

Control con dos actuadores de diferente ancho de banda: planta generalizada

© 2019, Antonio Sala Piqueras, Universitat Politècnica de València. Todos los derechos reservados.

Este código funcionó sin errores en Matlab R2018b.

Presentación en vídeo en: <http://personales.upv.es/asala/YT/V/dacthinf1.html> y <http://personales.upv.es/asala/YT/V/dacthinf2.html>.

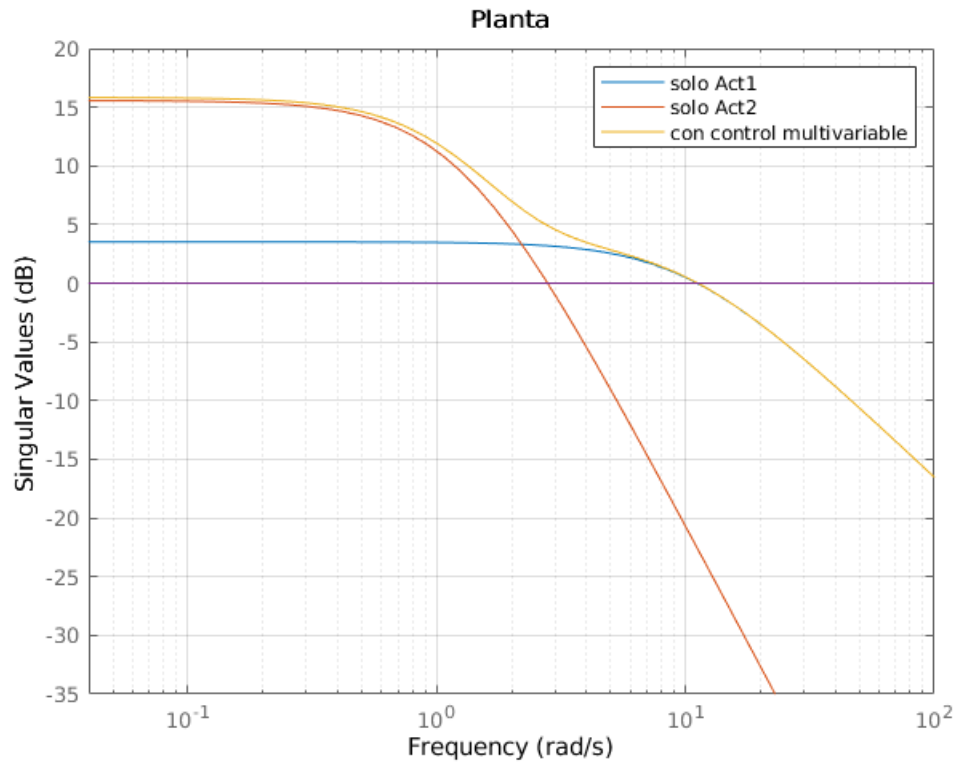
Objetivo: diseñar un controlador óptimo (a partir de una planta generalizada) que tenga en cuenta las diferentes características de dos actuadores de diferente ancho de banda, de forma similar a la estructura de control en cascada con actuador extra.

Tabla de contenidos

1.- Modelado y análisis de controlabilidad entrada-salida.....	1
2.- Planteamiento en forma planta generalizada.....	2
3.- Diseño de controladores.....	3
4.- Simulación ante tren de escalones y rampa.....	6

1.- Modelado y análisis de controlabilidad entrada-salida

```
s=tf('s'); G=[1.5/(0.1*s+1) 6/(0.8*s+1)^2]; %modelo escalado actuadores
saturan en 1
sigma(G(1),G(2),G,tf(1)), grid on, axis([.04 100 -35 20])
legend('solo Act1','solo Act2','con control multivariable');
title('Planta');
```



Con referencias de amplitud 1 (si la planta estuviera escalada a ± 1 en entradas y salidas y se deseara evitar la saturación --formalmente en régimen permanente senoidal--):

```
semiperiodo_lim_act1=pi/11
```

```
semiperiodo_lim_act1 = 0.2856
```

```
semiperiodo_lim_act2=pi/2.7
```

```
semiperiodo_lim_act2 = 1.1636
```

```
semiperiodo_lim_act2_ref3=pi/1.2
```

```
semiperiodo_lim_act2_ref3 = 2.6180
```

2.- Planteamiento en forma planta generalizada

Plantemos el control con un grado de libertad estandar: $u = K_{2 \times 1}(r - y)$, $y = G_{1 \times 2}u$

La planta generalizada, incluyendo las acciones de control como salidas para ponderarlas posteriormente será:

$$\begin{pmatrix} e \\ u_1 \\ u_2 \\ \vdots \\ e \end{pmatrix} = \begin{pmatrix} 1 & | & -g_{11} & -g_{12} \\ 0 & | & 1 & 0 \\ 0 & | & 0 & 1 \\ \vdots & | & \vdots & \vdots \\ 1 & | & -g_{11} & -g_{12} \end{pmatrix} \begin{pmatrix} r \\ \vdots \\ u_1 \\ u_2 \end{pmatrix}$$

```
%genplant: %inputs r u1 u2, %outputs error u1 u2 | error
GenP=[tf(1) -G;zeros(2,1) tf(eye(2));tf(1) -G];
size(GenP)
```

Transfer function with 4 outputs and 3 inputs.

3.- Diseño de controladores

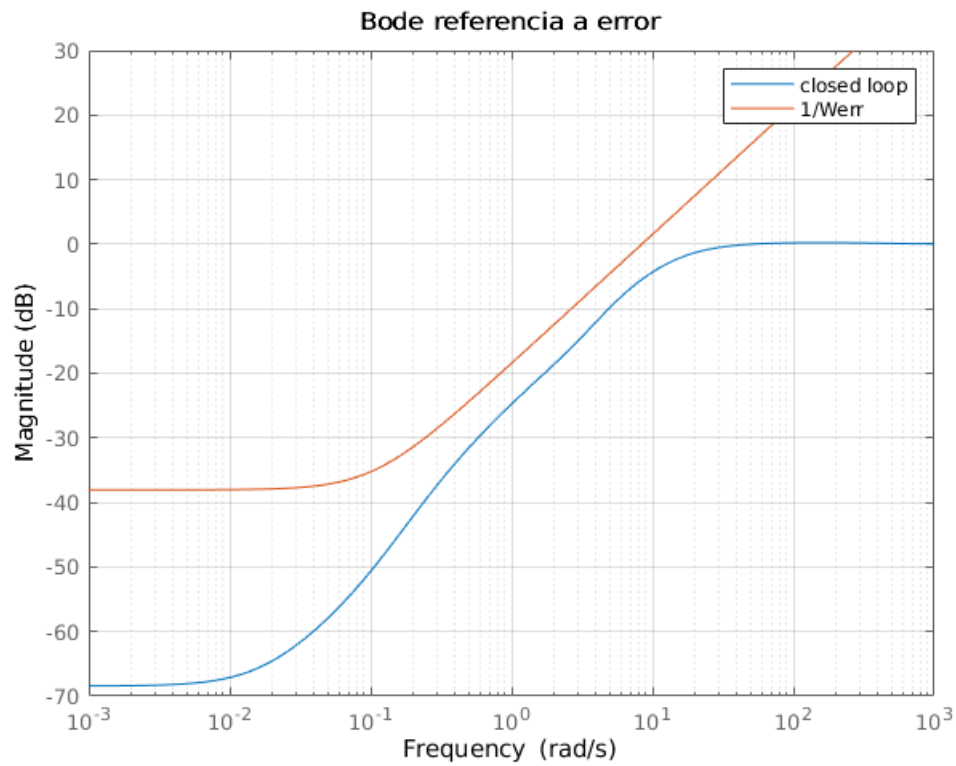
Evaluemos varias alternativas de diseño:

```
diseno=4;
if(disenyo==1) %solo gasto act. rapido
    bw=11.0;
    Wi=1+0.25/(s/.05+1); %referencia un pelin mas grande a baja frec.
    Wu=blkdiag(1,1e4);%actuadores saturan a 1.
elseif(disenyo==2) %solo gasto act. lento
    bw=1.74;
    Wi=1+2/(s/.05+1); %referencia mas grande a baja frec.
    Wu=blkdiag(1e4,1);
elseif(disenyo==3) %dos actuadores ponderan igual, rapidez maxima
    bw=10.5;
    Wi=1+5/(s/.05+1); %referencia mas grande a baja frec.
    Wu=blkdiag(1,1);%actuadores saturan a 1.
else %baja frecuencia solo deseo actuador 2:
    bw=8.4;
    Wi=1+4.65/(s/.05+1);
    Wu=blkdiag((s+0.64)/(s+.02),1);%actuadores saturan a 1. Exijo act 1
    pequeño a baja frec expresamente.
end
Werr=80/(s/(bw/80)+1); %Werr vale 0dB aprox en s=bw, y vale 1/80 a
frecuencia bw/80.
%bodemag(Wi), grid on
Wsalidas=ss(blkdiag(Werr,Wu,1));
GenPW=minreal(Wsalidas*GenP*blkdiag(Wi,eye(2)));%planta generalizada
ponderada

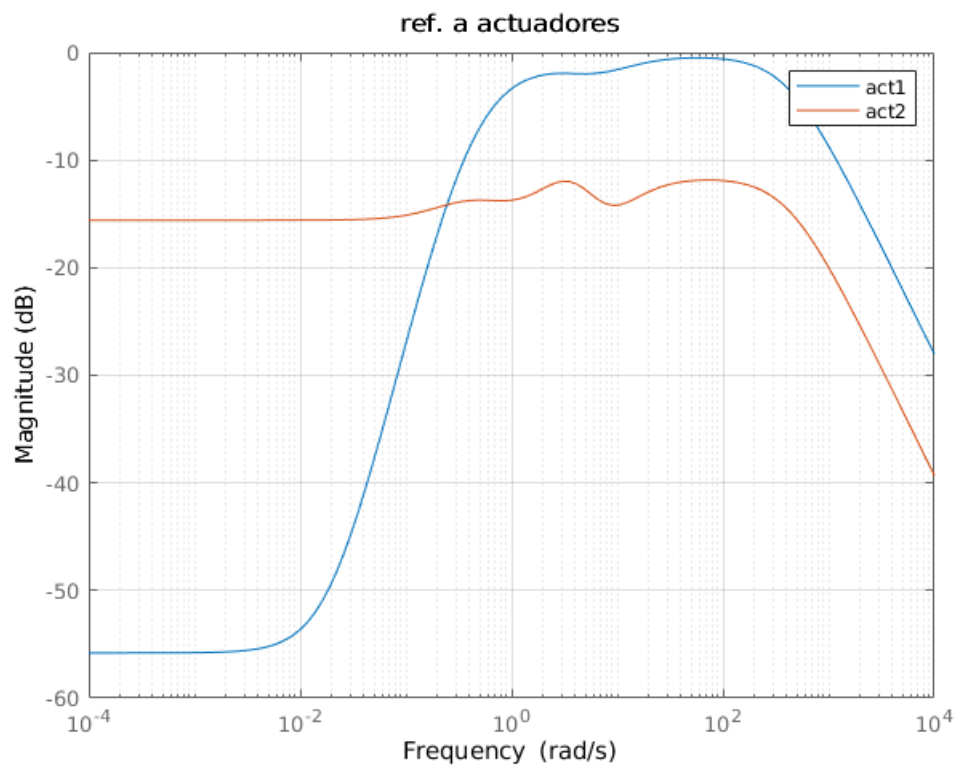
metodo='hinf'; %para probar h2 o hinfinito
if(strcmp(metodo,'h2'))
    [K,C12,Gam2,Info]=h2syn(GenPW,1,2);Gam2
elseif(strcmp(metodo,'hinf'))
    [K,C1inf,Gaminf,Info]=hinfsyn(GenPW,1,2);Gaminf
end
```

Gaminf = 0.9984

```
C1_noponderada=lft(GenP,K);
%dcgain(C1_noponderada) '
bodemag(C1_noponderada(1,:),1/Werr), grid on, title('Bode referencia a
error')
legend('closed loop','1/Werr')
```



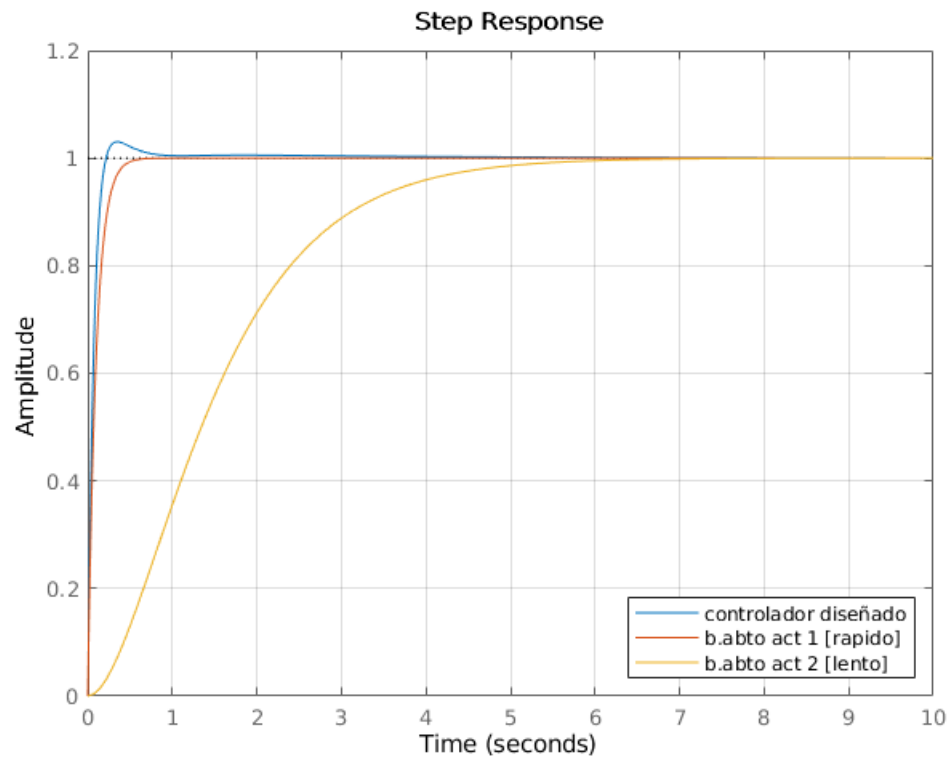
```
bodemag(Cl_noponderada(2,:),Cl_noponderada(3,:)), grid on, title('ref. a
actuadores'), legend('act1','act2')
```



```

step(1-C1_noponderada(1,1),G(1)/dcgain(G(1)),G(2)/dcgain(G(2)),10), grid on
legend('controlador diseñado','b.abto act 1 [rapido]','b.abto act 2 [lento]','Location','SouthEast')

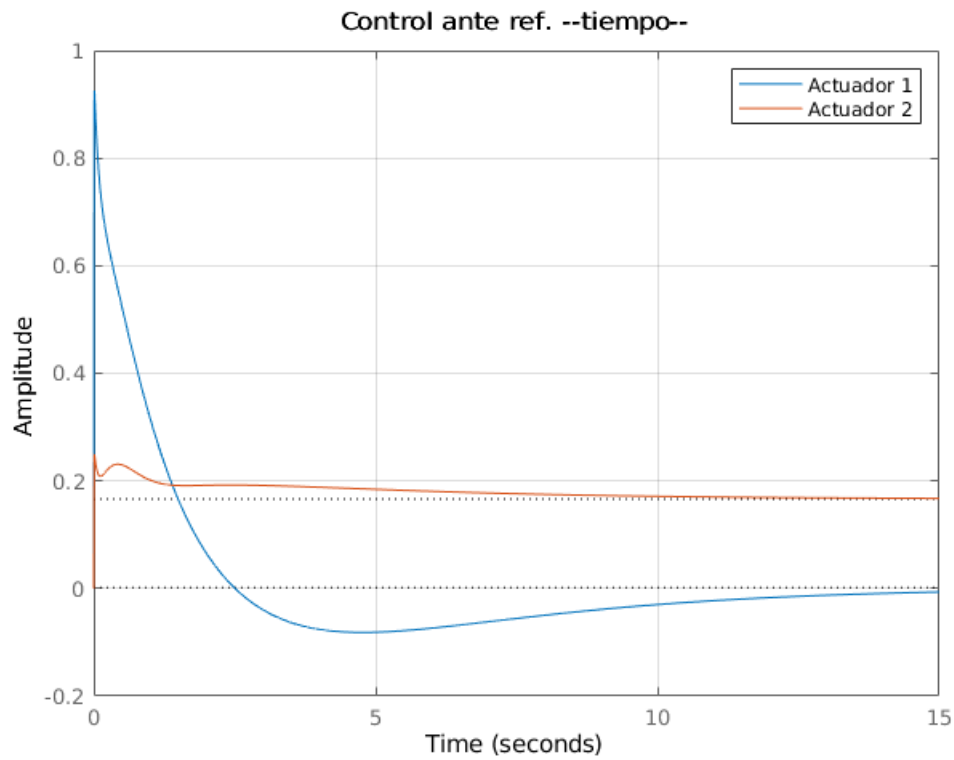
```



```

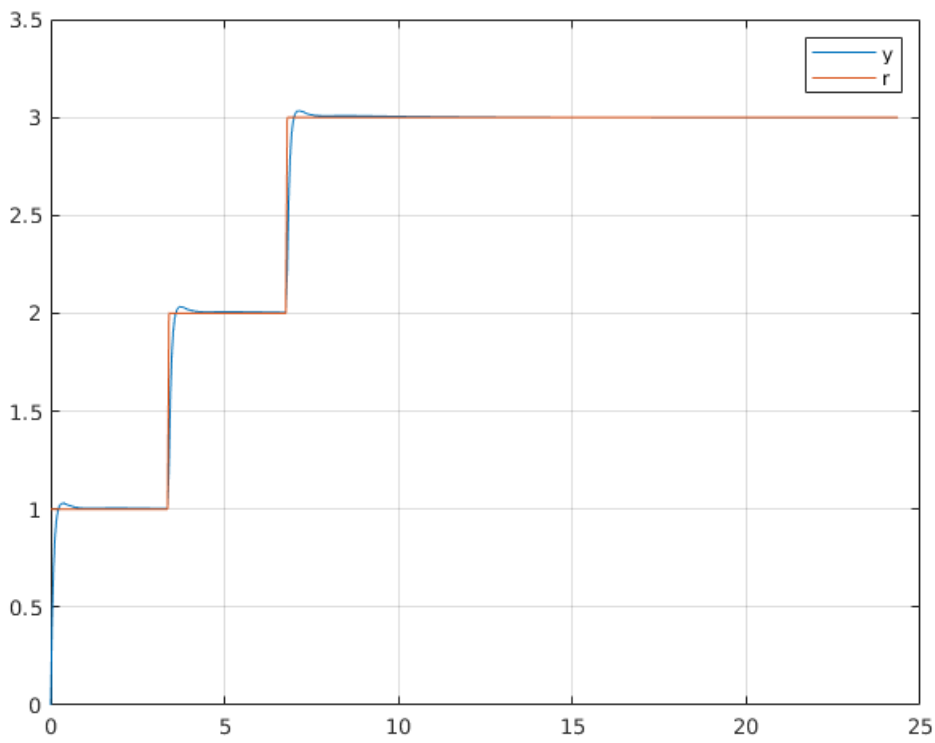
step(C1_noponderada(2,1),C1_noponderada(3,1),15), grid on
title('Control ante ref. --tiempo--')
legend('Actuador 1','Actuador 2');

```

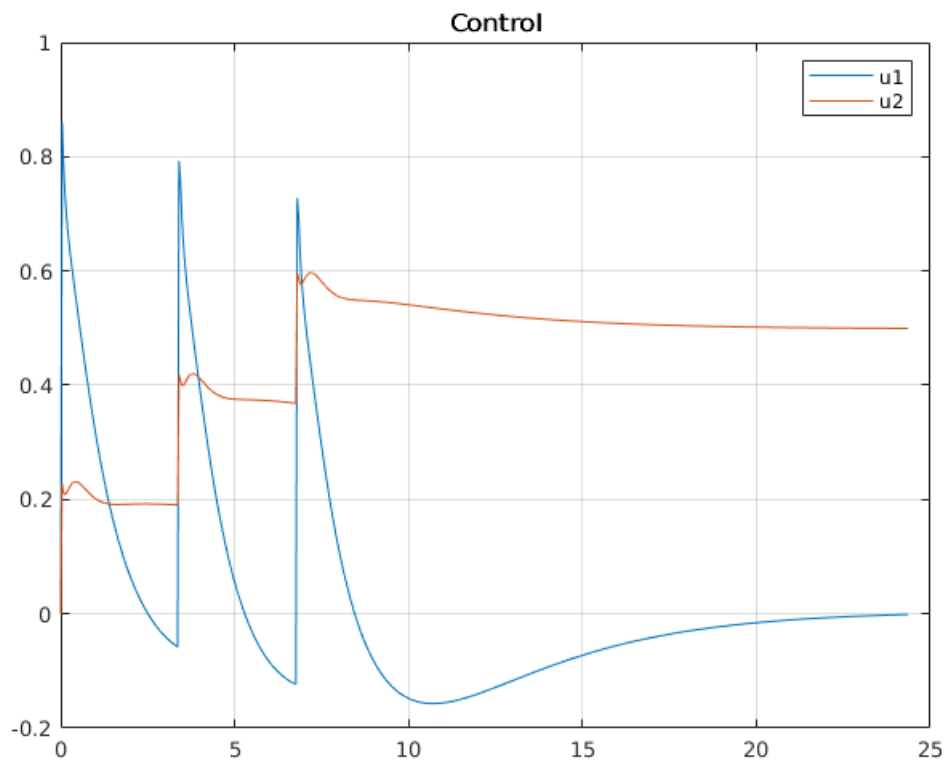


4.- Simulación ante tren de escalones y rampa

```
Ts=0.04;
refprueba=[ones(85,1); 2*ones(85,1); 3*ones(440,1)];
tiempo=[0:(length(refprueba)-1)]*Ts;
YY=lsim(Cl_noponderada,refprueba,tiempo);
plot(tiempo,[refprueba-YY(:,1) refprueba]), grid on, legend('y','r')
```



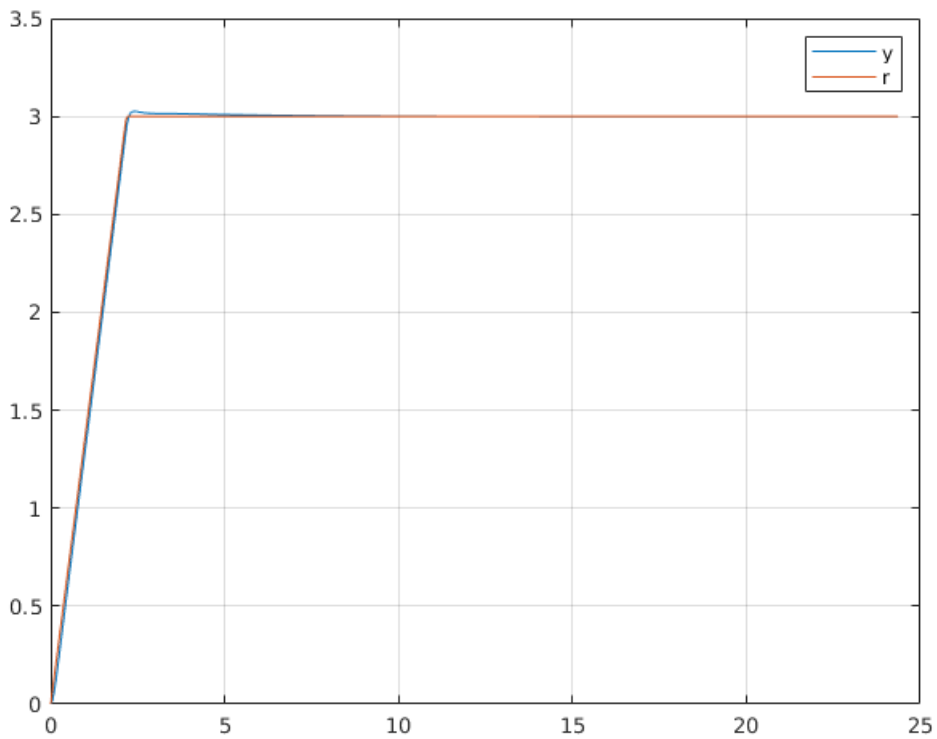
```
plot(tiempo,[YY(:,2:3)]), grid on, legend('u1','u2'), title('Control')
```



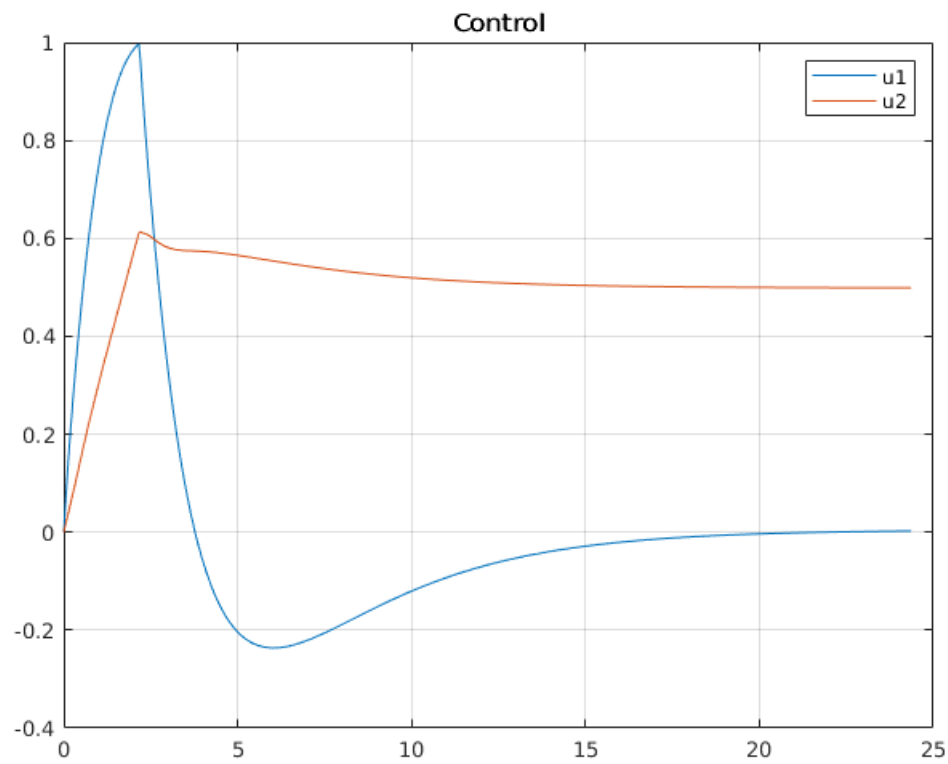
```
pendiente=[20 2.45 .4 1.38];
pte=pendiente(disenyo);
amplitudes=[1.2 3 3 3];
refprueba=min(pte*tiempo,amplitudes(disenyo));
tiempo_subida_ref=3/pte
```

```
tiempo_subida_ref = 2.1739
```

```
YY=lsim(C1_noponderada,refprueba,tiempo);
plot(tiempo,[refprueba-YY(:,1) refprueba]), grid on, legend('y','r')
```



```
plot(tiempo,[YY(:,2:3)]), grid on, legend('u1','u2'), title('Control')
```



Robustez (márgen incertidumbre coprime normalizada, no objetivo de este material):

```
ncfmargin(G,K)
```

```
ans = 0.3322
```