COMPARATIVA ENTRE ALGORITMOS EVOLUTIVOS EN PROBLEMAS DE MINIMIZACIÓN

ISC. Rául Fernando Galván Correa
raulgalvan92@outlook.com
Dr. Mauricio Olguín Carbajal
molguin@hotmail.com
Dr. Juan Carlos Herrera Lozada
jlozada@ipn.mx
Instituto Politécnico Nacional
Centro de Innovación y Desarrollo Tecnológico en Cómputo

Abstract

Se realizará una comparación entre 3 algoritmos evolutivos (Estrategias evolutivas, Evolución diferencial y Optimización por cúmulo de partículas-PSO) para observar su comportamiento en distintas funciones de prueba, obteniendo resultados que permitirán realizar una aproximación de cuáles algoritmos se desempeñan mejor en ciertos problemas de optimización.

1. Introducción

Los algoritmos evolutivos (también conocidos como bio-inspirados) son algoritmos heurísticos que están basados en el principio de la evolución a través de la supervivencia del más apto. El uso principal de este tipo de algoritmos es resolver problemas complejos de búsqueda, principalmente en problemas de optimización. En la mayoría de los trabajos de investigación se utilizan los algoritmos evolutivos para la resolución y optimización en problemas de la vida real a través de una función de adaptación (fitness), los cuales tienen restricciones al determinar el modelo matemático, adaptando los algoritmos para el correcto manejo de las restricciones. Entre los algoritmos evolutivos más conocidos se incluyen los algoritmos genéticos, estrategias evolutivas y programación genética. En conjunto todas las técnicas se conocen con el nombre de cómputo evolutivo. [1]

Estos algoritmos trabajan con una población de individuos P(t)={x1,x2,x3,...xn} para la iteración t, donde cada uno de los individuos x representa un punto de búsqueda en el espacio de soluciones de un problema en específico. Posteriormente el desempeño de un individuo xi se evalua en la función de adaptación (fitness). En los algoritmos evolutivos la población inicial evoluciona hacia mejores regiones del espacio de búsqueda a través de distintos procesos probabilísticos: 1) Elitismo: selección de los individuos más adaptados en la población, es decir los que obtienen mejores valores de minimización en la función de adaptación. 2) Modificación utilizando recombinación y/o mutación de individuos seleccionados. [2]

Las diferencias principales de los algoritmos evolutivos contra los tradicionales de búsqueda exhaustiva, aleatorios entre otros, son los siguientes: a) codificación de parámetros, b) búsquedas en paralelo con una población, c) Uso de una función de fitness sin necesidad de utilizar derivadas, d) reglas de transición probabilísticas entre una iteración y otra.

Para que los algoritmos evolutivos encuentren óptimos locales, utilizan básicamente dos técnicas : 1) Explorar áreas desconocidas en el espacio de búsqueda utilizando procesos y datos aleatorios para incrementar el espacio de exploración, 2) Explotar el conocimiento de los puntos previamente obtenidos.

En los últimos años se han propuesto nuevas heurísticas como Evolución Diferencial (ED) y Particle Swarm Optimization (PSO) para poder resolver problemas de optimización con restricciones. El objetivo del presente trabajo es analizar el comportamiento de diferentes algoritmos (Estrategia evolutiva (EE), Evolución Diferencial (ED), Optimización por cúmulo de partículas (PSO)) y determinar el desempeño de cada uno de los algoritmos con las mismas características del problema a resolver.[3]

2. Descripción del experimento

En el experimento se utilizan 10 funciones de prueba. Las características de cada uno de los problemas se presentan en la Tabla 1. Cabe mencionar que en la mayoría de los algoritmos evolutivos se tienen como condición de paro un número determinado de generaciones/iteraciones establecidas por el usuario o si la función fitness obtiene el valor de minimización que se está buscando. Sin embargo en el presente trabajo se tomó como condición de paro 10,000 llamadas a la función fitness, independientemente del número de iteraciones que represente debido a que algunos algoritmos realizan más de una evaluación por individuo de la función fitness por generación. De esta manera, se obtendrán valores tomando un parámetro en común con todos los algoritmos, de otra manera no es tan sencillo debido a que cada uno de los algoritmos definen variables de restricciones de manera distinta.

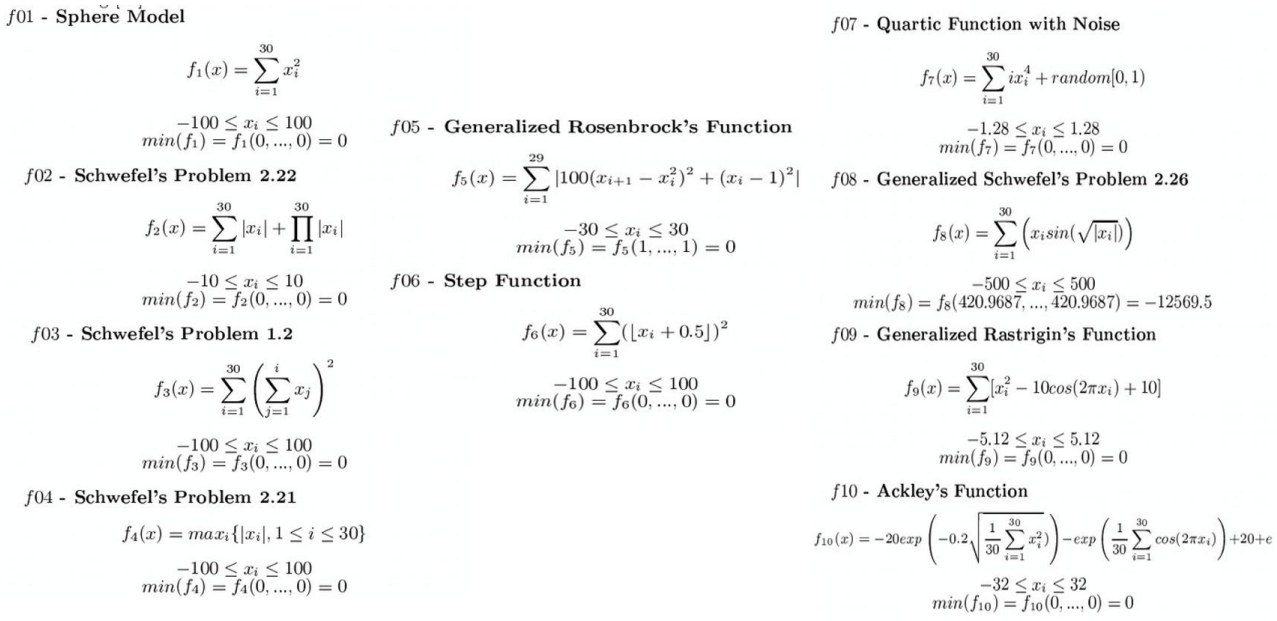

Figura 1 Funciones a evaluar.

3. Análisis del experimento

Como se mencionó anteriormente, se realizó la ejecución de cada uno de los algoritmos. En Figura 2 se puede observar un ejemplo del comportamiento de PSO, donde todas las partículas tienden hacia un mismo valor.

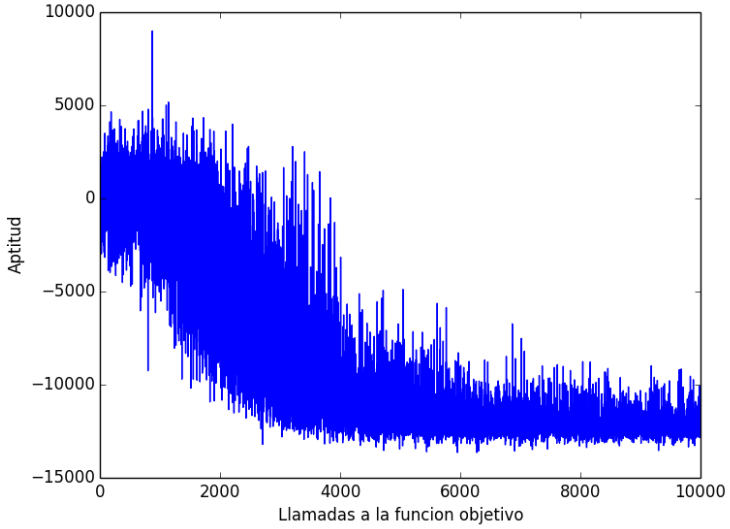

Figura 2. Tendencia de partículas del algoritmo PSO.

Como se mencionó anteriormente, se realizó la ejecución de los tres algoritmos tomando como criterio de parada 10,000 llamadas a la función objetivo. En la Tabla 3 se muestran los resultados obtenidos.

Tabla 1. Resultados algoritmos.

	Estrategia evolutiva	Evolución Diferencial	PSO	**Mínimo**
F1	5839	48.35	1.169	**0**
F2	106	0.097	2.481	**0**
F3	1309.82	0.168	0.428	**0**
F4	5.72	0.039	0.14	**0**
F5	17.6	5.646	2.375	**0**
F6	4475	5.837	5.722	**0**
F7	1.04	0.037	0.001	**0**
F8	-7517	-12375	-12569.3	**-12569.5**
F9	162.7	84.85	75.49	**0**
F10	21.71	3.43	1.718	**0**

De acuerdo a los resultados obtenidos en las Tabla1 se puede observar que se en la mayoría de los casos PSO obtiene mejores óptimos, mientras que Estrategias evolutivas es el que tiene los valores más lejanos del mínimo.

Los valores utilizados para el control de las restricciones en los algoritmos son los siguientes: Para el caso de Estrategias evolutivas: número de individuos=100, porcentaje de mutación=0.8; Evolución diferencial: número de individuos=100, porcentaje de cruza= 0.2, porcentaje de mutación= 1/5; PSO: Número de partículas=50, w(inercia) = 0.729, C1 (cognitive particle)= 1.49445, C2(social)= 1.49445.

4. Modelo

En la Figura 3, Figura 4 y Figura 5 se presentan los pseudocódigos de los modelos que representan a cada uno de los algoritmos.

A) Inicio
Generar una población aleatoria inicial con un tamaño = popsize
Evaluar cada individuo de la población
Para i = 1 hasta MaxGenerations
j=0
Mientras(j <= popsize)
El vector padre "j" genera un vector hijo "j´" variante *ED/rand/1/bin*
Evaluar el vector hijo "j´"
Comparar los vectores padre "j" e hijo "j" con **reglas de factibilidad**
El mejor permanecerá para la siguiente generación
j = j + 1
Fin de Mientras
Fin de Para
Fin

Figura 3. Pseudo-código Evolución Diferencial.

C) Inicio
Generar una población inicial con tamaño = μ
Evaluar cada individuo en la población
Para i = 1 hasta MaxGenerations
j=0
Mientras (j <= λ)
Selecciona 3 individuos aleatoriamente en μ
Aplicar recombinación para crear un hijo "j"
Aplicar mutación Gaussiana a "j"
Evaluar el hijo "j"
j = j + 1
Fin de Mientras
Escoger de entre los $\mu + \lambda$ individuos a los μ
Mejores **usando reglas de factibilidad**
Fin de Para
Fin

Figura 4. Pseudo-código Estrategias Evolutivas.

D) Inicio
Generar una población aleatoria inicial de
tamaño = *parsize*
Evaluar cada individuo en el cúmulo
Para i = 1 hasta MaxGenerations
Seleccionar el líder del cúmulo
con reglas de factibilidad
j=0
Mientras (j <= *parsize*)
Aplicar formula de vuelo a la partícula "j"
Aplicando fórmula con factor de inercia.
Evaluar la nueva posición de la partícula Actualizar el valor pbest (memoria) de la partícula "j" usando las **reglas de factibilidad**
j = j + 1
Fin de Mientras
Fin de Para
Fin

Figura 5. Pseudo-código PSO.

Una vez realizados los algoritmos se graficaron cada uno de los resultados para observar los comportamientos. En la Figura 6 se muestra un ejemplo de comparación entre los 3 algoritmos de la función 8.

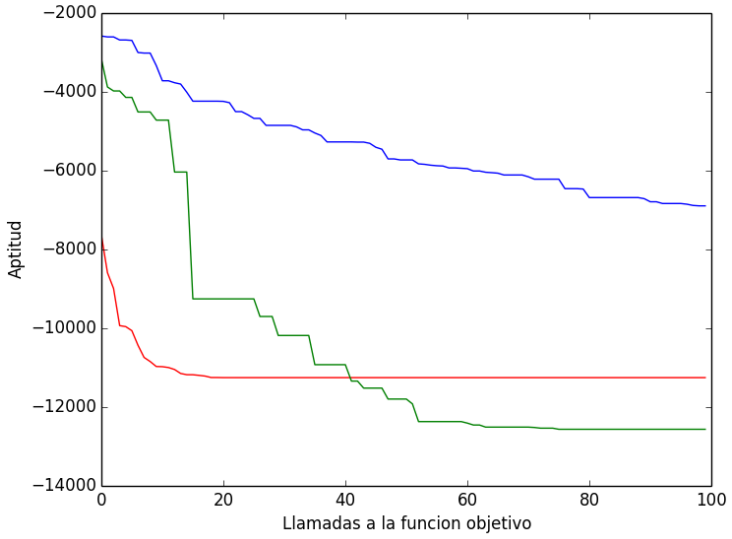

Figura 6. Comparación Estrategias Evolutivas (azul), Evolución diferencial (rojo) y PSO (verde).

5. Análisis

Como se puede observar en la Tabla 1 y Figura 5, los mejores óptimos los obtuvo PSO, mientras que los resultados más lejanos del mínimo fueron de Estrategias Evolutivas. Este comportamiento se repite en todas las funciones. Cabe mencionar que Evolución diferencial tuvo resultados muy cercanos a PSO, siendo igual de eficaces para las funciones presentadas.

6. Conclusiones

Las conclusiones principales se mencionan a continuación:

a) PSO mostró los resultados de calidad de manera consistente y más cercanos a los mínimos.

b) Evolución Diferencial obtuvo resultados competitivos.

c) Estrategias evolutivas mostró los resultados más pobres y en varios problemas se quedó lejos de los valores óptimos.

Como trabajo a futuro se busca realizar la comparación de los algoritmos con otras funciones de prueba y observar su funcionamiento, debido a que para ciertos problemas de optimización un algoritmo puede dar mejores soluciones que el resto.

1. Referencias y recursos electrónicos

1) Lucken C, Hermosilla A. y Barán B. Algoritmos Evolutivos para Optimización Multiobjetivo: un Estudio Comparativo en un Ambiente Paralelo Asíncrono. Campus Universitario de San Lorenzo. Pages 1-9. 2004.

2) Estevez P. Optimización Mediante Algoritmos Genéticos. ResearchGate. Pages 83-92. 2007

3) López-Ramírez B. y Mezura-Montes E. Comparación de algoritmos evolutivos y bio-inspirados en problemas de optimización con restricciones. Centro de Investigaciones en Óptica CIATEC. Pages 2-5.