

Gannet Optimization Algorithm

Dr. Broderick Crawford Labrín

Pontificia Universidad Católica de Valparaíso

Gannet Optimization Algorithm (GOA)

- Fue desarrollada por Shu-Chuan Chu et. al. en el año 2022 ^a.
- Es una metaheurística basado en población diseñada para resolver problemas de optimización continuos.
- Sus soluciones (individuos) iniciales se generan aleatoriamente y se van alterando bajo un conjunto de reglas de movimiento con criterios estocásticos.

^a *Gannet optimization algorithm : A new metaheuristic algorithm for solving engineering optimization problems*, Algorithms (2022)

- Ecuaciones de movimiento

$$X = \begin{bmatrix} x_{1,1} & x_{1,j} & \dots & x_{1,Dim} \\ x_{2,1} & x_{2,j} & \dots & x_{2,Dim} \\ \dots & \dots & \dots & \dots \\ x_{N,1} & x_{N,j} & \dots & x_{N,Dim} \end{bmatrix} \quad (1)$$

$$x_{i,j} = r_1 \times (UB_j - LB_j) + LB, i = 1, 2, \dots, N, j = 1, 2, \dots, Dim \quad (2)$$

- Donde:

- X son soluciones aleatorias con la que empieza el algoritmo
- $x_{i,j}$ el como se calculan las soluciones presenten en X
 - UB_j y LB_j son el Upper y Lower Bound de la j -ésima posición
 - N es el número total de individuos en la población
 - r_1 es un número aleatorio entre $[0, 1]$

- Ecuaciones de movimiento (Exploración)

$$t = 1 - \frac{iter}{MaxIt} \quad (3)$$

$$a = 2 \cdot \cos(2 \cdot \pi \cdot r_2) \cdot t \quad (4)$$

$$b = 2 \cdot V(2 \cdot \pi \cdot r_3) \cdot t \quad (5)$$

$$V(x) = \begin{cases} -\frac{1}{\pi} \cdot x + 1, & x \in (0, \pi) \\ \frac{1}{\pi} \cdot x - 1, & x \in (\pi, 2\pi) \end{cases} \quad (6)$$

- Donde:

- *iter* iteración actual
- *MaxIt* maximo de iteraciones
- r_2 es un número aleatorio entre $[0, 1]$
- r_3 es un número aleatorio entre $[0, 1]$

- Ecuaciones de movimiento (Exploración)

$$MX(x) = \begin{cases} X_i^t + u_1 + u_2, q \geq 0,5 \\ X_i^t + v_1 + v_2, q < 0,5 \end{cases} \quad (7)$$

$$u_2 = A \cdot (X_i^t - X_r^t) \quad (8)$$

$$v_2 = B \cdot (X_i^t - X_m^t) \quad (9)$$

- Donde:

- u_1 es un número aleatorio entre a y -a
- v_1 es un número aleatorio entre b y -b
- X_i^t es el i^{th} individuo en la población actual
- X_r^t es un individuo aleatorio elegido en la población actual

- Ecuaciones de movimiento (Exploración)

$$A = (2 \cdot r_4 - 1) \cdot a \quad (10)$$

$$B = (2 \cdot r_5 - 1) \cdot b \quad (11)$$

$$X_m^t = \frac{1}{N} \sum_{i=1}^N X_i^t \quad (12)$$

- Donde:

- r_4 y r_5 son números aleatorio entre $[0, 1]$
- X_m^t denota la posición promedio de los individuos en la población actual

- Ecuaciones de movimiento (Explotación)

$$capturability = \frac{1}{R \cdot t2} \quad (13)$$

$$t2 = 1 + \frac{iter}{MaxIt} \quad (14)$$

$$R = \frac{M \cdot vel^2}{L} \quad (15)$$

$$L = 0,2 + (2 - 0,2) \cdot r_6 \quad (16)$$

- Donde:

- *capturability* es la capacidad de captura
- r_6 es un número aleatorio entre $[0, 1]$
- $M = 2,5kg$ siendo el peso del gannet
- $vel = 1,5m/s$ es la velocidad del gannet en el agua

- Ecuaciones de movimiento (Explotación)

$$MX(x) = \left\{ \begin{array}{l} t \cdot \delta \cdot (X_i^t - best^t) + X_i^t, \text{capturability} \geq c \\ best^t - (X_i^t - best^t) \cdot P \cdot t, \text{capturability} < c \end{array} \right\} \quad (17)$$

$$\delta = \text{capturability} \cdot |X_i^t - best^t| \quad (18)$$

$$P = \text{levy}(\text{Dim}) \quad (19)$$

- Donde:

- $MX(x)$ se utiliza en el caso de que el gannet no atrape al pez y busca a la siguiente preza de forma aleatoria
 - $c = 0,2$ es una constante
- $best^t$ es el individuo con el mejor desempeño en la población actual

- Ecuaciones de movimiento (Explotación)

$$levy(Dim) = 0,01 \times \frac{\mu \cdot \sigma}{|v|^{\frac{1}{\beta}}} \quad (20)$$

$$\sigma = \left(\frac{\Gamma(1 + \beta) \cdot \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \cdot \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right) \quad (21)$$

- Donde:

- $levy$ es la función de vuelo de levy
 - σ y μ son números aleatorios entre $[0, 1]$
 - $\beta = 1,5$ siendo una constante predeterminada

Algorithm 1 Gannet optimization algorithm

Input: Population $X = \{x_1, x_2, \dots, x_N\}$

Output: Updated population $X' = \{x'_1, x'_2, \dots, x'_N\}$ and best

```
1: Initialize random gannet population X, memory matrix MX, constant M, constant VEL and constant C
2: for it = 1 to MaxIt do
3:     calculate the fitness of each X
4:     select best
5:      $t = 1 - (it / \text{maxIter})$ 
6:      $X_r = X[\text{randint}(0, N-1)]$ 
7:     for j=0 to dim do
8:          $X_m.append(\text{average}(X[:,j]))$ 
9:          $r = \text{random}()$ 
10:        if  $r \geq 0.5$ 
11:            for i=0 to N do
12:                 $q = \text{random}()$ 
13:                if  $q \geq 0.5$ 
14:                    for j=0 to dim do
15:                         $r2 = \text{random}()$ 
16:                         $r4 = \text{random}()$ 
17:                         $a = 2 * \cos(2 * \pi * r2) * t$ 
18:                         $A = (2 * r4 - 1) * a$ 
19:                         $u1 = \text{random}(-a, a)$ 
20:                         $u2 = A * (X[i,j] - X_r[j])$ 
21:                         $MX[i,j] = X[i,j] + u1 + u2$ 
22:                    else
23:                        for j=0 to dim do
24:                             $r3 = \text{random}()$ 
25:                             $r5 = \text{random}()$ 
26:                             $x = 2 * \pi * r3$ 
27:                            if  $x > \pi$ 
28:                                 $vx = ((1 / \pi) * x - 1)$ 
```

GOA: Pseudocódigo

```
29:                                     else
30:                                         vx = -(1 / pi) * x + 1)
31:                                     b = 2 * vx * t
32:                                     B = (2 * r5 - 1) * b
33:                                     v1 = random(-b, b)
34:                                     v2 = B * (X[i,j] - Xm[j])
35:                                     MX[i,j] = X[i,j] + v1 + v2
36:     else
37:         t2 = 1 + (it / maxIter)
38:         for i=0 to N do
39:             r6 = random()
40:             l = 0.2 + (2 - 0.2) * r6
41:             R = (M * VEL**2) / l
42:             capturability = 1 / (R * t2)
43:             if capturability ≥ C
44:                 for j=0 to dim do
45:                     δ = capturability * |X[i,j] - best[j]|
46:                     MX[i,j] = t * δ * (X[i,j] - best[j]) + X[i,j]
47:                 else
48:                     for j=0 to dim do
49:                         σ = ((Γ(1 + beta) * sin(π * β/2)) / (Γ((1 + β)/2) * β * 2^((β - 1)/2)))^(1/β)
50:                         p = levy() = 0.01 * ((μ * σ) / (|v|^(1/β)))
51:                         MX[i,j] = best[j] - (X[i,j] - best[j]) * p * t
52:         for MX
53:             calculate fitness for MX
54:             if the value of MXi is better than the value Xi, replace Xi with MXi
55:     return updated population X* and best
```

GOA: Ejemplo práctico - parámetros iniciales

Considerando

$$\text{Min } z = x_1^2 + x_2^2$$

Sujeto a

$$x_1, x_2 \in [-100, 100]$$

Configuración inicial de GOA:

- Tamaño de la población: 2 individuos
- Número máximo de iteraciones: 500 iteraciones
- Constantes:
 - $m = 2.5$
 - $\text{vel} = 1.5$
 - $c = 0.2$
 - $\beta = 1.5$

GOA: Ejemplo práctico - soluciones iniciales

Soluciones iniciales:

ind 1: [74.3772, 44.5327] / fitness: 7515.1284

ind 2: [29.2269, 31.7354] / fitness: 1861.3477

Mejor solución:

ind 2: [29.2269, 31.7354] / fitness: 1861.3477

Ecuación general GOA:

$$MX(x) = X = \begin{bmatrix} 74,3772 & 44,5327 \\ 29,2269 & 31,7354 \end{bmatrix}$$
$$t = 1 - \frac{iter}{MaxIt} = 1 - \frac{1}{500} = 0,9980$$
$$t2 = 1 + \frac{iter}{MaxIt} = 1 + \frac{1}{500} = 1,0020$$
$$randomIndex = randint(0, N - 1) = 0$$

$$X_r^t = X[randomIndex] = [74,3772, 44,5327]$$
$$X_m^t = \frac{1}{N} \sum_{i=1}^N X_i^t = [51,8021, 38,134]$$

Ecuación general GOA:

$$r = \text{random}() = 0,0025$$

$$r \leq 0,5 \rightarrow 0,0025 \leq 0,5$$

$$r_6 = \text{random}() = 0,4360$$

$$L = 0,2 + (2 - 0,2) \cdot r_6 = 0,2 + (2 - 0,2) \cdot 0,4360 = 0,9848$$

$$R = \frac{M \cdot \text{vel}^2}{L} = \frac{2,5000 \cdot 1,5000^2}{0,9848} = 5,7117$$

$$\text{capturability} = \frac{1}{5,7117 \cdot 1,0020} = 0,1747$$

$$\text{capturability} < c \rightarrow 0,1747 < 0,2$$

GOA: Ejemplo práctico - iter 1 - ind 1 - dim 1

Ecuación general GOA:

$$\mu = \text{random}() = 0,3363$$

$$\nu = \text{random}() = 0,4238$$

$$\sigma = \left(\frac{\Gamma(1+1,5) \times \sin\left(\frac{\pi 1,5}{2}\right)}{\Gamma\left(\frac{1+1,5}{2}\right) \times 1,5 \times 2^{\left(\frac{1,5-1}{2}\right)}} \right) = 0,6966$$

$$P = \text{levy}(\text{Dim}) = 0,01 \cdot \frac{0,3363 \cdot 0,6966}{|0,4238|^{\frac{1}{1,5}}} = 0,0042$$

$$MX_{1,1} = 29,2269 - (74,3772 - 29,2269) \cdot 0,0042 \cdot 0,9980 = 29,0399$$

Ecuación general GOA:

$$\mu = \text{random}() = 0,5623$$

$$\nu = \text{random}() = 0,8867$$

$$\sigma = \left(\frac{\Gamma(1+1,5) \cdot \sin\left(\frac{\pi 1,5}{2}\right)}{\Gamma\left(\frac{1+1,5}{2}\right) \cdot 1,5 \times 2^{\left(\frac{1,5-1}{2}\right)}} \right) = 0,6966$$

$$P = \text{levy}(\text{Dim}) = 0,01 \cdot \frac{0,5623 \cdot 0,6966}{|0,8867|^{\frac{1}{1,5}}} = 0,0042$$

$$MX_{1,2} = 31,7354 - (44,5327 - 31,7354) \cdot 0,0042 \cdot 0,9980 = 31,6812$$

Ecuación general GOA:

$$\begin{aligned}X_1 &= [74,3772, 44,5327] \rightarrow f_{X_1} = 7515,1284 \\MX_1 &= [29,0399, 31,6812] \rightarrow f_{MX_1} = 1847,0103 \\f_{MX_1} &< f_{X_1} \\1847,0103 &< 7515,1284 \rightarrow X_1 = MX_1\end{aligned}$$

Ecuación general GOA:

$$r = \text{random}() = 0,0902$$

$$r \leq 0,5 \rightarrow 0,0902 \leq 0,5$$

$$r_6 = \text{random}() = 0,8752$$

$$L = 0,2 + (2 - 0,2) \cdot r_6 = 0,2 + (2 - 0,2) \cdot 0,8752 = 1,7754$$

$$R = \frac{M \cdot \text{vel}^2}{L} = \frac{2,5000 \cdot 1,5000^2}{1,7754} = 3,1683$$

$$\text{capturability} = \frac{1}{3,1683 \cdot 1,0020} = 0,3150$$

$$\text{capturability} \geq c \rightarrow 0,3150 \geq 0,2$$

Ecuación general GOA:

$$\delta = 0,3150 \cdot |29,2269 - 29,2269| = 0,0000$$

$$MX_{2,1} = 0,9980 \cdot 0,0000 \cdot (29,2269 - 29,2269) + 29,2269 = 29,2269$$

Ecuación general GOA:

$$\delta = 0,3150 \cdot |31,7354 - 31,7354| = 0,0000$$

$$MX_{2,2} = 0,9980 \cdot 0,0000 \cdot (31,7354 - 31,7354) + 31,7354 = 31,7354$$

Ecuación general GOA:

$$X_2 = [29,2269, 31,7354] \rightarrow f_{X_2} = 1861,3477$$

$$MX_2 = [29,2269, 31,7354] \rightarrow f_{MX_2} = 1861,3477$$

$$f_{MX_1} \geq f_{X_1}$$

$$1847,0103 \geq 7515,1284 \rightarrow X_2 \text{ se mantiene}$$

GOA: Ejemplo práctico - validación restricciones

Restricción: $x_1, x_2 \in [-100, 100]$

Soluciones obtenidas en la iteración 1:

ind 1: [29.0399, 31.6812], infactibles: 0

ind 2: [29.2269, 31.7354], infactibles: 0

Reparación de soluciones:

ind 1: [29.0399, 31.6812] / fitness: 1847.0103

ind 2: [29.2269, 31.7354] / fitness: 1861.3477

Mejor solución:

ind 1: [29.0399, 31.6812] / fitness: 1847.0103

Ecuación general GOA:

$$MX(x) = X = \begin{bmatrix} 29,0399 & 31,6812 \\ 29,2269 & 31,7354 \end{bmatrix}$$
$$t = 1 - \frac{iter}{MaxIt} = 1 - \frac{2}{500} = 0,9960$$
$$t2 = 1 + \frac{iter}{MaxIt} = 1 + \frac{2}{500} = 1,0040$$
$$randomIndex = randint(0, N - 1) = 0$$

$$X_r^t = X[randomIndex] = [29,0399, 31,6812]$$
$$X_m^t = \frac{1}{2} \sum_{i=1}^2 X_i^t = [29,1334, 31,7083]$$

Ecuación general GOA:

$$r = \text{random}() = 0,5624$$

$$r > 0,5 \rightarrow 0,5624 > 0,5$$

$$q = \text{random}() = 0,0340$$

$$q < 0,5 \rightarrow 0,0340 < 0,5$$

GOA: Ejemplo práctico - iter 2 - ind 1 - dim 1

Ecuación general GOA:

$$r_3 = \text{random}() = 0,4207$$

$$r_5 = \text{random}() = 0,2298$$

$$V(2 \cdot \pi \cdot r_3) = V(2 \cdot \pi \cdot 0,4207) = V(2,643485141584701)$$

$$x \leq \pi \rightarrow 2,6435 \leq \pi$$

$$V(x) = -\frac{1}{\pi} \cdot 2,6435 + 1 = 0,1586$$

$$b = 2 \cdot 0,1586 \cdot 0,9960 = 0,3158$$

$$B = (2 \cdot 0,2298 - 1) \cdot 0,3158 = -0,1707$$

$$v_1 = \text{random}(-b, b) = \text{random}(-0,3158, 0,3158) = 0,2189$$

$$v_2 = -0,1707 \cdot (29,0399 - 29,1334) = 0,0160$$

$$MX_{1,1} = X_i^t + v_1 + v_2 = 29,0399 + 0,2189 + 0,0160 = 29,2747$$

Ecuación general GOA:

$$r_3 = \text{random}() = 0,7121$$

$$r_5 = \text{random}() = 0,9318$$

$$V(2 \cdot \pi \cdot r_3) = V(2 \cdot \pi \cdot 0,7121) = V(4,47451152395702)$$

$$x > \pi \rightarrow 4,4745 > \pi$$

$$V(x) = -\frac{1}{\pi} \cdot 4,4745 - 1 = 0,4243$$

$$b = 2 \cdot 0,4243 \cdot 0,9960 = 0,8452$$

$$B = (2 \cdot 0,9318 - 1) \cdot 0,8452 = 0,7299$$

$$v_1 = \text{random}(-b, b) = \text{random}(-0,8452, 0,8452) = -0,5048$$

$$v_2 = 0,7299 \cdot (31,6812 - 31,7083) = -0,0198$$

$$MX_{1,1} = X_i^t + v_1 + v_2 = 31,6812 - 0,5048 - 0,0198 = 31,1566$$

Ecuación general GOA:

$$\begin{aligned}X_1 &= [29,0399, 31,6812] \rightarrow f_{X_1} = 1847,0103 \\MX_1 &= [29,2747, 31,1566] \rightarrow f_{MX_1} = 1827,7387 \\f_{MX_1} &< f_{X_1} \\1827,7387 &< 1847,0103 \rightarrow X_1 = MX_1\end{aligned}$$

Ecuación general GOA:

$$r = \text{random}() = 0,7705$$

$$r > 0,5 \rightarrow 0,7705 > 0,5$$

$$q = \text{random}() = 0,9445$$

$$q \geq 0,5 \rightarrow 0,9445 \geq 0,5$$

Ecuación general GOA:

$$r_2 = \text{random}() = 0,9945$$

$$r_4 = \text{random}() = 0,6667$$

$$a = 2 \cdot \cos(2 \cdot \pi \cdot 0,9945) \cdot 0,9960 = 1,9908$$

$$A = (2 \cdot 0,6667 - 1) \cdot 1,9908 = 0,6636$$

$$u_1 = \text{random}(-a, a) = \text{random}(-1,9908, 1,9908) = 0,4578$$

$$u_2 = 0,6636 \cdot (29,2269 - 29,2747) = -0,0317$$

$$MX_{2,1} = X_{2,1}^t + u_1 + u_2 = 29,2269 + 0,4578 - 0,0317 = 29,6531$$

Ecuación general GOA:

$$r_2 = \text{random}() = 0,0270$$

$$r_4 = \text{random}() = 0,7965$$

$$a = 2 \cdot \cos(2 \cdot \pi \cdot 0,0270) \cdot 0,9960 = 1,9634$$

$$A = (2 \cdot 0,7965 - 1) \cdot 1,9634 = 1,1642$$

$$u_1 = \text{random}(-a, a) = \text{random}(-1,9634, 1,9634) = -1,1308$$

$$u_2 = 1,1642 \cdot (31,7354 - 31,1566) = 0,6739$$

$$MX_{2,1} = X_{2,1}^t + u_1 + u_2 = 31,7354 - 1,1308 + 0,6739 = 31,2784$$

Ecuación general GOA:

$$X_2 = [29,2269, 31,7354] \rightarrow f_{X_2} = 1861,3477$$

$$MX_2 = [29,6531, 31,2784] \rightarrow f_{MX_2} = 1857,6443$$

$$f_{MX_1} < f_{X_1}$$

$$1857,6443 < 1861,3477 \rightarrow X_2 = MX_2$$

GOA: Ejemplo práctico - validación restricciones

Restricción: $x_1, x_2 \in [-100, 100]$

Soluciones obtenidas en la iteración 1:

ind 1: [29.2747, 31.1566], infactibles: 0

ind 2: [29.6531, 31.2784], infactibles: 0

Reparación de soluciones:

ind 1: [29.2747, 31.1566] / fitness: 1827.7387

ind 2: [29.6531, 31.2784] / fitness: 1857.6443

Mejor solución:

ind 1: [29.2747, 31.1566] / fitness: 1827.7387

Ecuación general GOA:

$$MX(x) = X = \begin{bmatrix} -0,0024 & 0,0026 \\ -0,0018 & 0,001 \end{bmatrix}$$

$$t = 1 - \frac{iter}{MaxIt} = 1 - \frac{500}{500} = 0,0000$$

$$t2 = 1 + \frac{iter}{MaxIt} = 1 + \frac{2}{500} = 2,0000$$

$$randomIndex = randint(0, N - 1) = 0$$

$$X_r^t = X[randomIndex] = [-0,0024, 0,0026]$$

$$X_m^t = \frac{1}{2} \sum_{i=1}^2 X_i^t = [-0,0021, 0,0018]$$

Ecuación general GOA:

$$r = \text{random}() = 0,8373$$

$$r > 0,5 \rightarrow 0,8373 > 0,5$$

$$q = \text{random}() = 0,3772$$

$$q < 0,5 \rightarrow 0,3772 < 0,5$$

Ecuación general GOA:

$$r_3 = \text{random}() = 0,3630$$

$$r_5 = \text{random}() = 0,4736$$

$$V(2 \cdot \pi \cdot r_3) = V(2 \cdot \pi \cdot 0,3630) = V(2,280670145193084)$$

$$x \leq \pi \rightarrow 2,2807 \leq \pi$$

$$V(x) = -\frac{1}{\pi} \cdot 2,2807 + 1 = 0,2740$$

$$b = 2 \cdot 0,2740 \cdot 0,0000 = 0,0000$$

$$B = (2 \cdot 0,4736 - 1) \cdot 0,0000 = -0,0000$$

$$v_1 = \text{random}(-b, b) = \text{random}(-0,0000, 0,0000) = 0,0000$$

$$v_2 = -0,0000 \cdot (-0,0024 + 0,0021) = 0,0000$$

$$MX_{1,1} = X_i^t + v_1 + v_2 = -0,0024 + 0,0000 + 0,0000 = -0,0024$$

Ecuación general GOA:

$$r_3 = \text{random}() = 0,2384$$

$$r_5 = \text{random}() = 0,9012$$

$$V(2 \cdot \pi \cdot r_3) = V(2 \cdot \pi \cdot 0,2384) = V(1,4981150821851663)$$

$$x \leq \pi \rightarrow 1,4981 \leq \pi$$

$$V(x) = -\frac{1}{\pi} \cdot 1,4981 + 1 = 0,5231$$

$$b = 2 \cdot 0,5231 \cdot 0,0000 = 0,0000$$

$$B = (2 \cdot 0,9012 - 1) \cdot 0,0000 = 0,0000$$

$$v_1 = \text{random}(-b, b) = \text{random}(-0,0000, 0,0000) = 0,0000$$

$$v_2 = -0,0000 \cdot (0,0026 - 0,0018) = 0,0000$$

$$MX_{1,2} = X_i^t + v_1 + v_2 = 0,0026 + 0,0000 + 0,0000 = 0,0026$$

Ecuación general GOA:

$$\begin{aligned}X_1 &= [-0,0024, 0,0026] \rightarrow f_{X_1} = 0,0000 \\MX_1 &= [-0,0024, 0,0026] \rightarrow f_{MX_1} = 0,0000 \\f_{MX_1} &\geq f_{X_1} \\0,0000 &\geq 0,0000 \rightarrow X_1 \text{ se mantiene}\end{aligned}$$

Ecuación general GOA:

$$r = \text{random}() = 0,0942$$

$$r \leq 0,5 \rightarrow 0,0942 \leq 0,5$$

$$r_6 = \text{random}() = 0,4555$$

$$L = 0,2 + (2 - 0,2) \cdot r_6 = 0,2 + (2 - 0,2) \cdot 0,4555 = 1,0200$$

$$R = \frac{M \cdot vel^2}{L} = \frac{2,5000 \cdot 1,5000^2}{1,0200} = 5,5149$$

$$capturability = \frac{1}{5,5149 \cdot 2,0000} = 0,0907$$

$$capturability < c \rightarrow 0,0907 < 0,2$$

Ecuación general GOA:

$$\mu = \text{random}() = 0,4564$$

$$\nu = \text{random}() = 0,5737$$

$$\sigma = \left(\frac{\Gamma(1+1,5) \cdot \sin\left(\frac{\pi 1,5}{2}\right)}{\Gamma\left(\frac{1+1,5}{2}\right) \cdot 1,5 \times 2^{\left(\frac{1,5-1}{2}\right)}} \right) = 0,6966$$

$$P = \text{levy}(\text{Dim}) = 0,01 \times \frac{0,4564 \times 0,6966}{|0,5737|^{\frac{1}{1,5}}} = 0,0046$$

$$MX_{2,1} = -0,0018 - (-0,0018 - -0,0018) \cdot 0,0046 \cdot 0,0000 = -0,0018$$

Ecuación general GOA:

$$\mu = \text{random}() = 0,6519$$

$$\nu = \text{random}() = 0,6795$$

$$\sigma = \left(\frac{\Gamma(1+1,5) \cdot \sin\left(\frac{\pi 1,5}{2}\right)}{\Gamma\left(\frac{1+1,5}{2}\right) \cdot 1,5 \times 2^{\left(\frac{1,5-1}{2}\right)}} \right) = 0,6966$$

$$P = \text{levy}(\text{Dim}) = 0,01 \cdot \frac{0,6519 \cdot 0,6966}{|0,6795|^{\frac{1}{1,5}}} = 0,0059$$

$$MX_{2,2} = 0,0010 - (0,0010 - 0,0010) \cdot 0,0059 \cdot 0,0000 = 0,0010$$

Ecuación general GOA:

$$\begin{aligned}X_2 &= [-0,0018, 0,001] \rightarrow f_{X_2} = 0,0000 \\MX_2 &= [29,0399, 31,6812] \rightarrow f_{MX_2} = 0,0000 \\f_{MX_2} &\geq f_{X_2} \\0,0000 &\geq 0,0000 \rightarrow X_2 \text{ se mantiene}\end{aligned}$$

GOA: Ejemplo práctico - validación restricciones

Restricción: $x_1, x_2 \in [-100, 100]$

Soluciones obtenidas en la iteración 500:

ind 1: $[-0.0024, 0.0026]$, infactibles: 0

ind 2: $[-0.0018, 0.001]$, infactibles: 0

Reparación de soluciones:

ind 1: $[-0.0024, 0.0026]$ / fitness: 0.0000

ind 2: $[-0.0018, 0.001]$ / fitness: 0.0000

Mejor solución:

ind 2: $[-0.0018, 0.001]$ / fitness: 0.0000