ESCUELA DE POSGRADO

Curso:

CONTROL NO LINEAL

Tema:

Control Backstepping aplicado a un brazo robot 2DOF

Presentado por:

CONTRERAS MARTINEZ, DIMEL ARTURO

Docente:

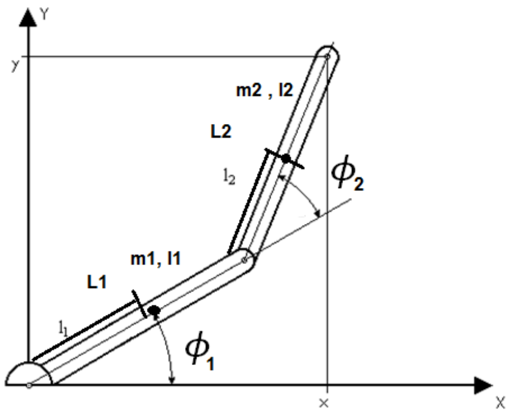
DR. ANTONIO MORÁN

2016

Control movimiento de brazo robótico horizontal de 2DOF

1. Modelamiento

Esquema del robot de 2DOF:



Parámetros:

$m1: masa\ de\ la\ barra\ 1$

$I1: Inercia\ de\ la\ barra\ 1\ respecto\ a\ su\ CM$

$L1 : longitud\ de\ la\ barra\ 1$

$l1 : distancia\ de\ la\ articulación\ \ al\ centro\ de\ masa\ de\ la\ barra\ 1$

$m2: masa\ de\ la\ barra\ 2$

$I2: Inercia\ de\ la\ barra\ 2\ respecto\ a\ su\ CM$

$L2 : longitud\ de\ la\ barra\ 2$

$l2 : distancia\ de\ la\ articulación\ \ al\ centro\ de\ masa\ de\ la\ barra\ 2$

Ecuación dinámica de Lagrange:

$$M_{(\phi)}\ddot{\phi} + C_{(\phi,\dot{\phi})} + G_{\phi}g = ST$$

Como es el robot es horizontal, no se afecta por la gravedad:

$$M_{(\phi)}\ddot{\phi} + C_{(\phi,\dot{\phi})} = ST$$

Siendo:

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$$

$$S = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix}$$

$M11 = I1 + m1 * l1 * l1 + m2 * L1 * L1 + m2 * L1 * l2 * cos(fi2)$

$M12 = m2 * L1 * l2 * cos(fi2)$

$M21 = I2 + m2 * l2 * l2 + m2 * L1 * l2 * cos(fi2)$

$M22 = I2 + m2 * l2 * l2$

$C1 = -m2 * L1 * l2 * (fi1p + fi2p)^2 * sin(fi2)$

$C2 = m2 * L1 * l2 * fi1p * fi1p * sin(fi2)$

2. <u>Control no lineal Backstepping</u>

$$M\ddot{\phi} + C = ST$$

$$\ddot{\phi} = -M^{-1}C + M^{-1}ST$$

Formamos el vector de estados:

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$x_1 = \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix}$$

$$x_2 = \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \end{bmatrix}$$

Formamos el sistema:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -M^{-1}C + M^{-1}ST$$

- cambio de variable

$$x_2 = v$$

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

Para estabilizar:

$$v = -K_1(x_1 - {x_1}^*)$$

$$K_1 = \begin{bmatrix} K_{11} & 0 \\ 0 & K_{12} \end{bmatrix}$$

Siendo $K_{11}\ y\ K_{12} > 0$

- Cambio de variable:

$$z = x_2 - v = x_2 + K_1(x_1 - x_1{}^*)$$

$$\dot{z} = \dot{x}_2 - \dot{v}$$

$$\dot{z} = -M^{-1}C + M^{-1}ST + K_1\dot{x}_1$$

Para estabilizar:

$$\dot{z} = -K_2(z - z^*)$$

Entonces:

$$T = S^{-1}M(M^{-1}C - K_1x_2 - K_2(z - z^*))$$

$$z - z^* = x_2 + K_1(x_1 - x_1{}^*) - x_2{}^*$$

$$z - z^* = x_2 - x_2{}^* + K_1(x_1 - x_1{}^*)$$

Finalmente la ley de control es:

$$T = S^{-1}C - S^{-1}M(K_1x_2 + K_2(x_2 - x_2{}^*) + K_2K_1(x_1 - x_1{}^*))$$

$$x_2{}^* = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Volviendo a las variables originales:

$$T = S^{-1}C - S^{-1}M(K_1\begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \end{bmatrix} + K_2\left(\begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \end{bmatrix}\right) + K_2K_1(\begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} - \begin{bmatrix} \phi_1{}^* \\ \phi_2{}^* \end{bmatrix}))$$

- Código implementado en Matlab

Parámetros de la planta:

```
m1 = 0.15;
L1 = 0.35;
l1 = 0.16;
I1 = 4.1e-3;
m2 = 0.12;
L2 = 0.30;
l2 = 0.12;
I2 = 3.2e-3;

S = [ 1  -1
        0   1 ];
Si = inv(S);
```

Gráfica de la figura a seguir y generación de referencias:

```
dt = 0.0005;
v = input('Velocidad eje x del robot: ');
Xc= 0.525; R=0.125;
ti=0; tf=pi/v;
t=ti:dt:tf; t=t'; nt=length(t);
theta = v*t;
xr = Xc + R*cos(theta);
yr = R*sin(theta);

x2my2 = xr.^2 + yr.^2;
r1A = acos(xr./sqrt(x2my2));
r1B = acos((x2my2 + L1*L1 - L2*L2)./(2*L1*sqrt(x2my2)));
r2 = acos((x2my2 - (L1*L1 + L2*L2))/(2*L1*L2));
r2 = -r2;
r1 = atan2(yr,xr) + r1B;
r1 = real(r1);
r2 = real(r2);
nr = length(r1);
angast = [ r1  r2 ];
velast = [ zeros(nr,1)   zeros(nr,1) ];
```

Condiciones iniciales

```
fi1  = 0;    fi2  = 0;
fi1p = 0;    fi2p = 0;
ang = [ fi1   fi2 ]';
vel = [ fi1p  fi2p ]';
```

Controlador Pesos:

```
K11 = input('Ganancia K11 (fi1)    [10]: ');
K12 = input('Ganancia K12 (fi2)    [10]: ');
K21 = input('Ganancia K21 (fi1p) [25]: ');
K22 = input('Ganancia K22 (fi2p) [25]: ');
K1 = diag([ K11  K12 ]);
K2 = diag([ K21  K22 ]);
```

Simulación: (***El controlador no lineal se calcula continuamente dentro del bucle***)

```
k = 1;
for tt = ti:dt:tf
   phi1(k,1) = fi1;     phi2(k,1) = fi2;    t(k,1) = tt;
   fi1 = ang(1,1);     fi2 = ang(2,1);
   fi1p = vel(1,1);    fi2p = vel(2,1);
   M11a = I1 + m1*l1*l1 + m2*L1*L1 + m2*L1*l2 * cos(fi2);
   M12a = m2*L1*l2*cos(fi2);
   M21a = I2 + m2*l2*l2 + m2*L1*l2*cos(fi2);
   M22a = I2 + m2*l2*l2;
   M = [ M11a  M12a
            M21a  M22a ];
   C1 = -m2*L1*l2*(fi1p+fi2p)^2*sin(fi2);
   C2 = m2*L1*l2*fi1p*fi1p*sin(fi2);
   C = [ C1
           C2 ];
  T = Si*C - Si*M*((K1+K2*K1)*(ang-angast(k,:)') + K2*(vel-velast(k,:)'));

  T1(k,1) = T(1,1);
  T2(k,1) = T(2,1);

   accel = inv(M)*(-C+S*T);
   ang = ang + dt*vel;
   vel = vel + dt*accel;
   k = k + 1;
end
```

Ploteo de Resultados:

```
xx = L1*cos(phi1) + L2*cos(phi1 + phi2);
yy = L1*sin(phi1) + L2*sin(phi1 + phi2);

phi1 = phi1*180/pi;
phi2 = phi2*180/pi;
r1g = r1*180/pi;
r2g = r2*180/pi;

figure(1);
subplot(2,1,1);  plot(t,phi1,t,r1g);
legend('phi1 real','phi1 est')
subplot(2,1,2);  plot(t,phi2,t,r2g);
legend('phi2 real','phi2 est')

figure(2);
plot(t,T1,t,T2);   title('Torques')
legend('T1','T2')

figure(3);
plot(xx,yy,xr,yr);
legend('referencia','real')
```

Evaluando la precisión de la trayectoria realizada por el robot:

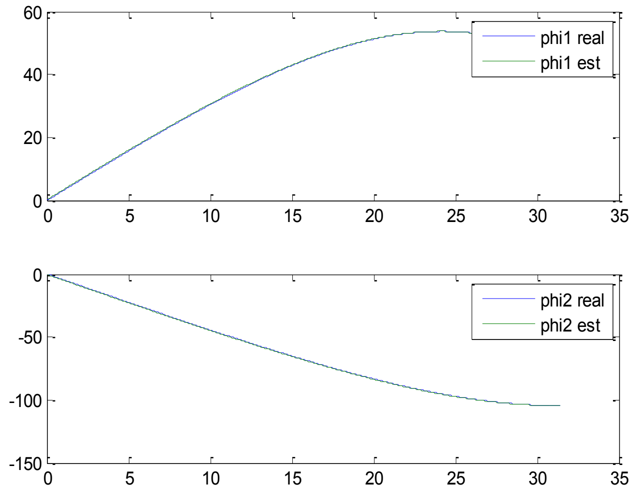
Error máximo y Desempeño:

```
error=sqrt((xr-xx).^2 + (yr-yy).^2);
errormax=max(error)*1000;
disp(['error máximo      [mm]: ', num2str(errormax)])

% calculamos el desempeño
errorx   = (xr-xx).^2;
errory   = (yr-yy).^2;
errorxy  = errorx + errory;
rrx      = xr.^2;
rry      = yr.^2;
sumrx    = sum(rrx);
sumry    = sum(rry);
desmp    = sqrt(sum(errorxy)/(sumrx+sumry))*100;
disp(['desempeño       [<=5%]: ',num2str(desmp)])
```

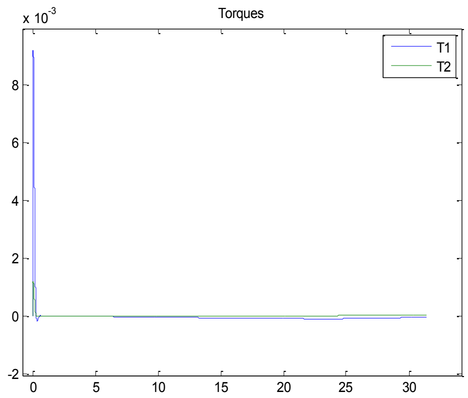
- <u>Pruebas realizadas</u>

I. K11 = 10 ; K12 = 10 ; K21 = 25 ; K22= 25 ; Vel = 0.1m/s

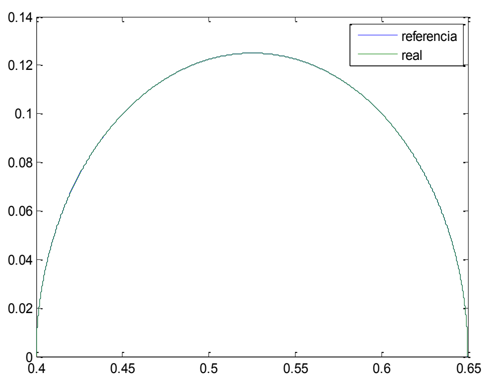
ϕ_1 , ϕ_2



Torques T1, T2



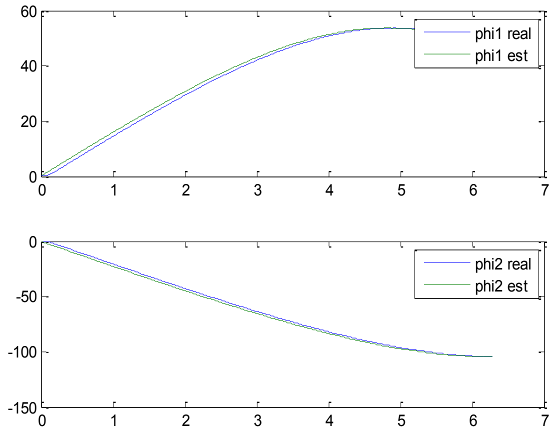
Trayectoria



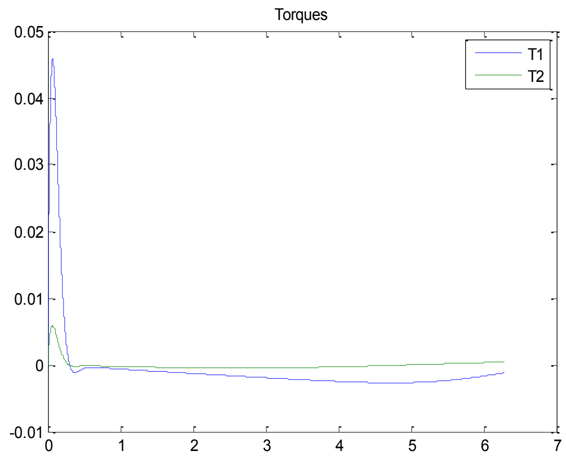
error máximo [mm]:	1.2468
desempeño :	0.22364

II. K11 = 10 ; K12 = 10 ; K21 = 25 ; K22= 25 ; Vel = 1 m/s

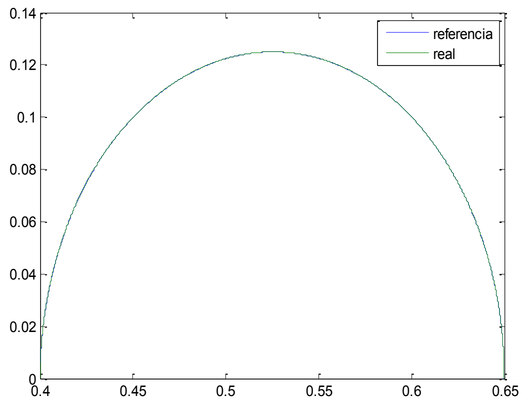
ϕ_1 , ϕ_2



Torques T1, T2



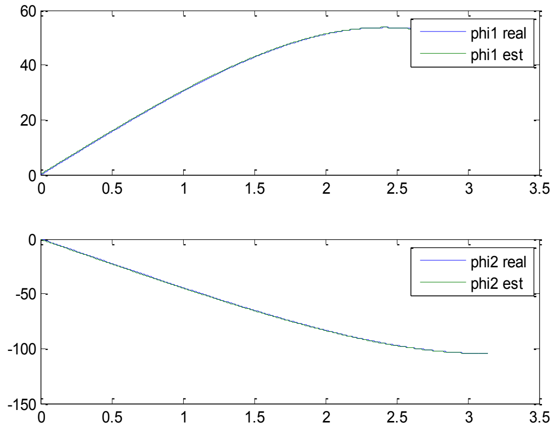
Trayectoria



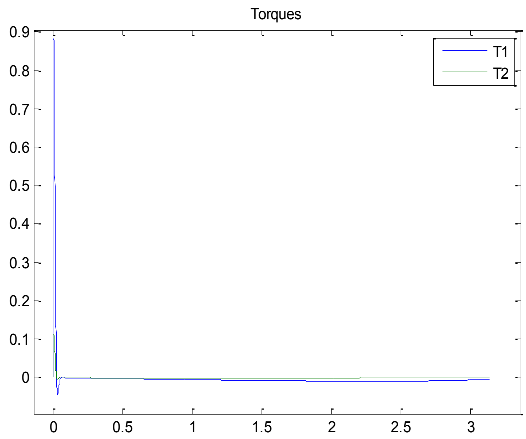
error máximo [mm]:	6.2348
desempeño :	1.1135

III. K11 = 100 ; K12 = 100 ; K21 = 200 ; K22= 200 ; Vel = 1 m/s

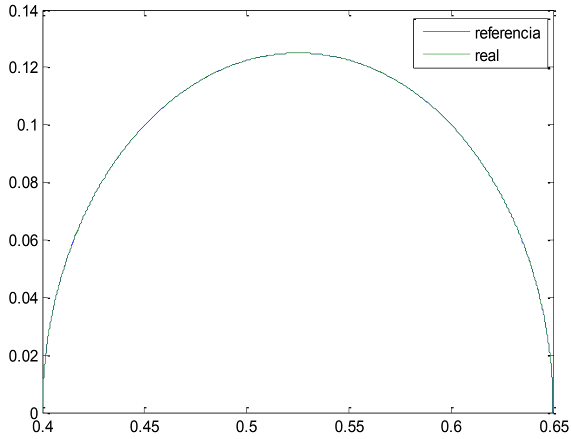
ϕ_1 , ϕ_2



Torques T1, T2



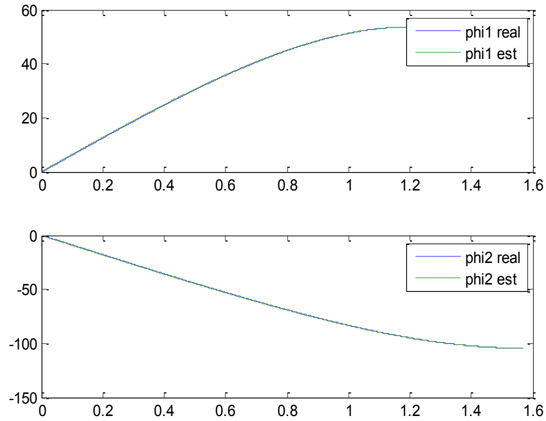
Trayectoria



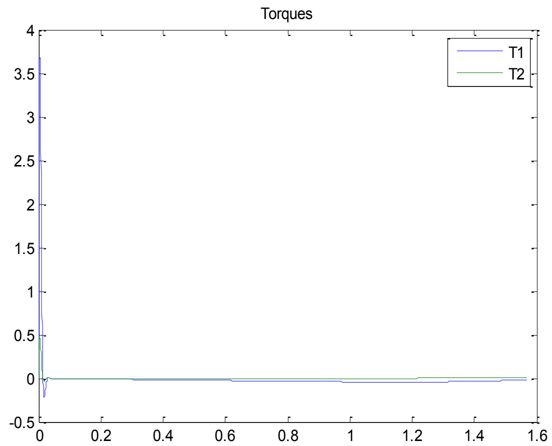
error máximo [mm]:	1.4019
desempeño :	0.24179

I. K11 = 200 ; K12 = 200 ; K21 = 400 ; K22= 400 ; Vel = 2 m/s

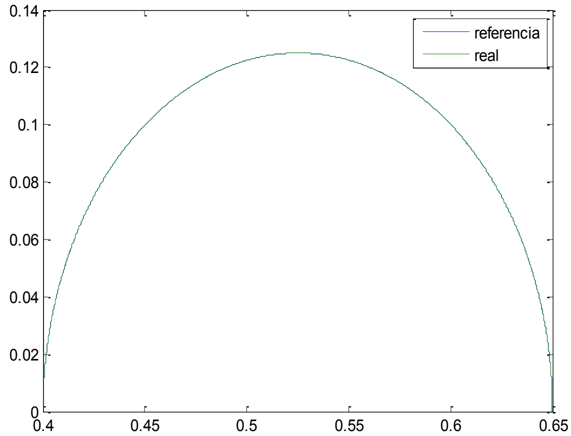
ϕ_1 , ϕ_2



Torques T1, T2



Trayectoria



error máximo [mm]:	1.4796
desempeño :	0.2539

3. Control lineal óptimo

A partir de la ecuación:

$$M_{(\phi)}\ddot{\phi} + C_{(\phi,\dot{\phi})} = ST$$

Se linealiza el sistema alrededor del estado:

$$\begin{bmatrix}\phi_1\\ \phi_2\end{bmatrix} = \begin{bmatrix}0\\ 0\end{bmatrix}$$

$$\begin{bmatrix}\dot{\phi}_1\\ \dot{\phi}_2\end{bmatrix} = \begin{bmatrix}0\\ 0\end{bmatrix}$$

Las matrices resultan:

$$C1 \ = \ -m2 * L1 * l2 * (fi1p + fi2p)^2 * sin(fi2) = 0$$

$$C2 \ = \ m2 * L1 * l2 * fi1p * fi1p * sin(fi2) = 0$$

Entonces:

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

$$M11 \ = \ I1 \ + \ m1 * l1 * l1 \ + \ m2 * L1 * L1 \ + \ m2 * L1 * l2$$

$$M12 \ = \ m2 * L1 * l2$$

$$M21 \ = \ I2 \ + \ m2 * l2 * l2 \ + \ m2 * L1 * l2$$

$$M22 \ = \ I2 \ + \ m2 * l2 * l2$$

$$M \ = \ \begin{bmatrix}M_{11} & M_{12}\\ M_{21} & M_{22}\end{bmatrix}$$

Resulta el siguiente sistema linealizado:

$$M\ddot{\phi} = ST$$

$$\dot{X} = \begin{bmatrix}\dot{\phi}_1\\ \dot{\phi}_2\\ \ddot{\phi}_1\\ \ddot{\phi}_2\end{bmatrix} = \begin{bmatrix}0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\phi_1\\ \phi_2\\ \dot{\phi}_1\\ \dot{\phi}_2\end{bmatrix} + \begin{bmatrix}0 & 0\\ 0 & 0\\ (M^{-1}S)_{11} & (M^{-1}S)_{12}\\ (M^{-1}S)_{21} & (M^{-1}S)_{22}\end{bmatrix}\begin{bmatrix}T_1\\ T_2\end{bmatrix}$$

Ley de control:

$$\begin{bmatrix}T_1\\ T_2\end{bmatrix} = -K_1(\phi_1 - {\phi_1}^*) - K_2(\phi_2 - {\phi_2}^*) - K_3\dot{\phi}_1 - K_4\dot{\phi}_2$$

Sistema linealizado:

```
M11 = I1 + m1*l1*l1 + m2*L1*L1 + m2*L1*l2;
M12 = m2*L1*l2;
M21 = I2 + m2*l2*l2 + m2*L1*l2;
M22 = I2 + m2*l2*l2;

 M = [ M11  M12
       M21  M22 ];
S = [ 1  -1
      0   1 ];
invM=inv(M);

A=[ zeros(2,2) eye(2,2)
    zeros(2,2) zeros(2,2) ];
B=[ zeros(2,2)
      invM*S    ];
```

Controlador Lineal Óptimo:

```
q1 = input('Ganancia K1 (fi1)  10: ');
q2 = input('Ganancia K2 (fi2)  10: ');
q3 = input('Ganancia K3 (fi1p)  0: ');
q4 = input('Ganancia K4 (fi2p)  0: ');
R=eye(2);
Q = diag([ q1 q2 q3 q4 ]);
P = are(A,B*inv(R)*B',Q);
K = inv(R)*B'*P;
```

Simulación:

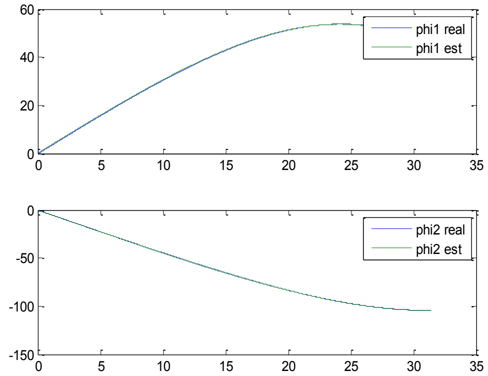
```
for tt = ti:dt:tf
   phi1(k,1) = fi1;     phi2(k,1) = fi2;    t(k,1) = tt;
   fi1 = ang(1,1);    fi2 = ang(2,1);
   fi1p = vel(1,1);   fi2p = vel(2,1);
   M11a = I1 + m1*l1*l1 + m2*L1*L1 + m2*L1*l2 * cos(fi2);
   M12a = m2*L1*l2*cos(fi2);
   M21a = I2 + m2*l2*l2 + m2*L1*l2*cos(fi2);
   M22a = I2 + m2*l2*l2;
   M = [ M11a  M12a
           M21a  M22a ];

   C1 = -m2*L1*l2*(fi1p+fi2p)^2*sin(fi2);
   C2 = m2*L1*l2*fi1p*fi1p*sin(fi2);
   C = [ C1
          C2 ];

   %Ley de control: %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   T = -K*[fi1-r1(k,1); fi2-r2(k,1);fi1p;fi2p];

   T1(k,1)=T(1,1);
   T2(k,1)=T(2,1);

   accel = inv(M)*(-C+S*T);
   ang = ang + dt*vel;
   vel = vel + dt*accel;
   k = k + 1;
end
```

- Pruebas realizadas con el controlador lineal

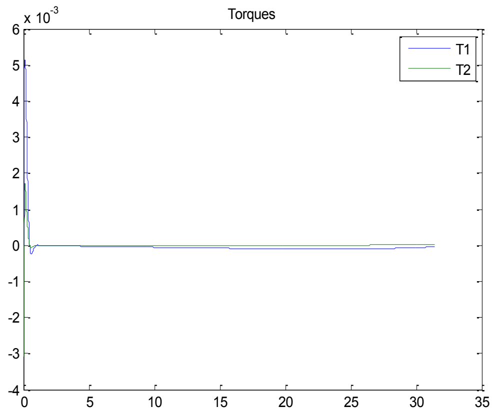
Se eligieron los pesos de tal manera que el sistema controlado trabaje bien.

I. q1 = 10 ; q2 = 10 ; q3 = 0 ; q4= 0 ; Vel = 0.1m/s

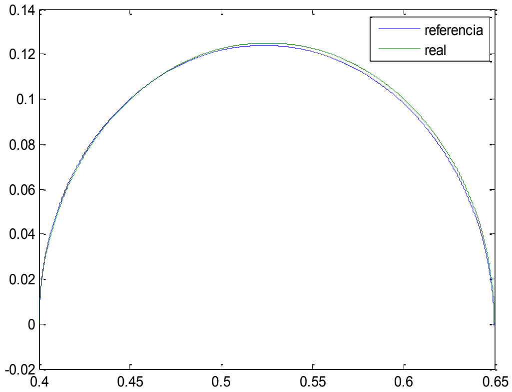
ϕ_1 , ϕ_2



Torques T1, T2



Trayectoria

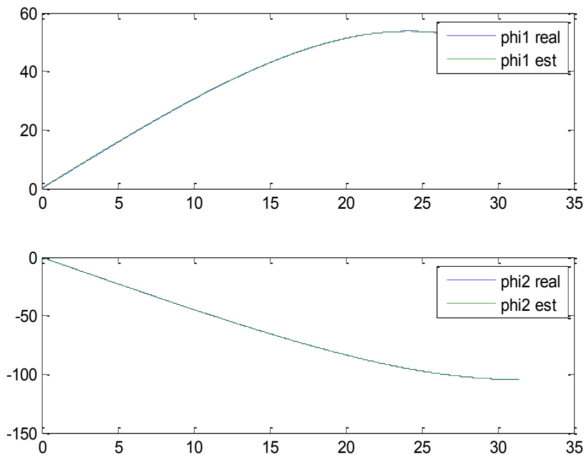


error máximo [mm]:	3.5962
desempeño :	0.39443

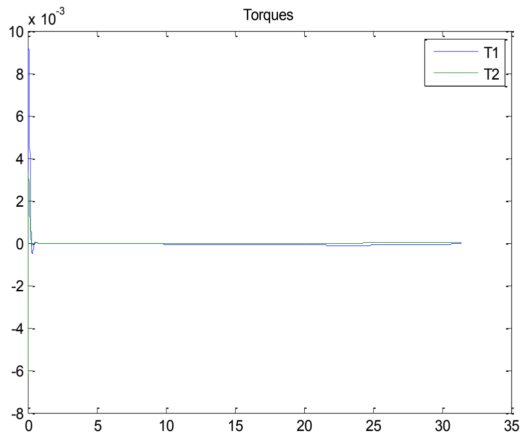
Como el error es grande se plantea aumentar los pesos:

II. q1 = 100 ; q2 = 100 ; q3 = 0 ; q4= 0 ; Vel = 0.1m/s

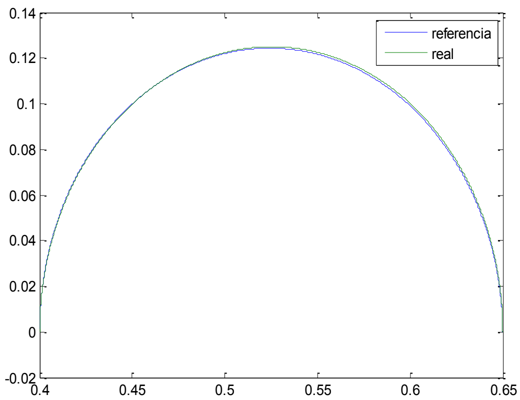
ϕ_1 , ϕ_2



Torques T1, T2



Trayectoria

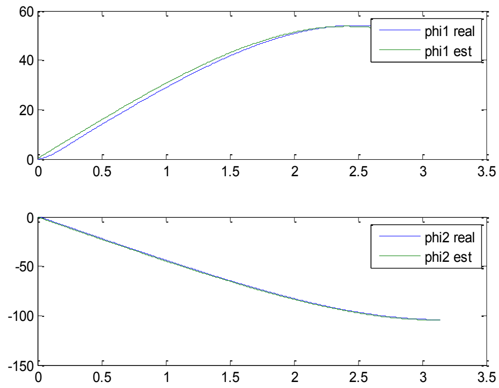


error máximo [mm]:	2.059
desempeño :	0.2261

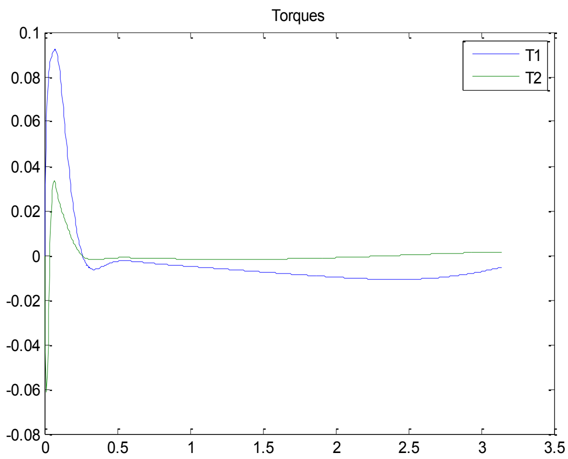
La precisión es aceptable.

III. q1 = 100 ; q2 = 100 ; q3 = 0 ; q4= 0 ; Vel = 1m/s

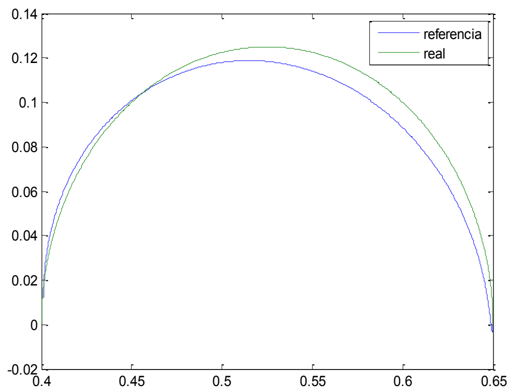
ϕ_1 , ϕ_2



Torques T1, T2



Trayectoria

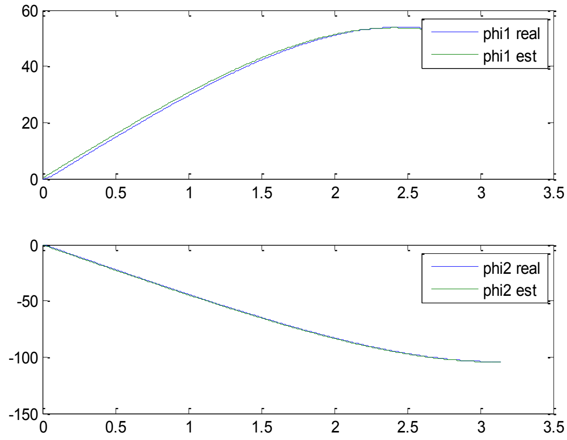


error máximo [mm]:	20.438
desempeño :	2.2204

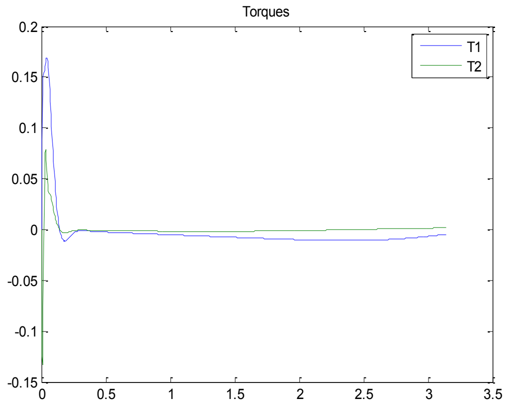
Como el error es grande se plantea aumentar los pesos:

IV. q1 = 1000 ; q2 = 1000 ; q3 = 0 ; q4= 0 ; Vel = 1m/s

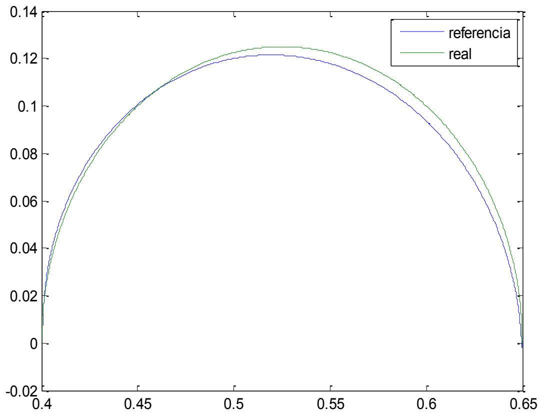
ϕ_1 , ϕ_2



Torques T1, T2



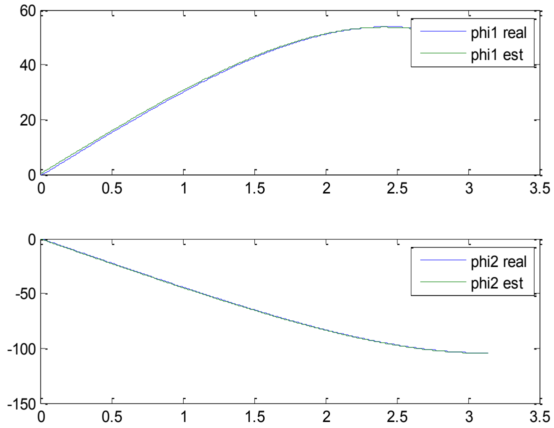
Trayectoria



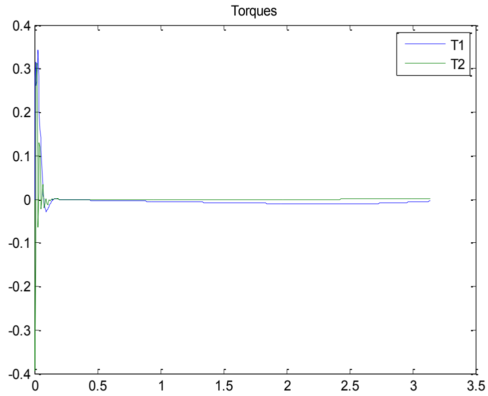
error máximo [mm]:	11.9282
desempeño :	1.2996

V. q1 = 10000 ; q2 = 10000 ; q3 = 0 ; q4= 0 ; Vel = 1m/s

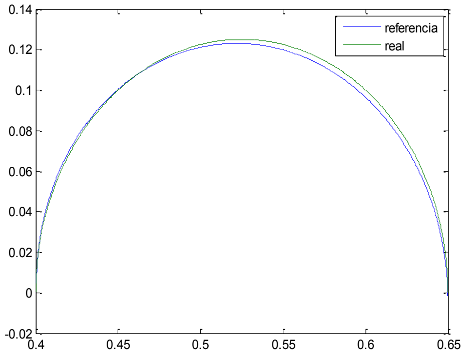
$\boldsymbol{\phi_1}$, $\boldsymbol{\phi_2}$



Torques T1, T2



Trayectoria



error máximo [mm]:	7.112
desempeño :	0.77537

VI. q1 = 100000 ; q2 = 100000 ; q3 = 0 ; q4= 0 ; Vel = 1m/s

RICCATI NO TIENE SOLUCIÓN, y no se ha podido disminuir más el error.

Conclusiones:

1. El controlador no lineal Backstepping entrega buenos resultados para velocidades bajas, medias y altas.

2. El controlador no lineal Backstepping requiere que las constantes K11, K12, K21, K22 sean positivas para que el sistema controlado sea estable.

3. Al aumentar la velocidad de movimiento en el controlador Backstepping se requiere mayor torque (principalmente el pico inicial).

4. El controlador lineal calculado con la planta del robot 2DOF linealizada alrededor de (0,0,0,0) entrega buenos resultados para velocidades bajas (menores a 0.1m/s). Sin embargo para velocidades mayores a 1m/s no se puede reducir el error por más que se aumenten los pesos (al aumentar mucho los pesos Riccati no tiene solución).