Alessandro Allegranzi

BU MET CS 767

Assignment 5Genetic Algorithms

6/10/2024

Table of Content

[1. Representation of the Data 1](#_Toc168508830)

[2. Representation of a Route 1](#_Toc168508834)

[3. Crossover 2](#_Toc168508836)

[4. Mutation 2](#_Toc168508843)

[5. Result on the Given Data 2](#_Toc168508845)

[6. Result on Your Data 2](#_Toc168508849)

[7. Source Code 2](#_Toc168508851)

[8. Comments on Performance 2](#_Toc168508853)

[References 2](#_Toc168508855)

[Evaluation 3](#_Toc168508856)

[Appendix 1 3](#_Toc168508857)

[Appendix 2 3](#_Toc168508858)

6/10/2023

MET CS 767 Assignment 5: GA’s

*Alessandro Allegranzi*

You are to implement the *traveling salesman problem* using a genetic algorithm in the manner outlined below—preferably in Python, otherwise Java. We’ll call this project T-Order because it has a unique way to represent routes. You can assume that there is a route connecting every pair of cities, and that no two distances are equal. The latter simplifies coding, as you will see.

Eric Braude created this technique to perform trouble-free crossover for Traveling Salesman. It may be published elsewhere (although we have not located any such reference), and if so, please do not such work for this assignment.

The instructions are otherwise the same as in the previous assignments.

It is recommended that you build this application by modifying an AI-generated genetic algorithm traveling salesman application rather than building from scratch, which is onerous.

### >>AI generation used (or check: I did not use AI generation here \_\_). Please collapse this.

PARAGRAPH DESCRIBING YOUR VALUE ADDED TO AI-GENERATED MATERIAL

Your response replaces this.

YOUR PROMPT SEQUENCE

[1] Your first prompt replaces this.

[2]

# Representation of the Data

## Taking into account how crossover below will operate (see Section 3), explain how T-Order will *represent* the data (e.g., “a list consisting of …”) in general.

## As an example, include showing how the following is to be represented when you use your representation:

## B(oston) to L(ondon) 3(k miles), L to M(umbai) 4.5, M to S(hanghai) 3.1, S to L 5.7, B to M 7.6, and B to S 7.8.

The above is only an example: your assignment should apply to any traveling salesman problem with unique node distances. You can refer to the nodes as cities if you wish.

your response replaces this

# Representation of a Route

## Explain how T-Order will *represent* a route in general. Include, as an example based on the above example, how the following route will be represented: *B to L to M to S to B*.

your response replaces this

# Crossover

## Define a crossover function consistent with the following.

## T-Order should create a child from two parents by simple cuts, as in the example below. (Bold and italics are added to clarify what part of the child comes from what part of the parents.)

## Parent 1: Boston 🡪 2nd closest unvisited city[[1]](#footnote-1) 🡪 2nd closest unvisited city[[2]](#footnote-2) 🡪 closest unvisited city 🡪 Boston

## *Parent 2: Boston 🡪 closest unvisited city 🡪 2nd closest unvisited city 🡪 closest unvisited city 🡪 Boston*

## Child route from these parents—with crossover point at 1:

## Boston 🡪 2nd closest unvisited city *🡪 2nd closest unvisited city 🡪 closest unvisited city 🡪 Boston*

your response replaces this

# Mutation

## Explain (clearly) how your T-Order performs mutation.

your response replaces this

# Result on the Given Data

## Describe the result from executing on the following example data …

## *B(oston) to L(ondon) 3(k miles), L to M(umbai) 4.5, M to S(hanghai) 3.1, S to L 5.7, B to M 7.6, and B to S 7.8*.

## What do you think of this result? Explain.

your response replaces this

# Result on Your Data

## Describe the result from executing your application on illustrative data of your choice. What do you think of the result? Explain.

your response replaces this

# Source Code

## Paste your source code below—or refer to an appendix. It should accompany this doc as well.

your response replaces this

# Comments on Performance

## Compare the performance of T-Order with at least one known Traveling Salesman GA implementation.

your response replaces this

# References

Show that you used a wide variety of resources by listing them below and clearly indicating in the body above where you used. Make sure to use proper referencing in your paper. We suggest using APA format, but other formats are fine as long as they clearly distinguish your work from work of others in your response. In general, observe the stated plagiarism rules.

[1] Shendy, Ramez. “Traveling Salesman Problem (TSP) using Genetic Algorithm (Python)”. *Medium*. <https://medium.com/aimonks/traveling-salesman-problem-tsp-using-genetic-algorithm-fea640713758>. Aug 5, 2023.

[2] …

# Evaluation



# Appendix 1

# Appendix 2

1. i.e., compared to all other direct hops from the city just visited (Boston in this case) [↑](#footnote-ref-1)
2. i.e., compared to all other direct hops from the city just visited [↑](#footnote-ref-2)