

SISTEMA DE PRODUCCION APLICADO
A LA SINTONIA DE UN CONTROLADOR PID

G. G. Acosta* - M. A. Mayoosky** - J. M. Catalfo***

* Secretario de Estudio de la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CICpBA)

** Secretario de Perfeccionamiento del Consejo Nacional de Investigaciones Científicas y Tecnológicas (CONICET)

*** Investigador Independiente del Consejo Nacional de Investigaciones Científicas y Tecnológicas (CONICET)

Los autores forman parte del Laboratorio de Electrónica Industrial, Control e Instrumentación (LEICI) - Facultad de Ingeniería - Universidad Nacional de La Plata. CC 91 1900-La Plata Argentina

Resumen:

La rapidez y sencillez con que las actuales herramientas para la construcción de Sistemas Expertos (shells) permiten diseñarlos, desarrollarlos y ponerlos en funcionamiento, hacen que se los utilice cada vez más en aquellas tareas que requieren de la experiencia humana para su más eficiente desempeño. Sintonizar los parámetros de un controlador tipo PID para que la respuesta en lazo cerrado con una planta con curva de reacción sigmoidea sea la especificada, es una de tales tareas. El presente artículo describe, desde el génesis hasta la simulación final en computadora, a uno de estos "controladores inteligentes".

Abstract:

The quickness and simplicity with which the nowadays tools for building Expert Systems (shells) allows to design, develop and implement them, makes that they are being used more and more in those tasks which requires from the human experience to perform more efficiently. To tune the PID controller parameters so as the response in a closed loop with a process with sigmoid reaction curve be the one specified, is just one of such tasks. The present paper shows, from the very beginning up to the final simulation by computer, one of this "intelligent controllers".

1. Introducción

Mucho se ha escrito ya sobre el ajuste de los controladores PID desde la aparición de los trabajos experimentales de Ziegler y Nichols [1] para una planta de una sola entrada y una sola salida.

Gran parte de este interés quizá reside en la practicidad, simplicidad y, por ende, gran aceptación a nivel industrial de dichos controladores. Ello se traduce en que desde hace muchos años sean vastamente utilizados en este ambiente.

No obstante, la sintonía final es siempre dejada en manos del operador experto. Un controlador así ajustado presenta la desventaja de no poder seguir las variaciones o perturbaciones que se producen en la planta bajo control, a menos que se ajusten nuevamente los valores de sus parámetros para que la respuesta temporal de todo el lazo se aproxime a la deseada. La metodología empleada por el experto del dominio es entonces observar la respuesta temporal, y en base a su propia experiencia, hacer el ajuste pertinente.

La idea central de este trabajo surge de propuestas anteriores [2], [3], y consiste en encapsular dicha experiencia en un sistema experto y aglutinarlo con un PID, para generar así lo que podríamos llamar un "controlador inteligente" con capacidad de autosintonizarse en función de la dinámica del proceso.

de alguna manera entonces un estudio de la teoría y técnicas de inteligencia artificial por un lado, y de reconocimiento de la forma de la respuesta temporal por el otro. La organización de este artículo se completa con una descripción de las reglas que constituyen la base de conocimientos y cómo se generaron, una descripción de los elementos de "hardware" y "software" utilizados (herramientas de cómputo, consulta y simulación), los resultados finales obtenidos y las correspondientes conclusiones.

2. Selección de la modalidad de programación

Cuando es posible resolver un determinado problema, pero no es posible explicar cómo o porqué se lo resuelve de esa manera y no de otra, estamos frente a una situación en la cual es aplicable, y con ventajas, un sistema experto que almacene esa experiencia heurística.

Los sistemas expertos son un subconjunto dentro de la teoría existente de Inteligencia Artificial [4], [5], para los cuales a su vez existe toda una metodología y técnicas de construcción [6], [7]. Si bien los sistemas concretados hasta la fecha distan mucho de los postulados teóricos iniciales, el estado actual del arte permite resolver mediante su utilización un sinnúmero de situaciones que estaban pendientes por carecer de un modelo adecuado.

Esta forma de razonar muy habitual en los expertos humanos, alcanza exitosamente el objetivo en más de un noventa y cinco por ciento de los casos, trabajando con datos imprecisos, inciertos y aún faltantes. Es más, la realidad cotidiana se presenta así y el cerebro humano es muy capaz de desenvolverse en ella. La contrapartida es obvia, y es el costo de pasar de un razonamiento inductivo o deductivo a un razonamiento abductivo: se pierde la infalibilidad de las nuevas conclusiones. Ese es el motivo de mayor peso a la hora de decidir entre, en el supuesto de ser un problema rigurosamente modelable (por ejemplo, matemáticamente) optar por la identificación del modelo sobre el que se va a actuar y realizar un control, ya sea adaptivo, ya óptimo o de algún tipo cuya solución compilada sea apropiada.

De todos modos hay una gran gama de situaciones en las cuales esto no es posible y aparece como interesante esta idea de atrapar el conocimiento.

Lo que se hace es, pues, anunciar los conceptos salientes para el ajuste del PID, parametrizarlos cuando sea menester y transformar este conocimiento en manipulable para la computadora. Un sencillo sistema experto con características que más adelante se detallarán es capaz de llevar adelante esta tarea.

3. Análisis de la respuesta temporal

Se observa la respuesta temporal de todo el sistema frente a una variación de la referencia o la carga en forma de escalón y se le extraen ciertos datos de interés. Esta tarea es análoga al reconocimiento de un patrón. De acuerdo a trabajos preliminares [8],[9] es posible obtener una curva del error en función del tiempo cuando la referencia varía en forma de escalón tal como:

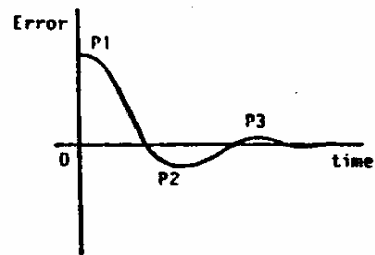


fig. 1

Análogamente con una variación en la carga:

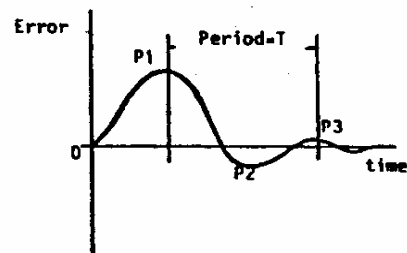


fig. 2

calcular el sobrepaso y amortiguamiento:

$$S_p = \frac{|P_z|}{|P_t|} \quad (1)$$

$$\xi = \frac{|P_z| + |P_a|}{|P_z| + |P_t|} \quad (2)$$

y observar el periodo natural T.

Este error, correspondiente a la respuesta total de la planta en lazo cerrado, es el patrón que habrá que modificar para que la respuesta sea satisfactoria. Esta puede optimizar alguna característica del error (ITAE, IAE, ISE) o conseguir una respuesta del tipo 'quarter decay' que es de preferencia en algunos casos [10].

4. Arquitectura del lazo experimental

Esquemáticamente el lazo puede verse en la sig. fig. 3:

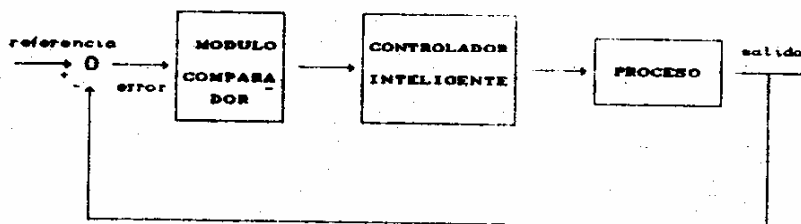


fig. 3

El proceso o planta a controlar pertenece a la familia representada por:

$$G_p(s) = \frac{k * e^{-s\tau}}{1 + a*s} \quad (3)$$

donde:

- k = ganancia de la planta
- τ = retardo puro de tiempo
- a = frecuencia natural del proceso

El controlador inteligente, como se ha mencionado previamente está constituido por un controlador PID, cuya función de transferencia está dada por:

$$G_c(s) = k_p \left(1 + \frac{1}{t_i * s} \right) \left(\frac{1 + t_d * s}{1 + f * t_d * s} \right) \quad (4)$$

donde:

- k_p = ganancia proporcional
- t_i = tasa integrativa
- t_d = tasa derivativa
- f = constante de suavizado

Estos parámetros son ajustados a los valores que proporciona la base de conocimientos luego de la consulta, como se ve en la figura 4:

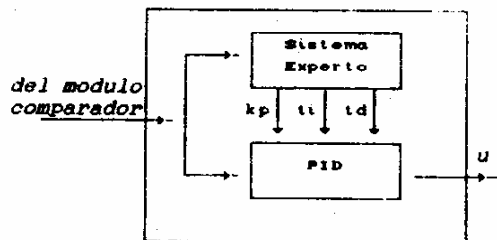


fig. 4

5. Principio general de funcionamiento

El primer paso antes de poner en marcha el sistema es encontrar un valor inicial para k_p , t_i y t_d . Generalmente, en las plantas industriales con experiencia previa se tienen valores aproximados. En el supuesto de ignorar estos datos, se ensayaría la respuesta del proceso a lazo abierto y, observando la respuesta, via Ziegler y Nichols, se obtendrían los valores iniciales apropiados para k_p , t_i y t_d [2].

Fijados los parámetros, se pone en marcha el lazo y se observa la respuesta. Si se hace presente una oscilación creciente indicando inestabilidad, un juego de alarmas y avisos previstos para esta situación informará el estado del proceso. En este

caso se debe utilizar el método mencionado anteriormente y darle valores adecuados de arranque a k_p , t_i y t_d .

De otro modo, se compara la respuesta con la que previamente se definió como deseada. Ello se consigue fijando valores límites para los picos (P_1 , P_2 y P_3), para sus relaciones (S_p y S_r), para el tiempo de establecimiento y, si fuera necesario, también para el error en estado estacionario. Se define en el software de simulación una banda de ruido máximo, encima de la cual comienzan a detectarse los picos.

Si de la anterior comparación resulta que la respuesta es satisfactoria, el PID continúa con su anterior valor de ajuste. Si no, se disparan las reglas con el conocimiento del "experto de campo" obteniéndose un nuevo ajuste. La fig. 5 muestra esa operación en términos de diagrama de flujo.

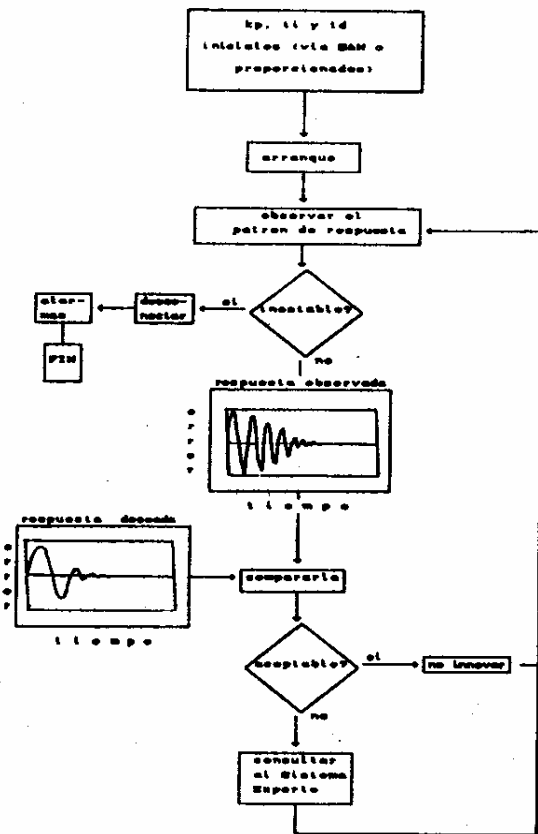


fig. 5

6. Descripción de la herramienta de cómputo

6.1. Hardware

Las pruebas se realizaron sobre un computador personal tipo AT, con coprocesador 287, 12 Mhz de frecuencia de reloj, y 640K de RAM.

6.2. Software

La simulación del lazo se efectuó con un programa de cálculo matricial.

El sistema experto fue desarrollado a partir de una herramienta para la construcción de sistemas expertos (shell) escrita enteramente en lenguaje C, que posee un motor de inferencias con encadenamiento hacia adelante, una memoria de trabajo de tipo agenda y la representación del conocimiento es en base a reglas de producción, [11]. El modelo de razonamiento es muy sencillo, puesto que no se dispone de acumulación de evidencias ni de ninguna forma elaborada de razonamiento aproximado.

7. La Base de Conocimiento

El trabajo puede dividirse en dos fases, la de elaboración del soporte de software necesario y la construcción de la base de conocimiento.

En la primera etapa, se desarrolló un programa interactivo que de acuerdo a las características de optimización de error y a algún conocimiento previo de la planta, proporciona la respuesta temporal a la cual el controlador intentará llevar al lazo. Es posible obtener así un primer juego de valores para k_p , t_i y t_d . Si el eventual usuario conociese algún conjunto de valores de k_p , t_i y t_d , esta parte sería innecesaria. Aunque pueda resultar obvio, cabe aclarar que el tiempo que tarde el controlador en proporcionar la respuesta satisfactoria, función directa de la cantidad de consultas a la base efectuadas, depende fuertemente del punto de partida (k_p , t_i y t_d iniciales) y el de llegada, luego de los sucesivos ajustes (k_p , t_i y t_d optimizados), aún suponiendo en el caso más optimista que el desplazamiento se efectuase sobre la superficie representada por la función distancia mínima.

La simulación del lazo puede ser descrita estructuralmente en los siguientes bloques:

Un primer bloque almacena las variables necesarias, entre ellas, k_p , t_i y t_d . La comunicación entre el programa de cálculos y simulación, y el sistema experto se efectúa por medio de

archivos con un formato previamente especificado. Lo cual disminuye la velocidad de cómputo de todo el sistema, aunque lo presentado en este artículo no está previsto para realizar un control en tiempo real.

Un segundo bloque arma en forma matricial las distintas partes del lazo. El retardo de la planta es simulado usando una aproximación de Padé de cuarto orden. El resto de las transferencias, según las ecuaciones (3) y (4). Una vez obtenidas las matrices que representan el modelo de estados, son ensayadas a una determinada entrada, por ejemplo, un escalón en la referencia.

Un tercer bloque se encarga de analizar el producto del bloque anterior, calculando los picos, sobrepicos, período natural, amortiguamiento, error de estado estacionario y mostrando en pantalla la respuesta en función del tiempo.

Estos datos son almacenados en archivos y quedan disponibles para ser analizados, si es necesario, por el sistema experto.

La elaboración de la base de conocimiento propiamente dicha se efectuó a partir de conocimiento público, disponible en bibliografía y artículos [12], [13], [14], [15], [16] y a partir de conocimiento privado. Este último consistente en la adquisición de experiencia por parte de los autores por medio de sucesivas simulaciones del PID en el lazo cerrado con distintas plantas de la familia representada por la ecuación (3). Ello se traduce en un crecimiento incremental de la misma.

La táctica de análisis fue congelar dos de los tres parámetros de ajuste del PID a los valores propuestos en [13] y variar el tercero. Se obtuvieron curvas cualitativamente similares, como la que se ve en la fig. 6, para distintos elementos de análisis de la respuesta (sobrepico, amortiguamiento, período, error estacionario). Ellas proporcionaron información acerca de los sentidos de variación de los parámetros, y qué características considerar simultáneamente (antecedentes de las reglas). Para el cálculo del valor exacto, como la respuesta que proporciona la base es un porcentaje en más o en menos del valor anterior del parámetro, se escogió una planta prototipo perfectamente conocida por los autores para, a partir de ella, comenzar a modificarla y estudiar hasta qué punto el controlador seguía actuando dentro de los límites admitidos como satisfactorios.

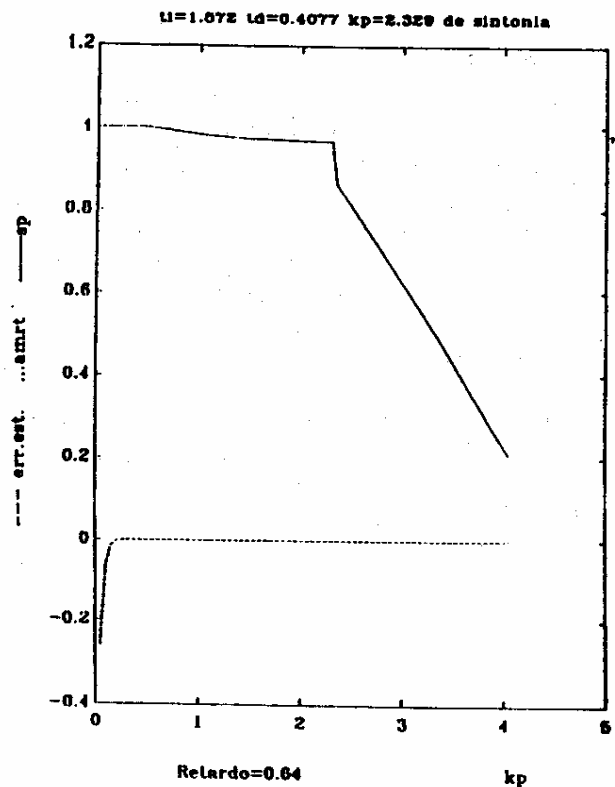


Fig. 6

Un mapa de la base de conocimiento muestra tres zonas perfectamente diferenciadas: un cuerpo de reglas de nivel supervisor, que sólo vigila la estabilidad del sistema y que, al dispararse, activa una serie de alarmas y avisos previstos para tal situación. El otro cuerpo, utiliza solamente los valores de los picos (P1, P2 y P3) para, en primera aproximación, llevar al sistema a un punto de funcionamiento aceptable, aunque no óptimo. Un tercer cuerpo de reglas, basándose en las observaciones del sobrepico, de ξ y de T_n , realiza un ajuste final más sutil.

Se listan a continuación algunas reglas de la base, en la tabla I, una de las correspondientes al primer ajuste, y en la tabla II, al ajuste fino. La sintaxis es propia del lenguaje utilizado.

B. Resultados

```
(defrule R8 "P1 dentro P2 fuera "
(declare (salience 100));se declara una prioridad
(kp ?kp)           ;si kp existe y tiene valor; y
(ti ?ti)           ;si idem ti; y
(P1 ?P1)           ;si idem el primer pico; y
(P2 ?P2)           ;si idem el segundo pico; y
(MAXP1 ?MAXP1)     ;si idem el maximo permitido al
(MINP2 ?MINP2)     ;si idem para el segundo pico; y
(test (< ?P1 ?MAXP1)) ;el primer pico es menor que el
(test (< ?P2 ?MINP2)) ;el segundo pico es menor que el
                    ;minimo permitido
=>
                    ;entonces

(fprintout t "R8" crlf) ;imprima en pantalla "R8"; y
(bind ?kp (* ?kp 1.1)) ;haga kp un 10 % mayor
(bind ?kp (* ?kp mu))  ;multiplique kp por la f.de pert.
(bind ?ti (* ?ti 0.8)) ;haga ti un 20 % menor
(bind ?ti (* ?ti mu))  ;multiplique ti por la f.de pert.
(fprintout t "P1 DENTRO. P2 FUERA " crlf) ;carteles de
(fprintout t "kp= " ?kp " ti= " ?ti crlf) ;aviso
(assert (kp ?kp))      ;cargue en la base los nuevos
(assert (ti ?ti))      ;valores de kp y ti
(dejar kp.mat kp ?kp)  ;almacene en sendos archivos los
(dejar ti.mat ti ?ti) ;nuevos valores de kp y ti
```

tabla I

```
(defrule R12 "sp con td bajo"
(declare (salience 10));se declara una prioridad
(Sobrep ?sp)       ;si la variable Sobrep existe y
(Amort ?amrt)      ;tiene valor; y
                    ;idem Amort(iguamiento); y
(AMESP ?ameesp)    ;idem con el amort. especificado; y
(SOBESP ?spesp)    ;idem con sobrep. especificado; y
(td ?td)           ;idem td; y
(test (> ?amrt ?ameesp));el valor de Amort es > el esp.; y
(test (<= ?sp ?spesp)) ;el valor de Sobrep es <= el espe-
                    ;cificado
=>
                    ;entonces
(fprintout t "R12" crlf) ;carteles de aviso
(fprintout t "SE AUMENTA EL SOBREPICO" crlf)
(bind ?td (* ?td 0.7));haga td un 30 % menor; y
(bind ?td (* ?td mu)) ;multiplique td por f.de pertenen.
(fprintout t "td= " ?td crlf);carteles de aviso
(dejar td.mat td ?td) ;almacenar en archivo el nuevo
                    ;valor de td

(defrule R17 "periodo alto por kp "
(declare (salience 10));se declara una prioridad
(kp ?kp)           ;si kp existe y tiene valor; y
(Tn ?per)          ;idem el periodo natural; y
(Sobrep ?sp)       ;idem el sobrepico; y
(PESP ?PESP)       ;idem el periodo especificado; y
(SOBESP ?spesp)    ;idem el sobrep. especificado; y
(test (> ?per ?PESP)) ;el periodo es > el especific.; y
(test (> ?sp ?spesp)) ;el valor del Sobrep. es > el es-
                    ;pecificado
=>
                    ;entonces
(fprintout t "R17" crlf);carteles de aviso
(bind ?kp (* ?kp 1.2));haga kp un 20 % mayor; y
(bind ?kp (* ?kp mu)) ;multiplique kp por f.de pertenen.
(assert (kp ?kp))      ;cargue en la base el nuevo kp
(fprintout t "SE BAJA EL PERIODO " crlf);carteles de
(fprintout t "kp= " ?kp crlf) ;aviso
(dejar kp.mat kp ?kp) ;almacenar en un archivo el nuevo
                    ;valor de kp
```

tabla II

El proceso utilizado para elaborar la base de conocimiento responde a la ecuación (3), con los siguientes valores:

$$k = 1$$

$$\tau = 0.64 \text{ seg.}$$

$$a = 2$$

A esta planta se le exigió que se comportara a lazo cerrado con el controlador inteligente de manera tal que la respuesta optimizase el ISK (integral temporal de la raíz cuadrada del error), cuando solamente se variase la referencia en forma de escalón unitario, de modo de obtener valores menores de ganancia frente a los que se hubiesen obtenido al variar la carga [12]. La respuesta óptima, de acuerdo a los trabajos de Kaya-Scheib [13], es la que se muestra en la fig.8, con los respectivos valores para los parámetros del PID.

Respuesta al escalon $kp=2.3294$, $ti=1.8716$, $td=0.4077$

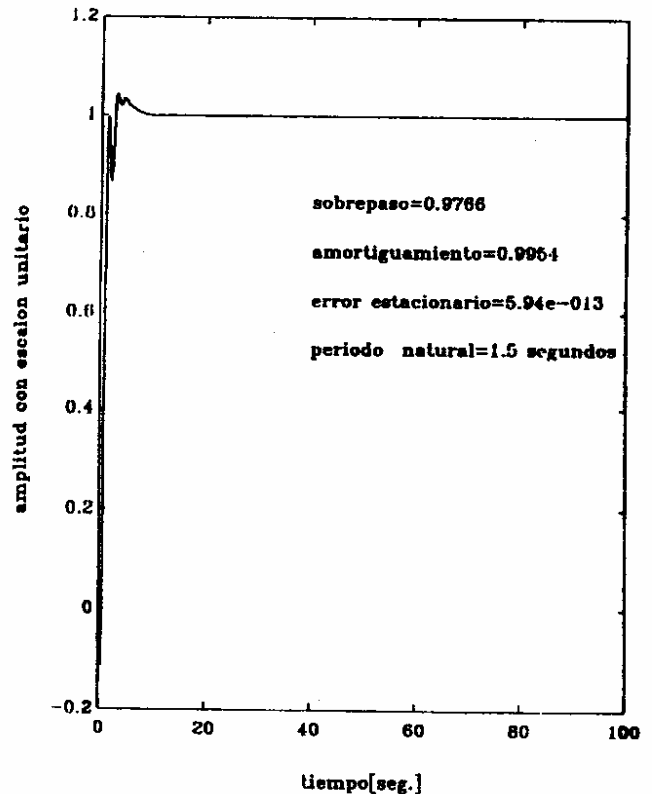


fig. 7

Se escogieron al azar valores iniciales para los parámetros del PID, a saber:

$k_p = 4$
 $t_i = 20$
 $t_d = 0$

Notar que éstos son muy diferentes de los valores que deben adoptarse para optimizar la respuesta ($k_p=2.3294$, $t_i=1.8716$, $t_d=0.4077$). No obstante ello, luego de seis variaciones de la referencia, el controlador consigue dar una respuesta a todo el lazo que es muy aproximada a la deseada. Dicho resultado puede observarse en las figuras 9 y 10. Los valores que fueron adoptando los parámetros del PID se adjuntan en la tabla III.

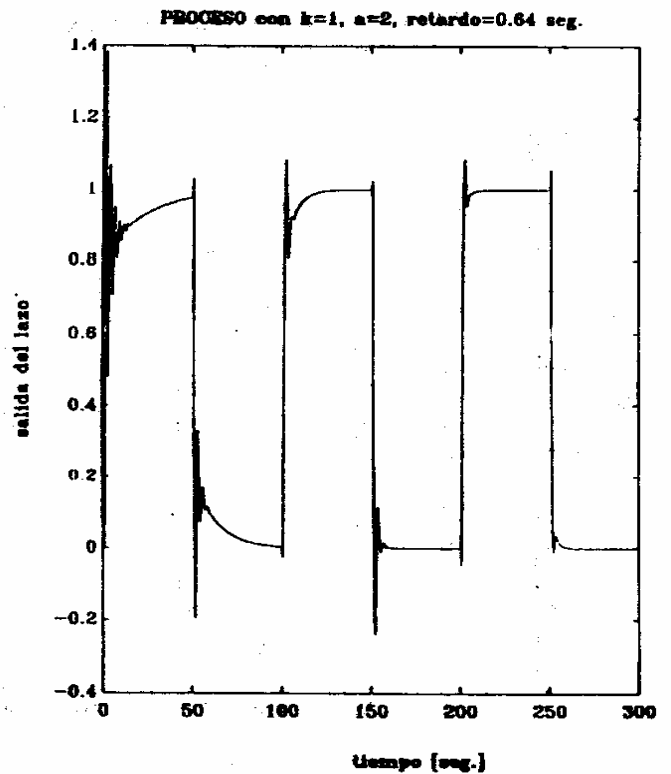


fig. 9

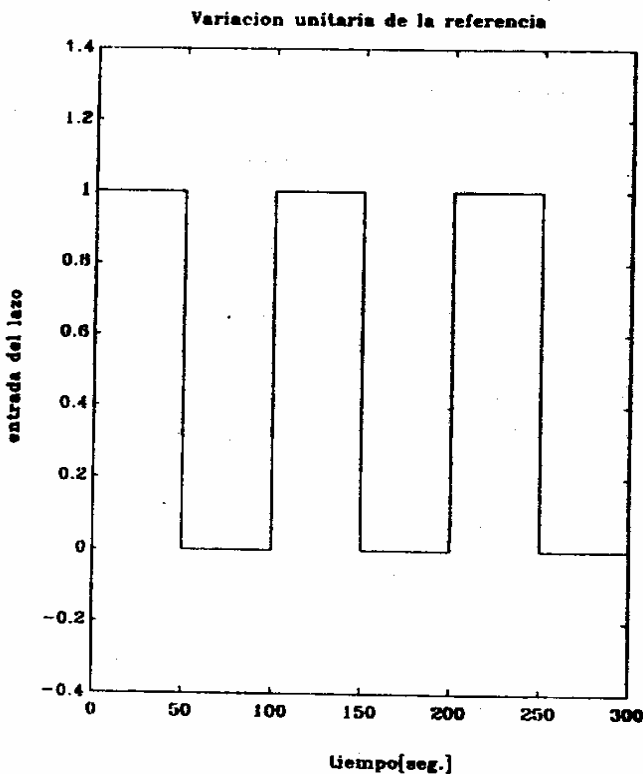


fig. 8

intervalo temporal (seg.)	valores de los parámetros
0-50	$k_p=4$ $t_i=20$ $t_d=0$
50-100	$k_p=3.2$ $t_i=10$ $t_d=0$
100-150	$k_p=2.36$ $t_i=5$ $t_d=0$
150-200	$k_p=2.36$ $t_i=2.5$ $t_d=0$
200-250	$k_p=2.43$ $t_i=2.5$ $t_d=0.11$
250-300	$k_p=2.31$ $t_i=2.5$ $t_d=0.105$

tabla III

También se realizaron pruebas variando el retardo y la ubicación del polo. Se verifica así la robustez del mismo frente a variaciones del proceso. Ello puede apreciarse en las figuras 10 y 11.

En la fig. 10, en el proceso variaron la posición del polo y el retardo fue aumentado al doble, sin embargo, la base de conocimiento pudo controlar esta nueva situación. Los valores de los parámetros del PID se ven en la tabla IV.

PROCESO con $k=1$, $a=5$, retardo=1.28 seg.

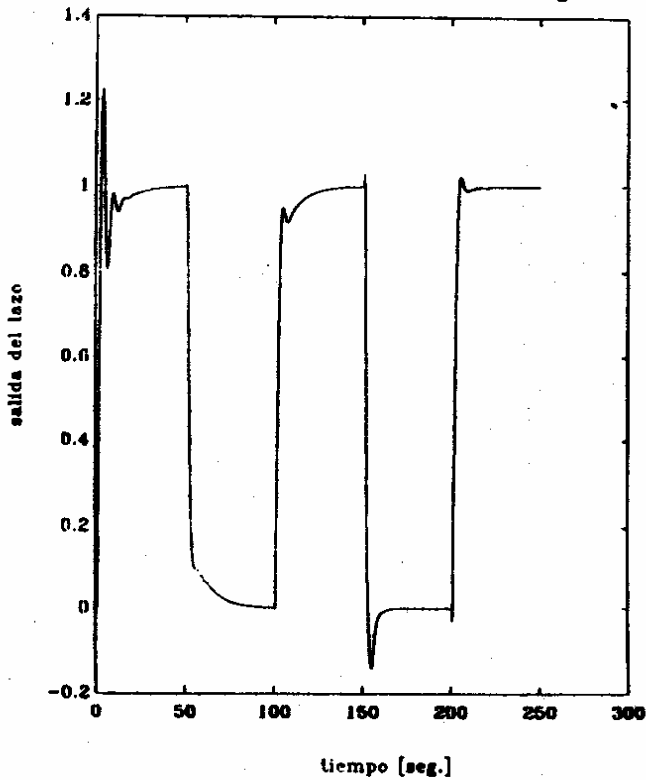


fig. 10

intervalo temporal (seg.)	valores de los parámetros
0-50	$k_p=2.5$ $t_i=10$ $t_d=0$
50-100	$k_p=1.50$ $t_i=8$ $t_d=0.11$
100-150	$k_p=2.304$ $t_i=8$ $t_d=0.105$
150-200	$k_p=2.304$ $t_i=4$ $t_d=0.105$
200-250	$k_p=2.37$ $t_i=5.6$ $t_d=0.105$

tabla IV

En la fig. 11, se varió directamente el modelo de la planta, puesto que no responde a la ecuación (3) original sino que tiene un polo adicional. Su función de transferencia en el dominio de la frecuencia compleja de Laplace es:

$$G_p(s) = \frac{1/2 * e^{-sT}}{(s + 0.5)(s + 5)} \quad (5)$$

Se observa que el comportamiento del controlador es altamente satisfactorio, aún en el caso de controlar un proceso con una función de transferencia distinta a la prevista. La tabla V muestra el recorrido de los parámetros del PID.

PROCESO con $k=0.5$, retardo=0.64 seg., polos en 0.5 y 5

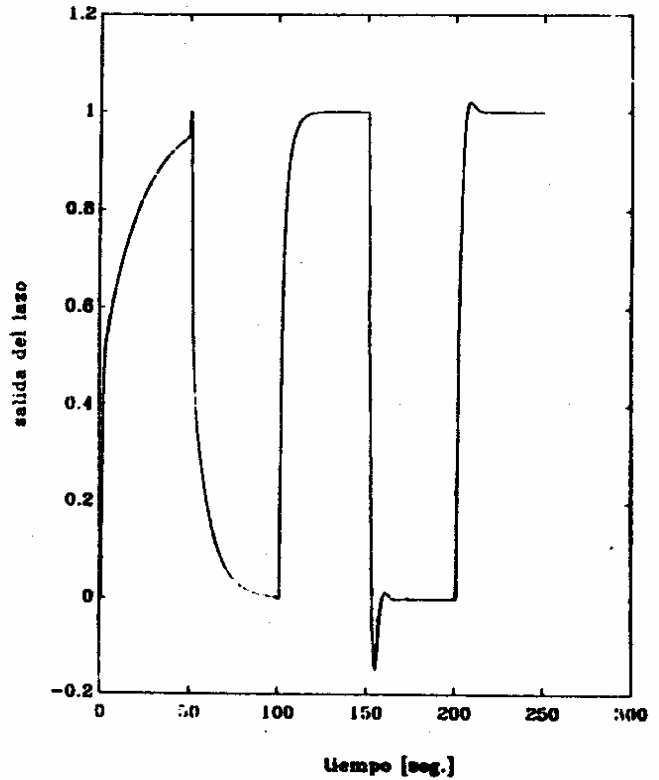


fig. 11

intervalo temporal (seg.)	valores de los parámetros
0-50	$k_p=4$ $t_i=10$ $t_d=0$
50-100	$k_p=4$ $t_i=5$ $t_d=0$
100-150	$k_p=4$ $t_i=2.5$ $t_d=0$
150-200	$k_p=4$ $t_i=1.25$ $t_d=0$
200-250	$k_p=3.76$ $t_i=1.75$ $t_d=0.15$

tabla V

4. Conclusiones

Los resultados experimentales han demostrado que el sistema experto se desempeña exitosamente controlando un gran número de plantas, aunque el mayor énfasis de este trabajo se puso en realidad en desarrollar un ambiente de software propicio para la generación y validación de bases de conocimiento para aplicarlas al control de procesos. Este es un paso previo a realizar un control experto en tiempo real, para lo cual es conveniente organizar a la base de un modo distinto, separando los cuerpos de reglas en tres bases diferentes para disminuir el espacio de búsqueda y con ello, el tiempo de consulta, con miras a poder efectuarla en un período de muestreo.

Otro aspecto interesante a considerar es la posibilidad de construir un controlador con reglas difusas, autoprogramable de acuerdo a un somero reconocimiento preliminar que se realice de la planta. En efecto, el nuevo valor del parámetro de ajuste del PID que proporciona la base es un producto tal como:

$$P_{aj} = P_{an} * x.yz * \mu \quad (6)$$

Aquí, p_{aj} es el parámetro de ajuste, p_{an} es el valor anterior del parámetro, $x.yz$ es el porcentaje de incremento o decremento (según x sea mayor o menor que cero, respectivamente, como puede apreciarse en las tablas I y II con algunas reglas), con respecto al valor anterior, y μ es un coeficiente de ajuste, que refleja qué tan distinta es la planta presente de aquella para la cual fue diseñada la base.

REFERENCIAS

- [1] Ziegler, J.G., and Nichols, B.N.: "Optimum settings for automatic controllers", Trans. ASME, 1942, 64, pp.759-768.
- [2] Aström, K.J., Anton, J.J., and Arzén, K.E.: "Expert Control", Automatica, 1986, vol.22, No. 3, pp. 227-286.
- [3] Alves da Silva, M., Gomide, F.A., Amaral, W.C.: "A rule based procedure for selftuning PID controllers", comunicación privada.
- [4] Rich, E.: "Artificial Intelligence", McGraw-Hill, 1983.
- [5] Winston, P.H.: "Artificial Intelligence", Addison-Wesley, 1984.
- [6] Barr, A., and Feigenbaum, E.A.: "The handbook of Artificial Intelligence", vol.I, y Cohen, P.R., and Feigenbaum, E.A.:idem vol. III, William Kaufmann, 1981.
- [7] Hayes-Roth, F., Waterman, D.A., and Lenat, D.B.: "Building Expert Systems", Addison-Wesley, 1983.
- [8] Kraus, T.W., and Myron, T.J.: "Self-tuning PID Controller uses pattern recognition approach", Control Engineering, vol 36, No. 6, June 1984, pp. 106-111.
- [9] Bristol, E.H.: "Pattern Recognition: an alternative to parameter identification in adaptive control", Automatica, No.13, 1977.
- [10] Shinskey, F.G.: "Process-Control Systems", McGraw-Hill, 1979, pp.91-122.
- [11] Acosta, G.G. y Catalfo, J.M.: "Algunos aspectos de los Sistemas Expertos y el lenguaje C", Informe Interno LEICI 40-89-6, 1989.
- [12] Rovira, A.A., Murrill, W.P., Smith, C.L.: "Tunning Controllers for Set-point Changes", Instruments & Control Systems, pp. 67-69, December 1969.
- [13] Kaya, A., and Scheib, T.J.: "Tuning of PID Controls of different structures", Control Engineering, July 1988, pp.62-65.
- [14] Porter, B., Jones, A.H., and McKeown, C.B.: "Real-time expert tunners for PI controllers", IEE Proceedings, vol 134, No. 4, July 1987.
- [15] Vega, P., Serrano J., Prada, C.: "A comparative study of automatic tuning methods for industrial PID controllers", Proceedings of the IAESTED International Symposium, February 1989, pp. 198-201.
- [16] Hoopes, H.S., Hawk, W.M.Jr., Lewis, R.C.: "A self-tuning controller", ISA Trans., Vol. 22, No.3, 1983, pp.49-58.



Gerardo Gabriel Acosta es graduado en Ingeniería electrónica por la Universidad Nacional de La Plata, Argentina, en 1986. Es miembro de la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CICPBA) y docente de la Facultad de

Ingeniería de la Universidad Nacional de La Plata. Sus áreas de interés abarcan la utilización de técnicas de Inteligencia Artificial para el control de procesos.



Miguel Angel Mayosky es ingeniero en electrónica por la Universidad Nacional de La Plata, Argentina, en 1983, y doctor en informática por la Universidad Autónoma de Barcelona, en 1989. Es miembro del Consejo Nacional de Investigaciones Científicas y

Tecnológicas (CONICET) y profesor de la Facultad de Ingeniería de la Universidad Nacional de La Plata. Sus áreas de interés abarcan los sistemas de multiprocesamiento, arquitecturas paralelas y redes neuronales aplicadas al control



Jose M. Catalfo es graduado en Ingeniería electrónica por la Universidad Nacional de La Plata, Argentina, en 1989. Ha trabajado en el campo de sistemas de control industrial como investigador. Durante 1981-1982 formó parte de la Organización Argentina para la Investigación Nuclear (OAIEN) en Argentina.

Actualmente es investigador asociado al Consejo Nacional de Investigaciones Científicas y Tecnológicas (CONICET) y docente en la Facultad de Ingeniería de la Universidad Nacional de La Plata.