

Optimización de Parámetros Utilizando los Métodos de Monte Carlo y Algoritmos Evolutivos. Aplicación a un Controlador de Seguimiento de Trayectoria en Sistemas no Lineales

C. Fernández, N. Pantano, S. Godoy, E. Serrano, G. Scaglia

Instituto de Ingeniería Química, Universidad Nacional de San Juan (UNSJ), CONICET, Av. Lib. San Martín Oeste 1109, San Juan J5400ARL, Argentina.

Resumen

En este trabajo se propone una estrategia de control en lazo cerrado para el seguimiento de perfiles óptimos previamente definidos para un bioproceso fed-batch. La mayor ventaja de este enfoque es que las acciones de control se calculan resolviendo un sistema de ecuaciones lineales, sin tener que linealizar el modelo matemático, lo que permite trabajar en cualquier rango. Además, se plantean tres técnicas para la sintonización de los parámetros del controlador diseñado. Primero se propone un método de Monte Carlo, el cual es un método probabilístico. En segundo lugar, se presenta una metodología basada en Algoritmos Genéticos, una técnica evolutiva de optimización. La tercera alternativa es el desarrollo de un Algoritmo Híbrido, diseñado a partir de la combinación de los dos métodos anteriores. En todos los casos, el objetivo es encontrar los parámetros del controlador que minimicen el error total de seguimiento de trayectorias. El desempeño del controlador se evalúa a través de simulaciones en condiciones normales de operación y frente a incertidumbre paramétrica, empleando los parámetros del controlador obtenidos.

Palabras Clave:

Control en lazo cerrado, sistemas no lineales, control de sistemas multivariables, método de Monte Carlo, Algoritmos Genéticos..

Parameters optimization applying Monte Carlo methods and Evolutionary Algorithms. Enforcement to a trajectory tracking controller in non-linear systems.

Abstract

In this work, a closed-loop control strategy is proposed. It allows tracking optimal profiles for a fed-batch bioprocess. The main advantage of this approach is that the control actions are computed from a linear equations system without linearizing the mathematical model, which allows working in any range. In addition, three techniques are developed to tune the controller. First, a completely probabilistic method, Monte Carlo. Second, a methodology based on Genetic Algorithms, an evolutionary optimization technique. Third, a Hybrid Algorithm, combining above algorithms advantages. Here, the objective function is to find the controller parameters that minimize the trajectory tracking total error. The controller performance is evaluated through simulations under normal operations conditions and parametric uncertainty, using the obtained controller parameters.

Keywords:

Closed loop control, nonlinear systems, multivariable control systems, Monte Carlo method, Genetic Algorithms.

1. Introducción

Dado el mayor interés por el concepto de desarrollo sustentable, los bioprocesos se han convertido en una prometedora alternativa en los últimos años. Estos procesos, permiten obtener una gran variedad de productos, como proteínas recombinantes, antibióticos, fármacos, alimentos, etc., e incluso, reemplazar procesos tradicionales de la industria química, logrando reducir gastos energéticos y problemas ambientales, generando compuestos biodegradables a partir de materias primas renovables (Rocha et al., 2014). Sin embargo, al trabajar con

organismos vivos aumenta la complejidad del modelado, optimización y control (Vilanova et al., 2017).

Desde el punto de vista de la ingeniería de control de procesos, el modo de operación fed-batch presenta grandes desafíos: sistema altamente no lineal, importante variación de los parámetros del modelo en función del tiempo, variabilidad en las condiciones iniciales, saturación de las acciones de control e influencia de perturbaciones externas, entre otros (Rani and Rao, 1999, Bayen and Mairet, 2013, Renard et al., 2006, Lee et al., 1999, Johnson, 1987, Daoutidis et al., 2016).

*Autor para la correspondencia: mcfernandez@unsj.edu.ar

En la actualidad se estudian varias técnicas para abordar la optimización y control de bioprocesos, tales como métodos estocásticos (Rocha et al., 2014, Sarkar and Modak, 2003, Sarkar and Modak, 2004, Ochoa et al., 2010), control robusto (Renard et al., 2006, Renard and Wouwer, 2008), control difuso (Cosenza and Galluzzo, 2012), control predictivo (Ashoori et al., 2009, Craven et al., 2014, Santos et al., 2012, Mohd and Aziz, 2015) y redes neuronales (Saint-Donat et al., 1991, Tholudur and Ramirez, 1996) entre otros. Estos métodos tienen buenos resultados en procesos biológicos, no obstante, tienen limitaciones respecto a la necesidad de conocimiento muy específico para su desarrollo, dificultad en el procesamiento matemático y la escasez de datos del proceso. En las estrategias de control óptimo a lazo abierto, la principal desventaja es que los errores de modelado y las perturbaciones externas del proceso no son tenidos en cuenta (Soni and Parker, 2004, Lee et al., 1999, Chang, 2003, Chung et al., 2006). Por lo tanto, es importante el diseño de un controlador que permita el seguimiento de perfiles óptimos de las variables más importantes del proceso, considerando las incertidumbres del sistema a lazo cerrado.

En este trabajo se presenta una técnica de control en lazo cerrado para el seguimiento de perfiles óptimos de un bioproceso fed-batch. La mayor ventaja es que las acciones de control se obtienen resolviendo un sistema de ecuaciones lineales, sin necesidad de linealizar el modelo matemático original. En comparación con las técnicas de control clásicas no se requieren conocimientos específicos de teoría de control, ya que todo el procedimiento se basa en la búsqueda de las condiciones que un sistema de ecuaciones debe cumplir para tener solución exacta. Esta estrategia es simple, versátil y efectiva, aún frente a cambios en las condiciones iniciales del proceso e incertidumbre paramétrica.

Considerando que la búsqueda de los parámetros óptimos del controlador es una tarea complicada, ya que en gran medida depende del proceso a ser controlado, y que no existe un método generalizado para todas las aplicaciones, en este trabajo se proponen tres algoritmos destinados a encontrar los parámetros más apropiados para el controlador mencionado, pudiendo ser extendidas a otros controladores de la bibliografía. La primera metodología, corresponde al algoritmo de Monte Carlo, que consiste en simular el proceso a lazo cerrado un cierto número de veces con valores aleatorios de los parámetros del controlador, para cada simulación se calcula el error total generado y se eligen aquellos parámetros que minimizan dicho error. La segunda estrategia de sintonización está basada en Algoritmos Genéticos, la cual es ampliamente utilizada para resolver problemas de optimización complejos (Holland, 1975, Sadatsakkak et al., 2015, Asadi et al., 2014, Yu et al., 2015, Ismail et al., 2014), donde el número de parámetros es grande y la solución analítica es difícil de obtener; este método imita la teoría de la evolución biológica propuesta por Darwin para la resolución de problemas. Para ello, se parte de una población inicial (que se elige de forma aleatoria) de la cual se seleccionan los individuos más capacitados para luego reproducirlos, mutarlos y finalmente obtener la siguiente generación de individuos que estarán más adaptados que la anterior. El tercer método propuesto, es un algoritmo Híbrido con el cual se busca combinar las ventajas de los dos métodos anteriores con el fin de mejorar la búsqueda y aumentar la confiabilidad de los resultados. Luego, cada algoritmo se compara teniendo en cuenta el error total de seguimiento y el costo computacional.

El caso de estudio propuesto para el control es la producción fed-batch de bioetanol, presentado en (Hunag et al., 2012). Cabe destacar que esta técnica puede ser aplicada en otros sistemas.

El artículo se organiza de la siguiente manera: la Sección 2 muestra una breve descripción del proceso. En la Sección 3 se presenta el esquema de diseño del controlador. La Sección 4 exhibe el desarrollo de las metodologías para la sintonización del controlador. Los resultados y discusión se encuentran en la Sección 5, donde se incluyen simulaciones para evaluar el sistema propuesto. Finalmente, la Sección 6 presenta las conclusiones.

2. Descripción del Proceso

El sistema en estudio es un biorreactor fed-batch para la producción de bioetanol. El microorganismo empleado es *Saccharomyces diastaticus*. El reactor es alimentado con dos sustratos, glucosa y fructosa, en proporciones iguales. El modelo matemático del proceso, ver (1) y (2), fue propuesto en (Hunag et al., 2012). Es un sistema de simple entrada y múltiples salidas, donde la entrada es la velocidad de alimentación de sustrato (U), y las salidas son las concentraciones de células (X), etanol (P_1), glicerol (P_2), glucosa (S_1) y fructosa (S_2) dentro del reactor.

$$\begin{cases} \dot{X}(t) = (\mu_1 + \mu_2)X - \frac{U}{V}X \\ \dot{S}_1(t) = -\left(\frac{q_{S_1/P_1}}{Y_{P_1/S_1}} + \frac{q_{S_1/P_2}}{Y_{P_2/S_1}}\right)X + \frac{U}{V}(\lambda S_f - S_1) \\ \dot{S}_2(t) = -\left(\frac{q_{S_2/P_1}}{Y_{P_1/S_2}} + \frac{q_{S_2/P_2}}{Y_{P_2/S_2}}\right)X + \frac{U}{V}((1-\lambda)S_f - S_2) \\ \dot{P}_1(t) = (q_{S_1/P_1} + q_{S_2/P_1})X - \frac{U}{V}P_1 \\ \dot{P}_2(t) = (q_{S_1/P_2} + q_{S_2/P_2})X - \frac{U}{V}P_2 \end{cases} \quad (1)$$

donde,

$$\begin{aligned} \dot{V}(t) &= U \\ \mu_1 &= \frac{\mu_{m_1} S_1}{K_{S_1} + S_1 + S_1^2 / K_{S_{11}}} \frac{K_{P_1}}{K_{P_1} + P_1 + P_1^2 / K_{P_{11}}} \\ \mu_2 &= \frac{\mu_{m_2} S_2}{K_{S_2} + S_2 + S_2^2 / K_{S_{21}}} \frac{K_{P_2}}{K_{P_2} + P_1 + P_1^2 / K_{P_{21}}} \\ q_{S_1/P_1} &= \frac{v_{S_1 P_1} S_1}{K_{S_1 P_1} + S_1} \frac{k_{S_1 P_1}}{k_{S_1 P_1} + P_1} \\ q_{S_1/P_2} &= \frac{v_{S_1 P_2} S_1}{K_{S_1 P_2} + S_1} \frac{k_{S_1 P_2}}{k_{S_1 P_2} + P_2} \\ q_{S_2/P_1} &= \frac{v_{S_2 P_1} S_2}{K_{S_2 P_1} + S_2} \frac{k_{S_2 P_1}}{k_{S_2 P_1} + P_1} \\ q_{S_2/P_2} &= \frac{v_{S_2 P_2} S_2}{K_{S_2 P_2} + S_2} \frac{k_{S_2 P_2}}{k_{S_2 P_2} + P_2} \end{aligned} \quad (2)$$

En (2), μ_1 y μ_2 representan la velocidad específica de crecimiento celular, q_{S_1/P_1} y q_{S_2/P_1} la velocidad específica de producción de etanol, y q_{S_1/P_2} y q_{S_2/P_2} la velocidad específica de producción de glicerol, en todos los casos en relación a glucosa y fructosa, respectivamente. La variable V simboliza el volumen en el reactor.

Las condiciones iniciales del proceso son: $X_0=1.5g/L$, $P_{10}=5.3g/L$, $P_{20}=0.0001g/L$, $S_{10}=8.6g/L$, $S_{20}=8.6g/L$, $V_0=1.35L$. La nomenclatura, descripción y valor de cada parámetro se encuentran en (Hunag et al., 2012).

3. Diseño del controlador

A continuación, se propone una metodología que permite calcular las acciones de control que deben ser aplicadas, en cada instante de muestreo, de forma tal que X , P_1 y P_2 sigan un perfil preestablecido. La técnica de control propuesta, permite el seguimiento de perfiles óptimos previamente definidos. Para su desarrollo, se asume que se dispone del modelo matemático que representa el proceso, que los perfiles de referencia que se desea seguir son conocidos y que cada variable de estado puede ser medida en línea en cada instante de muestreo (Bogaerts and Coutinho, 2014, Bogaerts and Wouwer, 2003, Dewasme et al., 2015, Hulhoven et al., 2006, Pantano et al., 2017, Vilums and Grigs, 2012, Fernández et al., 2018).

El primer paso es integrar las ecuaciones diferenciales de (1) usando algún método numérico, en este caso se propone utilizar el método de Euler.

$$\left(\frac{d\sigma}{dt}\right) = \frac{\sigma_{n+1} - \sigma_n}{T_s} \quad (3)$$

En (3), σ simboliza cada variable de estado, σ_n es el valor presente de σ medido en el reactor (en línea), y σ_{n+1} es el valor de σ en el próximo instante de muestreo. La constante T_s es el tiempo de muestreo ($T_s=0.1h$). El proceso dura 15.7h.

Las variables en el instante $n+1$ son aproximadas con la siguiente expresión:

$$\underbrace{\sigma_{ref\ n+1} - \sigma_{n+1}}_{error_{n+1}} = k_\sigma \underbrace{(\sigma_{ref\ n} - \sigma_n)}_{error_n} \quad (4)$$

$$\sigma_{n+1} = \sigma_{ref\ n+1} - k_\sigma (\sigma_{ref\ n} - \sigma_n)$$

En (4), el subíndice *ref* indica el valor de referencia de la variable de estado σ , el cual se asume conocido en cada instante de muestreo. k_σ simboliza los parámetros del controlador y toman valores entre 0 y 1, lo cual permite que el error de seguimiento tienda a cero con el avance del proceso.

A continuación se reemplaza (4) en (3) y el resultado se aplica en (1). Luego, se expresa el sistema de ecuaciones lineales en forma matricial, colocando las variables de estado en función de la acción de control (U):

$$\underbrace{\begin{bmatrix} -X_n/V_n \\ (\lambda S_f - S_{1n})/V_n \\ ((1-\lambda)S_f - S_{2n})/V_n \\ -P_{1n}/V_n \\ -P_{2n}/V_n \end{bmatrix}}_A \underbrace{U}_u = \underbrace{\begin{bmatrix} \frac{(X_{ref\ n+1} - k_X(X_{ref\ n} - X_n) - X_n)}{T_s} - (\mu_1 + \mu_2)X_n \\ \frac{(S_{1ref\ n+1} - k_{S_1}(S_{1ref\ n} - S_{1n}) - S_{1n})}{T_s} + \left(\frac{q_{S_1/P_1}}{Y_{P_1/S_1}} + \frac{q_{S_1/P_2}}{Y_{P_2/S_1}}\right)X_n \\ \frac{(S_{2ref\ n+1} - k_{S_2}(S_{2ref\ n} - S_{2n}) - S_{2n})}{T_s} + \left(\frac{q_{S_2/P_1}}{Y_{P_1/S_2}} + \frac{q_{S_2/P_2}}{Y_{P_2/S_2}}\right)X_n \\ \frac{(P_{1ref\ n+1} - k_{P_1}(P_{1ref\ n} - P_{1n}) - P_{1n})}{T_s} - (q_{S_1/P_1} + q_{S_2/P_1})X_n \\ \frac{(P_{2ref\ n+1} - k_{P_2}(P_{2ref\ n} - P_{2n}) - P_{2n})}{T_s} - (q_{S_1/P_2} + q_{S_2/P_2})X_n \end{bmatrix}}_b \quad (5)$$

En forma genérica, el sistema queda expresado de la siguiente manera:

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} U = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} \quad (6)$$

Para conocer U , es necesario pedirle al sistema que tenga solución exacta. En otras palabras, b debe ser combinación lineal de A (Strang, 2006), es decir, A y b deben ser paralelos. Existen diversas maneras de satisfacer esa condición, una de ellas es que el ángulo entre A y b sea cero:

$$\cos(A, b) = \frac{\langle A, b \rangle}{\|A\| * \|b\|} = 1 \quad (7)$$

En (7), la operación entre $\langle \rangle$ y $\| \cdot \|$ representa el producto interno y la norma de vectores en R^n , respectivamente.

Para garantizar que (6) tiene solución exacta, se define la “variable sacrificada”, denotada con “ ez ”. Esta variable toma valores diferentes a la referencia con el fin de asegurar que las otras variables sigan los perfiles deseados. Para seleccionarla se debe hacer un estudio de cada variable del sistema y la función que cumple cada una en el proceso. Para este sistema en particular se elige uno de los dos sustratos como sacrificada, ya que la evolución de las demás variables dependen enteramente de ellos. Se utiliza (7) para obtener el valor de la variable mencionada, una vez conocido, es reemplazado en (6).

Finalmente, U es calculado aplicando mínimos cuadrados a (6) (Strang, 2006).

Por otro lado, se define el “error de seguimiento ($\|e_n\|$)”:

$$\|e_n\| = \sqrt{\frac{((X_{refn} - X_n) / \max X_{refn})^2 + ((P_{1refn} - P_{1n}) / \max P_{1refn})^2}{+((P_{2refn} - P_{2n}) / \max P_{2refn})^2 + ((S_{2refn} - S_{2n}) / \max S_{2refn})^2}} \quad (8)$$

La Ec. (8) se utiliza para verificar el desempeño del controlador en cada instante de tiempo, tanto para sintonizarlo como para testear su funcionamiento ante diversas incertidumbres. Para la comparación entre simulaciones, se utiliza el “error total de seguimiento”, definido como la suma de los errores de seguimiento en cada instante de muestreo (157 puntos):

$$E_p = T_s \sum_{n=1}^{157} \|e_n\| \quad (9)$$

4. Optimización de los parámetros del controlador

En esta sección se presentan tres alternativas diferentes para la selección de los parámetros óptimos del controlador desarrollado (k_x , k_{S_1} , k_{S_2} , k_{P_1} y k_{P_2}). El propósito es encontrar aquellos parámetros que cumplan con la función objetivo definida en (10), es decir, que optimicen el funcionamiento del controlador. Cabe destacar que cualquiera de las tres estrategias a presentar pueden ser aplicadas a otros controladores, lo cual constituye uno de los mayores aportes de este trabajo. La Fig. 1 muestra un resumen de los algoritmos a tratar.

$$\text{Minimizar: } J = T_s \sum_{i=1}^{157} \|e_n\| \quad (10)$$

4.1. Monte Carlo (MC)

Consiste en un algoritmo aleatorio que hace elecciones al azar para producir un resultado. Esto implica que pueda dar un índice mayor o menor, es decir, está sujeto a probabilidad.

Monte Carlo es un algoritmo aleatorio que puede no encontrar el mínimo, pero la probabilidad de tal resultado es limitada. Una forma de reducir dicha probabilidad es simplemente ejecutar el algoritmo repetidamente (N veces) con opciones aleatorias independientes de los parámetros en cada momento (ver Teorema 2 de (Tempo and Ishii, 2007)). El número de simulaciones (N) se obtiene eligiendo una exactitud y confianza apropiada para limitar la posibilidad de una respuesta no deseada, luego se utiliza (11) para calcular el número de iteraciones.

$$N \geq \left\lceil \frac{\log \frac{1}{\delta}}{\log \frac{1}{1-\varepsilon}} \right\rceil \quad (11)$$

donde, δ es la confianza y ε es la exactitud.

Secuencia del algoritmo de Monte Carlo aplicado:

1. Se definen los parámetros del controlador a optimizar: k_σ .
2. Se determina el número de simulaciones a efectuar (N). Los valores de δ y ε se eligen dependiendo de la precisión deseada, en este caso: $\delta=0.01$ y $\varepsilon=0.005$, lo que resulta en $N=1000$.
3. En forma aleatoria se asigna un valor para cada parámetro del controlador, $0 < k_\sigma < 1$.
4. Se simula el proceso y se calcula J .
5. Se repiten los pasos 3 y 4 hasta completar las N iteraciones.
6. Finalmente, los parámetros del controlador que minimicen J son los seleccionados.

4.2. Algoritmos Genéticos (AG)

Los Algoritmos Genéticos son métodos evolutivos que pueden usarse para resolver problemas de búsqueda y optimización. Están basados en el proceso genético de los organismos vivos. A lo largo de las generaciones, las poblaciones evolucionan en la naturaleza acorde con los principios de la selección natural y la supervivencia de los más fuertes, postulados por Darwin (1859) (Troncoso and Suárez, 2017).

Básicamente, el algoritmo consiste en trabajar con una población de individuos, cada uno de los cuales representa una solución factible al problema dado. A cada individuo se le asigna un valor o puntuación (selección), relacionado con la bondad de dicha solución y, mediante operaciones propias del algoritmo (cruzamiento, mutación), los individuos modificados forman nuevas generaciones que permiten la evolución del sistema en búsqueda del valor óptimo (Mohanty et al., 2014, Rajarathinam et al., 2016, Hassan et al., 2013, Vishal et al., 2014).

Asumiendo que los parámetros del controlador del sistema en estudio son números reales, se trabaja con una codificación por valor directo. La metodología para la selección de los individuos más capacitados (para que éstos sean los que se reproduzcan con más probabilidad) en cada generación se efectúa a través de una selección elitista, en la cual se copia el mejor individuo o alguno de los mejores en la nueva población. El elitismo puede mejorar el funcionamiento de los algoritmos genéticos al evitar que se pierda la mejor solución. Para el cruzamiento o reproducción, se emplea la técnica de un punto, donde los dos cromosomas padres se cortan por un punto, se copia la información genética de uno

de los padres desde el inicio hasta el punto de cruce y el resto se copia del otro progenitor. El objetivo del cruce es conseguir que el descendiente mejore la aptitud de sus padres. Para la operación de mutación, se elige una posición en el cromosoma y se cambia dicho valor aleatoriamente. Para alcanzar la solución del problema se puede fijar un número máximo de iteraciones antes de finalizar el algoritmo o detenerlo cuando no se produzcan más cambios en la población (convergencia del algoritmo).

Secuencia del algoritmo genético aplicado:

1. Se definen los parámetros del controlador a optimizar: k_σ .
2. Se genera una población inicial $X_{(M,m)}$ de forma completamente aleatoria. Donde M es el número de individuos de dicha población inicial y m es el número de variables. Para este caso, $M=50$ y $m=5$. La matriz X representa el conjunto de individuos de la población. La función utilizada para generar la población inicial es: $X_{(i,j)} = \text{random}(\text{"unif", li, ls})$; donde: $i=1,2,\dots,M$; $j=1,2,\dots,5$; $li=0.01$ y $ls=0.99$.
3. Se define la función objetivo a optimizar, (10).
4. Se calcula la función objetivo J para el conjunto de individuos, vector $X_{(i,m)}$; $i=1,2,\dots,M$. De esta manera se genera una matriz llamada $J_{(M,1)}$. A cada vector $J_{(i,1)}$ le corresponde un vector de parámetros $X_{(i,m)}$.
5. Se ordena $J_{(i,1)}$ de menor a mayor. Es válido aclarar, que en este paso, también se ordena la matriz $X_{(M,m)}$ correspondiente con el mismo orden que $J_{(i,1)}$. Se generan: J_{ord} y X_{ord} (matrices ordenadas de menor a mayor en función del mínimo valor de J). En los pasos siguientes se comienza con las operaciones de Selección, Cruzamiento y Mutación propias del algoritmo genético.
6. *Selección:* Se eligen los primeros 10 conjuntos de $X_{ord(M,m)}$: $X_{SE(10,m)} = X_{ord(10,m)}$. La matriz de selección representa el conjunto de "Padres" de la primera generación.
7. *Cruzamiento:* se establece una probabilidad de cruzamiento del 90%. A partir del conjunto de parámetros seleccionados en el paso anterior (padres), se realizan cinco cruzamientos que dan origen a 10 hijos, generando una matriz de cruzamiento: $X_{CR(10,m)}$. Para llevar a cabo la operación de cruzamiento, primero se elige de forma aleatoria los dos individuos a cruzar (conjunto de parámetros) y luego la posición (es decir, el parámetro) a cruzar para cada individuo.
8. *Mutación:* se establece una probabilidad de mutación y se llevan a cabo 20 mutaciones. A partir de la matriz selección se elige de forma aleatoria el individuo a mutar y luego la posición. Se genera $X_{MU(20,m)}$.
9. Por último, para evitar que el AG converja en un mínimo local, se generan 10 individuos más (completamente aleatorios) con el mismo procedimiento que en el paso 2. Se genera $X_{RA(10,m)}$.
10. Finalmente, se arma una generación nueva que surge de las diferentes operaciones ejecutadas a partir de la población inicial generada en el paso 2. La nueva generación está formada por las matrices de selección, cruzamiento, mutación y aleatoria: $X_{(0-10,m)} = X_{SE(10,m)}$; $X_{(10-20,m)} = X_{CR(10,m)}$; $X_{(20-40,m)} = X_{MU(20,m)}$; $X_{(40-50,m)} = X_{RA(10,m)}$. La nueva generación está representada entonces por $X_{(50,m)}$.
11. Con la nueva generación de individuos, se calcula la función J para cada individuo como en el paso 3 y se repiten los siguientes pasos.

Se establece $L=50$, siendo L el número de generaciones.

4.3. Algoritmo Híbrido (AH)

El tercer método propuesto consiste en un híbrido entre Monte Carlo y Algoritmos Genéticos. La población inicial se genera empleando Monte Carlo, pero en esta ocasión se define una

confianza $\delta=0.1$ y exactitud $\varepsilon=0.01$, lo que resulta, según (11), en $N=500$ simulaciones. Por lo tanto, se parte de una población inicial de dicho tamaño: $M_{inicial}=500$.

A partir de ésta población inicial se seleccionan los diez mejores individuos, es decir, aquellos para los cuales J es mínimo, y se forma la matriz $X_{SE}(10,m)$. Los pasos siguientes se corresponden con los efectuados en el algoritmo genético (desde el paso 7 en adelante).

La idea de complementar los dos métodos mencionados en un híbrido, surge de la necesidad de reducir la probabilidad de que el AG converja en un mínimo local, refinando sus resultados para encontrar el mejor conjunto de parámetros del controlador, tal que el error total de seguimiento, J , sea mínimo.

4.4. Selección de parámetros óptimos

Para la evaluación de los métodos mencionados se efectúa una serie de pruebas en las que se simula el proceso bajo condiciones normales de operación, para cada caso se calcula el error total y se considera el costo computacional, el cual se mide en términos de tiempo de ejecución de cada algoritmo.

En el caso de MC se llevaron a cabo 1000 simulaciones. Los resultados obtenidos pueden verse en la Tabla 1. La Fig. 2(a) muestra el conjunto de parámetros buscados aleatoriamente. Los parámetros óptimos del controlador corresponden a la iteración N° 503.

Tabla 1: Resultados de las simulaciones

MC		AG		AH	
J	t (min)	J	t (min)	J	t (min)
0.0052	45	0.0082	150	0.0025	200

Para la evaluación del AG, se realizaron simulaciones considerando distintos aspectos del mismo. Se probaron diferentes porcentajes de cruzamiento fijando número de

generaciones y porcentaje de mutación. También, para un grado de cruzamiento dado y un número de generaciones determinado, se probaron distintos porcentajes de mutación. Teniendo en cuenta los resultados que se muestran en la Tabla 2, no se observan grandes diferencias. Tampoco cambia significativamente la convergencia del algoritmo. Por lo tanto, para evaluar el sistema, se utilizó un grado de cruzamiento del 100% y un grado de mutación del 80%. Para asegurar la convergencia del algoritmo, el número de generaciones se definió en 50. Los resultados de las simulaciones del AG se muestran en la Tabla 1. En la Fig. 2(b) se observa la evolución del algoritmo.

Tabla 2: Simulaciones con AG

	%	J		%	J
Mutación (Mu)	20	0.0082	Cruzamiento (Cr)	50	0.0083
	80	0.0084		80	0.0082
Cr = 100 %			L = 50	100	0.0083
L = 50	100	0.0085			

En la Tabla 1 se observan los resultados del AH. En la Fig. 2(c) se muestra la combinación de búsqueda aleatoria y algoritmo genético para encontrar los parámetros óptimos del controlador.

Como puede observarse, con las tres alternativas de sintonización se obtienen rendimientos similares. Esta posibilidad de sintonizar con éxito el controlador utilizando diferentes estrategias evidencia otra ventaja de la técnica de control propuesta.

La Tabla 3 detalla los parámetros del controlador obtenidos con los tres algoritmos descritos. Si bien los resultados son muy similares, los parámetros correspondientes al AH son los seleccionados para efectuar las simulaciones de la próxima sección ya que permiten obtener el menor costo, J .

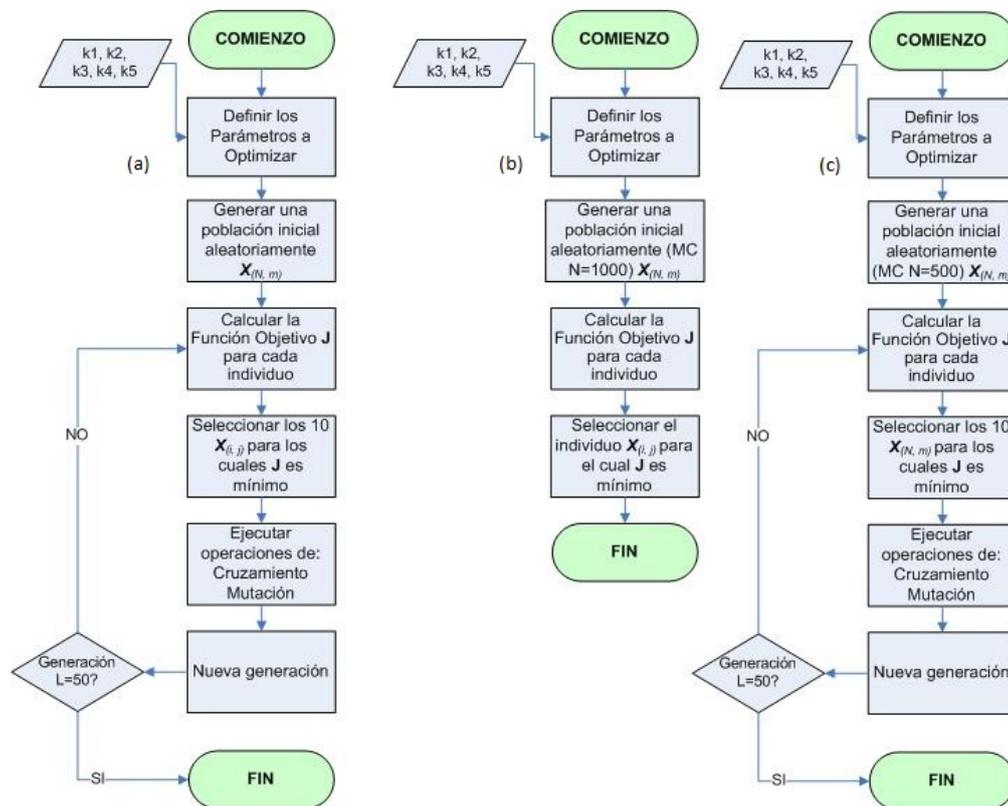


Fig. 1. Esquema de los métodos de sintonización de parámetros. (a) Algoritmos Genéticos, (b) Monte Carlo, (c) Híbrido.

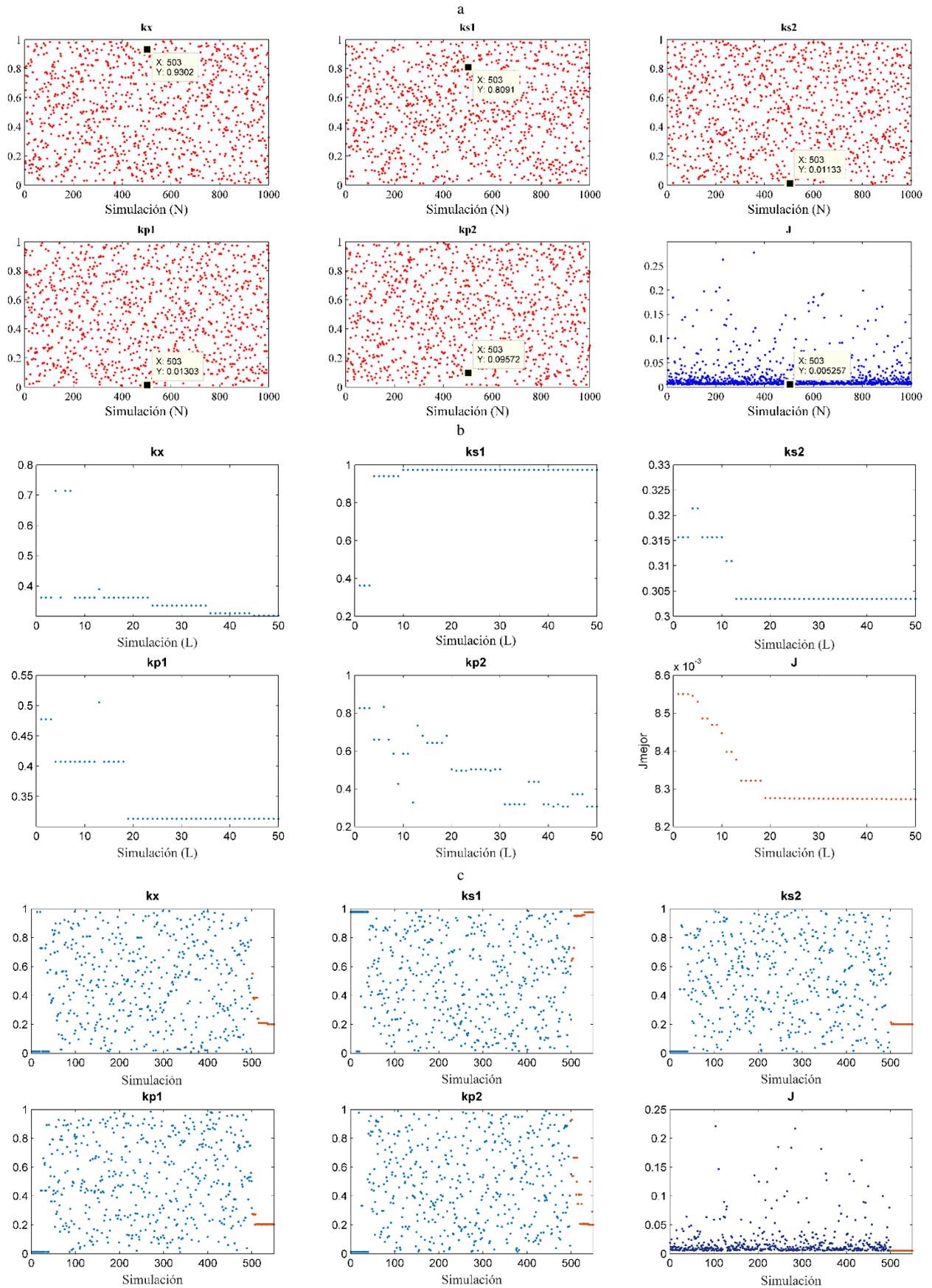


Fig. 2. Simulación para la sintonización de los parámetros del controlador. a) Monte Carlo, b) Algoritmos Genéticos, c) Híbrido.

Tabla 3: Parámetros del controlador

Algoritmo	k_x	k_{P1}	k_{P2}	k_{S1}	k_{S2}
MC	0.9302	0.0130	0.0957	0.8091	0.0113
AG	0.1558	0.2462	0.3187	0.9834	0.3039
AH	0.2153	0.2547	0.2889	0.9811	0.2716

La Fig. 3 muestra los resultados de simular el proceso con los tres grupos de parámetros de la Tabla 3. Puede apreciarse que la diferencia entre las corridas es ínfima, por lo que con cualquiera de las técnicas se puede obtener un excelente resultado.

Cabe destacar que el propósito de utilizar alguno de los métodos de sintonización propuestos es encontrar los k_{σ} que minimicen el error acumulado; sin embargo, al elegir dichos parámetros entre cero y uno, el error de seguimiento tenderá a cero, aunque no se apliquen las alternativas mencionadas.

5. Resultados y discusión

Para evaluar el desempeño del controlador se llevaron a cabo una serie de simulaciones empleando el software Matlab®. Se hace uso de los parámetros establecidos en la Tabla 3. En primer lugar, se simuló el sistema en condiciones normales de operación y luego, teniendo en cuenta incertidumbre paramétrica.

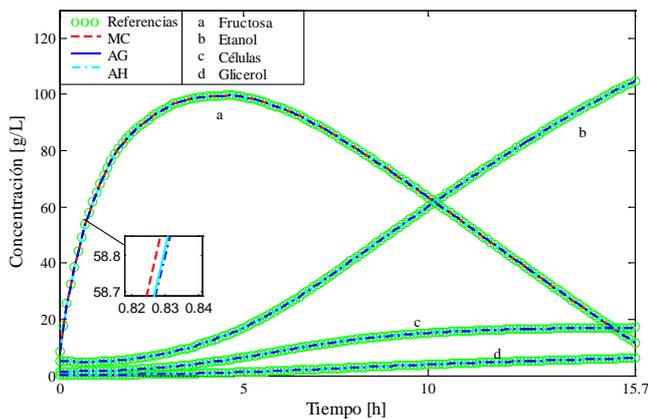


Fig. 3. Simulaciones utilizando los parámetros de controlador de la Tabla 3.

5.1. Desempeño del controlador en condiciones normales de operación

En esta sección se muestra el comportamiento del controlador propuesto en condiciones normales de operación, es decir, sin tener en cuenta incertidumbre paramétrica.

Los perfiles óptimos a seguir y los parámetros del sistema son aquellos presentados en (Hunag et al., 2012). Los parámetros del controlador son los determinados en la sección anterior, obtenidos con AH, ver Tabla 3.

En la Fig. 4 se muestra la evolución de las variables deseadas: concentración de células, fructosa, etanol y glicerol. Los perfiles de operación se comparan con los valores de referencia de (Hunag et al., 2012). Se puede ver cómo las variables de estado siguen perfectamente las referencias gracias a la acción de control. Esta

figura también ilustra el error de seguimiento ($\|e_n\|$), el cual a medida que el proceso avanza, tiende a disminuir y queda limitado a valores bajos, es decir, el controlador logra aproximar progresivamente el sistema a la referencia en cada instante de tiempo.

5.2. Desempeño del controlador bajo incertidumbre paramétrica

Una característica particular de los bioprocesos es la dificultad de conocer con exactitud los parámetros del sistema, los cuales, a menudo, varían con el tiempo (Wechselberger et al., 2010). Teniendo en cuenta esta situación, la siguiente prueba pretende evaluar la respuesta del controlador cuando los parámetros del sistema no son precisos o fluctúan a lo largo del proceso.

Con el fin de añadir incertidumbre al sistema, se propone utilizar el algoritmo de Monte Carlo ya descrito. En este caso, se consideran 1000 simulaciones (N), y a lo largo de cada una se varían aleatoriamente, en $\pm 15\%$ de su valor original, todos los parámetros del sistema (ver (Hunag et al., 2012)).

Los resultados de la simulación se muestran en la Fig. 4, donde se puede observar que el seguimiento de las variables deseadas se logra satisfactoriamente en el rango de variación establecido. El máximo E_p registrado es 0.509 correspondiente a la iteración 182.

5.3. Desempeño del controlador frente a un PID tradicional y un PID neuronal

A pesar de la existencia de técnicas sofisticadas de control para sistemas no lineales, los controladores PID siguen siendo los más utilizados en la industria (Imtiaz et al., 2014). Esto se debe a que poseen una estructura simple y es sencillo ajustar sus parámetros, en comparación con la complejidad de implementación en línea, la precisión insuficiente de las soluciones en línea y el tiempo computacional requerido para encontrar una solución por parte de otros controladores (Jin et al., 2014). Sin embargo, los controladores PID clásicos no responden adecuadamente frente a sistemas con largos retardos, salidas oscilatorias y subprocesos inestables, como es el caso de los bioprocesos. Es por ello que se desarrollaron estrategias que combinan la simplicidad de un PID con otras técnicas, y así poder aplicarlos en sistemas de esta índole (Imtiaz et al., 2014, Imtiaz et al., 2013, Pachauri et al., 2017).

En esta subsección se muestra una comparación entre el controlador propuesto, un PID tradicional y un PID neuronal, todos aplicados al mismo sistema de producción de etanol. El PID tradicional fue sintonizado de la misma manera que el controlador de álgebra lineal, utilizando las técnicas de optimización descritas, mientras que el PID neuronal fue diseñado siguiendo las pautas establecidas en (Rossomando and Soria, 2015a, Rossomando and Soria, 2015b). La Fig. 6 compara el funcionamiento de los tres controladores bajo condiciones normales de operación, en esta imagen es notable como las oscilaciones de la respuesta del PID tradicional aumentan el error de seguimiento, mientras que el desempeño del PID neuronal mejora levemente con respecto al anterior. En las Fig. 7 y 8 se muestran las respuestas de los controladores cuando la acción de control es afectada con dos perturbaciones distintas, ruido blanco y un escalón, respectivamente.

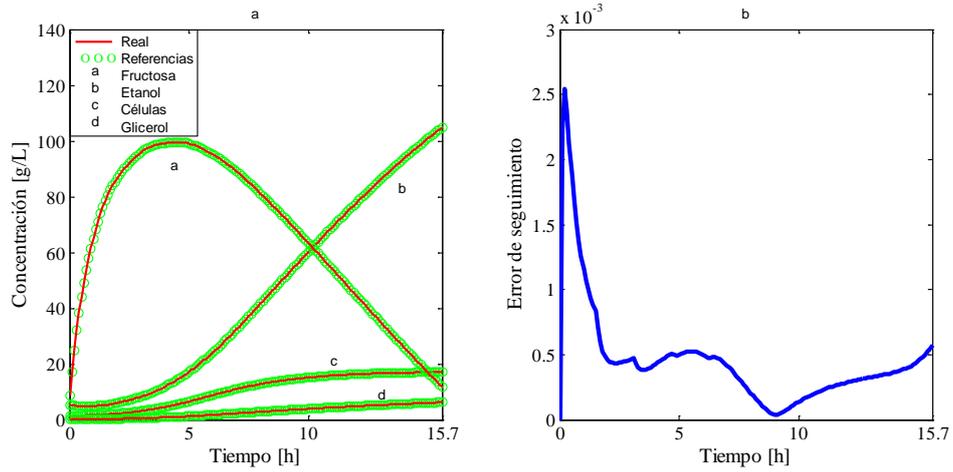


Fig. 4: Desempeño del controlador en condiciones normales de operación. a) Seguimiento de perfiles óptimos, b) Error de seguimiento.

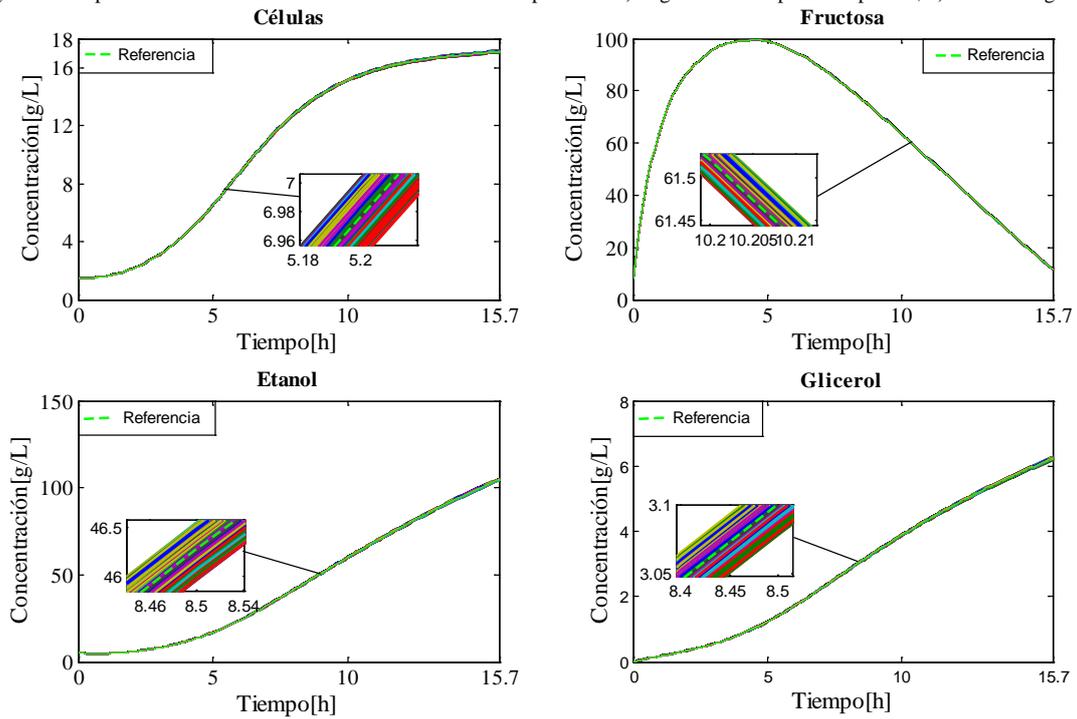


Figura 5: Seguimiento de perfiles óptimos teniendo en cuenta un $\pm 15\%$ de incertidumbre en todos los parámetros del sistema.

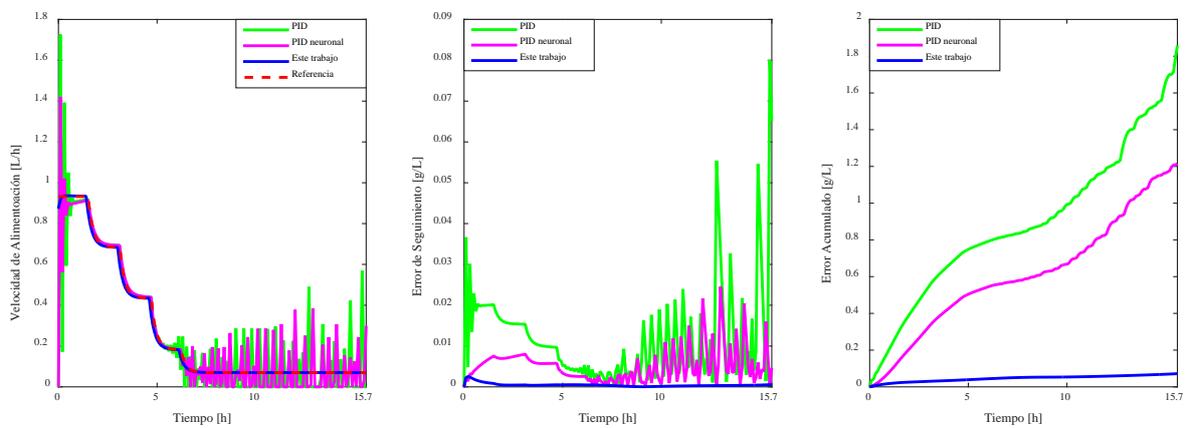


Figura 6: Comparación del funcionamiento de los tres controladores bajo condiciones normales de operación.

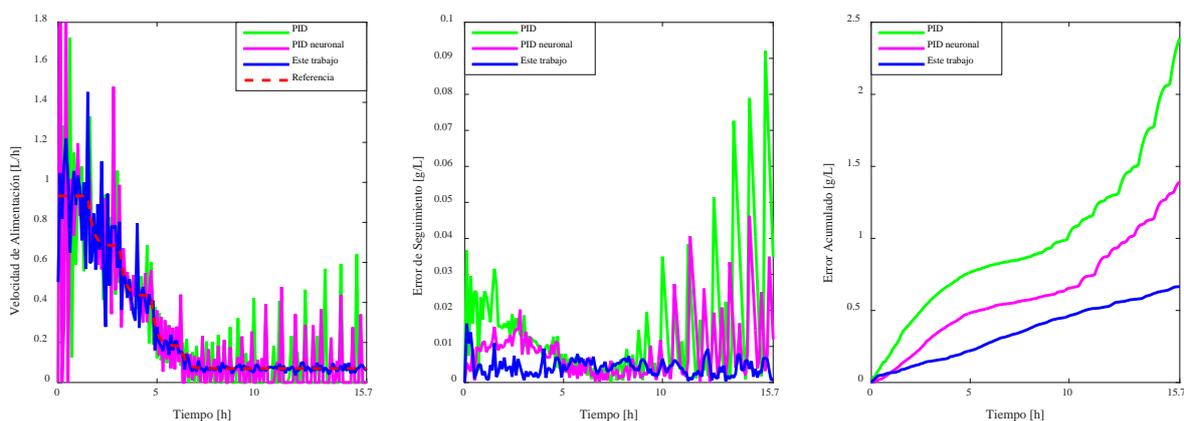


Figura 7: Comparación del funcionamiento de los tres controladores con perturbaciones de ruido blanco en la acción de control.

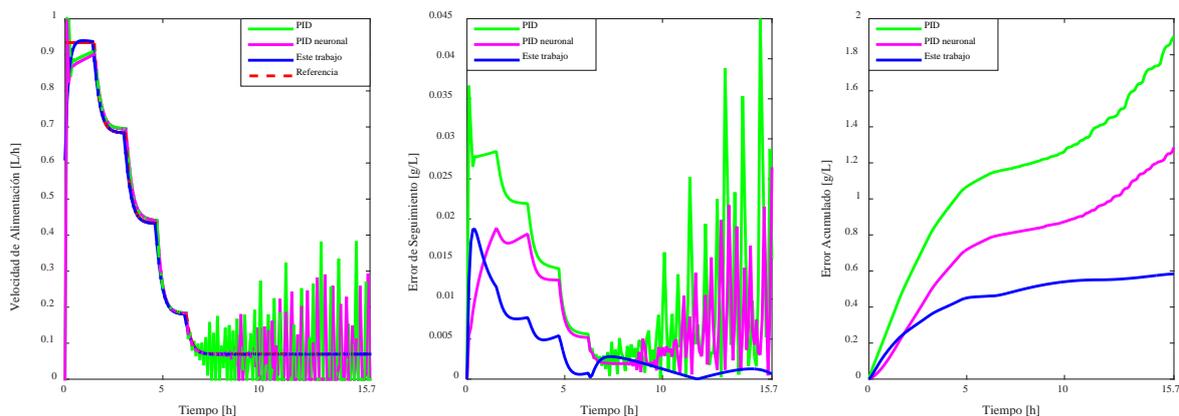


Figura 8: Comparación del funcionamiento de los tres controladores con una perturbación de escalón de en la acción de control.

6. Conclusión

En esta oportunidad, se ha presentado un controlador para sistemas no lineales multivariables y tres algoritmos distintos para su sintonización. La estrategia de control permite conocer las acciones de control a partir de un sistema de ecuaciones lineales, y, de esa manera, seguir trayectorias previamente definidas. En este caso, se aplicó la técnica a un biorreactor para la producción de bioetanol.

Las metodologías empleadas para la selección de los parámetros del controlador son: Monte Carlo, Algoritmos Genéticos y Algoritmo Híbrido, que surge de la combinación de las dos primeras. Las mismas pueden ser aplicadas en otras estrategias de control de la bibliografía. Además, las simulaciones demuestran rendimientos similares y excelentes resultados con las tres alternativas, lo que deja evidenciada una ventaja de la estrategia de control.

El desempeño del controlador fue evaluado bajo condiciones normales de operación, y sometiénolo a incertidumbre paramétrica. En ambos casos se observa un excelente seguimiento de los perfiles deseados, lo que lleva a la conclusión de que la técnica presentada es versátil frente a distintos cambios e incertidumbres en el sistema. Por otro lado, puede ser aplicada en otros sistemas no lineales multivariables sin tener mayores complicaciones en el planteo matemático.

Agradecimientos

Este trabajo ha sido realizado con el apoyo del Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) y

del Instituto de Ingeniería Química (IIQ) de la Universidad Nacional de San Juan. Se agradece la colaboración del Dr. Ing. Francisco Rossomando en la implementación del controlador PID neuronal.

Referencias

- Asadi, E., Da Silva, M. G., Antunes, C. H., Dias, L., Glicksman, L. 2014. Multi-objective optimization for building retrofit: A model using genetic algorithm and artificial neural network and an application. *Energy and Buildings*, 81, 444-456,
- Ashoori, A., Moshiri, B., Khaki-Sedigh, A., Bakhtiari, M. R. 2009. Optimal control of a nonlinear fed-batch fermentation process using model predictive approach. *Journal of Process Control*, 19, 7, 1162-1173,
- Bayen, T., Mairet, F. 2013. Minimal time control of fed-batch bioreactor with product inhibition. *Bioprocess and Biosystems Engineering*, 36, 10, 1485-1496, 10.1007/s00449-013-0911-9.
- Bogaerts, P., Coutinho, D. 2014. Robust nonlinear state estimation of bioreactors based on H_∞ hybrid observers. *Computers & Chemical Engineering*, 60, 315-328, <https://doi.org/10.1016/j.compchemeng.2013.09.013>.
- Bogaerts, P., Wouwer, A. V. 2003. Software sensors for bioprocesses. *ISA transactions*, 42, 4, 547-558,
- Cosenza, B., Galluzzo, M. 2012. Nonlinear fuzzy control of a fed-batch reactor for penicillin production. *Computers & Chemical Engineering*, 36, 273-281,
- Craven, S., Whelan, J., Glennon, B. 2014. Glucose concentration control of a fed-batch mammalian cell bioprocess using a nonlinear model predictive controller. *Journal of Process Control*, 24, 4, 344-357, <http://dx.doi.org/10.1016/j.jprocont.2014.02.007>.
- Chang, D. M. 2003. The Snowball Effect in Fed-Batch Bioreactions. *Biotechnology progress*, 19, 3, 1064-1070,

- Chung, Y.C., Chien, I.L., Chang, D.M. 2006. Multiple-model control strategy for a fed-batch high cell-density culture processing. *Journal of Process Control*, 16, 1, 9-26,
- Daoutidis, P., Zachar, M., Jogwar, S. S. 2016. Sustainability and process control: A survey and perspective. *Journal of Process Control*, 44, 184-206,
- Dewasme, L., Fernandes, S., Amribt, Z., Santos, L., Bogaerts, P., Wouwer, A. V. 2015. State estimation and predictive control of fed-batch cultures of hybridoma cells. *Journal of Process Control*, 30, 50-57,
- Fernández, M. C., Rómoli, S., Pantano, M. N., Ortiz, O. A., Patiño, D., Scaglia, G. J. 2018. A New Approach for Nonlinear Multivariable Fed-Batch Bioprocess Trajectory Tracking Control. *Automatic Control and Computer Sciences*, 52, 1, 13-24,
- Hassan, L. H., Moghavvemi, M., Almurib, H. A., Steinmayer, O. 2013. Application of genetic algorithm in optimization of unified power flow controller parameters and its location in the power system network. *International Journal of Electrical Power & Energy Systems*, 46, 89-97,
- Holland, J. H. 1975. *Adaptation in natural and artificial systems. An introductory analysis with application to biology, control, and artificial intelligence.* Ann Arbor, MI: University of Michigan Press,
- Hulhoven, X., Wouwer, A. V., Bogaerts, P. 2006. Hybrid extended Luenberger-asymptotic observer for bioprocess state estimation. *Chemical Engineering Science*, 61, 2, 7151-7160, <http://dx.doi.org/10.1016/j.ces.2006.06.018>.
- Hunag, W.H., Shieh, G. S., Wang, F.S. 2012. Optimization of fed-batch fermentation using mixture of sugars to produce ethanol. *Journal of the Taiwan Institute of Chemical Engineers*, 43, 1, 1-8,
- Intiaz, U., Assadzadeh, A., Jamuar, S. S., Sahu, J. N. 2013. Bioreactor temperature profile controller using inverse neural network (INN) for production of ethanol. *Journal of Process Control*, 23, 5, 731-742, <http://dx.doi.org/10.1016/j.jprocont.2013.03.005>.
- Intiaz, U., Jamuar, S. S., Sahu, J. N., Ganesan, P. B. 2014. Bioreactor profile control by a nonlinear auto regressive moving average neuro and two degree of freedom PID controllers. *Journal of Process Control*, 24, 11, 1761-1777, <http://dx.doi.org/10.1016/j.jprocont.2014.09.012>.
- Ismail, M., Moghavvemi, M., Mahlia, T. 2014. Genetic algorithm based optimization on modeling and design of hybrid renewable energy systems. *Energy Conversion and Management*, 85, 120-130,
- Jin, H., Chen, X., Yang, J., Wu, L., Wang, L. 2014. Hybrid intelligent control of substrate feeding for industrial fed-batch chlortetracycline fermentation process. *ISA transactions*, 53, 6, 1822-1837,
- Johnson, A. 1987. The control of fed-batch fermentation processes—a survey. *Automatica*, 23, 6, 691-705,
- Lee, J., Lee, S. Y., Park, S., Middelberg, A. P. J. 1999. Control of fed-batch fermentations. *Biotechnology Advances*, 17, 1, 29-48, [http://dx.doi.org/10.1016/S0734-9750\(98\)00015-9](http://dx.doi.org/10.1016/S0734-9750(98)00015-9).
- Mohanty, B., Panda, S., Hota, P. 2014. Controller parameters tuning of differential evolution algorithm and its application to load frequency control of multi-source power system. *International journal of electrical power & energy systems*, 54, 77-85,
- Mohd, N., Aziz, N. 2015. Control of bioethanol fermentation process: NARX-based MPC (NARX-MPC) versus linear-based MPC (LMPC). *CHEMICAL ENGINEERING*, 45,
- Ochoa, S., Wozny, G., Repke, J.-U. 2010. Plantwide optimizing control of a continuous bioethanol production process. *Journal of Process Control*, 20, 9, 983-998,
- Pachauri, N., Rani, A., Singh, V. 2017. Bioreactor temperature control using modified fractional order IMC-PID for ethanol production. *Chemical Engineering Research and Design*, 122, 97-112,
- Pantano, M. N., Serrano, M. E., Fernández, M. C., Rossomando, F. G., Ortiz, O. A., Scaglia, G. J. 2017. Multivariable Control for Tracking Optimal Profiles in a Nonlinear Fed-Batch Bioprocess Integrated with State Estimation. *Industrial & Engineering Chemistry Research*, 56, 20, 6043-6056,
- Rajarithnam, K., Gomm, J. B., Yu, D.-L., Abdelhadi, A. S. 2016. PID controller tuning for a multivariable glass furnace process by genetic algorithm. *International Journal of Automation and Computing*, 13, 1, 64-72,
- Rani, K. Y., Rao, V. R. 1999. Control of fermenters—a review. *Bioprocess Engineering*, 21, 1, 77-88,
- Renard, F., Wouwer, A. V. 2008. Robust adaptive control of yeast fed-batch cultures. *Computers & Chemical Engineering*, 32, 6, 1238-1248,
- Renard, F., Wouwer, A. V., Valentinotti, S., Dumur, D. 2006. A practical robust control scheme for yeast fed-batch cultures—an experimental validation. *Journal of Process Control*, 16, 8, 855-864,
- Rocha, M., Mendes, R., Rocha, O., Rocha, I., Ferreira, E. C. 2014. Optimization of fed-batch fermentation processes with bio-inspired algorithms. *Expert Systems with Applications*, 41, 5, 2186-2195,
- Rossomando, F. G., Soria, C. M. 2015a. Design and implementation of adaptive neural PID for non linear dynamics in mobile robots. *IEEE Latin America Transactions*, 13, 4, 913-918,
- Rossomando, F. G., Soria, C. M. 2015b. Identification and control of nonlinear dynamics of a mobile robot in discrete time using an adaptive technique based on neural PID. *Neural Computing and Applications*, 26, 5, 1179-1191,
- Sadatsakkak, S. A., Ahmadi, M. H., Bayat, R., Pourkiaei, S. M., Feidt, M. 2015. Optimization density power and thermal efficiency of an endoreversible Braysson cycle by using non-dominated sorting genetic algorithm. *Energy Conversion and Management*, 93, 31-39,
- Saint-Donat, J., Bhat, N., Mcavoy, T. J. 1991. Neural net based model predictive control. *International Journal of Control*, 54, 6, 1453-1468,
- Santos, L. O., Dewasme, L., Coutinho, D., Wouwer, A. V. 2012. Nonlinear model predictive control of fed-batch cultures of micro-organisms exhibiting overflow metabolism: assessment and robustness. *Computers & Chemical Engineering*, 39, 143-151,
- Sarkar, D., Modak, J. M. 2003. Optimisation of fed-batch bioreactors using genetic algorithms. *Chemical Engineering Science*, 58, 11, 2283-2296, 10.1016/S0009-2509(03)00095-2.
- Sarkar, D., Modak, J. M. 2004. Optimization of fed-batch bioreactors using genetic algorithm: multiple control variables. *Computers & Chemical Engineering*, 28, 5, 789-798,
- Soni, A. S., Parker, R. S. 2004. Closed-loop control of fed-batch bioreactors: A shrinking-horizon approach. *Industrial & engineering chemistry research*, 43, 13, 3381-3393,
- Strang, G. 2006. *Linear Algebra and Its applications*, USA.
- Tempo, R., Ishii, H. 2007. Monte Carlo and Las Vegas Randomized Algorithms for Systems and Control*: An Introduction. *European journal of control*, 13, 2, 189-203,
- Tholudur, A., Ramirez, W. F. 1996. Optimization of Fed-Batch Bioreactors Using Neural Network Parameter Function Models. *Biotechnology Progress*, 12, 3, 302-309,
- Troncoso, C., Suárez, A. 2017. Control del Nivel de Pulpa en un Circuito de Flotación Utilizando una Estrategia de Control Predictivo. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 14, 3, 234-245,
- Vilanova, R., Santín, I., Pedret, C. 2017. Control y Operación de Estaciones Depuradoras de Aguas Residuales: Modelado y Simulación. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 14, 3, 217-233,
- Vilums, S., Grigs, O. 2012. Application of functional state modelling approach for yeast *Saccharomyces cerevisiae* batch fermentation state estimation. 5th International Scientific Conference on Applied Information and Communication Technologies. Proceedings. 300-305.
- Vishal, V., Kumar, V., Rana, K., Mishra, P., Kumar, J. 2014. Online PI controller tuning for a nonlinear plant using genetic algorithm. *Computational Intelligence on Power, Energy and Controls with their impact on Humanity (CIPECH)*, 2014 Innovative Applications of, 2014. IEEE, 143-148.
- Wechselberger, P., Seifert, A., Herwig, C. 2010. PAT method to gather bioprocess parameters in real-time using simple input variables and first principle relationships. *Chemical Engineering Science*, 65, 21, 5734-5746,
- Yu, W., Li, B., Jia, H., Zhang, M., Wang, D. 2015. Application of multi-objective genetic algorithm to optimize energy efficiency and thermal comfort in building design. *Energy and Buildings*, 88, 135-143,

Apéndice A. Demostración de la tendencia del error a cero.

Como ya se mencionó anteriormente, los parámetros k_{σ} deben tomar valores entre cero y uno ($0 < k_{\sigma} < 1$), lo que permite que el error de seguimiento tienda a cero cuando n tienda a infinito. A continuación se detalla la demostración:

Resolviendo (6) con mínimos cuadrados se llega a la siguiente expresión:

$$U_n = (A^T A)^{-1} A^T b = \frac{a_1 b_1 + a_2 b_2 + a_3 b_3 + a_4 b_4 + a_5 b_5}{a_1^2 + a_2^2 + a_3^2 + a_4^2 + a_5^2} \quad (A.1)$$

A partir de (6):

$$\begin{aligned} \frac{a_1}{a_2} = \frac{b_1}{b_2} &\rightarrow b_2 = \frac{a_2}{a_1} b_1 \\ \frac{a_1}{a_3} = \frac{b_1}{b_3} &\rightarrow b_3 = \frac{a_3}{a_1} b_1 \\ \frac{a_1}{a_4} = \frac{b_1}{b_4} &\rightarrow b_4 = \frac{a_4}{a_1} b_1 \\ \frac{a_1}{a_5} = \frac{b_1}{b_5} &\rightarrow b_5 = \frac{a_5}{a_1} b_1 \end{aligned}$$

Sustituyendo (A.2) en (A.1):

$$\begin{aligned} U_n &= \frac{a_1 b_1 + (a_2^2 b_1) / a_1 + (a_3^2 b_1) / a_1 + (a_4^2 b_1) / a_1 + (a_5^2 b_1) / a_1}{a_1^2 + a_2^2 + a_3^2 + a_4^2 + a_5^2} \\ &= \frac{(b_1 / a_1)(a_1^2 + a_2^2 + a_3^2 + a_4^2 + a_5^2)}{a_1^2 + a_2^2 + a_3^2 + a_4^2 + a_5^2} = \frac{b_1}{a_1} \end{aligned} \quad (A.3)$$

Reemplazando (A.3) en (5):

$$X_{n+1} = X_{refn+1} - k_X (X_{refn} - X_n) + T_S [\mu_1(S_{1ezn}, P_{1n}) - \mu_1(S_{1n}, P_{1n})] X_n \quad (A.4)$$

Además:

$$e_{Xn+1} = X_{refn+1} - X_{n+1} \quad (A.5)$$

Combinando (A.4) y (A.5):

$$e_{Xn+1} = k_X (X_{refn} - X_n) - T_S [\mu_1(S_{1ezn}, P_{1n}) - \mu_1(S_{1n}, P_{1n})] X_n \quad (A.6)$$

La aproximación de Taylor de $\mu_1(S_{1n}, P_{1n})$ en el valor deseado $\mu_1(S_{1ezn}, P_{1n})$ es:

$$\mu_1(S_{1n}, P_{1n}) = \mu_1(S_{1ezn}, P_{1n}) + \left. \frac{d\mu_1(S_1, P_{1n})}{dS_1} \right|_{S_1=S_{1ezn} + \theta(S_{1n}-S_{1ezn})=S_{1\theta}} (S_{1n} - S_{1ezn}) \quad (A.7)$$

where $\rightarrow 0 < \theta < 1$

Reemplazando (A.7) en (A.6):

$$e_{Xn+1} = k_X (X_{refn} - X_n) + T_S \left. \frac{d\mu_1(S_1, P_{1n})}{dS_1} \right|_{S_{1\theta}} e_{S_{1n}} X_n \quad (A.8)$$

Combinando las expresiones (A.8) y aquellas obtenidas de repetir el mismo procedimiento para S_2, P_1 y P_2 , se llega a:

$$\begin{bmatrix} e_{Xn+1} \\ e_{S_{1n+1}} \\ e_{S_{2n+1}} \\ e_{P_{1n+1}} \\ e_{P_{2n+1}} \end{bmatrix} = \underbrace{\begin{bmatrix} k_X & 0 & 0 & 0 & 0 \\ 0 & k_{S_1} & 0 & 0 & 0 \\ 0 & 0 & k_{S_2} & 0 & 0 \\ 0 & 0 & 0 & k_{P_1} & 0 \\ 0 & 0 & 0 & 0 & k_{P_2} \end{bmatrix}}_L \begin{bmatrix} e_{Xn} \\ e_{S_{1n}} \\ e_{S_{2n}} \\ e_{P_{1n}} \\ e_{P_{2n}} \end{bmatrix} + T_S X_n \underbrace{\begin{bmatrix} \left. \frac{d\mu_1(S_1, P_{1n})}{dS_1} \right|_{S_{1\theta}} \\ 0 \\ 0 \\ \left. \frac{dq_{S_1/P_1}(S_1, P_{1n})}{dS_1} \right|_{S_{1\theta}} \\ \left. \frac{dq_{S_1/P_2}(S_1, P_{2n})}{dS_1} \right|_{S_{2\theta}} \end{bmatrix}}_{NL} e_{S_{1n}} \quad (A.9)$$

En (A.9), L es un sistema lineal que tiende a cero cuando k_{σ} toma valores entre cero y uno. NL es una no linealidad limitada que tiende a cero porque e_{S_n} tiene tendencia a cero. De esta manera queda evidenciado lo planteado en un comienzo.