Aplicación de dos Métodos de Inteligencia Artificial y uno de Gradiente para Controladores Predictivos Basados en Modelo no Lineal

Pablo García
Universidad Nacional de Colombia, Facultad de Minas, Grupo de Automática (GAUNAL)
Medellín, Colombia
pagarciari@unalmed.edu.co

y

Hernán D. Álvarez
Universidad Nacional de Colombia, Facultad de Minas, Grupo de Automática (GAUNAL)
Medellín, Colombia
hdalvare@unal.edu.co

Abstract

A comparison between two Artificial Intelligence (IA) optimization methods and a traditional gradient based method, applied to a model predictive controller (CPBM) is shown. The model used is non-linear and it includes the effect of a real pneumatic valve type Final Control Element (EFC), because that effect tends to adversely affect the performance of the CPBM when it's model has an ideal EFC. The CPBM application uses Genetic Algorithms and Bacterial Chemotaxis as IA techniques. The effectiveness of the techniques for a real implantation as well as the potential for IA techniques for CPBM are explored. The simulation uses an existent continuous stirred tank reactor (CSTR) benchmark problem.

Keywords: Model predictive control, Genetic Algorithms, Bacterial Chemotaxis.

Resumen

Se presenta una comparación entre dos métodos de optimización mediante técnicas de Inteligencia Artificial (IA) y un método tradicional de optimización por gradiente, todos aplicados a un Control Predictivo Basado en Modelo (CPBM). Se utiliza el modelo no lineal del proceso, incluyendo además el efecto de un Elemento Final de Control (EFC) real (tipo válvula neumática), puesto que tal efecto tiende a deteriora el comportamiento del CPBM cuando su modelo cuenta con un EFC ideal. La implantación del CPBM utiliza como técnicas de IA Algoritmos Genéticos y Quimiotactismo Bacteriano. Se pretende determinar cual resulta más efectiva para una implantación real y a su vez explorar la potencia de las técnicas de IA en un CPBM. La simulación utiliza un problema de referencia (bechkmark) existente para un reactor continuo en tanque agitado (CSTR).

Palabras claves: Control predictivo, Algoritmos Genéticos, Quimiotactismo Bacteriano.

1. Introducción

El Control Predictivo Basado en Modelo (CPBM) es una metodología de control moderna, la cual utiliza toda la información medible sobre el proceso, más un modelo de predicción de los estados dinámicos que lo representan, para tratar de encontrar la mejor acción de control solucionando un problema de optimización. El interés en el CPBM se debe a que el problema de optimización puede manejar restricciones, que son típicas en los procesos reales, como saturación en los elementos finales de control, valores de los estados que no son posibles, tales como concentraciones negativas de un reactivo o valores de estados que pongan en riesgo el proceso, como temperaturas o presiones extremas.

El problema de la implementación de un CPBM, es un problema de modelado y optimización. Este tipo de controladores necesitan de un buen modelo que prediga como es el comportamiento de la planta real. Idealmente, mientras mejor sea el modelo mejor será el control, pero, si la solución del modelo es más compleja el tiempo de muestreo mínimo aumentará, perjudicando el desempeño del controlador. Obtener la secuencia de acción de control óptima es un aspecto problemático del CPBM. El método de optimización debería encontrar la secuencia de acción de control óptima en el menor tiempo posible.

Este trabajo se enfoca en la evaluación de tres métodos de optimización para la implementación de un CPBM en un problema de referencia (Benchmark): un reactor continuo en un tanque perfectamente agitado (CSTR por sus siglas en Inglés), en el cual ocurre una reacción de primer orden. El método de gradiente evaluado es el de punto interior de la rutina *fmincon* de Matlab(R), los de inteligencia artificial son el algoritmo genético de la rutina *ga* de Matlab(R) y un algoritmo de quimiotactísmo bacteriano desarrollado en Matlab(R) por los autores de este artículo.

2. Control Predictivo Basado en Modelo

2.1 Definición del Problema.

El CPBM es un método de control óptimo que selecciona las acciones de control mediante la minimización de una función de costo, este problema puede formularse matemáticamente como:

$$\min_{x} J(x) \quad x \in \Omega \quad (1)$$

Donde x es la variable de optimización, $J(x)$ es la función de costo y Ω es el conjunto que restringe los valores que puede tomar x. Como los CPBMs se implementan en computadoras digitales lo usual es usar modelos en tiempo discreto, considerando como tiempo de muestreo el de lectura de los sensores. Un modelo no lineal en tiempo discreto puede describirse como:

$$x_{(k+1)} = f\left(x_k, u_k\right) \quad (2)$$

Donde x_k y u_k son vectores del estado y de la acción de control. El objetivo del CPBM consiste en encontrar una secuencia de acciones control [$u_{k+0|k}$, $u_{k+1|k}$,…., $u_{k+Nc-1|k}$], dados los estados presentes (x_k) conocidos, que minimice una función de costo, relacionada con el comportamiento futuro del sistema, que se predice gracias a un modelo dinámico del mismo, denominado modelo de predicción.

$$J\left(x_{k+j|k}, u_{k+j|k}, \Delta u_{k+j|k}\right) \quad (3)$$

Donde $x_{k+j|k}$, y $u_{k+j|k}$, son los valores de los estados y entradas que estén disponibles en el instante k y puedan ser predichos para los instantes $k+j$. Adicionalmente, $\Delta u_{k+j|k}$ es la diferencia entre la u actual y la u en un tiempo de muestreo anterior definida como:

$$\Delta u_{k+j|k} = u_{k+j|k} - u_{k+j-1|k} \quad (4)$$

El intervalo de tiempo futuro que se considera en la optimización se denomina horizonte de predicción Np. El horizonte de control Nc, la cantidad de muestras futuras en las que se optimizará la acción de control, determina el número de variables del problema de optimización. Así, reducir el horizonte de control facilita la solución de dicho problema.[1]

La función de costo se puede diseñar de forma que cumpla con una variedad de objetivos, minimizar la desviación en la salida y penalizar cambios bruscos en el elemento final de control. Es común utilizar un criterio de error cuadrático:

$$J = \sum_{i=0}^{Np} \alpha_{i|k} \| x^*_{k+i|k} - x^*_{ref\,i|k} \|^2_Q + \sum_{i=1}^{Nc} \beta_{j|k} \| \Delta u_{k+j|k} \|^2_R \quad (5)$$

Donde $x^* \subset x$ son los estados controlados. Usando el modelo de predicción en la función objetivo, esta se puede reducir a una función que solo depende de los estados actuales y la secuencia de acciones de control, así:

$$J\left(x_k, u_{k+j|k}\right) \quad (6)$$

2.2. El elemento final de control.

Muchas veces se desprecia la importancia dinámica del Elemento Final de Control (EFC), como un componente importante del lazo de control que afecta la dinámica global del proceso real. La tendencia corriente es la consideración de la curva instalada del EFC, pero se olvida considerar su respuesta dinámica, bajo el supuesto de alta velocidad de estos elementos. Sin embargo, cuando el EFC es una válvula neumática, su respuesta dinámica depende del tamaño nominal de la misma. Por ejemplo, en válvulas neumáticas de más de 3" es corriente encontrar tiempos de respuesta (τ) de varios segundos e incluso un minuto. Por esta razón, en este trabajo se tiene en cuenta el efecto dinámico del EFC, asumiendo que la válvula que puede representarse por medio de una dinámica de primer orden.

3. Métodos de Optimización

3.1 Método de punto interior

Los algoritmos basados en gradiente utilizan esta información ya que apunta en la dirección de mayor aumento de la función objetivo, así su vector opuesto apunta en la dirección de máximo descenso. Con este vector se genera una dirección de búsqueda sobre la que se toma un paso para llegar a un nuevo valor de la función objetivo que resulta menor [2]. El método de punto interior para optimización restringida se basa en resolver una secuencia de problemas de optimización aproximados de manera que el problema de optimización

$$\min_u J(u) \quad \text{s.a. } h(u)=0 \text{ y } g(u)\le 0 \qquad (7)$$

se convierte para cada $\mu > 0$ en:

$$\min_{u,s} J_\mu(u,s)=\min_u J_\mu(u)-\mu\sum_i \ln(s_i) \quad \text{s.a. } h(u)=0 \text{ y } g(u)\le 0 \qquad (8)$$

Donde hay tantas variables de relajación s_i como restricciones de desigualdad g. A medida que μ se acerca a 0 el mínimo de J_μ debería acercarse al de J [3]. El término logarítmico adicional se conoce como *función de barrera* y posee las siguientes propiedades:

$$f_{bar}(x)\approx 0 \text{ para } \max_i g_i(x) \ll 0$$

$$f_{bar}(x)\to\infty \text{ para } \max_i g_i(x)\uparrow 0$$

3.2 Algoritmos Genéticos

Los algoritmos genéticos son un método para resolver problemas de optimización restringidos que se basa en la selección natural, el proceso que gobierna la evolución biológica. El algoritmo genético modifica repetidamente una población de individuos solución. En cada paso, el algoritmo genético selecciona individuos de la población de acuerdo al valor de la función objetivo, llamados padres, para producir la próxima generación de *hijos*. En el transcurso de varias generaciones la población *evoluciona* hacia la solución óptima. Los algoritmos genéticos pueden aplicarse en la solución de una gran variedad de problemas de optimización incluyendo aquellos que no son apropiados para los algoritmos de optimización estándar como aquellos en los que la función objetivo es discontinua, no diferenciable, estocástica o altamente no lineal.[4]

El algoritmo genético comienza con la creación de una población inicial de elementos generados aleatoriamente. Después se crea una secuencia de generaciones de individuos. En cada generación se usan sus individuos para producir los de la siguiente generación por medio de los siguientes pasos:

- Ordenar los individuos de acuerdo con el valor que toma la función objetivo para cada uno.
- Escalar el valor para convertirlo en una escala más fácil de usar.
- Seleccionar los elementos *padres* basado en el valor de la función objetivo escalada.
- Los individuos con valores más bajos se consideran individuos *elite* y sobreviven para la siguiente generación.
- Producir *hijos* a partir de los *padres*, estos se producen a partir de la *mutación* de un individuo o del *entrecruzamiento* de dos.
- Reemplaza la generación actual con los *hijos* generados para producir la siguiente generación.

El algoritmo se detiene cuando se alcanza un criterio de finalización como un número máximo de generaciones, un tiempo máximo, un valor de la función objetivo dado, un número máximo de generaciones sin disminución en el valor de la función objetivo mínima, entre otros.

Matlab(R), en las herramientas de algoritmos genéticos y búsqueda directa (*Genetic Algorithms and Direct Search Toolbox*) implementa un algoritmo genético para optimización restringida por medio de la función *ga*.

3.2 Quimiotactísmo Bacteriano

El algoritmo de Quimiotactismo Bacteriano (QB o *Bacterial Chemotaxis* en inglés) fue originalmente propuesto en (Bremmerman and Anderson, 1990) como una optimización numérica de los pesos de una red neuronal artificial de propagación hacia delante. El mecanismo de búsqueda del óptimo por QB imita el movimiento típico de una colonia de bacterias dentro de un caldo de cultivo. Tales movimientos son una función directa de la concentración de nutrientes en dicho caldo de cultivo. La funcionalidad del movimiento se logra por un cierto código de comunicación que provee información sobre la concentración de nutrientes en varias direcciones alrededor de la posición actual de la colonia. El movimiento que finalmente dará la colonia en el próximo instante será en aquella dirección en la que la concentración de nutrientes incremente (Spiro et al., 1997; Kearns and Shimkets, 1998). El mismo criterio de movimiento en busca de un óptimo matemático se aplica durante las búsquedas numéricas de un máximo (o de un mínimo tomando el reciproco de la función) de cualquier función multivariable.

En un primer paso del algoritmo de QB se determinan los intervalos nominales y se establece un valor semilla para cada u_k que se va a optimizar. De este modo el algoritmo de QB inicia su trabajo explorando varias direcciones en el espacio de los parámetros. El algoritmo de QB básico puede ser descrito como la siguiente secuencia de pasos, considerando una única bacteria como si fuera la colonia completa:

1. Se toma de manera provisional el valor semilla de cada uno de los u_k como u_0. Se calcula el valor de la función de objetivo J con los valores actuales (u_0) y se toma dicho valor como el óptimo inicial J_0.
2. Considerando el intervalo de variación de cada u (u_{imin}, u_{imax}), se determina aleatoriamente matriz de términos de suma S con las mismas dimensiones de u. Este procedimiento acepta valores negativos o positivos la matriz.
3. Se un nuevo u usando la matriz S: $u_p = u_n + S$
4. Usando u_p se determina el valor de la función de objetivo J_p.
5. Si $J_p > J_n$ entonces u_p se almacena como u_{n+1} y se mantiene el S actual. Si $J_p < J_n$ entonces se mantiene el u_n original y se determina aleatoriamente un nuevo S. Si se genera un cierto número de S que producen pasos no válidos, se reduce el intervalo de generación de S y se vuelve al paso 2.
6. Si todavía no se cumple la condición de parada del algoritmo, se retorna al paso 3. De lo contrario, el algoritmo termina entregando como respuesta el u_n.

Respecto a la condición de parada del algoritmo, existen tres alternativas. La primera considera un número fijo de ciclos de reducción del intervalo de generación de la matriz S. La segunda es un número máximo de pasos válidos. La tercera consiste en evaluar el cambio en el valor de la función objetivo, si $J_n + J_{n+1} < \varepsilon$, donde ε es un valor de tolerancia la optimización concluye.

4. MODELO DINÁMICO DEL PROCESO

Para la aplicación de los controladores se utiliza un modelo de un CSTR en el que se produce 3,4-dimetilbifenilo por la isomerización del 2,2-dimetilbifenilo. La isomerización se lleva a cabo en un tanque con un volumen de trabajo de 7854 litros, el cual tiene un diámetro de 2 metros. El reactor está equipado con una chaqueta de enfriamiento que opera a presión atmosférica. Los objetivos de control para el proceso son la concentración de salida del 2,2'-dimetilbifenilo (CA 0.1 mol/l) y la temperatura del reactor (T 373.6 K). La reacción de isomerización es una reacción exotérmica irreversible de primer orden del tipo A→B, los parámetros cinéticos de la reacción se presentan en la Tabla 1:

Tabla 1 Parámetros cinéticos de la reacción

Parámetro cinético	Valor
Energía de activación	22.6 kcal/mol
Entalpía de reacción	-3.2 kcal/mol
Factor de frecuencia	5.37×10^{11} s^{-1}

Las propiedades físicas del dimetilbifenilo se muestran en la Tabla 2

El modelo dinámico se construye a partir de los balances de masa y de energía para el reactor, con las siguientes suposiciones: i) las propiedades físicas como la densidad y la capacidad calorífica de la solución son constantes en todo momento, ii) el mezclado en el reactor es perfecto, iii) no hay pérdidas de calor hacia el ambiente, iv) los caudales de entrada y salida del reactor son iguales. Con estas suposiciones se obtiene el siguiente sistema dinámico:

$$\frac{dC_A}{dt}=\frac{q}{V}(C_{A0}-C_A)-k_0 C_A \exp\left(-\frac{E}{RT}\right) \quad (6)$$

$$\frac{dT}{dt}=\frac{q}{V}(T_{A0}-T_A)-\frac{\Delta H k_0 C_A \exp\left(-\frac{E}{RT}\right)}{\rho Cp}-\frac{UA}{\rho Cp V}(T-T_j) \quad (7)$$

$$\frac{dT_j}{dt}=\frac{q_j}{V_j}(T_{0j}-T_j)-\frac{UA}{\rho Cp_r V_j}(T-T_j) \quad (8)$$

Tabla 2. Propiedades físicas dimetilbifenilo

Propiedad física	Valor
Peso molecular	182.26 kg/kmol
Densidad	866 kg/m^3
Capacidad calorífica	1.842 kJ/kg K

Considerando que el tiempo de residencia del refrigerante en la chaqueta es corto y que la temperatura de la chaqueta es aproximadamente la misma en toda la chaqueta pero diferente a la entrada, es posible despreciar la dinámica de la temperatura de la chaqueta. Al despreciar esta dinámica es posible representar la variable Tj por medio de la ecuación algebraica resultante del balance de energía para la chaqueta en estado estacionario usando la diferencia media logarítmica de temperaturas (LMTD). Esta ecuación se reemplaza en el balance de temperatura para obtener:

$$\frac{dT}{dt}=\frac{q}{V}(T_{A0}-T_A)-\frac{\Delta H k_0 C_A \exp\left(-\frac{E}{RT}\right)}{\rho Cp}-\frac{q_j \rho_r Cp_r}{\rho Cp V}(T_{0j}-T)\left[1-\exp\left(-\frac{UA}{q_j \rho_r Cp_r}\right)\right] \quad (9)$$

Tabla 3. Parámetros de diseño y operación del reactor

Parámetro de diseño	Valor
Volumen	7854 l
Flujo volumétrico nominal en el reactor	347.7 l/min
Flujo volumétrico nominal en la chaqueta	206.1 l/min
Concentración de A en la entrada	4.656 mol/l
Temperatura en la entrada	373 K
Coeficiente de transferencia de calor	2.44 kcal/s K m^2
Temperatura en la entrada de refrigerante	350 K

La representación del modelo en el espacio de estados es:

$$\dot{x_1}=\frac{u_2}{V}(C_{A0}-x_1)-k_0 x_1 \exp\left(-\frac{E}{R x_2}\right) \quad (10)$$

$$\dot{x_2}=\frac{u_2}{V}(T_0-x_2)-\frac{\Delta H k_0 x_1 \exp\left(-\frac{E}{R x_2}\right)}{\rho Cp}+\frac{u_2 \rho_r Cp_r}{\rho Cp V}(T_{0j}-x_2)\left[1-\exp\left(-\frac{UA}{u_2 \rho_r Cp_r}\right)\right] \quad (11)$$

Donde las variables de estado son $x_1=C_A$ y $x_2=T$, y las variables de entrada son el flujo de entrada al reactor $u_1=VR_i$ y a la chaqueta $u_2=VC_i$

Con el objetivo de establecer una diferencia entre el modelo de la planta y el modelo de predicción, se agrega al modelo de la planta un modelo con la dinámica de las válvulas que manipulan los flujos de entrada al reactor y a la chaqueta. Tales válvulas tienen un máximo de flujo de 500 lt/min para la válvula del reactor y de 600 lt/min para la válvula de flujo en la chaqueta. Para caracterizar la dinámica del EFC, se considera que las válvulas tienen un comportamiento de respuesta dinámica de primer orden con tiempo de respuesta de 30s. La representación en el espacio de estados es:

$$\frac{du_i}{dt}=\frac{1}{\tau}u \quad (12)$$

5. Resultados

Las simulaciones se realizaron utilizando Simulink(R) perturbando la planta con un escalón del 10% del valor de estado estacionario de C_{Ai} pasados 5 min de tiempo de simulación, en todos los algoritmos de optimización se usaron restricciones para *u* con los valores máximos y mínimos de flujo . El sistema se controló con un CPBM sintonizado con los siguientes parámetros:

Tabla 4. Parámetros del CPBM

Parámetro	Valor
Tiempo de muestreo	1 min
Horizonte de predicción	20 muestras
Horizonte de control	5 muestras
Q	[100 0; 0 1]
R	[1 0; 0 1]
α	1
β	0.0075

5.1 Método de gradiente

El método de gradiente se implementó usando la rutina *fmincon* de las herramientas de optimización de Matlab(R) con un máximo de evaluaciones de la función objetivo de 2000 y el algoritmo de punto interior.

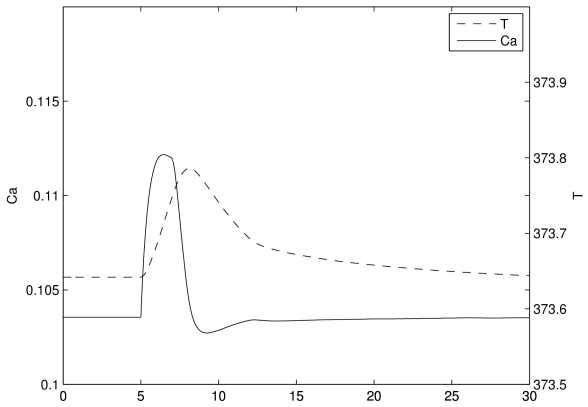

Figura 1. Trayectorias de las variables controladas, CPBM con el método de gradiente.

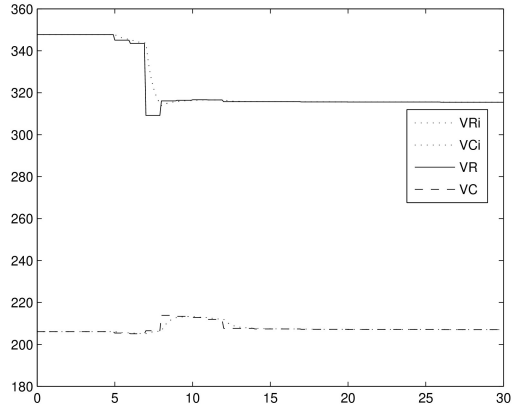

Figura 2. Trayectorias de las variables manipuladas, CPBM con el método de gradiente.

5.2 Algoritmo genético

El algoritmo genético utilizado fue el implementado en la rutina *ga* de las herramientas de búsqueda directa y algoritmos genéticos de Matlab(R). Dado que el tiempo de muestreo del controlador es de 1 min el algoritmo está limitado a una ejecución máxima de 1 min, adicionalmente la población inicial se limita a -10% +10% del valor de la *u* en el instante anterior y se garantiza que la *u* anterior hace parte de esta población inicial. El límite de generaciones es de 200 y de 25 generaciones sin mejora.

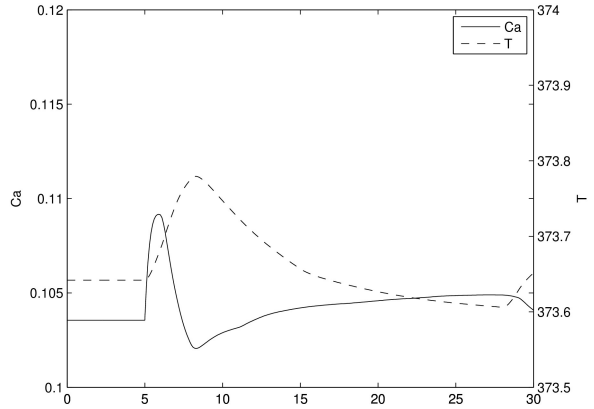

Figura 3 Trayectorias de las variables controladas, CPBM con algoritmo genético.

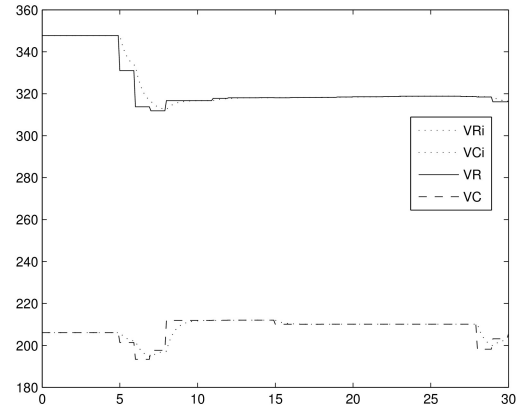

Figura 4 Trayectorias de las variables manipuladas, CPBM con algoritmo genético.

5.3 Quimiotactismo bacteriano

Para este trabajo se desarrolló un algoritmo de quimiotactismo bacteriano como el descrito anteriormente. Se usaron los siguientes parámetros: pasos no validos para reducción 10, número máximo de reducciones del intervalo de S continuas 10, divisor del tamaño del intervalo de *S* 10 y número máximo de pasos válidos 2500.

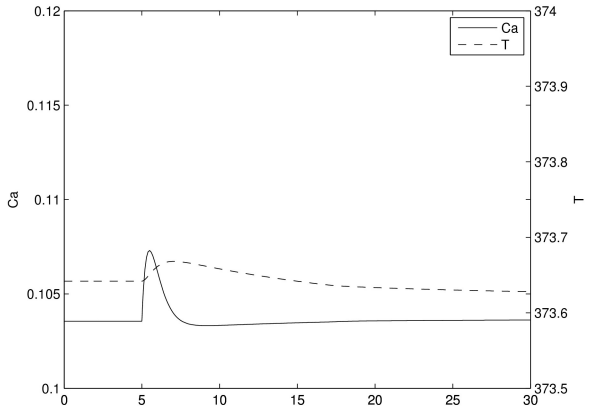

Figura 5 Trayectorias de las variables controladas, CPBM con quimiotactismo bacteriano.

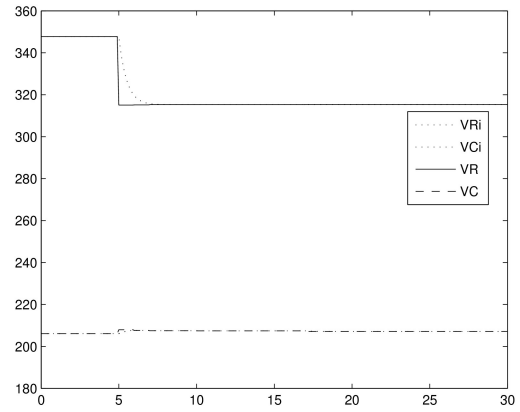

Figura 6 Trayectorias de las variables manipuladas, CPBM con quimiotactismo bacteriano.

5.4 ITAE y tiempo de optimización

El ITAE es la integral ponderada en el tiempo del error absoluto, se calculó para los diferentes controladores usando la definición en [5].

Tabla 5. ITAE

Método	ITAE Ca	ITAE T
Gradiente	0.1778	10.71
Algoritmo Genético	0.474	14
Quimiotactismo Bacteriano	0.05382	4.625

Para comparar la eficiencia en el computo de la solución al problema de optimización las tres simulaciones se realizaron en la misma máquina (Intel(R) Centrino(R) Duo 1.6 Ghz, 1GiB RAM) y se tomó el tiempo de duración de la optimización por medio de la función *toc* de Matlab(R). La siguiente tabla reporta el tiempo máximo, que es el tiempo que tardó la optimización más lenta y el tiempo total que corresponde a la suma de todos los tiempos de las optimizaciones.

Tabla 6. Tiempo de optimización, datos en segundos

Método	Tiempo máximo	Tiempo total
Gradiente	18.7363	198.4828
Algoritmo Genético	39.2556	332.8457
Quimiotactismo Bacteriano	49.8373	301.4682

6. Conclusiones

Se presentaron tres métodos de optimización diferentes para resolver el problema del Control Predictivo Basado en Modelo (CPBM). Dos de los métodos pertenecen a los Biológicamente Inspirados: Quimiotactismo Bacteriano (QB) y Algoritmos Genéticos. Se logró el objetivo de ver la aplicabilidad de tales métodos frente al método tradicional de gradiente, aunque la magnitud del problema tratado (particularmente por la magnitud de las dimensiones), permitió que el método de gradiente tuviera desempeño adecuado. Además, tener en el Bechmark usado tiempos de computo hasta de 1 minuto, dejó suficiente holgura para que los tres métodos funcionaran correctamente. Sin embargo, se evidencia de los resultados del proceso controlado, que el método de QB resultó el mejor método de optimización para el CPBM, aunque su tiempo de ejecución no es el mejor (cercano a 50 segundos frente a 19 para el método de Gradiente). La capacidad que tuvo el CPBM con QB para rechazar la perturbación eficientemente, se evidencia en su bajísimo valor del índice ITAE. Posiblemente este efecto se deba a una mejor sintonía lograda en el QB gracias a su sencillez y a lo intuitivo de su operación, contrastado con la relativa complejidad funcional del Algoritmo Genético y las particularidades del ajuste del algoritmo de Gradiente.
Queda abierto un campo de trabajo que complete algunas demostraciones matemáticas del comportamiento del algoritmo de QB, así como otras propiedades matemáticas que garanticen su operación eficiente independientemente de la aplicación que se aborde. Adicionalmente, la estabilidad del CPBM usando QB como algoritmo de solución de la optimización, debe ser comprobada y certificada matemáticamente siguiendo el conocimiento aplicable a la prueba de estabilidad de sistemas dinámicos operando en lazo cerrado.

Referencias

1. Limón, D. *Control predictivo de sistemas no lineales con restricciones: estabilidad y robustez. Tesis Doctoral*. Universidad de Sevilla, Sevilla, 2002.
2. Beers, K. J. *Numerical Methods for Chemical Engineering, Applications in Matlab(R)*. Cambridge University Press, Cambridge, 2007.
3. The Math Works. *Optimization Toolbox(TM) 4, Users Guide.* The MathWorks, Natick, 2009.
4. The MathWorks. *Genetic Algorithm and Direct Search Toolbox(TM), User's Guide*. The MathWorks, Natick, 2009.
5. Smith, C. A. y Corripio, A. B. *Principles and Practice of Automatic Process Control*. John Wiley and Sons, Inc. Nueva York, 1997.