

DESARROLLO DE UN CONTROLADOR PID ACCESIBLE COMO SERVIDOR OPC

Jesús M. Zamarreño, María J. Fuente, Luis F. Acebes
Dpto. de Ingeniería de Sistemas y Automática, Universidad de Valladolid
jesusm@autom.uva.es, maria@autom.uva.es, felipe@autom.uva.es

Resumen

Se ha desarrollado un controlador PID accesible como servidor OPC utilizando el software de simulación EcosimPro. Este controlador se convierte así en un módulo estándar que puede ser utilizado por cualquier aplicación con soporte OPC que necesite la funcionalidad de un PID. En este artículo, además de exponer las características y desarrollo del PID, se muestra un ejemplo de uso desde el HMI InTouch de la empresa Wonderware.

Palabras Clave: Integración de sistemas, PID, OPC.

1 INTRODUCCIÓN

El controlador PID [1] sigue siendo uno de los controladores más populares y extendidos en la industria, debido a su buen funcionamiento en la mayoría de los procesos y a que su operación y sintonía son fáciles de entender por los operarios de planta. El algoritmo del controlador PID está incluido en la mayoría de los sistemas de control, pero hay ocasiones en que nos podemos encontrar con que el software que queremos utilizar para controlar una planta carece de dicha capacidad o que el PID que tiene implementado no cubre nuestras necesidades.

En estas situaciones, se necesita un módulo externo con la funcionalidad de un PID y que sea fácil de comunicar con nuestro software.

Hoy día, el estándar OPC [4, 6] se ha convertido en el medio más común y versátil de intercambiar datos entre sistemas de control, supervisión e incluso aplicaciones ofimáticas (e.g., Excel). Prácticamente todo software actual que pueda tener algún uso industrial es capaz de comunicarse con fuentes de datos OPC. Algunos ejemplos serían Matlab (con la OPC Toolbox), LabView, Excel, etc.

La empresa Wonderware [5] comercializa un HMI (Human Machine Interface, Interfaz Hombre Máquina) denominado InTouch caracterizado por ser una solución de supervisión HMI y SCADA abierta y extensible. Según la propia empresa, su sistema es el

más popular del mundo, siendo utilizado en una de cada tres instalaciones industriales. InTouch puede conectarse a cientos de servidores OPC para acceder así al mayor número de fuentes de datos distintas. En versiones antiguas del software se incluía un módulo denominado InControl que permitía programar lazos de control PID en el propio SCADA. Sin embargo, en versiones actuales dicho módulo ya no existe, lo cual supone un handicap para determinados proyectos de automatización.

Por lo tanto, el objetivo del presente trabajo es desarrollar un módulo servidor OPC con capacidades de cálculo PID para su uso por cualquier software con capacidad cliente OPC y que requiera dicha funcionalidad.

2 EL CONTROLADOR PID

El algoritmo de control PID digital calcula en cada periodo de muestreo T (en segundos) la salida al proceso (Output to Process), $OP(t)$, en función de la señal de error $e(t)$ calculada como la diferencia entre los valores filtrados y normalizados del Set-Point, $SP(t)$, y la variable de proceso (Process Variable), $PV(t)$.

En caso de requerirse un filtrado de las señales, se dispone de sendos filtros de primer orden con constantes de tiempo α_{SP} y α_{PV} para las señales del Set-Point y variable de proceso, respectivamente.

El rango del controlador se define mediante los límites inferior y superior tanto de PV como de SP, denominados PV_{EULO} y PV_{EUHI} . Estos límites permiten convertir las señales reales X_{EU} , en unidades de ingeniería, en señales en unidades porcentuales $X_{\%}$ o normalizadas, aplicando la ecuación 1.

$$X_{\%} = \frac{100}{(PV_{EUHI} - PV_{EULO})} (X_{EU} - PV_{EULO}) \quad (1)$$

Los tres parámetros de sintonía del controlador PID son la ganancia proporcional K_p expresada en $\%/%$, el tiempo integral T_i expresado en minutos, y el tiempo derivativo T_d expresado también en minutos.

El PID implementado dispone de dos modos de operación: ideal o no interactivo (modo = 1) y real o interactivo (modo = 0). En cualquiera de los modos, se pueden seleccionar cuatro tipos de ecuaciones de la señal de control normalizada (tipo = 1(A), 2(B), 3(C), 4(D)) denominadas tradicionalmente como ecuación A o PID (ecuación 2), ecuación B o PI-D (ecuación 3), ecuación C o I-PD (ecuación 4) y ecuación D o I (ecuación 5).

$$OP(t) = Kp \left(e(t) + \frac{1}{Ti} \int e(t)dt + Td \frac{de(t)}{dt} \right) \quad (2)$$

$$OP(t) = Kp \left(e(t) + \frac{1}{Ti} \int e(t)dt - Td \frac{dPV_{\%}(t)}{dt} \right) \quad (3)$$

$$OP(t) = Kp \left(-PV_{\%} + \frac{1}{Ti} \int e(t)dt - Td \frac{dPV_{\%}(t)}{dt} \right) \quad (4)$$

$$OP(t) = \frac{Kp}{Ti} \int e(t)dt \quad (5)$$

Dichas ecuaciones se han discretizado de acuerdo a la fórmula incremental implementada en "Allen Bradley Logix5550", siendo respectivamente:

$$OP(t) = OP(t-T) + Kp \left(\begin{array}{l} (e(t) - e(t-T)) + \\ \frac{T}{Ti} e(t) + \frac{Td}{T} (e(t) - \\ 2e(t-T) + e(t-2T)) \end{array} \right) \quad (6)$$

$$OP(t) = OP(t-T) + Kp \left(\begin{array}{l} (e(t) - e(t-T)) + \\ \frac{T}{Ti} e(t) - \frac{Td}{T} (PV_{\%}(t) - \\ 2PV_{\%}(t-T) + PV_{\%}(t-2T)) \end{array} \right) \quad (7)$$

$$OP(t) = OP(t-T) + Kp \left(\begin{array}{l} -(PV_{\%}(t) - PV_{\%}(t-T)) + \\ \frac{T}{Ti} e(t) - \frac{Td}{T} (PV_{\%}(t) - \\ 2PV_{\%}(t-T) + PV_{\%}(t-2T)) \end{array} \right) \quad (8)$$

$$OP(t) = OP(t-T) + Kp \left(\frac{T}{Ti} e(t) \right) \quad (9)$$

En modo real o interactivo (modo = 0) se usan las mismas ecuaciones, pero se realiza una transformación de los parámetros reales (Kp_r , Ti_r , Td_r) a los ideales (Kp , Ti , Td) mediante las ecuaciones 10 a 12.

$$Kp = Kp_r \left(1 + \frac{Td_r}{Ti_r} \right) \quad (10)$$

$$Ti = (Ti_r + Td_r) \quad (11)$$

$$Td = \left(\frac{Ti_r Td_r}{Ti_r + Td_r} \right) \quad (12)$$

Para evitar la saturación del término integral (antiwind-up), la señal de control normalizada calculada, $OP(t)$, está limitada en el rango 0-100%. Sin embargo, la señal de control que se envía al proceso es una señal limitada dentro del rango especificado por los valores máximos y mínimos de la señal de control en campo (OPHI y OPLO). Además, dicha señal, que se calcula cada periodo de muestreo, se reconstruye usando un ZOH (Zero Order Hold, Mantenedor de Orden Cero).

El controlador puede funcionar en modo AUTOMático o modo MANual. En modo automático (autom = 1), el usuario especifica el valor deseado de SP. En modo manual (autom = 0), el usuario indica una señal de control (OPman) en la escala 0-100%.

Si está activado el SP Tracking (SPTracking = 1), cuando el controlador está en modo manual, el valor de la señal de referencia SP sigue al valor de la señal de proceso. De modo que, cuando el controlador se vuelva a situar en modo automático, el valor de la señal de referencia coincide con el valor de la señal de proceso y así se evitan los *bumpless*.

3 EL SERVIDOR OPC DEL PID

Para desarrollar el servidor OPC que represente al algoritmo de cálculo del PID, este ha sido programado previamente en el entorno de simulación EcosimPro [2], de forma que, haciendo uso de la OPCToolbox [7] del mismo entorno, es posible generar de forma muy sencilla el servidor OPC asociado.

Un servidor OPC viene definido por el espacio de nombres que ofrece a los clientes y por la lógica de ejecución en base a los valores que tengan los items. En el caso del servidor OPC del PID, por una parte, el espacio de nombres incluirá todas las variables propias del algoritmo de control y que se han descrito en el apartado anterior (ver figura 1).

Por otra parte, en la rama Deck Commands del mismo espacio de nombres disponemos de diversas variables booleanas que permiten la ejecución del algoritmo del PID según se requiera. Básicamente, podemos ejecutar el modelo del PID en dos modos: autónomo o supervisado.

Item	Value
Deck Variables.Kp	1
Deck Variables.Ti	2
Deck Variables.Td	0
Deck Variables.T	1
Deck Variables.autom	1
Deck Variables.OPman	50
Deck Variables.OP	0
Deck Variables.SP	50
Deck Variables.PV	50
Deck Variables.modo	1
Deck Variables.tipo	1
Deck Variables.SPTtracking	1
Deck Variables.PVEUHI	100
Deck Variables.PVEULO	0
Deck Variables.OPHI	100
Deck Variables.OPLO	0
Deck Variables.alfaSP	0,1
Deck Variables.alfaPV	0,1

Figura 1: Variables del PID en el servidor OPC

En modo autónomo, el algoritmo del PID se ejecuta de forma temporizada en tiempo real. Para ello, se debe poner a TRUE el item RealTimeFlag, y posteriormente poner a TRUE el item IntegFlag. Esto hará que el servidor OPC recalcula el valor de la OP cada periodo de muestreo T de forma autónoma.

En modo supervisado, es el cliente el que decide en qué momento se recalcula el valor de la OP. Para ello, en primer lugar el valor del item Sim_CINT debe coincidir con el periodo de muestreo T. Con esta condición, cada vez que el cliente ponga a TRUE el item StepCINTFlag, el algoritmo del PID recalculará el valor de la OP.

4 EJEMPLO DESDE INTOUCH

Para mostrar la utilidad del servidor OPC desarrollado, se va a mostrar un ejemplo en el que, desde InTouch se supervisa una planta de laboratorio y se establece un control PID de la misma.

El esquema de los diversos elementos que componen el sistema integrado se muestra en la figura 2. El servidor OPC que representa al PID corresponde al desarrollo explicado en el punto anterior. El servidor OPC del proceso corresponde al servidor OPC de una tarjeta de adquisición de datos (DAQ) que establece un acceso estándar tanto a la variable de proceso (PV) como a la variable de control (OP). El HMI de InTouch, cada periodo de muestreo, realiza 4 acciones:

1. Lee del servidor del proceso la variable de proceso PV.
2. Pasa dicho valor actualizado al servidor del PID para que este actualice el error (SP-PV) y por tanto recalcula la señal de control OP.

3. Lee del servidor del PID la señal de control OP.
4. Escribe en el servidor del proceso el nuevo valor de la variable de control OP.

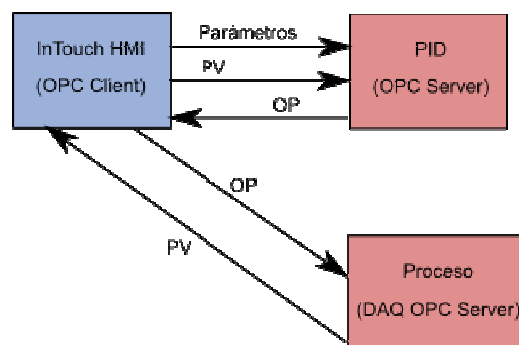


Figura 2: Integración de los 3 subsistemas

Como aplicación práctica de lo mostrado hasta aquí, vamos a considerar una planta piloto como la mostrada en la figura 3, situada en el laboratorio de Ingeniería de Sistemas y Automática de la Universidad de Valladolid.

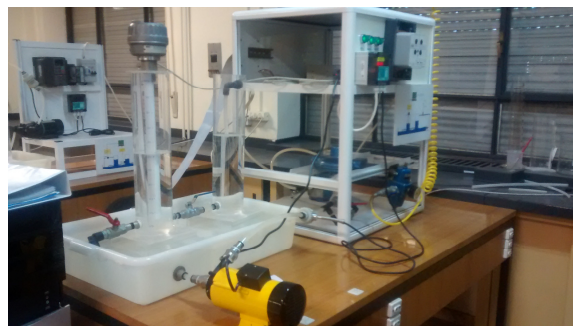


Figura 3: Planta piloto

Dicha planta consta de dos depósitos comunicantes; entrando un caudal de agua al primero de ellos y midiéndose el nivel en el segundo. Se puede regular el caudal que entra mediante una válvula de regulación. El caudal que sale del segundo depósito cae a una cubeta de donde se extrae con una bomba que aporta agua de nuevo a la entrada a través de la válvula. Por lo tanto, el sistema es autónomo en cuanto al consumo de agua. Por último, el caudal que circula por el circuito se mide a través de un caudalímetro situado entre la bomba y la válvula de regulación. En cuanto a señales, en este sistema se tienen disponibles medidas del nivel en el segundo depósito y del caudal a la entrada, y se puede actuar sobre la apertura de la válvula de regulación. Para controlar el nivel en el segundo depósito puede utilizarse un PID que, en función del nivel medido y la referencia deseada, actúe sobre la apertura de la válvula. Las señales de los dos sensores y del actuador se conectan físicamente a una tarjeta de

adquisición de datos que lleva asociado un servidor OPC de forma que, junto con el PID descrito en la sección 3, el esquema de la figura 2 puede ser llevado a la práctica.

Por tanto, sólo queda configurar InTouch para que se comuniquen con ambos servidores OPC y tenga acceso a las variables de interés (figura 4).

Name	Client Va...	Time	Client Q...	Subscr...	State	Location	Device Group
<input type="checkbox"/> OPman	-	-	-	-	Active	OPC_PID.Grupo2	OPC_PID.Grupo2
<input checked="" type="checkbox"/> modo	1	8:55:22	00C0	-	Active	OPC_PID.Grupo2	OPC_PID.Grupo2
<input checked="" type="checkbox"/> SP	20	8:49:47	00C0	-	Active	OPC_PID.Grupo2	OPC_PID.Grupo2
<input checked="" type="checkbox"/> TI	2	8:55:22	00C0	-	Active	OPC_PID.Grupo2	OPC_PID.Grupo2
<input checked="" type="checkbox"/> Td	0	8:55:22	00C0	-	Active	OPC_PID.Grupo2	OPC_PID.Grupo2
<input checked="" type="checkbox"/> OP	45,83496...	8:57:38	00C0	-	Active	OPC_PID.Grupo2	OPC_PID.Grupo2
<input checked="" type="checkbox"/> PV	20,27879...	8:57:38	00C0	-	Active	OPC_PID.Grupo2	OPC_PID.Grupo2
<input checked="" type="checkbox"/> Kp	2	8:55:22	00C0	-	Active	OPC_PID.Grupo2	OPC_PID.Grupo2
<input checked="" type="checkbox"/> autom	1	8:33:30	00C0	-	Active	OPC_PID.Grupo2	OPC_PID.Grupo2
<input type="checkbox"/> Output0	-	-	-	-	Active	OPC_PCL.Grupo_1	OPC_PCL.Grupo_1
<input checked="" type="checkbox"/> Inlet0	2388	8:57:39	00C0	-	Active	OPC_PCL.Grupo_1	OPC_PCL.Grupo_1
<input checked="" type="checkbox"/> Inlet1	1595	8:57:39	00C0	-	Active	OPC_PCL.Grupo_1	OPC_PCL.Grupo_1

Figura 4: Variables OPC accedidas desde InTouch

Y con estas variables, configurar las acciones a realizar por el HMI y desarrollar la interfaz de usuario. En las figuras 5 y 6 pueden verse dos de las pantallas desarrolladas. En la figura 5 puede verse un sinóptico de la planta junto con los principales valores asociados a la misma, y en la figura 6 la pantalla donde el usuario/operador puede cambiar los parámetros del PID así como visualizar gráficamente la evolución de las variables de interés.

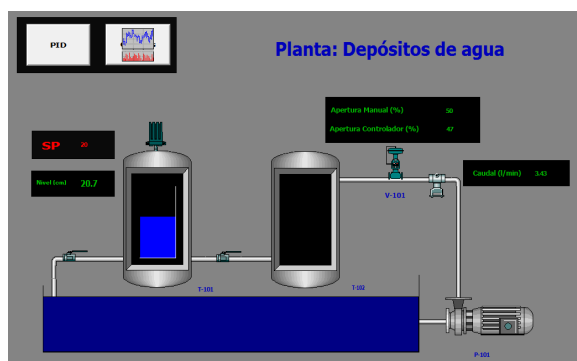


Figura 5: Sinóptico de la planta en InTouch

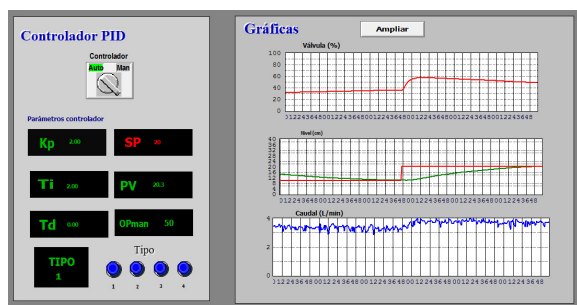


Figura 6: Parámetros del PID y gráficas de evolución

5 CONCLUSIONES

En este artículo se ha mostrado el desarrollo de un servidor OPC que es capaz de realizar los cálculos de un PID. Su desarrollo ha sido relativamente sencillo dado que, una vez programado el PID en el entorno de simulación EcosimPro, desde el propio entorno puede generarse el servidor OPC asociado.

Se ha mostrado su utilidad como módulo de soporte para todo aquel software que necesite los servicios de un PID; en concreto se ha mostrado un ejemplo de aplicación desde el HMI InTouch, de la empresa Wonderware.

El PID con soporte OPC puede ser descargado desde la página web del grupo de investigación en Control y Supervisión de Procesos de la Universidad de Valladolid [3].

Agradecimientos

Los autores agradecen al Ministerio de Economía y Competitividad y a la Unión Europea el soporte a través del proyecto DPI2012-39381-C02-02.

Referencias

- [1] Åström K.J. and Hägglund T. (2001) The future of PID control. *Control Engineering Practice*, Volume 9, Issue 11, November 2001, pp. 1163-1175.
- [2] EcosimPro (2015) <http://www.ecosimpro.com>. Online; último acceso: 19/05/2015.
- [3] GIR Control y Supervisión de Procesos (2015) Controlador PID con soporte OPC. <http://bit.ly/1SSXyqD>. Online; último acceso: 2/06/2015.
- [4] Iwanitz, F. and J. Lange (2002) OPC - Fundamentals, Implementation and Application. Hüthig, Heidelberg (Germany).
- [5] Wonderware Spain (2015) <http://www.wonderware.es>. Online; último acceso: 19/05/2015.
- [6] Zamarreño J.M. (2010) Acceso a datos mediante OPC. Andavira SL, Santiago de Compostela (Spain).
- [7] Zamarreño, J. M., Mazaeda, R., Caminero, J. A., Rivero, A. J. & Arroyo, J. C. (2014). A new plug-in for the creation of OPC servers based on EcosimPro© simulation software. *Simulation Modelling Practice and Theory*, 40, 86-94.