

TP SCILAB/MATLAB : la méthode PSO

La méthode Particle Swarm Optimization (ou PSO) est une méthode d'optimisation sans gradient, de type évolutionnaire faisant évoluer une population (ou essaim) d'éléments de \mathbb{R}^n (appelés aussi particules) en adaptant leur vitesse de manière collective afin de minimiser une fonction $J : \mathbb{R}^n \rightarrow \mathbb{R}$. Pour tout $1 \leq i \leq N_{pop}$ et pour une génération **gen** donnée, on note

- (i) x_i : la position actuelle de la i -ème particule de la population,
- (ii) v_i la vitesse de la i -ème particule,
- (iii) p_i : la meilleure position de la i -ème particule jamais trouvée (c'est à dire où J était minimale),
- (iv) p_g : la meilleure position jamais trouvée par une particule de l'essaim.

L'essaim est initialisé de manière aléatoire. De la génération **gen** à la génération **gen+1**, on effectue les opérations suivantes :

$$v_i := wv_i + c_1\phi_1 \otimes (p_i - x_i) + c_2\phi_2 \otimes (p_g - x_i)$$

et

$$x_i := x_i + v_i$$

où ϕ_1 et ϕ_2 sont deux vecteurs aléatoires suivant une loi uniforme de $[0, 1]^n$. Les constantes positives w , c_1 et c_2 sont respectivement appelées coefficient d'inertie et d'accélération. Le symbole \otimes signifie une multiplication terme à terme. Une vitesse maximale V_{max} est également imposée pour toute particule.

1. Programmer avec Scilab la méthode PSO précédente.
2. Appliquer la méthode PSO à l'exemple simple $n = 1$, $J(x) = x^2$ et avec $w = c_1 = c_2 = 0.7$. Tracer les trajectoires correspondantes des particules sur la parabole.
3. Appliquer la méthode PSO au cas de la fonction de Rastrigin sur $[-5, 5]^2$ c'est à dire

$$J(x_1, x_2) = \sum_{i=1}^2 (x_i^2 - \cos(2\pi x_i)) + 2$$

avec 10 particles. Qu'observe t-on ?