



UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA

Escuela Profesional de Ingeniería Electrónica

Informe

Laboratorio 01a: Introducción

Alumnos

Mamani Huanca, Jhoel Rene
Vilcapaza Goyzueta, Denilson Jean

Profesor: Dr. Juan C. Cutipa Luque

3 de septiembre de 2020

Resumen

Este documento presenta el informe sobre el modelamiento de un sistema péndulo invertido, con el método de espacio de estados. Se desarrollan controladores PID para controlar el ángulo en una posición inicial y se controla la posición. También a este modelado se le incluyen ruidos externos conocidos como disturbios para verificar la robustez del controlador.

Índice

1. Objetivos	1
2. Fundamento Teórico	1
3. Materiales y Equipamientos	1
4. Procedimientos	1
5. Cuestionario	19
6. Conclusiones y observaciones	19
Referencias Bibliográficas	20
Apéndice	21
Rúbrica	22

1. Objetivos

- Realizar el modelamiento en espacio de estados de un sistema péndulo invertido

2. Fundamento Teórico

El sistema de péndulo invertido es uno de los modelos físicos mas importantes para desarrollar dinámicas de sistemas, El péndulo invertido es un sistema inestable de bucle abierto y de bucle cerrado inherentemente con una dinámica altamente no lineal[3].

El PIC(sistema péndulo invertido sobre un carro) es considerado un sistema SIMO (Single Input, Multiple Output) el cual es inherentemente inestable, ya que al posicionar el péndulo con un ángulo menor o igual a 90 sobre la vertical superior, es imposible que permanezca recto, debido a que no existe alguna fuerza aplicada que lo mantenga sobre la vertical superior [1].

3. Materiales y Equipamientos

- Gnu-Octave
- Computador o Smartphone.
- Cuaderno de apuntes.
- Python (opcional).

4. Procedimientos

1. **Encontrar la solución numérica del sistema péndulo invertido (no lineal) y visualizar respuestas de las cuatro variables de estado en función de diferentes condiciones iniciales. Mostrar por lo menos 2 casos (ayuda: puede usar la función ODE45 de Gnu-Octave).**

Desarrollo matemático modelo no lineal de pendulo invertido

De la figura se obtiene lo siguiente:

$$I\ddot{\theta} = H_y l \sin(\theta) - H_x l \cos(\theta) \quad \text{Ec. Por momento de torsión} \quad (1)$$

$$H_x = m\ddot{x} - ml \sin(\theta) \dot{\theta}^2 + ml \cos(\theta) \ddot{\theta} \quad (2)$$

$$H_y - mg = ml(-\cos(\theta)) \dot{\theta}^2 - \sin(\theta) \ddot{\theta} \quad (3)$$

$$m\ddot{x} \cos(\theta) + ml \ddot{\theta} = mg \sin(\theta) \quad (4)$$

$$F = (M + m)\ddot{x} - ml \sin(\theta) \dot{\theta}^2 + ml \cos(\theta) \ddot{\theta} \quad (5)$$

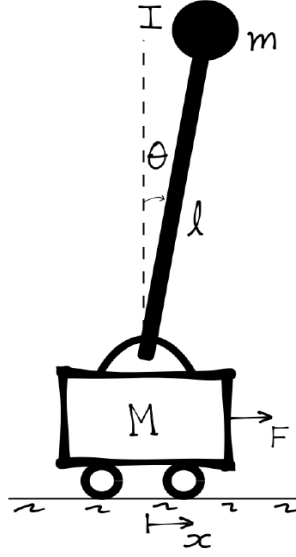


Figura 1: Péndulo Invertido modelo físico

Reemplazando las ecuaciones

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

$$0 = \frac{mg\sin(\theta)}{m\cos(\theta)} - \ddot{x} - \frac{ml\ddot{\theta}}{m\cos(\theta)} \dots (b)$$

despejando la ecuaciones (a) y (b), obtenemos:

$$\ddot{\theta} = \frac{-ml\sin(\theta)\cos(\theta)\dot{\theta}^2 + (m+M)g\sin(\theta) - \cos(\theta)F}{(m+M)l - ml\cos(\theta)^2} \quad (6)$$

Ecuaciones para la posición, de las ecuaciones (4) y (3) obtenemos:

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

Obtenemos:

$$\ddot{x} = \frac{F + ml\sin(\theta)\dot{\theta}^2 - mg\sin(\theta)\cos(\theta)}{m+M - m\cos(\theta)^2} \quad (7)$$

Modelando en espacio de estados

$$\dot{\theta} = \dot{x}_1$$

$$\dot{x}_2 = \frac{-ml\sin(\theta)\cos(\theta)\dot{\theta}^2 + (m+M)g\sin(\theta) - \cos(\theta)F}{(m+M)l - ml\cos(\theta)^2}$$

$$\dot{x}_3 = \dot{x}_1$$

$$\dot{x}_4 = \frac{F + m l \sin(\theta) \dot{\theta}^2 - m g \sin(\theta) \cos(\theta)}{m + M - m \cos(\theta)^2}$$

NOTA:

Por cuestiones de solución didáctica se realizó el cambio de variable "F" por "U"

CONSIDERACIONES:

Se utilizan parámetros Linealizados para calcular una respuesta, los cuales para el sistema de la figura 1 son:

- Masa carro (M)= 2Kg
- masa pendulo (m)= 0.1Kg
- gravedad (g) = 9.81 m/s
- longitud (l) = 0.5m
- Fuerza = 0.5N en casos discretos como constante.
Y también se realizará respuesta al impulso para simulación.

Programa Principal : Solución con ODE45 - Octave

```

1 %% LABORATORIO 01 - SCA      [PREGUNTA 01]
2 %
3 %      Jhoel Ren  Mamani Huanca
4 %      Denilson Vilcapaza Goyzueta
5 %      GRUPO: A
6 %
7 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
8 % 01. Sistema pendulo invertido(modelo no lineal) con ODE45
9 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
10 clc; clear; close all;
11
12 % definici n de los valores del sistema
13 M = 2; m = 0.1; l = 0.5; g = 9.8; u = 0.1; mu = 0;
14
15 % condicion inicial
16 %x0 = [pi/4;0;0;0];
17 x0 = [1 0 0.01 0];
18
19 Tf = 6;      % tiempo final
20 T = [ 0:0.01: Tf]; % tiempo
21
22 [ts,x] = ode45(@(t,x)penduloinv(t,x,u,m,M,l,g,mu), T, x0);
23
24 subplot(2,2,1)
25 plot(ts,x(:,1))
26 title('\theta')
27
28 subplot(2,2,2)
29 plot(ts,x(:,2))
30 title('\theta''')
31

```

```

32 subplot(2,2,3)
33 plot(ts,x(:,3))
34 title('x')
35
36 subplot(2,2,4)
37 plot(ts,x(:,4))
38 title('x'')
39
40 figure(2)
41 plot(ts,x)
42
43 lgd=legend('\theta','\theta','x','x','location','best');
44 %title(lgd,'PENDULO INVERTIDO')
45 title('PENDULO INVERTIDO'); grid

```

Programa Secundario : Función pendulo invertido

```

1 function dx = penduloinv(t,x,u,m,M,l,g,mu)
2
3 % ESTADOS
4 x1 = x(1);
5 x2 = x(2);
6 x3 = x(3);
7 x4 = x(4);
8
9 % Ecuaciones diferenciales
10 dx1 = x2; % theta
11 dx2 = ( u*cos(x1) - (M+m)*g*sin(x1) + (m*l*(cos(x1)*sin(x1))*x2^2)
12 ) / (m*l*(cos(x1))^2 - (M+m)*l);
13 %dx2 = (-m*l*sin(x(1))*cos(x(1))*x(2)^2+(m+M)*g*sin(x(1))-cos(x(1))
14 ) *F) / ((m+M)*l - m*l*cos(x(1))^2);
15 dx3 = x4; % x
16 dx4 = ( u + m*l*(sin(x1))*(x2)^2 - m*g*cos(x1)*sin(x1) ) / (M
17 + m - m*(cos(x1))^2);
18 % dx4 = (m*l*sin(x(1))*x(2)^2 - m*g*sin(x(1))*cos(x(1))+F) / (m+M - m
19 *cos(x(1))^2);
20
21 dx = [dx1,dx2,dx3,dx4];

```

Después de la simulación obtenemos como resultados las gráficas:

■ CASO 1: Para condiciones iniciales de:

Masa carro = 2Kg

masa pendulo = 0.1Kg

gravedad = 9.81

longitud = 0.5m

Fuerza = 0.5N

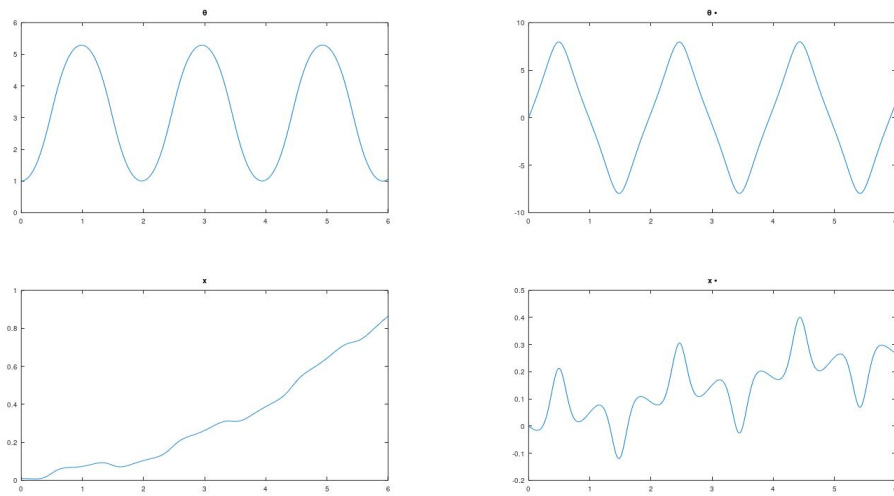


Figura 2: Péndulo Invertido no lineal

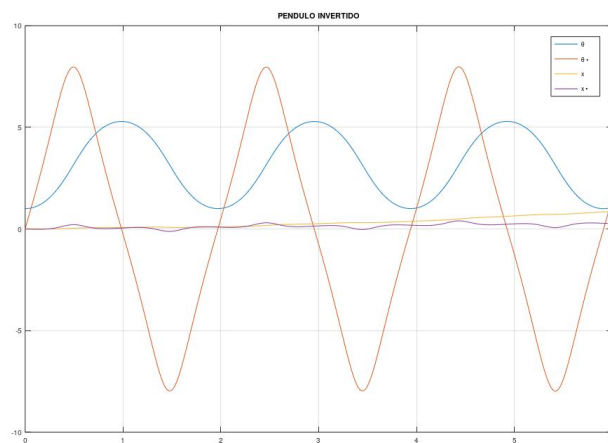


Figura 3: Péndulo Invertido

Figura 4: Péndulo invertido modelo no lineal

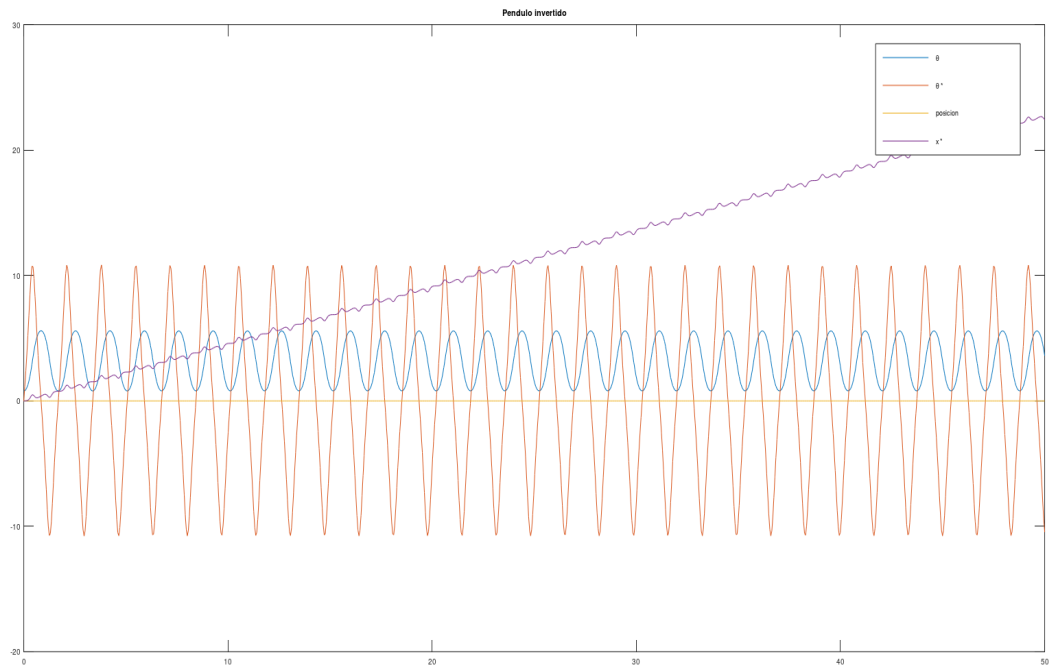
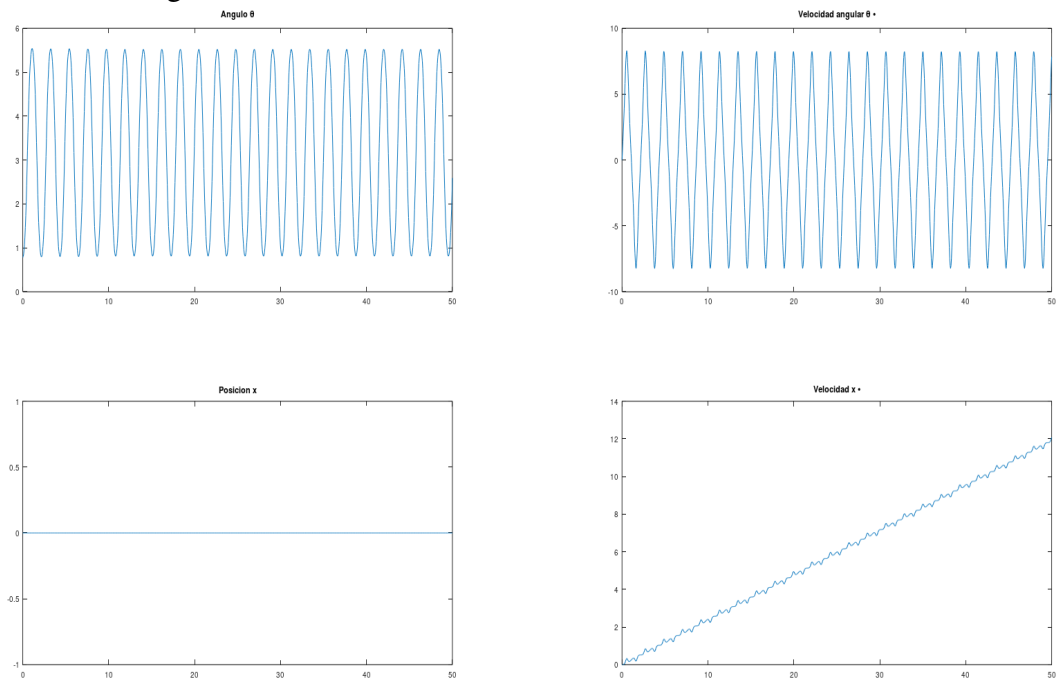


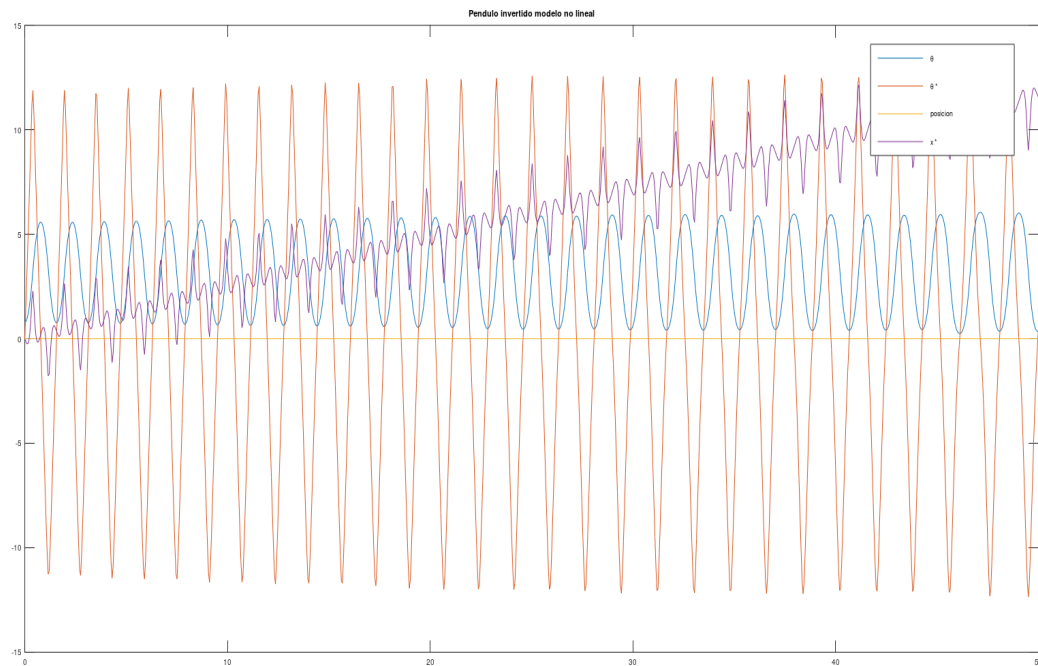
Figura 5: Gráficas de las variables de estado modelo no lineal



■ **CASO 2: Para condiciones iniciales de:**

Masa carro = 1Kg
masa pendulo = 1Kg
gravedad = 9.8
longitud = 0.5m
Fuerza = 0.5N

Figura 6: Gráficas de las variables de estado modelo no lineal



2. Repetir lo anterior con método de integración numérica Runge Kutta, codificado por usted mismo. Este método debe ser genérico; o sea, una función parecida a ODE45 que permita resolver cualquier sistema multivariable y no lineal (ver figura 2 para representación en diagrama de bloque).

Figura 7: Diagrama de Bloque del Péndulo



Incluyendo en el código el método de Runge Kutta, Creando un script(función), parte de código del método Runge Kutta.

Parte de código para Péndulo invertido

```

1
2 %% Funcion Range KUTA de 4TO ORDEN
3 function [t,xdo] = RK4(f,t0,tf,x0,h)
4
5     t = t0:h:tf;
6     length(t);
7     x=x0';
8     xdo=x0;
9     for i=1:(length(t)-1)
10         k1 = f(t(i),x);
11         k2 = f(t(i)+0.5*h,x+0.5*h*k1);
12         k3 = f(t(i)+0.5*h,(x+0.5*h*k2));
13         k4 = f(t(i)+h,(x+k3*h));
14         x = x + (1/6)*(k1+2*k2+2*k3+k4)*h;
15         xdo(i+1,:)=x;
16     end
17 end

```

Tenemos como resulatd las graficas siguientes:

■ CASO 1: Para condiciones iniciales de:

Masa carro = 2Kg

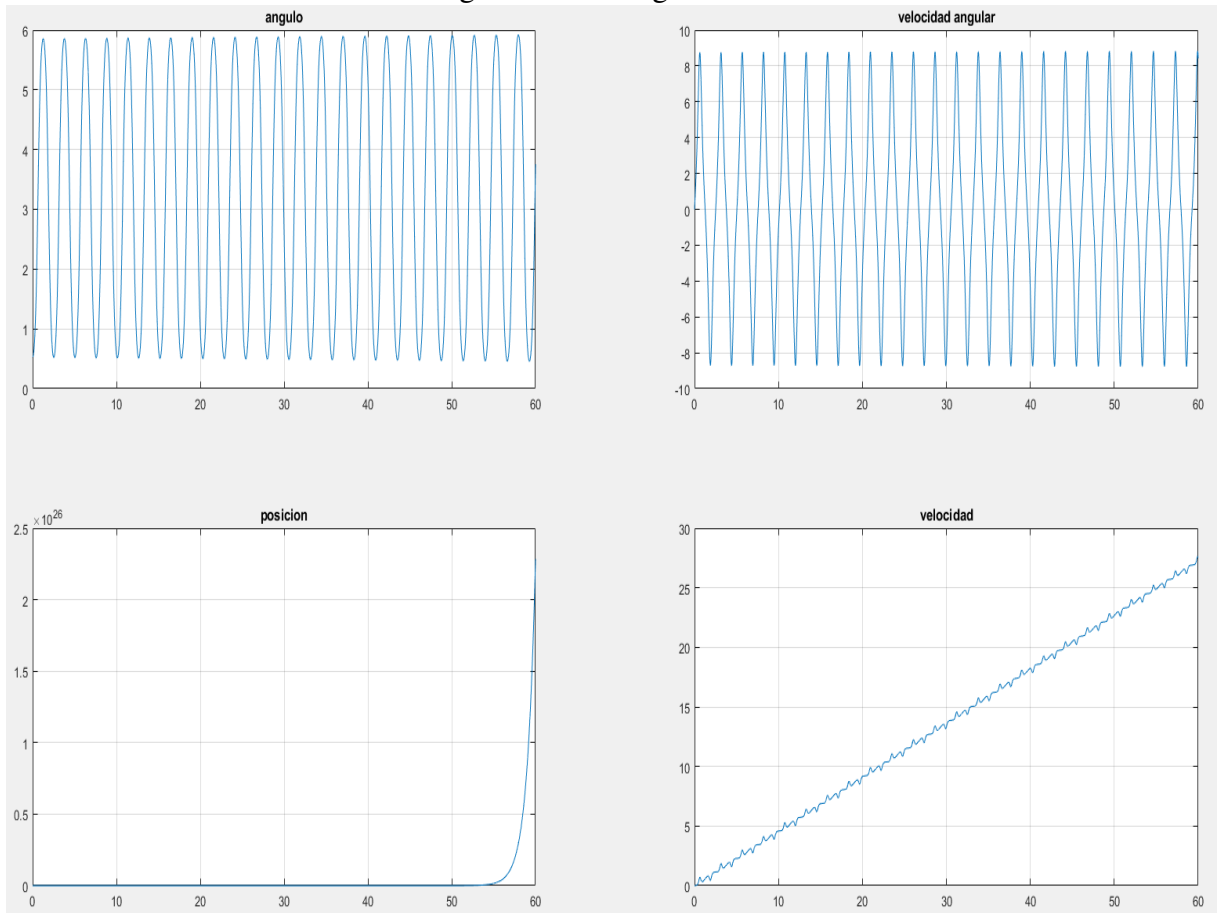
masa pendulo = 0.1Kg

gravedad = 9.8

longitud = 0.5m

Fuerza = 0.5N

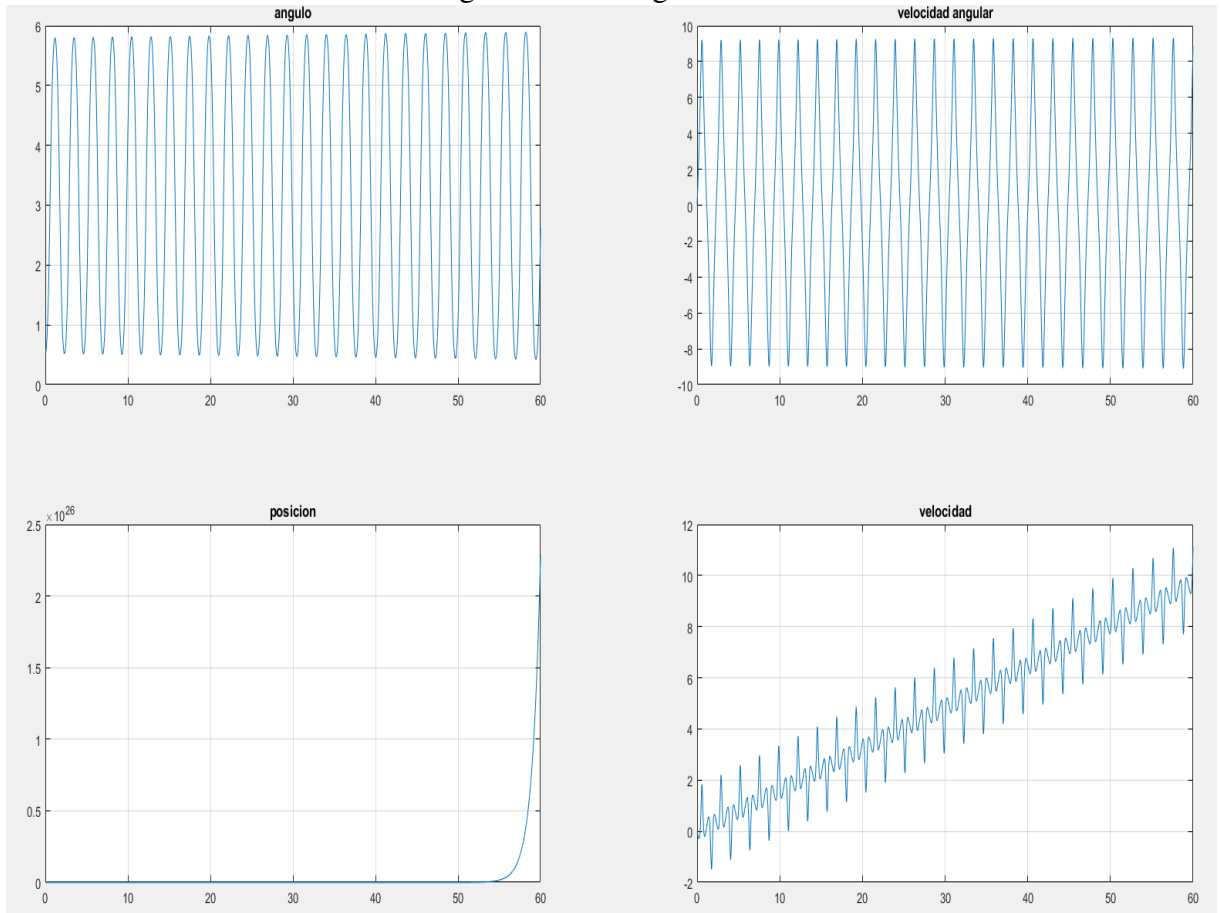
Figura 8: Caso 1 gráficas



■ **CASO 2: Para condiciones iniciales de:**

Masa carro = 1Kg
 masa pendulo = 1Kg
 gravedad = 9.8
 longitud = 0.5m
 Fuerza = 0.5N

Figura 9: Caso 2 gráficas



3. Linealizar el modelo de péndulo y obtener controladores PID para controlar la posición del carro y en ángulo del péndulo (ver figura 3 para representación en diagrama de bloques).

- calculo matemático PID

$$t_s(5\%) = \frac{3}{\rho}$$

despejando

$$\rho = 3$$

Maximo sobrepico:

$$Mp = \frac{-\pi}{\epsilon^{tag(\phi)}}$$

Despejando:

$$\phi = 66$$

Tiempo de asentamiento:

$$t_s = \frac{3}{\xi W_n}$$

Donde:

$$W_n = 7,5 \text{ rad/s}$$

Así:

$$W_d = 7,5 \sqrt{1 - 0,4^2}$$

$$W_d = 6,87$$

Luego por ubicación de los polos se obtiene:

$$s = -3 \pm j6,87$$

Luego se calcula DA la Deficiencia del ángulo La DA en el punto $S = -3 - j6,87$

$$DA + 0 = 180 + (180 - \arctan(\frac{6,87}{7,83})) + (\arctan(\frac{6,87}{1,93}))$$

$$DA = 180 + 137,62 + 77,4$$

$$DA = 395 = 360 + 35$$

$$DA = 35$$

Finalmente:

$$PID = G_c = \frac{6,25(s + 12,81)(s + 0,687)}{s}$$

Dando la Forma de PID ecuación:

$$G_{PID}(s) = K_p + \frac{K_i}{s} + K_d s$$

Luego obtenemos:

$$K_p = 84,3562$$

$$K_i = 55,003$$

$$K_D = 6,25$$

Para linealizar el modelo se considera parámetros $I = 0$, $\theta = 0$ con posición vertical el ángulo es muy pequeño que se considera 0.

Esto afecta a: $\text{sen}(\theta) = \text{sen}(0) = 0 = \theta$

Programa principal : Controlador PID $C(s)$ y Planta $P(s)$. Respuesta para péndulo estabilización

```

1 %% LABORATORIO 01 - SCA [PREGUNTA 03]
2 %
3 % Jhoel Ren Mamani Huanca
4 % Denilson Vilcapaza Goyzueta
5 % GRUPO: A
6 %
7 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
8 % 03. Linealizar el modelo de p ndulo y obtener controladores PIDs
9 % control posici n y ngulo
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11 clc, clear all, close all;
12
13
14 M = 2; m = 0.1; l = 0.5; g = 9.8; I = 0; b=0;
15
16 % Funcion de Transeferencia
17 numtf = [1];
18 dentf = [M*l 0 -(M+m)*g];
19 G=tf(numtf, dentf) %% Funcion de transferencia L.A. Angulo
20
21 s = tf('s');
22
23 %numtf2 = [M*l 0 -(M+2*m)*g];
24 numtf2 = [M*l 0 -(M)*g];
25 dentf2 = [M*M*l 0 -M*(M+m)*g 0 0];
26 G2=tf(numtf2, dentf2) %% Funcion de transferencia L.A. Posicion
27
28 %figure(1)
29 %rlocus(G)
30
31 % PID
32 Kp = 84.35625;
33 Ki = 55.003;
34 Kd = 6.25;
35
36 %PID = (Kd*s^2+Kp*s+Ki)/s;
37 C = pid (Kp, Ki, Kd);
38 T = feedback(G,C)
39 figure(1)
40 rlocus(T) % Para ver Lugar de raices con el PID polos zeros
41 t=0:0.01:7;
42 figure(2)
43 impulse(T,t) % Respuesta
44 axis([0, 2.5, -0.2, 0.2]);
45 title({'Respuesta p ndulo \theta ',' PID : Kp = 84.35625, Ki =
46 55.003, Kd = 6.25'});
47
48 % Kp2 = 43.351;
49 % Ki2 = 24.408;
50 % Kd2 = 7.772;
51
52 % PID Los valores de K son obtenidos por m todo matem tico
53 Kp2 = -7.0559;
54 Ki2 = -0.61746;
55 Kd2 = -20.1577;
56
57 %%C2 = pid (Kp2, Ki2, Kd2);
58 %%T2 = feedback(G2,C2)
59 %%
60 %%t2=0:0.01:120; % rangos

```

```

60 %%figure(3)
61 %%impulse(T2,t2) % Angulo
62 %%axis([0, 30, -0.05, 10]);
63 %%title({'Respuesta carro';' PID : Kp = 43.351, Ki = 24.408, Kd =
    8.772'});
64
65 q = (M+m)*(I+m*l^2)-(m*l)^2;
66 P_cart = (((I+m*l^2)/q)*s^2 - (m*g*l/q))/(s^4 + (b*(I + m*l^2))*s^3/
    q - ((M + m)*m*g*l)*s^2/q - b*m*g*l*s/q); %% funcion de
    transferencia para L.A
67
68 numm = P_cart;
69 denn = 1+G*C;
70 GG = numm/denn;
71
72 C3 = pid (100, 1, 20); %% Kp, Ki, Kd
73 %T3 = feedback(P_cart,C3)
74 T3 = feedback(1,G*C3)*P_cart; % Realimentaci n de figura anterior
75
76 t3=0:0.01:10;
77 figure(4)
78 impulse(T3,t3) % Respuesta Posici n carro
79 axis([0, 2.5, -0.05, 0.05]);
80 title({'Respuesta carro';' X '});

```

Gráficas luego de simular el programa anterior:

Para la respuesta de angulo se realizó primero el modelamiento matemático.

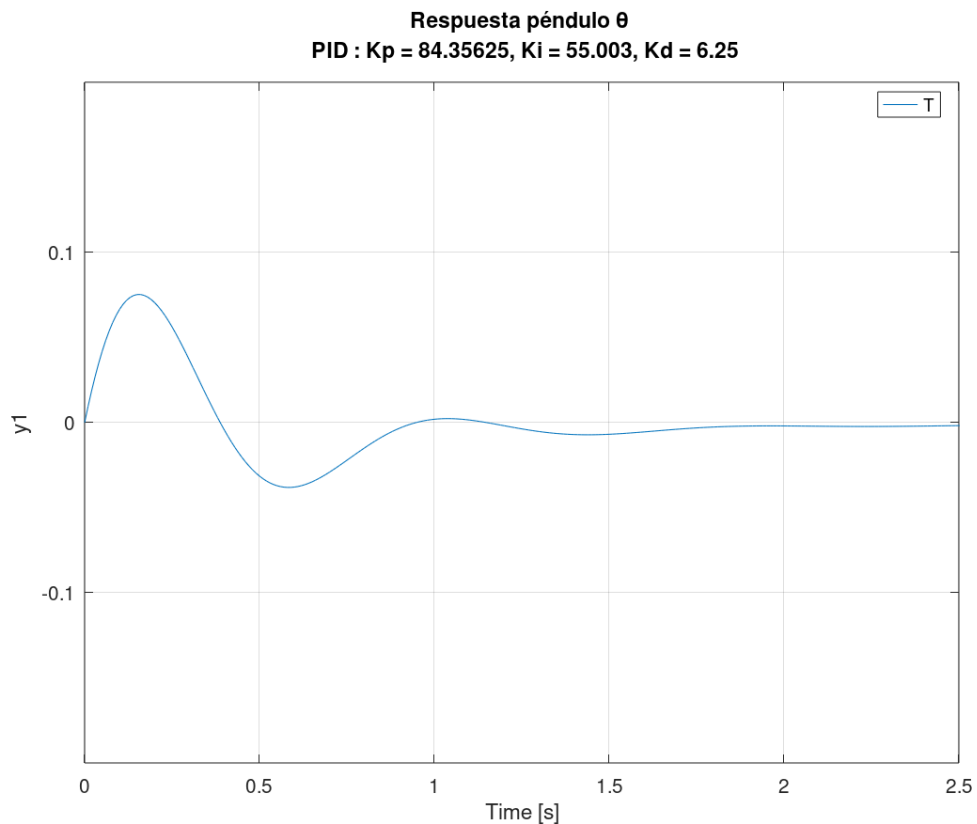


Figura 10: Respuesta ángulo con control PID.

En esta gráfica podemos apreciar que las consideraciones de Tiempo de asentamiento se cumplen, este PID que se agregó logra controlar el sistema realizándolo estable. Pa-

ra esto se eligió K_p , K_i y K_d según a método de lugar de raíces. Este análisis se realiza en papel.

Para comprobar la estabilidad luego de colocar el $C(s)$ que es el PID, se puede apreciar con Octave la función Rlocus que nos muestra el lugar de las raíces para ubicar s .

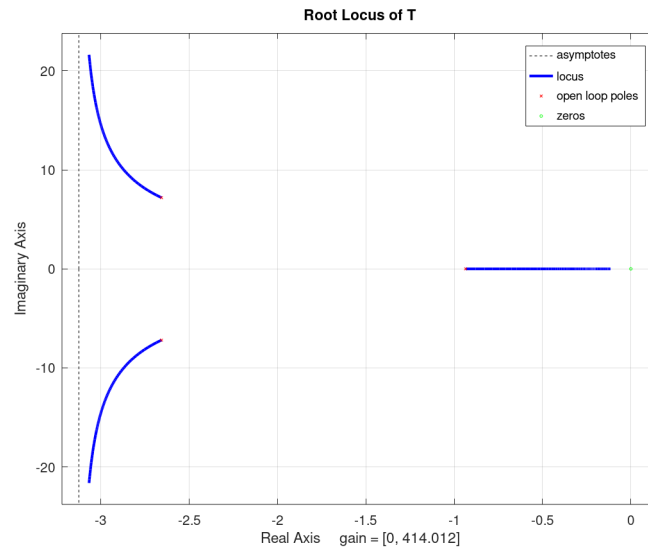


Figura 11: Rlocus para comprobar lugar de raíces.

También es apreciable que la sobreelongación $M_p \leq 25\%$ y también el tiempo de asentamiento $t_s = 1s$ considerando t_s de 5 %

Ahora, para la ver la respuesta a la posición en el carro se realiza una combinación para diseñar el controlador PID. Para este caso también se utilizó método de lugar de raíces para poder calcular la función de transferencia de nuestro controlador. En este caso, para la posición el control PID es $K_p = 100$, $K_i = 1$, $K_d = 20$. También por cálculo sale : $K_{p2} = -7.0559$; $K_{i2} = -0.61746$; $K_{d2} = -20.1577$;

Finalmente podemos decir que:

El controlador PID es capaz de estabilizar el ángulo del péndulo pero la posición del carro cambia y se desplaza a una velocidad constante por lo cual con este tipo de controlador no se consigue el objetivo planteado.[2].

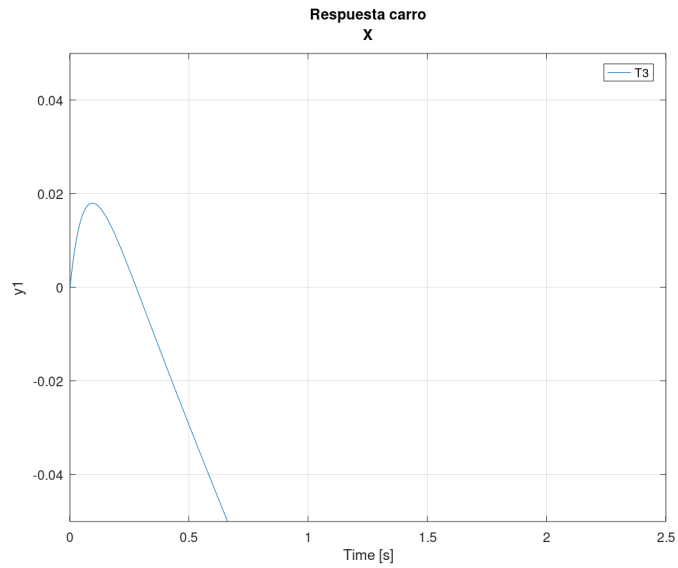


Figura 12: Respuesta X.

4. Con los controladores propuesto en el ítem anterior, obtener la respuesta del sistema controlado (controladores PID y modelo no lineal). Este script debe permitir ajustar manualmente las constantes de los controladores.

- calculo matemático PID

Programa principal : Control Sistema. Respuesta para péndulo estabilización

```

1 %% LABORATORIO 01 - SCA [PREGUNTA 04]
2 %
3 % Jhoel Ren Mamani Huanca
4 % Denilson Vilcapaza Goyzueta
5 % GRUPO: A
6 %
7 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
8 % 04.
9 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
10 clear all
11 close all
12 clc
13
14 f1 = @penduloinv;
15
16 Tf = 6; % tiempo final
17 T = [ 0:0.01: Tf]; % tiempo
18
19 x0 = [1 0 0.01 0]; % inicializacion
20
21 K = [-35 -34 -150 -34]; %Ganancia para ley control  $U = -kX$ 
22
23 [t,x]=ode45(f1,T, x0, [],K);
24
25 % PLOT!
26 figure(1)
27 subplot(2,2,1)
28 plot(t,x(:,1))
29 title('x')
30 grid
31
32 subplot(2,2,2)
33 plot(t,x(:,2))
34 title('x''')
35 grid
36
37 subplot(2,2,3)
38 plot(t,x(:,3))
39 title('\theta')
40 grid
41
42 subplot(2,2,4)
43 plot(t,x(:,4))
44 title('\theta''')
45 grid

```

Gráficas

5. En el script anterior, adicione una señal de disturbio con ruido gaussiano y un ligero impulso mayor. Repetir las simulaciones.

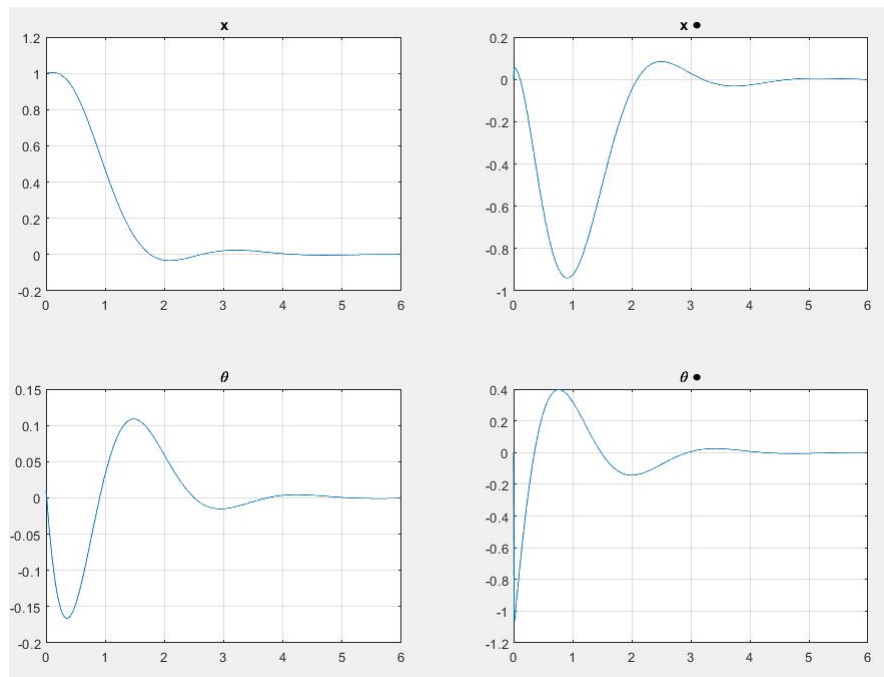


Figura 13: Respuestas Control de Sistema - ESTABLE.

Programa principal : Control Sistema. Ruido Gaussiano

```

1 %% LABORATORIO 01 - SCA [PREGUNTA 05]
2 %
3 %      Jhoel Ren Mamani Huanca
4 %      Denilson Vilcapaza Goyzueta
5 %      GRUPO: A
6 %
7 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
8 % 05.
9 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
10 %% LABORATORIO 01 - SCA [PREGUNTA 05]
11 %
12 %      Jhoel Ren Mamani Huanca
13 %      Denilson Vilcapaza Goyzueta
14 %      GRUPO: A
15 %
16 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
17 % 03. Ruido Gaussiano
18 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
19 clc; clear; close all;
20 %% Deefinicion de parametros:
21
22 m = 3; %
23 g = 9.81;
24 M = 10;
25 l = 1.5;
26 F1 = 0.001; % Magnitud de ruido plantaMagnitude of Plant White Noise
27 F2 = 0.001; % Magnitud de medida con ruido blanc
28
29 % Crear Modelo L.A
30 [A, B, C, D] = create_ol_sys(m, M, l, g)
31
32 % Observar y Control matricial
33 c = is_controllable(A, B);
34 o = is_observable(A, C);

```

```

35
36 %% LQR
37 G = optimal_lqr(A, B, C, D);
38
39 %% Laso Cerrado Modelo
40 % Rudiso
41 Ac = A-B*G;
42 clSys = ss(Ac, B, C, D)
43 % Ruido
44 clSysN = op2cl_noise(Ac, B, C, D, F1, F2);
45 % condiciones iniciales
46 y_0 = 1.5; %
47 dy_0 = 0; %
48 theta_0 = -pi; %
49 dtheta_0 = 0; %
50 tf = 7; %
51 dt = 0.1; %
52 % dt = tf/750; %
53 live = 't'; %
54 %SIMULAR
55 [y, t, x] = sim_inv_pend(tf, dt, F1, F2, y_0, dy_0, theta_0,
    dtheta_0, clSysN, live, l, G, B);

```

6. Plotear las curvas de respuesta similares a la figura 4.

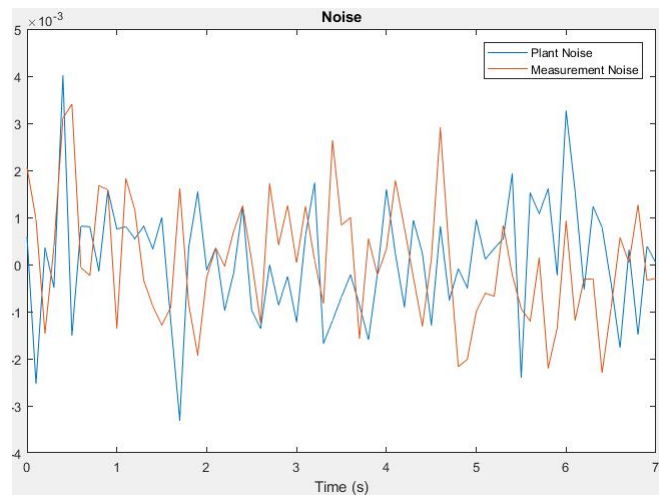


Figura 14: Grafica.

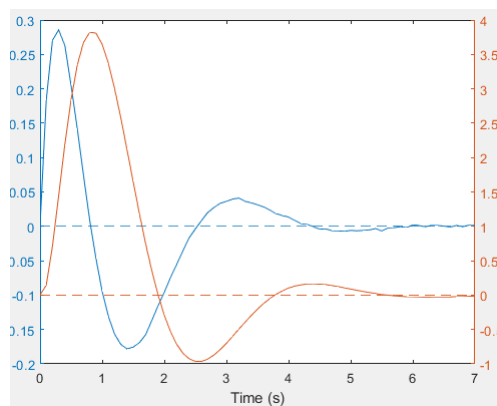


Figura 15: Grafica.

NOTA: Los archivos y scripts se encuentran en el enlace del laboratorio:

URL: https://drive.google.com/drive/folders/1Jmj5cY7hGSv_9vpqy_PuipJ0snVlCb01?usp=sharing

GIT HUB ARCHIVOS: https://github.com/JhoelRN/LAB01_SCA/tree/main/%5BLAB%2001%5D%20SCD_PenduloInvertido

7. (Opcional para sobresalientes) Repetir el ítem 4 en python (ayuda: puede usar las librerías scipy y matplotlib).

5. Cuestionario

Según los resultados obtenidos en la primera experiencia del laboratorio se observó que al sistema de péndulo invertido se le introdujo una fuerza que cambió al sistema, y por lo tanto el ángulo con que forma el péndulo y el punto de equilibrio oscilaba sin detenerse ya que se considera que está en un medio que no tiene fricción además se observó que su velocidad angular también cambia de manera oscilante, esto se debe que la velocidad angular es el cambio del angular con respecto al tiempo así que también se obtendrían oscilaciones. Con respecto a la posición esta debería de seguir aumentando como una recta ya que el medio en el vacío.

6. Conclusiones y observaciones

- Se comprendió el moldeamiento de un sistema no lineal a través del método de variables de estado y saber el comportamiento de cada parámetro que interviene un péndulo invertido.
- El controlador PID es capaz de estabilizar el ángulo del péndulo pero la posición del carro cambia y se desplaza a una velocidad constante por lo cual con este tipo de controlador no se consigue el objetivo planteado. El péndulo invertido es una planta en donde se pueden implementar varios tipos de controladores, y discriminar las ventajas de uno contra otro, como en los tiempos de respuesta, si bien uno de los controladores MPC tiene un tiempo mayor para estabilizarse no quiere decir que no sea el mejor ya que al no forzar un cambio brusco al carro mantiene sus componentes haciéndolo más robustos en comparación al resto de controladores propuestos PID y LQR.

Referencias

- [1] L. J. García, L. Ramírez, X. Siordia, and T. Matínez. Las leyes de newton en el modelado y control del pendulo invertido sobre un carro. *Revista Tecnología e Innovación*, 2016.
- [2] D. Jaramillo. Estabilización de un péndulo invertido aplicando mpc y lqr. *Revista Politécnica*, 34, 09 2015.
- [3] L. B. Prasad, B. Tyagi, and H. O. Gupta. Modelling and simulation for optimal control of nonlinear inverted pendulum dynamical system using pid controller and lqr. In *2012 Sixth Asia Modelling Symposium*, pages 138–143. IEEE, 2012.

Apéndice

Rúbrica

- e1: Identifica y diagnostica problemas y los prioriza de acuerdo a su impacto o relevancia.
- e2: Formula soluciones coherentes y realizables usando normas y estándares apropiados.
- e3: Utiliza las técnicas y metodologías de la ingeniería electrónica para plantear, analizar y resolver problemas de ingeniería.
- e4: Maneja equipos e instrumentos y utiliza software especializado propio del ejercicio profesional.

La tabla 1 refleja la evaluación del estudiante respecto este informe y mediante entrevistas.

Tabla 1: Rúbrica según Resultados del Estudiante

Alumno	e1	e2	e3	e4