



Le but de ce travail est de permettre à chaque groupe d'étudiant de programmer une métaheuristique qu'ils ont tiré au hasard.

Travail demandé :

- Coder l'algorithme proposé en C++ en se respectant le squelette imposée:
à télécharger sur ma page : <http://www.mage.fst.uha.fr/idoumghar/algorithm.h>

-Chaque algorithme doit être validé sur les benchmarks suivant :

Bent Cigar Function

$$f_1(\mathbf{x}) = x_1^2 + 10^6 \sum_{i=2}^D x_i^2 \quad (1)$$

Discus Function

$$f_2(\mathbf{x}) = 10^6 x_1^2 + \sum_{i=2}^D x_i^2 \quad (2)$$

Weierstrass Function

$$f_3(\mathbf{x}) = \sum_{i=1}^D \left(\sum_{k=0}^{k_{\max}} [a^k \cos(2\pi b^k (x_i + 0.5))] \right) - D \sum_{k=0}^{k_{\max}} [a^k \cos(2\pi b^k \cdot 0.5)] \quad (3)$$

where $a=0.5$, $b=3$, and $k_{\max}=20$.

Katsuura Function

$$f_5(\mathbf{x}) = \frac{10}{D^2} \prod_{i=1}^D \left(1 + i \sum_{j=1}^{32} \frac{|2^j x_i - \text{round}(2^j x_i)|}{2^j} \right)^{\frac{10}{D^{1.2}}} - \frac{10}{D^2} \quad (4)$$

HappyCat Function

$$f_6(\mathbf{x}) = \left| \sum_{i=1}^D x_i^2 - D \right|^{1/4} + (0.5 \sum_{i=1}^D x_i^2 + \sum_{i=1}^D x_i) / D + 0.5 \quad (5)$$

HGBat Function

$$f_7(\mathbf{x}) = \left| \left(\sum_{i=1}^D x_i^2 \right)^2 - \left(\sum_{i=1}^D x_i \right)^2 \right|^{1/2} + (0.5 \sum_{i=1}^D x_i^2 + \sum_{i=1}^D x_i) / D + 0.5 \quad (6)$$



- Pour la validation :
 - Nombre d'exécutions : 30
 - Dimension du problème : 20
 - Nombre d'individus dans la population : 40
 - Nombre total d'appel de la fonction objectif : 2×10^6
 - Calculer la moyenne et l'écart type.
 - Comparer avec un algorithme de la littérature.
- Faire un rapport avec une description détaillée de l'algo, diagrammes, etc.
- Faire une présentation PowerPoint (Attention au plagiat qui sera vérifié par un logiciel dédié)
- **Date limite d'envoi du projet (code + rapport + ppt) est :**
 - **Exactement une semaine avant le dernier cours**



- **Soutenance lors de la dernière séance du cours**

Ordre de passage (20min d'intervalle) :

- Imperialist Competitive Algorithm (ICA)
- Harmony Search Algorithm (HSA)
- Differential Evolution Algorithm (DEA)
- Backtracking Search optimization Algorithm (BSA)
- Particle Swarm Optimization (PSO)
- League Championship Algorithm (LCA)
- Weighted Differential Evolution Algorithm (WDEA)
- Teaching-learning-based optimization (TLBO)
- Firefly Optimization algorithm (FOA)
- Bat Optimization Algorithm (BOA)
- Lion Optimization Algorithm (LOA)
- Réseau de neurones Groupe 1
- Réseau de neurones Groupe 2
- Réseau de neurones Groupe 3