Artificial Intelligence

BS (CS) _Spring_2025

Lab_07 Manual

Learning Objectives:

1. Genetic Algorithm

Lab Manual

Genetic Algorithm:

Introduction:

Genetic Algorithm (GA) is an evolutionary optimization technique inspired by natural selection. It is used to solve complex problems by evolving a population of potential solutions over multiple generations. The algorithm mimics biological evolution through selection, crossover, and mutation to find the best solution.

The algorithm works by selecting the fittest individuals, combining them through crossover, and introducing mutations to maintain diversity, gradually converging towards an optimal solution.

Key Components:

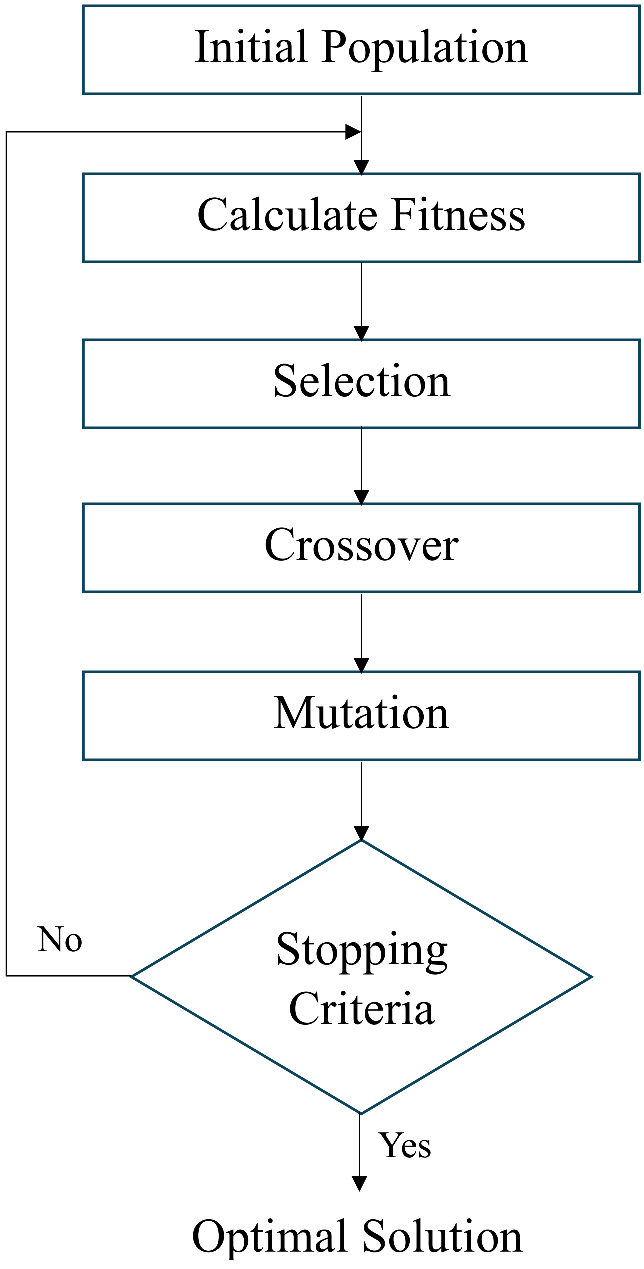
1. **Population:** A set of candidate solutions
2. **Fitness Function**: A function to evaluate how good a solution is.
3. **Selection**: Choosing the best candidates for reproduction.
4. **Crossover**: Combining two parent solutions to create offspring.
5. **Mutation**: Introducing random changes to maintain diversity

Step-by-Step Working of Genetic Algorithm:

1. **Initialize:**
   - Generate an initial population of possible solutions (chromosomes), usually randomly.
   - Define a fitness function to evaluate how good each solution is.
2. **Evaluate:**
   - Compute the fitness value for each chromosome in the population using the fitness function.
3. **Selection (Survival of the Fittest):**
   - Select the best solutions (parents) based on fitness scores.
4. **Crossover (Recombination):**
   - Combine genetic information from two parents to create offspring.
5. **Mutation (Introduce Randomness):**

- Apply small random changes to some chromosomes to maintain diversity.
- Helps the algorithm avoid getting stuck in local optima.

6. **Create New Generation:**
    - **R**eplace old population with the new generation.
    - Maintain population size.
7. **Repeat:**
    - Continue steps **2-6** until a stopping condition is met, such as:
        - A maximum number of generations is reached.
        - The best solution is good enough according to the fitness function.



Pseudocode for Genetic Algorithm:

```
Genetic_Algorithm(Problem, Population_Size, Crossover_Rate, Mutation_Rate,
Stopping_Criteria):

    Population ← Generate_Random_Solutions(Population_Size)

    Evaluate_Fitness(Population)

    while not Stopping_Criteria(Population) do:

        Parents ← Select_Parents(Population)

        Offspring ← Crossover(Parents, Crossover_Rate)

        Offspring ← Mutate(Offspring, Mutation_Rate)

        Evaluate_Fitness(Offspring)

        Population ← Select_Next_Generation(Population, Offspring)

    return Get_Best_Solution(Population)
```