

Introduction to Artificial Intelligence

Exercise 3: Evolutionary and Genetic Algorithms

Filip Szatkowski

Summer 2023

1 Task

Write a program that solves a given 2D function using a genetic algorithm.

Variant 1

Optimize Rosenbrock function: $f(x, y) = (1 - x)^2 + 100(y - x^2)^2$. Use Evolutionary Strategy (1+1).

Variant 2

Optimize Rastrigin function: $f(x, y) = 20 + (x^2 - 10 \cos(2\pi x)) + (y^2 - 10 \cos(2\pi y))$. Use Evolutionary Strategy (μ, λ) .

Variant 3

Optimize Bukin function: $f(x, y) = 100\sqrt{|y - 0.01x^2|} + 0.01|x + 10|$. Use Evolutionary Strategy $(\mu + \lambda)$.

Variant 4

Optimize Booth function: $f(x, y) = (x + 2y - 7)^2 + (2x + y - 5)^2$. Use Generational Genetic Algorithm.

Variant 5

Optimize Styblinski-Tang function: $f(x, y) = \frac{1}{2} \left(\frac{x^4}{2} - 16x^2 + 5x + \frac{y^4}{2} - 16y^2 + 5y \right)$. Use Steady-State $(\mu + 1)$ Genetic Algorithm.

General requirements

The user should be able to select the parameters of the algorithm, such as the range of values from which the population is initialized, mutation strength, mutation probability and others. Use number of generations as the termination criterion. You are free to use the selection and crossover operators of your choice. The program should measure the optimization time, and the optimization process should be visualized when running the program (note that you should exclude the additional time spent for visualization when measuring optimization time).

2 Report

- The report should include a short description of your algorithm and used genetic operators.
- Run the program with different sets of parameters and report the results as well as the time spent on optimization.
- Analyse the impact of parameters of the genetic algorithm on the solution and the optimization time.

3 Technical details

- The solution must be implemented in Python. Jupyter notebooks are allowed as well.
- Please ensure that your code adheres to basic standards of lean coding in accordance to PEP8. Additionally, it should contain comments in the crucial parts to help with readability and understanding.
- The clear instruction on how to run and test the code should be included.
- The code can be submitted as an archive or a link to the code repository.
- The submission of the final report is mandatory, and the task will not be accepted without it.

4 Handing-in guidelines

You should submit the source code of your solution and the final report via Teams at least a day before the online assessment of the exercise. Programs delivered after the deadline will not be assessed. The online assessment will take place during the labs.

5 Assessment Criteria

You can get $[0, 5]$ points for the lab. The following criteria will be used to evaluate your work:

- Proper implementation of the algorithm: 2 points.
- Clean and well-documented code, with clear explanations and well-implemented logic: 1 point.
- Final report including results: 1 point.
- Online assessment: 1 point.

In case of any questions contact Filip Szatkowski via MS Teams