Evolutionary Computation

Program – GA in Numerical Optimization

September 22, 2025

Objectives

Practice and get familiar with the most widely used evolutionary algorithm — genetic algorithm (GA). In this assignment you need to make use of the taught subject matters about GA's representation, crossover, mutation, and survivor to solve the given problem.

Problem Description

Write efficient programs to implement GAs to find the minimal solution of the Schwefel function (SCH):

$$f_{\text{SCH}}(\vec{x}) = 418.98291N - \sum_{i=1}^{N} x_i \sin\left(\sqrt{|x_i|}\right),$$

where $-512 \leq x_i \leq 511$ and $N = \mathbf{10}$. This function is a continuous, multimodal, non-convex, deceptive, and N-dimensional function with a global minimum of 0. The landscape of a 2-dimensional SCH function is plotted below.

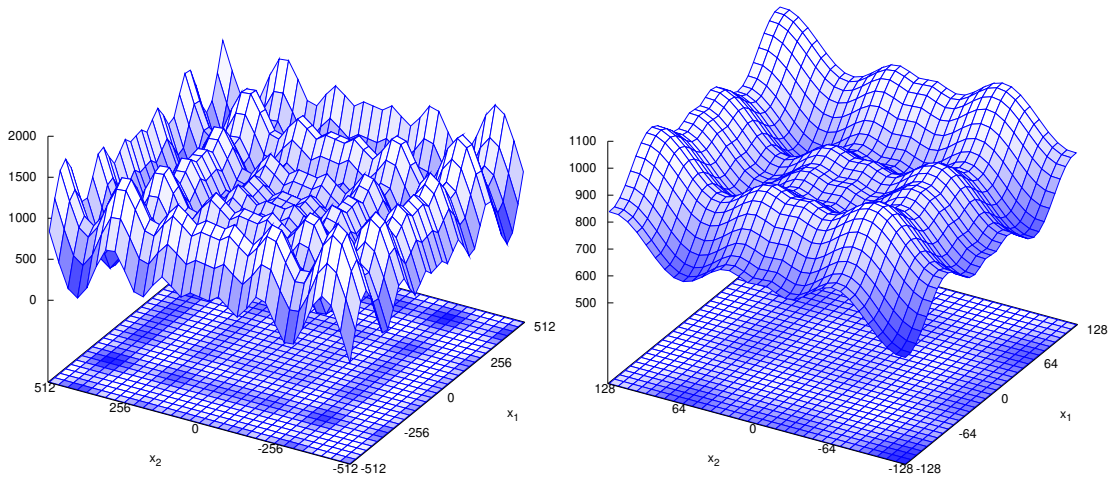


Figure 1: Fitness landscape of SCH

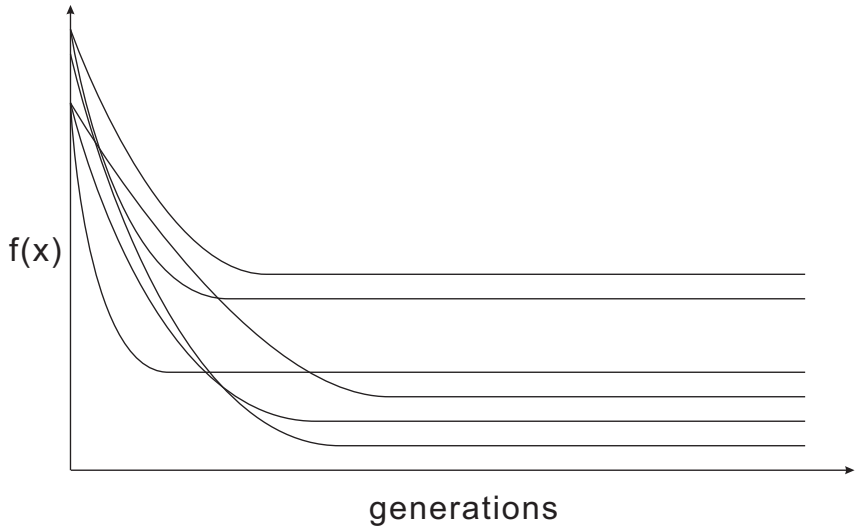
Requirements

You MUST

1. Implement **binary GA** and **real-valued GA** with different **operators** to solve the SCH:

	Binary GA	**Real-valued** GA
Representation	$c_i \in 2^{10}$	$c_i \in \mathbb{R}$
Population	Generational (size **100**)	
Parent selection	Tournament Selection ($n = 2$)	
Crossover ($p_c = 0.9$)	Uniform	
	2-point	Whole Arithmetic
Mutation ($p_c = 1/\ell$)	Bit-flip	Uniform
Survivor Selection	$\mu + \lambda$	
Termination	**500** generations	

2. Plot the **anytime behavior** (averaged over **30** trials) of the above GAs, e.g.,



3. **Compare** convergence speed and solution quality between different representations and operators; **give reasons** why some combinations perform better (or worse).

4. **Try other settings** for p_c, p_m, n and **discuss** their effects on convergence speed and solution quality.

5. Redo above steps 2–4 with a larger problem size $N = \mathbf{100}$.

Input/Output Format

- For detailed instructions on input and output formats, please refer to the GitHub page here.

Report

1. **Title, student id, name**
2. **Overview:** Present a brief description about your work in this assignment.
   (a) Your implementation approach
   (b) The results obtained
   (c) Any additional work (optional)
3. **Parameter analysis:** Analyze the effects of representation and parameters on the performance of GA for problem size $N = \mathbf{10}$ and $\mathbf{100}$:
   (a) Use anytime behavior plots to visually represent the effect of each parameter on the GA's performance against generations.
   (b) Analyze the impact of crossover rate p_c, mutation rate p_m, and tournament size n on the performance of your GA.
   (c) Binary vs. real-valued representation
      i. For each parameter, conduct experiments using binary and real-valued representations.
      ii. Compare their effects on convergence speed and solution quality.
      iii. Investigate and explain why do some combinations perform better (or worse) than others.
      iv. (Bonus) Analyze additional parameters, such as population size and number of generations used.
4. **Experience and conclusion**:
   (a) Your conclusion of this assignment
   (b) What have you learned from this assignment?
   (c) What difficulties did you encounter in this assignment?
   (d) Any feedback?

Grading

- **Coding (30%)**
  - Completion of the above requirements
  - Penalty
    * Compilation failure with Makefile (-5)
    * Not following input/output format (-5)
    * Incorrect filename (-3): (Please name the executable file `main` or `main.py`)
- **Report (70%)**
  - Grade will be based on the depth of analysis, the comprehensiveness of discussions, and the presentation quality: higher scores will be awarded for more detailed analyses and broader discussions.
  - Please name the file `report.pdf`

Submission

- **Due date: 2024/10/12**
- Source code (C/C++/Python) + Report (PDF file, no longer than **six** A4 pages in IEEE standard format (two-column, single-space, A4, 10pt))
- Zip (or rar) to a file named "(Student ID)_SCH.zip", and **upload to eeclass system**

Policy on AI-Generated Contents

*Work created by AI tools may **not** be considered original work and instead, considered **automated plagiarism**. It is derived from previously created texts from other sources that the models were trained on, yet doesn't cite sources.*

A responsible use of AI-based tools in completing coursework must be done in accordance with the following:

1. *You must clearly identify the use of AI-based tools in your work. Any work that utilizes AI-based tools must be clearly marked as such, including the specic tool(s) used. For example, if you use ChatGPT-4o, you must cite "ChatGPT-4o. (YYYY, Month DD of query). "Text of your query." Generated using OpenAI. https://chat.openai.com/"*
2. *You must be transparent in how you used the AI-based tool, including what work is your original contribution. An AI detector may be used to detect AI-driven work.*
3. *You must not use AI-based tools to generate the content of your work, but you may use them for editing.*
4. *You must ensure your use of AI-based tools does not violate any copyright or intellectual property laws.*
5. *You must not use AI-based tools to plagiarize without citation.*