### C:\Dropbox\DMU\Lessons\IMAT 1221-2 - Linear Algebra I - II\Phase Tests\2018-2019\Phase Test - Resit\dmulogo.pngCSIP5304 - Fuzzy Logic & Evolutionary Computing

### MSc Artificial Intelligence, 2023-2024

### Dr Zacharias Anastassi

# Weekly Exercises 3 - Evolutionary Computing

1. **Problem.** Select one problem from the Differential Evolution presentation slides, i.e. Rosenbrock, Himmelblau, Rastrigin or Ackley function. Use the DE code (which uses the DE/rand/1/bin variant), and modify it so that it solves one of these problems. Wherever you can select the number of variables, i.e. Rosenbrock or Rastrigin function, select nVar = 2. When examining whether the theoretical minimum has been reached, you can define a tolerance, e.g. 10^(-2), under which the absolute maximum error of the solution is considered acceptable, using the theoretical global minimum of each problem, also provided in the slides. Consider the initial run as Benchmark A. You can repeat the experiment several times and take the median value of the number of iterations needed. Provide some contour plots on the x-y plane, showcasing the progression between the stages.
2. **Effect of parameters.** Report the effect of different values of parameters, e.g. maximum iterations, population size, scaling factor, crossover probability and its range etc, and state the combination of values for the most efficient run. Consider this run (or median) as Benchmark B.
3. **Effect of variants.** Implement at least two more variants other than the DE/rand/1/bin that each differ in at least one of the x/y/z strategies. Determine the combination of variant and parameter values, including any new parameters introduced, that lead to the most efficient run. Consider this run (or median) as Benchmark C.
4. **Comparison.** Compare the Benchmarks A, B and C of your selected problem. Report on the common and different behaviours between the variants, and on the x/y/z strategies and parameters with the greatest effect. Interpret the results.

## Important notes

* Create a report that answers the above questions. The report will be evaluated based on its comprehensive but concise way to analyse, compare, present results, and draw conclusions.
* Please aim for 2-3 pages of text using Arial font and minimum size 11, excluding code, output, and graphs. The MATLAB code, output, if used, and some additional graphs can be included in an appendix, however the main part of the project, which includes the text and the main graphs, should independently provide the main points.
* The modified code should be copied as text in the report and also as separate .m files.
* Ensure that the code has optimum **performance** (e.g. avoid calculating the same formulas more than once where not necessary).
* Improve the **readability** (e.g. add some **comments** in commands, variables, and functions to explain what they do, even those included in the original code).
* Make the programme as **parameterised** as possible. This means that any constant should be defined at the start of the main script file (e.g. de.m) once, and then only the variable name should be passed to the rest of the script or functions.

## Deliverables

You should submit your report via VLE > Weekly Exercises 3 by the provided deadline.

Please submit directly, using multiple files if needed, rather than in a compressed file.