

# Diseño experimental de un regulador PID de bajo costo para complementar el aprendizaje en control de procesos

Cristian Guillén<sup>a</sup>, Alejandro Ramos<sup>b</sup> y Juan José Santana<sup>c</sup>  
Dpto. Ingeniería de Procesos, Universidad de Las Palmas de Gran Canaria, 35017 Campus  
Universitario de Tafira. España.

## RESUMEN

En este trabajo se presenta un diseño experimental de bajo costo para el *aprendizaje* de sistemas de *control PID*. Para el diseño del regulador empleado se ha recurrido a un *sistema embebido* (Arduino Nano) encargado del control del sistema, y “Processing” un software libre, para el desarrollo de una interfaz que facilite la comunicación con el microcontrolador, así como la recepción y almacenamiento de las variables de proceso. El tratamiento de datos se realiza con Scilab, así como la simulación del lazo de control. Este regulador de bajo costo y “libre” ha sido desarrollado para ayudar a los estudiantes de ingenierías a comprender con mayor profundidad el control PID de procesos químicos. Además, este regulador está basado en un sistema de software y hardware libre (open-source), siendo ésta su característica más importante, ya que permite a cualquier usuario realizar modificaciones en el diseño, así como sustituir sensores y otros componentes con facilidad. Este método consigue preparar al alumno a enfrentarse a un caso real, así como conocer y aprender a emplear herramientas necesarias para sintonizar un controlador PID. Por otro lado, permite al usuario probar diferentes configuraciones del PID en el regulador. En un principio, se ha diseñado para ser implementado en un sistema de cracking térmico de aceite de ricino, sin embargo, se ha probado su estabilidad en un sistema de baño maría (agua ó aceite). Se ha considerado esta opción ya que los requerimientos de material y sustancias, se pueden encontrar con facilidad en cualquier laboratorio docente.

## PALABRAS CLAVE:

Diseño experimental de bajo costo, sistema embebido, control PID, Método de la Curva de Reacción.

## 1. INTRODUCCIÓN

Este artículo es una pequeña parte del proyecto final de carrera de la Licenciatura en Ingeniería Química, Escuela Técnica Superior de Ingenieros Industriales de Canarias, departamento de Ingeniería de Procesos. Con la metodología propuesta se persigue mejorar la formación de estudiantes de enseñanzas superiores en control de procesos, fundamentalmente ingenierías industriales, especialidad en procesos químicos, ingenieros de automatización, ..., Además, se emplean software de código abierto, facilitando la incorporación de nuevos usuarios, ya que además, todos los software empleados son multiplataforma.

Más de la mitad de los controladores industriales que se usan hoy en día emplean esquemas de control PID o PID modificado. El algoritmo de control PID<sup>7</sup>:

$$m(t) = Kc \cdot \left( e(t) + \frac{1}{\tau_I} \int e(t) dt + \tau_D \cdot \frac{de(t)}{dt} \right) \quad (1)$$

La mayor parte de los controladores PID se ajustan en el lugar de emplazamiento. La utilidad de los controles PID estriba en que se aplican de forma casi general a la mayoría de los sistemas de control. En particular, cuando el modelo matemático del proceso no se conoce y, por lo tanto, no se pueden emplear métodos de diseño analíticos, es cuando los controles PID resultan más útiles. En el campo de los sistemas para control de procesos, es un hecho bien conocido que los esquemas de control PID básicos y modificados han demostrado su utilidad para aportar un control satisfactorio. Para el desarrollo del sistema de control empleamos 3 herramientas de software libre, **Arduino, Processing y Scilab**.

**Arduino** en realidad es 3 cosas: una placa de hardware libre y, un software y un lenguaje de programación libre. Este tipo de placas embebidas se están utilizando con mayor frecuencia en proyectos de investigación, así como proyector para

<sup>a</sup> cristianiqu@gmail.com, <sup>b</sup> alejandro.ramos@ulpgc.es, <sup>c</sup> juan.santana@ulpgc.es

mejorar la calidad educativa fundamente en enseñanzas superiores, fundamentalmente en diversas ramas de la ingeniería. Gracias a su bajo costo y versatilidad que ofrece, lo hacen una herramienta muy adecuada para el desarrollo de prototipos a nivel de laboratorio, con unas muy buenas prestaciones. Tanto es así, que cada vez se están realizando una cantidad mayor de proyectos con este tipo de dispositivos, así como el desarrollo de dispositivos innovadores presentados en multitud de congresos a nivel mundial en la actualidad. Cabe destacar algunos de los siguientes artículos de revistas, así como actas de algunas conferencias recientes en la que aparecen proyectos con Arduino. Con fines educativo: Artículo de enfoque para estudiantes, con el fin de introducirlos fácilmente en el modelado y desarrollo de un sistema real con arquitectura VoIP<sup>5</sup>; Diseño e implementación de un constructor con un enfoque educativo para mejorar la enseñanza y aprendizaje de procesos con sistemas embebidos y comunicación vía Wireless<sup>1</sup>. Artículos de investigación de diseños basados en Arduino: Medida del flujo cardíaco usando un Arduino Uno<sup>8</sup>; Sistema portátil basado en un sistema embebido (Arduino) para medición de la impedancia en sistemas biológicos<sup>2</sup>; Sistema versátil y de bajo costo para adquisición de datos en sistemas ecológicos basado en Arduino<sup>3</sup>.

**Processing.** La función del IDE de Processing es crear un sketch (código de programación) de forma fácil y rápida, para lo que se vale de las propias funciones que vienen por defecto en el software, así como el uso de librerías que amplían de forma considerable la potencia del software. Además, permite exportar nuestros códigos a un ejecutable en cualquiera de las plataformas existentes (Mac, Windows, Linux), y como no es de código abierto.

**Scilab.** Es un software de código abierto para análisis numérico, con un lenguaje de programación de alto nivel para cálculo científico. Es desarrollado por Scilab Enterprises, bajo la licencia CeCILL, compatible con la GNU General Public License. Las características de Scilab incluyen análisis numérico, visualización 2-D y 3-D, optimización, análisis estadístico, diseño y análisis de sistemas dinámicos, procesamiento de señales, e interfaces con Fortran, Java, C y C++. Mientras que la herramienta Xcos permite una interfaz gráfica para el diseño de modelos dinámicos (similar a Simulink de Matlab), con el que podemos estudiar lazos de control.

## 2. MATERIALES

Para validar la validez del método se ha elaborado un regulador PID de bajo costo, en la *Figura 1* podemos ver un esquema básico de cada uno de los componentes, así como la interconexión entre ellos necesario para la elaboración del regulador.

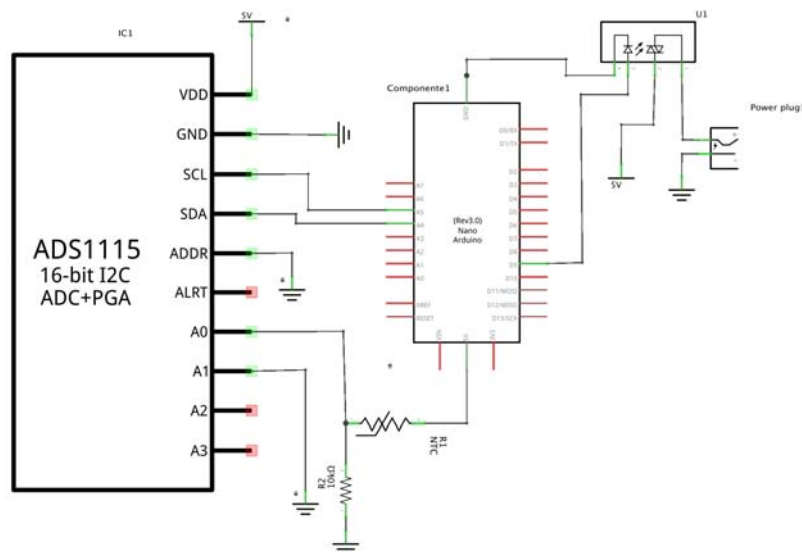


Figura 1. Esquema básico del diseño experimental.

El diseño escogido consta de los siguientes componentes:

1. Un convertidor analógico-digital ADS1115 (Texas Instrument), figura 2, esta PCB se utiliza para microcontroladores sin un convertidor de analógico-digital o cuando se desea un ADC de mayor precisión, el

ADS1115 proporciona una precisión de 16 bits. Este ADC puede funcionar de 2V a 5V, se puede medir una amplia gama de señales y gracias a la librería que proporciona el fabricante es fácil de usar. La conexión se realiza a través de I2C. Además, se pueden tener hasta 4 convertidores conectados al mismo bus I2C, lo que supondría un total de 16 entradas.



Figura 2. Conversor analógico-digital, ADS1115.

2. Microcontrolador ATmega 328P (Atmel) implementado en un sistema embebido, llamado Arduino Nano. Este controlador es el encargado de realizar el control PID de temperatura. Además, se encarga de capturar datos en función de la frecuencia de muestreo especificada, para posteriormente enviarlos al PC vía serial, para almacenarlos en un archivo.

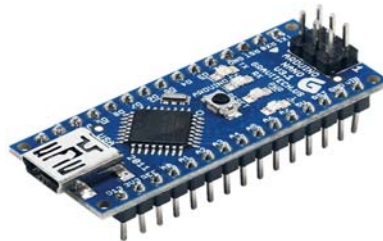


Figura 3. Microcontrolador ATmega 328P (Atmel). Arduino Nano R3.

3. Relé de estado sólido Omron G3MB. Este dispositivo conmuta el paso de la electricidad cuando una pequeña corriente es aplicada en sus terminales de control. Se emplea un relé OMROM que se caracteriza por la baja tensión necesaria en la puerta, 5V. Lo que supone una ventaja ya que el Arduino trabaja en 5V. Con el relé controlamos la potencia que se aporta al sistema.



Figura 4. Réle de estado sólido Omrom G3MB.

4. Termistor de inmersión, NTCAIMME3C90373 (Vishay Components). Sensor de temperatura por resistencia. Su funcionamiento se basa en la variación de la resistividad que presenta un semiconductor con la temperatura. Este tipo de termistor esta diseñado para estar en contacto con agua y otros fluido, se puede sumergir perfectamente.



5. Ordenador (PC). El PC se emplea como una herramienta para controlar y configurar el micro-controlador en cada momento, así como almacenar los datos obtenidos del mismo, correspondiente a los valores de las variables medidas. La comunicación entre PC(Processing) y el micro-controlador (Arduino) se realiza vía serial. Se ha elaborado un sketch en Processing (lenguaje basado en java, y de código abierto) para satisfacer estas necesidades.

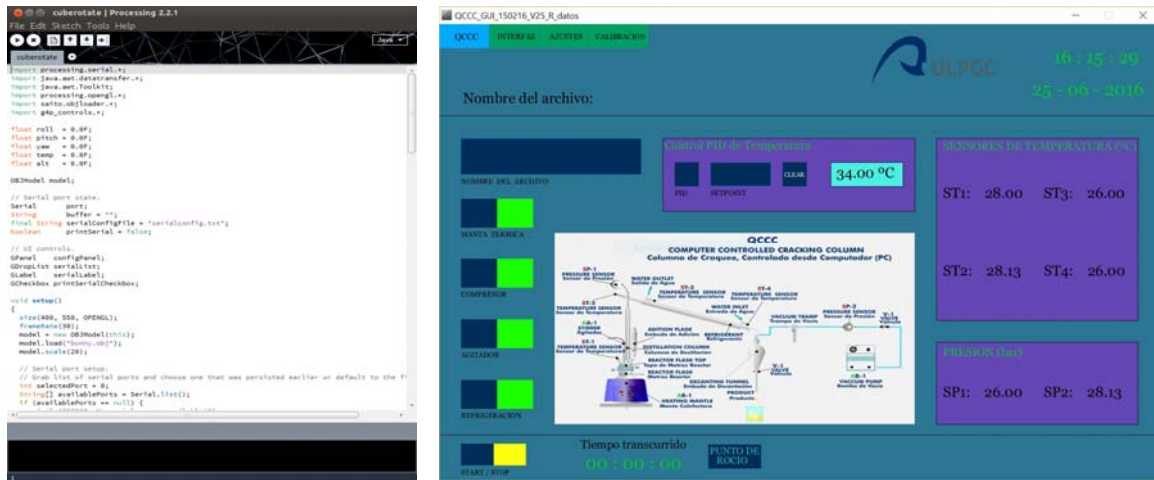


Figura 6. IDE de Processing e Interfaz generada mediante el sketch elaborado.

### 3. METODOLOGÍA

Para facilitar la implementación de este diseño experimental se ha considerado un calentador de agua, en el que se quiere controlar la temperatura en todo momento.

La metodología de trabajo pasa por una serie de etapas:

1. **Identificación del sistema**, Método de la Curva de Reacción<sup>7</sup>. Éste método es uno de los más ampliamente empleados, ya que es sencillo de aplicar y se obtienen buenos resultados en cuanto al modelo del sistema. Éste método sólo se puede aplicar a procesos cuya respuesta ante un entrada escalón sea de tipo sigmoideal, siendo ésta la respuesta más frecuente de la mayor parte de procesos químicos.

Para la obtención del modelo se aconseja seguir el siguiente procedimiento:

- Instalamos el regulador y lo conectamos al PC vía USB.
- Luego ejecutamos el sketch de Processing y comprobamos que tenemos total control sobre el sistema.
- Con el controlador en manual ejercemos una acción de control hasta alcanzar una temperatura de 50 °C aproximadamente. Ésta se debe de mantener constante +/-1°C antes de continuar.
- Aumentamos manualmente la potencia aportada, incrementándola entre un 10-30% del valor anterior de estado estacionario, manteniéndola constante hasta que alcance el nuevo estado estacionario.

Se ha elaborado un script para Scilab que nos permite realizar la representación y análisis de los datos obtenidos.

2. **Análisis de datos obtenidos en el ensayo de identificación**. Los datos obtenidos de forma empírica se almacenan en un archivo de texto con extensión “.txt”. En este archivo cada columna pertenece a una variable diferente. Posteriormente, se abre esta matriz de datos (archivo de texto) con Scilab, con el fin de realizar un tratamiento de los datos. Se ha realizado un script en Scilab con el fin de automatizar el tratamiento de datos. Así, con sólo introducir el nombre del archivo, podemos representar gráficamente los datos empíricos, así como ajustar estos datos a un modelo establecido. Se ha considerado ajustar los datos al siguiente modelo<sup>6</sup>, ya que la respuesta del proceso es del tipo “s tendida” :

$$y(t) = A * \left( 1 - e^{-B*t} * \left[ \frac{\sin(D*t+c)}{\sin(C)} \right] \right) \quad (2)$$

Para ajustar los datos experimentales al modelo propuesto, empleamos la función “fminsearch” de Scilab que nos permite minimizar el error entre el modelo propuesto y los datos experimentales, ajustando así los parámetros del modelo (A, B, C y D). Por consiguiente, obtenemos mediante análisis matemático los valores de los parámetros del modelo que hacen que minimice el error con los datos obtenidos mediante el ensayo.

3. A partir del ajuste realizado en el paso anterior, obtenemos los parámetros del modelo de primer orden con tiempo muerto mediante el método gráfico propuesto. Además, se realiza una **optimización** de los valores obtenidos mediante el software Scilab, con el fin de obtener unos **nuevos valores de los parámetros para el modelo de primer orden** que se ajusten mejor al modelo propuesto para la curva de datos empírica.

$$y(t) = K \cdot A \cdot \left(1 - e^{-\frac{(t-t_m)}{\tau}}\right) \quad (3)$$

4. A partir de aquí, tenemos dos conjuntos de parámetros obtenidos para un **modelo de primer orden con tiempo muerto**, y empleando ecuaciones empíricas<sup>7</sup> en control del procesos para este tipo de sistemas, obtenemos una configuración de los valores de las constantes del regulador PID distinta para cada uno. Además, mediante esta técnica se consigue una configuración estable del regulador, que podemos optimizar mediante técnicas de simulación.
5. Se **sintoniza el regulador** con cada conjunto de parámetros obtenidos en el paso anterior. Posteriormente se lleva el sistema de forma manual a una temperatura de unos 50 - 60°C aproximadamente. Luego pasamos el regulador a automático, esperamos a que alcance el estacionario unos minutos, y ejecutamos un cambio en el punto de consiga, estudiando así la respuesta del sistema. Así, tendremos la respuesta del sistema con cada conjunto de parámetros, por lo que podremos compararlas mediante un script de Scilab, elaborado con dicho propósito.

#### 4. RESULTADOS

Se realiza el *paso n° 1* de la metodología propuesta, obteniendo así la respuesta del sistema ante un cambio en escalón en la variable manipulada. Los datos obtenidos del ensayo corresponden con los círculos de color rojos de la *Figura 7*. Posteriormente realizamos el análisis de los datos con Scilab, ajustando los datos empírico según el *paso número 2* de la metodología propuesta (*expresión 1*), el ajuste corresponde a la línea negra de la *Figura 7*.

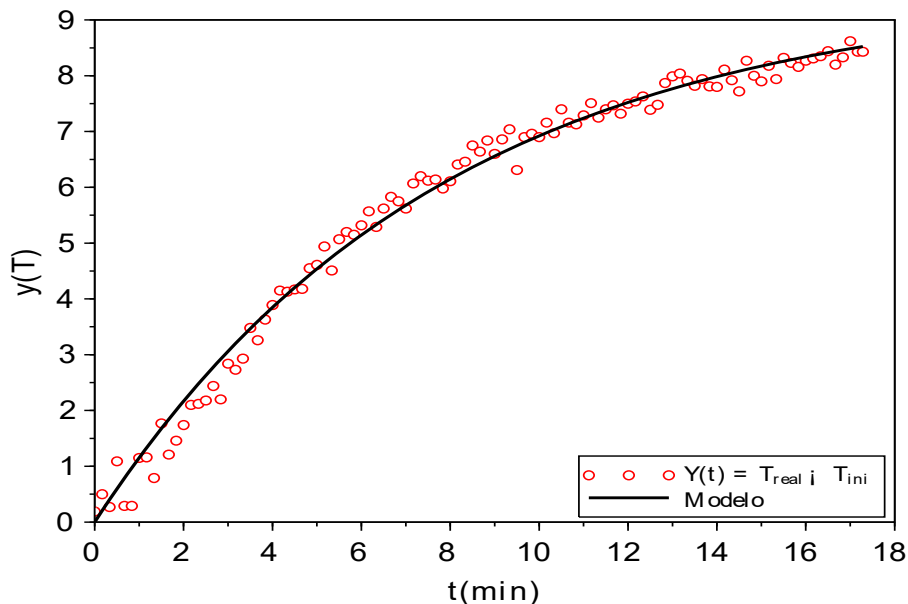


Figura 7. Ajuste de los datos empíricos al modelo planteado.

El modelo propuesto es muy complejo para la elaboración de un sistema de control. Sin embargo, nos resulta muy útil ya que se ajusta bastante bien a los datos experimentales, así podemos obtener fácilmente los parámetros del Método de la Curva de Reacción. Además, empleamos Scilab para obtener estos parámetros que caracterizan un modelo de primer orden con tiempo muerto. Este procedimiento, lejos de suponer un impedimento, supone una ventaja, ya que para el tipo de sistema seleccionado existe gran cantidad de información acerca del modelado de sistemas y diseño de reguladores, con muy buenos resultados<sup>6-7</sup>. Empleando los datos obtenidos del ajuste de los parámetros del modelo planteado (paso nº 2 de la metodología), podemos proceder al *paso nº 3*, calcular los parámetros del método de la curva de reacción.

$$\tau = \frac{3}{2} \cdot (t_{63} - t_{28}) = \frac{3}{2} \cdot (362 - 150) = 318 \text{ s} \quad t_m = t_{63} - \tau = 362 - 319 = 43 \text{ s}$$

$$K = \frac{\Delta y}{A} = \frac{8.28}{100} = 0.0828$$

Al realizar el *paso número 3* propuesto en la metodología obtenemos dos conjuntos de parámetros del modelo de la curva de reacción. Como se puede comprobar en la *Figura 8*, el conjunto de parámetros obtenidos a partir de la optimización del modelo para la curva de datos empírica, se ajusta en mejor medida a ésta. Así, se consigue un mejor ajuste (minimizamos el error) con respecto al modelo de los datos experimentales, lo que dará lugar a una configuración más estable del PID. A raíz de la realización de este paso, obtenemos los datos de la *Tabla 1*.

	Modelo de Curva de Reacción	Modelo de C. de Reacción optimizado.
Tiempo muerto $t_m$	43 s	30 s
Constante del proceso K	0.0828	0.096
Constante de tiempo $\tau$	318 s	385 s

Tabla 1. Parámetros del modelo de primer orden con tiempo muerto.

Por consiguiente, se obtienen unos valores que no varían en gran medida en ambos casos. Sin embargo, éstos últimos, los que hemos obtenido al optimizar los parámetros del modelo de primer orden, que dan lugar a una curva, que como era de esperar, se ajusta mucho mejor a los datos empíricos (*Figura 8*, línea de color rojo). Lo que conlleva, a una configuración del PID más estable, como se puede ver en la *Figura 9*.

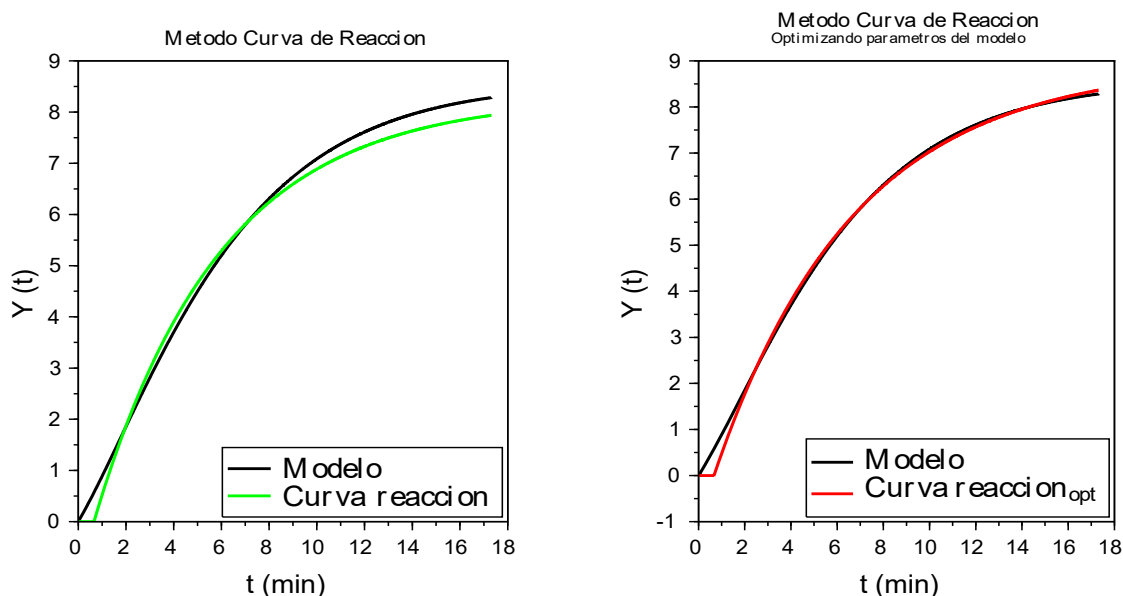


Figura 8. Comparación entre curva de reacción, y curva de reacción con parámetros optimizados.

Posteriormente, una vez obtenidos los parámetros del Método de la Curva de Reacción, siguiendo la metodología propuesta en bibliografía, y obtenidos los parámetros optimizados del modelo de primer orden empleado (paso nº 3 de la metodología), podemos proceder a obtener una configuración del PID para cada conjunto de datos, correspondiente al paso nº 4 de la metodología. Para ello se emplean las expresiones empíricas propuestas por Ziegler & Nichols<sup>7</sup>, y que se muestran en la *Tabla 2*.

		Modelo de Curva de Reacción	Modelo de C. de Reacción optimizado.
Constante proporcional (Kp)	$\frac{1}{K} \cdot \left(\frac{\tau}{t_m}\right)$	89	134
Constante integral (Ki)	$2 \cdot t_m$	1.43 min	1 min
Constante derivativa (Kd)	$0.5 \cdot t_m$	0.36 min	0.25 min

Tabla 2. Configuración del PID para cada modelo propuesto.

Una vez obtenidas ambas configuraciones del PID, realizamos el *paso número 5*, así podemos realizar una comparación entre distintas configuraciones con el fin de estudiar la estabilidad del regulador ante cambios en el punto de consigna con cada una de ellas. Esto podemos verlo en la *Figura 9*, en la que los puntos de color verde corresponden a la respuesta del sistema empleando la configuración propuesta inicialmente (parámetros sin optimizar), mientras que los puntos de color rojo corresponden a la respuesta del sistema con la configuración de los parámetros optimizados del modelo.

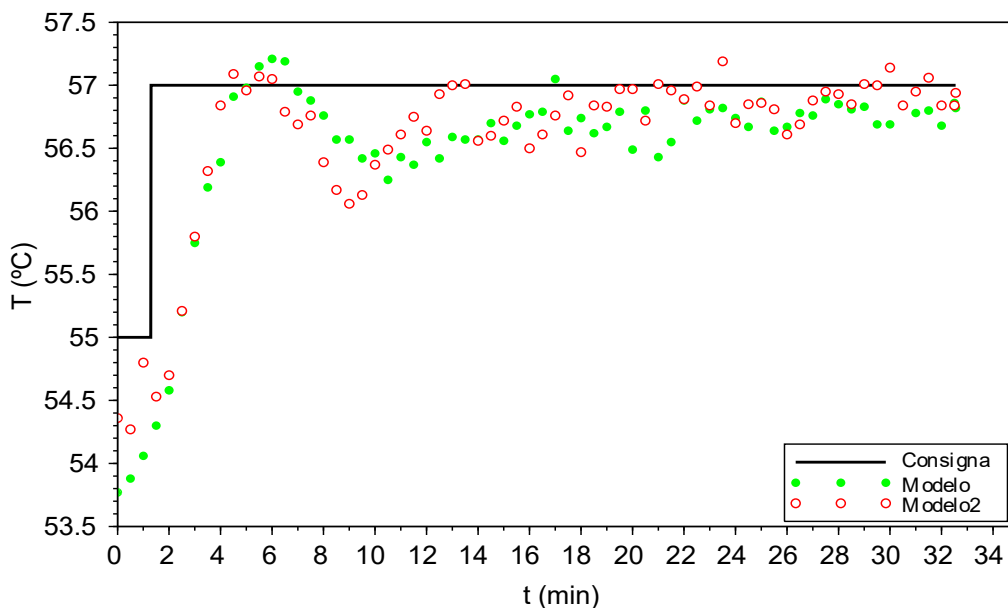


Figura 9. Comparación de comportamiento del regulador PID con ambos conjuntos de parámetros obtenidos

Como se puede observar en la figura anterior, la configuración del PID obtenida a partir de los parámetros optimizados para un modelo de primer orden con tiempo muerto da lugar a una configuración más estable del mismo. Como se puede ver, la variable de control alcanza antes la consigna, y en estado estacionario se ajusta mejor a la consigna. Por otro lado, si quisiéramos sintonizar un regulador PID, empleando técnicas de simulación, debemos utilizar criterios de sintonización como pueden ser ISE, ITAE ó IAE.

## 5. CONCLUSIONES

En este trabajo se presenta un diseño experimental de bajo costo para complementar el aprendizaje en control de procesos y reguladores PID, en el que se usa un sistema embebido (Arduino Nano). Este diseño ha sido desarrollado como una parte de un proyecto fin de carrera de la titulación de Ingeniería Química en la Universidad de Las Palmas de Gran Canaria. Además, se ha realizado el diseño de forma que sea flexible y basado en código abierto (open-source), que comprende tanto hardware como software.

Por consiguiente, el estudiante podría modificar tanto el algoritmo de control como sustituir cualquier componente por otro, con apenas realizar ciertas modificaciones en el sistema de control. Por consiguiente, este sistema puede ser empleado por el alumno no sólo para el modelado de un sistema mediante el Método de la Curva de Reacción, sino también para acercarse al mismo al uso de herramientas de análisis matemático tan potentes, como lo es Scilab. Además, se podría complementar el aprendizaje introduciendo al alumno al uso de técnicas para estudiar la estabilidad de reguladores (ISE, ITAE ó IAE), que no se comprenden en este artículo. Por otro lado, de forma indirecta, se acerca al alumno al uso de sensores (temperatura) y actuadores (relé), controlados por un sistema embebido. De esta manera se introduce al alumno al uso de herramientas muy potentes para el desarrollo de prototipos y equipos a nivel de laboratorio, favoreciendo la formación de jóvenes estudiantes, y aportando una base, que conlleve al alumno a comprender mejor como actúan los sistemas de control, así como técnicas de diseño y modelado. Como se ha podido ver en los resultados obtenidos, la metodología propuesta da lugar a la obtención de los parámetros optimizados del Método de la Curva de Reacción que se ajustan mucho mejor a los datos experimentales, dando lugar a una configuración del PID que tiene un comportamiento más estable en régimen permanente.

## REFERENCIAS

- [1] Cano, I. G. (2014). Designing and implementing a constructionist approach for improving the teaching-learning process in the embedded systems and wireless communications areas. *Computer Applications in Engineering Education* , 22(3):481-493.
- [2] Corbellini, S., & Vallan, A. (2014). Arduino-based portable system for bioelectrical impedance measurement. . *Medical Measurements and Applications (MeMeA), International Symposium* , 1-5.
- [3] Gandra, M., Seabra, R., & P.Lima, F. (2015). A low-cost, versatile data logging system for ecological applications. *Limnology and Oceanography* , Methods, 13(3):115-126.
- [4] Garcia, I., & Cano, E. (2014). Designing and implementing a constructionist approach for improving the teaching-learning process in the embedded systems and wireless communications areas. *Computer Applications in Engineering Education* , n/a.
- [5] Mekis, E., Djokic, I., Zejnelagic, S., & Matovic, A. (2015). Constructive approach in teaching of voip in line with good laboratory and manufacturing practice. *Computer Applications in Engineering Education* .
- [6] Ogata, K. (2008). *Sistemas de Control en Tiempo Discreto* (2ª Edición ed.). México: PRENTICE HALL.
- [7] Ollero de Castro, P., & Fernández Camacho, E. (2006). *Control e instrumentación de procesos químicos*. Sevilla: SINTESIS.
- [8] Pawar, P. (2014). Heart rate monitoring system using ir base sensor amp; arduino uno. *IT in Business, Industry and Government (CSIBIG)*, (págs. 1-3).