

Sistemas de Control II

Trabajo práctico N°2



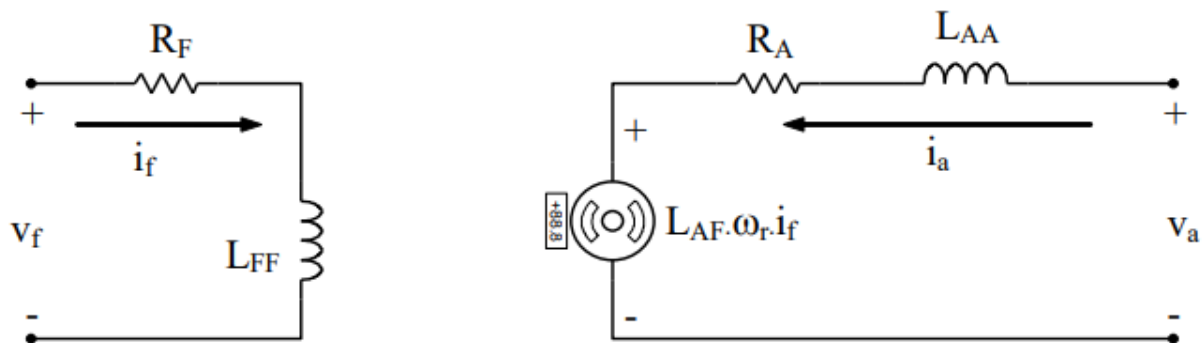
UNIVERSIDAD NACIONAL DE CÓRDOBA
FCEFyN

- Profesor: Dr. Ing. Pucheta, Julian
- Alumno: Valdez Benavidez, Mauricio L.

DISEÑO DE CONTROLADORES EN VARIABLES DE ESTADO EN TIEMPO CONTINUO

CASO DE ESTUDIO 1

SISTEMAS DE TRES VARIABLES DE ESTADO



Se parte del esquemático del circuito que representa a un motor de CC, las ecuaciones con torque TL no nulo y parámetros definidos $L_{AA}=5 \cdot 10^{-3}$; $J=0.004$; $R_A=0.2$; $B=0.005$; $K_i=6,5 \cdot 10^{-5}$; $K_m=0.055$

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

$$\frac{d\omega_r}{dt} = \frac{K_i}{J}i_a - \frac{B_m}{J}\omega_r + \frac{1}{J}T_L$$

$$\frac{d\theta_t}{dt} = \omega_r$$

Item[1] Implementar un sistema en variables de estado que controle el ángulo del motor, para consignas de $\pi/2$ y $-\pi/2$ cambiando cada 2 segundos y que el TL de $1,15 \cdot 10^{-3}$ aparece sólo para $\pi/2$, para $-\pi/2$ es nulo. Hallar el valor de integración Euler adecuado. El objetivo es mejorar la dinámica del controlador que muestra la Fig. 1.

Para hallar el valor de integración de Euler, se planteó la matriz A de estados y se calcularon los auto-valores de la misma. Luego se buscaron los coeficientes que representan a las dinámicas más rápidas y lentas del sistema.

Para determinar el paso o tiempo de integración, se busca el polo que se corresponde con la dinámica más rápida para la cual se llega a un 95% en:

$$t_{95\% \text{ dinamica}} = \frac{\ln(0.95)}{-39.9954} = 0.0013$$

El paso de integración h debe ser al menos 10 veces más chico que el tiempo calculado es decir

$$h = \frac{0.0013}{10} = 1.28 * 10^{-4}$$

Por simplicidad computacional se va a tomar un tiempo de $1 * 10^{-4}$ [s] el paso de integración

```
>> A
A =
   -40.0000   -11.0000    0
    0.0162   -1.2500    0
         0    1.0000    0

>> eig(A)
ans =
         0
   -1.2546
  -39.9954
```

Para determinar el tiempo de simulación para el cual se establece el sistema, se busca el polo que se corresponde con la dinámica más lenta para la cual se llega a un 5% en:

$$t_{5\% \text{ lenta}} = \frac{\ln(0.05)}{-1.2546} = 2.38$$

El tiempo debería ser al menos 5 veces más grande que el tiempo calculado es decir

$$tF = 2.38 * 5 = 12[s]$$

Hay que tener en cuenta que el tiempo anterior calculado es válido para una entrada que no varíe, por lo tanto como la consigna dice que cada 2[s] varíe la entrada, el sistema no llega a establecerse por completo, pero si vemos cómo reacciona al conmutar la entrada.

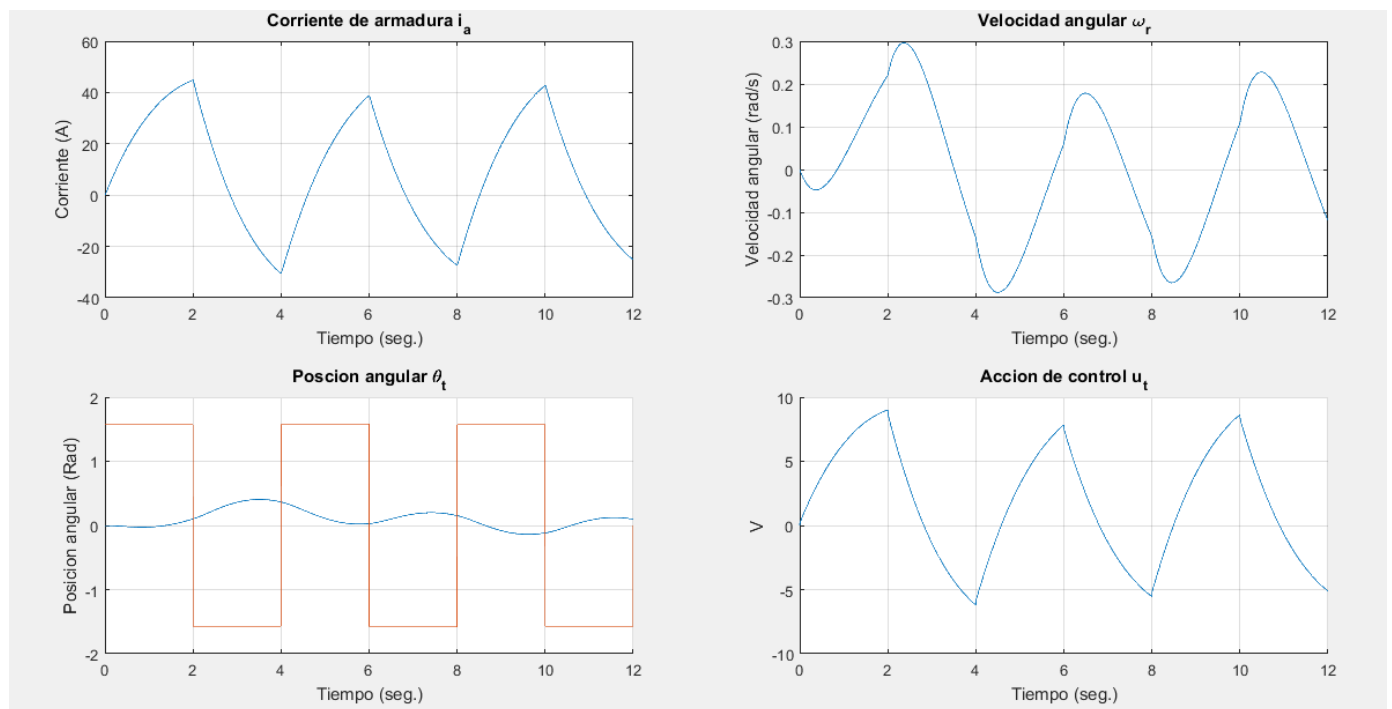
Se optó implementar un controlador LQR para poder obtener mejores resultados respecto al método de asignación de polos que en un principio se comenzó a aplicar a esta consigna. Además luego de bastantes pruebas se hicieron algunas consideraciones que se detallan a continuación:

- Se aumentó el tiempo de conmutación a uno 10 veces más grande, es decir 20[s].
- Se amplió el tiempo de simulación a 100 [s]
- Se agregó un integrador.
- Se colocó un saturador para la acción de control en +-12[V]

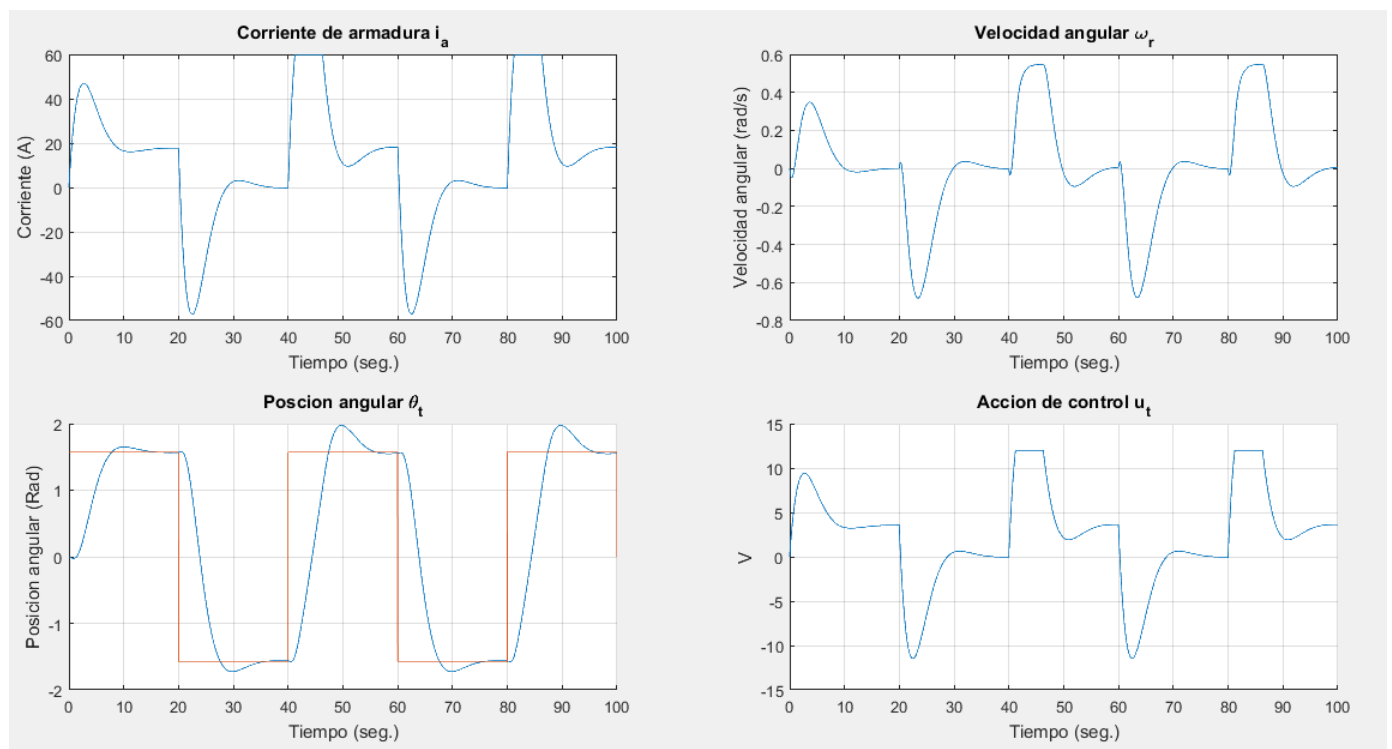
Los cambios mencionados se hicieron ya que para que el sistema pudiera seguir la referencia en los tiempos solicitados y con el TL asignado, la corriente de armadura se disparaba a valores del orden de los 1000 [A] o más. Entonces al aumentar el tiempo de conmutación, se le permitió al motor a llegar a la posición de referencia más relajado y con valores de corriente de armadura más aceptables.

El agregado del integrador permitió corregir la salida reduciendo el error que había entre la referencia y la salida misma.

Simulación con $T_{conmutación}=2[s]$



Simulación con $T_{conmutación}=20[s]$



Se observa en la simulación que al aumentar el tiempo de conmutación mejora el comportamiento para valores similares de corriente que en el caso de un tiempo de conmutación inferior.

Script utilizado para esta segunda parte

```
% Sistemas de Control II -FCEFYN-UNC
% Profesor: Dr.Ing. Pucheta, Julian
% Alumno: Valdez Benavidez, Mauricio Luciano
% Tp N° 2 - Caso de estudio 1 -
% Inciso 1
% Implementar un sistema en variables de estado que controle el ángulo
del motor,
% para consignas de  $\pi/2$  y  $-\pi/2$  cambiando cada 2 segundos y que el
% TL de  $1,15 \cdot 10^{-3}$  aparece sólo para  $\pi/2$ , para  $-\pi/2$  es nulo.
% Hallar el valor de integración Euler adecuado.
% El objetivo es mejorar la dinámica del controlador que muestra la
Fig. 1
%%
clc;clear all;close all;
%Parametros de motor
Laa=5e-3;J=0.004;
Ra=0.2;Bm=0.005;
Ki=6.5e-5;
Km=0.055;

%Matrices de estado
A = [-Ra/Laa -Km/Laa 0;Ki/J -Bm/J 0;0 1 0];
B = [1/Laa 0;0 -1/J;0 0]; %considerando el torque
C = [0 0 1]; %salida posicion

%Matrices ampliadas
Aa=[A zeros(3,1); -C 0];
Ba=[B(:,1); 0];
Ca=[C 0];

%LQR
Q=diag([10 1/100 1/100 10000/2.5]); R=10;
%Q=diag([60 .001 20 5000]); R=10;

Kamp=lqr(Aa,Ba,Q,R);

Tf=100;h=1e-4;pasos=Tf/h; t=linspace(0,Tf,pasos);
Ref=linspace(0,0,pasos);
TL=linspace(0,0,pasos);
Tref=pi/2;
Tl=1.15e-3;
tc=20;
ii=0;

for i=1:pasos-1
    ii=ii+h;
    if (ii>=tc)
        ii=0;
        Tref=Tref*-1;
    end
end
```

```

end
Ref(i)=Tref;
if Tref>=0
TL(i)=Tl;
else
    TL(i)=0;
end
end

%condiciones iniciales
ia(1)=0;           %Corriente de armadura x1
theta(1)=0;        %Velocidad angular x2
omega(1)=0;        %Posicion angular x3

estados=[ia(1);
         omega(1);
         theta(1)];

Xop=[0 0 0]';
x=[ia(1) omega(1) theta(1)]';

psi(1)=0;
integracion(1)=psi(1);

for i=1:round(Tf/h)

    psi_p=Ref(i)-C*estados;
    psi(i)=integracion+psi_p*h;

    u(i) = -Kamp(1:3)*estados-Kamp(4)*psi(i);

    %saturador en +-12V
    if u(i)>12
        u(i)=12;
    end
    if u(i)<-12
        u(i)=-12;
    end

    %Variables del sistema lineal
    ia(i)= x(1);
    omega(i)= x(2);
    theta(i)= x(3);

    x1_p=-Ra*x(1)/Laa-Km*x(2)/Laa+u(i)/Laa;
    x2_p=Ki*x(1)/J-Bm*x(2)/J-TL(i)/J;
    x3_p=x(2);

    xp=[x1_p; x2_p; x3_p];
    x=x+h*xp;

    estados=[ia(i);omega(i);theta(i)];

```

```
    integracion=psi(i);
end

figure(2)
subplot(2, 2, 1);
plot(t,ia);
title('Corriente de armadura i_a');
xlabel('Tiempo (seg.)');
ylabel('Corriente (A)');
grid on;

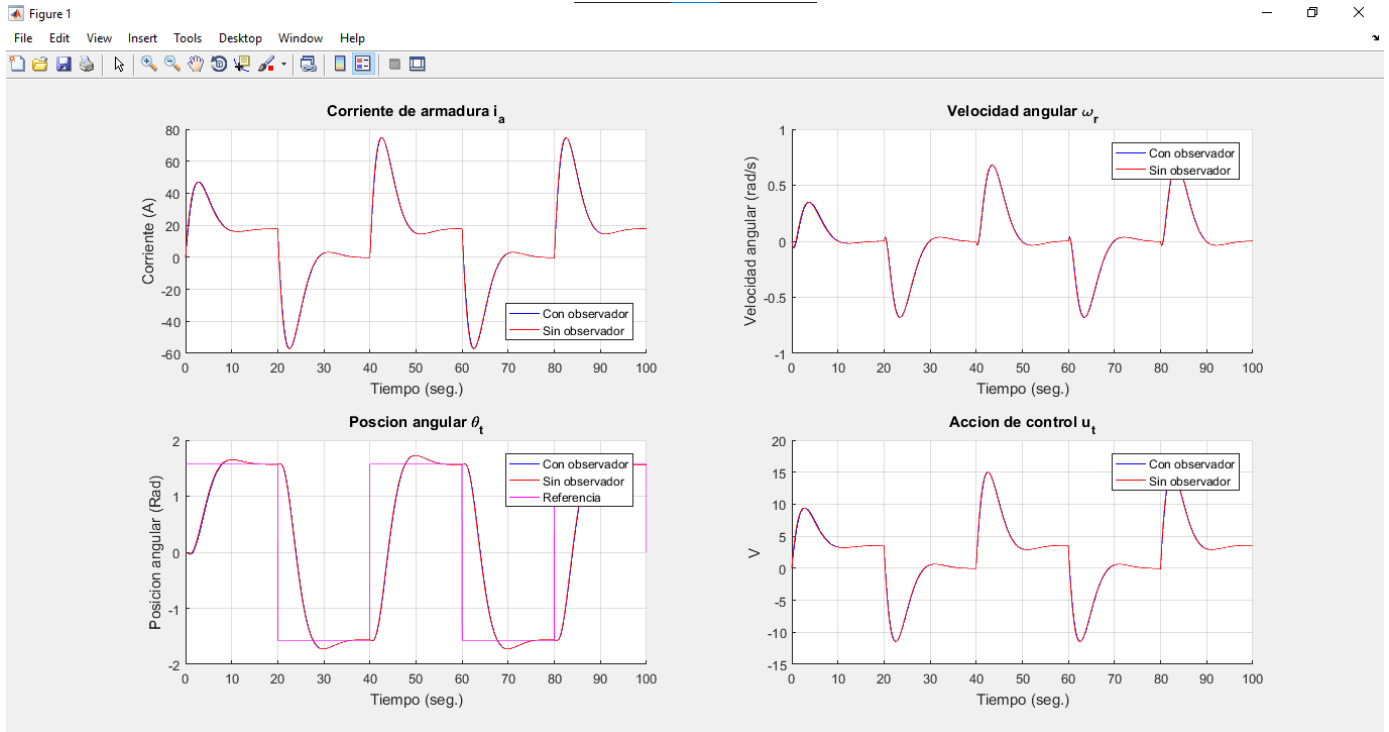
subplot(2, 2, 2);
plot(t,omega);
title('Velocidad angular \omega_r');
xlabel('Tiempo (seg.)');
ylabel('Velocidad angular (rad/s)');
grid on;

subplot(2, 2, 3);
hold on
plot(t,theta);
plot(t,Ref);
hold off
title('Poscion angular \theta_t');
xlabel('Tiempo (seg.)');
ylabel('Posicion angular (Rad)');
grid on;

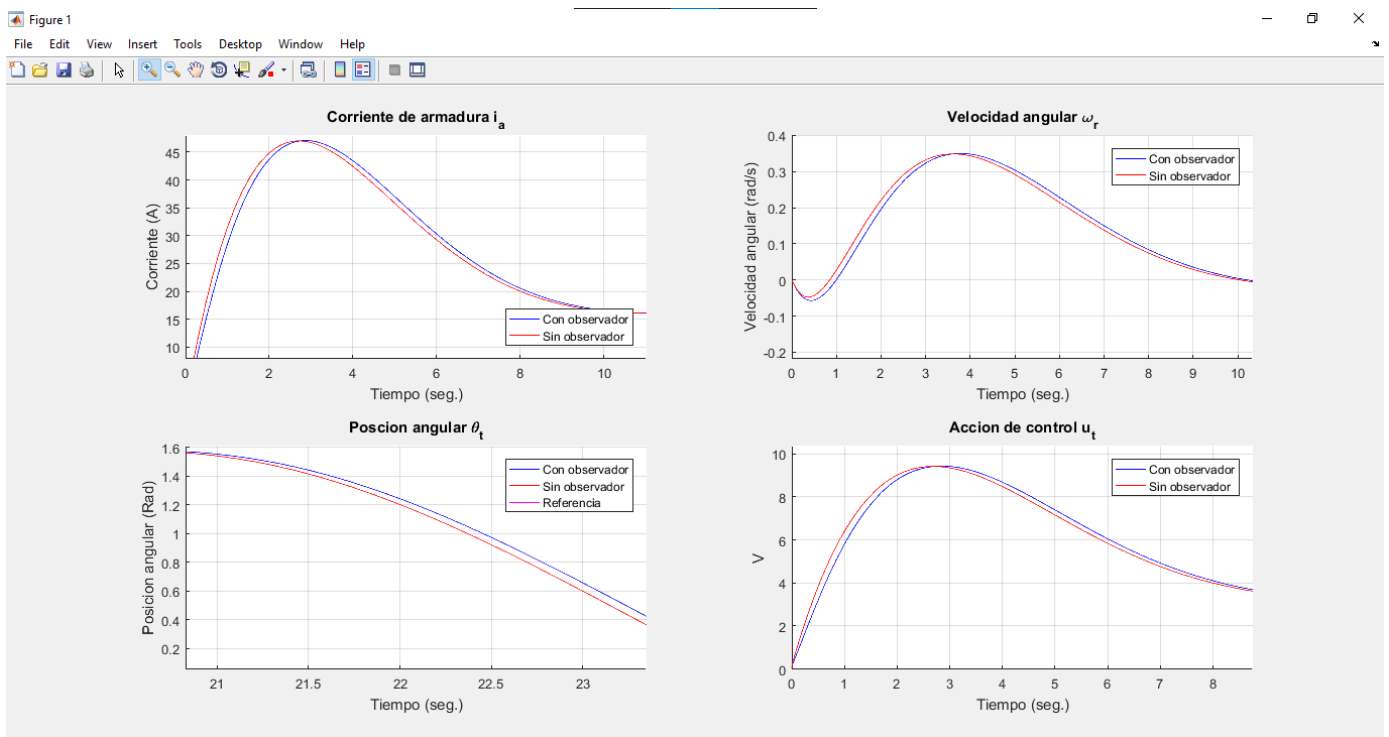
subplot(2, 2, 4);
hold on
plot(t,u);
hold off
title('Accion de control u_t');
xlabel('Tiempo (seg.)');
ylabel('V');
grid on;
```

Ítem [2] Considerar que no puede medirse la corriente y sólo pueda medirse el ángulo, por lo que debe implementarse un observador. Obtener la simulación en las mismas condiciones que en el punto anterior, y superponer las gráficas para comparar.

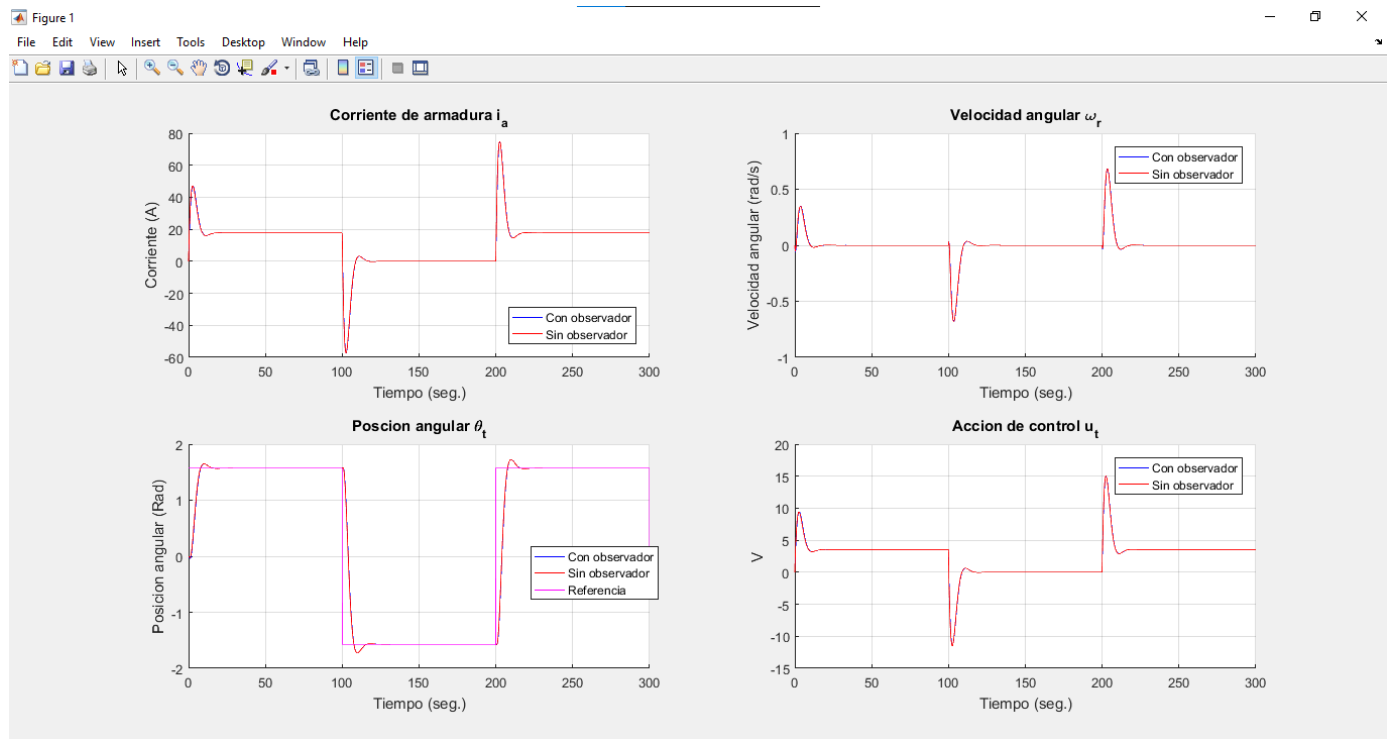
Al no poder medirse la corriente pero si el ángulo, se implementó un observador. Y se comparan las simulaciones con las obtenidas con el controlador LQR calculado anteriormente.



Prácticamente no se pueden individualizar ambas curvas, se hará zoom para ver las diferencias entre ellas en la próxima imagen.



Se probó aumentar el tiempo de conmutación para 100 [s] y un tiempo de simulación a 300[s] para observar la dinámica. Aquí se observa como el motor sigue a la referencia con ese TL indicado. Otra solución hubiese sido disminuir el TL de la consigna a uno menor y encontrar respuestas a menor tiempo y menores picos de corriente.



Script utilizado para esta segunda parte

```
% Sistemas de Control II -FCEfYn-UNC
% Profesor: Dr.Ing. Pucheta, Julian
% Alumno: Valdez Benavidez, Mauricio Luciano
% Tp N° 2 - Caso de estudio 1 -
% Inciso 2
% Considerar que no puede medirse la corriente y sólo pueda medirse el
ángulo,
% por lo que debe implementarse un observador. Obtener la simulación
en las mismas
% condiciones que en el punto anterior, y superponer las gráficas para
comparar.
%
%%

clc;clear all;close all;
%Parametros de motor
Laa=5e-3;J=0.004;
Ra=0.2;Bm=0.005;
Ki=6.5e-5;
Km=0.055;
```

```

%Matrices de estado
A=[-Ra/Laa -Km/Laa 0; Ki/J -Bm/J 0; 0 1 0];
B=[1/Laa; 0; 0];
C=[0 0 1];%salida solo el angulo
D=[0];

%Matrices ampliadas
Aa=[A zeros(3,1); -C 0];
Ba=[B(:,1); 0];
Ca=[C 0];

%LQR
Q=diag([10 1/100 1/100 10000/2.5]); R=10;
Kamp=lqr(Aa,Ba,Q,R);

%Observador-----
Ao=A';
Bo=C';
Co=B';

% Qo=diag([.0001 .1 .01]); Ro=1000;
Qo=1e2*diag([40000 0 0]); Ro=.01;
Ko=lqr(Ao,Bo,Qo,Ro);

Tf=100;h=1e-4;pasos=Tf/h; t=linspace(0,Tf,pasos); %Con Tf=300 y tc=100
Ref=linspace(0,0,pasos); % se logra una mejora
TL=linspace(0,0,pasos); % en la respuesta
Tref=pi/2;
Tl=1.15e-3;
tc=20;
ii=0;

for i=1:pasos-1
    ii=ii+h;
    if (ii>=tc)
        ii=0;
        Tref=Tref*-1;
    end
    Ref(i)=Tref;
    if Tref>=0
        TL(i)=Tl;
    else
        TL(i)=0;
    end
end

%condiciones iniciales
ia(1)=0; %Corriente de armadura x1
theta(1)=0; %Velocidad angular x2
omega(1)=0; %Posicion angular x3

```

```

estados=[ia(1); omega(1); theta(1)];
estados_obs=[ia(1);omega(1);theta(1)];

x=[ia(1) omega(1) theta(1)]';
xobs=[0 0 0]'; %inicializacion para el observador

psi(1)=0;
integracion(1)=psi(1);

for i=1:round(Tf/h)

    psi_p=Ref(i)-Ca(1:3)*estados-Ca(4)*integracion;
    psi(i)=integracion+psi_p*h;

    u(i) = -Kamp(1:3)*estados_obs-Kamp(4)*psi(i);
    %Variables del sistema lineal
    ia(i)= x(1);
    omega(i)= x(2);
    theta(i)= x(3);

    x1_p=-Ra*x(1)/Laa-Km*x(2)/Laa+u(i)/Laa;
    x2_p=Ki*x(1)/J-Bm*x(2)/J-TL(i)/J;
    x3_p=x(2);

    xp=[x1_p; x2_p; x3_p];
    x=x+h*xp;

    %-----con Observador-----

    ia_0(i)= xobs(1);
    omega_0(i)= xobs(2);
    theta_0(i)= xobs(3);

    y_sal_o(i) = C * estados_obs;
    y_sal(i) = Ca(1:3) * estados + Ca(4)*integracion;
    x_hat_p = A*xobs+B*u(i)+Ko*(y_sal(:,i)-y_sal_o(:,i));
    xobs = xobs + x_hat_p*h;

    estados=[ia(i);omega(i);theta(i)];
    integracion=psi(i);
    estados_obs=[ia_0(i);omega_0(i);theta_0(i)];

end

%-----Sistema sin observador-----

%condiciones iniciales
ia_so(1)=0; %Corriente de armadura x1
theta_so(1)=0; %Velocidad angular x2
omega_so(1)=0; %Posicion angular x3

estados_so=[ia_so(1);omega_so(1); theta_so(1)];
x_so=[ia_so(1) omega_so(1) theta_so(1)]';

```

```

psi_so(1)=0;
integracion_so(1)=psi_so(1);

for i=1:round(Tf/h)

    psi_p_so=Ref(i)-C*estados_so;
    psi_so(i)=integracion_so+psi_p_so*h;

    u_so(i) = -Kamp(1:3)*estados_so-Kamp(4)*psi_so(i);
    %Variables del sistema lineal
    ia_so(i)= x_so(1);
    omega_so(i)= x_so(2);
    theta_so(i)= x_so(3);

    x1_p=-Ra*x_so(1)/Laa-Km*x_so(2)/Laa+u_so(i)/Laa;
    x2_p=Ki*x_so(1)/J-Bm*x_so(2)/J-TL(i)/J;
    x3_p=x_so(2);

    xp_so=[x1_p; x2_p; x3_p];
    x_so=x_so+h*xp_so;

    estados_so=[ia_so(i);omega_so(i);theta_so(i)];
    integracion_so=psi_so(i);
end

%-----

subplot(2, 2, 1);
hold on
plot(t,ia,'b');
plot(t,ia_so,'r');
hold off
legend({'Con observador','Sin observador'},'Location','southeast')
title('Corriente de armadura i_a');
xlabel('Tiempo (seg.)');
ylabel('Corriente (A)');
grid on;

subplot(2, 2, 2);
hold on
plot(t,omega,'b');
plot(t,omega_so,'r');
hold off
legend({'Con observador','Sin observador'})
title('Velocidad angular \omega_r');
xlabel('Tiempo (seg.)');
ylabel('Velocidad angular (rad/s)');
grid on;

subplot(2, 2, 3);
hold on
plot(t,theta,'b');
plot(t,theta_so,'r');
plot(t,Ref,'m');

```

```
hold off
legend({'Con observador','Sin observador','Referencia'})
title('Poscion angular \theta_t');
xlabel('Tiempo (seg.)');
ylabel('Posicion angular (Rad)');
grid on;

subplot(2, 2, 4);
hold on
plot(t,u,'b');
plot(t,u_so,'r');
hold off
legend({'Con observador','Sin observador'})
title('Accion de control u_t');
xlabel('Tiempo (seg.)');
ylabel('V');
grid on;
```

Lecciones aprendidas y problemas que se presentaron

Identificadores de logros:

- ID 1.1.1.1: Se analizó la factibilidad de controlar un proceso a partir de su modelo matemático.
- ID 1.1.1.2: Expresar el modelo lineal multivariable de un proceso real en un punto de operación
- ID 1.1.1.4: Seleccionar las variables de estado de un proceso lineal para generar una expresión matricial lineal.
- ID 1.1.1.5: Inferir en la evolución temporal de procesos reales representados en variables de estado.
- ID 1.1.1.6: Diseñar observadores de procesos linealizados para controladores cuando no es posible medir sus variables.
- ID 1.1.1.7: Diseñar controladores para obtener una dinámica estable del proceso linealizado.

Problemas:

Me encontré con varias dificultades para poder encarar el diseño de los controladores, sobre todo porque es la primera vez que lo intento. Algunas dudas fueron salvadas en clases consultas de años anteriores y con compañeros de este año.

CASO DE ESTUDIO 2

SISTEMAS NO LINEAL DE CUATRO VARIABLES DE ESTADO

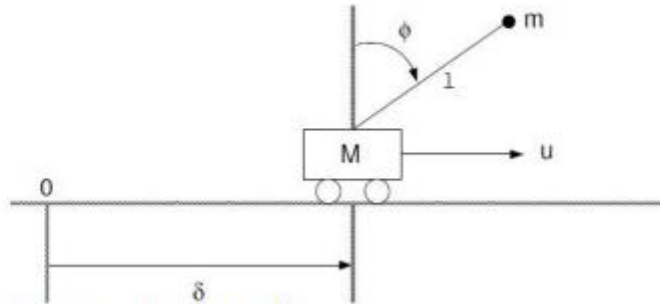


Fig. 2. Sistemas para modelar, extraído de ¹.

Para el caso del esquema del péndulo invertido de la Fig. 2 donde el modelo es,

$$\begin{cases} (M + m)\ddot{\delta} + m l \ddot{\phi} \cos \phi - m l \dot{\phi}^2 \sin \phi + F \dot{\delta} = u \\ l \ddot{\phi} - g \sin \phi + \ddot{\delta} \cos \phi = 0 \end{cases} \quad (4)$$

Con las variables de estado $x = [\delta \quad \dot{\delta} \quad \phi \quad \dot{\phi}]^T$, y los valores de los coeficientes de $m=0,1$; $F=0,1$; $l=1,6$; $g=9,8$; $M=1,5$. Determinar Δt y el tiempo de simulación adecuados.

Ítem [3] Calcular un controlador que haga evolucionar al péndulo en el equilibrio inestable, partiendo de una condición inicial nula en el desplazamiento y termine en -10 metros manteniendo la vertical. Determinar el ángulo máximo que puede alejarse de la vertical en $t=0$ para que el sistema cumpla el objetivo de control.

Para hallar el valor de integración de Euler, se planteó la matriz A de estados y se calcularon los auto-valores de la misma. Luego se buscaron los coeficientes que representan a las dinámicas más rápidas y lentas del sistema.

Para determinar el paso o tiempo de integración, se busca el polo que se corresponde con la dinámica más rápida para la cual se llega a un 95% en:

$$t_{95\% \text{ dinamica}} = \frac{\ln(0.95)}{-2.5582} = 0.0201$$

El paso de integración h debe ser al menos 10 veces más chico que el tiempo calculado es decir

$$h = \frac{0.0201}{10} = 2 \cdot 10^{-3} [s]$$

```
>> Mat_A
Mat_A =
    0    1.0000    0    0
    0   -0.0667   -0.6533    0
    0    0    0    1.0000
    0    0.0417    6.5333    0

>> eig(Mat_A)
ans =
    0
   -0.0625
  -2.5582
   2.5540
```

Por simplicidad computacional se va a tomar un tiempo de $1 * 10^{-3}$ [s] el paso de integración

Para determinar el tiempo de simulación para el cual se establece el sistema, se busca el polo que se corresponde con la dinámica más lenta para la cual se llega a un 5% en:

$$t_{5\% \text{ lenta}} = \frac{\ln(0.05)}{-0.0625} = 47.93$$

El tiempo debería ser al menos 5 veces más grande que el tiempo calculado es decir

$$tF = 47.93 * 5 = 240[s]$$

Se va a probar con un tiempo de simulación de 50[s] para agilizar el cálculo computacional y si hace falta se lo extenderá.

Para el cálculo de este controlador se decide utilizar un regulador lineal cuadrático (LQR) cuyos valores son:

$Q = \text{diag}([10 \ 1 \ 10 \ 1])$ dando mayor peso a la posición y al ángulo

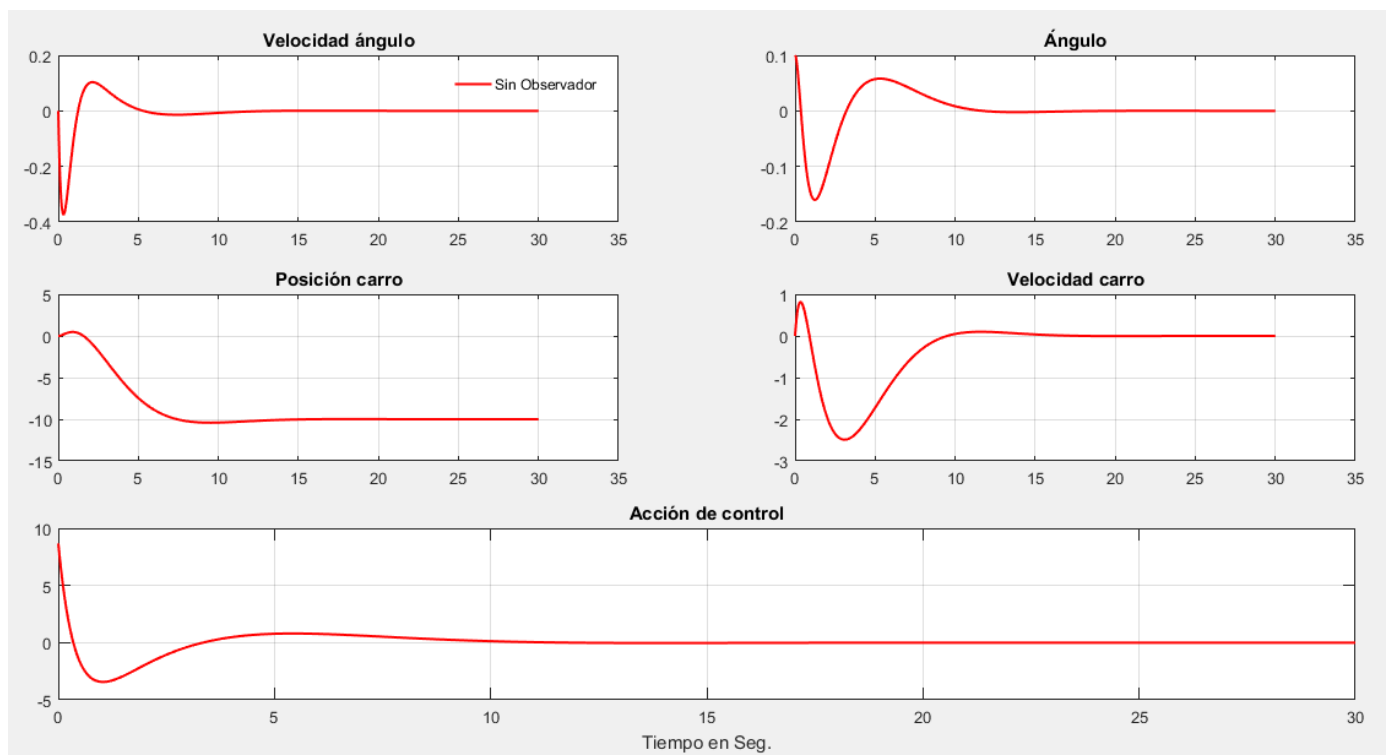
$R = 50$

Y se obtiene un valor de Ganancia K y polos a lazo cerrado:

$$K = \begin{bmatrix} -0.4472 & -1.6605 & -42.0340 & -16.5832 \end{bmatrix}$$

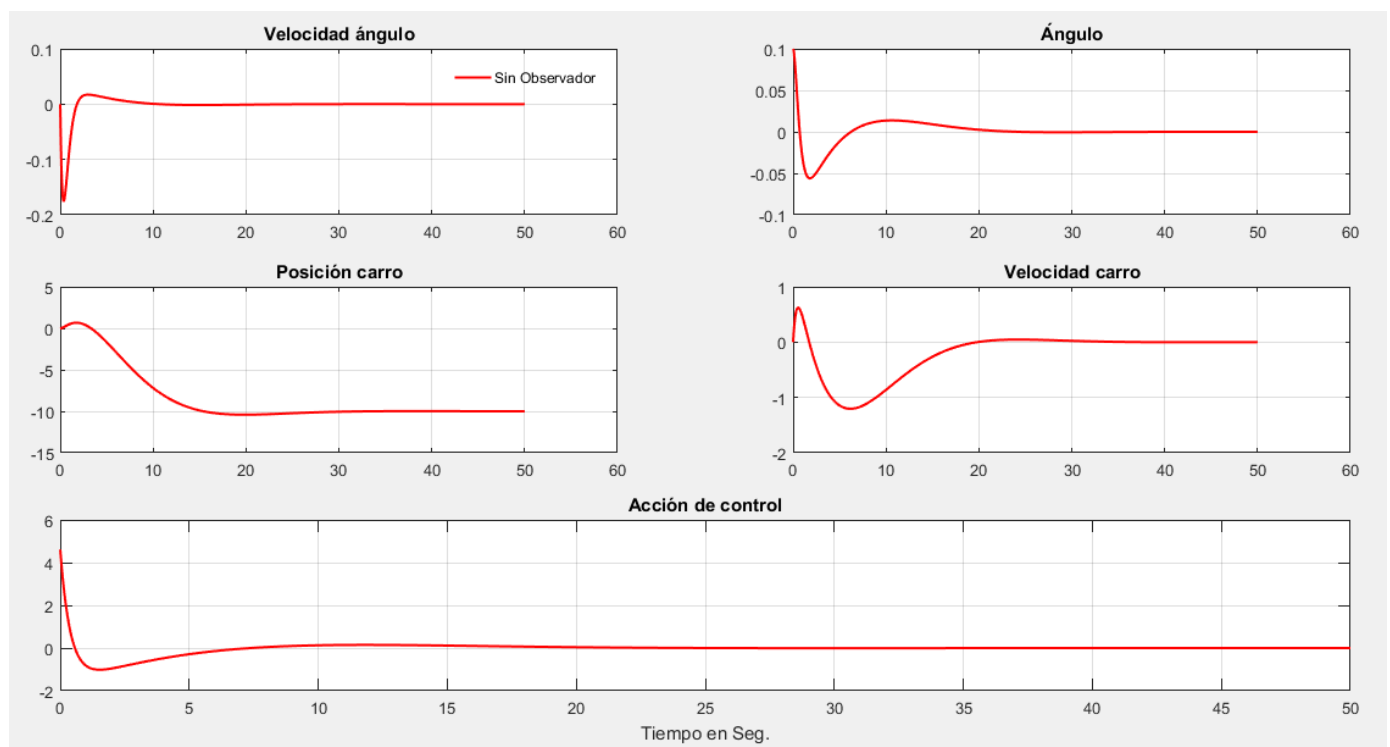
$$P = \begin{bmatrix} -0.3782 + 0.3693i & -0.3782 - 0.3693i & -2.5565 + 0.0215i & -2.5565 - 0.0215i \end{bmatrix}$$

Obteniendo una respuesta con el sistema como sigue:



Se prueba que pasa aumentando el valor de R y se recalcula el controlador y se simula:

$R=1000$

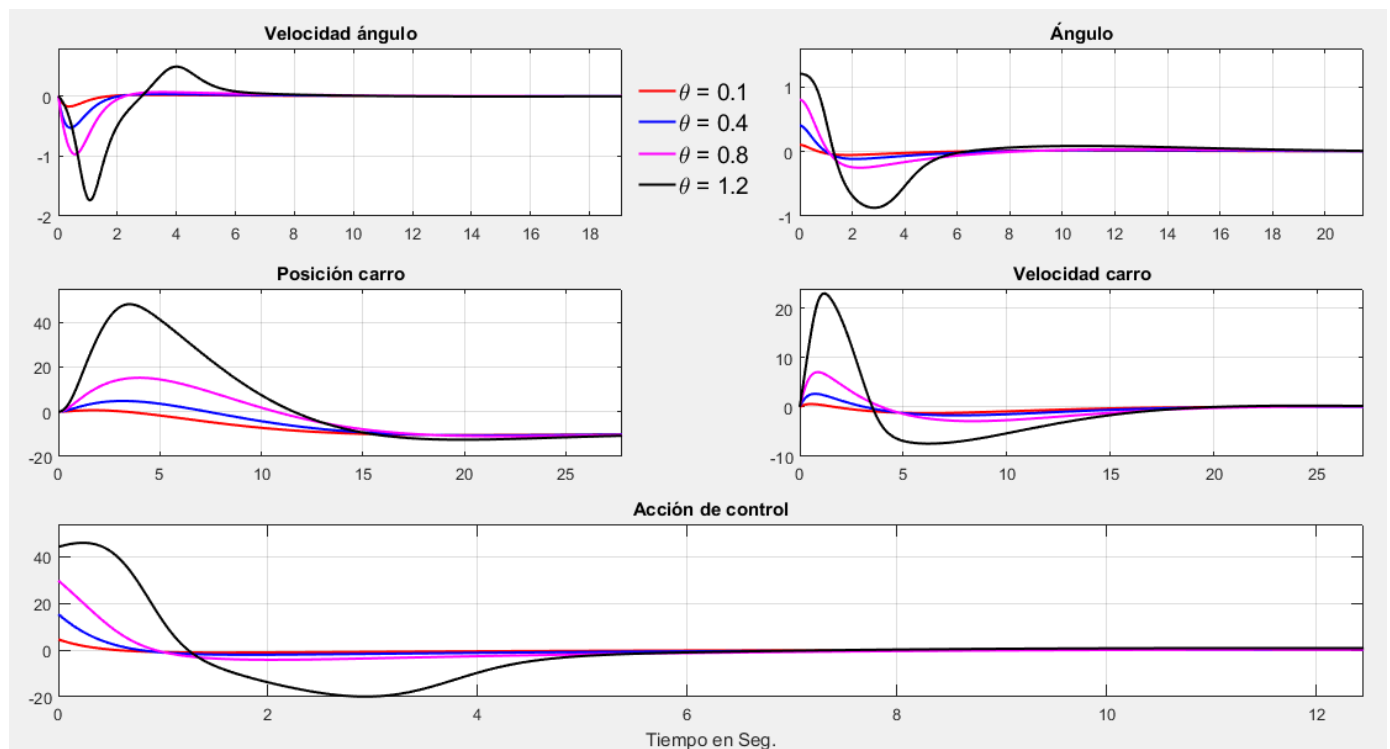


Se observa que el controlador un comportamiento más suave, las acciones de control disminuyen en magnitud y se llega a valores quizás más precisos que los anteriores y con menos energía requerida, sin embargo el tiempo de establecimiento del sistema aumentó, es decir el sistema se volvió más lento infiriendo entonces que si hay variaciones rápidas de referencia, el controlador no va a poder responder con los requerimientos, dejando al sistema ineficiente.

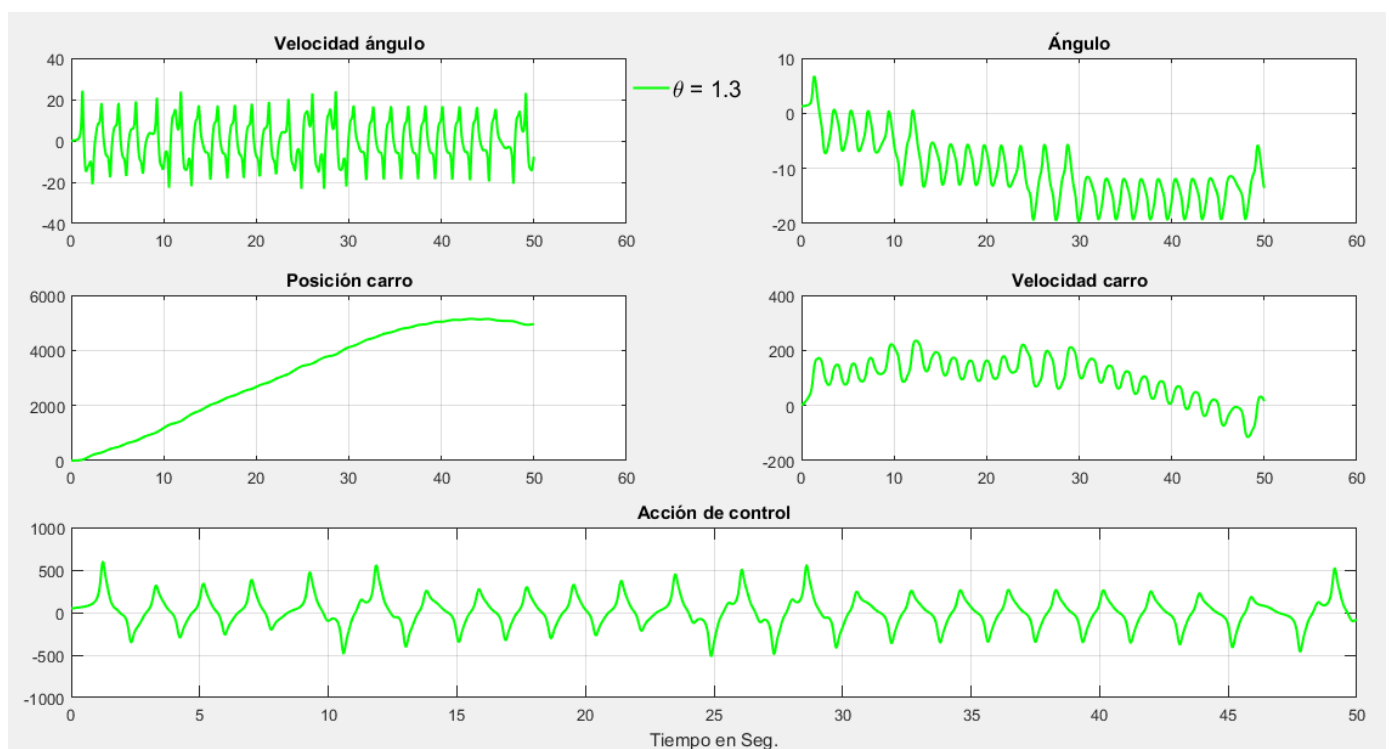
Para el resto del análisis se volvió al valor de $R=50$ y con ese valor se harán las próximas simulaciones.

Para la determinación del ángulo máximo que puede alejarse de la vertical para $t=0$ se fueron probando diferentes valores, hasta que se llegó al valor de 1.2 [rad], pasando ese valor ya la acción de control no puede estabilizar al sistema.

Se detallan graficas del funcionamiento para diferentes valores pequeños de ángulo alejado de la vertical y un caso pasando el valor máximo.



Ahora con 1.3 [rad] de ángulo inicial es decir pasando el máximo establecido, se observa como el controlador intenta controlar el sistema pero ya no puede seguir a la referencia.



Ítem [4] Incorporar un observador para el caso en que sólo puedan medirse el desplazamiento δ y el ángulo ϕ , repetir las simulaciones para las condiciones anteriores y graficar los resultados en gráficas superpuestas.

Se realiza un procedimiento similar, para el observador, pero definiendo nuevas matrices A y B para el observador.

Mat_A_O =

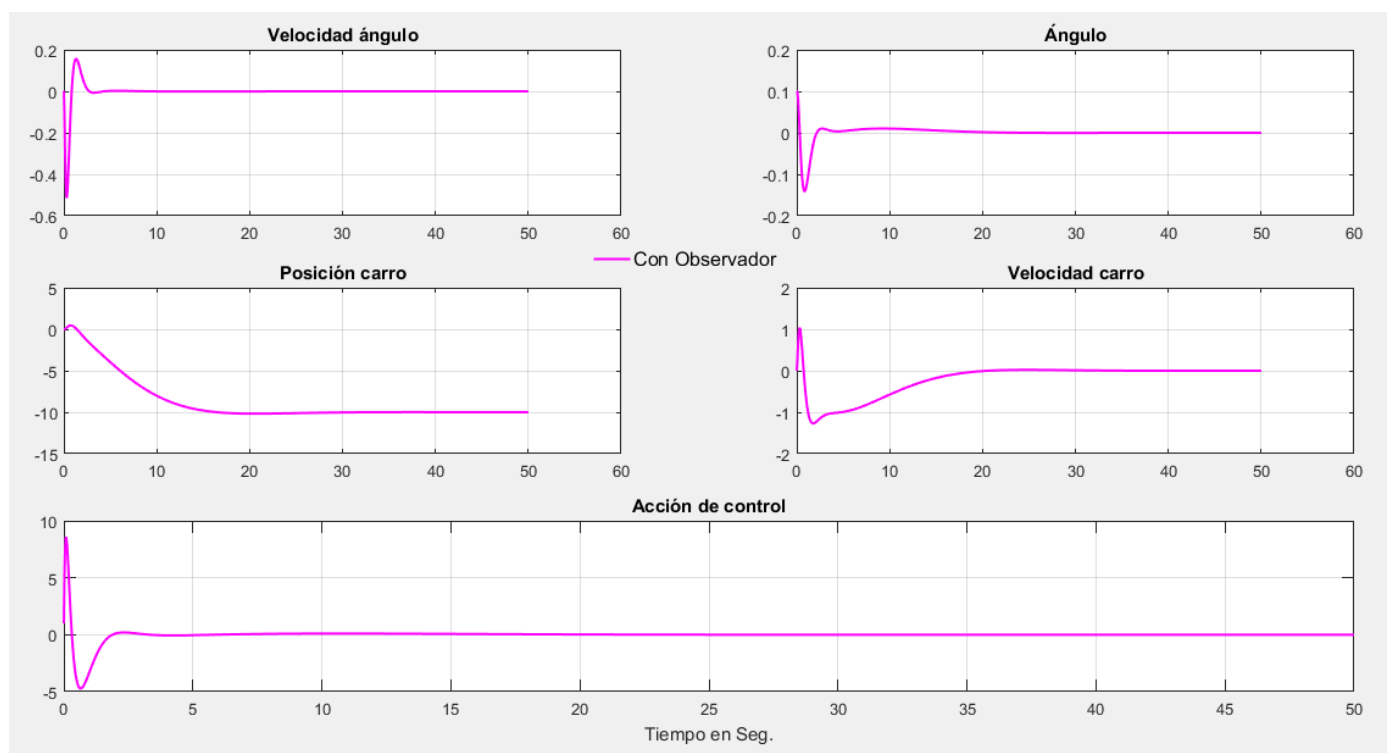
0	1.0000	0	0
0	-0.0667	-0.6533	0
0	0	0	1.0000
0	0.0417	6.5333	0

Mat_B_O =

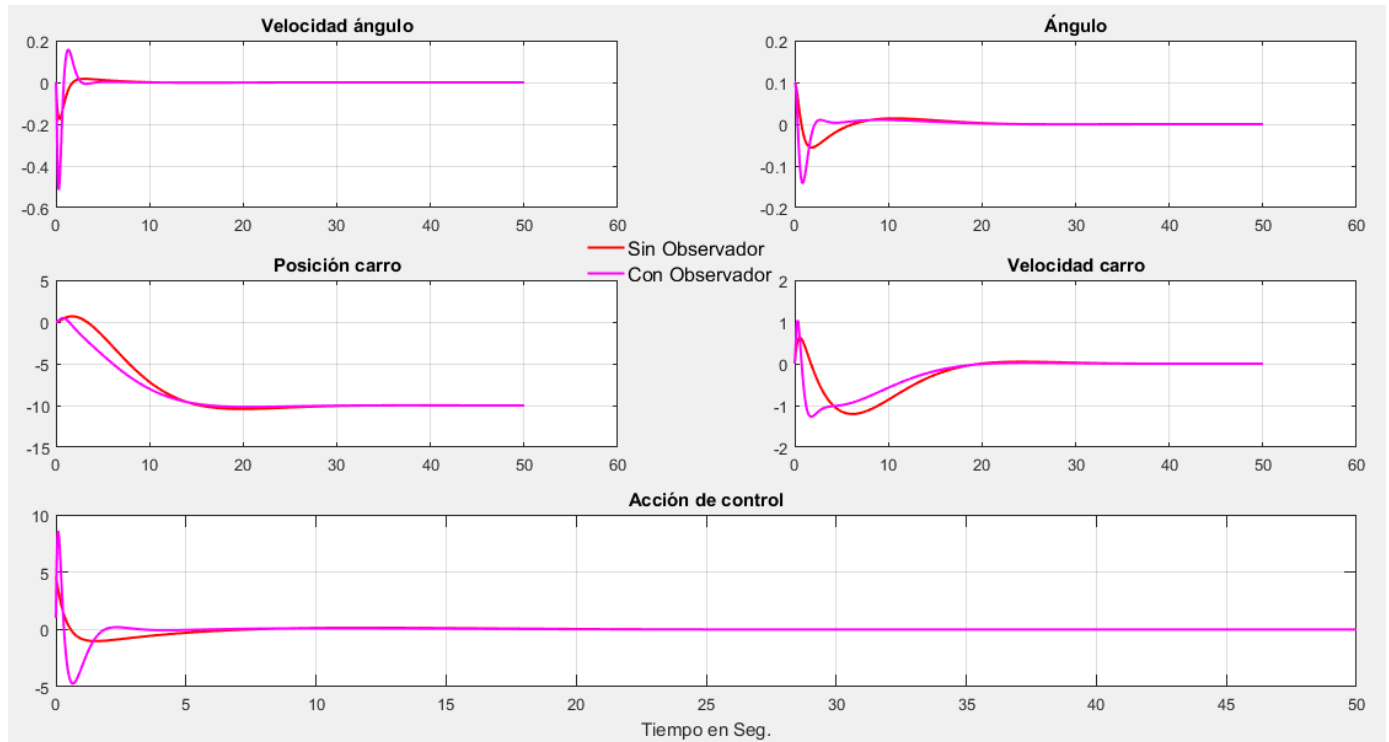
1
0
1
0

$Q_o = \text{diag}([1000 \ 50 \ 1000 \ 50])$ y $R_o = 10$

Quedando entonces la simulación del sistema con Observador



Superponiendo ambas salidas, se ve que tienen comportamientos similares y responden bien.



Script utilizado para esta primera parte

```
% Sistemas de Control II -FCEfYn-UNC
% Profesor: Dr.Ing. Pucheta, Julian
% Alumno: Valdez Benavidez, Mauricio Luciano
% Tp N° 2 - Caso de estudio 2 -
% Inciso 1
% Calcular un controlador que haga evolucionar al péndulo en el
equilibrio inestable,
% partiendo de una condición inicial nula en el desplazamiento y
termine en -10 metros manteniendo
% la vertical. Determinar el ángulo máximo que puede alejarse de la
vertical en t=0 para que el sistema
% cumpla el objetivo de control.
% Inciso2
% Incorporar un observador para el caso en que sólo puedan medirse
el desplazamiento y el ángulo,
% repetir las simulaciones para las condiciones anteriores y grafi-
car los resultados en
% gráficas superpuestas.
%%
close all; clc; clear all;
m=.1;Fricc=0.1; long=1.6;g=9.8;M=1.5;
h=0.001;tiempo=(50/h);p_pp=0;tita_pp=0;
%Condiciones iniciales
alfa(1)=.1; color='r';
% alfa(1)=.4; color='b';
% alfa(1)=.8; color='m';
% alfa(1)=1.2; color='k'; %Angulo inicial máximo para LQR sin obs
```

```

% alfa(1)=1.3; color='g'; %Angulo inicial para la inestabilidad del
sistema para LQR sin obs
omega(1)=0; p_p(1)=0; u(1)=0; p(1)=0; i=1; indice=0;
%Versión linealizada en el equilibrio inestable. Sontag Pp 104.
% estado=[p(i); p_p(i); alfa(i); omega(i)]
Mat_A=[0 1 0 0; 0 -Fricc/M -m*g/M 0; 0 0 0 1; 0 Fricc/(long*M)
g*(m+M)/(long*M) 0];
Mat_B=[0; 1/M; 0; -1/(long*M)];
Mat_C=[1 0 1 0]; %La salida es posición y ángulo

% Mat_M=[Mat_B Mat_A*Mat_B Mat_A^2*Mat_B Mat_A^3*Mat_B]; %Matriz
Controlabilidad
% rank(Mat_M); % verifico el rango de la matriz de controlabilidad

%METODO LQR
Q = diag([10 1 10 1]); % Posicion Carro, Velocidad Carro, Angulo,
Velocidad angular
% Q=Mat_C'*Mat_C;
R = 1000;
[K,S,P] = lqr(Mat_A,Mat_B,Q,R);
G=-inv(Mat_C*(eye(4)-inv(Mat_A+Mat_B*K))*Mat_B); %%Es igual a
Klqr(1)

%OBSERVADOR
Mat_A_O=Mat_A';
Mat_B_O=Mat_C';
Qo = diag([1000 50 1000 50]);
Ro = 10;
[Ko,So,Po] = lqr(Mat_A_O,Mat_B_O,Qo,Ro);
Go=Ko(1); %%Es igual a Klqr(1)
Ko=Ko';
% Mat_M_Dual=[Mat_B_O Mat_A_O*Mat_B_O Mat_A_O^2*Mat_B_O
Mat_A_O^3*Mat_B_O]; %Matriz Controlabilidad
% eig(Mat_A_O'-Ko*Mat_C) %Verifico que todos los polos estén en el
semiplano izquierdo
x_hat=[0;0;0;0]; %Inicializo el Observador

ref=-10; psi(1)=0;

while(i<(tiempo+1))
    estado=[p(i); p_p(i); alfa(i); omega(i)]; % Posicion Carro, Velo-
cidad Carro, Angulo, Velocidad angular
    estado_obs=[p(i); p_p(i); alfa(i); omega(i)];

    u(i)=-K*estado+G*ref; %color='r'; % lqr sin observador
    % u(i)=-K*x_hat+G*ref; color='m'; % con observador

    p_pp=(1/(M+m))*(u(i)-
m*long*tita_pp*cos(alfa(i))+m*long*omega(i)^2*sin(alfa(i))-
Fricc*p_p(i));
    tita_pp=(1/long)*(g*sin(alfa(i))-p_pp*cos(alfa(i)));
    p_p(i+1)=p_p(i)+h*p_pp;
    p(i+1)=p(i)+h*p_p(i);

```

```

omega(i+1)=omega(i)+h*tita_pp;
alfa(i+1)=alfa(i)+h*omega(i);
y_sal(i)=Mat_C*estado;
% OBSERVADOR
y_sal_O(i)=Mat_C*x_hat;
y_sal(i)=Mat_C*estado;
x_hatp=Mat_A*x_hat+Mat_B*u(i)+Ko*(y_sal(i)-y_sal_O(i));
x_hat=x_hat+h*x_hatp;
i=i+1;
end
figure(1);hold on; t=1:i;t=t*h;
subplot(3,2,1);plot(t,omega,color,'LineWidth',1.5);grid on; title('Velocidad ángulo');hold on;
% legend('\theta = 0.1','\theta = 0.4','\theta = 0.8','\theta = 1.2');legend('boxoff');
legend('Sin Observador','Con Observador');legend('boxoff');
subplot(3,2,2);plot(t,alfa,color,'LineWidth',1.5);grid on;title('Ángulo');hold on;
subplot(3,2,3); plot(t,p,color,'LineWidth',1.5);grid on;title('Posición carro');hold on;
subplot(3,2,4);plot(t,p_p,color,'LineWidth',1.5);grid on;title('Velocidad carro');hold on;
subplot(3,1,3);plot(t(1:end-1),u,color,'LineWidth',1.5);grid on;title('Acción de control');xlabel('Tiempo en Seg.');
```

```

% figure(2);hold on;
% subplot(2,2,1);plot(alfa,omega,color);grid on;xlabel('Ángulo');ylabel('Velocidad angular');hold on;
% subplot(2,2,2);plot(p,p_p,color);grid on;xlabel('Posicion carro');ylabel('Velocidad carro');hold on;legend('Sin Observador','Con Observador');legend('boxoff');
```

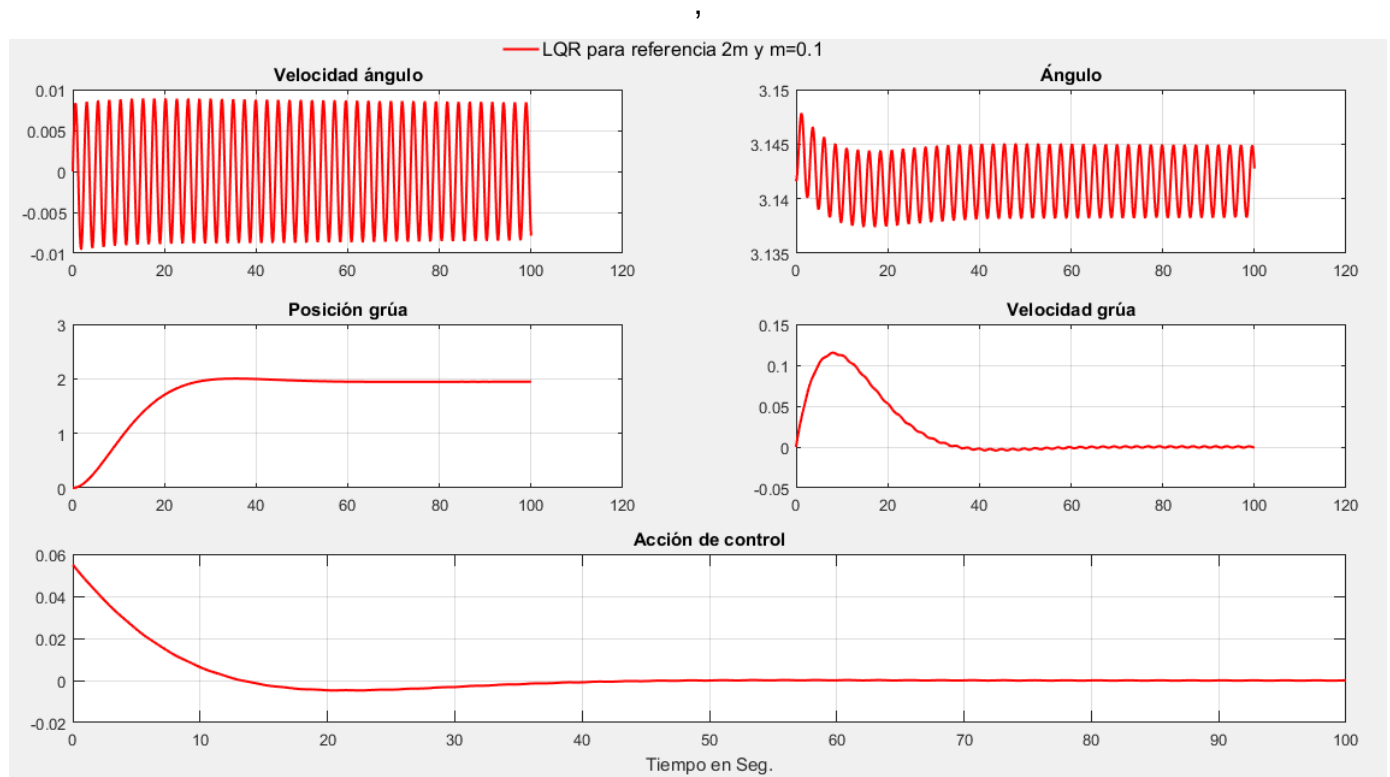
Ítem [5] Calcular un controlador que haga evolucionar al péndulo en el equilibrio estable, partiendo de una condición inicial nula en el desplazamiento y el ángulo en π que termine en 2 metros evitando las oscilaciones de la masa m , considerando que es una grúa. Una vez que $\delta=2$ modificar a m a un valor 10 veces mayor y volver al origen evitando oscilaciones.

Para el siguiente ítem, se diseña un controlador LQR para cuando el sistema traslada la $m=0.1$ y otro cuando el sistema traslada una masa 10 veces mayor.

Para el primer controlador se tiene la siguiente matriz Q y escalar R

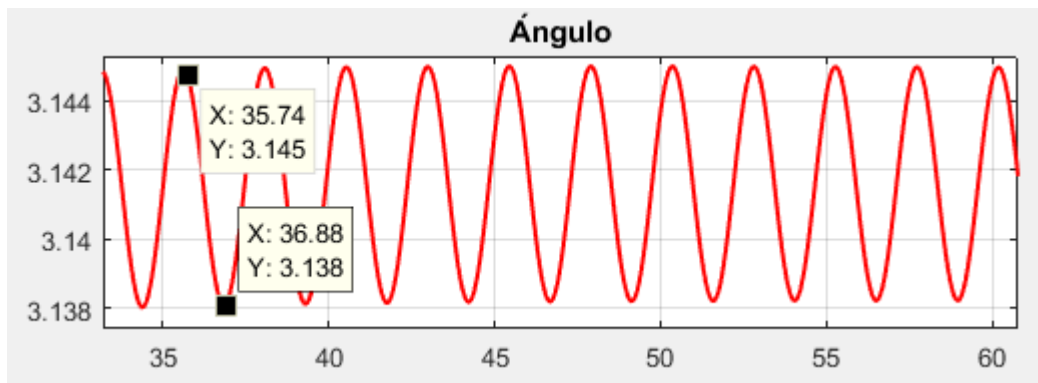
$Q = \text{diag}([1 \ .2 \ .6 \ .2])$; $R = 1000$;

Se eligieron los valores de Q poniendo mayor peso a la posición y el ángulo pero en magnitudes similares. Además se eligió un valor de R alto para suavizar el efecto del controlador en el sistema, es decir a entradas bruscas lo que se busca es q el sistema responda de manera suave ya que no son deseada las oscilaciones y que tampoco se pase demasiado la posición de la grúa respecto a la referencia fijada. El resultado del sistema con ese controlador es el siguiente:

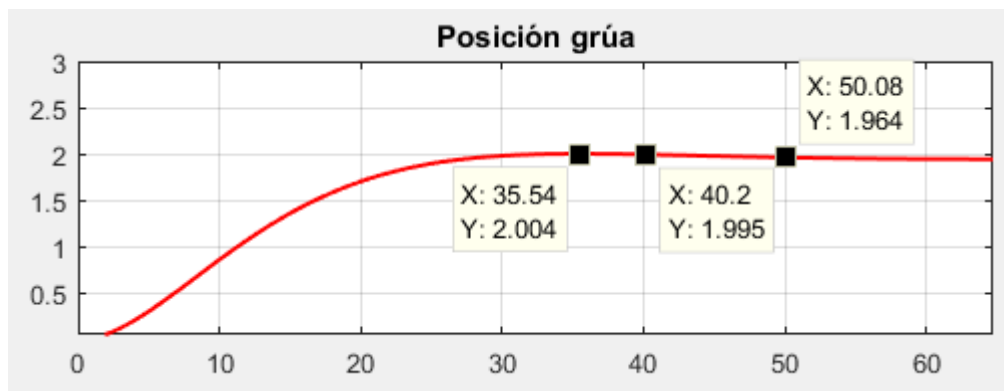


Haciendo zoom al comportamiento del ángulo, vemos que la oscilaciones son mínimas, por lo tanto se puede tomar como aceptable.

Lo mismo ocurre con la posición de la grúa, si bien tiene un sobrepaso de la referencia pero es ínfimo pero se asume que está dentro del rango de precisión de una grúa.



Una oscilación en valor absoluto de $0.007[\text{rad}]$ que se traducen en 0.401°



Se observa que la grúa pasa la posición de referencia por 4 [mm] y se mantiene con un error estable de aproximadamente 4 [cm].

Sin embargo como la consigna pide que una vez que llegue a los 2 metros se cambia la masa y se vuelve a la posición inicial y no especifica un tiempo de permanencia en los 2 [m] se toma como punto de inicio para el siguiente controlador el valor que corresponde para los 40 segundos de simulación.

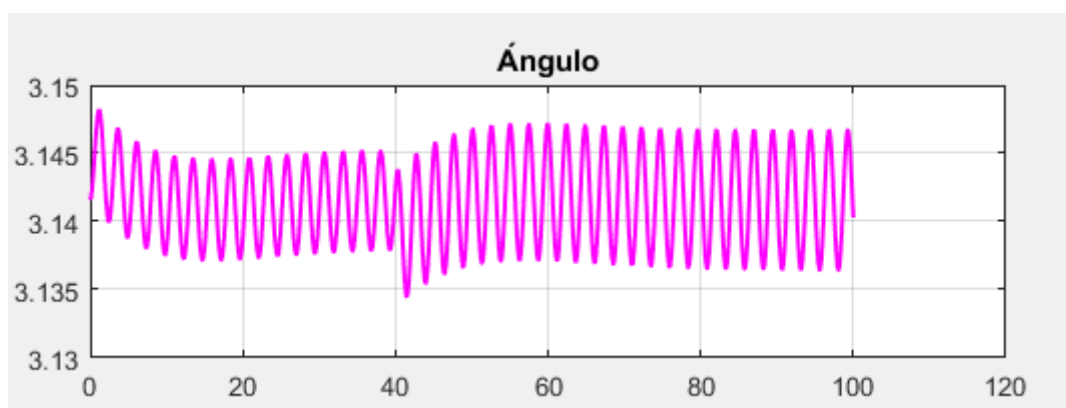
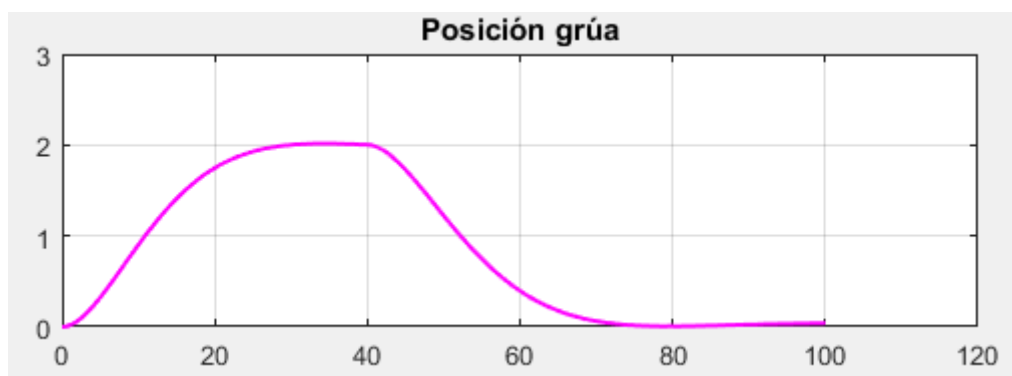
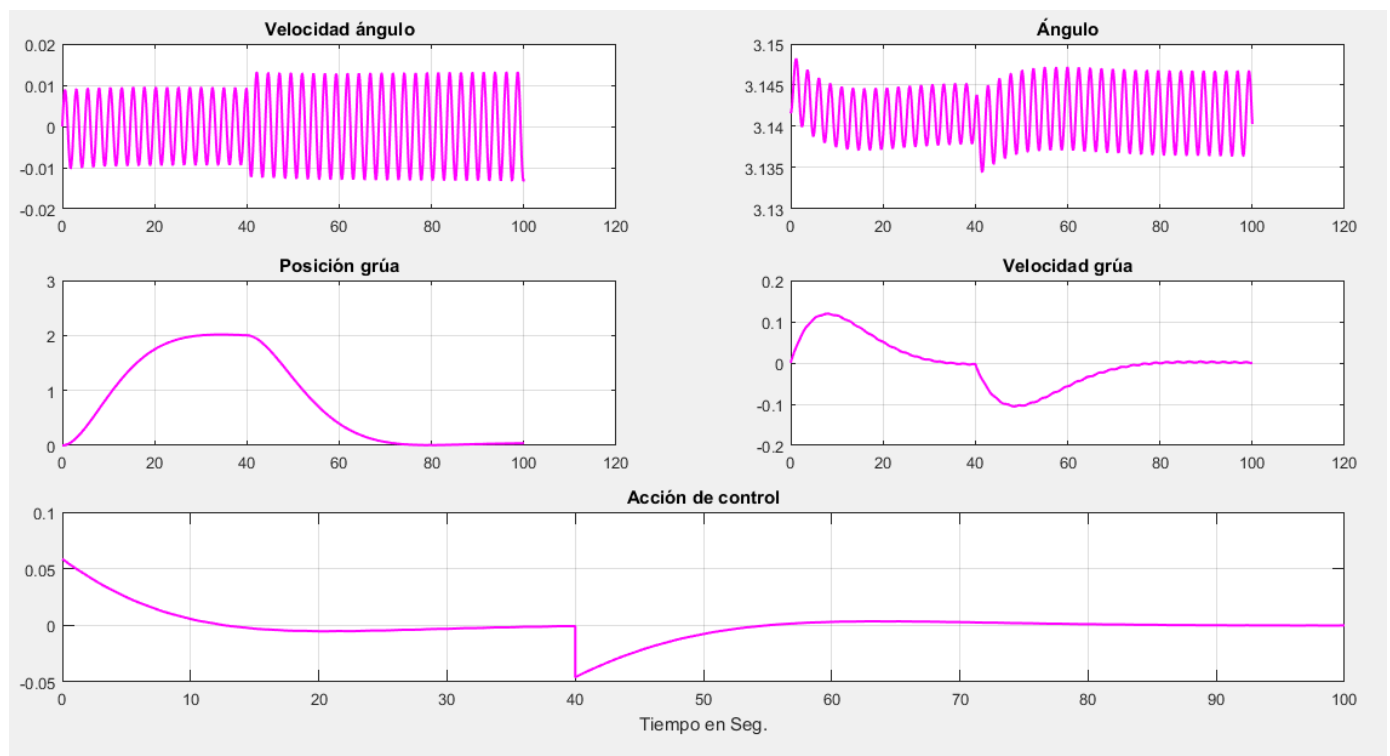
Ahora cuando la m se vuelve 10 veces más grande, el controlador anterior no puede manejarlo al sistema, por lo tanto se hace un cambio de controlador a los 40 segundos.

Este nuevo controlador LQR se lo diseña acompañado de un integrador para que el sistema siga a la referencia de manera precisa, recordemos que con una masa pequeña con acciones pequeñas de control y suaves se podía mover la grúa con cierta precisión. Con esta nueva masa se necesitan movimientos suaves ya que si la masa más grande comienza a oscilar será más difícil de estabilizar pero además una continua integración del error para ir corrigiendo la posición y poder seguir a la nueva referencia.

Estos son los valores para armar el nuevo controlador LQR

```
Q1 = diag([.59 .01 .5 .006 .0001]); R1 = 1000;
```

Se simula la acción de ambos controladores y se observa que el sistema responde bien, si bien las oscilaciones son mayores con la nueva carga, no siguen un comportamiento equivalente con la carga, es decir, la carga es 10 veces más grande pero no así las oscilaciones.

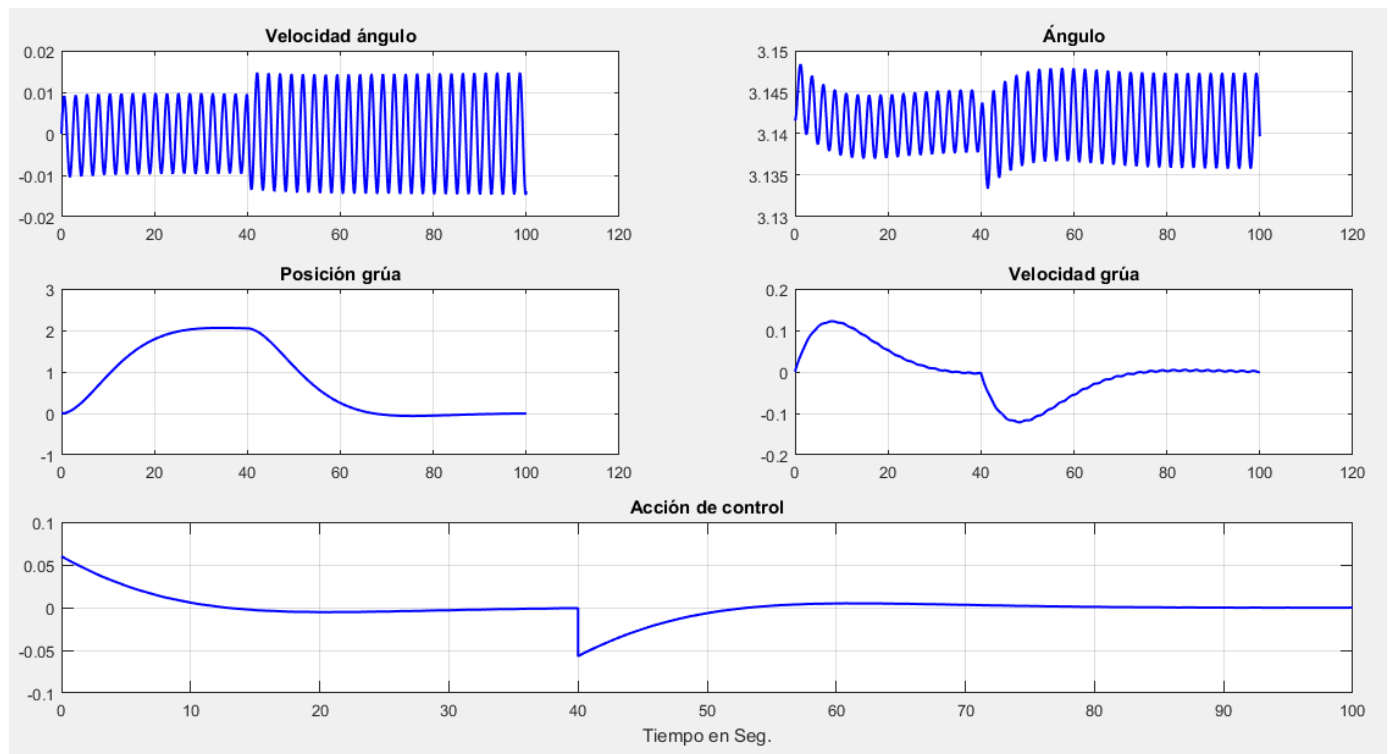


Ítem [6] Incorporar un observador para el caso en que sólo puedan medirse el desplazamiento δ y el ángulo ϕ , repetir las simulaciones para las condiciones anteriores y graficar los resultados en gráficas superpuestas para el equilibrio estable.

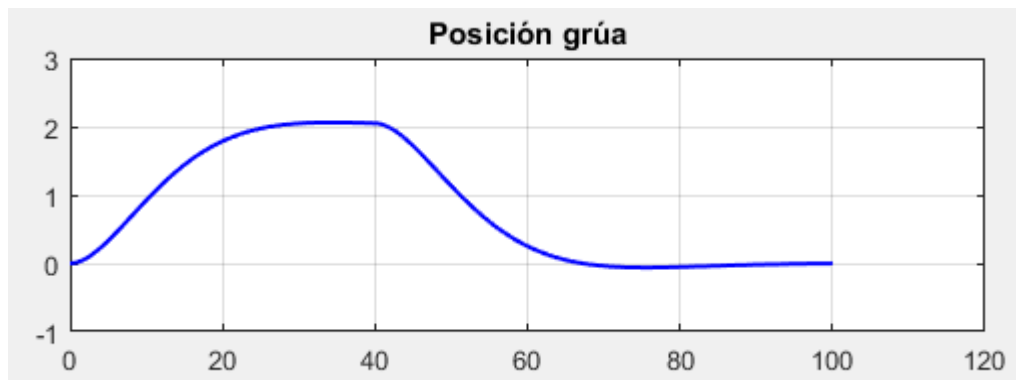
Como se pueden medir el desplazamiento y el ángulo, se modifica la matriz C para poder agregar esa salida y observarla.

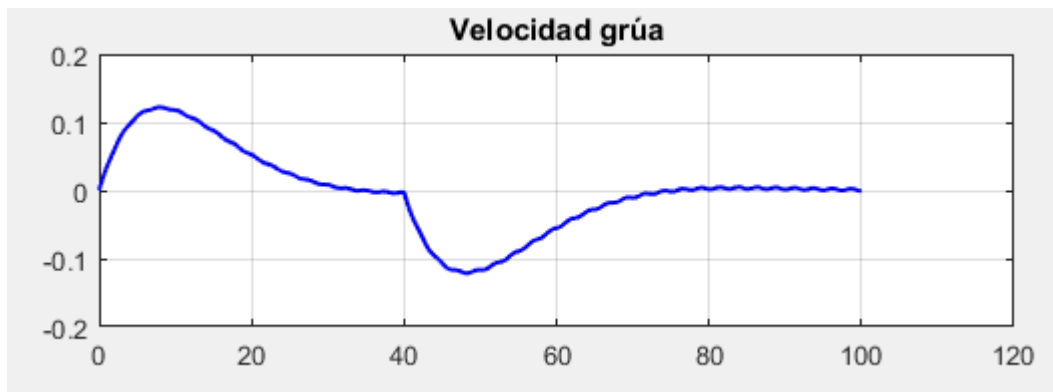
Se arman dos observadores tanto para la masa 0.1 y la masa 1.

La simulación pura con observadores queda:

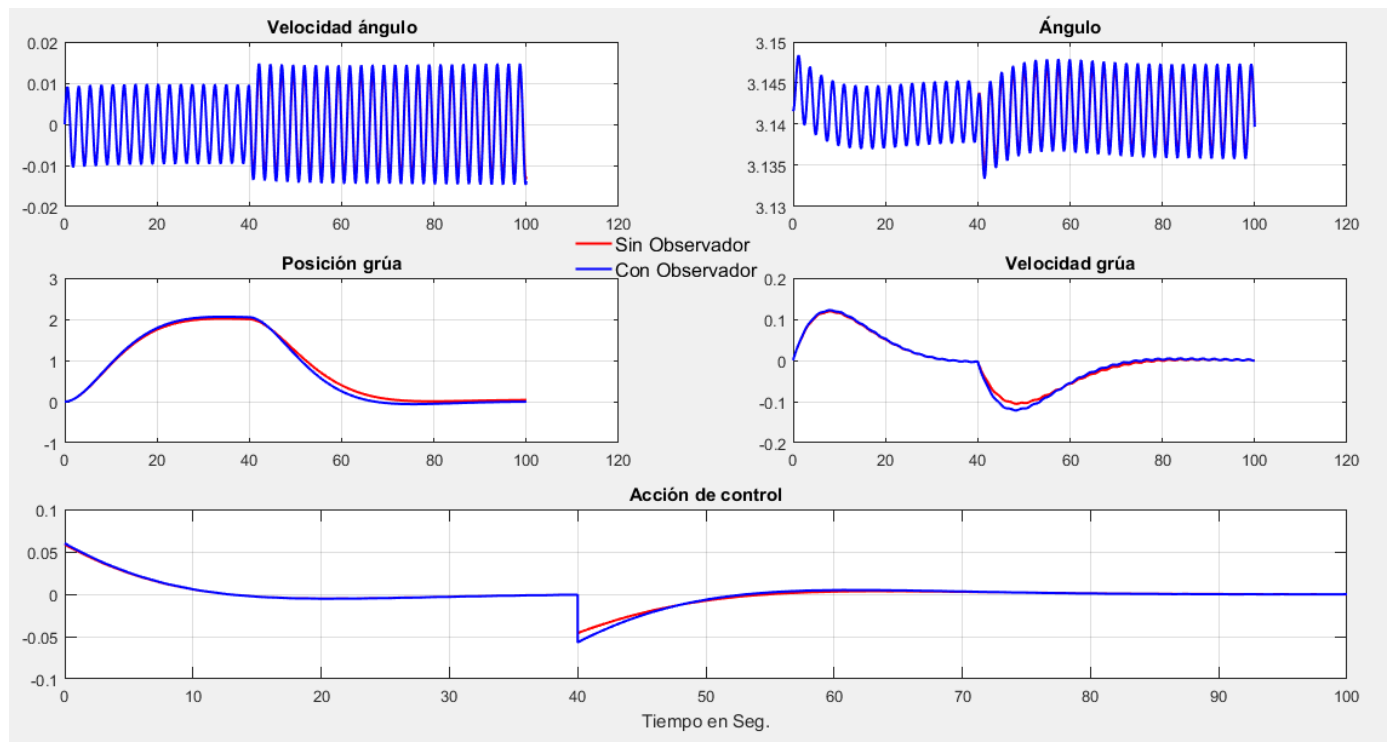


Se consiguen muy buenos resultados rápidamente, sólo hicieron falta algunas iteraciones

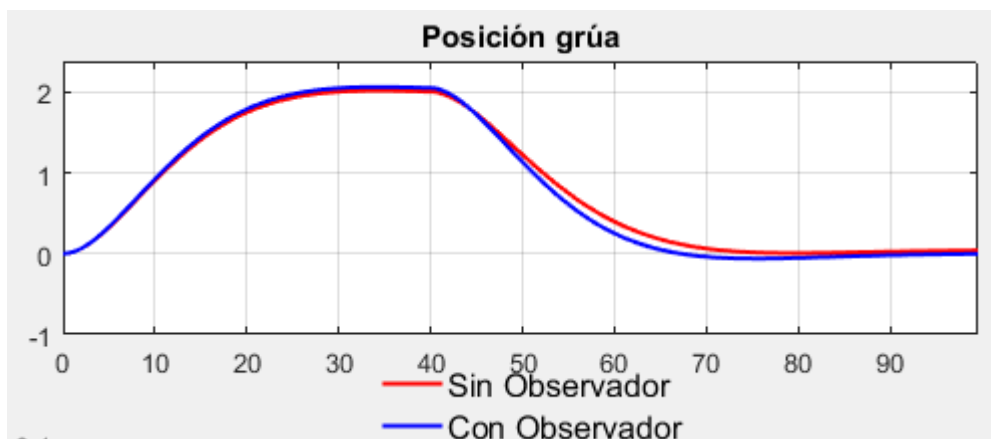


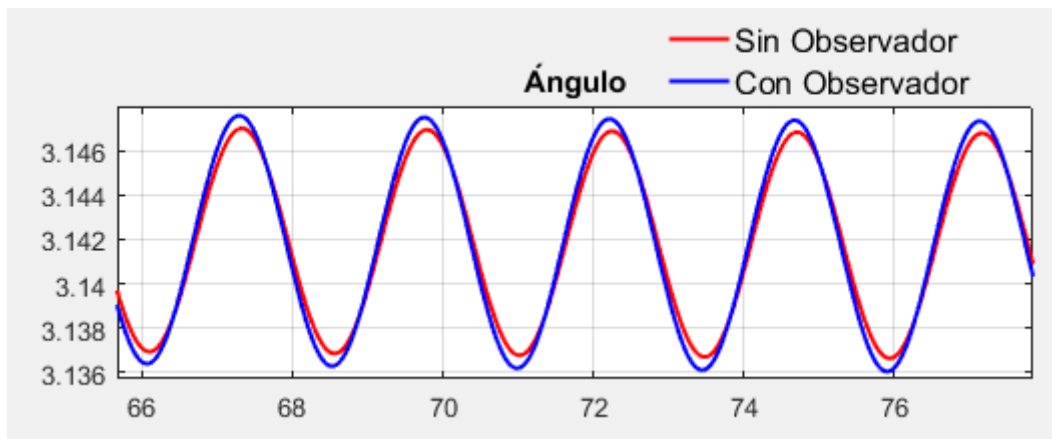


Ahora se comparan ambas salidas, con y sin observador



Se consiguen respuestas muy similares, se hace zoom a la posición y al ángulo para ver con más detalle:





Script utilizado para esta segunda parte

```
% Sistemas de Control II -FCEFYN-UNC
% Profesor: Dr.Ing. Pucheta, Julian
% Alumno: Valdez Benavidez, Mauricio Luciano
% Tp N° 2 - Caso de estudio 2 -
% Inciso 3
% Calcular un controlador que haga evolucionar al péndulo en el equi-
librio estable,
% partiendo de una condición inicial nula en el desplazamiento y
% el ángulo en pi que termine en 2 metros evitando las oscilaciones de
la masa m,
% considerando que es una grúa. Una vez que delta=2 modificar a m a un
% valor 10 veces mayor y volver al origen evitando oscilaciones.
%
% Inciso 4
% Incorporar un observador para el caso en que sólo puedan medirse el
desplazamiento
% delta y el ángulo phi, repetir las simulaciones para las condiciones
anteriores
% y graficar los resultados en gráficas superpuestas para el equili-
brio estable.
%%
% close all; clc; clear all;
m=.1;Fricc=0.1; long=1.6;g=9.8;M=1.5;
h=0.001;tiempo=(100/h);p_pp=0;tita_pp=0;
%Condiciones iniciales
alfa(1)=pi; color='m';

omega(1)=0; p_p(1)=0; u(1)=0; p(1)=0; i=1;
%Versión linealizada en el equilibrio inestable. Sontag Pp 104.
% estado=[p(i); p_p(i); alfa(i); omega(i)]
Mat_A=[0 1 0 0;0 -Fricc/M -m*g/M 0; 0 0 0 1; 0 -Fricc/(long*M) -
g*(m+M)/(long*M) 0];
Mat_B=[0; 1/M; 0; 1/(long*M)];
Mat_C=[1 0 0 0]; %La salida es posición

% Mat_M=[Mat_B Mat_A*Mat_B Mat_A^2*Mat_B Mat_A^3*Mat_B ];%Matriz Con-
trolabilidad
% rank(Mat_M);% verifico el rango de la matriz de controlabilidad
```

```

% %METODO LQR -----Controlador para m=.1 de 0 a 2 metros
Q = diag([.9 .2 .6 .2]);% Posicion Carro, Velocidad Carro, Angulo, Ve-
locidad angular
R = 1000;
[K_m1,S,P] = lqr(Mat_A,Mat_B,Q,R);
G_m1=-inv(Mat_C*(eye(4)-inv(Mat_A+Mat_B*K_m1))*Mat_B);

% -----Controlador para m=1 de 2 a 0 metros
% Necesito agregar un integradorG
Mat_Aa = [Mat_A zeros(4,1);-Mat_C 0];
Mat_Ba = [Mat_B; 0];
Mat_Ca=[Mat_C 0];

% Controlador LQR
Q1 = diag([0.59 .01 .5 .006 .0001]);% Posicion Carro, Velocidad Carro,
Angulo, Velocidad angular
R1 = 1000;
[K_m10,S_m10,P_m10] = lqr(Mat_Aa,Mat_Ba,Q1,R1);
G_m10= K_m10(1);

%OBSERVADOR
%redefino la matriz C para el observador
Mat_Co=[1 0 1 0]; %La salida es posición y ángulo

% % -----Controlador para m=.1 de 0 a 2 metros
Mat_A_O=Mat_A';
Mat_B_O=Mat_Co';
Qo = diag([1 .05 1 .05]);
Ro = 100;
[Ko_m1,So_m1,Po_m1] = lqr(Mat_A_O,Mat_B_O,Qo,Ro);
% Go=Ko_m1(1);
Ko=Ko_m1';
% Mat_M_Dual=[Mat_B_O Mat_A_O*Mat_B_O Mat_A_O^2*Mat_B_O
Mat_A_O^3*Mat_B_O];%Matriz Controlabilidad
% eig(Mat_A_O'-Ko*Mat_C) %Verifico que todos los polos estén en el se-
miplano izquierdo

% % -----Controlador para m=1 de 2 a 0 metros
Mat_A_O=Mat_A';
Mat_B_O=Mat_Co';
Qo_1 = diag([1 .1 1 .1]);
Ro_1 = 100;
[Ko_m10,So_m10,Po_m10] = lqr(Mat_A_O,Mat_B_O,Qo_1,Ro_1);
% Go=Ko_m10(1);
% Ko=Ko_m10';
% Mat_M_Dual=[Mat_B_O Mat_A_O*Mat_B_O Mat_A_O^2*Mat_B_O
Mat_A_O^3*Mat_B_O];%Matriz Controlabilidad
% eig(Mat_A_O'-Ko*Mat_C) %Verifico que todos los polos estén en el se-
miplano izquierdo

x_hat=[0;0;0;0]; %Inicializo el Observador

```

```

ref=2;
psi(1)=0;
K=K_m1; G=G_m1;
while(i<(tiempo+1))
estado = [p(i); p_p(i); alfa(i); omega(i)]; % Posicion Carro, Velocidad Carro, Angulo, Velocidad angular
estado_obs = [p(i); p_p(i); alfa(i); omega(i)];
psi_p = ref - Mat_C* estado;
psi(i+1) = psi(i)+psi_p*h;

    if (i>=40000)
        ref=0;
        K = K_m10(1:4);
        Ki = -K_m10(5);
        G = G_m10;
        Go=Ko_m10(1);
        Ko=Ko_m10';
        u(i)=-K*estado+G*ref+Ki*psi(i+1); color='r' %LQR con integrador , sin observador
    %         u(i)=-K*x_hat+G*ref; color='b';% con observador
    else
        u(i)=-K*estado+G*ref; %color='r';% LQR sin integrador ,sin observador
    %         u(i)=-K*x_hat+G*ref; %color='b';% con observador
    %
    end
%
p_pp=(1/(M+m))*(u(i)-
m*long*tita_pp*cos(alfa(i))+m*long*omega(i)^2*sin(alfa(i))-
Fricc*p_p(i));
tita_pp=(1/long)*(g*sin(alfa(i))-p_pp*cos(alfa(i)));
p_p(i+1)=p_p(i)+h*p_pp;
p(i+1)=p(i)+h*p_p(i);
omega(i+1)=omega(i)+h*tita_pp;
alfa(i+1)=alfa(i)+h*omega(i);
y_sal(i)=Mat_C*estado;
% OBSERVADOR
y_sal_O(i)=Mat_C*x_hat;
y_sal(i)=Mat_C*estado;
x_hatp=Mat_A*x_hat+Mat_B*u(i)+Ko*(y_sal(i)-y_sal_O(i));
x_hat=x_hat+h*x_hatp;
i=i+1;
end
figure(1);hold on; t=1:i;t=t*h;
subplot(3,2,1);plot(t,omega,color,'LineWidth',1.5);grid on; title('Velocidad ángulo');hold on;
% legend('LQR para referencia 2m y m=0.1');legend('boxoff');
legend('Sin Observador','Con Observador');legend('boxoff');
subplot(3,2,2);plot(t,alfa,color,'LineWidth',1.5);grid on;title('Ángulo');hold on;
subplot(3,2,3); plot(t,p,color,'LineWidth',1.5);grid on;title('Posición grúa');hold on;

```

```
subplot(3,2,4);plot(t,p_p,color,'LineWidth',1.5);grid on;title('Velocidad grúa');hold on;
subplot(3,1,3);plot(t(1:end-1),u,color,'LineWidth',1.5);grid on;title('Acción de control');xlabel('Tiempo en Seg.');
```

```
% figure(2);hold on;
% subplot(2,2,1);plot(alfa,omega,color);grid on;xlabel('Ángulo');ylabel('Velocidad angular');hold on;
% subplot(2,2,2);plot(p,p_p,color);grid on;xlabel('Posicion carro');ylabel('Velocidad carro');hold on;legend('Sin Observador','Con Observador');legend('boxoff');
```

Lecciones aprendidas y problemas que se presentaron

Identificadores de logros:

- ID 1.1.1.1: Se analizó la factibilidad de realizar un controlador para hacer evolucionar en el equilibrio inestables y en el equilibrio estable.
- ID 1.1.1.2: Expresar el modelo lineal multivariable de un proceso real en un punto de operación
- ID 1.1.1.4: Seleccionar las variables de estado de un proceso lineal para generar una expresión matricial lineal.
- ID 1.1.1.5: Al diseñar los controladores mediante el método de “LQR”, se debió tener en cuenta el método directo e indirecto de Lyapunov, por lo que la matriz Q fue definida positiva. Esto es condición suficiente para estabilidad asintótica.
- ID 1.1.1.8: Se diseñó un controlador mediante el método de “LQR”, por lo que se consideró tanto la dinámica del proceso como la magnitud de las acciones de control. Esto se realizó para el equilibrio inestable y para el equilibrio estable.
- ID 1.1.3.4: Como consecuencia de diseñar controladores mediante el método de “LQR”, indirectamente se tiene en cuenta el método directo e indirecto de Lyapunov.

Bibliografía

- Apuntes de las clases teóricas de Sistemas de Control 2 - Dr Ing Pucheta, Julian
- Ingeniería de control moderna, quinta edición – Ogata
- Links de internet :
 - <https://la.mathworks.com/help/index.html>
 - <https://controlautomaticoeducacion.com/analisis-de-sistemas/modelo-de-motor-dc/>
 - <https://www.eng.newcastle.edu.au/~jhb519/teaching/caut2/etc/InvPen/invSS.html>
 - <http://bionanouni.wdfiles.com/local--files/teaching-mt221c-d-horario-2013i/MT221-ControlModerno.pdf>
- Repositorio de GitHub donde están los códigos a disposición del docente ya que fue designado como colaborador.
<https://github.com/mauriciovaldez19/Sistemas-de-Control-II---FCEFYN-UNC>