

INSTITUTO TECNOLÓGICO DE BUENOS AIRES

CONTROL AUTOMÁTICO

TRABAJO PRÁCTICO FINAL

PÉNDULO INVERTIDO DE FURUTA

Alumnos:

Tobias SCALA
Leg. 55391

Guido PANAGGIO
Leg. 56214

Profesores:

Alejandro S. GHERSIN

Reurison SILVA RODRIGUES

Entregado: 6 de Octubre de 2021

Índice general

1. Introducción	2
2. Identificación del Sistema	3
2.1. Ensayo B	3
2.2. Ensayo D	4
2.3. Sistema en Espacio de Estados	4
3. Feedback control	5
3.1. Limitaciones	5
3.2. Diseño del controlador	5
3.3. Diseño del observador	6
3.4. Discretización	7
3.5. Implementación en micro-controlador	7
3.6. Simulación y resultados	7
3.6.1. Tiempo continuo	8
3.6.2. Tiempo Discreto	8
3.6.3. Simulación en micro-controlador	9
4. Loopshaping	11
4.1. Lazo Secundario	11
4.2. Lazo Primario	12
4.3. Simulación y Resultados	14
4.3.1. Tiempo Continuo	15
4.3.2. Tiempo Discreto	16
5. Conclusiones	19

Introducción

Es sabido que el Péndulo Invertido con Carrito es un sistema muy popular en el área de control. El mismo ha sido profundamente estudiado con el tiempo y se han publicado numerosos papers explicando tanto las ecuaciones físicas como la implementación de un controlador con la finalidad de mantener el péndulo en posición vertical. Dicho sistema presenta una limitación física el cual es la longitud máxima que puede recorrer el carrito (causada por el movimiento lineal del mismo). Esta limitación es salvada reemplazando el movimiento lineal por un movimiento rotativo. El sistema que salva esta limitación es el Péndulo de Furuta, el mismo será estudiado con profundidad en el presente trabajo de Control Automático.

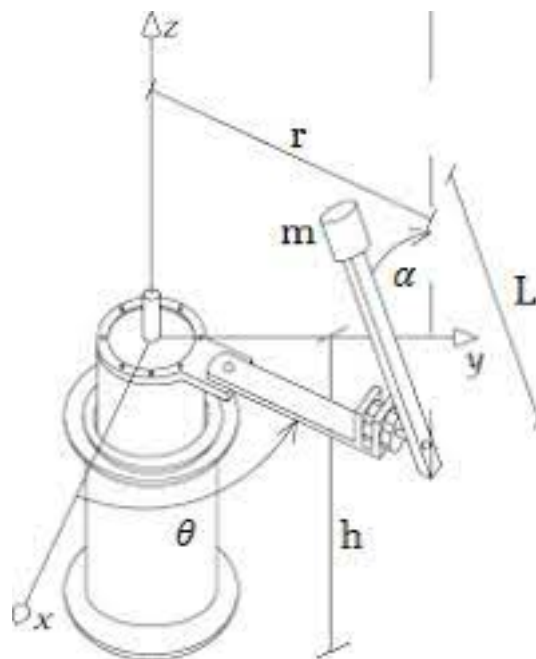


Figura 1.1: Péndulo de Furuta

Para el estudio del sistema Péndulo de Furuta se cuenta con papers y ensayos propuestos por la cátedra. Los ensayos son puramente virtuales simulando el modelo físico del sistema a través del Simscape de Matlab. Se experimentarán con estos ensayos con la finalidad de identificar el sistema y así obtener su función transferencia. Posteriormente, se diseñarán 2 tipos de controladores distintos (Loop Shaping y Espacio de Estados, cada uno obedeciendo la metodologías de diseño diseñado) para estabilizar el sistema y así cumplir el objetivo de mantener el péndulo en posición vertical. Dichos controladores serán diseñados en tiempo continuo y luego serán discretizados con un cierto período de muestreo para poder ser implementados con un pseudocódigo programado en Matlab, así como su posible implementación en un micro-controlador.

En primer lugar se presentará el diseño por realimentación lineal de estados y posteriormente el diseño por Loopshaping.

Identificación del Sistema

Para poder identificar el sistema, primero se deben plantear las ecuaciones diferenciales físicas que representan el mismo. Haciendo uso del paper *On the Dynamics of the Furuta Pendulum* (Benjamin Seth Cazzolato and Zebb Prime), se pueden conocer dichas ecuaciones las cuales se pueden observar en la siguiente imagen.

$$\ddot{\theta}_1 = \frac{\begin{pmatrix} \begin{bmatrix} -\hat{J}_2 b_1 \\ m_2 L_1 l_2 \cos(\theta_2) b_2 \\ -\hat{J}_2^2 \sin(2\theta_2) \\ -(1/2) \hat{J}_2 m_2 L_1 l_2 \cos(\theta_2) \sin(2\theta_2) \\ \hat{J}_2 m_2 L_1 l_2 \sin(\theta_2) \end{bmatrix}^T \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_1 \dot{\theta}_2 \\ \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \end{bmatrix} \\ + \begin{bmatrix} \hat{J}_2 \\ -m_2 L_1 l_2 \cos(\theta_2) \\ (1/2) m_2^2 l_2^2 L_1 \sin(2\theta_2) \end{bmatrix}^T \begin{bmatrix} \tau_1 \\ \tau_2 \\ g \end{bmatrix} \end{pmatrix}}{(\hat{J}_0 \hat{J}_2 + \hat{J}_2^2 \sin^2(\theta_2) - m_2^2 L_1^2 l_2^2 \cos^2(\theta_2))}$$

$$\ddot{\theta}_2 = \frac{\begin{pmatrix} \begin{bmatrix} m_2 L_1 l_2 \cos(\theta_2) b_1 \\ -b_2 (\hat{J}_0 + \hat{J}_2 \sin^2(\theta_2)) \\ m_2 L_1 l_2 \hat{J}_2 \cos(\theta_2) \sin(2\theta_2) \\ -(1/2) \sin(2\theta_2) [\hat{J}_0 \hat{J}_2 + \hat{J}_2^2 \sin^2(\theta_2)] \\ -(1/2) m_2^2 L_1^2 l_2^2 \sin(2\theta_2) \end{bmatrix}^T \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_1 \dot{\theta}_2 \\ \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \end{bmatrix} \\ + \begin{bmatrix} -m_2 L_1 l_2 \cos(\theta_2) \\ \hat{J}_0 + \hat{J}_2 \sin^2(\theta_2) \\ -m_2 l_2 \sin(\theta_2) (\hat{J}_0 + \hat{J}_2 \sin^2(\theta_2)) \end{bmatrix}^T \begin{bmatrix} \tau_1 \\ \tau_2 \\ g \end{bmatrix} \end{pmatrix}}{(\hat{J}_0 \hat{J}_2 + \hat{J}_2^2 \sin^2(\theta_2) - m_2^2 L_1^2 l_2^2 \cos^2(\theta_2))}$$

Figura 2.1: Ecuaciones diferenciales físicas del Péndulo de Furuta

Con estas ecuaciones diferenciales se puede observar que el sistema no es lineal (por lo tanto tendrá que ser linealizado alrededor de un punto de equilibrio) y que el mismo tiene 4 estados (los cuales son θ_1 , $\dot{\theta}_1$, θ_2 y $\dot{\theta}_2$). Siendo $\theta_1 = \alpha$ el ángulo del brazo rotatorio (arm) y $\theta_2 = \beta$ el ángulo del péndulo (pendulum). Cabe mencionar que el punto de equilibrio con el que se linealizará el sistema será aquél en el cual el sistema se estabilizará. Dicho punto es $\theta_1 = \alpha = 0$, $\dot{\theta}_1 = \dot{\alpha} = 0$, $\theta_2 = \beta = \pi$ y $\dot{\theta}_2 = \dot{\beta} = 0$.

También se puede observar, a través de las ecuaciones, que se tienen las siguientes incógnitas, las cuales son:

- J_{z1} : Momento de inercia del brazo rotatorio respecto del eje de giro.
- L_1 : Longitud del brazo rotatorio.
- J_{z2} : Momento de inercia del péndulo respecto del eje de giro.
- l_2 : Longitud desde el eje de giro hasta el centro de masa del péndulo.
- m_2 : Masa del péndulo.

Para poder obtener el valor de cada una de estas incógnitas, se realizan 2 ensayos virtuales los cuales se denominan *Ensayo B* y *Ensayo D*.

2.1. ENSAYO B

El ensayo B consiste en aplicar un torque constante de $1N \cdot m$ al brazo rotatorio en ausencia del péndulo para obtener la incógnita J_{z1} . Haciendo uso de la segunda ley de Newton para movimientos rotatorios se obtiene:

$$T = J_{z1} \cdot \ddot{\alpha}$$

$$J_{z1} = \frac{T \cdot (t_f - t_i)}{(\omega_f - \omega_i)}$$

$$J_{z1} = 1,75 \cdot 10^{-5} Kg \cdot m^2$$

A través de los archivos CAD de las piezas que conforman el largo del brazo rotatorio, se obtuvo que $L_1 = 0,08787m$.

2.2. ENSAYO D

El ensayo D consiste en 2 experimentos. El primer experimento consiste en aplicar un torque constante de $0,002N \cdot m$ al péndulo en ausencia de la fricción y la gravedad para obtener la incognita J_{z2} . Haciendo uso de la segunda ley de Newton para movimientos rotatorios se obtiene:

$$\begin{aligned} T &= J_{z2} \cdot \ddot{\beta} \\ J_{z2} &= \frac{T \cdot (t_f - t_i)}{(\omega_f - \omega_i)} \\ J_{z2} &= 4,29 \cdot 10^{-5} Kg \cdot m^2 \end{aligned}$$

El segundo experimento consiste en aplicar un torque constante de $0,002N \cdot m$ al péndulo en presencia de la fricción y la gravedad para obtener la incognita m_2 . Haciendo uso de la segunda ley de Newton para movimientos rotatorios se obtiene:

$$T - b_2 \dot{\beta} - l_2 \cdot (m_2 g \cdot \sin \beta) = J_{z2} \cdot \ddot{\beta}$$

Viendo que, en el experimento, el sistema se estabiliza, entonces tenemos que $\dot{\beta} = 0$ y que $\ddot{\beta} = 0$. Por lo tanto:

$$\begin{aligned} T &= l_2 \cdot (m_2 g \cdot \sin \beta) \\ l_2 &= \frac{T}{m_2 g \cdot \sin \beta} \\ l_2 &= 0,0867m \end{aligned}$$

A través de los archivos CAD de las piezas que conforman el péndulo, se obtuvo que $m_2 = 0,0038261Kg$.

2.3. SISTEMA EN ESPACIO DE ESTADOS

Una vez obtenidos los parámetros físicos del sistema con los ensayos mencionados, se procede a programar las ecuaciones diferenciales físicas de la figura 2 en Matlab, a linealizar el sistema y a representar el sistema en espacio de estados, cuya matriz A y sus autovalores tiene el siguiente valor:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 82,73 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 132,15 & 0 \end{pmatrix} \quad eig(A) = \begin{pmatrix} 0 \\ 0 \\ 11,4957 \\ -11,4957 \end{pmatrix}$$

Se concluye que el sistema es de cuarto orden. La función de transferencia del sistema Péndulo de Furuta es de la siguiente forma:

$$P(s) = \frac{25,423}{(s - 11,5) \cdot (s + 11,5)}$$

Se puede observar que el sistema es inestable y que tiene menos polos que autovalores. Esto significa que hubo doble cancelación polo-cero para los autovalores iguales a 0. Además, se obtuvo que el rango de la matriz de controlabilidad es 4 (por lo que el sistema es totalmente controlable pudiendo así colocar todos los polos del sistema en cualquier lugar con un controlador) y que el rango de la matriz de observabilidad es 2 (por lo que el sistema no es totalmente observable pudiendo así colocar algunos polos del sistema en cualquier lugar con un observador).

Feedback control

3.1. LIMITACIONES

Antes de comenzar el desarrollo del controlador de una planta dada se debe determinar si la misma es controlable y observable. Dada la linealización de la planta en el punto de equilibrio estos 2 factores se despejan inmediatamente utilizando las herramientas provistas por Matlab.

Utilizando el comando $cntr(A,B)$ se obtiene la matriz de controlabilidad y obteniendo el rango de esta con el comando $rank(Cm)$ se extrae que dicha matriz es de orden completo, esto nos sugiere que la planta linealizada es controlable. En cuanto a la observabilidad de la planta se utiliza el comando $obsv(A,C)$ y obteniendo el rango de dicha matriz resultante se observa que esta no es de rango completo, en concreto mientras que las dimensiones de la matriz de observabilidad es 4 su rango es de 3, se concluye inmediatamente que el sistema es no observable. Este problema será tratado mas adelante.

En este punto resulta importante señalar que el sistema si resulta, en efecto, observable si se realimentan los 2 estados α y β como es el caso de la simulación; teniendo en este caso 2 elementos no nulos en la matriz C se obtiene que para tal caso la matriz de observabilidad si resulta en una de rango completo, sin embargo se trabajará de tal forma que el estado visto a la salida es el ángulo β ya que resulta mas coherente desde el punto de vista practico y, a su vez, es la variable mas critica.

3.2. DISEÑO DEL CONTROLADOR

Dada la naturaleza de la planta a controlar se tomó la decisión de implementar un controlador utilizando el método del Regulador Cuadrático Lineal, por sus siglas en ingles **LQR**. Esto debido a que resulta mucho mas intuitivo asignarle un determinado *peso* a los estados presentes del sistema de forma que el regulador controle de manera mas rigurosa algunos frente a otros colocando debidamente los polos realizando la realimentación, utilizando el método de realimentación de estados convencional surge la pregunta de cual es el lugar idóneo para dichos polos. Para el problema tratado aquí se consideró que las variables de estado mas criticas con el ángulo α y la velocidad angular de este $\dot{\alpha}$ haciendo referencia a los mostrados en la figura 1.1.

Si bien se a dicho que se utiliza el método de diseño por LQR se solicita que el controlador implementado posea acción integral, afortunadamente Matlab también ofrece una herramienta para la obtención de las ganancias de realimentación a través de dicho procedimiento el cual recibe el nombre de **LQI**. De esta forma se crea un nuevo estado el cual es el error a la salida del sistema. El esquema en bloques del controlador resultante se puede observar en la figura 3.1:

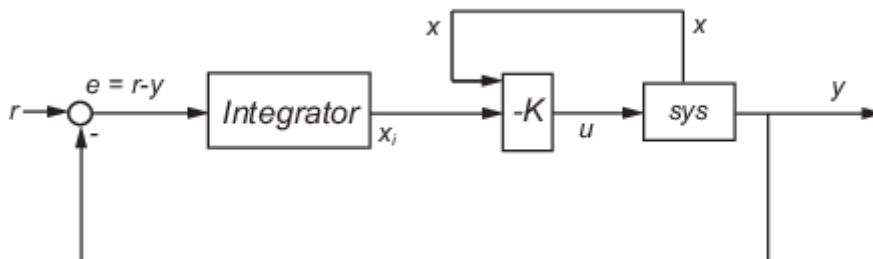


Figura 3.1: Diagrama en bloques con controlador LQI

3.3. DISEÑO DEL OBSERVADOR

Al momento de diseñar el observador se debe tener en consideración la limitación dada por la no observabilidad de la planta en caso de que el único estado que se pretende a la salida es el ángulo *beta*; para solventar este problema se implementará un observador de orden mínimo según los lineamientos expuestos en el libro Ingeniería de control Moderna del autor Ogata [2].

La situación que se plantea es que los ángulos α y θ son medibles (figura 1.1) mientras que las velocidades angulares $\dot{\alpha}$ y $\dot{\theta}$ no lo son, con lo cual deberán ser estimadas.

En este informe se omitirá el desarrollo teórico que sustenta el diseño del observador de orden reducido ya que estos están todos expuestos en la bibliografía con lo que se limitará a presentar los resultados fundamentales que hacen a la obtención de dicho observador, para este caso: observador de orden mínimo.

Como se ha dicho, se tienen variables medibles y variables que deberán ser estimadas, todas estas serán realimentadas con el controlador LQI a la entrada del sistema, el diagrama en bloques del sistema que se propone implementar se observa en la figura 3.2.

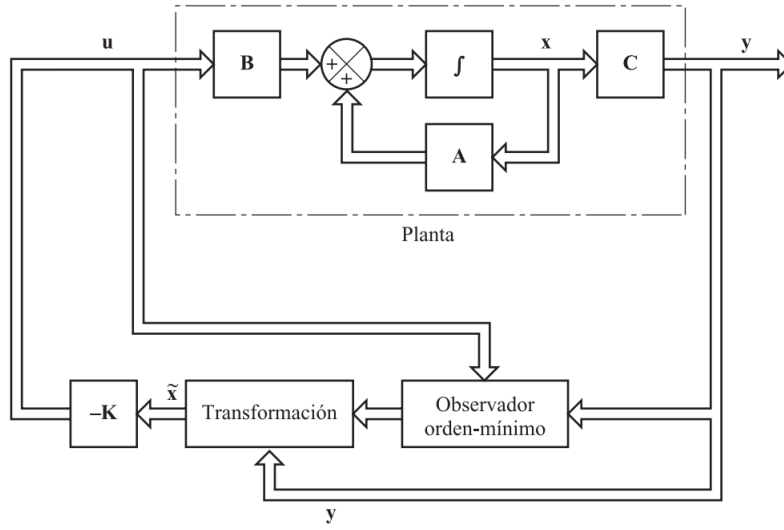


Figura 3.2: Sistema de control realimentado con estados observados con un observador de orden mínimo.

Siguiendo el procedimiento se particiona el vector de estados \mathbf{X} en 2 partes, una parte definida como \mathbf{X}_a la cual corresponde a los estados medibles a la salida y otras \mathbf{X}_b que corresponde a las no medibles. De esta forma el sistema de ecuaciones de estado y de salida particionado queda subdividido de la siguiente manera:

$$\begin{bmatrix} \dot{x}_a \\ \dot{x}_b \end{bmatrix} = \begin{bmatrix} A_{aa} & A_{ab} \\ A_{ba} & A_{bb} \end{bmatrix} \begin{bmatrix} x_a \\ x_b \end{bmatrix} + \begin{bmatrix} B_a \\ B_b \end{bmatrix} u$$

$$y = [1 \quad 0] \begin{bmatrix} x_a \\ x_b \end{bmatrix}$$

Realizando el desarrollo analítico expuesto en la bibliografía se deduce que el error en la estimación de los estados no observables viene dada por la siguiente expresión:

$$\dot{X}_b - \dot{\tilde{X}}_b = (A_{bb} - K_e \cdot A_{ab}) \cdot (X_b - \tilde{X}_b) \quad (3.1)$$

Donde \tilde{X}_b es la estimación de los estados no observador.

Se define el error de estimación como:

$$e = X_b - \tilde{X}_b \quad (3.2)$$

Con lo cual volviendo sobre la ecuación 3.1 se obtiene:

$$\dot{e} = (A_{bb} - K_e \cdot A_{ab}) \cdot e \quad (3.3)$$

Si se observa esta última ecuación se puede ver que obedece a la misma estructura a partir de la cual diseña un observador de estados de orden completo y convencional, en este caso K_e . Justamente este es el llamado observador de orden mínimo. Este será obtenible en caso que la nueva matriz de observabilidad definida a partir de A_{bb} y A_{ab} sea de orden completo. Para el problema planteado dicha condición se cumple con lo cual se tiene que el subsistema es observable. Por último, se debe colocar los polos de forma tal que el observador sea suficientemente mas rápido que el controlador, para eso si es conveniente utilizar el método de ubicación manual de polos con la línea de código $K_e = \text{place}(A_{bb}', A_{ab}', L)'$ asegurándose que los polos contenidos en el vector L hagan a un funcionamiento mas veloz del observador respecto al controlador.

Una vez obtenido K_e se procede con la implementación del sistema siguiendo el esquema expuesto en la bibliografía en la figura 3.3 en la cual se definen los parámetros que en esta se observan.

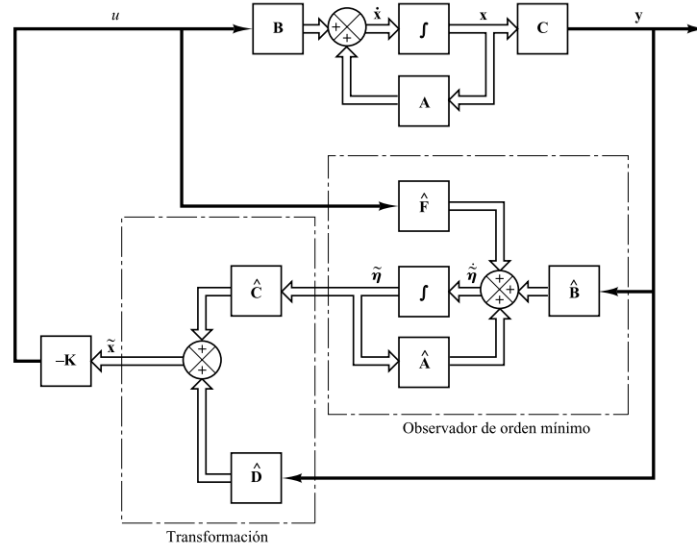


Figura 3.3: Sistema con realimentación de estado observado donde el observador es el observador de orden mínimo.

3.4. DISCRETIZACIÓN

Al momento de discretizar el sistema se debe pasar del modelo continuo hasta ahora utilizado a un modelo discreto, para esto se debe elegir un método de conversión así como un periodo de sampling lo suficientemente chico para cumplir con el criterio de Nyquist, en cuanto al método de conversión se decide utilizar el método de *Zero Orden Hold* dada su practicidad y en cuanto a la frecuencia de muestreo se deben observar los polos y seleccionar con criterio una frecuencia adecuada.

Una vez ya discretizado el sistema se procede de la misma forma que lo realizado para el sistema en tiempo continuo y tomando los recaudos necesarios al momento de realizar la simulación.

3.5. IMPLEMENTACIÓN EN MICRO-CONTROLADOR

Para la implementación en un micro-procesador se utiliza la función transferencia del controlador con observador de orden mínimo desde la entrada, esta se determina siguiendo los pasos vistos en la bibliografía y utilizando las matrices halladas hasta aquí.

Una vez obtenida la función transferencia (en tiempo continuo) del controlador con observador y realizando las simplificaciones necesaria de forma que la implementación no sea en exceso exigente para el micro-controlador se despeja el espacio de estados del controlador de forma que discretizando este se obtienen las matrices (A,B,C,D) las cuales serán programadas en el micro y este finalmente controla el péndulo invertido.

3.6. SIMULACIÓN Y RESULTADOS

En la sección 4.3 se muestra globalmente en que consiste la simulación realizada la cual puede visualizarse en la figura 4.9, con lo cual en este sección se limita a presentar únicamente los controladores finales implementados utilizando el método de realimentación de estados como se ha visto hasta aquí.

3.6.1. TIEMPO CONTINUO

El diagrama en bloques del controlador en tiempo continuo se muestra en la figura 3.4 como se puede ver esta misma sigue la estructura que se presento gráficamente en el diagrama de la figura 3.3.

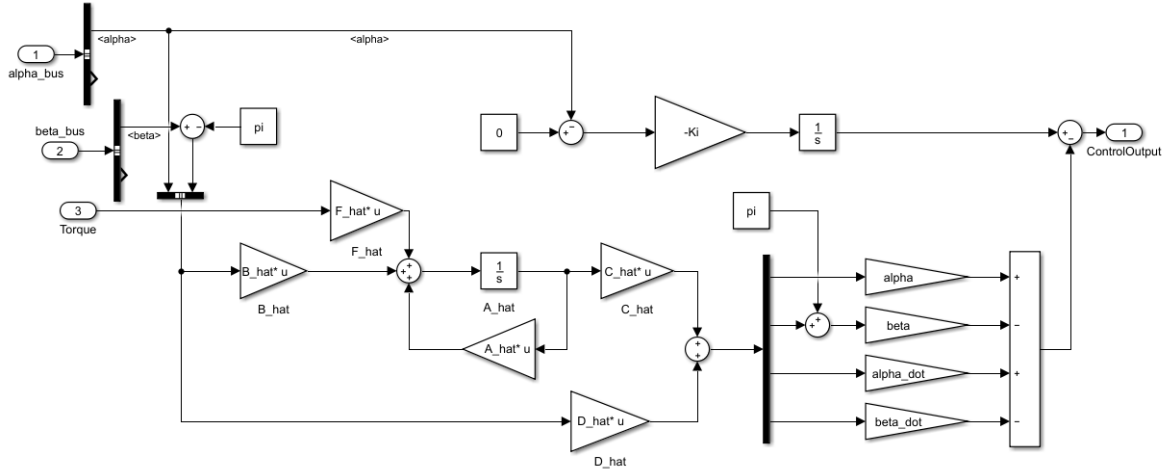


Figura 3.4: Diagrama en bloques del controlador junto con el observador de orden mínimo.

La evolución de los distintos estados para este controlador en tiempo continuo se muestra en la figura 3.5

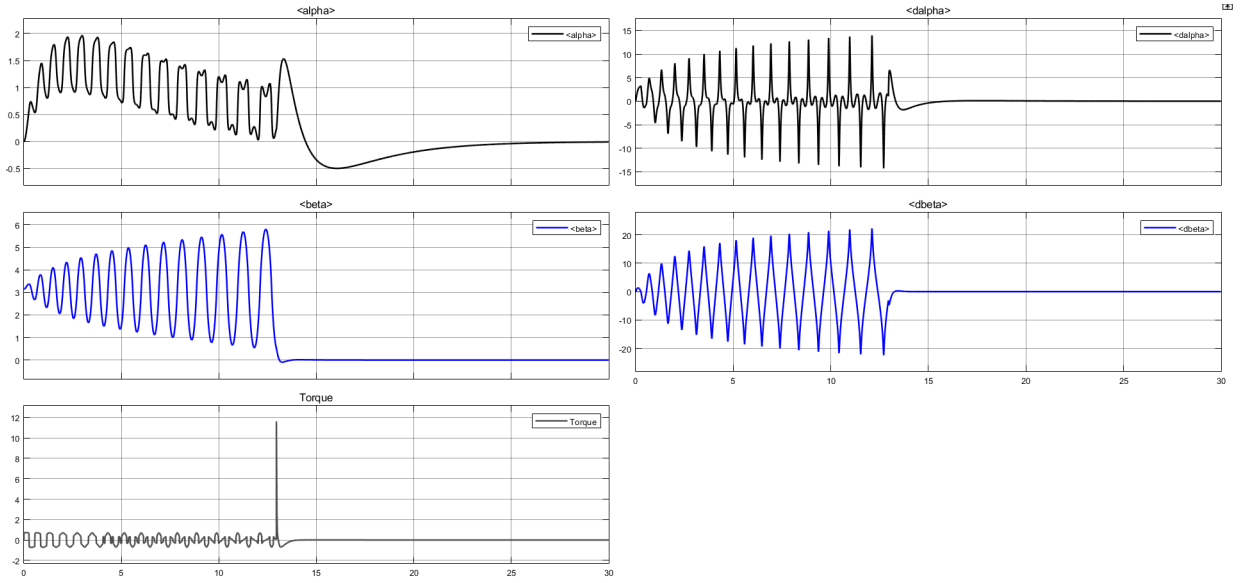


Figura 3.5: Resultados obtenidos en tiempo continuo.

Se puede observar aquí la actuación del controlador de Swing Up, a un ángulo β determinado actúa el controlador de Stay Up desarrollado. A su vez se puede apreciar a través de este mismo estado β como se consigue el que péndulo se mantenga en una posición vertical hacia arriba.

3.6.2. TIEMPO DISCRETO

La simulación en tiempo discreto sigue la estructura presentada para la simulación en tiempo continuo, para este caso es imperativo definir un periodo de sampling óptimo de forma que el control pueda ser correctamente realizado, dada la distribución final de los polos utilizando el regulador LQI se decide como periodo de muestreo óptimo el aplicar 2.5ms. De esta forma se respeta el criterio de Nyquist y control puede ser realizado sin sobreexigir el sistema.

El diagrama del controlador en tiempo discreto y los resultados obtenidos se muestran en las figuras 3.6 y 3.7 respectivamente.

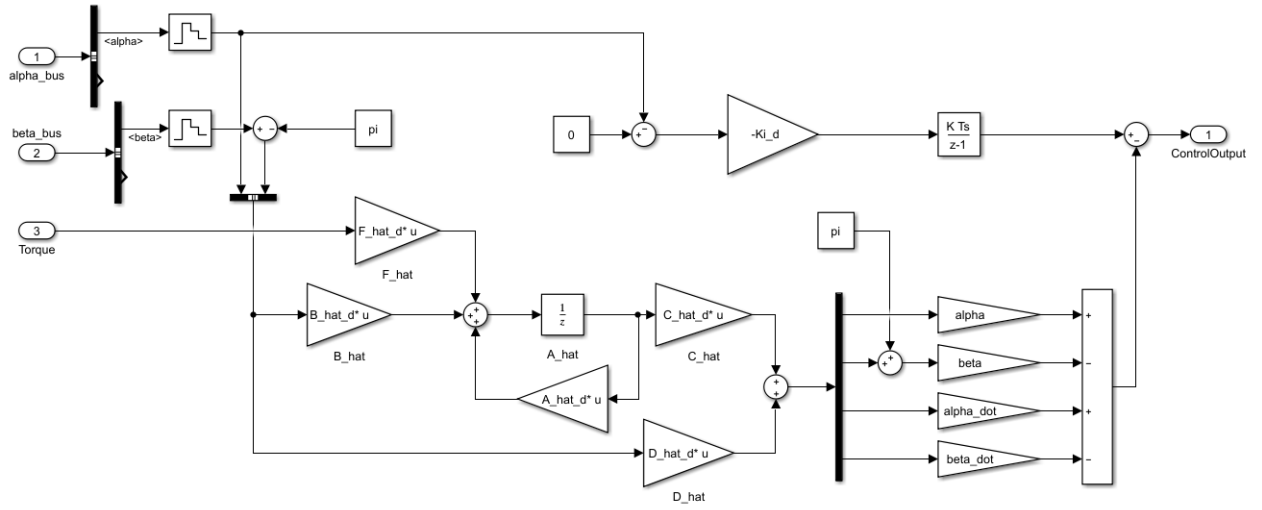


Figura 3.6: Controlador junto con el observador de orden mínimo en tiempo discreto.

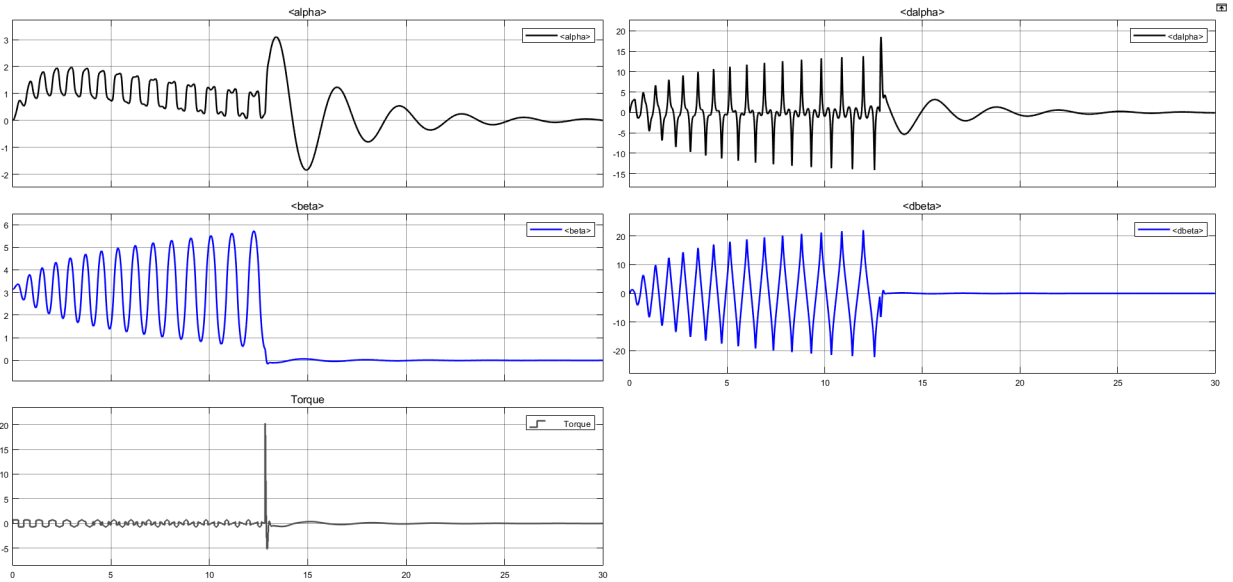


Figura 3.7: Resultados obtenidos en tiempo discreto.

3.6.3. SIMULACIÓN EN MICRO-CONTROLADOR

Por último en lo que respecta la implementación en un microprocesador se intentó realizar la simulación de este de la forma mas fiel posible a la realidad, esto es que las señales *alpha* y *beta* sean los inputs del modulo que representa el micro-controlador y que este realice el procesamiento necesario para arrojar el toque requerido a su salida, volviendo sobre lo comentado en la sección 3.5 esto mismo es la transferencia de la salida hacia la entrada.

La representación en espacio de estado de dicha transferencia es presentada simbólicamente por la bibliográfica con lo cual se pueden obtener así las matrices A, B, C y D las cuales se pueden observar en la figura 3.8.

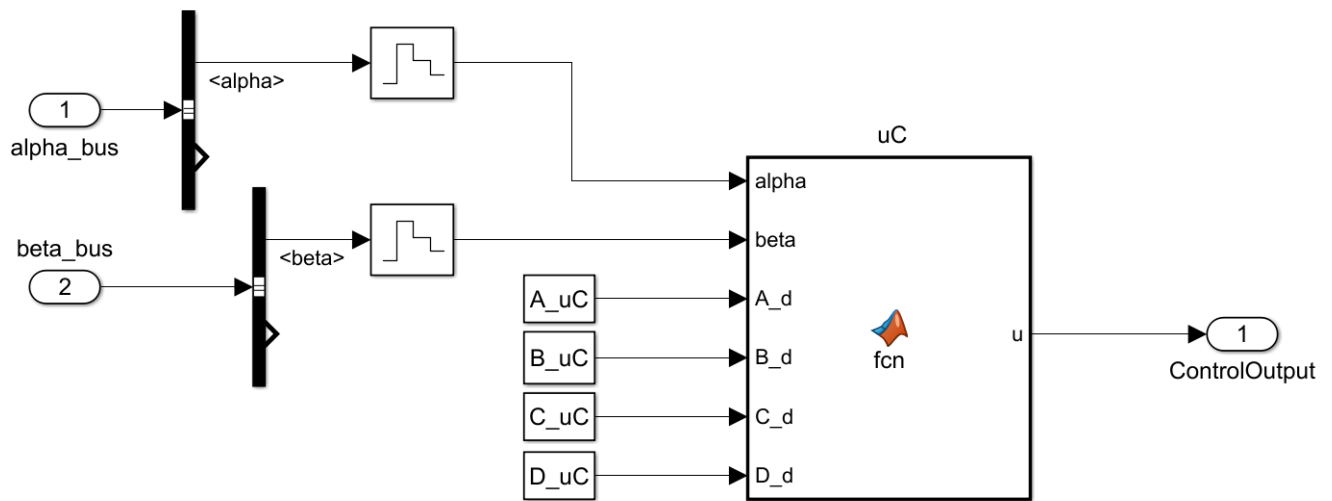


Figura 3.8: Representación del microcontrolador.

Operando con estas matrices de manera ya conocida se obtiene la salida deseada. Nuevamente los resultados se visualizan en la figura 3.9.

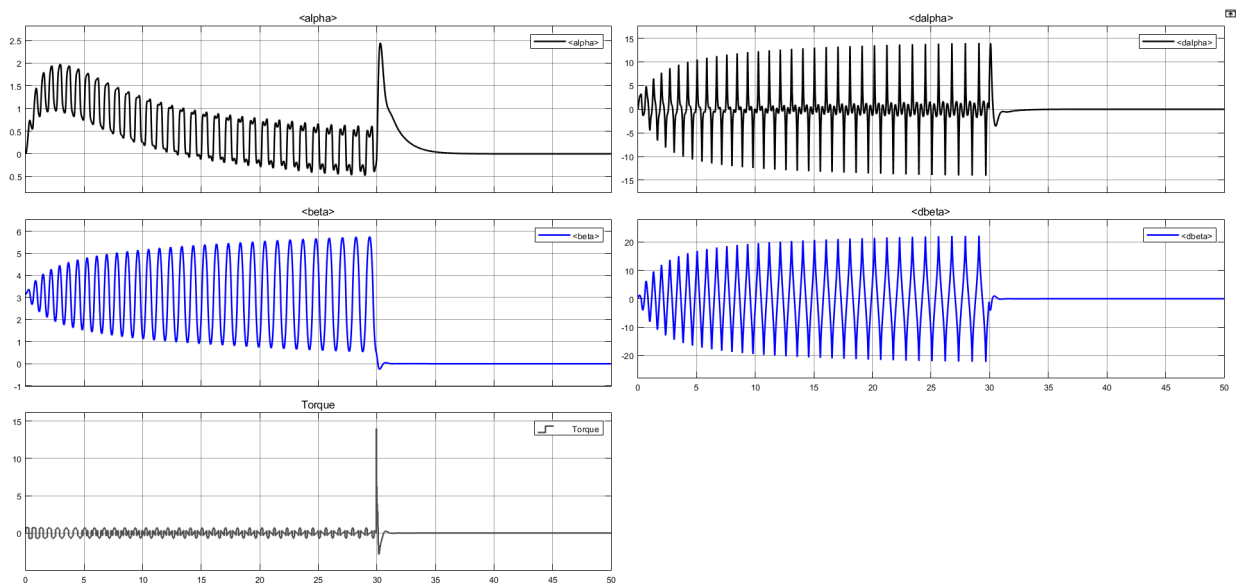


Figura 3.9: Resultados obtenidos en tiempo discreto.

Loopshaping

La topología del control a diseñar por Loop Shaping contiene 2 lazos de realimentación, con los cuales se realimenta el ángulo del brazo rotatorio α (lazo primario) y el ángulo del péndulo β (lazo secundario). Ya que habrán 2 lazos, también habrán 2 controladores (uno por cada lazo de realimentación) como se puede observar en la figura:

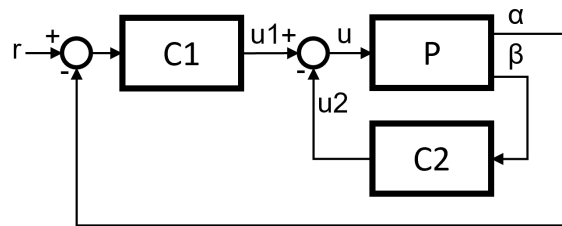


Figura 4.1: Diagrama de control por loop shaping con 2 realimentaciones.

Teniendo a mano la función transferencia del sistema Péndulo de Furuta, estamos en condiciones de diseñar ambos controladores de modo tal que el sistema se estabilice en el punto de equilibrio mencionado anteriormente.

4.1. LAZO SECUNDARIO

Se empieza diseñando el contronador C_2 del lazo secundario. Por ende, se asume $u_1 = 0$ quedando el siguiente esquema:

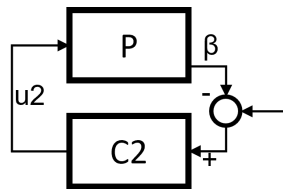


Figura 4.2: Diagrama de control por loop shaping (lazo secundario).

El objetivo es tratar de estabilizar el sistema lo mejor posible con el controlador C_2 para luego diseñar el controlador C_1 . Debido a que, en el lazo secundario, se quiere controlar el ángulo del péndulo (el cual es el estado más inestable) se requerirá un controlador lo suficientemente rápido. Por lo tanto, se diseña un controlador PD para el lazo secundario.

$$C_2 = \frac{150 \cdot (s + 11,5)}{(s + 100)}$$

De esta forma, se cancela el único polo estable del sistema y se agrega un polo lejos para que el controlador sea bipropio. Los diagramas de Root Locus y de Bode del conjunto $L_2 = C_2 \cdot P$ quedan se presentan a continuación:

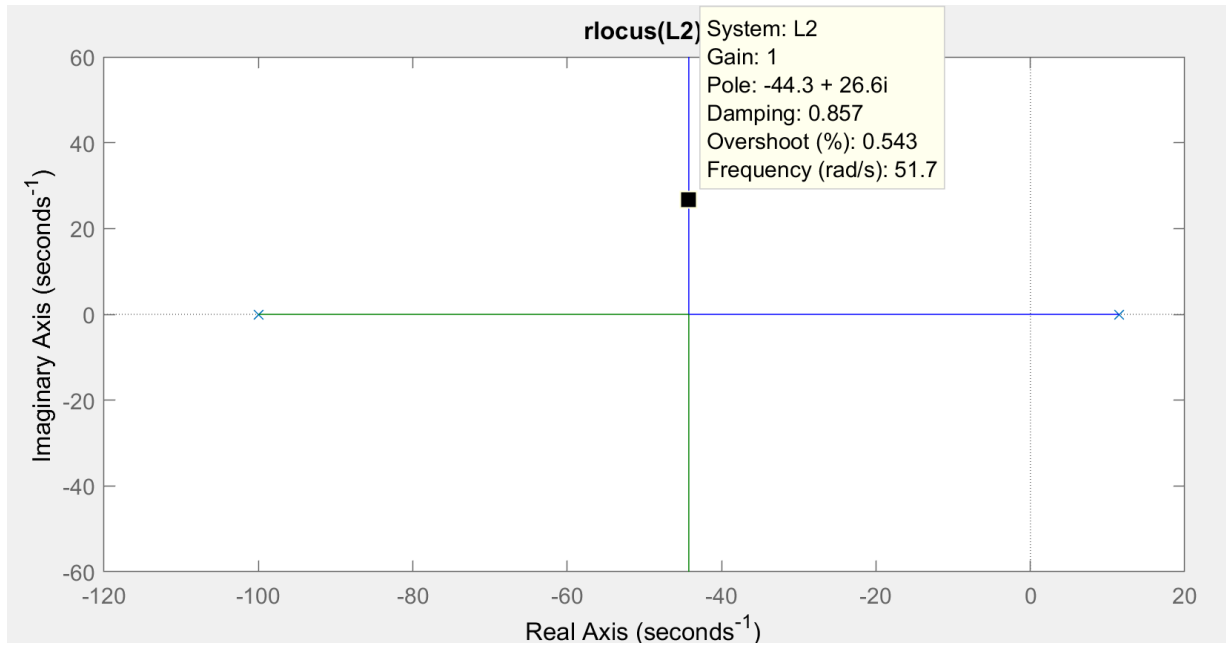


Figura 4.3: Diagrama de Root Locus (lazo secundario).

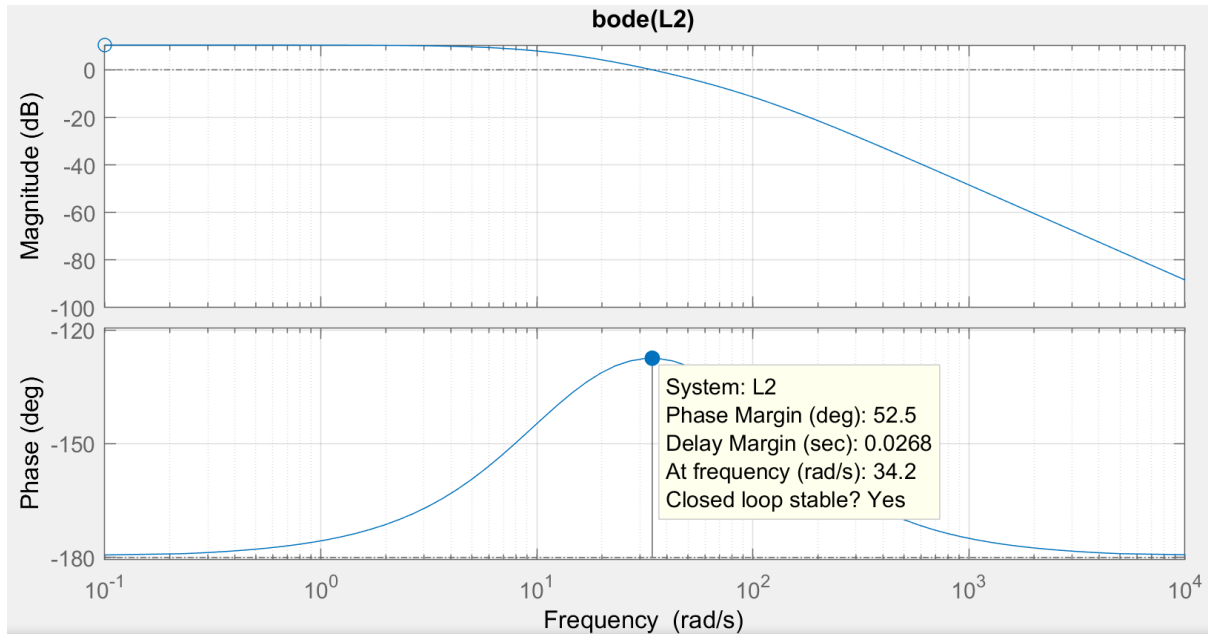


Figura 4.4: Diagrama de Bode (lazo secundario).

Se puede ver que se logró estabilizar el sistema con el lazo secundario. Por lo tanto, estamos en condiciones para diseñar el controlador C_1 del lazo primario.

4.2. LAZO PRIMARIO

Para el diseño del controlador C_1 del lazo primario, primero se tiene que encontrar la representación en espacio de estado del conjunto C_2 y P de tal forma que la salida sea el ángulo del brazo rotatorio α . Tras un desarrollo matemático, se obtuvo que:

$$\begin{pmatrix} \dot{x}_P \\ \dot{x}_{C_2} \end{pmatrix} = \begin{pmatrix} A_P - B_P D_{C_2} C_P & -B_P C_{C_2} \\ B_{C_2} C_P & A_{C_2} \end{pmatrix} \cdot \begin{pmatrix} x_P \\ x_{C_2} \end{pmatrix} + \begin{pmatrix} B_P \\ 0 \end{pmatrix} \cdot u_1$$

$$\alpha = (C'_P \ 0) \cdot \begin{pmatrix} x_P \\ x_{C_2} \end{pmatrix} \quad C'_P = (1 \ 0 \ 0 \ 0)$$

Ahora sí nos encontramos en condiciones para diseñar el controlador C_1 sabiendo que el esquema es el siguiente:

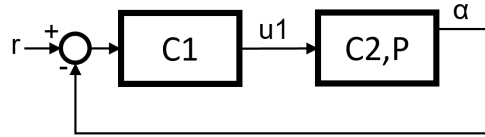


Figura 4.5: Diagrama de control por loop shaping (lazo primario).

En este caso, se quiere controlar el ángulo del brazo rotativo α . Dado que se podría requerir que el error en estado estacionario sea nulo, entonces se diseñará un controlador PID para el lazo primario.

$$C_2 = \frac{-0,038 \cdot (s + 11,5) \cdot (s + 0,1)^2 \cdot (s^2 + 88,5s + 3,812 \cdot 10^6)}{s \cdot (s + 100)^3 \cdot (s + 8,71)}$$

Cabe mencionar que se han agregado polos en -100 para que el controlador sea bipropio y que se ha usado ganancia negativa para invertir el diagrama de Root Locus y, así, sea más fácil estabilizar el sistema. Los diagramas de Root Locus, de Bode del sistema completo (con las 2 realimentaciones) se presentan a continuación:

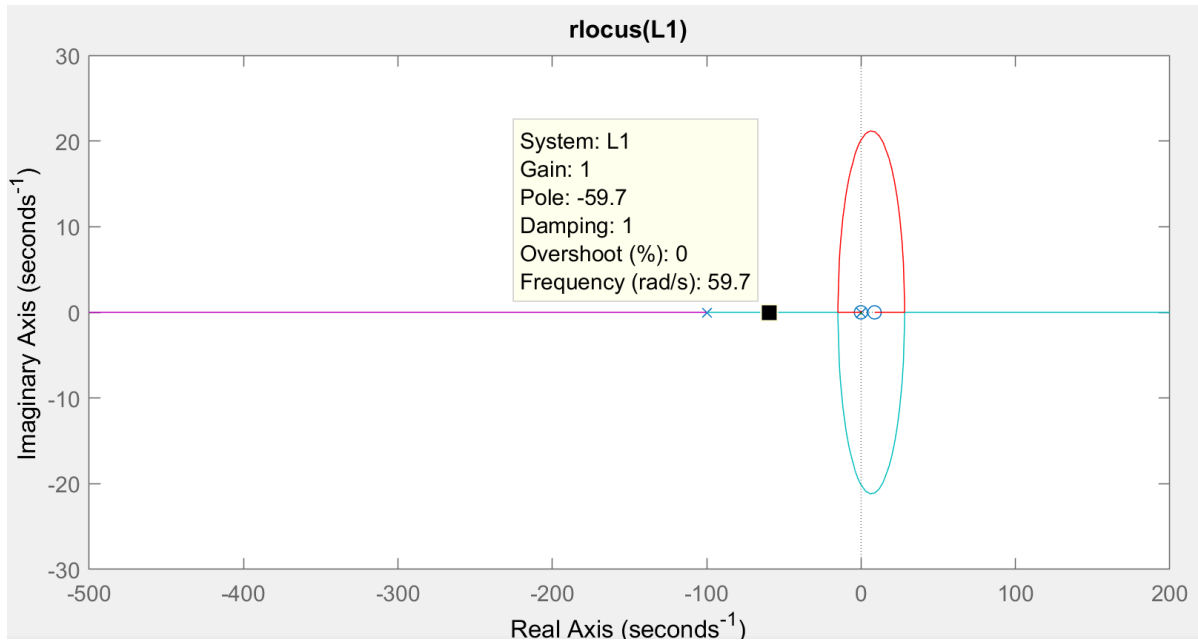


Figura 4.6: Diagrama de Root Locus del sistema completo.

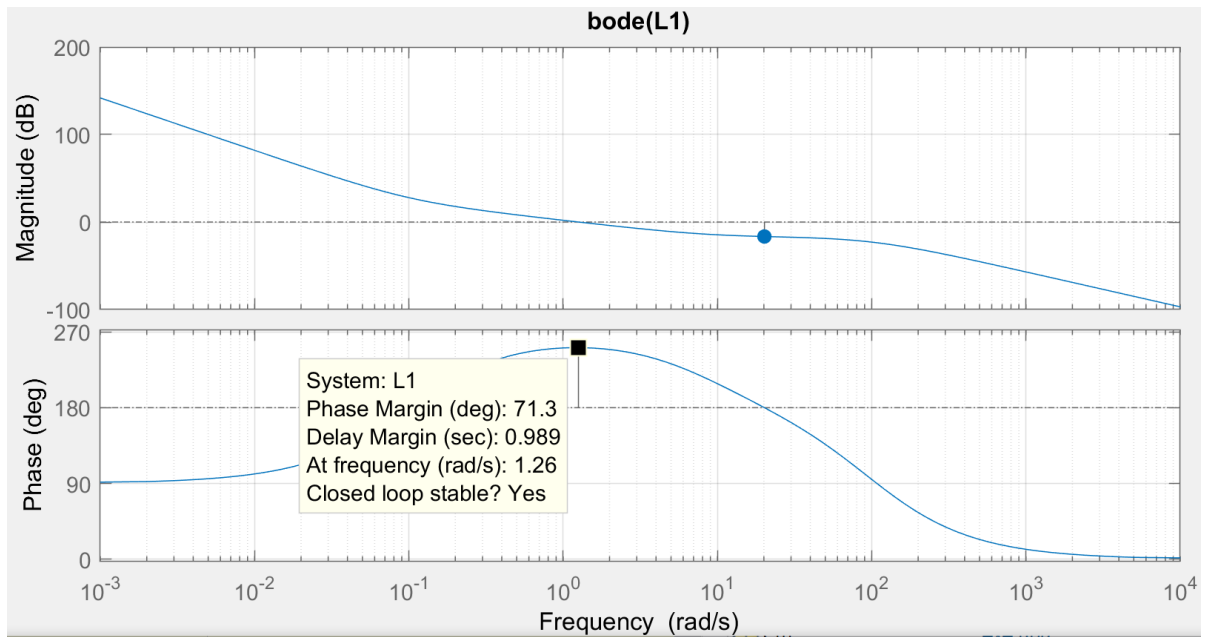


Figura 4.7: Diagrama de Bode del sistema completo.

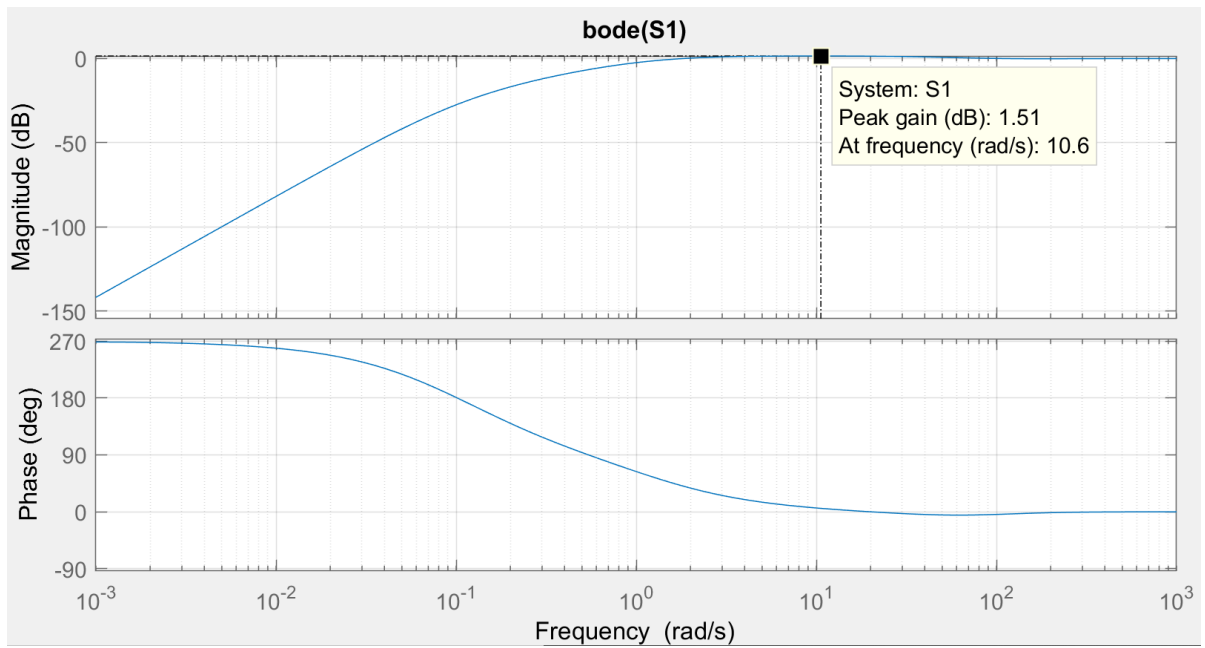


Figura 4.8: Diagrama de Bode de la sensibilidad del sistema completo.

A través de la figura del diagrama de Bode de la sensibilidad del sistema completo se extrae que el margen de estabilidad es cercano a 1. Por lo tanto, el sistema es lo suficientemente robusto.

$$S_m = 10^{\frac{-1.51}{20}} = 0.84$$

4.3. SIMULACIÓN Y RESULTADOS

Para poder realizar las simulaciones con el control diseñado, se utilizó el modelo virtual del sistema Péndulo de Furuta armado en Simscape de Matlab propuesto por la cátedra. El diagrama en bloques que conforman la simulación se muestra en la siguiente figura:

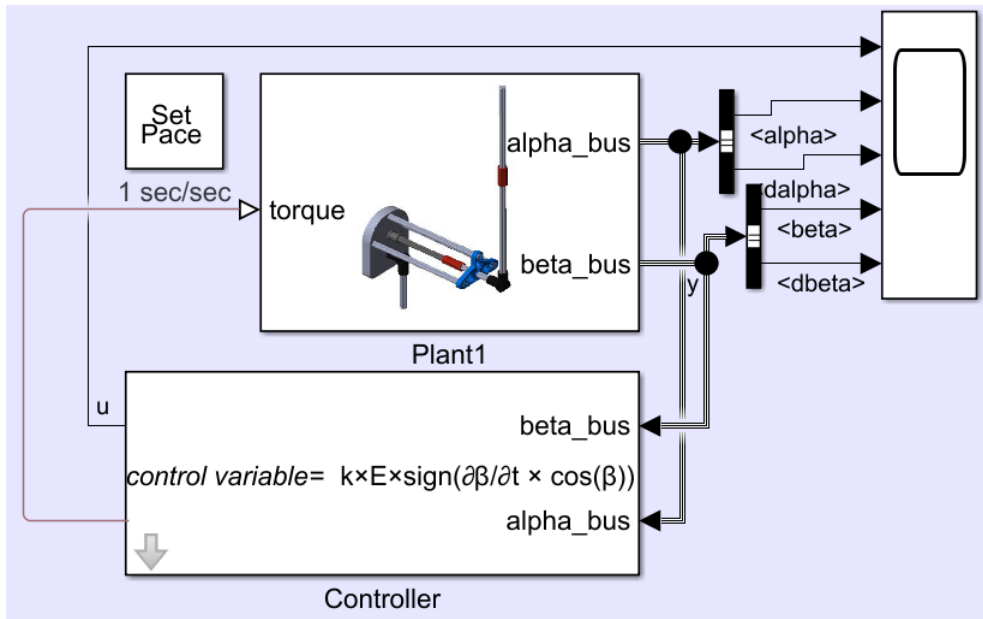


Figura 4.9: Diagrama en bloques del sistema completo en Simulink de Matlab.

Donde el bloque *Plant1* conforma el modelo virtual del sistema en Simscape y el bloque *Controller* contiene 2 tipos de controladores. El primer controlador realiza el auto swing up y el segundo controlador sirve para estabilizar el sistema en el punto de equilibrio (el cual fue diseñado previamente).

4.3.1. TIEMPO CONTINUO

A continuación se muestra el diagrama en boques del controlador en tiempo continuo. Cabe mencionar que dicho diagrama se encuentra dentro del bloque *Controller*.

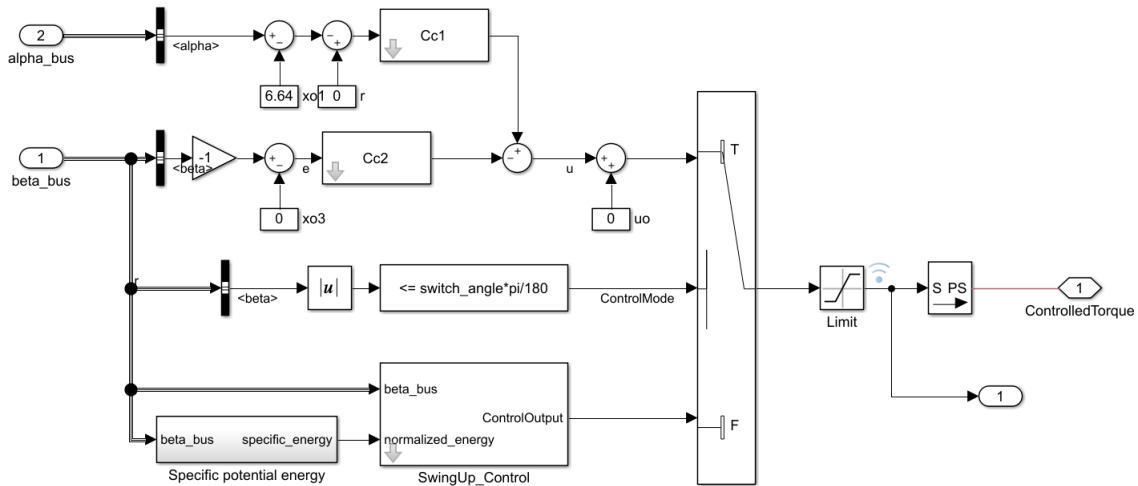


Figura 4.10: Diagrama en bloques del controlador del sistema completo en tiempo continuo.

Tras simular el diagrama en bloques, se obtuvo la siguiente respuesta:

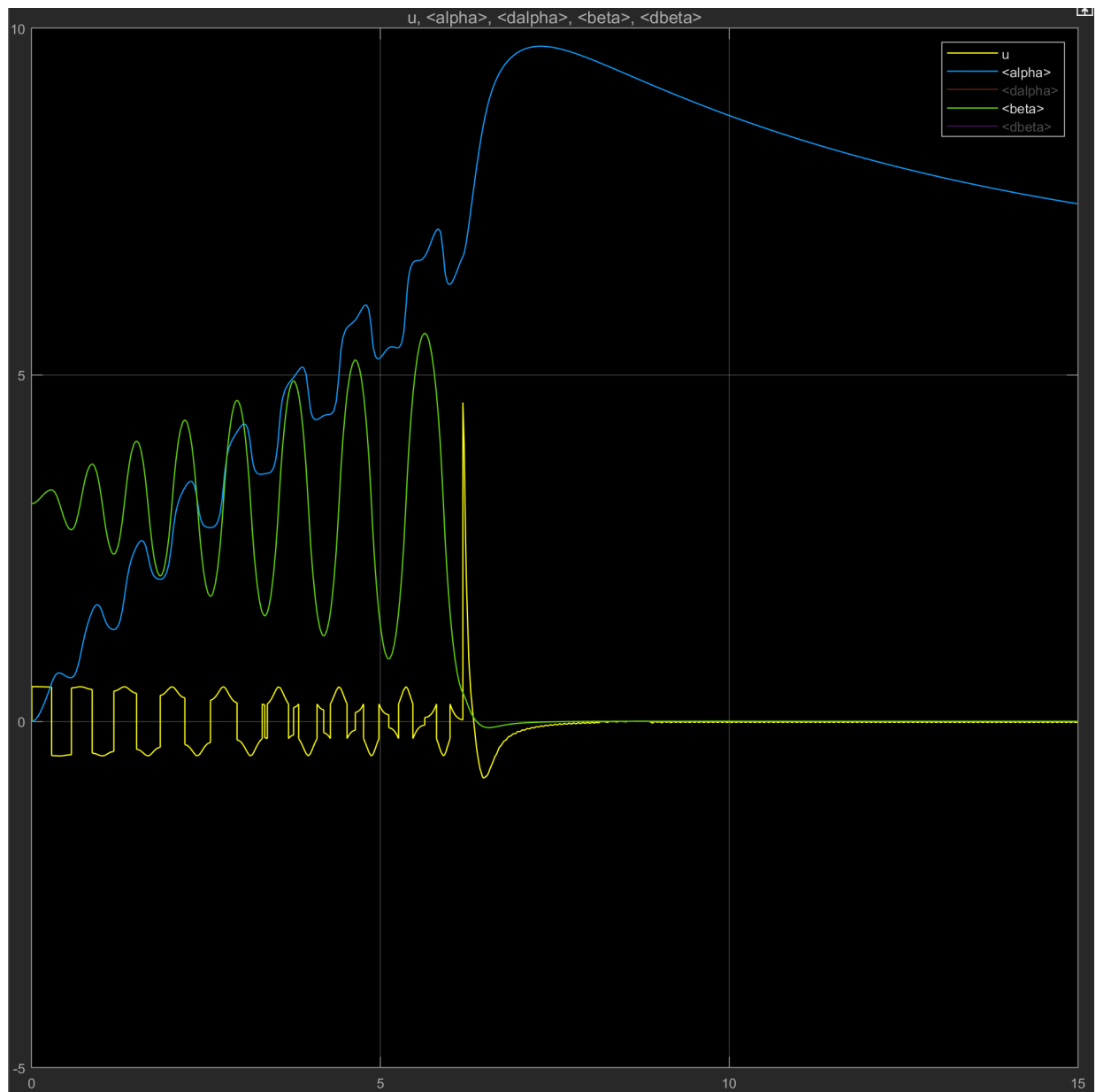


Figura 4.11: Respuesta del sistema con el controlador por Loop Shaping en tiempo continuo.

Las oscilaciones que se muestran corresponden al auto swing up del péndulo y, como se puede observar, una vez alcanzado un rango cercano al punto de equilibrio, el controlador por Loop Shaping empieza a actuar estabilizando el mismo.

4.3.2. TIEMPO DISCRETO

A continuación se muestra el diagrama en bloques del controlador en tiempo discreto. Cabe mencionar que dicho diagrama se encuentra dentro del bloque *Controller*.

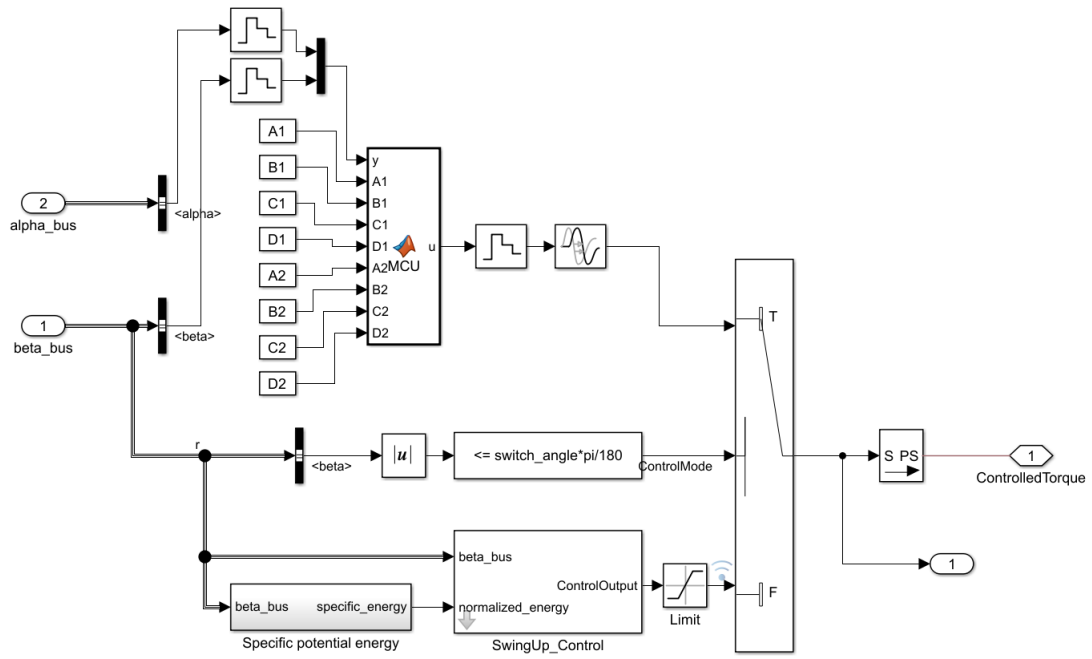


Figura 4.12: Diagrama en bloques del controlador del sistema completo en tiempo discreto.

Tras simular el diagrama en bloques con un período de muestreo $T_s = 20ms$, se obtuvo la siguiente respuesta:

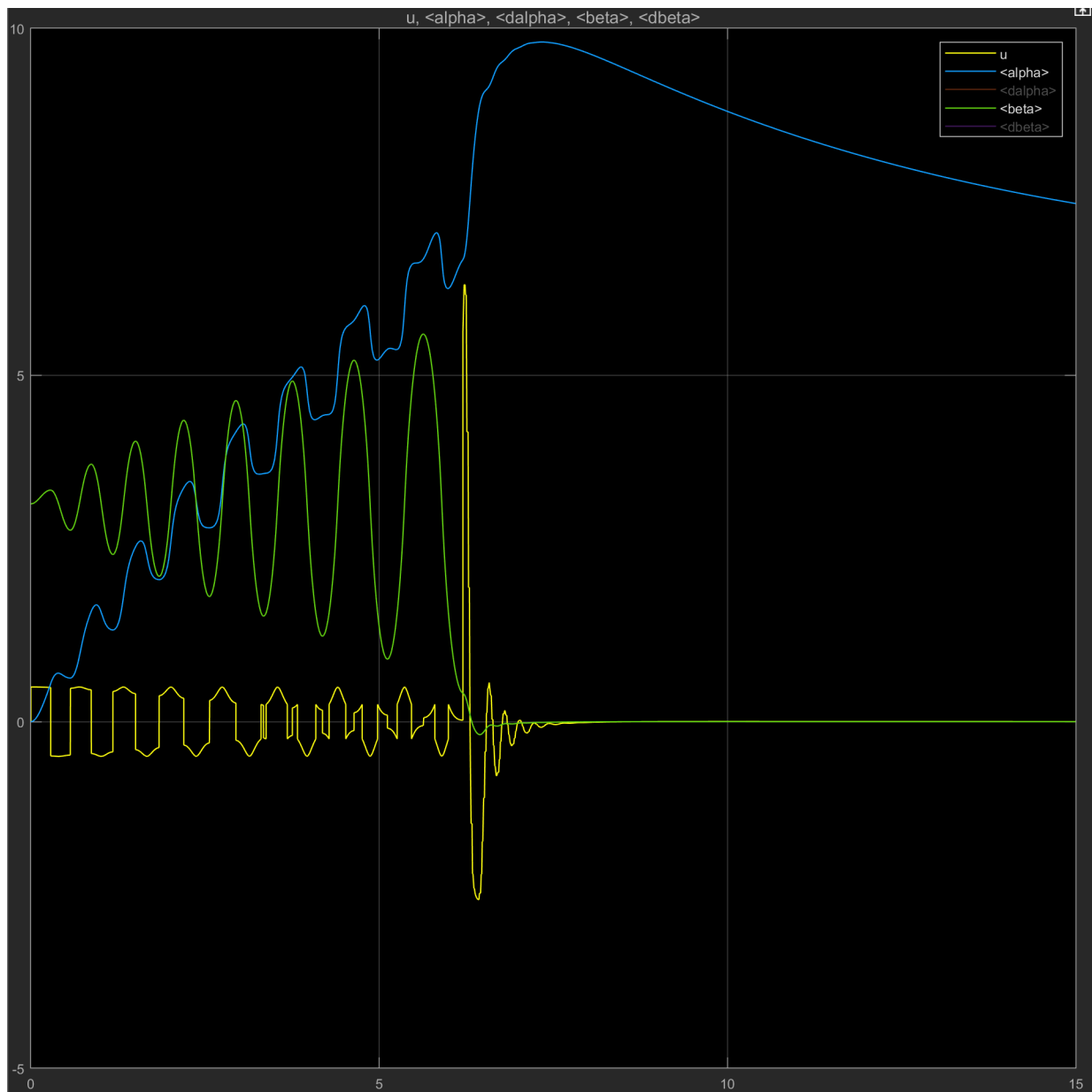


Figura 4.13: Respuesta del sistema con el controlador por Loop Shaping en tiempo discreto.

Nuevamente, las oscilaciones que se muestran corresponden al auto swing up del péndulo y, como se puede observar, una vez alcanzado un rango cercano al punto de equilibrio, el controlador por Loop Shaping empieza a actuar estabilizando el mismo.

Conclusiones

En el presente trabajo se han puesto en práctica muchos de los conocimientos adquiridos en el curso, recorriendo así de forma completa el proceso de caracterización y control de una planta que, en este caso, ha sido el péndulo invertido de Furuta.

Se ha linealizado el comportamiento de la planta en un punto de equilibrio que, en este caso, es aquél con el ángulo $\beta = 0$ (medido desde el eje imaginario vertical en sentido hacia arriba) y a su vez con velocidad angular nula. Para este proceso de linealización se han aplicado las técnicas vistas en el curso así como también se ha recurrido a los papers [1] y [4] que se muestran en la sección de bibliografía y se realizaron los ensayos correspondientes en la simulación para determinar los parámetros de la planta.

Una vez hecha la linealización, se realizó al desarrollo de los controladores. Este punto resultó muy enriquecedor ya que se han implementado distintos controladores a través de diferentes metodologías también vistas en el curso. A su vez, estas, se han debido de apoyar en determinadas técnicas que hicieron posible la aplicación de dicho control. Entre estas, se destaca el proceso de diseño de un observador de orden mínimo según lo expuesto en el libro [2], así como también el control por loop shaping con 2 realimentaciones expuesto en el libro [3] que se muestran en la bibliografía. Esto ha sido de suma importancia para la realización del observador y con ello el funcionamiento del regulador.

Por último realizando la discretización de los controladores se llega finalmente a una posible aplicación en un micro-controlador. Queda a futuro esta implementación en un entorno físico que exceda de lo alcanzado por las simulaciones realizadas.

Bibliografía

- [1] BENJAMIN SETH CAZZOLATO, *On the Dynamics of the Furuta Pendulum*.
- [2] KATSUHIKO OGATA, *Ingeniería de control moderna*, quinta edición, 2010.
- [3] SALGADO - GOODWIN - GRAEBE, *Control System Design*.
- [4] OSCAR DANILO MONTOYA GIRALDO, JUAN GUILLERMO VALENZUELA HERNÁNDEZ, DIDIER GIRALDO BUITRAGO, *Control global del Péndulo Rotacional Invertido empleando modelos de energía*, Universidad Tecnológica de Pereira, 2012.