IN4050 - Assignment 1

Gustav Lillesæter - gustavli

Innhold

1 Introduction

Some algorithms to solve TSP. Exhaustive, Hill climbing and Genetic Algorithm.

2 How to run the program

1. unzip folder
2. run jupyter notebook

I also added a printout of my own executions as notebook.pdf, as some of them take quite some time.

3 Exhaustive Search

The shortest tour among the first 10 cities was: Copenhagen →Hamburg →Brussels →Dublin →Barcelona →Belgrade →Istanbul →Bucharest →Budapest →Berlin →Copenhagen
Length: 7486.31
Time of execution: 3778.3 seconds

As a quick estimation: 10 cities is 10! possible routes. (if we ignore the calculation of the length) 24 cities is 24! possible routes. Which means there is roughly $\frac{24!}{10!} = 1.7 * 10^{17}$ more routes. As 10 cities took around 1 hour. It would take around $1.7 * 10^{17}$ hours.

4 Hill Climbing

5 Genetic Algorithm

The GA did find the best solution often for 10 cities. When population increased, the solution also improved and with population of 400 for 10 cities, the algorithm usually found the best route. For 24 cities on the other hand, there was a lot more spread in the results.

The running time for 10 cities improved drastically. The exhaustive algorithm took 3357 seconds while the GA took just a few seconds.

The Exhaustive algorithm inspected N! routes.

The GA inspected the initial population size P, and in addition it inspected 50 new routes every generation for G generations. This equals to P+(50*G) solutions.

6 Hybrid Algorithm

I tried implementing the functions by just changing the selection functions. But i struggled with keeping track of the fitness vs the distance of the route. The results were a lot higher than i had hoped.

If I am required to deliver this assignment again, I would appreciate some guidance on how I can make this algorithm work.