

Optimización con Interacciones Gravitacionales

Juan J. Flores, Rodrigo López, and Julio Barrera,

Resumen—Computación evolutiva esta inspirada en la naturaleza para formular metaheurísticas capaces de optimizar varios tipos de problemas. Una familia de algoritmos han surgido basándose en esta idea; e.g. algoritmos genéticos, estrategias evolutivas, optimización con enjambre de partículas (PSO), optimización con colonia de hormigas (ACO), etc. En esta publicación se muestra una metaheurística basada en población que se inspira en las fuerzas gravitacionales producidas por las interacciones de las masas de un conjunto de cuerpos. El algoritmo propuesto es capaz de encontrar los óptimos de funciones unimodales y multimodales comunmente utilizadas para medir el desempeño de los algoritmos evolutivos. El algoritmo propuesto supera a Optimización con Enjambre de Partículas (PSO) con nichos, no depende del parámetro radio ni de nichos para resolver problemas multimodales. Se compara OIG con otras metaheurísticas respecto al número medio de evaluaciones que se necesitan para encontrar las soluciones óptimas

Temas Claves: Optimización, interacciones gravitacionales, computación evolutiva, metaheurística

I. INTRODUCCIÓN

Problemas de optimización multimodal tratan con funciones objetivo que comúnmente contienen mas de una solución con el óptimo global. Para encontrar todas las soluciones con el óptimo global en problemas multimodales con métodos numéricos clásicos, normalmente estos métodos se ejecutan varias veces con diferentes valores iniciales esperando encontrar todos los óptimos globales. Este tipo de técnicas no garantizan la localización de todas las soluciones. Además, este tipo de técnicas no son las mas adecuadas para explorar funciones multimodales con espacios de búsqueda grandes y complejos. En la literatura de computación evolutiva existen una variedad de metaheurísticas desafiando los problemas típicos de optimización clásica. e.g. En optimización con enjambre de partículas con nichos; la mejor partícula crea un nicho de radio r que contiene a un grupo de partículas hasta que el nicho este lleno; después se selecciona la siguiente mejor partícula que no forma parte de algún nicho y las partículas que se encuentran dentro de su radio forman otro nuevo nicho; el proceso se repite hasta que todas las partículas forman parte de un nicho. Estiramiento de función objetivo presentado por Parsopolous [7], [8] es otro algoritmo que modifica el espacio de búsqueda de la función objetivo para remover los óptimos locales y prevenir la convergencia prematura en optimización con enjambre de partículas. En un problema de minimización, un posible mínimo local es estirado para convertirlo en máximo local permitiendo explorar otras secciones del espacio de búsqueda identificando nuevas

soluciones. el Algoritmo GSA presentado por Rashedi [9], es una metaheurística gravitacional sin memoria capaz de encontrar solo una solución con el óptimo global en problemas multimodales y unimodales, donde la masa mas pesada representa la mejor solución y la constante gravitacional G es utilizada para ajustar la precisión de la búsqueda.

Este trabajo explora las propiedades de las interacciones gravitacionales para diseñar una metaheurística útil diseñada para encontrar óptimos en problemas unimodales y multimodales. La sección 2 se refiere a la motivación principal de este trabajo: La ley de la gravitación universal de Newton. En la sección 3 se define la metaheurística para funciones unimodales y multimodales Optimización con Interacciones Gravitacionales (OIG). En la sección 4 se presenta la metaheurística OIG con diferentes funciones unimodales y multimodales. En la sección 5 se presentan las conclusiones de este trabajo.

A. Ley de la Gravitación Universal de Newton

La fuerza de atracción de dos partículas es proporcional a sus masas e inversamente proporcional a su distancia. La ley de la atracción universal fue propuesta por Isaac Newton [6]. Esta ley se enuncia en la Definición 1.

DEFINICIÓN 1: La fuerza entre dos partículas cualesquiera con masas m_1 y m_2 , separadas por una distancia r , es una fuerza de atracción actuando a lo largo de una línea uniendo a las partículas; ver la Ecuación (1).

$$F = G \frac{m_1 m_2}{r^2} \quad (1)$$

donde G es la constante de la gravitación universal.

Las fuerzas entre dos partículas con masa son un par de acción y reacción. Las dos partículas con masas m_1 y m_2 ejercen fuerzas atractoras F_{12} y F_{21} hacia cada partícula, en donde sus magnitudes son iguales pero sus direcciones son opuestas.

La constante de la gravitación universal G es una constante obtenida empíricamente que se involucra en el cálculo de la atracción gravitacional entre partículas con una masa asignada; esta constante puede ser determinada por el método de máxima desviación [10].

$$G = 6.673 \times 10^{-11} N(m/kg)^2 \quad (2)$$

La fuerza gravitacional es extremadamente débil comparada con otras fuerzas fundamentales; v.g. la fuerza electromagnética es 39 órdenes de magnitud mayor que la fuerza de gravedad.

La ley de la gravitación universal de Newton puede ser escrita en notación vectorial, considerando la fuerza de las masas y la dirección de cada fuerza. La notación vectorial es mostrada en la Ecuación (3).

Juan J. Flores and Rodrigo López, División de Estudios de Posgrado, Facultad de Ingeniería Eléctrica, Universidad Michoacana de San Nicolás de Hidalgo, Morelia, Mich, e-mail: juanf@umich.mx, rlopez@faraday.fie.umich.mx.

Julio Barrera, e-mail: julio.barrera@gmail.com

$$F_{12} = -G \frac{m_1 m_2}{|r_{12}|^2} \hat{r}_{12} \quad (3)$$

dónde F_{12} es la fuerza ejercida por m_1 en m_2 , G es la constante gravitacional, m_1 y m_2 son las masas de las partículas, $|r_{12}|$ es la distancia euclidiana entre las partículas m_1 y m_2 , y \hat{r}_{12} es el vector unitario, definido como $\frac{r_2 - r_1}{|r_2 - r_1|}$, r_1 y r_2 son las localidades de las partículas m_1 y m_2 , esta interacción gravitacional se puede observar en la Figura 1.

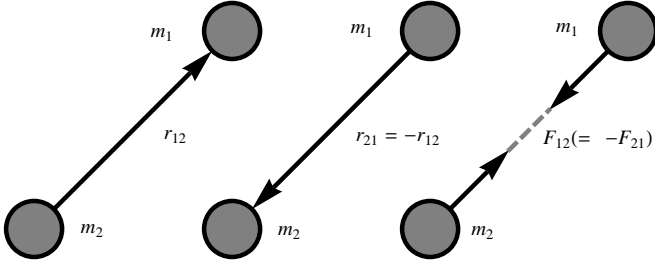


Fig. 1. La fuerza ejercida sobre m_2 por m_1 , La fuerza ejercida sobre m_1 por m_2 , Las dos fuerzas ejercidas por m_1 y m_2 provoca que se atraigan mutuamente con fuerzas equivalentemente opuestas $F_{21} = -F_{12}$

En la Figura 1 se ejemplifica la manera en que los cuerpos gravitacionales interactúan. de derecha a izquierda:

- La fuerza ejercida sobre m_2 (por m_1), F_{21} , es de dirección opuesta al desplazamiento, r_{12} , de m_2 hacia m_1 .
- La fuerza ejercida sobre m_1 (por m_2), F_{12} , es de dirección opuesta al desplazamiento, r_{21} , de m_1 hacia m_2 .
- $F_{21} = -F_{12}$, las fuerzas son un par de acción-reacción.

B. Optimización por Interacciones Gravitacionales

Para encontrar mas de un óptimo existen una gran variedad de algoritmos evolutivos, v.g. algoritmos genéticos [2], estrategias evolutivas, [3], optimización por colonia de hormigas [1], optimización por enjambre de partículas [4], algoritmo PSO basado en interacciones electrostáticas (EPSO), etc. También existen trabajos relacionados al diseño de metaheurísticas que toman en cuenta la distancia para determinar la pertenencia a una aglomeración respecto a la partícula que será actualizada, v.g. FER-PSO [5]. En este apartado se propone la metaheurística Optimización por Interacciones Gravitacionales (OIG), capaz de resolver problemas de optimización. La motivación del diseño de esta metaheurística es encontrar propiedades útiles y analogías que puedan relacionar los problemas de optimización con la teoría de la gravitación Universal de Isaac Newton.

Bajo esta premisa, las interacciones son provocadas por un conjunto de cuerpos con una masa asignada y se utilizan para guiar la búsqueda para encontrar el óptimo global en un problema de optimización.

1) *Interacciones Gravitacionales para Optimización Unimodal*: OIG es una meta-heurística basada en población donde un conjunto de cuerpos son inicializados de manera dispersa en el espacio de búsqueda bajo una distribución aleatoria uniforme.

La aptitud de los cuerpos localizados en el espacio de búsqueda son mapeados como masas en un campo gravitacional donde las soluciones son evolucionadas.

Cada cuerpo guarda su posición actual B y la mejor posición hasta el momento B_{best} de acuerdo a una función de aptitud. Los cuerpos son capaces de interactuar de una forma síncrona discreta por un número de épocas. Las interacciones de los cuerpos siguen la ley gravitacional de newton y mueven cada cuerpo a una nueva localidad de tal manera que la toda la población tiende a alcanzar el óptimo (o muchos óptimos globales para problemas multimodales).

La función de aptitud es un mapeo que transforma un vector $X = [x_1, x_2, \dots, x_n]$ a un escalar $f(X)$. Este mapeo asocia el valor de la aptitud $f(X)$ a cada localidad $X = [x_1, x_2, x_3 \dots x_n]$ del espacio de búsqueda. Se asigna un cuerpo B a cada localidad X en el espacio de búsqueda donde se encuentra un individuo de la población. Al cuerpo B se le asigna una masa cuya magnitud es una función de la aptitud de su localidad.

La ley universal de Newton describe la fuerza de atracción existen entre dos cuerpos puntuales con masas (descrito de forma vectorial en la Ecuación 3). Sustituyendo, se obtiene la Ecuación (4)

$$\vec{F}_{ij} = \frac{M(f(B_i)) \cdot M(f(B_{j,best}))}{|B_i - B_{j,best}|^2} B_i \hat{B}_{j,best} \quad (4)$$

Donde M es la función de mapeo que asocia el valor de aptitud f de dominio $\{x : x \in \mathbb{R}\}$ a una masa de codominio $\{y : y \in (0, 1]\}$ para cada posición del cuerpo B_i . este mapeo es calculado con la Ecuación (5).

$$M(f(B_i)) = \left(\frac{f(B_i) - \min f(B)}{\max f(B_{best}) - \min f(B)} (1 - \text{mapMin}) + \text{mapMin} \right)^2 \quad (5)$$

Donde B_i es la posición del i -ésimo cuerpo y B_j es el j -ésimo cuerpo que contribuye ejerciendo una fuerza en la masa B_i ; $|B_i - B_j|$ es la distancia euclidiana y \hat{B}_{ij} es el vector unitario entre los cuerpos B_i y B_j ; $f(B_i)$ es la aptitud del cuerpo B_i , $\min f(B)$ es el valor mínimo de aptitud de las posiciones actuales de los cuerpos, $\max f(B_{best})$ es el valor máximo de aptitud de las mejores posiciones B_i hasta el momento, mapMin es una constante con un pequeño valor positivo cercano a cero, tal que $1 - \text{mapMin}$ re-escala el valor de aptitud $f(B_i)$ a una masa con valores entre $[0, 1)$. El resultado es elevado al cuadrado para enfatizar las mejores y peores aptitudes.

Una característica de el método propuesto es la total interacción de las partículas; v.g. cada cuerpo B_i interactúa con otro cuerpo B_j a través de sus masas. Las interacciones de todas las partículas con la partícula B_i contribuyen a su desplazamiento, según la fuerza resultante. La Ecuación (6) calcula la fuerza resultante \vec{F}_i que se dirige hacia un cuerpo B_k desde la posición de la masa B_i y esta fuerza es ejercida sobre el cuerpo B_i por los cuerpos B_j .

$$\vec{F}_i = \sum_{j=1}^n \frac{M(f(B_i)) \cdot M(f(B_{j,best}))}{|B_i - B_{j,best}|^2} (B_i \hat{B}_{j,best}) \quad (6)$$

Donde \vec{F}_{ik} es la fuerza resultante de la suma de todos los vectores de fuerza entre $M(B_i)$ y $M(B_{j,best})$, $|B_i - B_{j,best}|$ es la distancia euclidiana entre la posición actual del cuerpo B_i y la mejor posición hasta el momento del cuerpo B_j . Para prevenir errores numéricos, se calcula la fuerza entre las masas $M(B_i)$ y $M(P_{j,best})$ solo si $|B_i - B_j| \geq \times 10^{-5}$, $B_i \hat{B}_{j,best}$ es el vector unitario que dirige la fuerza.

Para estimar el desplazamiento que pudiera mejorar la solución de la partícula B_i , es necesario resolver la Ecuación (4) para B_j . Considerando que se quiere encontrar aquella localidad del cuerpo B_k con $M(f(B_k)) = 1$, B_k es calculada utilizando la Ecuación (7).

$$B_k = \sqrt{\frac{M(f(B_i))}{|\vec{F}_i|}} \hat{F}_i \quad (7)$$

Para actualizar la posición de los cuerpos se utilizan las Ecuaciones (8) y (9).

$$V_{new} = \chi(V + R \cdot C \cdot B_k) \quad (8)$$

$$B_{t+1} = B + V_{new} \quad (9)$$

Donde V es la velocidad actual de B_i , R es un número real aleatorio generado en el rango de $[0,1)$ y es multiplicado por la constante de interacción gravitacional C . Para esperar una exploración aleatoria con distancias con una media igual al desplazamiento calculado, se hace $C = 2.01$. Este desplazamiento es restringido multiplicando por una constante χ con un valor de 0.86, para evitar que los cuerpos exploren lejos del espacio de búsqueda y puedan converger. B_k es el desplazamiento principal calculado por la Ecuación (7).

El algoritmo OIG completo es desrito por los Algoritmos 1, 2 y 3. El Algoritmo 1 calcula el total de la fuerza que ejercen las masas $M(f(B_j))$ y $M(f(B_i))$; para prevenir convergencia prematura y división por 0, solo se calculan aquellos pares de cuerpos con una distancia mas grande que ϵ . El Algoritmo 2 calcula las velocidades de los cuerpos; este algoritmo recibe los cuerpos y calcula la fuerza resultante que atrae las masas asignadas a B_i . Para prevenir la división por 0 se calcula la distancia solo si $|F_{total}| > 0$, la nueva velocidad es calculada por la Ecuación (8). Finalmente se actualizan las velocidades asociadas a B_i . El algoritmo 3 calcula las nuevas posiciones B de cada iteración t ; el algoritmo recibe como parámetros el rango de búsqueda, el número de cuerpos $nCuerpos$, y el máximo número de iteraciones $maxIter$. El Algoritmo calcula las velocidades con **calcularVelocidades()** (Algoritmo 2) y recibe como parámetros los cuerpos, y actualiza sus posiciones con **actualizarPosicion()** (Algoritmo 4), la cual implementa la Ecuación (9), **limitarPosiciones()** (Algoritmo 5) limita las posiciones de los cuerpos al espacio de búsqueda, **actualizarAptitud()** actualiza la aptitud de acuerdo a las nuevas posiciones de los cuerpos y finalmente se actualizan las mejores posiciones hasta el momento con **actualizarBbest()**.

Algoritmo 1 calcularFtotal(indice, cuerpos)

Entrada: *indice, cuerpos*

Salida: *Ftotal*

```

1:  $B \leftarrow \text{cuerpos}$ 
2:  $i \leftarrow \text{indice}$ 
3:  $F_{total} \leftarrow 0$ 
4: para  $j \leftarrow 1$  hasta tamañoLista( $B$ ) hacer
5:   si  $\text{distancia}(B_i, P_j) > \epsilon$  entonces
6:      $F_{total} \leftarrow F_{total} + \frac{\hat{P}_{ij} M(f(B_i)) M(f(P_j))}{\text{distancia}(B_i, B_j)^2}$ 
7:   fin si
8: fin para
```

Algoritmo 2 calcularVelocidades(cuerpos)

Entrada: *cuerpos*

Salida: *Ftotal*

```

1:  $B \leftarrow \text{cuerpos}$ 
2: para  $i \leftarrow 1$  hasta tamañoLista( $B$ ) hacer
3:    $F_{total} \leftarrow \text{calcularFtotal}(i)$ 
4:   si  $|F_{total}| > 0$  entonces
5:      $\text{distancia} \leftarrow \sqrt{M(f(B_i)) / |F_{total}|}$ 
6:   si no
7:      $\text{distancia} \leftarrow 0$ 
8:   fin si
9:    $V_{new} \leftarrow \chi(V + R \cdot C \cdot \text{distancia} \cdot F_{total})$ 
10:  actualizarVelocidad( $B_i, V_{new}$ )
11: fin para
```

Algoritmo 3 InteraccionesGravitacionales(rangos, nCuerpos)

Entrada: *rangos, nCuerpos*

Salida: *cuerpos*

```

1:  $\text{cuerpos} \leftarrow \text{inicializarParticulas}(nCuerpos, rangos)$ 
2: para  $t \leftarrow 0$  to  $maxIter$  hacer
3:   calcularVelocidades( $\text{cuerpos}$ )
4:   actualizarPosición()
5:   limitarPosicion()
6:   actualizarAptitud()
7:   actualizarBbest()
8: fin para
```

Algoritmo 4 actualizarPosición(cuerpos)

Entrada: *cuerpos*

```

1:  $B \leftarrow \text{cuerpos}$ 
2: para  $i \leftarrow 0$  hasta tamañoLista( $B$ ) hacer
3:    $B_i \leftarrow \text{Posicion}(B_i) + \text{Velocidad}(B_i)$ 
4: fin para
5:  $\text{cuerpos} \leftarrow B$ 
```

Salida: *cuerpos*

Algoritmo 5 limitarPosición(*rangos*, *cuerpos*)

Entrada: *rangos*, *cuerpos*

```

1:  $B \leftarrow \text{cuerpos}$ 
2:  $D \leftarrow \text{tamañoLista}(\text{rangos})$ 
3: para  $i \leftarrow 0$  hasta  $\text{tamañoLista}(B)$  hacer
4:   para  $d \leftarrow 1$  hasta  $D$  hacer
5:     si  $\text{Minimo}(\text{rangos}_d) \geq B_{i,d}$  entonces
6:        $B_{i,d} \leftarrow \text{Minimo}(\text{rangos}_d)$ 
7:     si no
8:       si  $\text{Maximo}(\text{rangos}_d) \leq B_{i,d}$  entonces
9:          $B_{i,d} \leftarrow \text{Maximo}(\text{rangos}_d)$ 
10:    fin si
11:  fin si
12: fin para
13: fin para
14:  $\text{cuerpos} \leftarrow B$ 
Salida:  $F_{\text{total}}$ 

```

Este esquema desempeña buenos resultados para problemas unimodales. Los resultados del desempeño del algoritmo son presentados en la Subsección I-D.

2) *Interacciones Gravitacionales para Optimización Multimodal*: En la subsección anterior se mostró el algoritmo básico de la metaheurística Interacciones gravitacionales. Este esquema demuestra que funciona para problemas con un solo óptimo global. Para problemas multimodales con mas de un óptimo global es necesario agregar un componente cognitivo, análogo al utilizado en PSO [4]; el componente cognitivo siempre recuerda la mejor posición visitada hasta el momento por el cuerpo y es multiplicado por una constante C .

Las nuevas posiciones de los cuerpos son calculadas con las Ecuaciones (10) y (11).

Agregando el componente cognitivo a la Ecuación (8) y usando el factor de constricción χ (Ecuación (12)), se forma la nueva Ecuación (10) este modelo es capaz de encontrar mas de un óptimo en problemas multimodales. El efecto de este componente es hacer la búsqueda local mas robusta restringiendo a los cuerpos a la búsqueda local, hasta que las fuerzas gravitacionales de las aglomeraciones sobrepasen la fuerza ejercida por su componente cognitivo.

$$V_{\text{new}} = \chi (V + C_1 \cdot R_1 \cdot (B_{\text{best}} - B) + C_2 \cdot R_2 \cdot B_k) \quad (10)$$

$$B_{t+1} = B + V_{\text{new}} \quad (11)$$

donde, C_1 y C_2 son constantes del componente cognitivo y de interacción gravitacional, R_1 y R_2 son números reales aleatorios en el rango de $[0, 1)$ y χ es la restricción de inercia, usada para prevenir que los cuerpos exploren mas allá del espacio de búsqueda. χ Es calculada con la Ecuación (12).

$$\chi = \frac{2\kappa}{|2 - \phi - \sqrt{\phi^2 - 4\phi}|} \quad (12)$$

donde $\phi = C_1 + C_2 > 4$, C_1 y C_2 ; κ es un valor arbitrario entre los valores $(0, 1]$. En nuestro algoritmo determinamos $C_1 = C_2 = 2.01$. El factor de constricción ayuda al algoritmo a converger a través de las iteraciones.

To make multimodal Gravitational Interactions Algorithm (Algorithms 1, 2, and 3) described in the previous subsection, we replace line 8 in Algorithm 2 by Equation 10. Para hacer que Optimización por Interacciones Gravitacionales pueda resolver problemas multimodales con mas de un óptimo global (Algoritmos 1, 2, y 3), descrito en la subsección anterior, se reemplaza la línea 8 en el Algoritmo 2 por la Ecuación (10).

C. Experimentos

Para probar el desempeño de Optimización por interacciones Gravitacionales para funciones unimodales y multimodales se probaron ambas versiones con algunas funciones usadas comúnmente para medir el rendimiento de diferentes tipos de metaheurísticas.

1) *Funciones de Prueba*: Esta subsección se muestra el rendimiento del algoritmo de Optimización con Interacciones Gravitacionales unimodal y multimodal con 3 funciones unimodales y 4 funciones multimodales, las cuales son mostradas en la Tabla I.

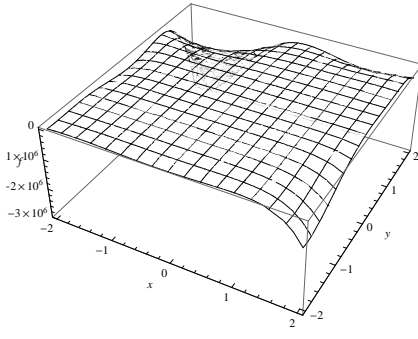
TABLE I
FUNCIONES DE PRUEBA UTILIZADAS PARA LOS EXPERIMENTOS

| Funciones de Prueba Unimodales | | |
|----------------------------------|--|--|
| $U1$ | $U1 = [1 + (1 + (x + y + 1)^2) \cdot (19 - 14x + 3y^2 + 6xy + 3y^2)] \cdot [(30 + (2x - 3y)^2) \cdot (18 - 32x + 12x^2 + 48y - 36xy + 27y^2)]$ | $-2 \leq x, y \leq 2$ |
| $U2$ | $U2 = (x + 2y - 7)^2 + (2x + y - 5)^2$ | $-10 \leq x \leq 10$ $-10 \leq y \leq 10$ |
| $U3$ | $U3 = -1100 \cdot (w^2 - x)^2 + (w - 1)^2 + (y - 1)^2 + 90 \cdot (y^2 - z)^2 + 10.1 \cdot ((x - 1)^2 + (z - 1)^2) + 19.8 \cdot (x^{-1}) \cdot (z - 1)$ | $-10 \leq w, x \leq 10$ $-10 \leq y, z \leq 10$ |
| Funciones de Prueba Multimodales | | |
| $M1$ | $M1 = -((y - \frac{5.1x^2}{4\pi^2} + \frac{5x}{\pi} - 6)^2 + 10(1 - \frac{1}{8\pi})\cos(x) + 10)$ | $-5 \leq x \leq 10$ $0 \leq y \leq 15$ |
| $M2$ | $M2 = \sin(5\pi x)^6$ | $-0 \leq x \leq 1$ |
| $M3$ | $M3 = -(x^2 + y - 11)^2 - (x + y^2 - 7)^2$ | $-6 \leq x, y \leq 6$ |
| $M4$ | $M4 = -4((4 - 2.1x^2 + \frac{x^4}{3})x^2 + xy + (-4 + 4y^2)y^2)$ | $-1.9 \leq x \leq 1.9$ $-1.1 \leq y \leq 1.1$ |

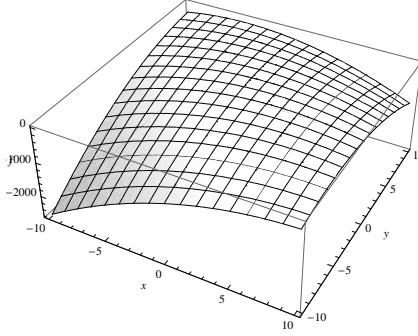
Para optimización unimodal se utilizaron las funciones de la Figura 3: $U1$ es la función de Goldstein y Price mostrada en la Figura 2(a), $U2$ es la función Booth mostrada en la Figura 2(b) y $U3$ es la función de Colville de 4 variables. Para optimización multimodal se utilizaron las funciones de la Figura 3 $M1$ es la función Branin's RCOS con 4 óptimos globales (sin ningún óptimo local) mostrado en la Figura 3(a), $M2$ es la función de Deb univariable con 6 máximos globales mostrado en la Figura 3(b), $M3$ es la función de Himmelblau con 4 óptimos globales mostrados en la Figura 3(c), $M4$ es la función del Camello de 6 jorobas con 2 óptimos globales y 4 óptimos locales mostrado en la Figura 3(d).

D. Resultados

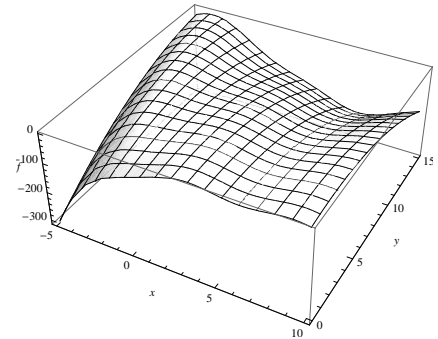
En los experimentos realizados se consideró $\epsilon = 1 \times 10^{-3}$, al igual que para el Algoritmo 1 para ser un error aceptable que determine si la solución obtenida alcanzó el óptimo. Se inicializaron 100 cuerpos para 1000 iteraciones, como condición de parada se consideró la inhabilidad de todos los



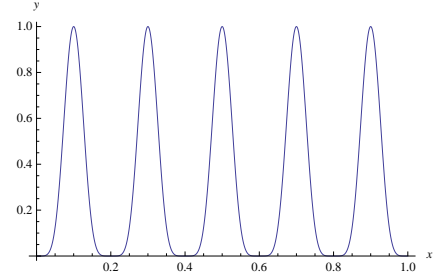
(a) Función de Goldstein y Price



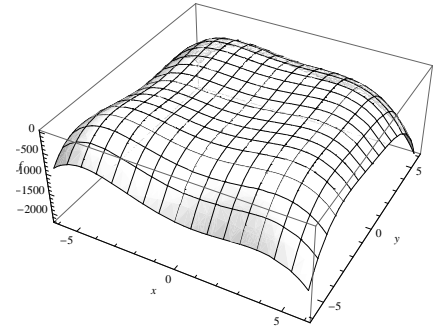
(b) Función Booth



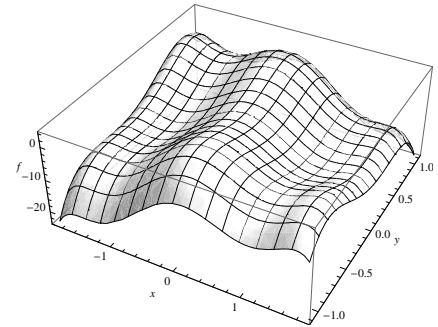
(a) función Branin's RCOS



(b) Función Deb



(c) Función de Himmelblau



(d) Función Camello de 6 jorobas

Fig. 2. Funciones para probar el desempeño de OIG unimodal

TABLE II
RESULTADOS DE LOS EXPERIMENTOS

| Función | PSO | | | OIG Unimodal | | |
|-----------|--------------|----------|-------|----------------|-----------|-------|
| | Evaluaciones | | Éxito | Evaluaciones | | Éxito |
| | μ | σ | | μ | σ | |
| <i>U1</i> | 1,394.44 | 399.22 | 20% | 16,523.33 | 13,928.90 | 100% |
| <i>U2</i> | 1,130.77 | 330.11 | 60% | 6,057.55 | 3,984.54 | 70% |
| <i>U3</i> | 764.00 | 777.75 | 83% | 530.00 | 208.69 | 100% |
| Función | NPSO | | | OIG Multimodal | | |
| | Evaluaciones | | Éxito | Evaluaciones | | Éxito |
| | μ | σ | | μ | σ | |
| <i>M1</i> | 2,529.17 | 764.13 | 80% | 2,803.33 | 972.90 | 100% |
| <i>M2</i> | 276.66 | 81.72 | 100% | 390.00 | 88.44 | 100% |
| <i>M3</i> | 3,400.00 | 0.00 | 00.3% | 2,323.33 | 288.496 | 100% |
| <i>M4</i> | 1,136.67 | 303.41 | 100% | 1,600.00 | 501.721 | 100% |

cuerpos de mejorar las aptitudes de sus memorias por más de 1×10^{-4} , o cuando el algoritmo haya encontrado todos los óptimos. Cada experimento fue repetido 30 veces. Para PSO con nichos fueron asignados diferentes tamaños de radio r , y el diferente número de máximo de partículas en cada nicho $nMax$, para *M1* se usó $r = 0.5$ y $nMax = 50$, *M2* con $r = 0.1$ y $nMax = 15$, *M3* con $r = 0.5$ y $nMax = 30$, y *M4* con $r = 0.5$ y $nMax = 25$.

El desempeño de Optimización con Interacciones Gravitacionales se compara con Optimización con enjambre de Partículas con nichos (NPSO) en la Tabla II, considerando la media μ y la desviación estándar σ de las evaluaciones requeridas para encontrar todos los óptimos globales mostrado en la columna **Evaluaciones**, y el porcentaje de éxitos mostrada en la columna **Éxitos**, se considera éxito cuando el algoritmo encuentra todos los óptimos de las funciones de prueba para encontrar todos los óptimos.

Los resultados obtenidos muestran que las versiones uni-

Fig. 3. Funciones de aptitud utilizadas para OIG multimodal

modales y multimodales de Interacciones Gravitacionales presentan una mayor probabilidad de converger a un óptimo global, evitando la convergencia prematura que presentan PSO y PSO con nichos con un número similar de evaluaciones.

E. Conclusiones

Se presentó una nueva heurística, Optimización por Interacciones Gravitacionales que demostró ser mas confiable que PSO. Una de las ventajas de este algoritmo es que no necesita

parámetros adicionales como el radio r y el máximo número de partículas $nMax$ que necesita el algoritmo de PSO con nichos. En problemas con alta dimensionalidad el radio r es difícil de determinar a priori, así que se prueban diferentes valores de r hasta encontrar uno con el cual la metaheurística funcione correctamente ya que no es posible hacer un análisis visual de la función objetivo para determinar este valor. La versión general (OIG multimodal) es utilizada para resolver problemas unimodales (Incluyendo el componente cognitivo), por lo tanto OIG resuelve ambos casos sin alguna información a priori. Añadiendo el componente cognitivo se habilita el algoritmo para resolver ambos casos de optimización, Mientras GSA puede solo resolver problemas unimodales. Mientras OIG prueba que puede encontrar las soluciones con el valor del óptimo global, GSA solo determina uno de ellos.

REFERENCES

- [1] Alberto Colomi, Marco Dorigo, and Vittorio Maniezzo. Distributed optimization by ant colonies. In *Proceedings of the Parallel Problem Solving from Nature Conference*. Elsevier Publishing, 1992.
- [2] David E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, 1 edition, January 1989.
- [3] Rechenberg Ingo. *Evolutionsstrategie '94*. PhD thesis, Technische Universität Berlin, 1994.
- [4] J. Kennedy and R. Eberhart. *Swarm Intelligence*. Evolutionary Computation. Morgan Kaufmann Publisher, 2001.
- [5] Xiaodong Li. A multimodal particle swarm optimizer based on fitness euclidean-distance ratio. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation (GECCO '07)*, pages 78–85, New York, NY, USA, 2007. ACM.
- [6] I. Newton. *Newtons Principia Mathematica*. Física. Ediciones Altaya, S.A., 21 edition, 1968.
- [7] K. E. Parsopoulos, G.D. Magoulas, Uxbridge Ub Ph, M.N. Vrahatis, and V.P. Plagianakos. Stretching technique for obtaining global minimizers through particle swarm optimization. In *Proceedings of the Particle Swarm Optimization Workshop*, pages 22–29, 2001.
- [8] K. E. Parsopoulos, V. P. Plagianakos, G. D. Magoulas, and M. N. Vrahatis. Improving the particle swarm optimizer by function “stretching”. *Nonconvex Optimization and its Applications*, 54:445–458, 2001.
- [9] Esmat Rashedi, Hossein Nezamabadi-pour, and Saeid Saryazdi. Gsa: A gravitational search algorithm. *Information Sciences*, 179(13):2232–2248, 2009.
- [10] Hallyday. Robert, Resnick. David. *Physics Part I*. Physics. John Wiley and Sons Inc, 1966.