Lancaster University

School of Computing and Communications

SCC361 AI Course Work

Introduction:

Marking Scheme

20% of the final mark for the SCC361 module is based on this coursework.

At the end of this document there is an Appendix explaining what a mark means in Lancaster University.

Submission deadline

The deadline for submission of your CW is 6pm, Friday, 11 December, 2020 (end of week 10) and should be submitted electronically on Moodle. The cut-off deadline is 4pm, Monday 14 December, 2020 (with late submission incurring one grade penalty). Submissions after this deadline are not acceptable according to the University regulations.

IMPORTANT:
The implementation should be done in **MATLAB 2017a**. The code should be run by pressing MATLAB **RUN** bottom (**without requiring any changes in the codes**). If your code consists of more than one "`.m`" file, the name of the main file should be "`main.m`". The output should be a figure (as explained below in the CW). The report file should be in **PDF** format. **Don't copy** your code in the report. Put **all** your codes and PDF report in a folder "CW_lastname_firstname.zip" (replace lastname and firstname with your names). Submit your "CW_lastname_firstname.zip" file on Moodle.

A template for the report will be uploaded on Moodle. You just need to fill in the template and convert it to PDF for submission.

Solving TSP using Genetic Algorithm

The *Traveling Salesman Problem* (*TSP*) is one of the most famous problems in computer science. Here we describe the problem and you will implement a Genetic Algorithm (GA) to find a solution, and show and analyse your results. These are to be done in MATLAB. GA has been introduced and discussed as part of the week 6 lecture. There was also a lab about GA in week 6 to give you an initial understanding of the GA approach, but this Task will be applying GA to a different problem than the one in the lab.

TSP consists of attempting to find the shortest complete tour through a series of points (cities), starting and ending with the same point. Finding the shortest route that visits a set of locations is an exponentially difficult problem: finding the shortest path for 20 cities is much more than twice as hard as 10 cities. An exhaustive search of all possible paths would be guaranteed to find the shortest, but is computationally intractable for all but small sets of locations. For larger problems, optimization techniques, such as GA, are needed to intelligently search the solution space and find near-optimal solutions.

Mathematically, traveling salesman problem can be represented as a graph, where the locations are the nodes and the edges (or arcs) represent direct routes between the nodes. The weight of each edge is the distance between the nodes. It is a minimization problem starting and finishing at a specified vertex after having visited each other vertex exactly once. The goal is to find the path with the shortest sum of weights. Below, we see a simple five-node graph:

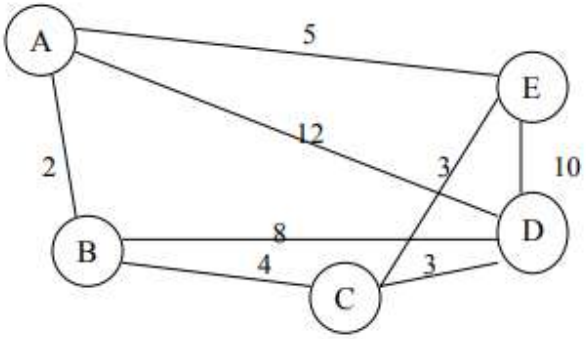


Figure 1- Shortest route example: the problem lies in finding a minimal path passing from all vertices once. For example, the path1 {A, B, C, D, E, A} and the path2 {A, B, C, E, D, A} pass all the vertices but path1 has a total length of 24 and path2 has a total length of 31.

In this task, you will be given the (x,y) location of 100 cities in "`xy.mat`" file. So, each population member (chromosome) will have 100 genes.

Finding a solution to the travelling salesman problem requires that you set up a genetic algorithm in a specialized way. For instance, a valid solution needs to represent a route where every location is included at least once and only once. If a route contains a single location more than once, or missed a location out completely it would not be valid. To ensure the genetic algorithm does indeed meet this requirement special types of mutation and crossover methods are needed. Firstly, the mutation method should only be capable of shuffling the route, it shouldn't ever add or remove a location from the

route, and otherwise it would risk creating an invalid solution. You also need to pick a crossover method which can enforce the same constraint.

Implement Genetic Algorithm

Your main task is to implement with MATLAB a genetic algorithm that attempts to find a near-optimal solution. You cannot use MATLAB's "ga" function, so you have to implement something similar to what you did in the lab in week 6.

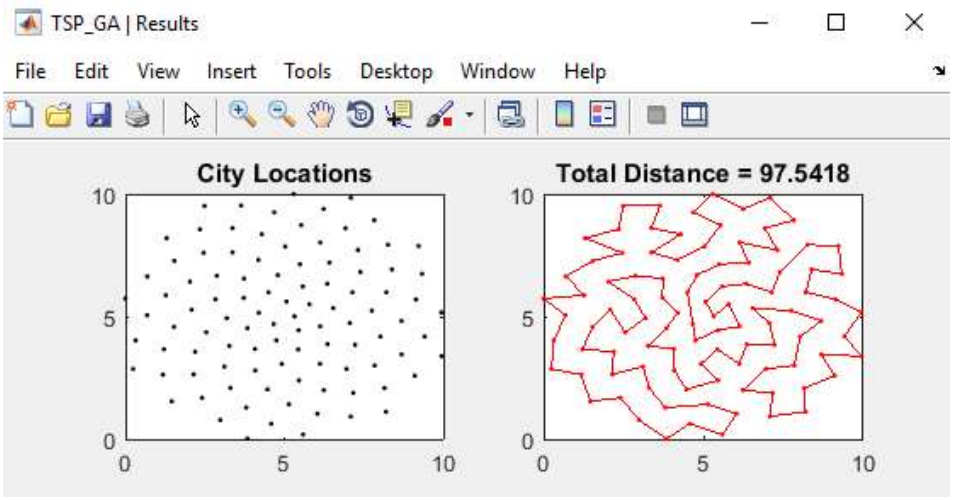
Your algorithm should make use of crossover and mutation as described above. Begin with an initial population of at least 50 members and then increase to 200 members (start with 50 members, then try 100, 150 and 200 members). Run your algorithm for at least 1000 generations/iterations and then increase to 10000 (start with 1000 generations, then try 2000, 4000, 6000, 8000, and 10000 generations/iterations). Choose the best ones.

You will need to make many design decisions on how to implement the algorithm and what parameter values to use. For example, you could try different selection methods including roulette-wheel selection and tournament selection to see which one is better. **Submit the all the written codes**. Your mark will depend not only on the code that you write but also on how well you document your design decisions. In your report (**a template will be uploaded on Moodle**), you should answer the following questions:

1- After running your GA code for 10 times, what was the average fitness value? More precisely, you need to save the best fitness value for each of 10 runs and then take their average.

2- What path did the most-fit individual in the final generation take through the cities? Run the following code to visualize the path of the most-fit individual in the last generation/iteration. Note that "`xy`" variable is a 100×2 matrix consisting of the (x,y) location of 100 cities and `optRoute` variable (integer array) is the best route found by the algorithm (i.e., the most-fit individual in the final generation). `optRoute` is 1×100 vector. This code will show a figure as shown below but the connections between cities and total distance might be different.

```
figure('Name','TSP_GA | Results','Numbertitle','off');
subplot(2,2,1);
pclr = ~get(0,'DefaultAxesColor');
plot(xy(:,1),xy(:,2),'.','Color',pclr);
title('City Locations');
subplot(2,2,2);
rte = optRoute([1:100 1]);
plot(xy(rte,1),xy(rte,2),'r.-');
title(sprintf('Total Distance = %1.4f',minDist));
```



3- Run your GA algorithm 10 times without cross-over operator and report the average fitness value. You need to save the best fitness value for each of 10 runs and then take their average.

4- Run your GA algorithm 10 times without mutation operator and report the average fitness value. You need to save the best fitness value for each of 10 runs and then take their average.

5- Report the optimal parameters of your GA algorithm. The parameters include: selection method, cross-over method, mutation method, cross-over probability, mutation probability, number of generations/iterations, number of chromosomes in population,

Marks allocation for the course work:

Code efficiency (Time complexity)	**30**
Code commenting and writing style	**10**
Results (average of 10 runs should be close to optimal)	**40**
Presentation and writing of the report	**20**
Total:	**100**

Appendix

What a Mark Means in Lancaster University

A (Distinction)

Critical Understanding of Topic

Excellent understanding and exposition of relevant issues; insightful and well informed, clear evidence of independent thought; good awareness of nuances and complexities; appropriate use of theory.

Structure of Research

Substantial evidence of well implemented independent research and / or Substantial evidence of well selected evidence to support argument.

Use of Literature

Excellent use of literature to support argument /points.

Conclusion

Excellent; clear implications for theory and/or practice.

Language

Excellent; a delight to read.

Structure and Presentation

Arguments clearly structured and logically developed; sensible weighting of parts; meaningful diagrams; properly formatted references.

B (Very Good Pass)

Critical Understanding of Topic

Clear awareness and exposition of relevant issues; some awareness of nuances and complexities but tendency to simplify matters; based on appropriate choice and use of theory.

Structure of Research

Some evidence of independent research reasonably well implemented and / or some evidence of identification of suitable evidence to support argument.

Use of Literature

Good use of literature to support arguments.

Conclusion

Very good; draws together main points; some implications for theory and/or practice

Language

Carefully written; negligible errors.

Structure and Presentation

Arguments clearly structured and logically developed; good weighting of parts; meaningful diagrams; properly formatted references.

C (Good Pass)

Critical Understanding of Topic

Shows awareness of issues and theories; attempts at analysis but tendency to lapse into description

Structure of Research

Some evidence of independent research reasonably well implemented and / or some evidence of identification of suitable evidence to support argument.

Use of Literature

Use of standard literature to support arguments.

Conclusion

Reasonable conclusion that summarises essay; a few implications for theory and/or practice.

Language

A few errors; generally satisfactory.

Structure and Presentation

Arguments reasonably clear but undeveloped; some meaningless diagrams or poor structure.

D (Pass)

Critical Understanding of Topic

Work shows understanding of topic but at superficial level; no more than expected from attendance at lectures; some irrelevant material; too descriptive.

Structure of Research

Insufficient evidence of independent research and / or very limited evidence used to support argument.

Use of Literature

Use of secondary literature to support arguments.

Conclusion

Conclusion does not do justice to body of essay; too short; no implications.

Language

Some errors; grammar and syntax need attention.

Structure and Presentation

Arguments not very clear; poor organisation of material; poor use of diagrams; poor referencing.

F4 (Marginal Fail)

Critical Understanding of Topic

Establishes a few relevant points but superficial and confused; much irrelevant material; very little or no understanding of the issues raised by the topic or topic misunderstood; content largely irrelevant; no choice or use of theory; essay almost wholly descriptive; no grasp of analysis with many errors and/or omissions.

Structure of Research

No evidence of independent research and / or No attempt to identify suitable evidence to support argument.

Use of Literature

Relies on a superficial repeat of class notes.

Conclusion

No recognisable conclusion.

Language

Frequent errors; needs urgent attention.

Structure and Presentation

Arguments often confused and undeveloped; no logical structure; very poor organisation of material; many meaningless diagrams; negligible referencing.

F1-F3 (Clear Fail)

Critical Understanding of Topic

Establishes a few relevant points but superficial and confused; much irrelevant material; very little or no understanding of the issues raised by the topic or topic misunderstood; content largely irrelevant; no choice or use of theory; essay almost wholly descriptive; no grasp of analysis with many errors and/or omissions.

Structure of Research

No evidence of independent research and / or No attempt to identify suitable evidence to support argument.

Use of Literature

No significant reference to literature.

Conclusion

No recognisable conclusion.

Language

Frequent errors; needs urgent attention.

Structure and Presentation

Arguments often confused and undeveloped; no logical structure; very poor organisation of material; many meaningless diagrams; negligible referencing.