Д. И. Альромхин

GENETIC ALGORITHM

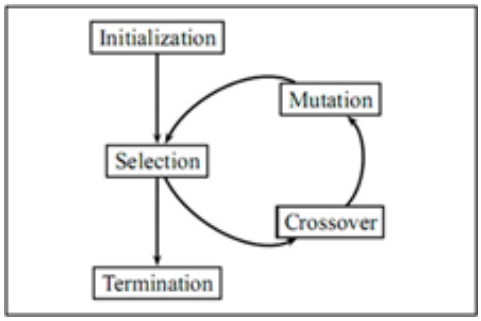
Introduction

It's an optimization technique used to solve nonlinear or nondifferentiable optimization problems, they use concepts from evolutionary biology to search for a global minimum to an optimization problem, Start with an initial generation of candidate solutions that are tested against the objective function. Subsequent generations evolve from the first through selection, crossover and mutation. We then generate subsequent generations of points from the first generation through selection crossover and mutation.

I. How evolution works (binary case)

- 1-Selection Retain the preforming bit strings from one generation to the next.
- Favour these for production
- Parent1 = [1 0 1 0 0 1 1 0 0 0]
- Parent2 = [1 0 0 1 0 0 1 0 1 0]
- 2-Crossover Parent1 = [1 0 1 0 0 1 1 0 0 0]
- Parent2 = [1 0 0 1 0 0 1 0 1 0]
- Child = [1 0 1 1 0 0 1 0 1 0]
- 3-Mutation Parent = [1 0 1 0 0 1 1 0 0 0]
- Child = [0 1 0 1 0 1 0 0 0 1]

This is a binary problem where the variables in the optimization problem can either take on a value of 0 or 1.



The selection means to retain the best preforming parent from one generation to the next, so if we have parent1 and parent2 from the previous generation, through selection those make it through to the next generation just because they performed well in the previous generation.

Because they performed well they might also be used for crossover and in crossover what we do is select common similarities between the different parent variables and keep those the same to create children variables that will be in the next generation.

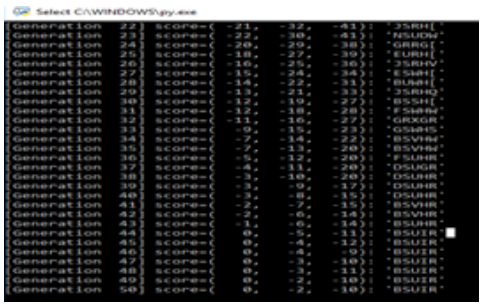
The last thing that happens is what is known as mutation where we take parent and mutate certain variables to take on random values and we create a child based of the mutation, mutation allows genetic algorithms to avoid falling into local minima and it helps to explore the solution space well. Let's see how this works on an optimization problem to find the phrase "BSUIR" using programming language python.

[Generation Number] score=(max fitness, average fitness, min fitness): 'char' [Generation 1] score=(-102, -364, -696): '[.HQe'

This is the first generations from my genetic algorithm, the first step that the genetic algorithm does is it evaluate all these characters and determines the fitness function for each on of them, the next thing it will select a few good solutions as the parent to continue to the next generation and use them to create subsequent generation, it keeps generating until the algorithm converges.

The algorithm can converge through a variety of convergence criteria, in this example I used a fixed number of generations(60gen).

A fitness function is a particular type of objective function that is used to summarise, as a single quantity used to characterize the performance of a device, system or method, relative to its alternatives, how close a given design solution is to achieving the set aims. Fitness functions are used in genetic programming and genetic algorithms to guide simulations towards optimal design solutions. Let's see how the program works:



As we can see the algorithm converges at the 44th generation and it's the solution of this problem.

Джорж Ихаб Альромхинч, студент гр.420611 ФИТиУ БГУИР, georgealromhen93@gmail.com.
Научный рукаводитель: Шилин Леонид Юрьевичч, декан факултета ИТиУ БГУИР.