

Tasmanian Devil Optimization

Dr. Broderick Crawford Labrín

Pontificia Universidad Católica de Valparaíso

Tasmanian Devil Optimization (TDO)

- Fue desarrollada por Mohammad Dehghani y Pavel Trojovský et. al. en el año 2015 ^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 *Tasmanian Devil Optimization: A New Bio-Inspired Optimization Algorithm for Solving Optimization Algorithm*, IEEE (2022)

- Ecuaciones de movimientos general

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

$$F = \begin{bmatrix} F(x_1) \\ F(x_i) \\ F(x_N) \end{bmatrix} \quad (2)$$

- Donde:

- X es la poblacion de demonios de tasmania con:
 - X_i es i -ésimo candidato
- F un vector con los valores de la función objetivo

TDO: Ecuaciones de movimiento

- Ecuaciones de movimientos general

$$CP_i = X_k, i = 1, 2, \dots, N, k \in 1, 2, \dots, N | k \neq i \quad (3)$$

$$X_{i,j}^{new,S2} = \begin{cases} x_{i,j} + r \cdot (CP_{i,j} - l \cdot x_{i,j}), F_{cpi} < F_i \\ x_{i,j} + r \cdot (x_{i,j} - CP_{i,j}), otherwise \end{cases} \quad (4)$$

$$X_i = \begin{cases} X_{i,j}^{new,S2}, F_i^{new,S2} < F_i \\ X_i, otherwise \end{cases} \quad (5)$$

- Donde:

- CP_i es el proceso de selección de la carroña y presas
- $X_{i,j}^{new,S2}$ analiza el valor de la presa
 - r es un numero aleatorio de 0 a 1
 - l elige 1 o 2 aleatoriamente
- X_i calcula la nueva posición

TDO: Ecuaciones de movimiento

- Ecuaciones de movimientos general

$$R = 0,01 \cdot \left(1 - \frac{iter}{maxIter}\right) \quad (6)$$

$$X_{i,j}^{new,S2} = x_{i,j} + (2r - 1) \cdot R \cdot x_{i,j} \quad (7)$$

$$X_i = \left\{ \begin{array}{l} X_{i,j}^{new}, F_i^{new} < F_i \\ X_i, otherwise \end{array} \right\} \quad (8)$$

- Donde:

- R indica el rango en el que el demonio de Tasmania sigue a la presa
- $X_{i,j}^{new,S2}$ calcula una nueva posición basada en el rango R
- X_i actualiza la posición del demonio de tasmania

Algorithm 1 Tasmanian Devil Optimization

```
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 population X  
2: for it = 1 to MaxIt do  
3:     calculate the fitness of each X  
4:     select Best  
5:     for i=0 to N do  
6:         k = Select a random element from the array after removing the element at index 'i'  
7:         CPi = X[k]  
8:         Xnew = X[i]  
9:         for j=0 to dim do  
10:            if f(Ci) is better than f(X[i])  
11:                I = randint(1,2)  
12:                Xnew[j] = X[i,j] + random() * (Ci[j] - I * X[i,j])  
13:            else  
14:                Xnew[j] = X[i,j] + random() * (X[i,j] - Ci[j])  
15:            if f(Xnew) is better than f(X[i]): X[i] = Xnew  
16:        r = random()  
17:        if r  $\geq 0.5$   
18:            R = 0.01 * (1 - (it / maxIter))  
19:            for j=0 to dim do  
20:                Xnew[j] = X[i,j] + (2 * random() - 1) * R * Xnew[j]  
21:            if f(Xnew) is better than f(X[i]): X[i] = Xnew  
22: return updated population X' and Best
```

TDO: 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 TDO:

- Tamaño de la población: 2 individuos
- Número máximo de iteraciones: 100 iteraciones

TDO: Ejemplo práctico - soluciones iniciales

Soluciones iniciales:

ind 1: [-44.0676, -26.1725] / fitness: 2626.9553

ind 2: [-51.3111, 72.8825] / fitness: 7944.6912

Mejor solución:

ind 1: [-44.0676, -26.1725] / fitness: 2626.9553

TDO: Ejemplo práctico - iter 1 - ind 1

Ecuaciones generales:

$$arange = arange(N) = arange(2) = [1, 2]$$

$$arange = delete(arange, i) = delete(arange, 1) = [2]$$

$$k = randomChoice(arange) = 2$$

$$CP_i = X_k = X_2 = [-51,3111, 72,8825]$$

$$X^{new} = X_i = X_1 : [-44,0676, -26,1725]$$

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

$$f(CP_i) \geq f(x_1) \rightarrow 7944,6912 \geq 2626,9553$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (x_{i,j} - CP_{i,j})$$

$$X_{1,1}^{new,S2} = -44,0676 + 0,6615 \cdot (-44,0676 + 51,3111) = -39,2764$$

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

$$f(CP_i) \geq f(x_1) \rightarrow 7944,6912 \geq 2626,9553$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (x_{i,j} - CP_{i,j})$$

$$X_{1,1}^{new,S2} = -26,1725 + 0,4113 \cdot (-26,1725 - 72,8825) = -66,9113$$

TDO: Ejemplo práctico - iter 1 - ind 1

$X_1 = [-44.0676, -26.1725] \rightarrow F_{X_1} = 2626,9553$
 $X^{new} = [-39.2764, -66.9113] \rightarrow F_{X^{new}} = 6019,7616$
 $F_{X^{new}} \geq F_{X_1}$
 $6019,7616 \geq 2626,9553 \rightarrow X_1$ se mantiene
 $r = random() = 0,9863$

Entra al segundo "for"

$r \geq 0,5 \rightarrow 0,9863 \geq 0,5$

$R = 0,01 \cdot \left(1 - \frac{iter}{maxIter}\right) = 0,01 \cdot \left(1 - \frac{1}{100}\right) = 0,0099$

$$X_{i,j}^{new,S2} = x_{i,j} + (2r - 1) \cdot R \cdot x_{i,j}$$

$$X_{1,1}^{new,S2} = -44,0676 + (2 \cdot 0,9922 - 1) \cdot 0,0099 \cdot -39,2764 = -44,4504$$

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

$$X_{i,j}^{new,S2} = x_{i,j} + (2r - 1) \cdot R \cdot x_{i,j}$$

$$X_{1,2}^{new,S2} = -26,1725 + (2 \cdot 0,4993 - 1) \cdot 0,0099 \cdot -66,9113 = -26,1716$$

TDO: Ejemplo práctico - iter 1 - ind 1

$X_1 = [-44.0676, -26.1725] \rightarrow F_{X_1} = 2626,9553$
 $X^{new} = [-44.4504, -26.1716] \rightarrow F_{X^{new}} = 2660,7919$
 $F_{X^{new}} \geq F_{X_1}$
 $2660,7919 \geq 2626,9553 \rightarrow X_1$ se mantiene

Ecuaciones generales:

$$arange = arange(N) = arange(2) = [12]$$

$$arange = delete(arange, i) = delete(arange, 1) = [1]$$

$$k = randomChoice(arange) = 1$$

$$CP_i = X_k = X_1 = [-44.0676, -26.1725]$$

$$X^{new} = X_i = X_2 : [-51.3111, 72.8825]$$

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

$$f(CP_i) < f(x_2) \rightarrow 2626,9553 < 7944,6912$$

$$l = \text{randint}(1, 2) = 1$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (CP_{i,j} - l \cdot x_{i,j})$$

$$X_{2,1}^{new,S2} = -51,3111 + 0,5949 \cdot (-44,0676 - 1 \cdot -51,3111) = -47,0023$$

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

$$f(CP_i) < f(x_2) \rightarrow 2626,9553 < 7944,6912$$

$$l = \text{randint}(1, 2) = 2$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (CP_{i,j} - l \cdot x_{i,j})$$

$$X_{2,2}^{new,S2} = 72,8825 + 0,9342 \cdot (-26,1725 - 2 \cdot 72,8825) = -87,7491$$

TDO: Ejemplo práctico - iter 1 - ind 2

$X_2 = [-51.3111, 72.8825] \rightarrow F_{X_2} = 7944,6912$
 $X^{new} = [-47.0023, -87.7491] \rightarrow F_{X^{new}} = 9909,1099$
 $F_{X^{new}} \geq F_{X_1}$
 $9909,1099 \geq 7944,6912 \rightarrow X_2$ se mantiene
 $r = random() = 0,0395$

TDO: Ejemplo práctico - validación restricciones

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

Soluciones obtenidas en la iteración 1:

ind 1: $[-44.0676, -26.1725]$, infactibles: 0

ind 2: $[-51.3111, 72.8825]$, infactibles: 0

Reparación de soluciones:

ind 1: $[-44.0676, -26.1725]$ / fitness: 2626.9553

ind 2: $[-51.3111, 72.8825]$ / fitness: 7944.6912

Mejor solución:

ind 1: $[-44.0676, -26.1725]$ / fitness: 2626.9553

TDO: Ejemplo práctico - iter 2 - ind 1

Ecuaciones generales:

$$arange = arange(N) = arange(2) = [1, 2]$$

$$arange = delete(arange, i) = delete(arange, 1) = [2]$$

$$k = randomChoice(arange) = 2$$

$$CP_i = X_k = X_2 = [-51.3111, 72.8825]$$

$$X^{new} = X_i = X_1 = [-44.0676, -26.1725]$$

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

$$f(CP_i) \geq f(x_1) \rightarrow 7944,6912 \geq 2626,9553$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (x_{i,j} - CP_{i,j})$$

$$X_{1,1}^{new,S2} = -44,0676 + 0,9411 \cdot (-44,0676 + 51,3111) = -37,2511$$

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

$$f(CP_i) \geq f(x_1) \rightarrow 7944,6912 \geq 2626,9553$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (x_{i,j} - CP_{i,j})$$

$$X_{1,2}^{new,S2} = -26,1725 + 0,1789 \cdot (-26,1725 - 72,8825) = -43,8913$$

TDO: Ejemplo práctico - iter 2 - ind 1

$X_1 = [-44.0676, -26.1725] \rightarrow F_{X_1} = 2626,9553$
 $X^{new} = [-37.2511, -43.8913] \rightarrow F_{X^{new}} = 3314,0885$
 $F_{X^{new}} \geq F_{X_1}$
 $3314,0885 \geq 2626,9553 \rightarrow X_1$ se mantiene
 $r = random() = 0,3724$

TDO: Ejemplo práctico - iter 2 - ind 2

Ecuaciones generales:

$$arange = arange(N) = arange(2) = [1, 2]$$

$$arange = delete(arange, i) = delete(arange, 1) = [1]$$

$$k = randomChoice(arange) = 1$$

$$CP_2 = X_k = X_1 = [-44.0676, -26.1725]$$

$$X^{new} = X_2 = X_2 : [-51.3111, 72.8825]$$

Entra al segundo "for"

$$r \geq 0,5 \rightarrow 0,7013 \geq 0,5$$

$$R = 0,01 \cdot \left(1 - \frac{iter}{maxIter}\right) = 0,01 \cdot \left(1 - \frac{2}{100}\right) = 0,0098$$

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

$$f(CP_i) < f(x_2) \rightarrow 2626,9553 < 7944,6912$$

$$l = \text{randint}(1, 2) = 1$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (CP_{i,j} - l \cdot x_{i,j})$$

$$X_{2,2}^{new,S2} = -51,3111 + 0,0811 \cdot (-44,0676 - 1 \cdot -51,3111) = -50,7240$$

TDO: Ejemplo práctico - iter 2 - ind 2 - dim 2

$$f(CP_i) < f(x_2) \rightarrow 2626,9553 < 7944,6912$$

$$l = \text{randint}(1, 2) = 1$$

$$X_{i,j}^{\text{new},S2} = x_{i,j} + r \cdot (CP_{i,j} - l \cdot x_{i,j})$$

$$X_{2,2}^{\text{new},S2} = 72,8825 + 0,6265 \cdot (-26,1725 - 1 \cdot 72,8825) = 10,8277$$

TDO: Ejemplo práctico - iter 2 - ind 2

$$X_2 = [-51.3111, 72.8825] \rightarrow F_{X_2} = 7944,6912$$

$$X^{new} = [-50.724, 10.8277] \rightarrow F_{X^{new}} = 2690,1593$$

$$F_{X^{new}} \geq F_{X_1}$$

$$2690,1593 \geq 7944,6912 \rightarrow X_2 = X^{new}$$

$$r = \text{random}() = 0,7012$$

Entra al segundo "for"

$$r \geq 0,5 \rightarrow 0,7012 \geq 0,5$$

$$R = 0,01 \cdot \left(1 - \frac{\text{iter}}{\text{maxIter}}\right) = 0,01 \cdot \left(1 - \frac{2}{100}\right) = 0,0098$$

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

$$X_{i,j}^{new,S2} = x_{i,j} + (2r - 1) \cdot R \cdot x_{i,j}$$

$$X_{2,1}^{new,S2} = -50,7240 + (2 \cdot 0,1083 - 1) \cdot 0,0098 \cdot -50,7240 = -50,3345$$

TDO: Ejemplo práctico - iter 2 - ind 2 - dim 2

$$X_{i,j}^{new,S2} = x_{i,j} + (2r - 1) \cdot R \cdot x_{i,j}$$

$$X_{2,2}^{new,S2} = 10,8277 + (2 \cdot 0,3406 - 1) \cdot 0,0098 \cdot 10,8277 = 10,7939$$

$$X_2 = [-50.724, 10.8277] \rightarrow F_{X_2} = 7944,6912$$

$$X^{new} = [-50.3345, 10.7939] \rightarrow F_{X^{new}} = 2650,0722$$

$$F_{X^{new}} \geq F_{X_2}$$

$$2650,0722 < 7944,6912 \rightarrow X_2 = X^{new}$$

TDO: Ejemplo práctico - validación restricciones

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

Soluciones obtenidas en la iteración 2:

ind 1: $[-44.0676, -26.1725]$, infactibles: 0

ind 2: $[-50.3345, 10.7939]$, infactibles: 0

Reparación de soluciones:

ind 1: $[-44.0676, -26.1725]$ / fitness: 2626.9553

ind 2: $[-50.3345, 10.7939]$ / fitness: 2650.0722

Mejor solución:

ind 1: $[-44.0676, -26.1725]$ / fitness: 2626.9553

Ecuaciones generales:

$$arange = arange(N) = arange(2) = [12]$$
$$arange = delete(arange, i) = delete(arange, 1) = [2]$$
$$k = randomChoice(arange) = 2$$
$$CP_i = X_k = X_2 = [-9.8719e-16, -2.7150e-15]$$
$$X^{new} = X_i = X_1 = [-9.7995e-16, -4.6953e-15]$$

TDO: Ejemplo práctico - iter 100 - ind 1 - dim 1

$$f(CP_i) < f(x_1) \rightarrow 0,0000 < 0,0000$$

$$l = \text{randint}(1, 2) = 2$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (CP_{i,j} - l \cdot x_{i,j})$$

$$X_{1,1}^{new,S2} = 0,0000 + 0,2782 \cdot (-0,0000 - 2 \cdot -0,0000) = 0,0000$$

TDO: Ejemplo práctico - iter 100 - ind 1 - dim 2

$$f(CP_i) < f(x_1) \rightarrow 0,0000 < 0,0000$$

$$l = \text{randint}(1, 2) = 1$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (CP_{i,j} - l \cdot x_{i,j})$$

$$X_{1,1}^{new,S2} = 0,0000 + 0,7843 \cdot (0,0000 - 1 \cdot -0,0000) = 0,0000$$

TDO: Ejemplo práctico - iter 100 - ind 1

Aclaración: En la comparación de $F_{X^{new}}$ y F_{X_1} fueron cortados decimales

$$X_1 = [-9,7995e - 16, -4,6953e - 15] \rightarrow F_{X_1} = 0,0000$$

$$X^{new} = [-7,0932e - 16, -3,1420e - 15] \rightarrow F_{X^{new}} = 0,0000$$

$$F_{X^{new}} \geq F_{X_1}$$

$$0,0000 < 0,0000 \rightarrow X_1 = X^{new}$$

$$r = \text{random}() = 0,1740$$

Ecuaciones generales:

$$arange = arange(N) = arange(2) = [12]$$

$$arange = delete(arange, i) = delete(arange, 1) = [1]$$

$$k = randomChoice(arange) = 1$$

$$CP_i = X_k = X_1 = [-7.0932e-16, -3.1420e-15]$$

$$X^{new} = X_i = X_2 = [-9.8719e-16, -2.7150e-15]$$

$$f(CP_i) \geq f(x_1) \rightarrow 0,0000 \geq 0,0000$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (x_{i,j} - CP_{i,j})$$

$$X_{2,1}^{new,S2} = -0,0000 + 0,2460 \cdot (-0,0000 + 0,0000) = -0,0000$$

$$f(CP_i) \geq f(x_1) \rightarrow 0,0000 \geq 0,0000$$

$$X_{i,j}^{new,S2} = x_{i,j} + r \cdot (x_{i,j} - CP_{i,j})$$

$$X_{2,1}^{new,S2} = -0,0000 + 0,9970 \cdot (-0,0000 + 0,0000) = -0,0000$$

TDO: Ejemplo práctico - iter 100 - ind 2

$$X_2 = [-9.8719\text{e-}16, -2.7150\text{e-}15] \rightarrow F_{X_2} = 0,0000$$

$$X^{new} = [-39.2764, -66.9113] \rightarrow F_{X^{new}} = 0,0000$$

$$F_{X^{new}} \geq F_{X_1}$$

$$0,0000 < 0,0000 \rightarrow X_2 = X^{new}$$

$$r = \text{random}() = 0,5106$$

Entra al segundo "for"

$$r \geq 0,5 \rightarrow 0,5107 \geq 0,5$$

$$R = 0,01 \cdot \left(1 - \frac{\text{iter}}{\text{maxIter}}\right) = 0,01 \cdot \left(1 - \frac{100}{100}\right) = 0,0000$$

$$X_{i,j}^{new,S2} = x_{i,j} + (2r - 1) \cdot R \cdot x_{i,j}$$

$$X_{2,1}^{new,S2} = -0,0000 + (2 \cdot 0,1554 - 1) \cdot 0,0000 \cdot -0,0000 = -0,0000$$

$$X_{i,j}^{new,S2} = x_{i,j} + (2r - 1) \cdot R \cdot x_{i,j}$$

$$X_{2,2}^{new,S2} = -0,0000 + (2 \cdot 0,1674 - 1) \cdot 0,0000 \cdot -0,0000 = -0,0000$$

TDO: Ejemplo práctico - iter 100 - ind 2

$$X_2 = [-1.0555\text{e-}15, -2.2892\text{e-}15] \rightarrow F_{X_2} = 0,0000$$

$$X^{new} = [-1.0555\text{e-}15, -2.2892\text{e-}15] \rightarrow F_{X^{new}} = 0,0000$$

$$F_{X^{new}} < F_{X_1}$$

$$2660,7919 < 2626,9553 \rightarrow X_2 = X^{new}$$

TDO: Ejemplo práctico - validación restricciones

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

Soluciones obtenidas en la iteración 100:

ind 1: $[-7.0932\text{e-}16, -3.1420\text{e-}15]$, infactibles: 0

ind 2: $[-1.0555\text{e-}15, -2.2892\text{e-}15]$, infactibles: 0

Reparación de soluciones:

ind 1: $[-7.0932\text{e-}16, -3.1420\text{e-}15]$ / fitness: 0.0000

ind 2: $[-1.0555\text{e-}15, -2.2892\text{e-}15]$ / fitness: 0.0000

Mejor solución:

ind 2: $[-1.0555\text{e-}15, -2.2892\text{e-}15]$ / fitness: 0.0000