

Búsqueda de patrones basada en trayectorias extraídas de la respuesta de sistemas de segundo orden.

Pattern search based on trajectories extracted from second-order system responses.

Jesus Edgar Elizondo Nuñez¹
jesus.elizondo5693@alumnos.udg.mx

Carlos Guzman¹
carlos.guzman8813@alumnos.udg.mx

Elivier Reyes¹
elivier.reyes8810@alumnos.udg.mx

Hector Escobar¹
hector.11294@gmail.com

Alberto Luque¹
alberto.lchang@academicos.udg.mx

¹ Universidad de Guadalajara

Resumen: Recientemente, en la literatura se han introducido varios esquemas metaheurísticos nuevos. Aunque todos estos enfoques consideran fenómenos muy diferentes como metáforas, los patrones de búsqueda utilizados para explorar el espacio de búsqueda son muy similares. Por otro lado, los sistemas de segundo orden son modelos que presentan diferentes comportamientos temporales según el valor de sus parámetros. Tales comportamientos temporales pueden concebirse como patrones de búsqueda con múltiples comportamientos y configuraciones simples. En este artículo, se presentan un conjunto de nuevos patrones de búsqueda para explorar eficientemente el espacio de búsqueda. Estos emulan la respuesta de un sistema de segundo orden. El conjunto propuesto de patrones de búsqueda se ha integrado como una estrategia completa de búsqueda, llamada Algoritmo de Segundo Orden (SOA), para obtener la solución global de problemas de optimización complejos. Para analizar el rendimiento del esquema propuesto, se ha comparado en un conjunto de problemas representativos de optimización, que incluyen formulaciones de referencia multimodales, unimodales e híbridas. Los resultados numéricos demuestran que el método SOA propuesto exhibe un rendimiento notable en términos de precisión y altas tasas de convergencia.

Palabras clave: métodos metaheurísticos; patrones de búsqueda; sistemas de segundo orden; métodos evolutivos.

Summary:

Recently, several new metaheuristic frameworks have been introduced in the literature. Although these approaches rely on very different metaphors, the search patterns used to explore the search space are quite similar. On the other hand, second-order systems are models that exhibit various temporal behaviors depending on the values of their parameters. These temporal behaviors can be interpreted as search patterns with multiple behaviors and simple configurations. This paper presents a set of new search patterns designed to efficiently explore the search space, which emulate the response of a second-order system. The proposed set of search patterns is integrated into a complete search strategy called the **Second-Order Algorithm (SOA)** to find the global solution to complex optimization problems. To evaluate the performance of the proposed scheme, it has been compared across a set of representative optimization problems, including multimodal, unimodal, and hybrid benchmark formulations. Numerical results demonstrate that the proposed SOA method exhibits remarkable performance in terms of accuracy and high convergence rates.

Keywords: metaheuristic methods; search patterns; second-order systems; evolutionary methods.

1. Introducción

Los algoritmos metaheurísticos se refieren a esquemas de optimización genéricos que emulan el funcionamiento de diferentes procesos naturales o sociales. En los enfoques metaheurísticos, la estrategia de optimización se lleva a cabo mediante un conjunto de agentes de búsqueda. Cada agente mantiene una posible solución al problema de optimización y se genera inicialmente considerando una solución factible aleatoria. Una función objetivo determina la calidad de la solución de cada agente. Utilizando los valores de la función objetivo, en cada iteración, se modifica la posición de los agentes de búsqueda, empleando un conjunto de patrones de búsqueda que regulan sus movimientos dentro del espacio de búsqueda. Estos patrones de búsqueda son modelos abstractos inspirados en procesos naturales o sociales (Cuevas et al., 2020). Estos pasos se repiten hasta que se alcanza un criterio de parada. Los esquemas metaheurísticos han confirmado su supremacía en diversas aplicaciones del mundo real en circunstancias donde los métodos clásicos no pueden ser adoptados.

En esencia, no existe una clasificación clara de los métodos metaheurísticos. A pesar de esto, se han propuesto varias categorías que consideran diferentes criterios, como la fuente de inspiración, el tipo de operadores o la cooperación entre los agentes. En relación con la inspiración, los algoritmos metaheurísticos inspirados en la naturaleza se clasifican en tres categorías: basados en la evolución, basados en enjambres y basados en la física. Los enfoques basados en la evolución corresponden a las estrategias de búsqueda más consolidadas que utilizan elementos de evolución como operadores para producir patrones de búsqueda. En consecuencia, operaciones como la reproducción, la mutación, la recombinación y la selección se utilizan para generar patrones de búsqueda durante sus operaciones. Los ejemplos más representativos de técnicas basadas en la evolución incluyen las Estrategias Evolutivas (EE) (Beyer & Schwefel, 2002; Hansen, 2023; T, 1991), Algoritmos Genéticos (AG) (Tang et al., 1996), Differential Evolution (DE) (Storn & Price, 1997) y Self-Adaptive Differential Evolution (JADE) (Zhang & Sanderson, 2007). Las técnicas inspiradas en enjambres utilizan esquemas de comportamiento extraídos de la interacción colaborativa de diferentes animales o especies de insectos para producir una estrategia de búsqueda. Recientemente en la literatura se ha publicado un gran número de enfoques basados en enjambres. Entre los enfoques inspirados en enjambres más populares se encuentra el Algoritmo de Búsqueda de Cuervos (CSA) (Askarzadeh, 2016), Colonia Artificial de Abejas (ABC) (Karaboga et al., 2014), Algoritmo de Optimización por Enjambre de Partículas (PSO) (Kennedy & Eberhart, 1995; Marini & Walczak, 2015; Poli et al., 2007), Algoritmo de Luciérnagas (FA) (Yang, 2009), Cuckoo Search (CS) (Yang & Deb, 2009), Bat Algorithm (BA) (Yang, 2010), Gray Wolf Optimizer (GWO) (Mirjalili et al., 2014), Moth-flame optimization algorithm (MFO) (Mirjalili, 2015) por mencionar algunos. Los algoritmos metaheurísticos que consideran un esquema basado en la física utilizan modelos físicos simplificados para producir patrones de búsqueda para sus agentes. Algunos ejemplos de las técnicas basadas en la física más representativas incluyen el States of Matter Search (SMS) (Cuevas et al., 2014; Valdivia-Gonzalez et al., 2017), Simulated Annealing (SA) algorithm (Kirkpatrick et al., 1983; Rutenbar, 1989; Siddique & Adeli, 2016), Gravitational Search Algorithm (GSA) (Rashedi et al., 2009), Water Cycle Algorithm (WCA) (Eskandar et al., 2012), Big Bang-Big Crunch (BB-BC) (Erol & Eksin, 2006) y Electromagnetism-like Mechanism (EM) (Birbil & Fang, 2003). La Figura 1 muestra visualmente la taxonomía de la clasificación metaheurística. Aunque todos estos enfoques consideran fenómenos muy diferentes como metáforas, los patrones de búsqueda utilizados para explorar el espacio de búsqueda se basan exclusivamente en elementos espirales o modelos de

atracción (Kennedy & Eberhart, 1995; Marini & Walczak, 2015; Mirjalili et al., 2014; Poli et al., 2007; Yang, 2009, 2010). Bajo tales condiciones, el diseño de muchos métodos metaheurísticos se refiere a la configuración de un patrón de búsqueda reciclado que ha demostrado ser exitoso en enfoques anteriores para generar nuevos esquemas de optimización mediante una modificación marginal.

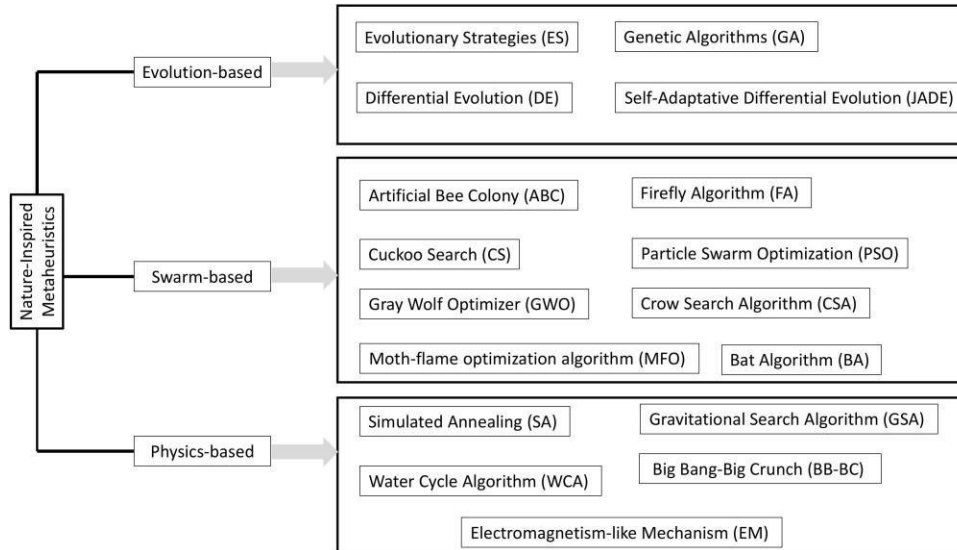


Figura 1. Taxonomía visual de los esquemas metaheurísticos inspirados en la naturaleza.

Por otro lado, el orden de una ecuación diferencial se refiere al grado más alto de derivada considerado en el modelo. Por lo tanto, un modelo cuya formulación de entrada-salida es una ecuación diferencial de segundo orden se conoce como un sistema de segundo orden (Zill, 2012). Uno de los elementos principales que hace que un modelo de segundo orden sea importante es su capacidad para presentar comportamientos muy diferentes, dependiendo de la configuración de sus parámetros. A través de sus distintos comportamientos, como oscilatorio, sub amortiguado o sobre amortiguado, un sistema de segundo orden puede mostrar respuestas temporales distintas (Haidekker, 2020). Estos comportamientos pueden observarse como trayectorias de búsqueda desde la perspectiva de los esquemas metaheurísticos. Por lo tanto, con sistemas de segundo orden, es posible generar movimientos oscilatorios dentro de una región determinada o construir patrones de búsqueda complejos alrededor de diferentes puntos o secciones del espacio de búsqueda.

En este artículo, se introduce un conjunto de nuevos patrones de búsqueda para explorar eficientemente el espacio de búsqueda. Estos patrones emulan la respuesta de un sistema de segundo orden. El conjunto propuesto de patrones de búsqueda se ha integrado como una estrategia completa de búsqueda, llamada Algoritmo de Segundo Orden (SOA), para obtener la solución global de problemas de optimización complejos. Para analizar el rendimiento del esquema propuesto, se ha comparado en un conjunto representativo de problemas de optimización, que incluyen formulaciones multimodales, unimodales e híbridas. Los resultados competitivos demuestran los resultados prometedores de los patrones de búsqueda propuestos.

Las principales contribuciones de esta investigación se pueden expresar de la siguiente manera:

1. Se introduce un nuevo algoritmo de optimización basado en la física, denominado

SOA. Utiliza patrones de búsqueda obtenidos a partir de la respuesta de sistemas de segundo orden.

2. Se proponen nuevos patrones de búsqueda como alternativa a los conocidos en la literatura.

3. Se evalúa la significancia estadística, la velocidad de convergencia y la proporción de explotación-exploración de SOA en comparación con otros algoritmos metaheurísticos populares.

4. SOA supera a otros algoritmos competidores en dos conjuntos de problemas de optimización.

El resto de este documento está estructurado de la siguiente manera: Se presenta una breve introducción de los sistemas de segundo orden en la sección 2; en la sección 3, se discuten los patrones de búsqueda más importantes en los métodos metaheurísticos; en la sección 4, se definen los patrones de búsqueda propuestos; en la sección 5, se describe la medición de la exploración-explotación; en la sección 6, se introduce el esquema propuesto; en la sección 7 se presentan los resultados numéricos; en la sección 8, se discuten las principales características del enfoque propuesto; en la sección 9, finalmente, se extraen las conclusiones.

2. Sistemas de Segundo Orden

Un modelo cuya formulación de entrada $R(s)$ - salida $C(s)$ es una función de transferencia de lazo cerrado de segundo orden se conoce como un sistema de segundo orden. Uno de los elementos principales que hace que un modelo de segundo orden sea importante es su capacidad para presentar comportamientos muy diferentes según la configuración de sus parámetros. Un modelo de segundo orden genérico se puede formular bajo la siguiente expresión (Zill, 2012):

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (1)$$

Donde ζ y ω_n representan la razón de amortiguamiento y ω_n la frecuencia natural, respectivamente, mientras que s simboliza el dominio de Laplace.

El comportamiento dinámico de un sistema se evalúa en términos de la respuesta temporal obtenida mediante una señal de paso unitario como entrada $R(s)$. El comportamiento dinámico se define como la forma en que el sistema reacciona, tratando de alcanzar el valor de uno a medida que evoluciona el tiempo. El comportamiento dinámico del sistema de segundo orden se describe en términos de ζ y ω_n (Haidekker, 2020). Suponiendo tales parámetros, el sistema de segundo orden presenta tres comportamientos diferentes: Subamortiguado ($0 < \zeta < 1$), críticamente amortiguado ($\zeta = 1$), y sobre amortiguado ($\zeta > 1$).

2.1. Comportamiento Subamortiguado ($0 < \zeta < 1$)

En este comportamiento, los polos (raíces del denominador) de la ecuación (1) son complejos

conjugados y se encuentran en la mitad izquierda del plano s . Bajo tales condiciones, la respuesta subamortiguada $C_u(s)$ el sistema en el dominio de Laplace se puede caracterizar de la siguiente manera:

$$C_u(s) = \frac{\omega_n^2}{s(s + \zeta\omega_n) + \omega_n^2(1 - \zeta^2)} \quad (2)$$

Aplicando operaciones de fracciones parciales y la transformada inversa de Laplace, se obtiene la respuesta temporal que describe el comportamiento subamortiguado $C_u(t)$ como se indica en la ecuación (3):

$$c_u(t) = 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1 - \zeta^2}} \sin(\omega_n \sqrt{1 - \zeta^2} + \tan^{-1}(\frac{\sqrt{1 - \zeta^2}}{\zeta})) \quad (3)$$

Si $\zeta = 0$, se presenta un caso especial en el cual la respuesta temporal del sistema es oscilatoria. La salida de estos comportamientos se visualiza en la Figura 2 para los casos de $\zeta = 0$, $\zeta = 0.2$, $\zeta = 0.5$ y $\zeta = 0.707$. En el comportamiento subamortiguado, la respuesta del sistema comienza con una alta aceleración. Por lo tanto, la respuesta produce un sobrepaso que supera el valor de uno. El tamaño del sobrepaso depende inversamente del valor de ζ .

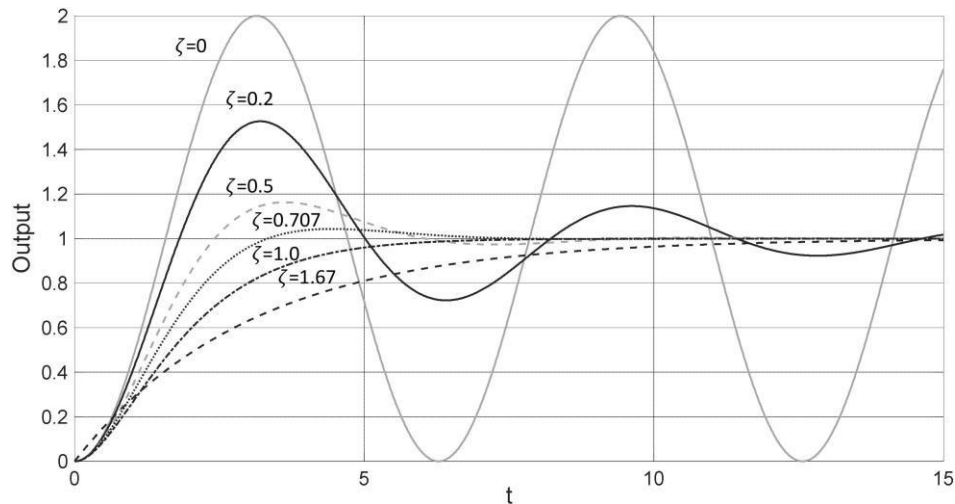


Figura 2. Respuestas temporales del sistema de segundo orden considerando sus diferentes comportamientos: Sub amortiguado ($0 < \zeta < 1$), críticamente amortiguado ($\zeta = 1$), y sobre amortiguado ($\zeta > 1$).

2.2. Comportamiento críticamente amortiguado ($\zeta = 1$)

En el comportamiento críticamente amortiguado, los dos polos de la función de transferencia de la ecuación (1) presentan números reales con el mismo valor. Por lo tanto, la respuesta del sistema críticamente amortiguado $C_c(s)$ en el dominio de Laplace puede ser descrita como:

$$C_c(s) = \frac{\omega_n^2}{s(s + \omega_n)^2} \quad (4)$$

Aplicando la transformada inversa de Laplace a (4), la respuesta temporal al comportamiento críticamente amortiguado $C_c(t)$ es representado por el modelo:

$$c_c(t) = 1 - e^{-\omega_n t}(1 + \omega_n t) \quad (5)$$

En este tipo de comportamiento, la respuesta del sistema presenta un patron similar a lo sistemas de primer orden. El cual, alcanza el valor de uno sin experimentar ningún excedente. La salida de este sistema esta representado visualmente en la Figura 2.

2.3. Comportamiento Sobrearmortiguado ($\zeta > 1$)

En este caso, los polos de la función de transferencia son numeros reales con con diferente valor. La respuesta de $C_o(s)$ en el dominio de Laplace es modelada como se indica en (6)

$$C_o(s) = \frac{\omega_n^2}{s(s + \zeta\omega_n + \omega_n\sqrt{\zeta^2 - 1})(s + \zeta\omega_n - \omega_n\sqrt{\zeta^2 - 1})} \quad (6)$$

Por su parte, la ecuación (7) representa la respuesta temporal del sistema sobrearmortiguado tras aplicar la transformada inversa de Laplace a (6)

$$c_o(t) = 1 - e^{-(\zeta - \sqrt{\zeta^2 - 1})\omega_n t} \quad (7)$$

En un comportamiento sobrearmortiguado, el sistema reacciona lentamente hasta alcanzar el valor de uno. La desaceleración de la respuesta depende de ζ , donde un mayor valor representa una repuesta más lenta. La respuesta de este comportamiento e representada en la figura 2 con un valor de $\zeta=1.67$.

3. Patrones de búsqueda en Metaheurísticos

Un patrón de búsqueda es un conjunto de movimientos producidos por una regla o modelo con el fin de examinar soluciones prometedoras en el espacio de búsqueda. La generación de patrones de búsqueda eficientes para la correcta exploración del valor óptimo dentro del espacio de búsqueda puede ser complicada, especialmente cuando existen múltiples óptimos locales. Recientemente, varios esquemas metaheurísticos han sido introducidos en la literatura. Aunque todos estos enfoques consideran fenómenos muy diferentes como metáforas, los patrones de búsqueda utilizados para explorar el espacio de búsqueda son muy similares.

La exploración y la explotación corresponden a las características más importantes de un patrón de búsqueda. La exploración se refiere a la capacidad de un patrón de búsqueda para examinar un conjunto de soluciones distribuidas en áreas distintas del espacio de búsqueda. Por otro lado, la explotación representa la capacidad del patrón de búsqueda para mejorar la precisión y calidad de las soluciones existentes evaluando la localidad de dichas soluciones. La combinación de ambos mecanismos en un patrón de búsqueda es crucial para lograr el éxito al resolver un problema de optimización particular.

Los problemas de optimización, desde la perspectiva Metaheurística, inicializan una población $P^k(\{x_1^k, \dots, x_N^k\})$ de N soluciones candidatas (individuos) que evolucionan tras cada generación, desde un punto inicial $k = 1$, hasta un número determinado $k = Maxgen$. Cada individuo

$x_i^k (i \in [1, \dots, N])$ perteneciente a la población, corresponde a un elemento dimensional-d $\{x_{i,1}^k, \dots, x_{i,d}^k\}$, el cual simboliza las variables de decisión involucradas en el problema de optimización.

En cada generación, el patrón de búsqueda actúa sobre cada individuo de la población P^k , para producir una nueva población P^{k+1} . La calidad de cada individuo x_i^k se evalúa en con respecto a la función objetivo $J(x_i^k)$, cuyo valor representa que tan apta es la solución x_i^k . Conforme evoluciona el proceso metaheurístico, se conservan los mejores individuos $\mathbf{b} \{b_1, \dots, b_d\}$, dado que estos representan las mejores soluciones encontradas hasta el momento.

En general, los patrones de búsqueda actúan sobre cada individuo x_i^k , usando como referencia los mejores elementos almacenados en $\mathbf{b}$. Posteriormente, y, dependiendo del modelo metaheurístico empleado, un conjunto de reglas de movimiento modifica la posición de x_i^k hasta que la posición $\mathbf{b}$ ha sido alcanzada. La idea detrás de este mecanismo es examinar las soluciones que se encuentran en la trayectoria de x_i^k a $\mathbf{b}$, con el objetivo de encontrar mejores soluciones que las pertenecientes a al actual valor de $\mathbf{b}$. Los patrones de búsqueda para generar las trayectorias de x_i^k a $\mathbf{b}$ difieren del modelo empleado.

Dos de los modelos de búsqueda más populares son las trayectorias de “atracción” y “espiral”. Los modelos de atracción generan movimientos de atracción x_i^k a $\mathbf{b}$. Los modelos de atracción son empleados ampliamente por una extensa cantidad de métodos metaheurísticos tales como PSO (Kennedy & Eberhart, 1995; Marini & Walczak, 2015; Poli et al., 2007), FA (Yang, 2009), CS (Yang & Deb, 2009), BA (Yang, 2010), GSA (Rashedi et al., 2009), EM (Birbil & Fang, 2003), y DE (Storn & Price, 1997). Por su parte, los modelos de espiral producen una trayectoria en espiral alrededor de las mejores soluciones pertenecientes a $\mathbf{b}$. Algunos de los modelos que emplean una trayectoria en espiral son WOA y GWO. La Figura 3, muestra las trayectorias que se generan con los modelos de atracción y de espiral.

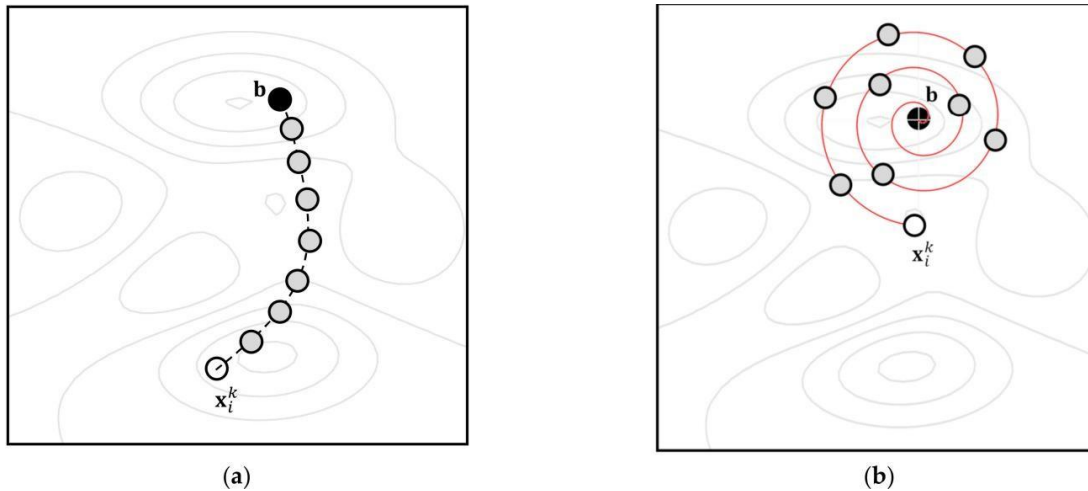


Figura 3. Trayectorias producidas por a) atracción, y b) espiral.

4. Patrón de búsqueda propuesto

En este artículo, se presentan un conjunto de nuevos patrones de búsqueda para explorar el espacio de búsqueda de manera eficiente. Estos patrones emulan la respuesta de un sistema de segundo

orden. El conjunto de patrones de búsqueda propuesto se emplea como una estrategia de búsqueda completa con la finalidad de obtener la solución global de problemas de optimización complejos. Dado que el esquema propuesto se basa en la respuesta de sistemas de segundo orden, puede considerarse como un algoritmo basado en la física. En este enfoque, la respuesta temporal del sistema de segundo orden se utiliza para generar la trayectoria desde la posición de $x_{i,1}^k, \dots, x_{i,d}^k$ hasta $b = \{b_1, \dots, b_d\}$. Con el uso de estos modelos, es posible generar trayectorias más complejas que permiten una mejor evaluación del espacio de búsqueda. Ante tales condiciones, se consideran las tres respuestas de un sistema de segundo orden para producir patrones de búsqueda distintos. Estos son el subamortiguado, críticamente amortiguado y sobreamortiguado, modelados por las expresiones:

$$x_{i,j}^k = \frac{1}{h} \left(1 - \frac{e^{-\zeta \omega_n k}}{\sqrt{1 - \zeta^2}} \sin \left(\omega_n \sqrt{1 - \zeta^2} k + \tan^{-1} \frac{\sqrt{1 - \zeta^2}}{\zeta} \right) \right) (b_j - x_{i,j}^k) \quad (8)$$

$$x_{i,j}^k = (1 - e^{-\omega_n t} (1 + \omega_n t)) (b_j - x_{i,j}^k) \quad (9)$$

$$x_{i,j}^k = (1 - e^{-(\zeta - \sqrt{\zeta^2 - 1}) \omega_n k}) (b_j - x_{i,j}^k) \quad (10)$$

Donde $i \in [1, N]$ corresponden a los agentes de búsqueda, mientras que $j \in [1, N]$ simboliza las variables de decisión o dimensión del problema. Dado que el comportamiento de cada patrón de búsqueda depende del valor de ζ , es fácil combinar elementos para producir trayectorias interesantes. En la Figura 4 se muestran algunos ejemplos de trayectorias producidas por diferentes valores de ζ .

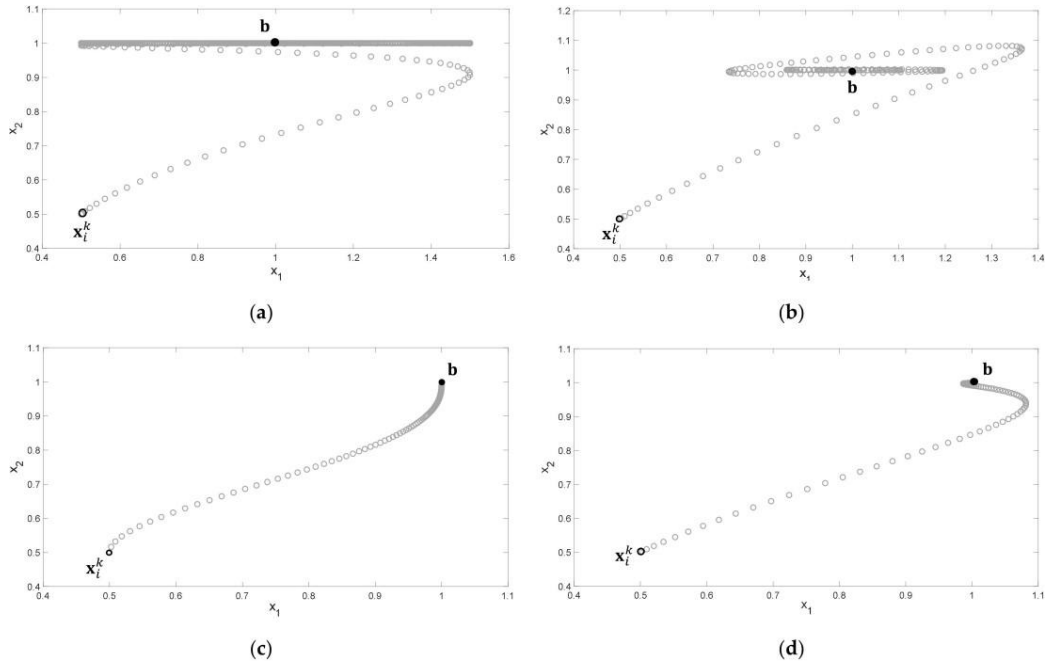


Figura 4. Algunos ejemplos de las trayectorias producidas usando diferentes valores de ζ , a) $x_{i,1}^k \leftarrow \zeta = 0$, y $x_{i,2}^k \leftarrow \zeta = 1$, b) $x_{i,1}^k \leftarrow \zeta = 0.1$, y $x_{i,2}^k \leftarrow \zeta = 0.5$, c) $x_{i,1}^k \leftarrow \zeta = 1$, y $x_{i,2}^k \leftarrow \zeta = 1.67$, d) $x_{i,1}^k \leftarrow \zeta = 0.5$, y $x_{i,2}^k \leftarrow \zeta = 1$

En dicha Figura, se asume un caso de dos dimensiones ($d=2$), donde la posición inicial de x_t^k es (0.5, 0.5), y la posición final o mejor posición es (1,1). La Figura 4a, muestra el escenario de $x_{i,1}^k \leftarrow \zeta = 0$ y $x_{i,2}^k \leftarrow \zeta = 1$. Por su parte, la Figura 4b muestra el caso de $x_{i,1}^k \leftarrow \zeta = 0.1$, y $x_{i,2}^k \leftarrow \zeta = 0.5$. Mientras que la Figura 4c presenta el caso $x_{i,1}^k \leftarrow \zeta = 1$, y $x_{i,2}^k \leftarrow \zeta = 1.67$. Finalmente, la Figura 4d muestra el escenario $x_{i,1}^k \leftarrow \zeta = 0.5$, y $x_{i,2}^k \leftarrow \zeta = 1$. De estas figuras, se puede apreciar que las respuestas de segundo orden pueden producir una gran cantidad de trayectorias complejas, que incluye a gran parte de otros patrones de búsqueda encontrados en la literatura. En todos estos casos (a)-(d), el valor asignado a ω_n ha sido 1.

5. Equilibrio entre exploración y explotación

Los métodos metaheurísticos emplean un conjunto de agentes de búsqueda para examinar el espacio de búsqueda con el objetivo de identificar una solución satisfactoria para una formulación de optimización. En los esquemas metaheurísticos, los agentes de búsqueda que presentan los mejores valores de fitness tienden a regular el proceso de búsqueda, produciendo una atracción hacia ellos. En estas condiciones, a medida que evoluciona el proceso de optimización, la distancia entre los individuos disminuye, al tiempo que se acentúa el efecto de explotación. Por otro lado, cuando la distancia entre individuos aumenta, las características del proceso de exploración son más evidentes.

Para calcular la distancia relativa entre los individuos (aumento y disminución), se utiliza un indicador de diversidad conocido como índice de diversidad dimensional (Morales-Castañeda et al., 2020). Según este enfoque, la diversidad se formula de la siguiente manera:

$$Div_j = \frac{1}{N} \sum_{i=1}^N |median(x^j) - x_{i,j}| \quad Div = \frac{1}{d} \sum_{j=1}^d Div_j \quad (11)$$

donde $median(x^j)$ simboliza la mediana de la dimensión j de todos los agentes de búsqueda. $x_{i,j}$ representa la decisión variable j del individuo i . N es el número de individuos en la población P^k mientras que d corresponde al número de dimensiones de la formulación de optimización. La diversidad Div_j (de la j -ésima dimensión) evalúa la distancia relativa entre la variable j de cada individuo y su valor mediano. La diversidad completa Div (de toda la población) corresponde a la diversidad promedio en cada dimensión. Ambos elementos Div_j y Div se calculan en cada iteración.

Habiendo evaluado los valores de diversidad, el nivel de exploración y explotación se puede calcular como el porcentaje del tiempo que una estrategia de búsqueda invierte explorando o explotando en términos de sus valores de diversidad. Estos porcentajes se calculan en cada iteración mediante los siguientes modelos,

$$XLP\% = \left(\frac{Div}{Div_{max}} \right) \times 100 \quad XPT\% = \left(\frac{|Div - Div_{max}|}{Div_{max}} \right) \times 100 \quad (12)$$

Donde Div_{max} simboliza el valor máximo de diversidad obtenido durante el proceso de optimización. El porcentaje de exploración $XPL\%$ corresponde al tamaño de la exploración como la tasa entre Div y Div_{max} . Por otro lado, el porcentaje de explotación $XPT\%$ simboliza el nivel de explotación. $XPT\%$

se calcula como el porcentaje complementario a $XPL\%$ ya que la diferencia entre Div_{max} y Div se genera debido a la concentración de individuos.

6. Algoritmo metaheurístico propuesto

El conjunto de patrones de búsqueda basados en los sistemas de segundo orden se ha integrado como una estrategia de búsqueda completa para obtener la solución global de problemas complejos de optimización. En esta sección se describe el método metaheurístico completo, llamado Algoritmo de Segundo Orden (SOA).

El esquema considera cuatro etapas diferentes: (A) Inicialización, (B) generación de trayectoria, (C) reinicio de elementos defectuosos y (D) evitar mecanismo de convergencia prematura. Los pasos (B)–(D) se ejecutan secuencialmente hasta que se alcanza un criterio de parada. La Figura 5 muestra el diagrama de flujo del método metaheurístico completo.

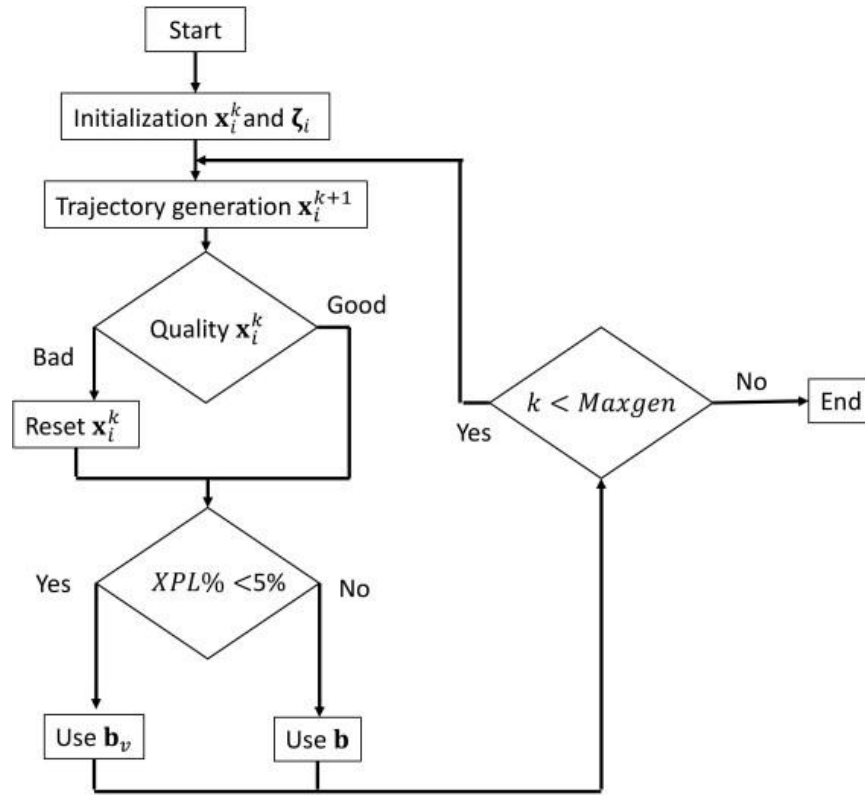


Figura 5. Diagrama de flujo del método metaheurístico propuesto basado en la respuesta de sistemas de segundo orden.

6.1. Inicialización

En la primera iteración $k = 0$, se produce aleatoriamente una población P^0 de N agentes $\{x_1^0, \dots, x_N^0\}$ considerando la siguiente ecuación:

$$x_{i,j}^0 = rand \cdot (b_j^{high} - b_j^{low}) + b_j^{low} = 1, 2, \dots, N; j = 1, \dots, d \quad (13)$$

Donde b_j^{high} y b_j^{low} son los límites de la variable de decisión j y $rand$ es un número aleatorio distribuido uniformemente entre $[0,1]$.

A cada individuo x_i de la población se le asigna un vector $\zeta_i = \{\zeta_{i,1}, \dots, \zeta_{i,d}\}$ cuyos elementos $\zeta_{i,j}$ determinan el comportamiento de la trayectoria de cada j -ésima dimensión. Inicialmente, cada elemento $\zeta_{i,j}$ se establece en un valor aleatorio entre $[0,2]$. Bajo este intervalo, todos los comportamientos de segundo orden son posibles: subamortiguado ($0 < \zeta < 1$), críticamente amortiguado ($\zeta = 1$) y sobreamortiguado ($\zeta > 1$).

6.2. Generación de Trayectoria

Una vez inicializada la población, se obtiene el mejor elemento de la población b . Luego, la nueva posición x_i^{k+1} de cada agente x_i^k se calcula como una trayectoria generada por un sistema de segundo orden. Una vez determinadas todas las nuevas posiciones en la población P^k , también se define el mejor elemento b .

6.3. Restablecimiento de Elementos Defectuosos

A cada agente x_i^k se le permite moverse en su propia trayectoria durante diez iteraciones. Después de diez iteraciones, si el agente de búsqueda x_i^k mantiene el peor desempeño en términos de la función de aptitud, se reinicializa tanto en la posición como en su vector ζ_i . En tales condiciones, el agente de búsqueda estará en otra posición y con la capacidad de realizar otro tipo de comportamiento de trayectoria.

6.4. Evadir el Mecanismo de Convergencia Prematura

Si el porcentaje de exploración $XPL\%$ es inferior al 5%, el mejor valor b se reemplaza por el mejor valor virtual b_v . El elemento b_v se calcula como el valor promedio de los cinco mejores individuos de la población. La idea detrás de este mecanismo es identificar una nueva posición para generar diferentes trayectorias que eviten que el proceso de búsqueda quede atrapado en un óptimo local.

7. Resultados experimentales

Para evaluar los resultados del algoritmo SOA propuesto, se ha llevado a cabo un conjunto de experimentos. Dichos resultados se han comparado con los obtenidos por la Colonia Artificial de Abejas (ABC) (Karaboga et al., 2014), la Estrategia de Evolución de Adaptación de la Matriz de Covarianza (CMAES) (Hansen, 2023), el Algoritmo de Búsqueda de Cuervos (CSA) (Askarzadeh, 2016), la Evolución Diferencial (DE) (Storn & Price, 1997), el Algoritmo de Optimización de Polilla (MFO) (Mirjalili, 2015) y la Optimización de Enjambre de Partículas (PSO) (Kennedy & Eberhart, 1995), que se consideran los esquemas metaheurísticos más populares en muchos estudios de optimización (Boussaïd et al., 2013). Para la comparación, todos los métodos se han configurado de acuerdo con sus parámetros publicados.

En nuestro análisis, el tamaño de la población N se ha fijado en 50 agentes de búsqueda. El número máximo de iteraciones (Maxgen) para todas las funciones se ha fijado en 1000. Este criterio de parada se ha decidido para mantener la compatibilidad con trabajos similares publicados en la literatura (Han et al., 2014; Meng & Pan, 2016). Para evaluar los resultados, se consideran tres

indicadores diferentes: La media de las mejores soluciones hasta el momento (AB), la mediana de las mejores soluciones hasta el momento (MB) y la desviación estándar (SD) de las mejores soluciones hasta el momento. En el análisis, cada problema de optimización se resuelve 30 veces con cada algoritmo, por lo tanto se obtienen 30 resultados. A partir de todos estos valores, el valor medio de todas las mejores soluciones encontradas representa la solución media Best-so-far (AB).

Asimismo, se calcula la mediana de los 30 resultados para generar MB y se estima la desviación estándar de los 30 datos para obtener SD de las mejores soluciones hasta el momento. Los indicadores AB y MB corresponden a la precisión de las soluciones, mientras que SD a su dispersión y, por tanto, a la robustez del algoritmo.

7.1 Funciones multimodales

En esta subsección, el enfoque SOA se evalúa considerando 12 funciones unimodales ($f_1(x)$ - $f_{12}(x)$) detalladas en la Tabla 1 del Apéndice A. Este tipo de funciones presentan superficies de optimización que implican múltiples óptimos locales. Por esta razón, estas funciones presentan más complicaciones en su solución. En este análisis, se examina el rendimiento del método SOA en comparación con ABC, CMAES, CSA, DE, MFO y PSO en términos de las funciones multimodales. Los experimentos se llevan a cabo con funciones objetivo que operan en 30 dimensiones ($d = 30$).

		ABC	DE	CMAES	CSA	PSO	MFO	SOA
$f_1(x)$	AB	8.9132622	0.7932535	2.8976×10^{-19}	55.918504	0.2012803	27.983271	0.1119774
	MD	8.4392750	0.7993166	2.4779×10^{-19}	57.038263	4.1459×10^{-23}	25.365199	1.0714×10^{-10}
	SD	2.6748059	0.1378538	1.5343×10^{-19}	6.2032578	1.1022968	12.283254	0.2272439
$f_2(x)$	AB	2	2	2	1,897,783.3	27.4	2	2
	MD	2	2	2	35,691.155	2	2	2
	SD	9.9512×10^{-12}	0	0	9,636,632.1	113.43495	0	0
$f_3(x)$	AB	2	2	2	3,620,834.4	34.723128	2	2
	MD	2	2	2	676,981.46	9	2	2
	SD	2.3308×10^{-11}	0	0	7,265,352.7	113.36672	0	0
$f_4(x)$	AB	0.1371551	0.002	1.7942×10^{-6}	0.0862919	2.2285×10^{-8}	5.5194×10^{-10}	1.164×10^{-11}
	MD	0.1349076	0.01	0	0.0892256	0	0	7.7118×10^{-12}
	SD	0.0399861	0.123	5.4833×10^{-6}	0.0213332	1.2206×10^{-7}	3.0231×10^{-9}	1.0995×10^{-11}
$f_5(x)$	AB	13,781,291	1,331,987.7	22,307.195	44,274,761	82.539625	85.756149	71.964984
	MD	13,876,263	1,365,502.1	72.377516	46,153,728	81.698488	85.665615	72.362277
	SD	3,237,147.7	306,385.34	50,923.637	10,118,180	7.1916726	3.2857088	0.9929591
$f_6(x)$	AB	1.152×10^{85}	5.850×10^{81}	1.812×10^{83}	6.429×10^{83}	1.397×10^{81}	3.0883×10^{81}	1.0051×10^{81}
	MD	4.622×10^{84}	3.405×10^{81}	7.928×10^{82}	2.939×10^{83}	5.977×10^{80}	7.9607×10^{80}	4.901×10^{80}
	SD	1.685×10^{85}	7.33×10^{81}	3.047×10^{83}	7.551×10^{83}	2.197×10^{81}	5.6651×10^{81}	1.9457×10^{81}
$f_7(x)$	AB	30.033333	30	30	58.766666	30	33.633333	30
	MD	30	30	30	59	30	30	30
	SD	0.18257419	0	0	1.77498583	0	5.4550409	0
$f_8(x)$	AB	9.2	8.0666666	1.0666666	19,543.266	0.0333333	2000.0333	0
	MD	9	8	0	19,797	0	0	0
	SD	2.5784250	1.9464084	3.1941037	2077.7459	0.1825741	4842.3277	0
$f_9(x)$	AB	-745.05202	-1125.4815	-1127.8626	-725.09353	-1071.7869	-1031.2617	-1146.3478
	MD	-743.4462	-1174.9722	-1132.5748	-719.10777	-1068.9596	-1033.6178	-1145.2467
	SD	25.137593	78.706	25.809999	25.815652	34.055847	34.244188	10.928362
$f_{10}(x)$	AB	110,282.54	665,278.86	-4930	1,170,939.0	45,556.260	222,833.73	-501.79356
	MD	96,461.061	673,449.49	-4930	1,126,234.2	5076.8152	71,582.051	-332.82466
	SD	44,933.417	129,147.27	3.7318×10^{-9}	159,175.64	75,990.880	305,159.37	663.92006
$f_{11}(x)$	AB	-18.26109	-26.056561	-29.6576	-16.504756	-28.367666	-28.863589	-30
	MD	-18.131984	-26.092349	-29.9286	-16.183447	-28.14029	-29.070145	-30
	SD	1.6366873	0.5428906	0.466	1.16917142	1.5408536	1.2837415	0
$f_{12}(x)$	AB	1502.3129	369.60375	786.36819	519.17242	196.95838	261.52332	11.905761
	MD	1457.6865	368.82916	778.72369	465.93238	213.00730	252.76747	0.3841574
	SD	420.70611	35.389136	215.93893	228.69860	86.542862	106.52353	29.544101

Tabla 1. Resultados de minimización de funciones de referencia multimodales.

En la Tabla 1 se muestran los mejores resultados promediados (AB) considerando 30 experimentos independientes. También se presentan los valores medios (DM) y las desviaciones estándar (DE).

En la Tabla 1 se observa que el esquema SOA propuesto obtiene un mejor rendimiento que los algoritmos ABC, CMAES, CSA, DE, MFO y PSO en las funciones $f_1(x)$, $f_4(x)$, $f_5(x)$, $f_6(x)$, $f_8(x)$, $f_9(x)$, $f_{10}(x)$, $f_{11}(x)$ y $f_{12}(x)$. Sin embargo, los resultados de SOA son similares a los obtenidos por DE, CMAES y MFO en las funciones $f_2(x)$, $f_3(x)$ y $f_7(x)$.

8. Conclusiones

Un patrón de búsqueda es un conjunto de movimientos producidos por una regla o modelo, con el fin de examinar soluciones prometedoras del espacio de búsqueda. En este artículo, se introduce un conjunto de nuevos patrones de búsqueda para explorar el espacio de búsqueda de forma eficiente. Emulan la respuesta de un sistema de segundo orden. En estas condiciones, se consideran tres respuestas diferentes de un sistema de segundo orden para producir tres patrones de búsqueda distintos, como subamortiguado, críticamente amortiguado y sobreamortiguado. Estos conjuntos de patrones de búsqueda propuestos se han integrado como una estrategia de búsqueda completa, denominada Algoritmo de Segundo Orden (SOA), para obtener la solución global de problemas de optimización complejos.

La forma de los patrones de búsqueda permite equilibrar las capacidades de exploración y explotación atravesando eficientemente el espacio de búsqueda y evitando las regiones subóptimas. La eficacia de la SOA propuesta se ha evaluado mediante 20 funciones de referencia estándar. Los resultados demuestran la eficacia de los patrones de búsqueda generados en las problemáticas más complejas.

Para comparar el rendimiento del esquema SOA, también se han probado en el mismo entorno experimental muchas otras técnicas de optimización populares, como la colonia de abejas artificial (ABC), la estrategia de evolución de adaptación de la matriz de covarianza (CMAES), el algoritmo de búsqueda de cuervos (CSA), la evolución diferencial (DE), el algoritmo de optimización de llama de polilla (MFO) y la optimización de enjambre de partículas (PSO). Las futuras líneas de investigación incluyen temas como las capacidades multiobjetivo, la incorporación de mapas caóticos y la inclusión de procesos de aceleración para resolver otros problemas de optimización a escala real.

10. Referencias

- Askarzadeh, A. (2016). A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm. *Computers & Structures*, 169, 1–12. <https://doi.org/10.1016/j.compstruc.2016.03.001>
- Beyer, H.-G., & Schwefel, H.-P. (2002). Evolution strategies – A comprehensive introduction. *Natural Computing*, 1(1), 3–52. <https://doi.org/10.1023/A:1015059928466>
- Birbil, Ş. İ., & Fang, S.-C. (2003). An Electromagnetism-like Mechanism for Global Optimization. *Journal of Global Optimization*, 25(3), 263–282. <https://doi.org/10.1023/A:1022452626305>

- Boussaïd, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. *Information Sciences*, 237, 82–117. <https://doi.org/10.1016/j.ins.2013.02.041>
- Cuevas, E., Echavarría, A., & Ramírez-Ortegón, M. A. (2014). An optimization algorithm inspired by the States of Matter that improves the balance between exploration and exploitation. *Applied Intelligence*, 40(2), 256–272. <https://doi.org/10.1007/s10489-013-0458-0>
- Cuevas, E., Gálvez, J., Avila, K., Toski, M., & Rafe, V. (2020). A new metaheuristic approach based on agent systems principles. *Journal of Computational Science*, 47, 101244. <https://doi.org/10.1016/j.jocs.2020.101244>
- Erol, O. K., & Eksin, I. (2006). A new optimization method: Big Bang–Big Crunch. *Advances in Engineering Software*, 37(2), 106–111. <https://doi.org/10.1016/j.advengsoft.2005.04.005>
- Eskandar, H., Sadollah, A., Bahreininejad, A., & Hamdi, M. (2012). Water cycle algorithm – A novel metaheuristic optimization method for solving constrained engineering optimization problems. *Computers & Structures*, 110–111, 151–166. <https://doi.org/10.1016/j.compstruc.2012.07.010>
- Haidekker, M. A. (2020). *Linear Feedback Controls: The Essentials*. Elsevier.
- Han, M., Liu, C., & Xing, J. (2014). An evolutionary membrane algorithm for global numerical optimization problems. *Information Sciences*, 276, 219–241. <https://doi.org/10.1016/j.ins.2014.02.057>
- Hansen, N. (2023). The CMA Evolution Strategy: A Tutorial (arXiv:1604.00772). arXiv. <https://doi.org/10.48550/arXiv.1604.00772>
- Karaboga, D., Gorkemli, B., Ozturk, C., & Karaboga, N. (2014). A comprehensive survey: Artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*, 42(1), 21–57. <https://doi.org/10.1007/s10462-012-9328-0>
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, 4, 1942–1948 vol.4. <https://doi.org/10.1109/ICNN.1995.488968>
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598), 671–680. <https://doi.org/10.1126/science.220.4598.671>
- Marini, F., & Walczak, B. (2015). Particle swarm optimization (PSO). A tutorial. *Chemometrics and Intelligent Laboratory Systems*, 149, 153–165. <https://doi.org/10.1016/j.chemolab.2015.08.020>
- Meng, Z., & Pan, J.-S. (2016). Monkey King Evolution: A new memetic evolutionary algorithm and its application in vehicle fuel consumption optimization. *Knowledge-Based Systems*, 97, 144–157. <https://doi.org/10.1016/j.knosys.2016.01.009>
- Mirjalili, S. (2015). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm.

- Knowledge-Based Systems, 89, 228–249. <https://doi.org/10.1016/j.knosys.2015.07.006>
- Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey Wolf Optimizer. *Advances in Engineering Software*, 69, 46–61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
- Morales-Castañeda, B., Zaldívar, D., Cuevas, E., Fausto, F., & Rodríguez, A. (2020). A better balance in metaheuristic algorithms: Does it exist? *Swarm and Evolutionary Computation*, 54, 100671. <https://doi.org/10.1016/j.swevo.2020.100671>
- Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. *Swarm Intelligence*, 1(1), 33–57. <https://doi.org/10.1007/s11721-007-0002-0>
- Rashedi, E., Nezamabadi-pour, H., & Saryazdi, S. (2009). GSA: A Gravitational Search Algorithm. *Information Sciences*, 179(13), 2232–2248. <https://doi.org/10.1016/j.ins.2009.03.004>
- Rutenbar, R. A. (1989). Simulated annealing algorithms: An overview. *IEEE Circuits and Devices Magazine*, 5(1), 19–26. <https://doi.org/10.1109/101.17235>
- Siddique, N., & Adeli, H. (2016). Simulated Annealing, Its Variants and Engineering Applications. *International Journal on Artificial Intelligence Tools*, 25(06), 1630001. <https://doi.org/10.1142/S0218213016300015>
- Storn, R., & Price, K. (1997). Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11(4), 341–359. <https://doi.org/10.1023/A:1008202821328>
- T, B. (1991). A Survey of Evolution Strategies. *Proc. of Fourth Internal. Conf. on Genetic Algorithms*. <https://cir.nii.ac.jp/crid/1573950398979648512>
- Tang, K. S., Man, K. F., Kwong, S., & He, Q. (1996). Genetic algorithms and their applications. *IEEE Signal Processing Magazine*, 13(6), 22–37. <https://doi.org/10.1109/79.543973>
- Valdivia-Gonzalez, A., Zaldívar, D., Fausto, F., Camarena, O., Cuevas, E., & Perez-Cisneros, M. (2017). A States of Matter Search-Based Approach for Solving the Problem of Intelligent Power Allocation in Plug-in Hybrid Electric Vehicles. *Energies*, 10(1), Article 1. <https://doi.org/10.3390/en10010092>
- Yang, X.-S. (2009). Firefly Algorithms for Multimodal Optimization. In O. Watanabe & T. Zeugmann (Eds.), *Stochastic Algorithms: Foundations and Applications* (pp. 169–178). Springer. https://doi.org/10.1007/978-3-642-04944-6_14
- Yang, X.-S. (2010). A New Metaheuristic Bat-Inspired Algorithm. In J. R. González, D. A. Pelta, C. Cruz, G. Terrazas, & N. Krasnogor (Eds.), *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)* (pp. 65–74). Springer. https://doi.org/10.1007/978-3-642-12538-6_6
- Yang, X.-S., & Deb, S. (2009). Cuckoo Search via Lévy flights. *2009 World Congress on Nature &*

Biologically Inspired Computing (NaBIC), 210–214. <https://doi.org/10.1109/NABIC.2009.5393690>

Zhang, J., & Sanderson, A. C. (2007). JADE: Self-adaptive differential evolution with fast and reliable convergence performance. 2007 IEEE Congress on Evolutionary Computation, 2251–2258. <https://doi.org/10.1109/CEC.2007.4424751>

Zill, D. G. (2012). A First Course in Differential Equations with Modeling Applications. Cengage Learning.



Esta obra está bajo una licencia de Creative Commons
Reconocimiento-NoComercial-CompartirIgual 2.5 México.