

# Informe 1 Control avanzado: Péndulo de Furuta

1<sup>st</sup>Damian Bravo  
Ing. Civil Mecatronica  
Universidad de Talca

Talca, Chile  
dbravo18@alumnos.utalca.cl

2<sup>nd</sup>Joaquín Gutierrez  
Ing. Civil Mecatronica  
Universidad de Talca

Chepica, Chile  
jgutierrez18@alumnos.utalca.cl

**Abstract**—Este informe aborda la realización del primer la primera actividad del ramo Control Avanzado, la cual consiste de simular un sistema no lineal, linealizar el sistema, determinar un punto de operación estable y otro inestable, y finalmente comparar el sistema lineal con el no lineal dando a conocer como se comporta el sistema en cuanto se aleja del punto de operación elegido. Para este caso, el sistema se modela mediante s-funtion lv 2 en MATLAB, mientras que el sistema elegido es un tipo de péndulo invertido conocido como Péndulo de Furuta.

## I. INTRODUCCIÓN

El control de sistemas dinámicos representa uno de los pilares fundamentales en la ingeniería contemporánea, permitiendo la manipulación precisa y eficiente del comportamiento de sistemas para alcanzar objetivos específicos. Dentro de este vasto campo, el estudio del péndulo invertido ha capturado la atención de investigadores y estudiantes debido a su complejidad inherente y su amplia gama de aplicaciones en áreas como la robótica, el control de vehículos y la estabilización de sistemas.

En este contexto, el "Péndulo de Furuta" emerge como un caso particularmente interesante y desafiante. El "Péndulo de Furuta" es un sistema compuesto por un brazo que puede girar horizontalmente gracias a un motor y que está conectado en uno de sus extremos a un péndulo que se mueve libremente en el plano vertical. Esta configuración única lo convierte en un objeto de estudio relevante en el campo del control de sistemas no lineales, ya que presenta desafíos adicionales en términos de modelado, análisis y diseño de controladores.

El presente informe se enfoca en la simulación y análisis del "Péndulo de Furuta" como parte de la primera actividad del ramo de Control Avanzado. A través de la utilización de herramientas avanzadas como MATLAB y técnicas de modelado precisas, se busca comprender a fondo el comportamiento del "Péndulo de Furuta" y explorar aspectos importantes para su posterior control, como son las ecuaciones que describen al sistema, la similitud entre el sistema linealizado y el sistema no linealizado a medida que el sistema se aleja del punto de operación, así mismo, se explora la estabilidad del sistema, así como su controlabilidad y observabilidad.

Durante el desarrollo de este informe, se describirá en detalle el proceso de modelado del "Péndulo de Furuta", dando a conocer las ecuaciones de estado del mismo, el método

de linealización utilizado, así como la utilización s-funtion y simulink en MATLAB, se presentará la metodología empleada para la simulación y análisis del sistema, y se discutirán los resultados obtenidos.

## II. MARCO TEÓRICO

### A. Espacio de estados

El estado de un sistema dinámico se define como el conjunto mínimo de variables, conocidas como variables de estado, que, junto con la entrada para un tiempo inicial  $t_0$ , determinan completamente el comportamiento futuro del sistema. Este concepto es fundamental en el análisis y diseño de sistemas dinámicos y se aplica no solo a sistemas físicos, sino también a sistemas biológicos, económicos, sociales y otros.

Las variables de estado son esenciales para describir completamente el comportamiento de un sistema dinámico. Si se necesitan al menos  $n$  variables  $x_1, x_2, \dots, x_n$  para describir completamente el comportamiento del sistema, donde  $n$  es la dimensión del espacio de estados, entonces estas variables se consideran un conjunto de variables de estado.

Es importante destacar que las variables de estado no necesariamente deben ser cantidades físicamente medibles u observables. Se pueden seleccionar variables que representen estados abstractos o aspectos internos del sistema. Sin embargo, en la práctica, suele ser conveniente seleccionar variables de estado que sean físicamente medibles, ya que esto simplifica la implementación de estrategias de control, esto está conectado con el concepto de controlabilidad y observabilidad.

El estado del sistema en un instante de tiempo  $t$  después del tiempo inicial  $t_0$  se representa mediante un vector  $\mathbf{x}(t)$ . Este vector, conocido como vector de estado, contiene las  $n$  variables de estado del sistema en ese momento específico.

El espacio de estados es un espacio  $n$ -dimensional cuyos ejes están formados por las variables de estado  $x_1, x_2, \dots, x_n$ . Cada punto en este espacio representa un estado del sistema en un instante de tiempo dado. El análisis en el espacio de estados proporciona una representación geométrica intuitiva del comportamiento dinámico del sistema y es fundamental en el diseño de controladores y sistemas de control.

En muchos casos, las ecuaciones de estado de un sistema dinámico se pueden expresar en forma matricial, siempre y cuando sean lineales, de la forma:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

donde:  $\mathbf{x}$  es el vector de estado del sistema,  $\mathbf{u}$  es el vector de entrada o control,  $\mathbf{A}$  es la matriz del sistema,  $\mathbf{B}$  es la matriz de entrada o control.

Estas ecuaciones describen la evolución temporal del sistema en términos de sus variables de estado y la entrada o control aplicado. El análisis y diseño de sistemas dinámicos suelen involucrar la selección adecuada de las matrices  $\mathbf{A}$  y  $\mathbf{B}$  para lograr ciertos objetivos de control y rendimiento.

### B. Controlabilidad

En el libro "Ingeniería de Control moderna" [1] se explica que un sistema es controlable en el tiempo  $t_0$ , si se puede transferir desde **un estado inicial**  $X_0(t_0)$  **a cualquier otro estado final**, mediante un vector de control sin restricciones, en un intervalo de tiempo finito, por otro lado, dicho sistema sera completamente controlable **si todos los estados del sistema cumplen dicha condición**.

La matriz de controlabilidad se puede encontrar de varias formas distintas, estas formas están explicadas en el capitulo 9-6, del libro [1], sin embargo, en el caso de este trabajo se utilizaran las herramientas disponibles en MATLAB, usando la linea de código presente en 1

Listing 1. Funcion Matriz Controlabilidad

```
ctrb(A,B);
```

Donde ,  $\mathbf{A}$  es una matriz  $n \times n$  y  $\mathbf{B}$  una matriz  $n \times 1$  , donde  $n$  es el numero de estados, parte de la representación canónica de un sistema en espacio de estados lineal, de la forma:

$$\dot{\mathbf{x}} = \mathbf{A} * \mathbf{x} + \mathbf{B} * \mathbf{U}$$

De esta manera, para saber si un sistema es controlable mediante este método, hay que ver que el rango de la matriz de controlabilidad, si el rango es igual al numero de variables de estado del sistema ,entonces el sistema es completamente controlable, mientras que si esta entre 1 y  $n$  entonces el sistema tiene el mismo numero de estados controlables, mientras que si el valor es 0, el sistema no es controlable.

### C. Observabilidad

La observabilidad se refiere a la capacidad de determinar los estados de un sistema dinámico a partir de sus salidas observadas. Un sistema dinámico sera completamente observable, cuando se puedan determinar todas las variable de estado del sistema a partir de las salidas del mismo. Por el contrario, si hay variables de estado que no pueden ser observadas a partir de las salidas, el sistema se considera parcialmente observable o no observable.

La observabilidad de un sistema se puede analizar utilizando la matriz de observabilidad, que es una herramienta importante en el diseño de sistemas de control. Si el rango de la matriz de

observabilidad es igual a la dimensión del espacio de estados  $n$ , entonces el sistema es completamente observable. Por otro lado, si el rango de la matriz de observabilidad es menor que  $n$ , entonces el sistema es parcialmente observable y algunas variables de estado pueden no ser observables a partir de las salidas.

En el caso de este trabajo se usan las herramientas presentes en MATLAB, de tal manera que la observabilidad se puede determinar mediante las lineas de código 2

Listing 2. Funcion Matriz Controlabilidad

```
abs(A);
```

Donde nuevamente  $\mathbf{A}$  es la matriz  $n \times n$  que se multiplica por las variables de estado, y  $\mathbf{C}$  son las salidas observables del sistema (para este caso la salida observable es la 2° variable de estado, si fuera la primera o todas, la matriz tendría 1 en los lugares correspondientes a cada variable de estado observable).

### D. Estabilidad

La estabilidad de un sistema está asociado a la capacidad de respuesta finita de un sistema para una entrada finita al mismo sistema. Esto quiere decir que el sistema al recibir una función acotada, dará por respuesta otra función acotada, sin divergir hacia  $+\infty$  o  $-\infty$ . Una manera de estudiar o revisar la estabilidad de un sistema es encontrando la ubicación de los polos de la matriz  $\mathbf{A}$  del la versión linealizada del sistema, si los polos de dicha matriz están en el semi-plano izquierdo del espacio, es decir que la parte real de todos los polo es negativa, entonces el sistema es estable, por otro si al menos un polo esta en el semi-plano derecho, entonces el sistema sera inestable, así mismo, si existen polos complejos conjugados, entonces el sistema tiene una respuesta oscilatoria. Cabe destacar que en un sistema, la estabilidad depende entonces netamente de la matriz  $\mathbf{A}$ , matriz donde se encuentran todos los polos del sistema. En MATLAB, los polos de una matriz se pueden encontrar mediante el siguiente comando 3

Listing 3. Funcion Matriz

```
polos_1 = eig(A_1)
```

### E. Sistema de un Péndulo de Furuta

El sistema desarrollado se describe con detenimiento en el artículo [2], dicho sistema consiste de una barra (la cual llamaremos barra a), la cual se mueve libremente en el plano horizontal, esta barra tiene un encoder en uno de sus extremos y un motor en su centro, el cual esta empotrado en una estructura estática, así mismo, en el otro extremo de esta barra esta enganchado a un péndulo invertido que gira libremente en el plano vertical.

Las variables de estado que rigen este sistema son el ángulo  $\theta, \phi, \dot{\theta}, \dot{\phi}$ , estos ángulos están ilustrados en la imagen 1, mientras que las que las ecuaciones diferenciales que describen al sistema son:

$$M(\theta) = \begin{pmatrix} M_2 l_{bi}^2 + 2 C_x M_2 l_{bi} + I_x \sin(\theta)^2 + I_z + J C_z M_2 l_{bi} \cos(\theta) \\ C_z M_2 l_{bi} \cos(\theta) \\ I_x \end{pmatrix} \quad (1)$$

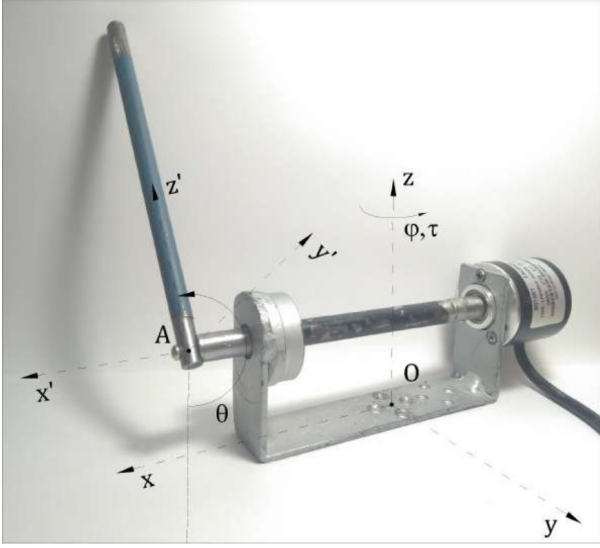


Fig. 1. Imagen demostrativa Péndulo Furuta

$$D(\theta, \dot{\phi}) = \begin{pmatrix} 2 I_x \dot{\theta} \cos(\theta) \sin(\theta) & -C_z M_2 \dot{\theta} l_{bi} \sin(\theta) \\ -I_x \dot{\phi} \cos(\theta) \sin(\theta) & 0 \end{pmatrix} \quad (2)$$

$$F(\theta) = \begin{pmatrix} 0 \\ C_z M_2 g \sin(\theta) \end{pmatrix} \quad (3)$$

$$U = \begin{pmatrix} T_{fu} + K i \\ T_{fp} \end{pmatrix} \quad (4)$$

$$\dot{i} = -\frac{K \dot{\phi} - V + R i}{L} \quad (5)$$

Las ecuaciones 1 a 5 representan el sistema dinámico compuesto por la barra a, el péndulo invertido y el motor, sin embargo aún queda despejar las variables de estado, esto se logra mediante la ecuación 7

$$Q = [\phi; \theta]; \quad (6)$$

$$\ddot{Q} = M^{-1}(U - D * \dot{Q} - F); \quad (7)$$

Finalmente mediante las ecuaciones 1 a 7 se pueden encontrar las variables de estado  $\theta, \phi, \dot{\theta}, \dot{\phi}, i$ , permitiendo representar la ecuación en el espacio de estados, también se puede linealizar la ecuación en torno a un punto de operación mediante el Jacobiano.

### III. DESARROLLO

El desarrollo de este trabajo se divide en 4 partes principales, 1° la implementación del modelo en MATLAB mediante las s-function, 2° el encontrar los 2 puntos de operación del sistema (es decir donde el sistema está en equilibrio), uno estable y uno inestable, 3° la linealización del sistema en torno a este punto de operación y finalmente la implementación del sistema linealizado y una comparación de ambos modelos.

#### A. Implementación del "Péndulo de Furuta" en MATLAB

Según el sitio oficial de MATLAB, una s-funtion es una subrutina vinculada de forma dinámica al motor de ejecución de MATLAB, se puede cargar y ejecutar automáticamente. así mismo, se puede enlazar a un bloque en Simulink para hacer simulaciones con bloques. Para implementar el sistema mediante una s-function de MATLAB, lo que se hizo fue:

- 1) Se crea un bloque el cual se nombró para una fácil identificación cómo "FurutaNoLineal".
- 2) Se configuran las entradas y salidas, (1 y 5 respectivamente).
- 3) Inicializamos las condiciones iniciales de las variables de estado
- 4) Inicializamos los parámetros para posteriormente calcular las matrices de la ecuación (7), consiguiendo así  $\ddot{Q}$ .
- 5) Calculamos la variable de estado de la corriente
- 6) Cómo  $\dot{x}_1 = x_3$  y  $\dot{x}_2 = x_4$ , tenemos todos los valores para  $\dot{x} = x_3, x_4, \ddot{Q}, x_5$ , teniendo en consideración  $\ddot{Q} = \dot{x}_3 \dot{x}_4$
- 7) Finalmente, se definen las salidas cómo las variables de estado.

Es necesario destacar que en la implementación de la s-funtion, se considero como única entrada un voltaje, mientras que se consideran en primer lugar 5 salidas, las cuales son las 5 variables de estado del sistema  $([\phi, \theta, \dot{\phi}, \dot{\theta}, i])$ , con el fin de comprar estas y encontrar el rango en los que el sistema lineal sirve como una aproximación fiel al sistema original.

#### B. Puntos de operación

Los puntos de operación de interés para nosotros son los puntos de equilibrio, dónde al no tener ninguna entrada no debería haber un cambio en el sistema. Analíticamente, estos puntos serían aquellos que se encuentran para la variable  $\Theta$  en  $\pi$  o en 0, siendo el primero al estar en un punto dónde hay más energía potencial un punto de equilibrio inestable, mientras que en el punto de operación con ángulo 0 al no tener energía potencial sería un punto de equilibrio estable.

En MATLAB, una vez obtenido el modelo no-lineal uno puede simular el modelo para encontrar los puntos de operación, para ello utilizamos el comando "trim", siguiendo los pasos a continuación:

- 1) Crear un modelo de simulink del modelo FurutaNoLineal con 1 entrada y 5 salidas.
- 2) Crear un Script para el cálculo, registrando las condiciones iniciales  $X_0, U_0$  y  $Y_0$ , teniendo en consideración que la variable  $X_2 = \pi$  ó 0 (2).
- 3) Posteriormente, se definen que parámetros se dejarán "fijos" y cuáles "libres" para ir variando en la búsqueda del punto de operación, en este caso sería  $X_f = 2$ , dejando fija la segunda variable de estado fija y todas las otras variables de estado, entrada y salidas libre para el cálculo (3).
- 4) Se utiliza el comando trim con los parámetros calculados anteriormente, devolviendo los puntos de operación para X, U e Y (4).

## Condiciones iniciales:

```
x0 = [0;pi;0;0;0];  
u0 = [0];  
y0 = [0;0;0;0;0];
```

Fig. 2. Definición condiciones iniciales MATLAB Péndulo Furuta

## Variables fijas:

```
xf = [2];  
uf = [];  
yf = [];
```

Fig. 3. Definición variables fijas Péndulo Furuta MATLAB

Cálculo Punto de operación en Theta = Pi.

```
[x_op,u_op,y_op] = trim("trim_simulation",x0,u0,y0,xf,uf,yf)  
  
x_op = 5x1  
0  
3.1416  
0  
0  
0  
  
u_op = 0  
y_op = 5x1  
0  
3.1416  
0  
0  
0
```

Fig. 4. Obtención Puntos de Operación Péndulo Furuta

Destacar que el punto de operación  $\theta=\pi$ , se eligió mediante lógica en un primer lugar, pues en este punto el péndulo invertido se encuentra en el lugar mas alto posible, y ante cualquier perturbación caería. Así mismo, para el punto  $\theta=0$ , se considero a simple vista como punto de operación estable. Esto se comprueba después en la sección de controlabilidad y observabilidad.

### C. Linealización para el punto de operación

Aclarar primero que la razón por la que se linealizan diversos sistemas dinámicos no lineales es debido a que gran parte de la teoría de control es aplicable a sistemas lineales, mientras que para el caso de los sistemas no lineales, las estrategias de control aplicables son mucho mas acotadas.

Teniendo el punto de operación, uno puede linealizar el modelo dinámico para encontrar las matrices  $[A,B,C,D]$  y tener el modelo lineal para tal punto. Esto se puede hacer de forma analítica mediante el Jacobiano, sin embargo Matlab tiene una función sencilla llamada "linmod", el cual recibe

como parámetro el modelo y los puntos de operación, recibiendo como respuesta las matrices  $[A,B,C,D]$  5

```
[A,B,C,D] = linmod("trim_simulation",x_op,u_op)
```

```
A = 5x5  
0 0 1.0000 0 0  
0 0 0 1.0000 0  
0 1.1278 0 0 0  
0 2.2193 0 0 0  
0 0 -0.0306 0 -0.2158  
  
B = 5x1  
0  
0  
-0.2667  
-0.7913  
0.0232  
  
C = 5x5  
1.0000 0 0 0 0  
0 1.0000 0 0 0  
0 0 1.0000 0 0  
0 0 0 1.0000 0  
0 0 0 0 1.0000  
  
D = 5x1  
0  
0
```

Fig. 5. Linealización Péndulo Furuta

### D. Determinación de Controlabilidad, Observabilidad y número de estados Estables y Controlables

Una vez obtenido el modelo linealizado en torno a un punto de operación dado (es este caso con  $\theta=\pi$ ) y con él las matrices  $[A,B,C,D]$ , se puede estudiar la controlabilidad y observabilidad del sistema. En este caso, al ejecutar las funciones descritas en la sección de Marco teórico se determinó que el sistema es completamente controlable ya que el rango de la matriz de controlabilidad es igual al número de variables de estado del sistema 7 y completamente observable por las mismas razones 6.

### Observabilidad

```
observabilidad = rank(observ(A,C))
```

```
observabilidad = 5
```

Fig. 6. Observabilidad Péndulo Furuta en punto de operación inestable

### Controlabilidad

```
controlabilidad1 = rank(ctrb(A,B))
```

```
controlabilidad1 = 5
```

Fig. 7. Controlabilidad Péndulo Furuta en punto de operación inestable

Finalmente se obtiene la estabilidad de los puntos de operación descritos en la sección anterior como se puede observar en los polos mostrados en las figuras 9 (punto de operación  $\theta = 0$ ) para el punto de operación estable y 10 para el punto de operación inestable.

Cálculo de las matrices de espacio de estados.

```
[A,B,C,D] = linmod("trim_simulation",x_op,u_op)
```

A = 5x5

0	0	1.0000	0	0
0	0	0	1.0000	0
0	1.1278	0	0	0
0	-2.2193	0	0	0
0	0	-0.0306	0	-0.2158

B = 5x1

0
0
0.2666
-0.2580
0.0232

C = 5x5

1.0000	0	0	0	0
0	1.0000	0	0	0
0	0	1.0000	0	0
0	0	0	0	1.0000
0	0	0	0	0

D = 5x1

0
0
0
0
0

Fig. 8. Matriz A polo estable ( $\theta = 0$ )

Polos

```
polos_1 = eig(A)
```

polos\_1 = 5x1 complex

0.0000 + 0.0000i
-0.2158 + 0.0000i
0.0000 + 0.0000i
0.0000 + 1.4897i
0.0000 - 1.4897i

Fig. 9. Estabilidad Péndulo Furuta en punto de operación estable

Polos

```
polos_1 = eig(A)
```

polos\_1 = 5x1

0
-0.2158
0
1.4897
-1.4897

Fig. 10. Estabilidad Péndulo Furuta en punto de operación inestable

## E. Implementación

Una vez obtenido las matrices del espacio de estados, se crea otro modelo de S-function para el modelo lineal de la misma forma que se realizó en el modelo FurutaNoLineal. Para ello se realizó lo siguiente:

- 1) Se configuran las entradas y salidas.
- 2) Se configuran las condiciones iniciales.
- 3) Se asignan las matrices A,B,C,D obtenidas anteriormente.
- 4) se define x cómo las variables de estado, y U cómo la variable de entrada
- 5) se asigna la ecuación cómo  $\dot{x} = Ax + Bu$ , siendo  $\dot{x}$  para este caso las derivadas y las salidas.

1) *Comparación de resultados y discusión:* Una vez creado un simulink con los 2 sistemas representados y conectados (11), es posible comprar los resultados de ambos modelos con el fin de comprender los rangos donde el sistema linealizado es una buena aproximación del sistema original. Como se

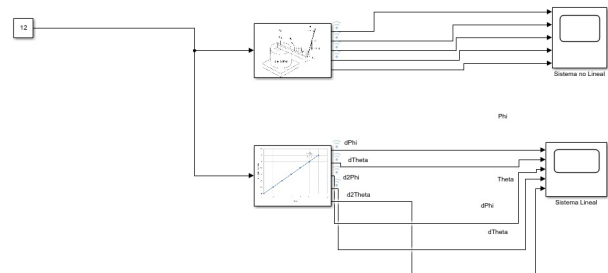


Fig. 11. Simulink Péndulo Furuta y sist. linealizado

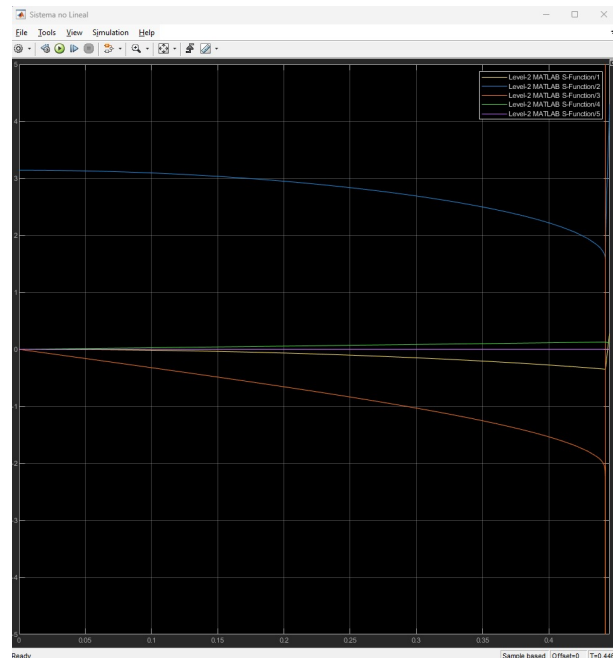


Fig. 12. Evolución variables de estado Péndulo Furuta en punto de operación inestable, sistema no lineal



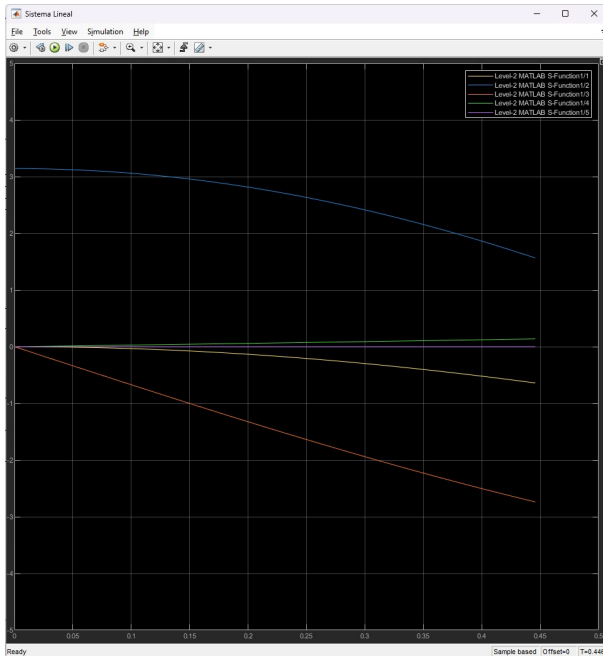


Fig. 13. Evolución variables de estado Péndulo Furuta en punto de operación inestable, sistema lineal

puede observar en las figuras 13 y 12, las variables de estado para una entrada de tipo constante de 12 V son similares en torno al punto de operación  $\theta = \pi$ , sin embargo es posible percibir cambios conforme el sistema se aleja del punto de operación, el tiempo transcurrido en la simulación es relativamente bajo (de alrededor de 0.5 segs) , sin embargo los sistemas no se diferencian con tanta rapidez, situación que se debiera tomar en cuenta al momento de pensar en controlar el sistema.

#### IV. CONCLUSIÓN

A partir de lo desarrollado e implementado entre modelo linealizado y modelo No-lineal del Péndulo de Furuta, se puede apreciar la importancia de los puntos de operación y su importancia en lo que es controlabilidad, linealidad, observabilidad, etc. Todo esto sirve para tener en consideración el comportamiento esperado para el modelo en un punto determinado y las acciones necesarias para más adelante poder aplicar control al mismo. Se encontró que el sistema es completamente controlable y observable en ambos puntos de operación, lo que indica que es posible diseñar estrategias de control efectivas para manipular su comportamiento dinámico. Además, se determinó la estabilidad de los puntos de operación mediante el análisis de los polos de la matriz A, confirmando que el punto de operación con  $\theta = 0$  es estable y el punto con  $\theta = \pi$  es inestable.

#### REFERENCES

- [1] K. Ogata, *Ingeniería de control moderna*. Pearson Educación, 2003.
- [2] A. Bellati, F. Cancela, and N. Pérez Blengio, "Construcción y control del péndulo de furuta." 2021.