Genetic Algorithm Steps

A Genetic Algorithm (GA) is a search heuristic used to solve optimization and search problems, inspired by the principles of natural selection and genetics. Here's a breakdown of the five main steps:

1. Initialization

The process begins with a randomly generated initial population, often called a ”population of chromosomes” or ”individuals.” Each individual (chromosome) is a possible solution to the problem, represented by a binary string (0s and 1s).

Example: Consider an initial population of chromosomes:

- A1: 01000
- A2: 11101
- A3: 10110

In this example, each binary string represents a candidate solution in the population.

2. Fitness Assignment

Each chromosome is evaluated based on a fitness function, which is a measure of how well it solves the problem. The fitness function assigns a score to each chromosome, indicating its ”fitness” or suitability as a solution.

Example: If the objective is to maximize $f(x) = x^2$, the fitness of each chromosome is calculated by converting the binary string to a decimal value (X) and applying the function $f(x)$.

Illustration:

- A1 (01000): $X = 8$, Fitness $= 8^2 = 64$
- A2 (11101): $X = 29$, Fitness $= 29^2 = 841$
- A3 (10110): $X = 22$, Fitness $= 22^2 = 484$

3. Selection

Selection is the process of choosing individuals for reproduction based on their fitness. Higher-fitness individuals are more likely to be selected, simulating the "survival of the fittest" concept. Techniques such as roulette wheel selection or tournament selection are commonly used.

Example: Given fitness values, assign selection probabilities proportional to the fitness scores.

Illustration: A2 might have a higher probability of selection than A1 or A3 due to its higher fitness score.

4. Crossover (Reproduction)

Crossover is a genetic operator used to combine the genetic information of two parent chromosomes to create offspring. A crossover point is selected, and segments of the parent chromosomes are swapped to produce new chromosomes.

Example: For A1 = 01000 and A2 = 11101, choosing a crossover point at 3 could result in:

- Offspring 1: 01101
- Offspring 2: 11000

This step introduces genetic diversity in the population.

5. Mutation

Mutation is a genetic operator that alters one or more bits in a chromosome to introduce new genetic structures. This prevents the algorithm from getting stuck in a local optimum by exploring new areas of the solution space.

Example: For A3 = 10110, mutating the third bit results in:

- Mutated Chromosome: 10010

Mutation maintains diversity within the population and enables exploration of the solution space.

Termination

The algorithm repeats the selection, crossover, and mutation steps until a stopping criterion is met. Common termination conditions include reaching a maximum number of generations or achieving a satisfactory fitness level. The best solution found during the iterations is taken as the optimal solution.

Each step progressively refines the population towards better solutions, using the principles of natural selection, crossover, and mutation. This process continues iteratively until the algorithm converges on an optimal or near-optimal solution.

Example Analysis of Genetic Algorithm

The goal is to maximize the fitness function $f(x) = x^2$ within the given range (0 to 31).

Step 1: Initial Population Selection

A random initial population of binary strings (chromosomes) is selected. Each binary string represents an integer value (X) when converted from binary to decimal.

String No.	Initial Population (Binary)	X Value (Decimal)
1	01100	12
2	11001	25
3	00101	5
4	10011	19

Table 1: Initial Population Table

Step 2: Calculating Fitness Values

For each individual in the population, we calculate the fitness value $f(x) = x^2$.

String No.	X Value	Fitness $f(x) = x^2$
1	12	144
2	25	625
3	5	25
4	19	181

Table 2: Fitness Table

Total Fitness Sum = 144 + 625 + 25 + 181 = 1155

Step 3: Calculating Probability and Expected Count

Calculate the probability of each chromosome being selected, using

$$\text{Probability} = \frac{\text{Fitness}}{\text{Total Fitness}}$$

Calculate the expected count for each chromosome using

$$\text{Expected Count} = \text{Probability} \times \text{Population Size}$$

Step 4: Selection Based on Actual Count

Based on the expected count, we round to determine how many times each chromosome will be selected.

String No.	Fitness	Probability	% Probability	Expected Count
1	144	0.1247	12.47	0.4987
2	625	0.5411	54.11	2.1645
3	25	0.0216	2.16	0.0866
4	181	0.3126	31.26	1.2502

Table 3: Probability and Expected Count Table

String No.	Initial Population	Expected Count	Actual Count
1	01100	0.4987	1
2	11001	2.1645	2
3	00101	0.0866	0
4	10011	1.2502	1

Table 4: Selection Table

Step 5: Crossover

Choose pairs for crossover. For each pair, select a crossover point and swap segments of the chromosomes after the crossover point.

String No.	Mating Pool	Crossover Point	Offspring after Crossover	X Value	Fitness
1	01100	4	01101	13	169
2	11001	4	11000	24	576
3	11001	2	11011	27	729
4	10011	2	10001	17	289

Table 5: Crossover Table

Step 6: Mutation

Introduce mutations by flipping one bit in each chromosome to introduce variation.

String No.	Offspring after Crossover	Mutation Chromosome for Flipping	Offspring after Mutation	X Value	Fitness
1	01101	10000	11101	29	841
2	11000	00000	11000	24	576
3	11011	00000	11011	27	729
4	10001	00101	10100	20	400

Table 6: Mutation Table

Step 7: Result Summary

After mutation, we obtain new fitness values. Summing up all the fitness values, we get:

$$\text{Total Fitness Sum} = 841 + 576 + 729 + 400 = 2546$$
$$\text{Average Fitness} = \frac{2546}{4} = 636.5$$
$$\text{Maximum Fitness} = 841$$

Lecture Note by: H.L.N. Himanshi - Department of Artificial Intelligence and Non-Linear Analysis, Faculty of Mathematics and Computer Science, University of Lodz, Poland