UNIVERSIDAD AUTÓNOMA DE YUCATÁN
FACULTAD DE INGENIERÍA

Tarea 3 de física computacional

Optimización usando evolución diferencial.

- Uno de los agoritmos evolutivos usado frecuentemente en optimización es el método de evolución diferencial. Supongamos una función $f : A \subset \mathbb{R}^n \to \mathbb{R}$ y supongamos que el espacio de búsqueda es un hiper-rectangulo cuyos límites inferiores y superiores están dados por los vectores $\mathbf{b}_L$ y $\mathbf{b}_U$ respectivamente. Denotemos por $\mathbf{X}_0$ la matriz cuyos renglones contienen a los individuos en la generación cero, es decir, $\mathbf{X}_0$ es la población inicial. Denotemos por $\mathbf{x}_{i;0}$ el renglón $i$ésimo de $\mathbf{X}_0$, digamos que la poblacion inicial tiene m individuos, entonces i va de uno a m. Denotemos por $x^j_{i;0}$ la componente j de $\mathbf{x}_{i;0}$, j va de uno a n. La población inicial se puede construir como sigue.

$$x^j_{i;0} = rand^j(0,1) \cdot (b^j_U - b^j_L) + b^j_L \tag{1}$$

con $b^j_{U,L}$ la componente j de $\mathbf{b}_{U,L}$ y $rand^j(0,1)$ un número aleatorion en (0,1). El superíndice j indica que un número aleatorio debe generarse para cada j.

La mutación en la generación g, se realiza como sigue

$$\mathbf{v}_{i;g} = \mathbf{x}_{r_0;g} + F \cdot (\mathbf{x}_{r_1;g} - \mathbf{x}_{r_2;g}). \tag{2}$$

A $\mathbf{x}_{r_0;g}$ se le llama vector base el cual se perturba con la diferencia de otros dos vectores. r_0, r_1, $r_2 \in \{1, 2, ...m\}$, $r_1 \neq r_2 \neq r_3 \neq i$. F es el “factor de escalamiento de la mutación”, en general es un número mayor que cero pero si se escoge como un aleatorio en (0,1) la variante del metodo se llama “dither”.

El crossover (la cruza) en la generación g se realiza por

$$\mathbf{u}_{i;g} = u^j_{i;g} = \begin{cases} v^j_{i;g} & \text{si } rand^j(0,1) \leqslant Cr \text{ or } j = j_{\text{rand}} \\ & \\ x^j_{i;g} & \text{en otro caso.} \end{cases} \tag{3}$$

Con C_r la probabilidad de cruza, definida por el usuario, $j_{\text{rand}} \in [1, n]$. Finalmente, la selección se realiza por

$$\mathbf{x}_{i;g+1} = \begin{cases} \mathbf{u}_{i;g} & \text{si } f(\mathbf{u}_{i;g}) \leqslant f(\mathbf{x}_{i;g}) \\ \mathbf{x}_{i;g} & \text{en otro caso} \end{cases} \tag{4}$$

Definiendo operadores para la generacion de población inicial –$\mathbf{XP}_0(m, \mathbf{b}_L, \mathbf{b}_U)$, para la mutación –$\mathbf{Vmut}(X, F, bL, bU)$, para la cruza –$\mathbf{Ucruza}(X, V, Cr)$– y para la selección –$\mathbf{S}(fob, X, U)$; siendo fob la funcion a optimizar, el algoritmo de evolución diferencial puede efectuarse como se muestra en Algoritmo 1, donde $\mathbf{B}(X, fob)$ es un operador que extrae el mejor individuo $\mathbf{x}_{best}$ de la población X.

A) Implemente el método de evolución diferencial en Fortran.

Algoritmo 1 Evolución Diferencial

$X \leftarrow \mathbf{XP}_0(m, \mathbf{b}_L, \mathbf{b}_U)$
while (no se cumpla el criterio de paro) **do**
  $V \leftarrow \mathbf{Vmut}(X, F, bL, bU)$
  $U \leftarrow \mathbf{Ucruza}(X, V, Cr)$
  $X \leftarrow \mathbf{S}(fob, X, U)$
end while
$\mathbf{x}_{best} \leftarrow \mathbf{B}(X, fob)$

B) Optimice usando evolución diferencial, las funciones de la tabla 1.

Tabla 1: Funciones de prueba. $\mathbf{a} \cdot \mathbf{b}$ es el producto escalar, de $\mathbf{a}$ con $\mathbf{b}$, D es la dimensionalidad del problema—dimensión de $\mathbf{x}$.

Ejemplo	Function objetivo	Rango de búsqueda	Vaor óptimo	D	Error aceptable
Cosine Mixture	$f_1(\mathbf{x}) = \mathbf{x}\cdot\mathbf{x} - 0.1 \sum_{i=1}^{D} \cos(5\pi x_i)$	$[-1, 1]$	$-0.1D$	30	10^{-5}
Griewank	$f_2(\mathbf{x}) = 1 + \mathbf{x}\cdot\mathbf{x}/4000 - \prod_{i=1}^{D} \cos(x_i/\sqrt{i})$	$[-600, 600]$	0	30	10^{-5}
Inverted cosine wave	$f_3(\mathbf{x}) = -\sum_{i=1}^{D-1} \exp[-(x_i^2 + x_{i+1}^2 + 0.5\, x_i\, x_{i+1})/8] \times \cos\left(4\sqrt{x_i^2 + x_{i+1}^2 + 0.5\, x_i\, x_{i+1}}\right)$	$[-5, 5]$	$1 - D$	10	10^{-5}
Michalewicz	$f_4(\mathbf{x}) = -\sum_{i=1}^{D} \sin x_i \, \sin^{20}(i\, x_i^2/\pi)$	$[0, \pi]$	-9.660151715641	10	10^{-5}
Rastrigin	$f_5(\mathbf{x}) = 10D + \mathbf{x}\cdot\mathbf{x} - 10 \sum_{i=1}^{D} \cos(2\,\pi\, x_i)$	$[-5.12, 5.12]$	0	30	10^{-5}
Rosenbrock	$f_6(\mathbf{x}) = \sum_{i=1}^{D-1} (x_i - 1)^2 + 100(x_{i+1} - x_i^2)^2$	$[-30, 30]$	0	30	10^{-2}
Shubert	$f_7(\mathbf{x}) = \sum_{i=1}^{5} i \cos[(i+1)x_1 + i] \times \sum_{i=1}^{5} i \cos[(i+1)x_2 + i]$	$[-10, 10]$	-186.7309088310	2	10^{-5}
Sinusoidal problem	$f_8(\mathbf{x}) = -A \prod_{i=1}^{D} \sin(x_i - z) - \prod_{i=1}^{D} \sin[B(x_i - z)]$, $A = 2.5,\ B = 5,\ z = 30$	$[0, 180]$	$-(A+1)$	10	10^{-2}
Sphere	$f_9(\mathbf{x}) = \mathbf{x}\cdot\mathbf{x}$	$[-5.12, 5.12]$	0	10	10^{-5}
Zakharov	$f_{10}(\mathbf{x}) = \mathbf{x}\cdot\mathbf{x} + \left(\sum_{i=1}^{D} 0.5\, i\, x_i\right)^2 + \left(\sum_{i=1}^{D} 0.5\, i\, x_i\right)^4$	$[-5, 10]$	0	10	10^{-5}