Evolutionary Algorithm Overview

Yash Savera
DAIICT, Gandhinagar
202203074@daiict.ac.in

Het Khunt
DAIICT, Gandhinagar
202203011@daiict.ac.in

April 2025

1 About Evolutionary Algorithm

An evolutionary algorithm is inspired by natural evolution. In nature, species evolve through variations, and individuals with stronger traits tend to survive and reproduce more successfully.

1.1 Darwin's Principle of Evolution

According to Darwin's principle, stronger individuals have a higher chance of survival in environments with limited resources such as food, shelter, and mates.

1.2 Relation to Algorithms and Mathematics

Similarly, in evolutionary algorithms, new solutions are generated from existing ones using variation mechanisms like crossover and mutation. Over several generations, better solutions tend to survive and evolve toward optimal or near-optimal solutions, often reaching the Pareto front in multi-objective problems.

2 Crossover

Crossover involves combining parts of two parent solutions to produce new offspring. These offspring inherit traits from both parents.

Example:

```
P1: 1 0 1 0 1 1 | 0 0 1 0
P2: 0 1 0 1 0 0 | 0 1 1 0

O1: 1 0 1 0 1 1 | 0 1 1 0
O2: 0 1 0 1 0 0 | 0 0 1 0
```

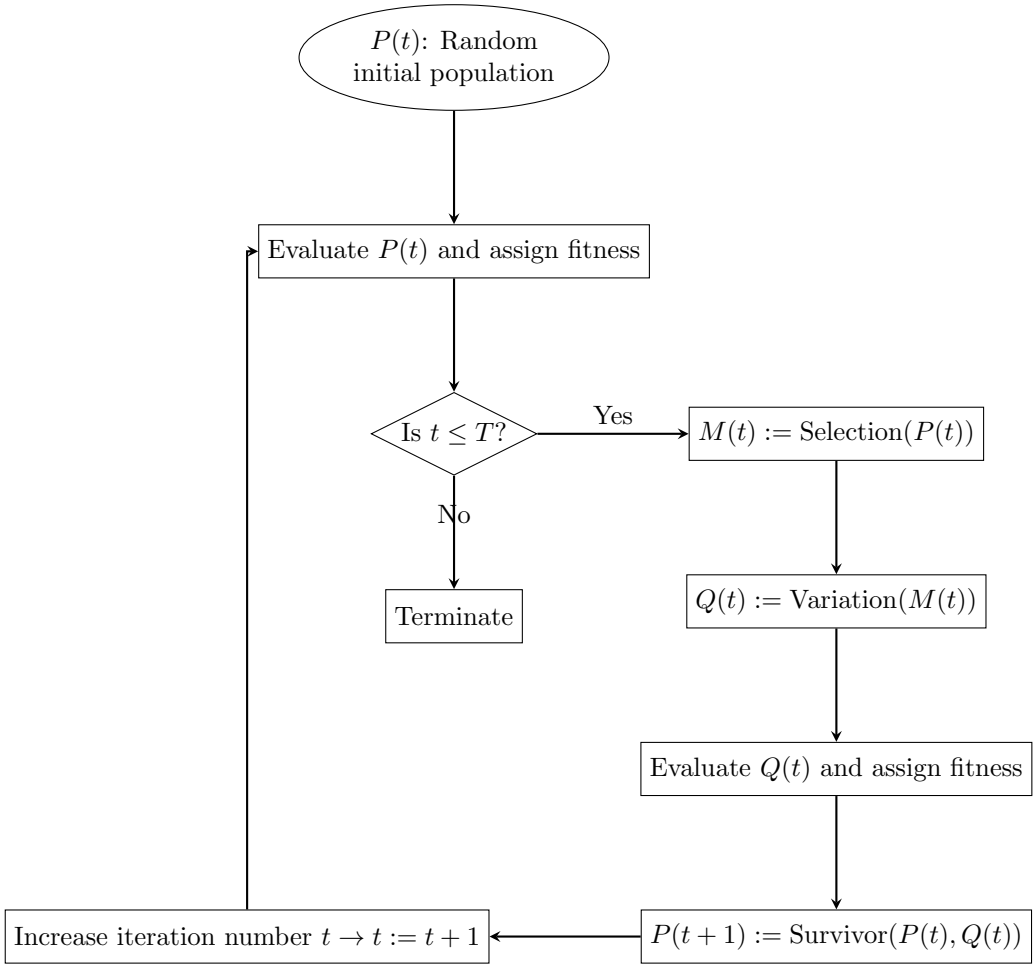
3 Mutation

Mutation involves randomly modifying a part of a solution to introduce variation and potentially discover better solutions.

Example:

```
Original: 1 0 1 0 1 1 0 1 1 0
Mutated : 1 0 1 1 1 1 0 1 1 0
```

4 Flowchart of Evolutionary Computation Technique



5 Algorithm

Algorithm 1 Evolutionary Computation Technique

Require: Maximum generations T

1: Initialize random population $P(1)$
2: Evaluate $P(1)$
3: $t \leftarrow 1$
4: **while** $t \leq T$ **do**
5: $M(t) \leftarrow \text{Selection}(P(t))$
6: $Q(t) \leftarrow \text{Variation}(M(t))$
7: Evaluate $Q(t)$
8: $P(t+1) \leftarrow \text{Survivor}(P(t), Q(t))$
9: $t \leftarrow t+1$
10: **end while**

Basically, we start with any random set of n solutions. Then, we perform mutation, and from the mutated solutions, we apply crossover to generate new ones. After evaluating the new solutions, we select the best n among them. This process is repeated for a good number of iterations."

6 Advantages of Evolutionary Computation (EC) Techniques

- Applicable to problems where no well-defined or effective method is available.
- Particularly useful for solving complex problems such as non-linear, noisy, or discrete-variable optimization tasks.
- Well-suited for problems where multiple solutions are desired, such as multi-objective optimization.
- Requires no prior assumptions about the structure or properties of the problem space.

7 Disadvantages of Evolutionary Computation (EC) Techniques

- There is no guarantee of finding the optimal solution within a finite amount of time.
- Maintaining and evolving a population of solutions can be computationally expensive.

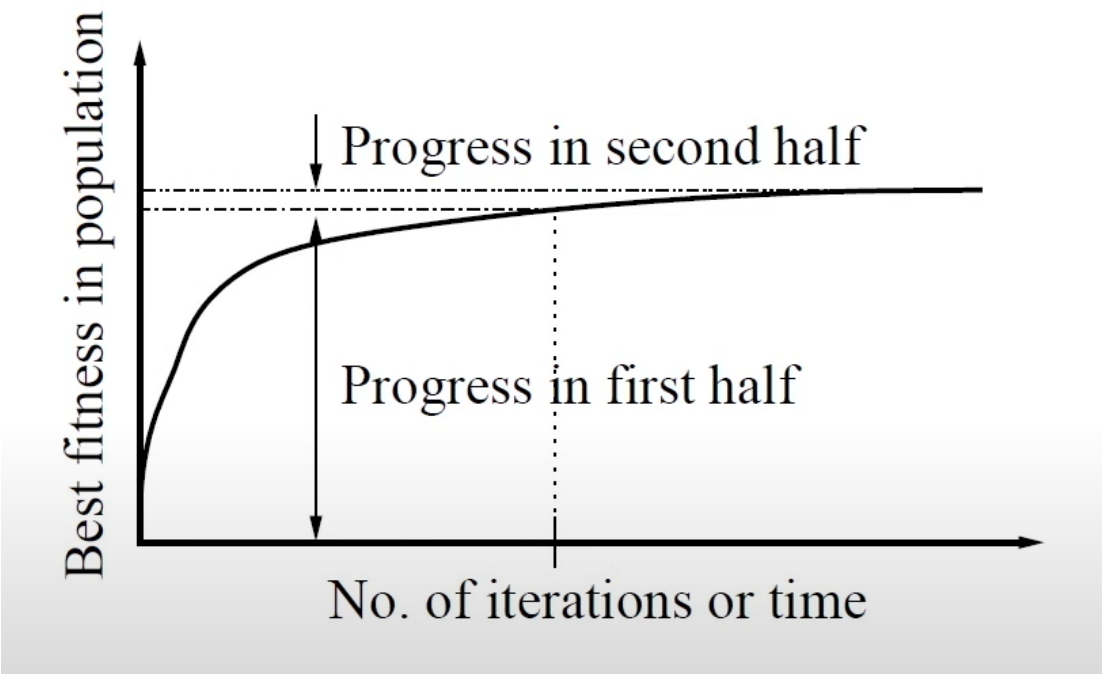


Figure 1: Fitness vs. Iterations