

## Funciones en MATLAB para ajuste empírico de controladores PID

A continuación se incluyen las cabeceras (el mismo texto que se obtendría de la función al invocarla con el comando help) de un conjunto de funciones \*.m para el ajuste de controladores PID, siguiendo los procedimientos descritos en el capítulo 3 (Ajuste empírico). Las funciones han sido programadas en MATLAB por Fernando Morilla y se distribuyen codificadas para que se puedan ejecutar pero no se puedan modificar.

Una de las funciones, concretamente “**est\_rla**”, sirve para los parámetros ( $K$ ,  $T_p$  y  $T_o$ ) de un modelo de primer orden más retardo que mejor aproximan la respuesta escalón del proceso, otra, “**est\_cl**”, sirve para estimar la ganancia crítica ( $k_c$ ) y el período de oscilación mantenida ( $t_c$ ) de un proceso a partir del análisis de una experiencia de ciclo límite con relé con histeresis. Mientras que “**kctc\_m1**” permite determinar la ganancia crítica y el período de oscilación mantenida asociada al un modelo de primer orden más retardo.

Otras siete funciones (“**ah**”, “**amigo**”, “**gm**”, “**hah**”, “**hhc**”, “**sint\_ci**” y “**zn**”) recogen todas las fórmulas de sintonía descritas en el apartado 3.3.

El conjunto de funciones se acompaña de tres programas \*.m, concretamente “**est\_mod**”, “**ajuste**” y “**pru\_ajuste**”, se recomienda ejecutarlos o ver su contenido como ejemplo de utilización de estas funciones en las tres fases del ajuste empírico de controladores PID: 1) estimación, 2) ajuste de parámetros del controlador y 3) prueba del ajuste en simulación. El primer programa muestra cómo hacer uso de dos archivos de datos, “**exp\_la.mat**” y “**exp\_cl.mat**” y guardar los parámetros estimados en los archivos “**ktpto.mat**” y “**kctc.mat**”. El programa “**ajuste**” muestra cómo a partir de los parámetros estimados, contenidos en “**ktpto.mat**” o “**kctc.mat**”, se pueden calcular los parámetros de control y guardarlos en el archivo “**kptitd.mat**”. Y el tercer programa muestra cómo se puede comprobar en simulación el ajuste realizado, hace uso de los parámetros del modelo del proceso contenidos en el archivo “**ktpto.mat**” y los parámetros del controlador contenidos en el archivo “**kptitd.mat**”.

**Observación:** Los 16 archivos \*.p son totalmente necesarios para que las funciones y los programas \*.m se ejecuten correctamente. Y también lo es el archivo “**sim\_mod.mdl**” de SIMULINK, que se puede explorar y se puede utilizar como lo hace el programa “**pru\_ajuste**” pero que no se debe modificar.

### Función “**ah**”.

Ajuste del controlador PID aplicando las fórmulas de sintonía de Aström y Hägglund (1984) con características del proceso obtenidas en lazo cerrado.

Las posibles llamadas a esta función son:

```
kp=ah(kc,mg)
[kp,ti,td]=ah(kc,tc,mf,alfa,1)
[kp,ti,td]=ah(kc,tc,mg,alfa,2)
```

Donde:

- $k_c$  y  $t_c$  representan la ganancia crítica y el período de oscilación mantenida del proceso
- $m_f$  y  $m_g$  la especificación correspondiente de margen de fase o margen ganancia
- $\alpha$  la razón que se desea entre la constantes de tiempo derivativa e integral del controlador
- 1 ó 2, un valor entero para indicar el tipo de ajuste (1 = por Margen de Fase, 2 = por Margen de Ganancia) que se quiere realizar

## Función “amigo”.

Ajuste de los controladores PI y PID aplicando las fórmulas de sintonía AMIGO (Aström y Hägglund, 2005) con características del proceso obtenidas en lazo cerrado o en lazo abierto conocida su ganancia en estado estacionario.

Además de los parámetros de control, estas fórmulas aconsejan sobre la estructura de control: (I-P y I-PD si  $b=0$ ) o (PI y PI-D si  $b=1$ ).

Las posibles llamadas a esta función son:

```
[kp,ti,b]=amigo(k,kc,tc,1)
[kp,ti,b]=amigo(k,tp,to,2)
[kp,ti,td,b]=amigo(k,kc,tc,1)
[kp,ti,td,b]=amigo(k,tp,to,2)
```

Donde:

- $k$ ,  $t_p$  y  $t_o$ , la ganancia, la constante de tiempo y el retardo de un modelo del proceso
- $k_c$  y  $t_c$  representan la ganancia crítica y el período de oscilación mantenida del proceso

## Función “est\_cl”.

Función para estimar la ganancia crítica ( $k_c$ ) y el período de oscilación mantenida ( $t_c$ ) de un proceso a partir del análisis de una experiencia de ciclo límite con relé con histeresis.

La llamada a esta función es:

```
[kc,tc]=est_cl(t,control,salida,amplitud,epsilon)
```

Donde:

- $t$ , vector columna con los instantes de muestreo
- control, vector columna con la señal de control (entrada al proceso)
- salida, vector columna con la salida (ciclo límite) del proceso
- amplitud y epsilon, la amplitud y la histeresis del relé

Observación: la función no analiza si la salida del proceso está en ciclo límite, simplemente se limita a determinar sus características. Por tanto es el usuario quién debe garantizar la existencia de ciclo límite antes de hacer uso de esta función.

La función presupone que todas las muestras están tomadas con el mismo período de muestreo. Si no es así, el resultado de la estimación carece de validez.

## Función “est\_rla”.

Función para estimar los parámetros de un modelo de primer orden con retardo que mejor aproxima la respuesta escalón de un proceso.

Utilizando el método de los dos puntos (al 28.3% y al 63.2%).

La llamada a esta función es:

`[k,tp,to]=est_rla(t,entrada,salida)`

Donde:

- t, vector columna con los instantes de muestreo
- entrada, vector columna con la entrada escalón al proceso
- salida, vector columna con la respuesta escalón del proceso

Observación: la función no analiza si la entrada al proceso es un escalón, es el usuario quién debe garantizar que eso es así, antes de hacer uso de esta función.

La función presupone que todas las muestras están tomadas con el mismo período de muestreo. Si no es así, el resultado de la estimación carece de validez.

## Función “gm”.

Ajuste del controlador PID aplicando las fórmulas de sintonía de González y Morilla (1994) con características del proceso obtenidas en lazo abierto.

La llamada a esta función es:

`[kp,ti,td,wn]=gm(k,tp,to,delta,alfa)`

Donde:

- k, tp y to, la ganancia, la constante de tiempo y el retardo de un modelo del proceso
- delta, la especificación del coeficiente de amortiguamiento
- alfa la razón que se desea entre la constantes de tiempo derivativa e integral del controlador
- wn, la frecuencia natural que tendrá la respuesta del sistema

Observación: en el caso PID resuelve con la condición  $td=alfa*ti$  si es posible, y en caso contrario para un alfa menor del elegido pero siempre dentro del rango  $0 < alfa \leq 0.25$

## Función “hah”.

Ajuste del controlador PI aplicando las fórmulas de sintonía de Hang, Aström y Ho (1991) con características del proceso obtenidas en lazo cerrado y conocida su ganancia en estado estacionario.

La llamada a esta función es:

`[kp,ti]=hah(k,kc,tc)`

Donde:

- k, la ganancia de un modelo del proceso
- kc y tc representan la ganancia crítica y el período de oscilación mantenida del proceso

## Función “hhc”.

Ajuste del controlador PID aplicando las fórmulas de sintonía de Ho, Hang y Cao (1995) con características del proceso obtenidas en lazo abierto.

Las posibles llamadas a esta función son:

```
[kc,ti,wcf]=hhc(k,tp,to,mf,mg)
[kp,ti,td,wcf]=hhc(k,tp1,tp2,to,mf,mg,tipo_pid)
```

Donde:

- k, tp y to, la ganancia, la constante de tiempo y el retardo de un modelo del proceso
- mf y mg las especificaciones de margen de fase y margen de ganancia
- wcf, la frecuencia a la que se espera conseguir el margen de fase
- tipo\_pid, solo necesario en el caso PID, para indicar de que tipo de algoritmo PID se trata:
  - 1 = No interactivo
  - 2 = Interactivo
 pues aunque las fórmulas sólo contemplan el caso interactivo, se ha incluido las fórmulas de conversión para el no interactivo

### Función “kctc\_m1”.

Función para calcular la ganancia crítica y el período de oscilación mantenida del modelo de primer orden con retardo puro.

La llamada a esta función es:

```
[kc,tc]=kctc_m1(k,tp,to)
```

Donde:

- k, tp y to, son la ganancia, la constante de tiempo y el retardo del modelo.

### Función “sint\_ci”.

Ajuste del controlador PI o PID, aplicando las fórmulas de sintonía en lazo abierto con criterios integrales (MIAE, MISE o MITAE), para cambios en la carga o en el punto de consigna.

Las posibles llamadas a esta función son:

```
[kp,ti]=sint_ci(k,tp,to,crit_sint,tipo_cambio)
[kp,ti,td]=sint_ci(k,tp,to,crit_sint,tipo_cambio,tipo_pid)
```

Donde:

- k, tp y to, la ganancia, la constante de tiempo y el retardo de un modelo del proceso
- crit\_sint, el criterio de sintonía elegido:
  - 1 = Criterio MIAE
  - 2 = Criterio MISE
  - 3 = Criterio MITAE
- tipo\_cambio, el cambio para el que se desea ajustar el controlador:
  - 1 = Cambio en la carga
  - 2 = Cambio en el punto de consigna
- tipo\_pid, solo necesario en el caso PID, para indicar de que tipo de algoritmo PID se trata:
  - 1 = No interactivo
  - 2 = Interactivo
  - 3 = Paralelo

**Función “zn”.**

Ajuste del controlador PID aplicando las fórmulas de sintonía de Ziegler y Nichols (1942) con características del proceso obtenidas en lazo cerrado o en lazo abierto.

Las posibles llamadas a esta función son:

```
kp=zn(kc)
kp=zn(k,tp,to)
[kp,ti]=zn(kc,tc)
[kp,ti]=zn(k,tp,to)
[kp,ti,td]=zn(kc,tc)
[kp,ti,td]=zn(k,tp,to)
```

Donde:

- kc y tc representan la ganancia crítica y el período de oscilación mantenida del proceso
- k, tp y to, la ganancia, la constante de tiempo y el retardo de un modelo del proceso