Approved

DATA STRUCTURES AND ALGORITHMS PROPOSAL FOR THE PROJECT

Dr. Ayaz Hassan

Project Title

GENETIC ALGORITHMS: Theoretical and Experimental Analysis with Python Implementation

Group Members

Project is designed by the following group members:

- *Mubaraka Shabbir*
- *Yabudullah Ahmed Bakhtiar*
- *Fizza Rubab*

Project Description

A genetic algorithm (or GA) is a stochastic search technique used in computing to find true or approximate solutions to optimization and search problems. GAs are categorized as global search heuristics. GAs are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover (also called recombination). GAs find extensive applications in combinatorial optimization problems and hence is a great tool for engineering design, robotics, and resource scheduling, etc.

A genetic algorithm is a nature-inspired algorithm which mimics the Darwin's Natural Selection theory. It is based on three concepts: selection, reproduction, and mutation. A list of solutions represents a population where the individual solutions are called chromosomes or individuals. We generate a random set of individuals (solutions), select the best ones based on their fitness, cross them over and finally, slightly mutate the offspring. So, some individuals die and get replaced by new offsprings eventually creating a new generation. After each iteration, a new generation is created with better individuals (solutions) until the algorithm converges. Either the program has found the best solution or a fixed number of generations have completed or our time and space resources have been used up depending upon the programmers' constraint.

Our project implements the above genetic algorithm on 3 important computing problems in python:

- *Travelling Salesperson (TSP) problem to find the smallest route between cities.*
- *Find the global maxima of a mathematical function over a given domain and range.*
- *Solve the 0-1 Knapsack Problem to maximize the value of items within the given weight constraints.*

We will provide an experimental analysis to find the stable runtime of GA for the above problems. The performance will be evaluated based on multiple datasets. A thorough theoretical analysis will be

carried out to determine the time complexity of GAs and identify their efficiency and Big O. Furthermore, for the 1-0 Knapsack Problem, we will implement it using different selection strategies such as Roulette wheel selection and the Rank selection. We will also deploy different crossover strategies such as one-point crossover or uniform crossover on the same problem set. Performance for each GA with different operators' strategies will be evaluated to find out which strategy gives quick and accurate results.

Project Outcomes

Our project GENETIC ALGORITHMS: Theoretical and Experimental Analysis with Python Implementation, will provide a deep insight into GAs and their implementation. Following outcomes will be met through the completion of our project:

- *Implementation of Genetic Algorithms on real-world optimization problems to find the optimal solution.*
- *Evaluation of time complexity and order of growth of GAs that is largely dependent on fitness, crossover, selection and mutation operators.*
- *Comparison and Contrast of GAs performance on the same objective function but deploying different selection and crossover strategies.*
- *Experimental Analysis and Performance evaluation for multiple datasets by calculating the stable execution time of GA.*
- *Implementation and manipulation of list data structure (ADT) to represent populations and individuals in the Genetic Algorithm.*
- *Insight to modular programming used in Genetic Algorithms.*

Rationale

Primary reason for choosing Genetic Algorithm as our project resides in its real world applications. It is one of the most popular Nature Inspired Algorithm for combinatorial optimization used heavily in engineering design, traffic routing and robotics. However, less work has been done on its complexity analysis and efficiency improvement strategies as we have learned in CS-102 DS&A. Therefore we will shed more light on its complexity, data structures involved and its python implementation in real world problems.

Resources

Below is a tentative list of libraries we will use for implementation of algorithm in python programming language:

- *Random library*
- *Tkinter library*

Below is the tentative list of websites we will refer to for help and clarifications:

- *Wikipedia.org*
- *Researchgate.com*

Below is the tentative lists of books we will use for assistance for the completion of this project:

- *An Introduction to Genetic Algorithms by Melanie Mitchell*