University of Konstanz Winter Semester, 2025
Cyber-physical Systems Group Assignment 5
Group member name(s): Benedict Finsterwalder, Florian Shabani
Group member UID(s): 1564941, 1587539

INF-16580: Evolutionary Robotics

Introduction

In this task we implemented an evolutionary algorithm to solve the Ackley optimization problem in three dimensions. The objective was to minimize the function

$$f(x,y,z) = -20\exp\left(-0.2\sqrt{\frac{1}{3}(x^2+y^2+z^2)}\right) - \exp\left(\frac{1}{3}(\cos(2\pi x)+\cos(2\pi y)+\cos(2\pi z))\right) + 20 + e$$

on the interval $x, y, z \in [-32.768, 32.768]$. The global minimum is at $(0,0,0)$ with $f(0,0,0) = 0$. To convert this into a maximization problem, we defined fitness as fitness $= 1/(f(x,y,z)+1)$, where the optimal fitness is 1.0.

Solution Method

We implemented a complete evolutionary algorithm. Our implementation can be seen on Github. The algorithm uses the following components

Genetic Representation Each individual is represented as a real-valued vector $[x, y, z]$ where each coordinate is bounded by $[-32.768, 32.768]$.

Population Size We used a population of 100 individuals, initialized uniformly at random across the entire search space.

Selection Method We used tournament selection with tournament size 3. In each tournament, three individuals are randomly selected and the one with highest fitness wins and is chosen for recombination.

Recombination 2 parents are chosen with the selection method. For each coordinate, the child randomly inherits from either parent with equal probability.

Mutation Gaussian mutation with per-gene mutation rate. Each coordinate independently has a probability (mutation rate) to be mutated by adding Gaussian noise. Mutated values are clipped to stay within bounds.

Replacement Strategy Generational replacement with elitism. The top 2 individuals are preserved unchanged to the next generation, while the remaining 98 are generated through selection, crossover, and mutation.

Results

We tested multiple parameter configurations. The plots show fitness progression over generations.

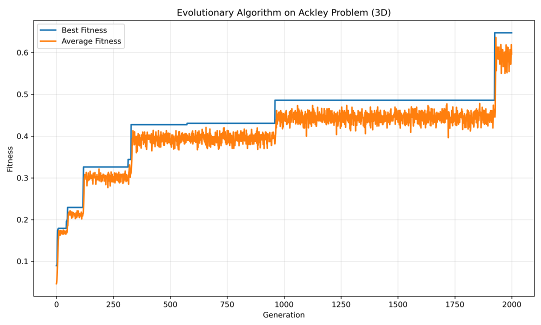
We varied mutation rates $(0.01, 0.1, 0.3, 0.5)$, population sizes $(50, 100, 200)$, and generation counts $(100, 2000)$.

We achieved fitness scores of up to 0.8 by tweaking basic parameters. However, this is not a very large score and we could clearly see we had a problem by observing the large jumps of best scores in our graph.

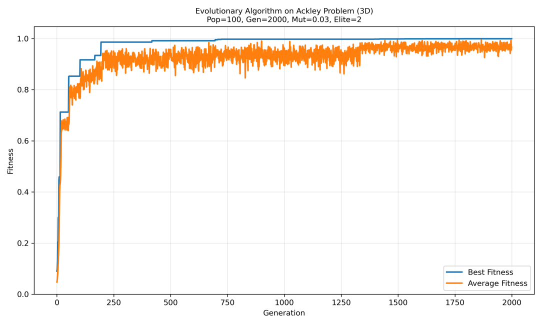
The biggest jump in our results was when we switched the mutation from mutating all 3 parameters to having an equal chance of mutating each coordinate separately. We went from fitness values of 0.6-0.8 to 0.99 or even more, which essentially solved our problem. In the images we see the difference of before and after this change.

Typical final results achieved Ackley values below 0.001, corresponding to fitness above 0.995, which demonstrated successful convergence very close to the global optimum at $(0, 0, 0)$.

To achieve the highest results we implemented an adaptive mutation rate. We split the generations into three phases, with the mutation strength decreasing in each phase (5.0, 2.0, 0.2) so that we can get more sensitive adjustments later on. And for our final result we let the program run for 10,000 generations which gave us a final solution with values [-9.46266727e-07 8.29708291e-07 -7.78903167e-07] Ackley value 0.00000342 Fitness 0.99999658



Before ind. coordinate mutation



After ind. coordinate mutation

Conclusion

We successfully implemented an evolutionary algorithm that consistently solves the Ackley optimization problem. The algorithm reliably converges to near-optimal solutions close to the global minimum. The per-gene mutation strategy allows fine-grained exploration of the search space, while elitism ensures good solutions are preserved.