida, el sistema puede caracterizarse como MISO (Multiple Input Single Output) o MIMO (Multiple Input Multiple Output). Esta clasificación depende de la cantidad de variables consideradas como salidas del sistema.

Teniendo en cuenta que el vector \dot{x} representa la derivada temporal del conjunto de variables de estado, es posible expresar la dinámica del sistema mediante la siguiente ecuación matricial de estado:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K_m}{L_a} & 0 \\ \frac{K_i}{J} & -\frac{B_m}{J} & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & -\frac{1}{J} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_a \\ T_L \end{bmatrix}$$

La definición de la ecuación de salida dependerá de los criterios adoptados en el análisis del sistema. No obstante, su formulación deberá ajustarse a la estructura general establecida para los modelos en espacio de estados:

$$y = C \cdot x + D \cdot u$$

Por otro lado, se tiene que la relación de entrada - salida que tiene el actuador no es 100 % lineal, sino que tiene una zona muerta de 0,5V como se muestra a continuación:

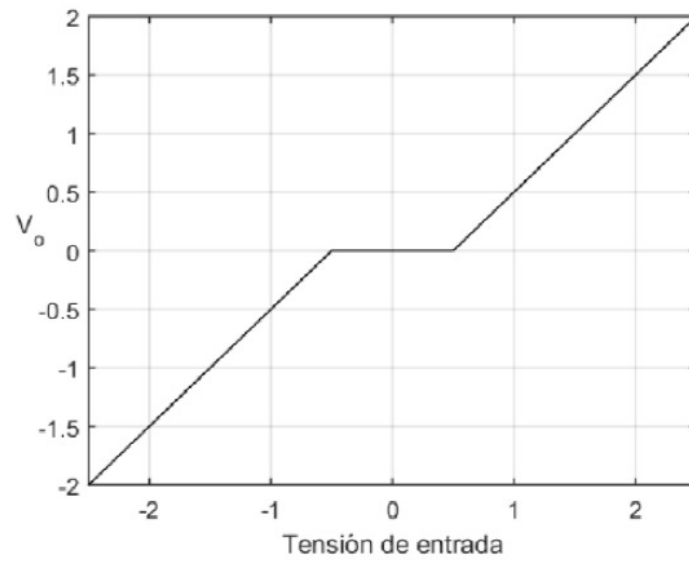


Figura 2: No linealidad del controlador

1.1.1. Inciso 1

Este inciso consiste en lo siguiente: empleando los parámetros hallados en el Trabajo Práctico N°1 para el Motor CC, implementar un sistema en variables de estado que controle el ángulo del motor, para consignas de $\pi/2$ y $-\pi/2$ cambiando cada 5 segundos y que el TL es el descrito en la planilla de datos, comparando el desempeño con el obtenido con el PID digital del TP N°1. Hallar el valor de integración de Euler adecuado.

Objetivo: acelerar la dinámica del controlador verificando el resultado con las curvas del archivo xls adjunto.

1. Evitando que la tensión supere los 5 Volts en valor absoluto, especificar el tiempo de muestreo necesario para que el controlador cumpla el objetivo.
2. Asumiendo que no puede medirse directamente la corriente, pero sí la velocidad y el ángulo, proponer un controlador que logre el objetivo.
3. Determinar el efecto de la no linealidad en la acción de control, y verificar cuál es el máximo valor admisible de esa no linealidad.

Entonces, los parámetros obtenidos anteriormente con el método de Chen son los siguientes:

$$K_i = 0,2804$$

$$J = 0,0031$$

$$B_m = 0$$

$$L_{aa} = 0,0217$$

$$R_a = 2,4390$$

$$K_m = 0,2623$$

Y ahora que se conocen los parámetros, se elaboraron las siguientes matrices de estado en tiempo continuo:

$$A_c = \begin{bmatrix} -\frac{R_a}{L_{aa}} & -\frac{K_m}{L_{aa}} & 0 \\ \frac{K_i}{J} & -\frac{B_m}{J} & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$B_c = \begin{bmatrix} \frac{1}{L_{aa}} & 0 \\ 0 & -\frac{1}{J} \\ 0 & 0 \end{bmatrix}$$

$$C_c = [0 \quad 0 \quad 1]$$

$$D_c = [0 \quad 0]$$

Ya teniendo las matrices de estado, se las asoció a un sistema de tiempo continuo llamado G con la orden `ss(A_c , B_c , C_c , D_c)` de MATLAB.

Ahora bien, con la orden $\text{eig}(A_c)$ se obtuvo los autovalores del sistema, y a continuación se obtuvo los valores de tiempos correspondientes a las dinámicas rápida y lenta del sistema:

$$t_R = 504,66 \mu s$$

$$t_L = 278,5 \text{ ms}$$

Posteriormente se tomó un tiempo de muestreo T_m cuatro veces menor al valor de t_R , y un tiempo total de simulación de 10 segundos para así poder capturar al menos dos ciclos de variación del ángulo de referencia.

A continuación se discretizó el sistema mediante un ZOH con la orden $c2d(G, T_m, 'zoh')$, se le asignó el nombre G_d al nuevo sistema y de ahí se obtuvieron las matrices de estado de tiempo discreto A_d , B_d , C_d y D_d .

Luego se procedió a elaborar las matrices de controlabilidad (M_c) y alcanzabilidad (M_a) como se muestra a continuación:

$$M_a = [B_d \quad A_d B_d \quad A_d^2 B_d]$$

$$M_c = [B_d \quad A_d B_d \quad A_d^2 B_d \quad A_d^3 B_d]$$

Entonces se calculó el rango de ambas matrices y se corroboró que ambos valores coincidían en un valor igual a 3, con lo cual se concluye que el sistema es controlable.

A continuación se definieron las matrices A y B del sistema ampliado:

$$A_A = \begin{bmatrix} A_d & \mathbf{0}_{3 \times 1} \\ -C_d(1, :)A_d & 1 \end{bmatrix}$$

$$B_A = \begin{bmatrix} B_d \\ -C_d(1, :)B_d \end{bmatrix}$$

Posteriormente, se procedió a diseñar el controlador LQR, y para hacer eso se dimensionó sus elementos a prueba y error hasta que el control fuera lo suficientemente bueno. Las matrices d_c (matriz de pesos), Q_c y el valor R_c (penalización sobre la acción de control) quedaron de la siguiente manera:

$$d_c = [9000 \quad 900 \quad 0,004 \quad 0,3]$$

$$Q_c = \text{diag}(d_c) = \begin{bmatrix} 9000 & 0 & 0 & 0 \\ 0 & 900 & 0 & 0 \\ 0 & 0 & 0,004 & 0 \\ 0 & 0 & 0 & 0,3 \end{bmatrix}$$

$$R_c = 0,001$$

Cabe aclarar que la matriz de pesos d_c tiene sus pesos colocados según el siguiente orden de variables: corriente, velocidad, ángulo, psita.

Entonces, ya teniendo las matrices anteriormente calculadas se procedió a calcular el controlador con las siguientes instrucciones de MATLAB:

$$K_{\text{LQR.C}} = \text{dlqr}(A_A, B_A, Q_c, R_c)$$

$$K = K_{\text{LQR.C}}(1:3)$$

$$K_i = -K_{\text{LQR.C}}(4)$$

A continuación se definen las matrices A, B y C del sistema dual:

$$A_o = A_d^T$$

$$B_o = C_d^T$$

$$C_o = B_d^T$$

Ahora teniendo las matrices A, B y C del sistema dual, se procedió a diseñar el observador LQR como se muestra a continuación:

$$d_o = [0,01 \quad 10^6 \quad 0,1]$$

$$Q_o = \text{diag}(d_o) = \begin{bmatrix} 0,01 & 0 & 0 \\ 0 & 10^6 & 0 \\ 0 & 0 & 0,1 \end{bmatrix}$$

$$R_o = 0,005$$

$$K_o = (\text{dlqr}(A_o, B_o, Q_o, R_o))^T$$

Ya teniendo todos los elementos calculados anteriormente, calculando la cantidad de puntos a utilizar para el Euler, y definiendo una zona muerta de $\pm 0,5V$, se procedió a hacer la simulación del controlador (más detalles sobre el Euler en el repositorio GitHub) para el sistema correspondiente y se obtuvieron las siguientes gráficas:

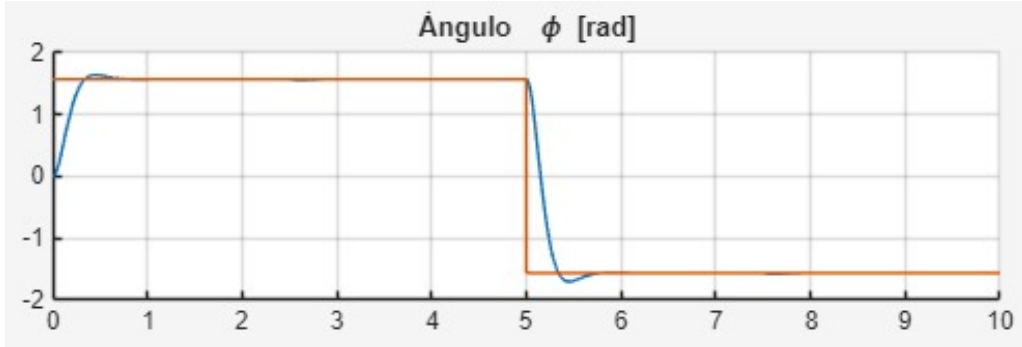


Figura 3: Ángulo en función del tiempo

Como puede verse, el sistema llega a estabilizarse mucho antes de que cambie el ángulo de referencia con un sobrepaso ínfimo.



Figura 4: Corriente en función del tiempo

Como puede verse, la corriente nunca supera el valor de 1A.

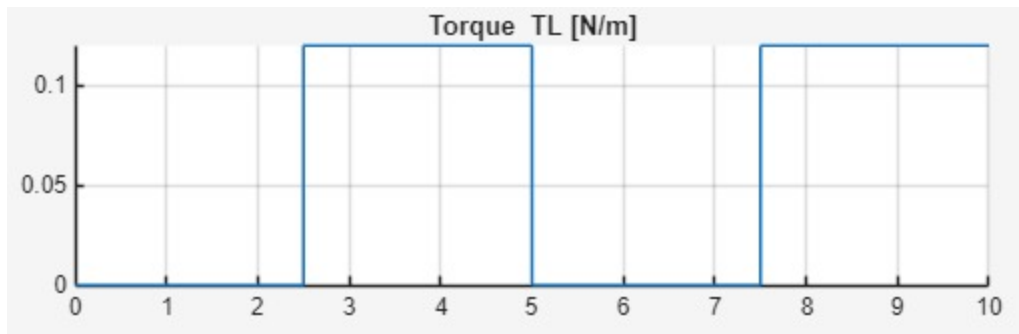


Figura 5: Torque en función del tiempo

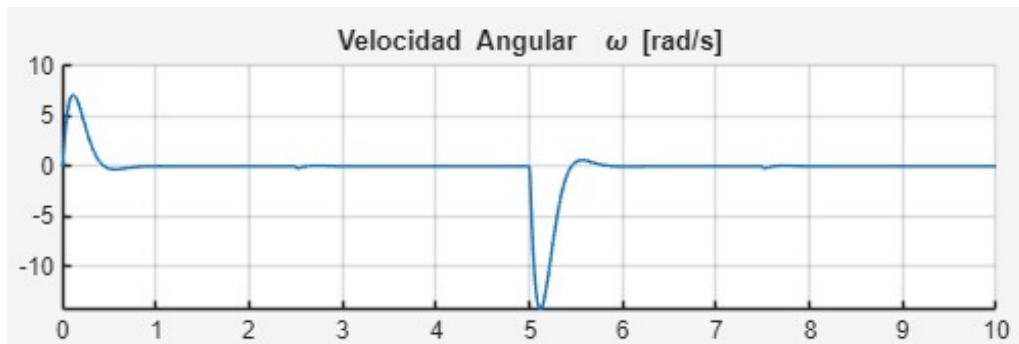


Figura 6: Velocidad angular en función del tiempo



Figura 7: Acción de control en función del tiempo

Como puede verse, la acción de control está acotada en el rango $[-5V;5V]$.

Ahora bien, es importante mencionar nuevamente que los resultados mostrados corresponden al caso de una no linealidad de 0,5V. A continuación se muestra qué sucede con la performance del controlador cuando se aplica una no linealidad mayor (3V):

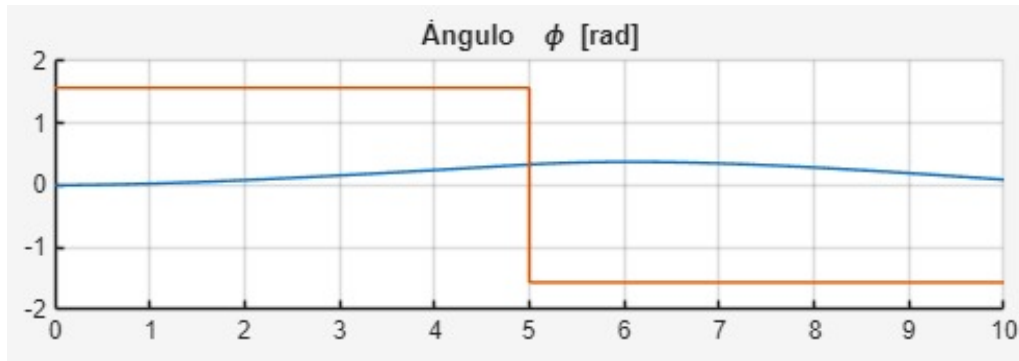


Figura 8: Ángulo en función del tiempo

Como puede verse, para una zona muerta de $\pm 3V$ suceden dos cosas: el sistema se hace extremadamente lento y además tiene un error en régimen demasiado grande (puede verse que el ángulo tiende a un valor considerablemente menor a $\pi/2$).

1.1.2. Desafíos enfrentados y lecciones aprendidas

Durante el desarrollo del sistema de control basado en variables de estado para el motor de corriente continua, surgieron múltiples desafíos tanto desde el punto de vista teórico como práctico. Uno de los primeros desafíos consistió en la correcta formulación del modelo dinámico en espacio de estados, partiendo de las ecuaciones diferenciales del sistema físico. La identificación adecuada de las variables de estado y la correcta formulación de las matrices A_c , B_c , C_c y D_c fue clave para garantizar la precisión del modelo y, en consecuencia, del diseño del controlador.

Un aspecto particularmente desafiante fue la discretización del sistema. Dado que la dinámica rápida del motor se encontraba en el orden de los microsegundos, fue necesario definir un tiempo de muestreo significativamente pequeño (cuatro veces menor que la constante de tiempo rápida) para garantizar una buena representación del sistema en tiempo discreto. Esto implicó un compromiso entre fidelidad del modelo y carga computacional.

Otro aprendizaje relevante fue el diseño del controlador LQR. Si bien el método proporciona una solución óptima teórica, en la práctica fue necesario ajustar los pesos de la matriz Q_c y el valor de penalización R_c de forma iterativa, hasta lograr una respuesta rápida, sin sobrepaso significativo y sin violar las restricciones sobre la acción de control (en este caso, mantener la tensión dentro del rango de $\pm 5V$). Este proceso evidenció la necesidad de combinar teoría con juicio ingenieril en el diseño de controladores.

Asimismo, al asumir que no se disponía de una medición directa de la corriente, se abordó el diseño de un observador de estados mediante LQR dual. En este contexto, se observó que la correcta selección de los pesos del observador afectaba drásticamente la calidad de la estimación, y por ende, el desempeño del sistema controlado.

Finalmente, uno de los aspectos más significativos fue el tratamiento de la no linealidad introducida por la zona muerta del actuador. Si bien una zona muerta de $\pm 0.5V$ no comprometía el desempeño, al incrementarla a $\pm 3V$ se verificó un deterioro crítico del sistema: la respuesta se tornó extremadamente lenta y apareció un error en régimen inadmisibles. Esta situación permitió comprobar empíricamente la sensibilidad del controlador a no linealidades no modeladas, y la necesidad de considerar mecanismos compensatorios en aplicaciones reales.

1.1.3. Identificación de logros

A continuación se enumeran los logros alcanzados durante la resolución de este inciso del trabajo práctico:

- ID1.1.1.1 Analiza la factibilidad de controlar un proceso real conociendo su modelo.
- ID1.1.1.2 Expresa el modelo lineal multivariable de un proceso real en un punto de operación.
- ID1.1.1.4 Selecciona las variables de estado de un proceso lineal para generar una expresión matricial lineal.
- ID1.1.1.5 Infiere la evolución temporal de procesos reales representados en variables de estado.
- ID1.1.1.6 Diseña observadores de procesos linealizados para controlarlos cuando no es viable medir sus variables.
- ID1.1.1.7 Diseña controladores para obtener una dinámica estable del proceso linealizado.

1.2. Caso de Estudio 2

1.2.1. Inciso 2

Este inciso consiste en lo siguiente: implementar el mismo sistema controlador del ítem anterior, empleando los valores del motor siguiente:

$$R_a = 2,27$$

$$L_a = 0,0047$$

$$K_i = 0,25$$

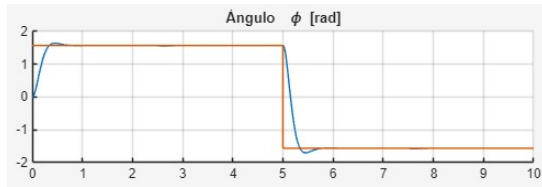
$$K_m = 0,25$$

$$B_m = 0,00131$$

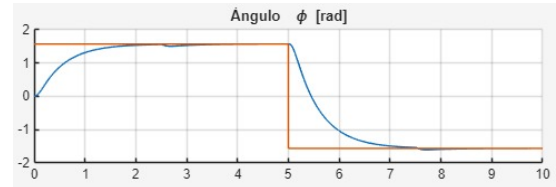
$$J_m = 0,00233$$

Generar gráficas de evolución de la velocidad y ángulo comparando los desempeños cualitativamente para todos los casos.

Para este caso se utilizó el controlador diseñado en el primer inciso y se lo puso a prueba (con método de Euler) para los parámetros recién enumerados. A continuación se muestra una comparativa en la performance del controlador para el caso en el que se utilizaron los parámetros calculados con el método de Chen y para el caso que ahora se propone analizar (parámetros propuestos por la consigna):



(a) Ángulo en función del tiempo para parámetros del Chen

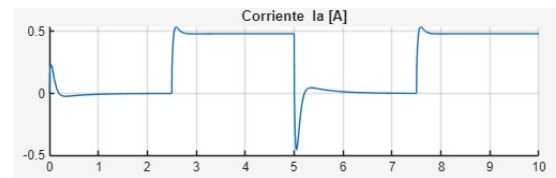


(b) Ángulo en función del tiempo para parámetros reales

Como puede verse, la performance del controlador a la hora de aplicar los parámetros reales se empobrece en cuanto a tiempo de establecimiento, aunque deja de tener sobrepasamiento.

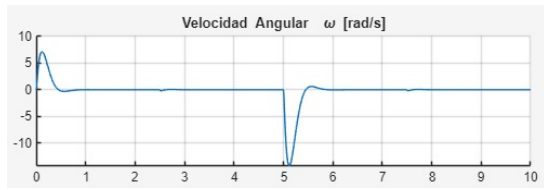


(a) Corriente en función del tiempo para parámetros del Chen



(b) Corriente en función del tiempo para parámetros reales

Como puede verse, aumenta considerablemente el consumo de corriente, pero eso no es un problema, ya que el consumo sigue siendo menor a 1A.



(a) Velocidad angular en función del tiempo para parámetros del Chen



(b) Velocidad angular en función del tiempo para parámetros reales

Como puede verse, se redujo considerablemente la velocidad con la que el motor se mueve, lo cual era esperable debido a que como se vio más arriba, aumentó el tiempo de establecimiento de la variable de posición angular.



(a) Acción de control en función del tiempo para parámetros del Chen



(b) Acción de control en función del tiempo para parámetros reales

Como puede verse, se redujo considerablemente la acción de control, lo cual explica que el sistema ya no tenga tan buena performance en cuanto a rapidez.

1.3. Caso de Estudio 3

Para este caso de estudio se propone analizar y controlar el movimiento de un carro que sostiene una pequeña masa (a modo de grúa). El esquema del sistema es el siguiente:

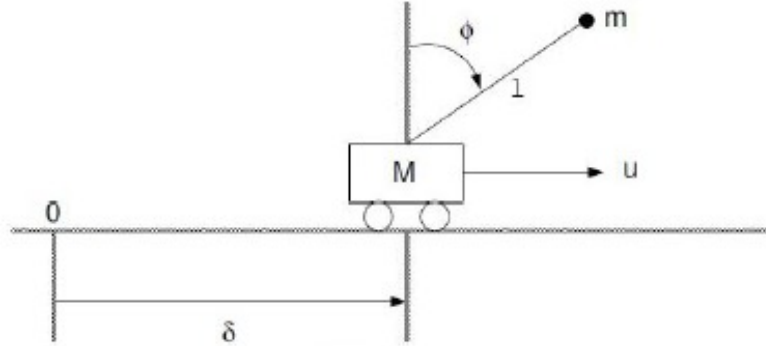


Figura 13: Sistema del péndulo

Para el caso del esquema del péndulo invertido presentado, el modelo matemático es el siguiente:

$$(M + m)\ddot{\delta} + ml\ddot{\phi}\cos\phi - ml\dot{\phi}^2\sin\phi + F\dot{\delta} = u$$

$$l\ddot{\phi} - g\sin\phi + \ddot{\delta}\cos\phi = 0$$

Además se tiene el vector de estados definido de la siguiente manera:

$$\mathbf{x} = [\delta \quad \dot{\delta} \quad \phi \quad \dot{\phi}]^T$$

Y los valores asignados para los parámetros del sistema son los siguientes:

$$m = 0,1$$

$$F = 0,1$$

$$l = 1,6$$

$$g = 9,8$$

$$M = 1,5$$

1.3.1. Inciso 3

Este inciso consiste en calcular el sistema controlador que haga evolucionar al péndulo en el equilibrio estable.

El objetivo de control consiste en lo siguiente: partiendo de una condición inicial nula en el desplazamiento y el ángulo en π , hacer que el carro se desplace a 10 metros evitando las oscilaciones de la masa m , considerando que es una grúa. Una vez que $\delta = 10$, modificar a m a un valor 10 veces mayor y volver al origen evitando oscilaciones.

Entonces, las consignas a seguir son las siguientes:

- Considerar que sólo puede medirse el desplazamiento δ y el ángulo ϕ .
- Determinar Δt y el tiempo de simulación necesario para el controlador en tiempo continuo.
- Especificar el rango posible para el tiempo de muestreo para implementar el sistema en un micro-controlador.
- Determinar el efecto de la no linealidad en la acción de control, y verificar cuál es el máximo valor admisible de esa no linealidad.

Lo primero que se hizo fue definir las matrices de estado A, B, C y D (en tiempo continuo), para lo cual se tuvo en cuenta la necesidad de definir dos veces a la matriz A: una vez para el primer recorrido (ida desde los 0 metros hasta los 10 metros) y otra vez para el segundo recorrido (vuelta desde los 10 metros hasta los 0 metros). Esto de definir dos veces a la matriz A se hizo así porque la masa del péndulo no es la misma en ambos recorridos. Entonces las matrices quedan definidas de la siguiente manera:

$$A_{c1} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{F}{M} & -\frac{m_1 g}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{F}{lM} & -\frac{g(m_1+M)}{lM} & 0 \end{bmatrix}$$

$$A_{c2} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{F}{M} & -\frac{m_2 g}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{F}{lM} & -\frac{g(m_2+M)}{lM} & 0 \end{bmatrix}$$

$$B_c = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{lM} \end{bmatrix}$$

$$C_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$D_c = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Ya teniendo las matrices de estado, se las asoció a dos sistemas de tiempo continuo llamados G_{c1} y G_{c2} con las órdenes $ss(A_{c1}, B_c, C_c, D_c)$ y $ss(A_{c2}, B_c, C_c, D_c)$ (respectivamente) de MATLAB.

Posteriormente, se procedió a calcular los autovalores de las matrices A_{c1} y A_{c2} con el fin de calcular las dinámicas del sistema y poder especificar el tiempo de muestreo y el tiempo total de simulación, llegando así a un tiempo de muestreo de 0,5ms y un tiempo total de simulación de 40 segundos.

A continuación se procedió a discretizar los sistemas mediante un ZOH con la orden $c2d(G_c, Tm, 'zoh')$, y se le asignaron los nombres G_{d1} y G_{d2} a los nuevos sistemas, y de ahí se obtuvieron las matrices de estado de tiempo discreto para ambos sistemas.

Luego se procedió a elaborar las matrices de controlabilidad (M_{c1} y M_{c2}) y alcanzabilidad (M_{a1} y M_{a2}) como se muestra a continuación:

$$M_{c1} = \begin{bmatrix} B_{d1} & A_{d1}B_{d1} & A_{d1}^2B_{d1} & A_{d1}^3B_{d1} & A_{d1}^4B_{d1} \end{bmatrix}$$

$$M_{c2} = \begin{bmatrix} B_{d2} & A_{d2}B_{d2} & A_{d2}^2B_{d2} & A_{d2}^3B_{d2} & A_{d2}^4B_{d2} \end{bmatrix}$$

$$M_{a1} = \begin{bmatrix} B_{d1} & A_{d1}B_{d1} & A_{d1}^2B_{d1} & A_{d1}^3B_{d1} \end{bmatrix}$$

$$M_{a2} = \begin{bmatrix} B_{d2} & A_{d2}B_{d2} & A_{d2}^2B_{d2} & A_{d2}^3B_{d2} \end{bmatrix}$$

Entonces se calculó el rango de las cuatro matrices y se corroboró que todas tenían rango igual a 4, con lo cual se concluye que el sistema es controlable dada la igualdad de rangos.

A continuación se definieron las matrices A y B de ambos sistemas ampliados:

$$A_{a1} = \begin{bmatrix} A_{d1} & \mathbf{0}_{4 \times 1} \\ -C_{d1}(1, :)A_{d1} & I_1 \end{bmatrix}$$

$$B_{a1} = \begin{bmatrix} B_{d1} \\ -C_{d1}(1, :)B_{d1} \end{bmatrix}$$

$$A_{a2} = \begin{bmatrix} A_{d2} & \mathbf{0}_{4 \times 1} \\ -C_{d2}(1, :)A_{d2} & I_1 \end{bmatrix}$$

$$B_{a2} = \begin{bmatrix} B_{d2} \\ -C_{d2}(1, :)B_{d2} \end{bmatrix}$$

Posteriormente, se procedió a diseñar los controladores LQR, y para eso se dimensionó el valor numérico de sus elementos a prueba y error hasta que el control fuera lo suficientemente bueno. El diseño quedó de la siguiente manera:

$$d_1 = \begin{bmatrix} 10 & 10 & 100 & 100 & 0,001 \end{bmatrix} \Rightarrow Q_1 = \text{diag}(d_1), \quad R_1 = 1$$

$$[K_{\text{LQR1}}, -, -] = \text{dlqr}(A_{a1}, B_{a1}, Q_1, R_1)$$

$$K_1 = K_{\text{LQR1}}(1:4), \quad K_{i1} = -K_{\text{LQR1}}(5)$$

$$d_2 = \begin{bmatrix} 10 & 10 & 100 & 100 & 0,001 \end{bmatrix} \Rightarrow Q_2 = \text{diag}(d_2), \quad R_2 = 1$$

$$[K_{\text{LQR2}}, -, -] = \text{dlqr}(A_{a2}, B_{a2}, Q_2, R_2)$$

$$K_2 = K_{\text{LQR2}}(1:4), \quad K_{i2} = -K_{\text{LQR2}}(5)$$

A continuación se procedió a diseñar los observadores LQR, y para ello primero se definieron los sistemas duales de la siguiente manera:

$$A_{o1} = A_{d1}^\top, \quad B_{o1} = C_{d1}^\top, \quad C_{o1} = B_{d1}^\top$$

$$A_{o2} = A_{d2}^\top, \quad B_{o2} = C_{d2}^\top, \quad C_{o2} = B_{d2}^\top$$

Ya teniendo definido el sistema dual, se procedió a diseñar las matrices d_o , Q_o y R_o de los observadores como se muestra a continuación:

$$d_{o1} = \begin{bmatrix} 1 & 50 & 500 & 0,1 \end{bmatrix}, \quad Q_{o1} = \text{diag}(d_{o1}), \quad R_{o1} = \text{diag}([0,1 \quad 0,1])$$

$$d_{o2} = \begin{bmatrix} 1 & 50 & 500 & 0,1 \end{bmatrix}, \quad Q_{o2} = \text{diag}(d_{o2}), \quad R_{o2} = \text{diag}([0,1 \quad 0,1])$$

A continuación se procedió a realizar el cálculo del observador de la siguiente manera:

$$K_{o1} = (\text{dlqr}(A_{o1}, B_{o1}, Q_{o1}, R_{o1}))^\top$$

$$K_{o2} = (\text{dlqr}(A_{o2}, B_{o2}, Q_{o2}, R_{o2}))^\top$$

Ya teniendo hechos todos los cálculos anteriormente especificados y definiendo las condiciones iniciales del sistema, se procedió a hacer la simulación (más detalles en el repositorio GitHub) del mismo con la acción del controlador, y los resultados obtenidos son lo que se muestran a continuación:



Figura 14: Velocidad angular del péndulo

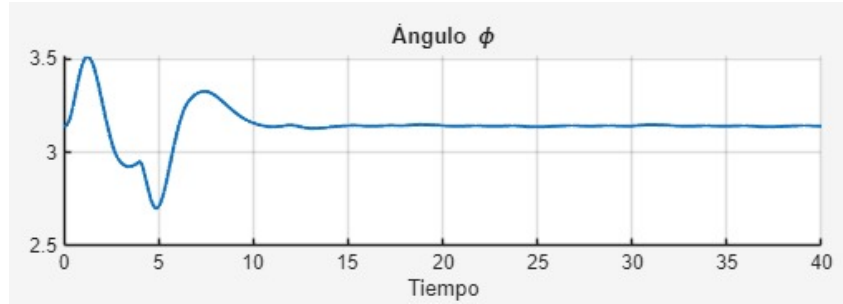


Figura 15: Posición angular del péndulo

Como puede verse, la posición angular del péndulo varía a lo sumo un 11,4% con respecto a π , que es el valor de referencia.



Figura 16: Velocidad de la grúa



Figura 17: Posición de la grúa



Figura 18: Acción de control

Ahora bien, cabe aclarar que los resultados obtenidos corresponden a una no linealidad de $\pm 0,5$ para el controlador. A continuación se muestra cómo la performance del controlador se empobrece considerablemente cuando se lleva el valor de la no linealidad al límite:



Figura 19: Velocidad angular del péndulo

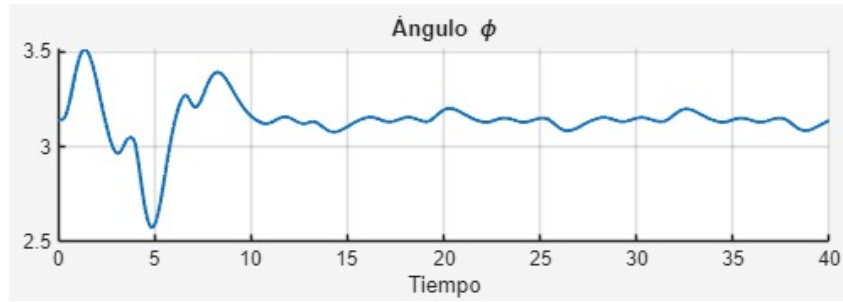


Figura 20: Posición angular del péndulo

Como puede verse, ahora al péndulo le cuesta mucho más estabilizar su posición, es decir, se hace más evidente la condición de equilibrio inestable.



Figura 21: Velocidad de la grúa



Figura 22: Posición de la grúa

Como puede observarse, se hace evidente que la grúa ya no es capaz de mantener una posición cercana al reposo.

1.3.2. Desafíos enfrentados y lecciones aprendidas

El desarrollo del sistema de control para una grúa con una masa suspendida presentó múltiples desafíos tanto desde el punto de vista teórico como práctico. Uno de los principales retos fue abordar la **no linealidad inherente** al sistema físico, particularmente en la interacción entre el desplazamiento del carro y la dinámica angular del péndulo invertido.

Un segundo desafío relevante fue el **cambio abrupto de condiciones dinámicas** debido a la variación de la masa suspendida (de m_1 a $m_2 = 10 \cdot m_1$). Este cambio requirió rediseñar matrices de estado y calcular controladores y observadores adaptados a cada condición. La necesidad de mantener un rendimiento aceptable en ambas etapas del recorrido (ida y vuelta) obligó a plantear controladores LQR separados, lo cual aumentó la complejidad del diseño y del código de simulación.

Además, el diseño del controlador implicó determinar **parámetros de ponderación adecuados** en las matrices Q y R del algoritmo LQR. Esto se realizó por ensayo y error, priorizando un compromiso entre estabilidad, suavidad de la acción de control y rápida convergencia hacia la referencia, evitando al mismo tiempo oscilaciones indeseadas.

El diseño del observador fue otro punto crítico, ya que solo se disponía de mediciones del desplazamiento del carro y del ángulo del péndulo. Esta restricción exigió el uso de un **observador de estado basado en LQR dual**, el cual resultó ser efectivo pero sensible al ajuste de ganancias y a la elección del modelo discreto. Se comprobó que incluso pequeñas variaciones en los valores de Q_o y R_o modificaban considerablemente la respuesta estimada del sistema.

En cuanto a la implementación temporal, se enfrentó el reto de determinar un **tiempo de muestreo adecuado**, lo suficientemente pequeño para capturar la dinámica rápida del sistema, pero sin sobrecargar los recursos computacionales. Se concluyó que un tiempo de muestreo de 0.5 ms era necesario para garantizar la precisión y estabilidad del sistema discreto, aunque ello representa un reto para implementaciones en tiempo real.

Finalmente, se observó que **la tolerancia a las no linealidades es limitada**. Si bien el sistema se comporta adecuadamente para una no linealidad moderada ($\pm 0,5$), al incrementar este rango, el desempeño del controlador se degrada notablemente, como se evidenció en las simulaciones.

1.3.3. Identificación de logros

A continuación se enumeran los logros alcanzados durante la resolución de este inciso del trabajo práctico:

- ID1.1.1.1 Analiza la factibilidad de controlar un proceso real conociendo su modelo.
- ID1.1.1.2 Expresa el modelo lineal multivariable de un proceso real en un punto de operación.
- ID1.1.1.4 Selecciona las variables de estado de un proceso lineal para generar una expresión matricial lineal.
- ID1.1.1.5 Infiere la evolución temporal de procesos reales representados en variables de estado.
- ID1.1.1.6 Diseña observadores de procesos linealizados para controlarlos cuando no es viable medir sus variables.
- ID1.1.1.7 Diseña controladores para obtener una dinámica estable del proceso linealizado.

2. Conclusiones

El presente trabajo práctico abordó tres instancias clave en el estudio y diseño de controladores aplicados a sistemas dinámicos, mediante una metodología basada en variables de estado y realimentación óptima.

En el primer inciso, se desarrolló un modelo detallado del motor de corriente continua a partir de parámetros identificados experimentalmente, sobre el cual se diseñó un controlador óptimo tipo LQR con realimentación de estados y observador. Esta solución permitió mejorar notablemente la respuesta del sistema respecto del controlador PID utilizado previamente, cumpliendo con las restricciones sobre la acción de control y acelerando la dinámica de seguimiento del ángulo del eje motor. Se destacó también el impacto de la zona muerta del actuador, cuyo efecto fue cuantificado y analizado con el fin de determinar el límite de no linealidad tolerable por el sistema.

En el segundo inciso, se validó el desempeño del controlador diseñado aplicándolo al modelo del motor con sus parámetros reales. Este paso fue fundamental para analizar la robustez del diseño frente a incertezas de modelado. Se verificó que, aunque se observa una degradación en la calidad de la respuesta respecto del modelo estimado, el sistema conserva su estabilidad y desempeño aceptable, siempre y cuando las variaciones paramétricas no excedan ciertos límites.

Finalmente, en el tercer inciso, se abordó el problema del control de una grúa con carga suspendida, un sistema inherentemente subactuado y multivariable. A través del modelado linealizado y el diseño de un controlador en espacio de estados, se logró minimizar tanto el error en posición como las oscilaciones de la carga. Este caso resaltó la potencia del control óptimo para regular dinámicas acopladas y sensibles, típicas en sistemas con restricciones físicas y cinemáticas complejas.

En conjunto, los tres incisos permitieron aplicar herramientas avanzadas de modelado, análisis y control sobre sistemas reales, consolidando así criterios teóricos y prácticos fundamentales en el diseño moderno de controladores.

3. Bibliografía

Referencias

- [1] Pucheta, J. A. *Apuntes – Sistemas de Control II*. 2025. FCEFyN - UNC.
- [2] Nise, N. S. *Control Systems Engineering*. 7th ed., Wiley, 2015.
- [3] Ogata, K. *Modern Control Engineering*. 5th ed., Prentice Hall, 2010.
- [4] Dorf, R. C., Bishop, R. H. *Modern Control Systems*. 13th ed., Pearson, 2017.
- [5] Franklin, G. F., Powell, J. D., Emami-Naeini, A. *Feedback Control of Dynamic Systems*. 8th ed., Pearson, 2019.
- [6] Kwakernaak, H., Sivan, R. *Linear Optimal Control Systems*. Wiley-Interscience, 1972.
- [7] Slotine, J.-J. E., Li, W. *Applied Nonlinear Control*. Prentice Hall, 1991.

4. Sitios Web

- **MathWorks – MATLAB y Simulink** <https://www.mathworks.com> Documentación oficial de MATLAB y Simulink, herramientas ampliamente utilizadas para modelado, simulación y análisis de sistemas dinámicos.
- **Control Tutorials for MATLAB and Simulink – University of Michigan** <https://ctms.engin.umich.edu/CTMS> Sitio educativo con tutoriales paso a paso sobre modelado, diseño y simulación de sistemas de control.
- **IEEE Xplore Digital Library** <https://ieeexplore.ieee.org> Biblioteca digital con artículos académicos y conferencias especializadas en ingeniería eléctrica, sistemas de control, identificación de sistemas, entre otros.

5. Repositorio GitHub

El código fuente y la documentación complementaria del trabajo práctico se encuentran disponibles en el siguiente enlace:

<https://github.com/Esteban3003/SistC2-Esteban-Suarez/tree/main/TP2>