

Evolución diferencial para la optimización global de procesos de ingeniería química

Maritza Yanet Martínez Zecua, Luis Angel Salamanca Vázquez,
Leticia Flores Pulido, Edgar Alfredo Portilla Flores,
Arturo Ortiz Arroyo

- ¹ Universidad Autónoma de Tlaxcala, Facultad de Ciencias Básicas, Ingeniería y Tecnología, Apizaco, Tlaxcala, México
- ² Instituto Politécnico Nacional, Centro de Innovación y Desarrollo Tecnológico en Cómputo, Ciudad de México, México
- yanetzecua.13@gmail.com, angel.salamanca.vazquez@gmail.com,
leticia.flores.p@uatx.mx, arturo.ortiz@uatx.mx, aportilla@ipn.mx

Resumen. El diseño de reactores es un aspecto fundamental para la producción de casi todas las sustancias químicas industriales, ya que la selección de un sistema de reacción que opere de la forma más segura y eficiente posible puede ser la clave del éxito o del fracaso productivo de una planta de productos químicos. Existen dos grandes procesos dentro de una planta química: los reactores y los procesos en redes de calentadores de sustancias químicas. En este artículo se definen dos problemas de esta clase por medio de el enfoque de optimización. Una vez que se hace esta clase de definiciones, se procede a optimizar sus funciones con la intención de mejorar los mínimos propuestos por planteamientos previos en el estado del arte por medio de técnicas no tradicionales, es decir, haciendo uso de técnicas heurísticas. Se implementaron entonces seis variantes del algoritmo de evolución diferencial. Cada una de las variantes optimiza las funciones de ambos problemas dentro del contexto de ingeniería química, alcanzando mínimos comparablemente mas rapido que enfoques previos. Se realiza una comparativa entre las seis variantes del algoritmo y se menciona cual de ellos alcanza una mejor factibilidad y convergencia.

Palabras clave: evolución diferencial, diseño de intercambiador de calor, diseño de red de reactores, poptimización, metaheurística.

Evolution for the Global Optimization of Chemical Engineering Processes

Abstract. The design of reactors is a fundamental aspect for the production of almost all industrial chemical substances, since the selection of a reaction system that operates in the safest and most efficient way possible can be the key to success or the productive failure of a chemical products plant. There are two major processes within a chemical plant: reactors and processes in networks of chemical substance heaters. In this

article two problems of this kind are defined by means of the optimization approach. Once this kind of definition is made, we proceed to optimize its functions with the intention of improving the minimums proposed by previous approaches in the state of the art by means of non-traditional techniques, that is, making use of technical heuristic. Six variants were then implemented in the differential evolution algorithm. Each of the variants optimizes the functions of both problems within the context of chemical engineering, reaching minimums comparably faster than previous approaches. A comparison is made between the six variants of the algorithm and it is mentioned which of them achieves a better feasibility and convergence.

Keywords: evolution on differential, design or heat exchanger, design or reactor network, optimization, metaheuristic.

1. Introducción

Los algoritmos evolutivos (AE) basan su funcionamiento en el proceso de selección natural propuesto por Charles Darwin. Estos algoritmos clasificados como metaheurísticos no aseguran obtener el óptimo global, sin embargo producen una muy buena solución en un tiempo de cómputo razonable, además que no se requiere del conocimiento específico del problema a resolver. Debido a que se pretende que el método de solución no tenga una alta sensibilidad a las condiciones iniciales para la búsqueda de soluciones, se eligió que las estrategias del algoritmos evolutivos utilizadas en el presente trabajo fueran las siguientes: generación aleatoria de una población inicial y creación de nuevas direcciones de búsqueda basadas en la distribución de las soluciones (individuos) de la población actual. Se busca entonces, que el proceso de optimización avance, y que la selección de las direcciones de búsqueda (las cuales en un principio son altamente aleatorias) estén basadas en las mejores soluciones encontradas a partir de la población actual. Un algoritmo que satisface lo establecido anteriormente es el denominado Evolución Diferencial (ED) [2]. Uno de los trabajos realizados en aplicaciones de ingeniería química es el trabajo realizado en [3] donde Evolución Diferencial es aplicada en dos problemas no-lineales de ingeniería química, además de que se realiza la comparación de su rendimiento con el del algoritmo ramificación y acotación, estos problemas nosotros también los aplicamos y así mismo comparamos los resultados que ellos obtuvieron con los nuestros.

Más adelante se presenta Evolución Diferencial Modificado (EDM) para la optimización de procesos químicos no-lineales[4], en el cual se elabora una modificación a lo que es Evolución Diferencial (ED), mediante el argumento de que a pesar de que ED ha demostrado ser un eficiente, efectivo y robusto método de optimización evolutiva, aun toma largo tiempo computacional para optimizar costosas funciones objetivas, por lo tanto se considero necesario mejorar la tasa de convergencia sin comprometer la calidad de la solución. En dicha investigación el algoritmo de EDM es implementado en cinco problemas de ingeniería química no lineales, comparando los resultados obtenidos contra Evolución Diferencial

(ED) considerando el historial de convergencia (tiempo de CPU y el número de ejecuciones convergidas a óptimo global) así como las técnicas estadísticas, tomando en cuenta la variabilidad de los resultados. Otro trabajo similar es el realizado en [5], donde se resuelven problemas que pertenecen a la amplia clase de NLP restringidas dos veces diferenciables. El algoritmo se basa en un enfoque de ramificación y unión, donde se obtiene un límite inferior en la solución óptima en cada nodo mediante la generación automática de un problema válido de subestimación convexo.

Finalmente en [6] por medio de búsqueda y ramificación se resuelven múltiples problemas no lineales aplicandose en diseño de procesos, de igual manera se muestran los problemas de la red de calentadores y de reactores químicos con planteamientos de optimiación desde otros enfoques. Este artículo entonces describe la propuesta de el uso de evolución diferencial para resolver un problema de optimización de aplicaciones de ingeniería química, se analiza el algoritmo de ED y finalmente se muestran resultados de factibilidad y convergencia, comparables con los planteamientos dentro del estado del arte. Se logra identificar la variante mas eficiente de ED para cada problema y se mejoran los resultados obtenidos en planteamientos variados para las mismas aplicaciones.

2. Propuesta de investigación

El reactor químico realiza un proceso de concentración de soluciones y concentraciones químicas, a partir de ciertas condiciones y funciones ya definidas que permiten modelar un proceso de optimización para generar concentraciones químicas como nafta, gasolina y otro tipo de sustancias derivadas del petróleo. Posteriormente, se dan dos problemas de optimización como entrada al algoritmo de evolución diferencial en seis variantes y se procede a realizar experimentos para analizar cual de ellos arroja un mejor desempeño para la función objetivo a los casos de estudio presentados por medio de comparaciones, después con los resultados obtenidos en la optimización se procede a mostrarlo mediante una interfaz demostrativa. Este proceso se presenta en la Figura 1.

3. Evolución diferencial

La optimización es un proceso de búsqueda, de la mejor solución o conjunto de soluciones en un problema específico, bajo circunstancias determinadas, como método para alcanzarla se utiliza Evolución Diferencial (ED) donde la idea principal es un nuevo esquema para generar vectores. ED es un algoritmo de optimización estocástico propuesto por Storn y Price en 1995 para resolver problemas de optimización numérica [1]. Se trata de un algoritmo evolutivo, altamente competitivo en problemas de optimización global, que basa su estrategia en la búsqueda poblacional. De este modo, ED parte de una población de NP individuos, también llamados vectores, que representan las soluciones candidatas al problema en cuestión, como se muestra en la ecuación (1):

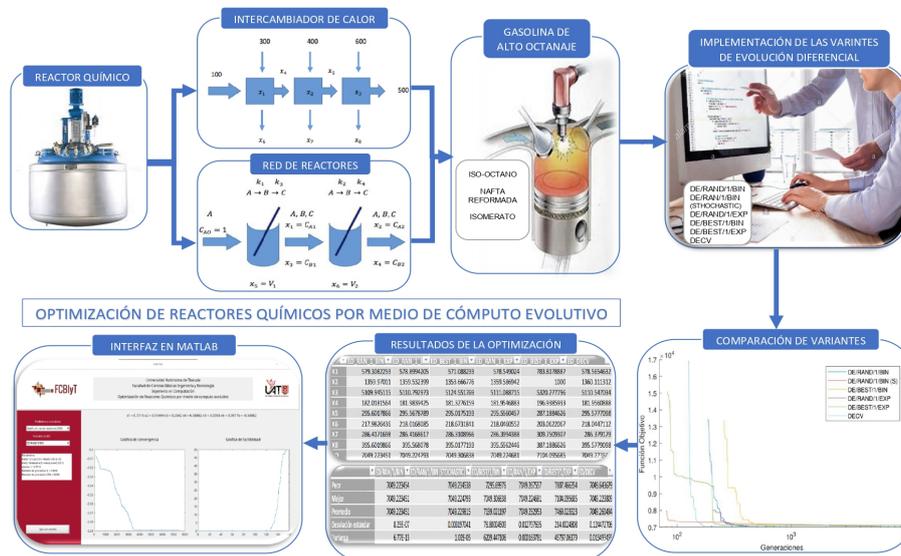


Fig. 1. Propuesta de Evolución Diferencial para la Optimización de Procesos en Ingeniería Química.

$$x_{g,i} = (x_{g,i,j}), j = 1, \dots, D, i = 1, \dots, NP, \quad (1)$$

donde $x_{g,i,j}$ es una variable de decisión del problema, los parámetros de cada vector son indexados con la variable j , que se extiende desde 1 hasta D . Los vectores son identificados a través de un índice de población, i . La g corresponde al número de la generación actual, $g = 1, \dots, gmax$. A continuación se describen cada una de las etapas del algoritmo de ED.

En la *Etapa de Inicialización*, los individuos de la población son inicializados aleatoriamente dentro de sus límites inferiores (L_j) y superiores (U_j), es decir: $L_j \leq x_j \leq U_j$. Dichos individuos serán sometidos a operaciones de mutación, recombinación y selección para producir nuevos descendientes que participarán en la siguiente generación reemplazando a la anterior. En la *Etapa de Mutación*, se enfatiza la dirección de búsqueda puesto que se controla la magnitud con la que los vectores se desplazan en el espacio de búsqueda y la velocidad de convergencia hacia la solución óptima. Por consiguiente, la mutación diferencial añade una diferencia de dos vectores ponderada por un factor escala, a un tercer vector, todos ellos seleccionados aleatoriamente, como se muestra en la ecuación (2):

$$v_{g,i} = x_{g,r1} + F(x_{g,r1} - x_{g,r1}), r_1 \neq r_2 \neq r_3 \neq i \quad (2)$$

El parámetro $F \in [0, 1)$ controla el rango en que la población evoluciona. En la *Etapa de Cruza (CR)* se toma como un proceso complementario a la mutación

con la cual se logra el intercambio de información entre dos o más vectores para crear uno o más vectores hijo. Valores altos implican menor influencia del padre. Existen dos tipos de recombinación: discreta (el vector hijo obtiene los parámetros del vector padre o del vector mutante) y la continua o aritmética (el vector hijo hereda sus parámetros por medio de una combinación lineal de vectores. En la *Etapa de Selección* se evalúa si el valor de la función objetivo del vector trial $u_{g,i}$ es menor (asumiendo minimización) que el vector target $x_{g,i}$, de ser así, reemplazará su posición en la siguiente generación, de otro modo, el mejor se mantiene, como se expresa en la ecuación (3):

$$x_{g+1,i} = \begin{cases} u_{g,i} & \text{si } (u_{g,i}) \leq f(x_{g,i}) \\ x_{g,i} & \text{de otro modo} \end{cases} \quad (3)$$

A continuación, en la Figura 2 se muestra el tradicional algoritmo de Evolución Diferencial implementado en los experimentos.

```

1: Inicio
2: Crear una población inicial aleatoriamente
    $x_{0,i} \ i = 1, \dots, NP$ 
3: Evaluar población  $f(x_{0,i}) \ i = 1, \dots, NP$ 
4: for g=0 to MAX_GEN do
5:   for i=1 to NP do
6:     Seleccionar aleatoriamente  $r_1 \neq r_2 \neq r_3 \neq i$ 
7:      $j_{rand} = randint[1, D]$ 
8:     for j=1 to D do
9:       if ( $rand_j[0,1] < CR$  or  $j = j_{rand}$ ) then
10:         $u_{g,i,j} = x_{g,r1,j} + F(x_{g,r2,j} - x_{g,r3,j})$ 
11:       else
12:         $u_{g,i,j} = x_{g,i,j}$ 
13:       end if
14:     end for
15:     if  $f(u_{g,i}) \leq f(x_{g,i})$  then
16:       $x_{g+1,i} = u_{g,i}$ 
17:     else
18:       $x_{g+1,i} = x_{g,i}$ 
19:     end if
20:   end for
21: end for
22: Fin

```

Fig. 2. Algoritmo de Evolución Diferencial, variante DE/rand/1/bin.

Existe un esquema de nomenclatura desarrollado para hacer referencia a las variantes del algoritmo de Evolución Diferencial, utilizando la siguiente notación: **DE/x/y/z**. Donde: **DE** se refiere al algoritmo Evolución Diferencial (Differential Evolution), **x** especifica la forma en la que el vector base (r_0) es seleccionado, puede ser de forma aleatoria (rand), el mejor individuo hasta el momento (best), entre otros. **y** indica el número de diferencias de vectores que contribuyen al

diferencial. \mathbf{z} representa el tipo de recombinación que se utiliza, en las versiones del algoritmo DE se utiliza recombinación binomial **bin** o exponencial **exp**.

4. Implementación del diseño de red de intercambiador de calor

Los intercambiadores de calor son equipos principalmente para transferir calor entre corrientes calientes y frías. Tienen caminos separados para las dos corrientes y operan continuamente. El diseño de la red del intercambiador de calor se realiza para hacer frente a una situación en la que una variedad de flujos de procesos deben sufrir cambios de estado simultáneos que se pueden lograr mediante la adición y / o eliminación de calor. Las ventajas económicas concurrentes con el uso eficiente de flujos de proceso, como medios de calentamiento o enfriamiento, sirven como incentivo para administrar cuidadosamente las especificaciones de los equipos auxiliares e instalaciones tales como enfriadores de agua y vapor de los calentadores encendidos. Bajo las circunstancias habituales, sin embargo, no está claro de antemano cómo hacer frente a las demandas de calor y los suministros de manera que se optimice el rendimiento del sistema requerido. Este problema aborda el diseño de una red de intercambiador de calor como se muestra en la Figura 3. Un flujo frío debe ser calentado de 100 °F hasta 500 °F utilizando tres flujos calientes con diferentes temperaturas de entrada. El objetivo es minimizar el tamaño del área total de intercambio de calor. Este problema se ha tomado como referencia de Floudas & Pardalos, con un planteamiento realizado en 1990 [8]. Incluso este caso de estudio ha sido resuelto por Adjiman en 1998 [5] utilizando el algoritmo $\alpha\beta$, también conocido como el algoritmo de búsqueda y ramificación.

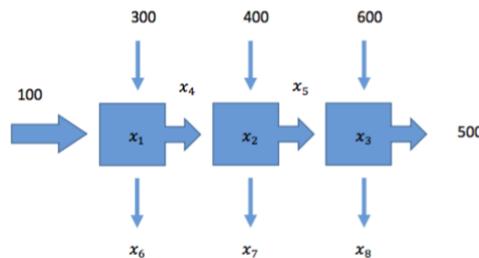


Fig. 3. Caso de Estudio 1: Diseño de la Red de Intercambio de Calor.

La función a minimizar está dada por la ecuación (4):

$$\min f(x) = x_1 + x_2 + x_3. \quad (4)$$

Sujeto a las restricciones descritas en las ecuaciones de la (5) a la (10):

$$g_1(x) = 0.0025(x_4 + x_6) - 1 = 0, \quad (5)$$

$$g_2(x) = 0.0025(-x_4 + x_5 + x_7) - 1 = 0, \quad (6)$$

$$g_3(x) = 0.01(-x_5 + x_8) - 1 = 0, \quad (7)$$

$$g_4(x) = 100x_1 - x_1x_6 + 833.33252x_4 - 83333.333 \leq 0, \quad (8)$$

$$g_5(x) = x_2x_4 - x_2x_7 - 1250x_4 + 1250x_5 \leq 0, \quad (9)$$

$$g_6(x) = x_3x_5 - x_3x_8 - 2500x_5 + 1250000 \leq 0, \quad (10)$$

donde los rangos de las variables son sujetos a las expresiones (11), (12) y (13):

$$100 \leq x_1 \leq 10000, \quad (11)$$

$$1000 \leq x_2, x_3 \leq 10000, \quad (12)$$

$$10 \leq x_4, x_5, x_6, x_7, x_8 \leq 1000, \quad (13)$$

donde x_1, x_2 y x_3 son áreas de intercambio de calor y x_4, x_5, x_6, x_7 y x_8 son temperaturas de flujos, como se muestra en la Figura 4. La mejor solución está localizada en los puntos mostrados en la expresión (14):

$$x^* = \langle 579.19, 1360.13, 5109.92, 182.01, 295.60, 217.9, 286.40, 395.60 \rangle. \quad (14)$$

Se sabe además que $f(x)^* = 7049.25$.

5. Implementación del diseño de red de reactores

Este es un ejemplo propuesto por Ryoo & Sahinidis en 1995 [6], es un problema de diseño de una red de reactores, descrito en el sistema mostrado en la Figura 4. Involucra el diseño de una secuencia de dos reactores de *tanque agitado continuo* (CSTR), donde la reacción consecutiva $A \rightarrow B \rightarrow C$ toma lugar. El objetivo es maximizar la concentración del producto B en la salida del flujo. Este problema es conocido por causar dificultades para algunos métodos de optimización global [8].

La función a minimizar está dada por la ecuación (15):

$$\min f(x) = -x_4. \quad (15)$$

Sujeto a las siguientes restricciones de las ecuaciones de la (16) a la (20):

$$g_1(x) = x_1 + k_1x_2x_5 = 1, \quad (16)$$

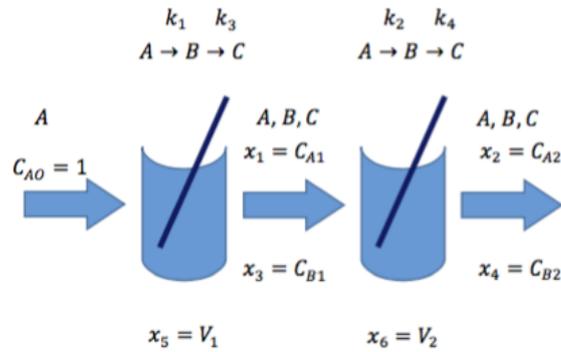


Fig. 4. Caso de Estudio 2: Diseño de la Red de Reactores.

$$g_2(x) = x_2 - x_1 + k_2 x_2 x_6 = 0, \quad (17)$$

$$g_3(x) = x_3 + x_1 + k_3 x_3 x_5 = 1, \quad (18)$$

$$g_4(x) = x_4 - x_3 + x_2 - x_1 + k_4 x_4 x_6 = 0, \quad (19)$$

$$g_5(x) = x_5^{0.5} + x_6^{0.5} \leq 4, \quad (20)$$

donde los rangos de las variables son se definen en las expresiones (21) a la (26):

$$0 \leq x_1 \leq 1, \quad (21)$$

$$0 \leq x_2 \leq 1, \quad (22)$$

$$0 \leq x_3 \leq 1, \quad (23)$$

$$0 \leq x_4 \leq 1, \quad (24)$$

$$10^{-5} \leq x_5 \leq 16, \quad (25)$$

$$10^{-5} \leq x_6 \leq 16. \quad (26)$$

Y las constantes tienen los valores descritos en las expresiones (27) a la (30):

$$k_1 = 0.09755988, \quad (27)$$

$$k_2 = 0.99k_1, \quad (28)$$

$$k_3 = 0.0391908, \quad (29)$$

$$k_4 = 0.9k_3. \quad (30)$$

El óptimo global está descrito en la expresión (31):

$$x^* = \langle 0.771462, 0.516997, 0.204234, 0.388812, 3.036504, 5.096052 \rangle. \quad (31)$$

Se sabe además que $f(x)^* = -0.388812$. Este caso constituye un caso de prueba muy complejo puesto que se puede observar un mínimo local con una función objetivo que es muy cercana a la solución global. Las soluciones locales son $f = -0.375$ y $f = -0.3881$. Por si no fuera suficientemente interesante, las dos soluciones locales utilizan solamente uno de los dos reactores mientras la solución global hace uso de ambos reactores.

6. Parámetros de sintonización

Al tratar con Algoritmos Evolutivos (AE), no se puede dejar a un lado la fuerte dependencia de sus parámetros de entrada con respecto al comportamiento del algoritmo, ya que éstos juegan un papel importante en la búsqueda de mejores soluciones; de ahí que realizar una configuración adecuada de sus parámetros sea crucial para obtener un correcto desempeño del algoritmo. La configuración de parámetros aplicada en nuestro trabajo, es aquella que suele hacerse sin una clara justificación del valor asignado, siendo a prueba y error, entre un rango de valores aleatorios con una calibración de parámetros, los valores adecuados para ellos son obtenidos mediante un proceso previo donde se ejecuta el algoritmo muchas veces con diferentes valores de parámetros, y mediante validación estadística o búsqueda, se proveen aquellos valores que proporcionen buenos resultados.

En esta última sección se muestra el análisis de los parámetros de las diferentes versiones de Evolución Diferencial donde se determinaron los parámetros relevantes del algoritmo y sus correspondientes rangos de aplicación para cada problema de optimización. Para ello las seis variantes del algoritmo se trabajaron bajo los mismos parámetros a excepción del número de generaciones respecto a cada problema, como se muestra en la Tabla 1 y en la Tabla 2.

7. Gráficas de factibilidad y convergencia

A continuación se muestran las gráficas de factibilidad para ambos casos de estudio. También se muestran las tablas de convergencia para las 6 variantes de ED.

Tabla 1. Tabla de parámetros para el Diseño de Intercambiador de Calor.

Parámetro	Valor
Factor de Cruza	Aleatorio[0.8,1.0]
Factor de Mutación	Aleatorio[0.3,0.9]
Número de población	50
Generacion máxima	3000
Epsilon	1×10^{-6}

Tabla 2. Tabla de parámetros para el Diseño de Red de Reactores.

Parámetro	Valor
Factor de Cruza	aleatorio[0.8,1.0]
Factor de Mutación	aleatorio[0.3,0.9]
Número de población	50
Generacion máxima	40000
Epsilon	1×10^{-6}

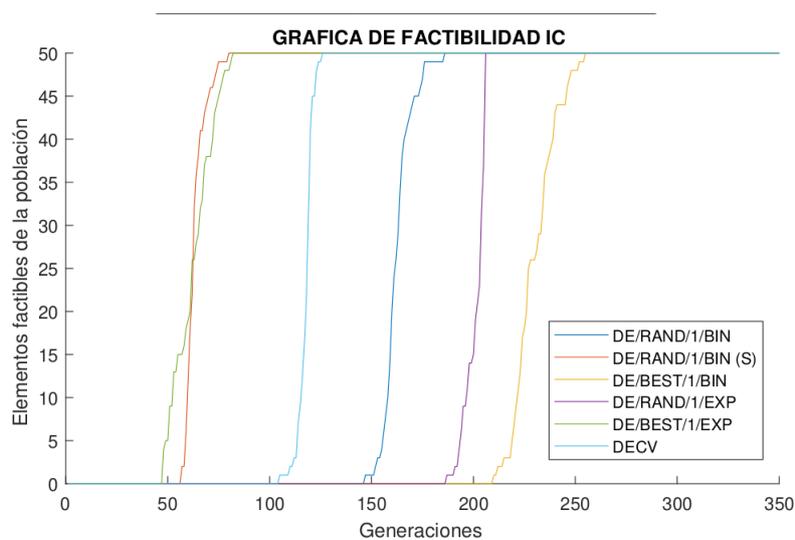


Fig. 5. Gráfica de Factibilidad del diseño de intercambiador de calor.

En la Figura 5, se visualiza que para el Caso 1: Red de intercambiador de calor, tanto el algoritmo de DE/RAND/1/BIN y DE/BEST/1/EXP son los que encuentran mas soluciones factibles que el resto de las variantes de ED. Esto indica que el espacio de búsqueda es explorado de manera mas eficiente, encontrando el óptimo en la mayoría de los casos. Dicha gráfica de factibilidad puede observarse en la Figura 5.

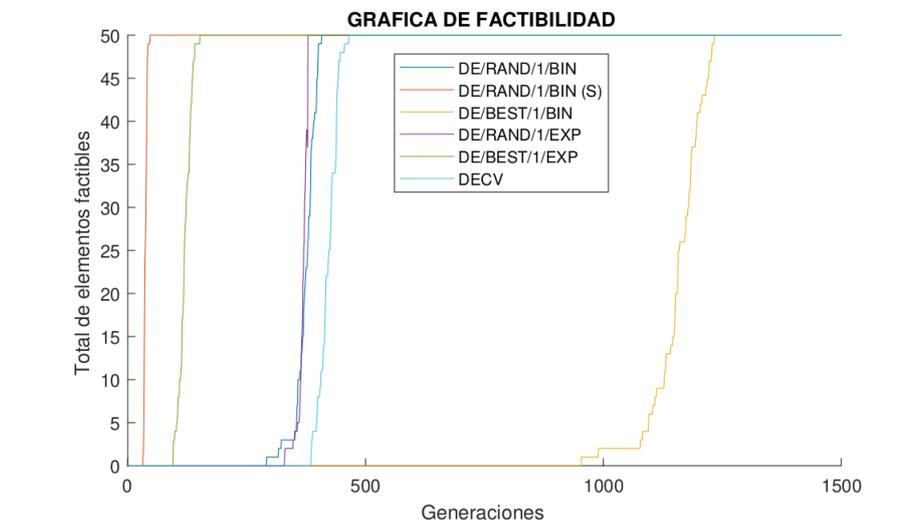


Fig. 6. Gráfica de Factibilidad del diseño de red de reactores.

Respecto a la factibilidad para el Caso 2: Red de reactores, puede mencionarse que DE/RAND/1/BIN, logró nuevamente ser la variante más eficiente en cuanto a la exploración del espacio de búsqueda en zonas factibles (ver Figura 6). Por otro lado, en cuanto a las convergencias de los algoritmos, podemos observar que para el Caso 1: Red de intercambiador de calor, se tiene una convergencia casi inmediata con DE/RAND/1/BIN, como se observa en la Figura 7. Para la convergencia del Caso 2: Red de reactores, se muestra una convergencia inmediata con DE/RAND/1/EXP (ver Figura 8).

8. Conclusiones

Una vez analizadas las factibilidades y convergencias, podemos mostrar que en los casos de estudio de procesos de ingeniería química los resultados obtenidos indican que respecto al primer caso de estudio con el *intercambiador de calor*, definido previamente en tres trabajos planteados en [5] y [6], se logran los mismos valores que ellos reportan en su literatura, sin embargo con el planteamiento de [3], se logra alcanzar un mejor valor que el que se muestra en los artículos.

En lo que corresponde al segundo caso de estudio, es decir, a la *red de reactores*, los resultados de [7] y [8] solo logran encontrar óptimos locales. Pero en el planteamiento realizado en [6] y debido a su correcta exposición del problema, el resultado es el óptimo global planteado en su artículo, no obstante es importante destacar que con ED logramos un mejor resultado respecto al óptimo global; teniendo en cuenta que no se reporta un ϵ en ninguno de los artículos. dicho lo anterior, se utilizó un ϵ de 1×10^{-6} puesto que los autores presentan

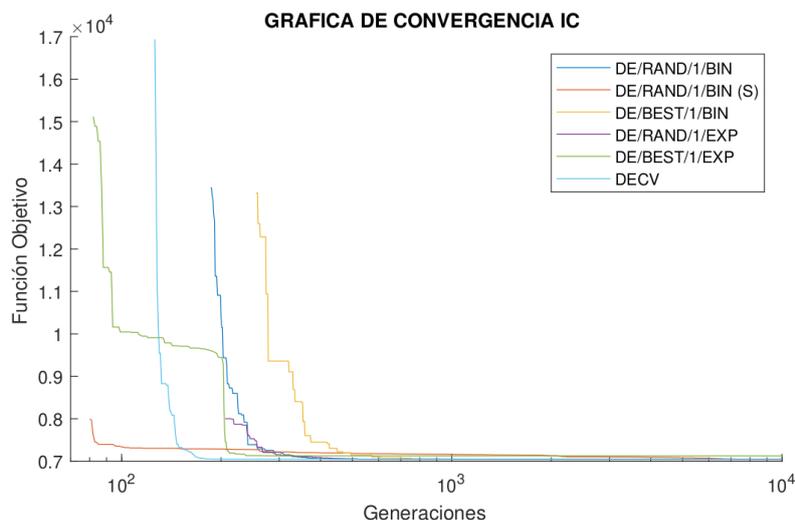


Fig. 7. Gráfica de Convergencia del diseño de intercambiador de calor.

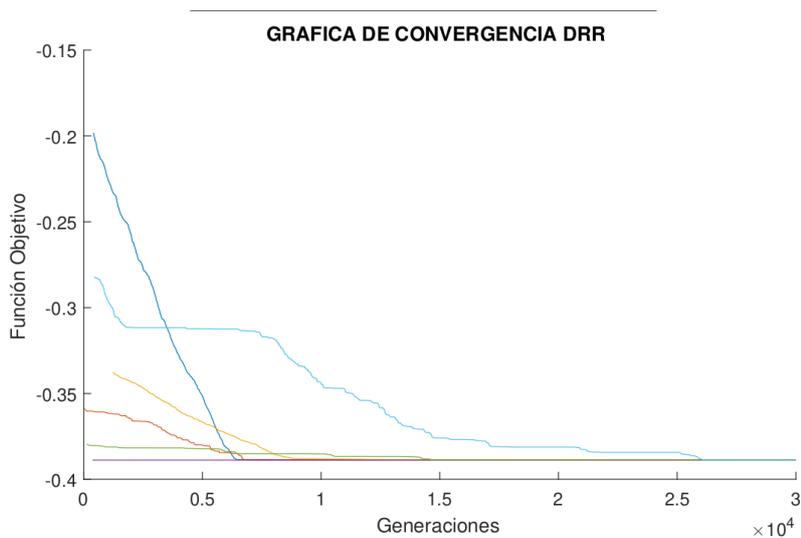


Fig. 8. Gráfica de Convergencia del diseño de red de reactores.

sus resultados con un máximo de cinco cifras significativas, estimando que con el valor de ϵ antes mencionado se cubre este rango de precisión.

Finalmente se enfatiza que la adecuada implementación de los métodos metaheurísticos brindan un buen rendimiento, eficiencia y por lo tanto un mejor enfoque en problemas de optimización, especialmente la variante de DE/RAND/1/BIN que resultó ser la variante mas eficiente en cuanto a exploración del

espacio de búsqueda del caso de estudio. También es importante mencionar que en cuanto a explotación, existe una particularidad de los algoritmos, es decir, DE/RAND/1/BIN es el mas eficiente para el Caso del intercambiador de calor, pero para el caso de la red de reactores, DE/RAND/1/EXP resulta ser el que mejor explota el espacio de soluciones factibles, puesto que encuentra un óptimo global mas rápido que las demas variantes de evolución diferencial.

Referencias

1. Storn, R., Price, K.: Differential Evolution, A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. Technical Report, International Computer Science Institute, Berkeley (1995)
2. Vela, C. R., Puente, J., Alonso, C. L., Varela, R.: Computación Evolutiva para Resolución de CSPs. Revista Iberoamericana de Inteligencia Artificial, 7, 20 (2003)
3. Babu, B. V., Angira, R.: Evolutionary computation for global optimization of Non-linear chemical engineering processes. Birla Institute of Technology and Science, 31, 5 (2003)
4. Babu, B. V., Angira, R.: Modified differential evolution (MDE) for optimization of non-linear chemical processes. Computers and Chemical Engineering, vol. 30, pp. 989–1002 (2006)
5. Adjiman, C.S., Androulakis, I.P., Floudas, C.A.: A Global Optimization Method, BB, for general twice differentiable constrained NLPs – II. Implementation and computational results, Computers and Chemical Engineering, vol. 22(9), pp. 1159–1179 (1998)
6. Ryoo, H.S., Sahinidis, B.P.: Global Optimization of Nonconvex NLPs and MINLPs with Application in Process Design. Computers and Chemical Engineering, 19, 551 (1995)
7. Floudas, C.A., Aggarwal, A., Ciric A.R. Global optimum search for nonconvex NLP and MINLP problems. Computers & Chemical Engineering, vol. 13, no. 10, pp. 1117–1132, Editor Pergamon (1998)
8. Floudas, C.A., Christodoulos, A., Pardalos P.M.: State of the Art in Global Optimization. Computational Methods and Applications. 654 p., Kluwer Academic Publishers (1996)