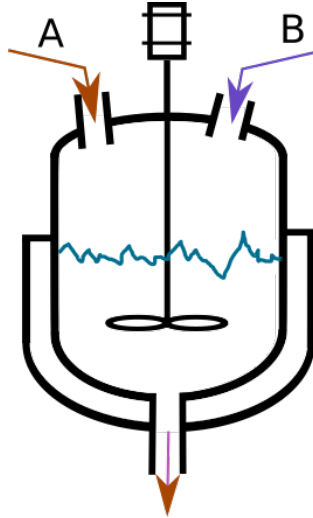


Control multibucle de un tanque de mezclado (RGA)

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

Este código ejecutó correctamente en Matlab R2019a

Motivación y objetivos: Este código ilustra el comportamiento en bucle cerrado con control multibucle de dos opciones de control: el emparejamiento recomendado por la RGA o el alternativo no recomendado.



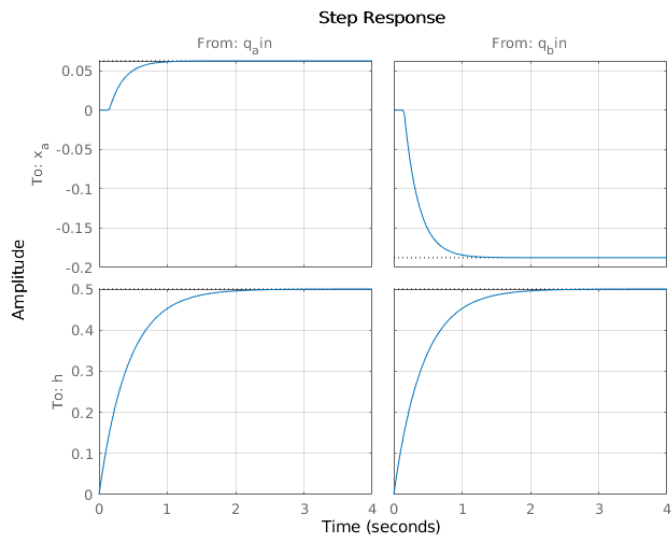
En otros materiales, se ha obtenido un modelo en representación interna, en cierto punto de funcionamiento, que cargamos aquí:

```
load ModeloTanqueLin
size(TanqueMezcladoLin)
```

State-space model with 2 outputs, 2 inputs, and 2 states.

Añadiremos un retardo (que aproximará dinámica de tiempo de mezclado/difusión, dinámica del sensor...) en la concentración (1st order+delay).

```
TanqueMezcladoLin.OutputDelay=[0.15;0];
step(TanqueMezcladoLin), grid on
```



```
Tanque_tf=minreal(tf(TanqueMezcladoLin))
```

```
Tanque_tf =
```

```
From input "q_ain" to output...
                                0.2941
x_a: exp(-0.15*s) * -----
                                s + 4.706
```

```
                                1.176
h:  -----
                                s + 2.353
```

```
From input "q_bin" to output...
                                -0.8824
x_a: exp(-0.15*s) * -----
                                s + 4.706
```

```
                                1.176
h:  -----
                                s + 2.353
```

```
Continuous-time transfer function.
```

Ganancia y ganancia relativa:

Su ganancia estática es:

```
gan=dcgain(TanqueMezcladoLin)
```

```
gan = 2x2
      0.0625   -0.1875
      0.5000    0.5000
```

y su RGA:

```
RGA=gan.*inv(gan')
```

```

RGA = 2x2
      0.2500    0.7500
      0.7500    0.2500

```

por lo que recomienda el emparejado no-diagonal (q_a-> altura, q_b->concentración de b).

Probaremos ambos emparejamientos.

Comprobación emparejamiento diagonal malo (concentr--y1,qa--u1), (nivel--y2,qb--u2)

Sintonicemos reguladores de forma separada.

```
g11=Tanque_tf(1,1), g22=Tanque_tf(2,2)
```

```
g11 =
```

```

From input "q_ain" to output "x_a":
      0.2941
exp(-0.15*s) * -----
              s + 4.706

```

```
Continuous-time transfer function.
```

```
g22 =
```

```

From input "q_bin" to output "h":
      1.176
-----
      s + 2.353

```

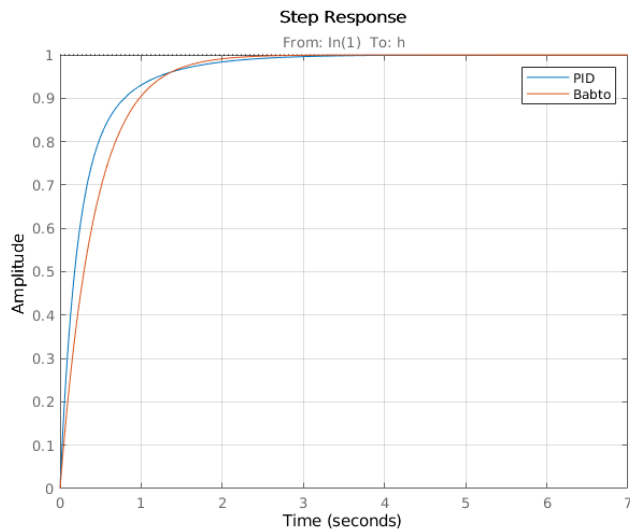
```
Continuous-time transfer function.
```

Cerraremos el bucle "q_b->nivel" y "q_a->concentración".

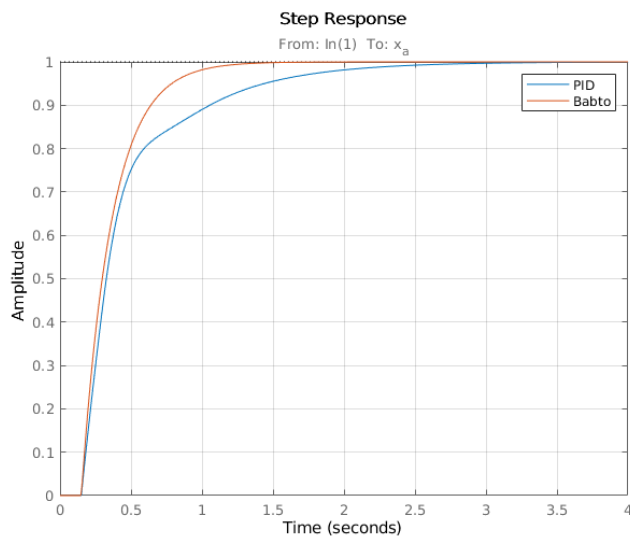
```

s=tf('s');
PIDnivel=3.5+6/s;
buclenivelaislado=feedback(g22*PIDnivel,1);
step(buclenivelaislado,g22/dcgain(g22)), grid on, legend('PID','Babto')

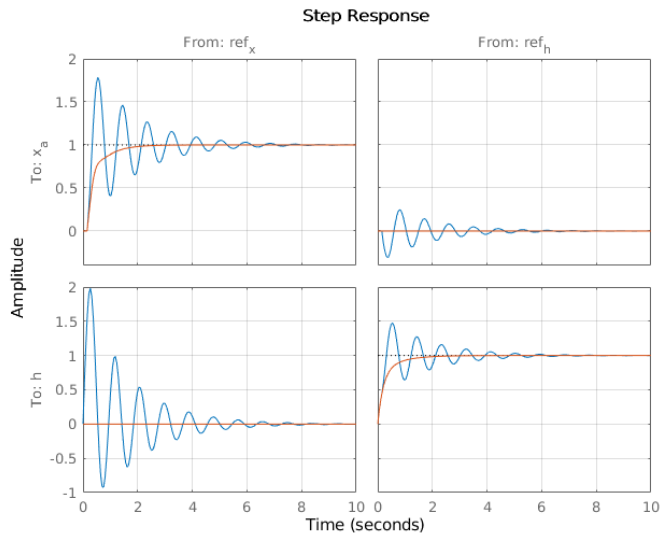
```



```
PIDconc=(11+33/s);
bucleconc_aislado=feedback(g11*PIDconc,1);
step(bucleconc_aislado,g11/dcgain(g11)), grid on, legend('PID','Babto')
```



```
cosa=feedback(TanqueMezcladoLin*[PIDconc 0;0 PIDnivel],eye(2));
cosa.InputName={'ref_x','ref_h'};
step(cosa,blkdiag(bucleconc_aislado,buclenivelaislado)), grid on %todos
```



Comprobación emparejamiento no diagonal (concentr--y1,qb--u2), (nivel--y2,qa--u1)

Sintonicemos reguladores de forma separada.

```
g12=Tanque_tf(1,2), g21=Tanque_tf(2,1)
```

```
g12 =
```

```
From input "q_bin" to output "x_a":  
-0.8824  
exp(-0.15*s) * -----  
s + 4.706
```

```
Continuous-time transfer function.
```

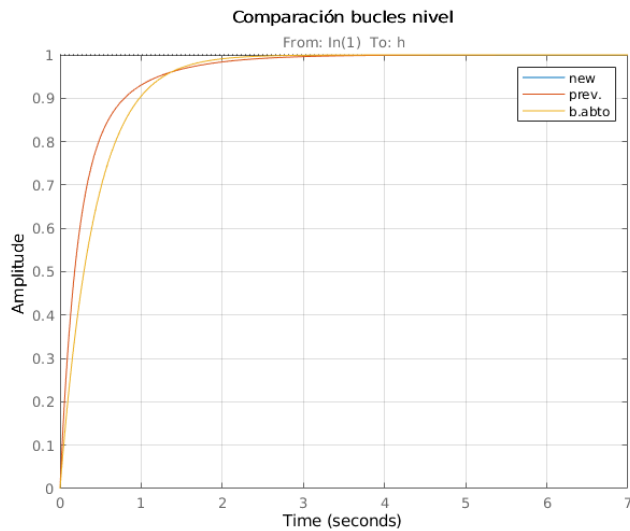
```
g21 =
```

```
From input "q_a" to output "h":  
1.176  
-----  
s + 2.353
```

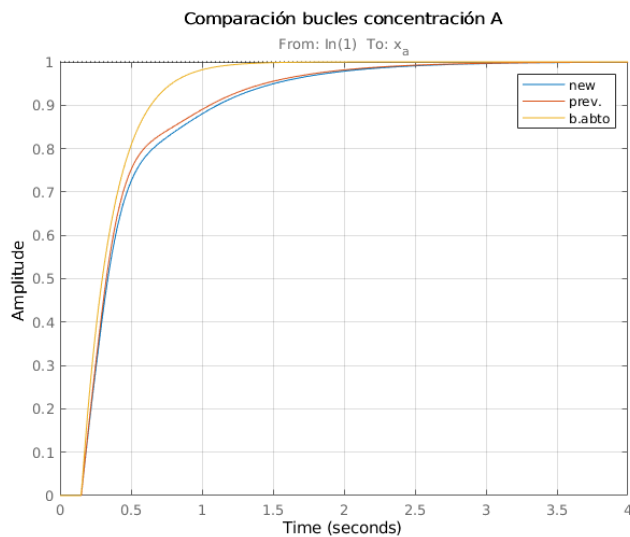
```
Continuous-time transfer function.
```

Cerraremos el bucle "q_b->concentracion" y "q_a->nivel".

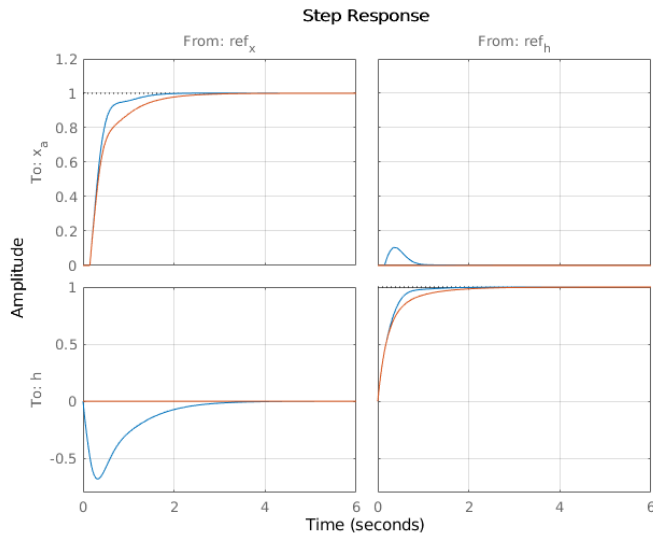
```
s=tf('s');  
niveלותraopcion=buclenivelaislado;  
buclenivelaislado=feedback(g21*PIDnivel,1);  
step(buclenivelaislado,niveלותraopcion,g21/dcgain(g21)), grid on  
title('Comparación bucles nivel'), legend('new','prev.','b.abto')
```



```
PIDconc=-(3.5+10.5/s);
concentracion_otraopcion=bucleconc_aislado;
bucleconc_aislado=feedback(g12*PIDconc,1);
step(bucleconc_aislado,concentracion_otraopcion,g12/dcgain(g12)), grid on
title('Comparación bucles concentración A'), legend('new','prev.','b.abto')
```



```
cosa=feedback(TanqueMezcladoLin*[0 PIDnivel;PIDconc 0],eye(2));
cosa.InputName={'ref_x','ref_h'};
step(cosa,blkdiag(bucleconc_aislado,buclenivelaislado)), grid on %todos
```



Nota: no hemos considerado saturación de actuadores (antiwindup), ni actividad de alta frecuencia (ancho de banda limitado en actuadores, ruido de medida...).

Conclusiones

Si se sintonizan los reguladores "de uno en uno, aisladamente" con los otros desconectados, el emparejamiento con RGA más cercano a 1 presenta unos resultados "multibucle" más parecidos a los del diseño individual inicial. En el emparejamiento "malo", cerrar el otro bucle cambia mucho los resultados.