



CSE 3353 ALGORITHMS: LAB 5

Overview: Using both Simulated Annealing (SA) and Particle Swarm Optimization (PSO) techniques, solve the traveling sales man problem (TSP) (find a Hamiltonian circuit) for a given list of nodes and positions (graph). Compare and contrast your results with those from Lab 3 and 4.

Due: December 12, 2018 @ 3pm. *This lab is not required*

If completed, it will be used in whichever following scenario best benefits student.

1. Student has ≥ 90 average: If completed final is optional, if no final is taken grade will be calculated out of total of 875 points. (Midterm (150), Discussions (100), Labs (625, 125/ea))
2. If completed, top 4 lab grades will be used in course average.

Code Requirements:

- Always use Node 1 as your start/finish point of TSP
- Read node list in from a text file which uses same format as the positions.txt file from Lab 2
 - NodeID x, y, z \rightarrow 1,1.23,3.45,4.112 \rightarrow int,float,float,float
 - You can derive your own list of nodes and positions
 - Code will be tested against a private list of nodes/positions. Code will only be tested up to the number of nodes identified in writeup as total nodes able to analyze.
- Return optimal (shortest) path for given list of nodes
- Start with a list of 4 nodes and increase to the maximum number that is feasible with your computer for both PSO and SA algorithms
- When coding SA, you must identify and test multiple techniques for:
 - How and when T is decreased
- When coding PSO, you must identify and test multiple techniques for:
 - Different learning factors
 - Different Max Velocities
- Design your code to be compatible with system designed in Lab 3 and 4. If any modifications to the architecture are made, document and describe why and what modifications were made.

Report Requirements:

- Plot timing with excel for each algorithm in a single graph. X axis should be number of nodes in graph, Y axis should be total timing.
 - Also provide a table to summarize data in graph
 - This should include results from Lab 3 for comparison
- Provide a summary of results obtained and be sure to explain time complexity for each algorithm
 - Provide a graph that shows a comparison of timing captured on code execution vs the asymptotic timing determined for given algorithm
- Explain and document your design decision for your specific algorithm implementations. Should include write up that explains architecture decisions and how you designed for functionality and extensibility as well as a UML diagram.
- Explain and discuss in detail how you're the variations of PSO and SA caused changes in the outcomes and be sure to explain why those modification/changes caused the output obtained.

Submit inside of your class git repo.