This document may be modified if further clarification is needed. Check regularly!

Hello folks,

Here is the scope of your Homework, which has a 30% impact on the evaluation of your Introduction to Heuristic Algorithms course:

So far, we have seen 5 different optimization algorithms in our lessons. We will learn 2 more algorithms until the midterm exam. Each group will run and test the benchmark functions given to you with the algorithms.

1. Genetic Algorithm
2. Simulated Annealing
3. Grey Wolf Optimization
4. Particle Swarm Optimization
5. Harmony Search Algorithm

1.Grup	
2.Grup	
3.Grup	
4.Grup	
9.Grup	

5.Grup	
6.Grup	
7.Grup	
8.Grup	
10.Grup	

Equal conditions must be established for the comparison of different optimization algorithms. The most important indicator of this is the total number of evaluations. In other words, it is necessary to make an equal number of fitness value calculations in total.

In general, the total number of fitness calculations is as follows.

Total fitness calculations = Population size * number of iterations

However, in some iterations, there may be differences in the number of fitness calculations when local search is performed in addition to the global search. To follow this, you can define a parameter (counter) and increase it by +1 for each fitness calculation.

Robustness

We cannot decide whether an algorithm works well by running it only once for a function. These are algorithms that ultimately work with random values. Sometimes we can experience very good, sometimes very bad results. For this, we have to check the consistency (robustness) of the algorithm. While doing this, we will run an algorithm enough (for example, **5** times) and look at the arithmetic mean and standard deviation of the results. We will also compare our results against best_fitness, avg_fitness, and std_dev.

Parameter Optimization

Each algorithm may have different parameters within itself. These parameters need to be tested at ± certain intervals. For example, 0.05 and 0.1 are recommended values for mutation rate in the genetic algorithm. We should look at all values between [0.01, 0.15]. These are internal parameters.

Unlike this, each algorithm has similar parameters: Population number, iteration number, etc. Some algorithms work well with a high population size and few iterations, while others work well with a low population size and many iterations. While optimizing the parameters here, different pop_size & iteration_number attempts will be made with 2 nested for loops.

All functions will be run for dim=30

GA:

pop_size: 100, 250, 500, 1000

num_of_generations: 100, 250, 500, 1000

mut_prob: 0.01, 0.05, 0.1, 0.15

crossover_type: 1-point crossover, 2-point crossover, uniform crossover

selection_type: roulette_wheel, tournament_selection

SA:

First, convert the code to MINIMIZATION algorithm.

Initial Temperature: 1000, 5000, 10000

C (cool down factor): 1

Decrease temperature periodically: Arithmetic or Geometric

GWO:

pop_size: 100, 250, 500, 1000

num_of_generations: 100, 250, 500, 1000

a: let decrease from {4,3,2}

PSO:

pop_size: 100, 250, 500, 1000

num_of_generations: 100, 250, 500, 1000

HS:

HMS: 100, 250, 500, 1000

Bw: 0.1, 0.2, 0.25, 0.30

HMCR: 0.90, 0.95, 0.99

PAR: 0.15 0.25, 0.3 0.4,

Coding

You can code algorithms in a programming language of your choice. Your codes should be clear, understandable and easily modifiable. Write clearly what the functions do, which parameters the variables hold, with comment lines.

Functions

Check the Benchmark Functions file. There are 10 functions defined in the file. Use corresponding lower and upper bounds for each function. Compare EACH algorithm on EACH function with the best parameters you have found above.

Homework Folder:

Make a FINAL TABLE which comparing 5 algorithms on 9 functions. (9x5 Table)

An Excel file for each function in which the given and different parameters are executed with each algorithm. (Total 45 files)

Run.py files

Source codes (Only if you made modification in algorithm files – IF NEEDED)

DEADLINE: 08.11.2023 Friday 23.59