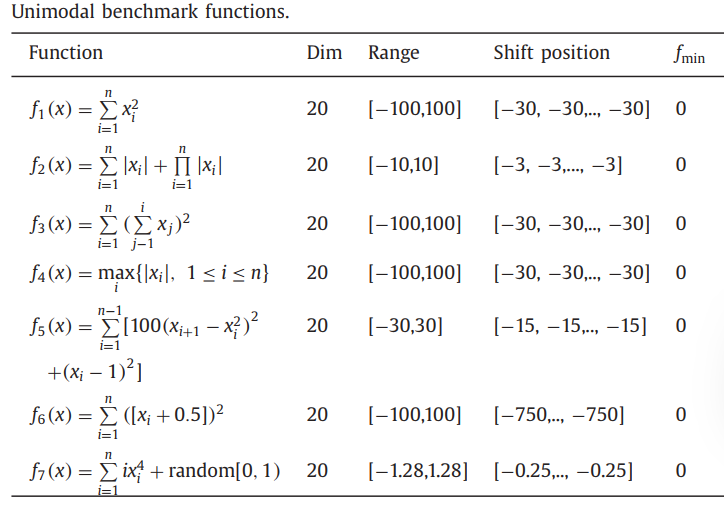
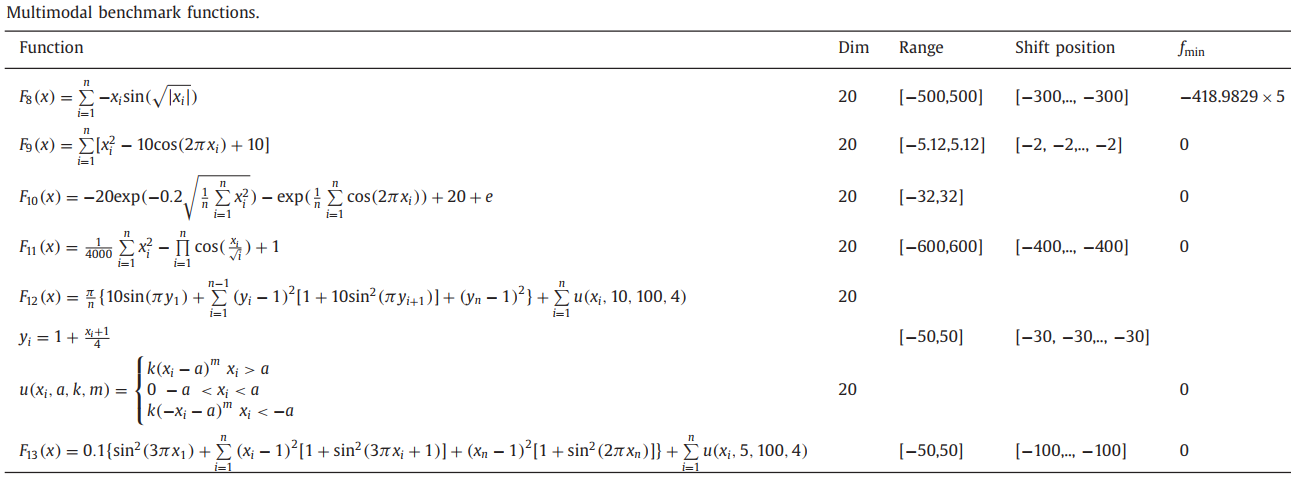
**Benchmarking latest optimization algorithms**

In the field of optimization using evolutionary algorithms, several test cases should be employed to confirm the performance of an algorithm. This is due to the stochastic nature of these algorithms, in which a proper and sufficient set of test functions and case studies should be employed to confidently make sure that the superior results are not happened by chance. However, there is no clear definition of suitability for a set of benchmark cases studies. Therefore, researchers try to test their algorithms on as many test cases as possible. In this project, we also employ several test functions with different characteristics similar to many ref. papers. The set of cases studies employed includes families  
of test functions: unimodal, multi-modal functions. The details are provided in the tables below.





The first family of test functions has no local optima and there is only one global optimum. This makes them highly suitable for testing the convergence speed and exploitation of algorithms. The second group of test functions, however, has multiple local solutions in addition to the global optimum. These characteristics are beneficial for testing local optima avoidance and explorative ability of an algorithm.

**Project Requirement**

1. For solving the aforementioned test functions, **a total of 40 search agents** are allowed to determine the global optimum over **5000 iterations**. You need to compare the results you achieved with respect to those obtained **using GWO (also with 40 agents )**

2. (max 50 points) Run 20 times for each test function (F1 to F13 mentioned above), each with 5,000 iterations, save your result for each function in a separate worksheet (i.e. F1, ..F13)

Your worksheet should look something like

Worksheet Fi (i = 1,..13)

|  | Run1\_yourOPT | … | Run20\_yourOPT | Run1\_GWO | … | Run20\_yourGWO |
| --- | --- | --- | --- | --- | --- | --- |
| Iter. 1 |  |  |  |  |  |  |
| … |  |  |  |  |  |  |
| Iter. 5000 |  |  |  |  |  |  |

The worksheet is used to generate a chart to compare different algorithms later.

Plot the convergence curve for each function (your OPT and PSO)

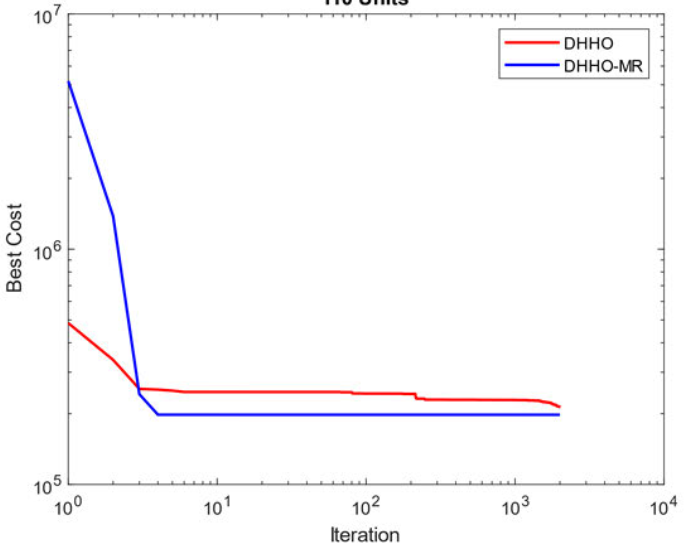


Figure 1. An example convergence curve

You also need to **record the computational time** required to run each function benchmark. In Matlab, use a combination of “tic” and “toc” as follows.

tic;

your\_code;

toc;

**You have to save the result in 20 trials of each function to calculate total time taken, some functions take only a few seconds to complete.**

**The 13 test functions can be found in your respective source code. If not, you can find those in GWO source code.**

3. (max 15 points) Propose possible improvements to your algorithm, remember for each improvement, you have to repeat the function benchmarking (20 times of F1-F13, computational time)

**4. Save all your work in 1 Excel file with all relevant code & a short document to report all your findings, if you choose existing improvements, provide necessary reference**

5. (max 35 points) Apply your improved algorithm to solve one optimization problem, you can refer to the problem mentioned on your paper or propose a new one (remember to attach your code for this question)

6. Helpful resources

* Free MATLAB (online):

codeocean.com (only 2 hours a month, you can register multiple accounts to circumvent)

nanohub.org (no limitation but very slow) (HIGHLY RECOMMENDED)

* Author’s resources (go to Projects)

<http://www.alimirjalili.com/GWO.html>

**Reference**

Papers and code are in the following link:

<https://drive.google.com/drive/folders/1-mGSJ_sDS_Fn_biuw2cVBwZiCVb56JSy?usp=drive_link>