

Benchmark para la Evaluación de Algoritmos de Auto-ajuste de Controladores PID

Julio A. Romero, Roberto Sanchis

Departamento de Ingeniería de Sistemas Industriales y Diseño
Universitat Jaume I
Campus del Riu Sec
E-12080 Castelló de la Plana
España
(e-mail: romeroj@esid.uji.es, rsanchis@esid.uji.es)

Resumen: En este documento se propone un benchmark para la evaluación de algoritmos de auto-ajuste de controladores PID. Para ello se definen índices cuantitativos que permiten medir el comportamiento de los algoritmos, tanto durante la realización del experimento como en la fase de control. El benchmark presentado pretende ser una herramienta que permita realizar estudios comparativos entre algoritmos de auto-ajuste. *Copyright © 2011 CEA.*

Palabras Clave: Benchmark, Auto-ajuste, PID.

1. INTRODUCCIÓN

En el presente documento se describe un benchmark para la evaluación de algoritmos de autoajuste de controladores PID desarrollado para el concurso anual organizado por el grupo de Ingeniería de Control de CEA-IFAC del curso 2010-2011. Toda la información relacionada con el concurso puede consultarse en la siguiente dirección:

<http://www.ceautomatica.es/og/ingenier%C3%ADA-de-control/benchmark-2010-11>

Es un hecho bien conocido que en la mayoría de las aplicaciones de control en la industria se utiliza el algoritmo de control PID. Actualmente muchos de los controladores PID disponibles en el mercado están provistos de algún método de auto-ajuste que, tras realizar un experimento sobre el proceso, determina de forma automática los parámetros del controlador. Esto simplifica la instalación de estos dispositivos y por tanto la puesta en funcionamiento de los bucles de control. Es por ello que la disponibilidad de un mecanismo de auto-ajuste eficiente es uno de los puntos más valorados a la hora de evaluar las prestaciones de los controladores PID comerciales. En (Leva *et al.*, n.d.) se puede ver un resumen de los métodos de auto-ajuste que implementan varios fabricantes en sus controladores PID, aunque el detalle exacto de los algoritmos es una información celosamente guardada por cada fabricante, ya que es uno de los puntos más importantes de un controlador PID comercial.

Uno de los trabajos de mayor relevancia en el auto-ajuste de controladores PID ha sido (Aström and Hägglund, 1984). Éste se basa en la idea de que la mayoría de los sistemas oscilan cuando son realimentados a través de un relé debido a la aparición de un ciclo límite. Midiendo la amplitud y periodo de las oscilaciones es posible identificar un punto de la respuesta de frecuencia del sistema y utilizar esa información para calcular los parámetros del controlador. La importancia de este trabajo viene dada por las bondades del método de realimentación con

relé respecto a otros métodos, entre los que se pueden destacar las siguientes: 1) durante el experimento el proceso está en bucle cerrado y se mantiene cerca del valor nominal de operación; 2) la amplitud de las oscilaciones se puede controlar mediante la configuración del relé. Trabajos posteriores se han centrado en mejorar la exactitud en la estimación y conseguir mejores prestaciones de los controladores. Otra estrategia bien conocida es la de realizar el auto-ajuste usando un modelo paramétrico obtenido a partir de un experimento realizado sobre el proceso. En la mayoría de los casos se asume un modelo simple, por ejemplo de primer orden con retardo de tiempo.

La relevancia alcanzada por los métodos de auto-ajuste en la aplicación práctica del algoritmo de control PID ha hecho que el desarrollo de nuevos métodos sea un área de investigación muy activa. Numerosos trabajos han sido publicados proponiendo métodos que intentan mejorar el desempeño de los sistemas controlados, ver por ejemplo (da Ma and jian Zhu, 2006; Ho *et al.*, 2003; Vivek and Chidambaram, 2005; Tan *et al.*, 2006; Arrieta *et al.*, 2010; Deya and Mudib, 2009; Leva *et al.*, 2010).

A pesar de cantidad de métodos de auto-ajuste y variaciones de estos que existen, no hay una metodología desarrollada para la evaluación de los mismos. En este documento se propone un benchmark para evaluar el desempeño de algoritmos de auto-ajuste de controladores PID. Para ello se definen índices cuantitativos que permiten medir el comportamiento de los algoritmos, tanto durante la realización del experimento como en la fase de control, valorando además el funcionamiento con procesos de muy distinta dinámica.

La estructura del documento es como se describe a continuación. En la sección 2 se abordan las consideraciones generales para la evaluación de un método de auto-ajuste, destacando aquellos aspectos que deben ser tenidos en cuenta. En la sección 3 se presentan los modelos propuestos para evaluar los algoritmos. La sección 4 describe de forma detallada los índices utilizados para la evaluación del comportamiento. En la sección

5 se describe una herramienta de simulación desarrollada para llevar a cabo la evaluación de algoritmos de autoajuste según la propuesta del benchmark. En la sección 6 se muestra un ejemplo de evaluación de un método de auto-ajuste bien conocido: el método de Ziegler y Nichols de bucle cerrado. Finalmente, en la sección 7 se presentan las conclusiones.

2. CONSIDERACIONES PARA LA EVALUACIÓN DE MÉTODOS DE AUTO-AJUSTE

En general el funcionamiento de un controlador con auto-ajuste tiene dos fases. La primera es la fase de experimento, durante la cual se obtiene información del comportamiento dinámico del sistema. Esta fase finaliza con el cálculo de los parámetros del controlador, haciendo uso de la información experimental obtenida. La segunda fase es la de control, en la que el controlador pasa a controlar el sistema para cumplir las especificaciones deseadas.

Desde un punto de vista práctico, un método de auto-ajuste debe cumplir dos requisitos:

- Lograr un ajuste adecuado del controlador y de esta forma garantizar el comportamiento deseado en la fase de control, por ejemplo minimizar el efecto de las perturbaciones. Además debe cumplir las especificaciones de robustez.
- Conseguir el requisito anterior provocando la menor excitación posible en el sistema durante la etapa de experimento. Por ejemplo, mediante un experimento corto en el cual la salida del sistema no alcance valores excesivamente alejados del valor nominal de funcionamiento.

Evidentemente, el cumplimiento de estos dos requisitos simultáneamente supone un compromiso entre uno y otro: mientras mayores limitaciones se tengan en la fase de experimento para aplicar cambios en la entrada del sistema, menor será la información de la dinámica a partir de la cual realizar el ajuste del controlador que garantice cumplir los requerimientos del sistema en bucle cerrado.

Un método de auto-ajuste será mejor en la medida que cumpla en mayor grado los dos requisitos anteriores. Esta es la base de la metodología de evaluación que se propone. En este sentido, el comportamiento de un algoritmo será evaluado atendiendo a su comportamiento en la fase de experimento, así como al desempeño del sistema controlado en la fase de control. Si I_e es el índice de comportamiento del algoritmo en la fase de experimento, e I_c es el índice de desempeño en la fase de control, el índice total del algoritmo de auto-ajuste I dependerá de ambos índices.

$$I = f(I_e, I_c) \quad (1)$$

A continuación se relacionan las características más importantes del método de evaluación que se propone.

- El comportamiento de un método de auto-ajuste es evaluado para un conjunto de modelos que representan los tipos de dinámicas más comunes en la industria de procesos.
- El criterio de desempeño que se mide en la fase de control es la atenuación de las perturbaciones en la entrada, teniendo en cuenta la robustez del bucle de control.
- El criterio de desempeño que se mide en la fase de experimento tiene en cuenta tanto la duración del experimento como la variación de la salida durante el mismo.

- El cálculo de los índices I_e e I_c se hace de forma que los resultados obtenidos para los distintos modelos sean comparables entre sí. Para ello se realiza una normalización de los índices teniendo en cuenta la dinámica concreta de cada modelo.
- La normalización del índice I_c se hace respecto de resultados de comportamiento óptimo de controladores PID publicados en (H. Panagopoulos and T. Hagglund, 2002), de manera que los valores se pueden interpretar como relativos a esos resultados.
- La normalización del índice I_e se hace respecto de los resultados obtenidos mediante el método de realimentación con relé para cada modelo, por lo que los valores del índice pueden ser interpretados como relativos a esos resultados.

En las siguientes secciones se describen más en detalle los aspectos de la metodología propuesta, los modelos utilizados para la evaluación de los algoritmos y el cálculo de los índices de comportamiento, tanto en las fases de experimento y control, como el índice global.

3. MODELOS PARA LA EVALUACIÓN

Un aspecto que debe tenerse en cuenta en la evaluación de un método de auto-ajuste es su comportamiento al ser aplicado a procesos con dinámicas diferentes. Sistemas con retardo de tiempo, de fase no mínima, con polos múltiples, con polos distintos, son algunos ejemplos de los tipos de respuestas que pueden ser encontrados en la industria. En (H. Panagopoulos and T. Hagglund, 2002) se propone un conjunto de modelos que abarcan esas características. Las ecuaciones (2)-(6) muestran esos modelos. En el proceso G_3 se han multiplicado por 10 las constantes de tiempo para que los tiempos de respuesta de todos los modelos sean similares y así poder usar un mismo periodo de muestreo en la implementación del controlador. Los métodos de auto-ajuste serán evaluados sobre todos y cada uno de estos modelos.

$$G_1(s) = \frac{1}{s(s+1)^3} \quad (2)$$

$$G_2(s) = \frac{e^{-5s}}{(s+1)^3} \quad (3)$$

$$G_3(s) = \frac{1}{(1+10s)(1+2s)(1+0,4s)(1+0,08s)} \quad (4)$$

$$G_{4,5,6,7}(s) = \frac{1}{(s+1)^\alpha} \quad \alpha = 4, 5, 6, 7 \quad (5)$$

$$G_8(s) = \frac{-2(s-2)}{(s+1)^3} \quad (6)$$

4. ÍNDICES DE COMPORTAMIENTO

4.1 Índice de la Fase de Control

En la fase de control se ha definido un índice que tiene en cuenta tanto la atenuación de las perturbaciones, mediante el valor del IAE de la salida para una perturbación de magnitud unitaria (IAE_p), como la robustez, medida mediante el máximo de la función de sensibilidad, ecuación (7).

$$M_s = \max \left\{ \frac{1}{|1 + G(j\omega)G_c(j\omega)|} \right\} \quad (7)$$

Tabla 1. Índices de referencia para la fase de control.

Modelo	$IAE_{1,4}$	IAE_2
G_1	30.44	9.579
G_2	11.01	9.062
G_3	0.23	0.059
G_4	2.407	1.377
G_5	4.060	2.839
G_6	5.600	4.250
G_7	7.156	5.614
G_8	8.395	6.161

donde $G(j\omega)$ y $G_c(j\omega)$ son las funciones de transferencia del proceso y del controlador respectivamente.

El valor del índice IAE_p depende de la dinámica del sistema, por lo que para poder comparar el comportamiento de un mismo método de auto-ajuste en diferentes modelos es necesario normalizar dicho valor. Dicha normalización se realiza con respecto al comportamiento que se consigue con un ajuste óptimo del controlador PID que minimiza el IAE de la respuesta ante perturbación. En concreto se utilizan los resultados que aparecen en (H. Panagopoulos and T.Hagglund, 2002), donde los parámetros del PID con función de transferencia dada por la ecuación (8) se obtienen mediante la solución numérica de un problema de optimización con restricciones para cada modelo representado por las ecuaciones (2)-(6).

$$G_c(s) = K_p \left(1 + \frac{T_{ds}}{1 + \frac{T_{ds}}{N'}} + \frac{1}{T_{is}} \right) \quad (8)$$

En ese artículo aparecen los valores del índice IAE de la perturbación para diseños con $M_s = 1,4$ y $M_s = 2$: $IAE_{1,4}$ e IAE_2 , Tabla 1. Los valores del modelo G_3 simplemente se han multiplicado por 10, al haber escalado las constantes de tiempo en ese valor.

Para dicha normalización se pueden usar $IAE_{1,4}$ o IAE_2 . Si se elige IAE_2 , entonces el valor normalizado del IAE de la respuesta ante perturbación se calcula como:

$$IAE_p^* = \frac{IAE_p}{IAE_2} \quad (9)$$

A partir de M_s e IAE_p^* el índice para evaluar la fase de control queda expresado por la ecuación (10), donde M_s aparece elevado al cuadrado para reflejar el hecho de que pequeñas variaciones de M_s provocan modificaciones importantes en la respuesta del sistema.

$$I_c = \gamma_1 M_s^2 + \gamma_2 IAE_p^* \quad (10)$$

Los coeficientes γ_1 y γ_2 se calculan para cada modelo de forma que el plano definido por la ecuación (10) pase por los puntos:

$$(M_s^2, IAE_p^*, I_c) = \{(0, 0, 0); (1,4^2, IAE_{1,4}/IAE_2, 1); (2^2, IAE_2/IAE_{1,4}, 1)\}$$

como se muestra en la figura 1. La línea recta que pasa por los puntos $(1,4^2, IAE_{1,4}/IAE_2, 1)$ y $(2^2, IAE_2/IAE_{1,4}, 1)$, representa el lugar de los puntos en los que $I_c = 1$. Si $I_c \approx 1$, entonces los resultados conseguidos por el método de auto-ajuste en la etapa de control se aproximan a los resultados

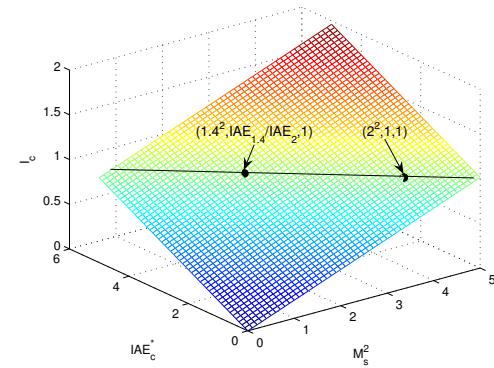


Figura 1. Plano que define el comportamiento de un método de auto-ajuste en la fase de control.

conseguidos mediante la optimización numérica planteada en (H. Panagopoulos and T.Hagglund, 2002). Cuanto mayor sea el valor de I_c , menos cerca estará el controlador ajustado del comportamiento óptimo. Los controladores diseñados pueden tener cualquier valor de M_s (el índice valora el compromiso entre la robustez y el rendimiento conseguido).

4.2 Índice de la Fase de Experimento

Como se ha comentado anteriormente, en la fase de experimento se debe tener en cuenta la duración del experimento, T_e , y las variaciones de la salida del sistema respecto del valor nominal de operación. Una medida que tiene en cuenta tanto esas variaciones como el tiempo de duración es el valor del IAE de la salida durante el experimento, IAE_e .

El valor de IAE_e depende de la dinámica del sistema. En el conjunto de modelos utilizados para la evaluación hay sistemas con muy diversas velocidades de respuesta. Para poder comparar el comportamiento de un mismo método de auto-ajuste para varios modelos, el índice de comparación debe ser relativo a la dinámica de cada modelo. Esto se puede conseguir mediante la normalización del IAE_e respecto del valor de IAE obtenido para un experimento que se pueda considerar como referencia y sea común a todos los modelos.

Se propone como experimento de referencia el de realimentación con relé de amplitud unitaria, de forma que el IAE de referencia (IAE_R) sea el IAE de un periodo de oscilación en dicho experimento.

Los valores de IAE_R (y del periodo de oscilación T_R) obtenidos en estas condiciones para los modelos (2)-(6) se muestran en la Tabla 2. Respecto de estos valores se puede normalizar los valores de IAE_e que se obtienen para cada modelo con el método de auto-ajuste que se deseé evaluar, quedando el índice de la fase de experimento como:

$$I_e = IAE_e^* = \frac{IAE_e}{IAE_R} \quad (11)$$

Consideración de ruido de medida Para hacer una evaluación realista de un método de auto-ajuste hay que incluir un ruido de medida en la salida del sistema. En el caso ideal de ausencia de ruido de medida, las variaciones de la salida durante la fase de experimento, y por tanto el IAE_e , se pueden hacer tan pequeñas como se quiera: basta con aplicar entradas con

Tabla 2. Índices de referencia para la fase de experimento.

Modelo	IAE_R	T_R
G_1	10.52	11.08
G_2	10.08	15.34
G_3	0.18	6.0
G_4	1.306	6.32
G_5	2.445	8.68
G_6	3.720	10.90
G_7	5.052	13.02
G_8	5.733	7.98

Tabla 3. Amplitud del ruido usado para cada modelo.

Modelo	A_r
G_1	0.075
G_2	0.0485
G_3	0.0025
G_4	0.0165
G_5	0.0205
G_6	0.027
G_7	0.031
G_8	0.061

amplitudes infinitesimales. En la práctica, el ruido de medida limita los valores mínimos de la entrada, ya que la relación entre la salida del sistema y el ruido de medida debe permitir extraer la información de la dinámica necesaria para el ajuste del controlador.

Teniendo en cuenta lo anterior, el ruido de medida para cada modelo se ha fijado de manera que en los experimentos de referencia la relación entre la salida del sistema y el ruido de medida sea la misma para todos los modelos. Concretamente, la amplitud del ruido de medida se calcula mediante la siguiente ecuación:

$$A_r = \frac{A_R}{\beta}; \quad (12)$$

donde A_r es la amplitud del ruido de medida para cada modelo, A_R es la amplitud de las oscilaciones en los experimentos de referencia para cada modelo (con amplitud de relé unitaria) y β es la relación entre A_R y A_r .

En los experimentos de referencia se ha considerado $\beta = 10$, que es un valor razonable de relación señal/ruido que hace que la respuesta del sistema no quede oculta por el ruido. Los valores de A_r para cada modelo se muestran en la Tabla 3.

4.3 Índice Global

Se propone calcular el índice global de la fase de control y de la fase de experimento como la media geométrica de los índices para cada proceso:

$$I_{c_total} = \sqrt[8]{\prod_{j=1}^8 I_{c_j}} \quad (13)$$

$$I_{e_total} = \sqrt[8]{\prod_{j=1}^8 I_{e_j}} \quad (14)$$

Finalmente, se propone combinar los índices de la fase de control y la de experimento para calcular el índice global del benchmark, que tendrá la expresión:

$$I_{total} = I_{c_total} \sqrt[6]{I_{e_total}} \quad (15)$$

La raíz sexta en el índice de experimento se propone para favorecer en la evaluación a aquellos métodos de auto-ajuste que consigan una mejor respuesta del sistema controlado.

Además de calcular el índice global I_{total} que será usado para la evaluación del algoritmo, también se realiza el cálculo de índices totales para cada una de las magnitudes tenidas en cuenta para la evaluación, o sea I_{e_total} , M_{s_total} , $IAE_{c_total}^*$ y I_{c_total} . Estos se calculan como la media geométrica de los índices correspondientes para todos los modelos.

5. HERRAMIENTA PARA LA SIMULACIÓN

Para evaluar de forma sencilla un algoritmo de autoajuste según el benchmark propuesto, se ha desarrollado una herramienta de simulación que permite realizar los experimentos y calcular los índices de comportamiento de forma automática. La herramienta ha sido desarrollada en Matlab y consta de varios elementos, los cuales se describen a continuación.

La implementación del algoritmo de auto-ajuste a evaluar se realiza en una función (*algoritmo.m*) que es llamada por un bloque *Embedded Matlab Function* integrada en un modelo de Simulink (*simulador_autoajuste.mdl*). El periodo de ejecución de esa función es de 10 ms. En cada periodo la función dispone de una medición de la salida del proceso, y modifica la entrada de control a través de un retenedor de orden cero.

Cada periodo la *Embedded Function* llama a la función *algoritmo.m*, pasándole como parámetros el valor de la medición de la salida y el tiempo. La función por su parte devolverá el valor de la acción de control para el proceso, los parámetros del controlador PID (K_p , T_d , T_i , N), y una variable booleana que indica si la fase de experimento continúa (1) o ha terminado (0).

Si la variable booleana devuelta por la función *algoritmo.m* vale 1, la *Embedded Function* no hará caso de los parámetros del PID, y aplicará al proceso la acción de control devuelta por la función. Por el contrario, si la variable booleana devuelta por la función *algoritmo.m* vale 0, se termina la fase de experimento y empieza la fase de control, por lo que dicha función ya no volverá a ser llamada. Los parámetros del PID devueltos en ese momento por la función *algoritmo.m* son los definitivos, definiéndose el controlador PID discreto que se aplicará a partir de ese momento al periodo fijo de 10 ms.

En la fase de experimento, el algoritmo del usuario puede definir la acción de control de forma arbitraria. Una vez en la fase de control, la acción de control quedará determinada por el algoritmo PID discreto predefinido que es una aproximación de Euler del PID con filtro de derivador dado por la ecuación (8).

La simulación se realizará partiendo de condiciones iniciales nulas. En el instante $t=250$ se introduce una perturbación a la entrada del proceso en forma de escalón unitario. El algoritmo debe finalizar el cálculo de los parámetros del controlador antes de ese instante, y comunicar la finalización de la fase de experimento devolviendo un cero en la variable booleana.

Desde un fichero llamado *evalua_benchmark_auto_pid.m* se recorren los 8 modelos del conjunto de prueba, y se llama a la simulación para realizar el experimento, dando como resultado final unas gráficas con los índices parciales, y el valor del índice total.

Todos los ficheros antes mencionados están disponibles en la siguiente dirección url, donde se exponen además las condiciones de participación en el concurso organizado por el grupo de Ingeniería de Control de CEA-IFAC en el curso 2010-2011, basado en este benchmark:

<http://www.ceautomatica.es/og/ingenier%C3%ADA-de-control/benchmark-2010-11>

Además está la función *algoritmo_base.m* que sirve de base para programar el algoritmo de auto-ajuste. Se debe cambiar su nombre a *algoritmo.m* para que el modelo que realiza la simulación la pueda llamar. En la función *algoritmo.m* está programado el algoritmo de Ziegler y Nichols de bucle cerrado, como ejemplo de algoritmo de auto-ajuste.

Por último destacar que se ha comprobado el funcionamiento de la herramienta de simulación en las versiones de MATLAB 7.0 de 2004 y 7.7 de 2008b.

6. EJEMPLO

Para ilustrar y servir de ejemplo se ha programado el algoritmo de auto-ajuste de Ziegler-Nichols, basado en la realimentación con relé según (Aström and Hägglund, 1984). Concretamente se ha implementado este método considerando dos valores de amplitud del relé distintas: $u = 0,5$ y $u = 1$. Teniendo en cuenta que el ruido para cada modelo es el dado en la Tabla 3, los valores de $u = 0,5$ y $u = 1$ implican relaciones señal/ruido (*SR*) en la salida del sistema durante el experimento de 5 y 10 respectivamente.

El ajuste de los controladores se ha realizado a partir de los valores de la amplitud (A_o) y periodo (T_o) de las oscilaciones. A_o y T_o se calculan cuando el sistema alcanza oscilaciones sostenidas de amplitud constante, concretamente cuando se cumple la siguiente condición:

$$A_k - A_{k-1} < h \quad (16)$$

donde A_k y A_{k-1} son las amplitudes de los ciclos de oscilación actual y anterior respectivamente y h es la histéresis del relé, que se ha fijado en un valor algo mayor que la mayor amplitud del ruido de medida de los 8 procesos: $h = 0,2 > A_{r,max}$.

En el momento en que se cumple la condición (16) la fase de experimento se da por concluida y se calculan los parámetros del controlador.

En las figuras 2 y 3 se muestran los resultados de los índices obtenidos en cada caso, tal y como los muestra la aplicación de Matlab que realiza la evaluación. En la primera columna se muestran los índices del experimento. En la primera gráfica, el tiempo de duración del experimento normalizado al experimento de referencia ($T_e^* = \bar{T}_e/T_R$). En la segunda gráfica, el valor medio absoluto de la salida normalizado (IAE_e^*/T_e^*), y finalmente en la tercera gráfica el valor del índice de la fase de experimento ($I_e = IAE_e^*$), tanto para cada proceso como el total. En la segunda columna se muestran los índices de la fase de control. En la primera figura el valor de M_s , en la segunda el valor del IAE ante perturbación normalizado, y en la tercera, el valor del índice de la fase de control, I_c , así como el valor del índice total. Estas gráficas pueden servir de ayuda para mejorar el algoritmo.

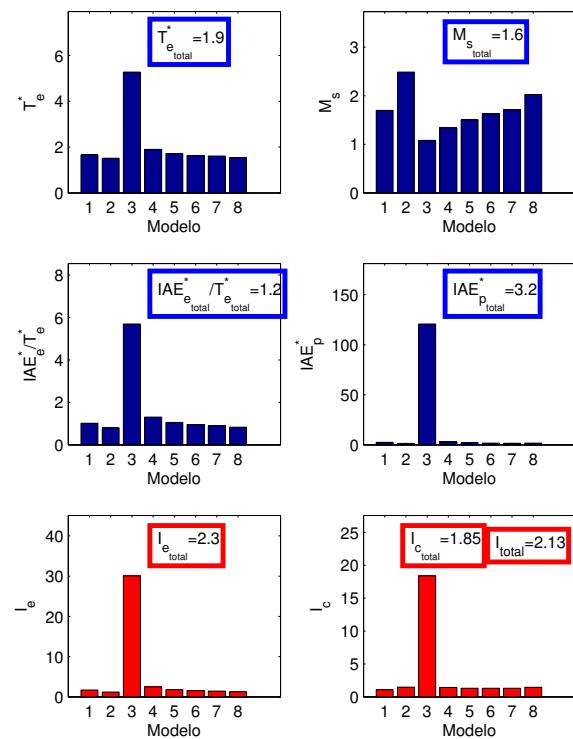


Figura 2. Resultados de los índices obtenidos para un experimento con relé con amplitud de las oscilaciones 5 veces la amplitud del ruido de medida fijado para cada modelo.

Como se ve, para el método con $SR = 10$ el índice total de la fase de experimento (I_e) es mayor que en el caso de $SR = 5$, por ser mayor la amplitud de las oscilaciones.

Sin embargo, los índices de la fase de control son menores en el caso de $SR=10$. Esto se debe a que para valores de $SR=5$ el ruido enmascara la salida real del sistema, lo cual induce errores en la estimación de A_o y T_o . Para $SR=10$ es posible estimar de forma más exacta los valores de A_o y T_o a partir de los cuales se calculan los parámetros del PID.

Respecto del índice total, éste es menor para el caso de $SR=10$ debido a la forma en que se ponderan en la ecuación final los índices de experimento y de control (se da más peso al índice de la fase de control).

7. CONCLUSIONES

En este documento se propone un benchmark para la evaluación de algoritmos de auto-ajuste de controladores PID. La metodología de evaluación propuesta en el benchmark se basa en la definición de índices para medir el comportamiento de los algoritmos tanto en la fase de experimento como en la de control. El benchmark presentado puede ser de utilidad para realizar estudios comparativos entre algoritmos de auto-ajuste. Se ha desarrollado una herramienta para la simulación, con el objetivo de facilitar la aplicación del benchmark para la evaluación de nuevos algoritmos de auto-ajuste de controladores PID.

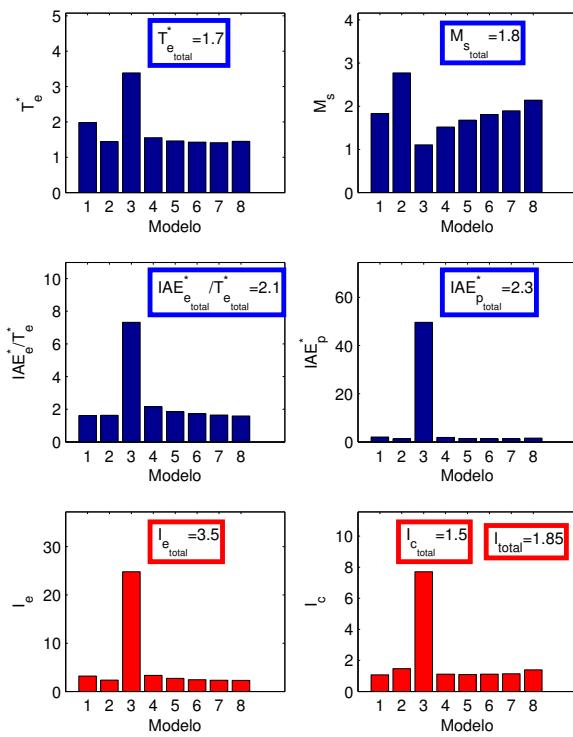


Figura 3. Resultados de los índices obtenidos para un experimento con relé con amplitud de las oscilaciones 10 veces la amplitud del ruido de medida fijado para cada modelo.

REFERENCIAS

- Arrieta, O., A. Visioli and R. Vilanova (2010). Pid autotuning for weighted servo/regulation control operation. *Journal of Process Control* **20**, 472–480.
- da Ma, Ming and Xin jian Zhu (2006). A simple auto-tuner in frequency domain. *Computers & Chemical Engineering* **30**(4), 581–586.
- Deya, Chanchal and Rajani K. Mudib (2009). An improved auto-tuning scheme for pid controllers. *ISA Transactions* **48** (2009), 396–409.
- H. Panagopoulos, K. J. Åström and T. Hagglund (2002). Design of PID controllers based on constrained optimization. *IEE Proc.-Control Theory Appl.* **149**(1), 32–40.
- Ho, W. K., Y. Hong, A. Hansson, H. Hjalmarsson and J. W. Deng (2003). Relay auto-tuning of pid controllers using iterative feedback tuning. *Automatica*, **39**(1), 149–157.
- Leva, A., C. Cox and A. Ruano (n.d.). Hands-on pid autotuning: a guide to better utilization.
- Leva, Alberto, Sara Negro and Alessandro Vittorio Papadopoulos (2010). Pi/pid autotuning with contextual model parametrisation. *Journal of Process Control* (20), 452–463.
- Åström, K. J. and T. Hägg (1984). Automatic tuning of simple regulators with specifications on phase and amplitude margins. *Automatica* **20**(5), 645–651.
- Tan, K. K., T. H. Lee, S. Huang, K. Y. Chua and R. Ferdous (2006). Improved critical point estimation using a preload relay. *Journal of Process Control*, **16**(5), 445–455.
- Vivek, Sathe and M. Chidambaran (2005). An improved relay auto tuning of pid controllers for unstable foptd systems. *Computers & Chemical Engineering*, **29**(10), 2060–2068.