

Open Ended Lab Report

Artificial Intelligence

Third Year-Computer and Information Systems Engineering

Batch:2022



Group Members:

Mr. Mohammed Ishhaq (CS-22152)

Mr. Zaid Ahamed (CS-22154)

Mr. Muhammed (CS-22155)

Submitted to: Miss. Hameeza Ahmed

Submission Date: 25/11/2024

Table of Contents

S. No.	Description	Pg. No.
1	Introduction	3
2	Problem Statement	3
3	Details of The Model	4
4	Implementation	5
5	Results	6
6	Conclusion	6

Introduction

Genetic Algorithms (GAs) are optimization techniques inspired by the process of natural selection. They are commonly used for solving complex problems where traditional methods are inefficient. **This report explores the application of a Genetic Algorithm to solve the Knapsack Problem.**

Problem Statement

The Knapsack Problem involves selecting a subset of items with given weights and values to maximize the total value without exceeding a specified weight capacity. For this study, the problem is defined as follows:

Items:

Item 1: Weight = 2, Value = 3

Item 2: Weight = 3, Value = 4

Item 3: Weight = 4, Value = 5

Item 4: Weight = 5, Value = 8

Knapsack Capacity: 9

Details of Model

The Genetic Algorithm employs the following components to solve the problem:

- 1. Encoding:** Each solution is represented as a binary chromosome where each bit indicates whether an item is selected (1) or not (0).
- 2. Population:** A set of chromosomes representing potential solutions.
- 3. Fitness Function:** The total value of selected items, with a penalty applied if the total weight exceeds the knapsack capacity.
- 4. Selection:** Roulette wheel selection is used to choose parents based on fitness.
- 5. Crossover:** Single-point crossover is applied to generate offspring from two parent chromosomes.
- 6. Mutation:** Random flipping of bits is performed to introduce diversity.
- 7. Termination:** The algorithm runs for a fixed number of generations.

Implementation

The algorithm was implemented in Python. An initial population of chromosomes was randomly generated. At each generation, the population was evaluated using the fitness function, and new chromosomes were produced using selection, crossover, and mutation. The process was repeated for 10 generations. Below are the key parameters used:

- Population Size: 6
- Gene Length: 4 (corresponding to 4 items)
- Crossover Rate: 0.8
- Mutation Rate: 0.1
- Number of Generations: 10

Results

The algorithm successfully converged to an optimal solution. The final results are summarized below:

- Best Chromosome: 1110
- Selected Items:
 - Item 1 (Weight = 2, Value = 3)
 - Item 2 (Weight = 3, Value = 4)
 - Item 3 (Weight = 4, Value = 5)
- Total Value: 12
- Total Weight: 9 (within the capacity of 9)

Conclusion

This study demonstrated the effectiveness of Genetic Algorithms in solving the Knapsack Problem. The algorithm successfully identified a high-quality solution within the specified constraints. The approach is versatile and can be adapted to other optimization problems by modifying the encoding, fitness function, and parameters. Future work could involve exploring alternative selection methods, dynamic parameter tuning, or hybrid approaches to further enhance performance.