Artificial Intelligence

CS21B040,CS21B024

→ TSP using simulated annealing and Variable neighborhood search

Compilation:

c++ -std=c++11 -O3 solution.cpp -o solution
or run
make compile

Running a test case:

./solution ./test_case_tsp_file.tsp SEED
Ex: ./solution ./problems_cleaned/ch130.tsp
SEED value is optional

To run all given test-cases:

make runall
run this before this step c++ -std=c++11 -O3 solution.cpp -o solution and run for a test case.
VNS Section:

```
[pavan@fedora TSP]$ c++ -std=c++11 -O3 solution.cpp -o solution
[pavan@fedora TSP]$ ./solution ./problems_cleaned/ch130.tsp
------------------------------Variable Neighborhood Search---=----------------------------------
130
BEST KNOWN IS 6110
130==130
Best_known cost: 6110

time 1714108607
Initial solution: 77 126 49 105 114 68 88 48 102 90 64 124 89 42 130 9 53 57 60 112 50 119 2 95 55 117 45 62 37 22 32 122 36
30 76 111 66 79 129 12 93 56 28 73 67 10 3 19 44 123 61 101 34 94 29 115 85 100 84 25 38 121 23 65 18 52 109 59 4 96 58 120 6
 104 71 46 116 17 51 128 80 14 108 125 82 99 39 103 31 98 86 70 97 87 24 33 1 5 110 35 54 43 92 47 11 118 81 91 63 8 40 26 72
 16 106 20 27 7 107 113 75 127 15 78 69 83 13 74 41 21
Initial cost: 46575
Final cost: 6110
77 94 89 110 98 68 63 48 25 113 32 36 84 119 111 123 101 82 57 9 56 65 52 75 74 99 73 92 38 106 53 120 58 49 72 91 6 102 10 1
4 67 13 96 122 55 60 51 42 44 93 37 22 47 40 23 33 21 126 121 78 66 85 125 90 59 30 83 3 114 108 8 18 46 80 118 20 4 35 54 2
50 130 71 1 41 39 117 112 115 28 62 105 128 16 45 5 11 76 109 61 129 124 64 69 86 88 26 7 97 70 107 127 104 43 34 17 31 27 19
 100 15 29 24 116 95 79 87 12 81 103
-------------------------------------------Simulated Annealing----------------------------------
```

This will print **initial random vector** generated and its initial cost
after VNS the final vector and its cost

SA and Stats section:

```
----------------------------------------Simulated Annealing----------------------------------
130
BEST KNOWN IS 6110
130==130

time 1714108608
Initial solution: 77 126 49 105 114 68 88 48 102 90 64 124 89 42 130 9 53 57 60 112 50 119 2 95 55 117 45 62 37 22 32 122 36
30 76 111 66 79 129 12 93 56 28 73 67 10 3 19 44 123 61 101 34 94 29 115 85 100 84 25 38 121 23 65 18 52 109 59 4 96 58 120 6
 104 71 46 116 17 51 128 80 14 108 125 82 99 39 103 31 98 86 70 97 87 24 33 1 5 110 35 54 43 92 47 11 118 81 91 63 8 40 26 72
 16 106 20 27 7 107 113 75 127 15 78 69 83 13 74 41 21
Initial cost: 46575
Final cost: 8729
104 34 54 46 18 108 8 21 33 22 37 47 40 44 93 20 71 39 115 28 66 85 125 90 78 23 122 55 6 102 10 14 67 13 96 60 91 65 56 57 8
2 123 111 101 9 52 75 74 99 73 81 29 15 94 89 110 98 113 32 36 84 119 77 103 92 72 49 58 53 106 38 12 87 95 116 100 27 31 17
2 1 45 5 11 129 124 88 26 7 97 70 63 68 25 48 86 69 64 61 109 76 16 128 62 83 126 121 59 30 114 3 117 112 105 41 130 50 80 11
8 35 4 42 51 120 79 24 19 43 107 127
----------------Stats----------------
Best Known Solution: 6110
VNS Solution: 6110
SA Solution: 8729
Better to use VNS for this
[pavan@fedora TSP]$
```

This will print the initial random vector and sa vector and their cost values
and stats of best known ,vns ,sa costs at the end.

At the end we also store these route information in **vns_output.txt , sa_output.txt** file these files are must to visualize data

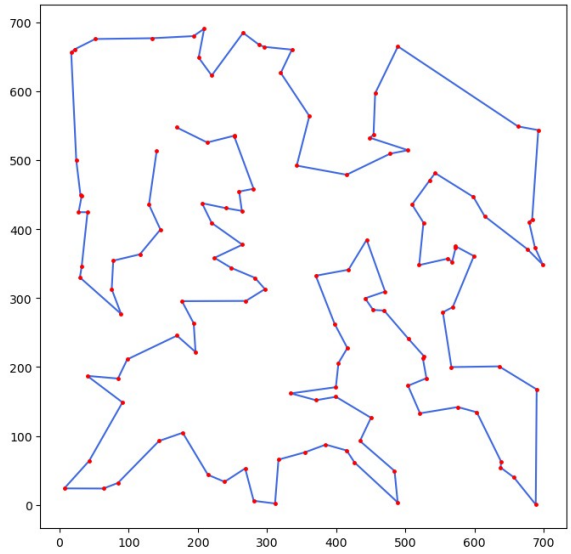
DATA VISUALIZATIONS

Open plotGraphs.ipynb

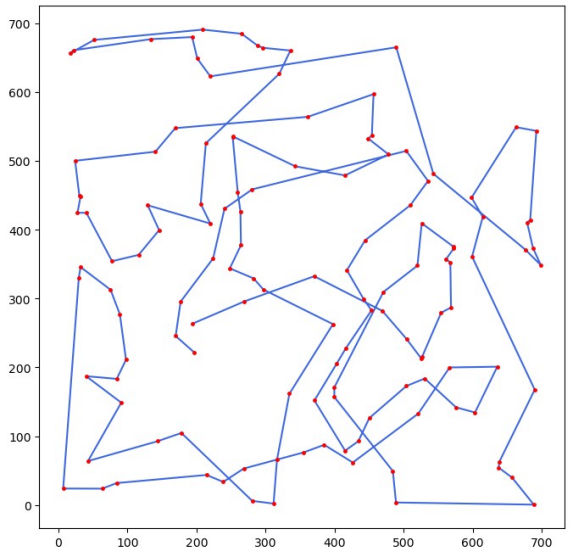
vns_output.txt and sa_ouput.txt files are must to run this file.

Second cell gives the TSP route with VNS vector and Third cell gives TSP Route with SA vector

for ch130.tsp test case

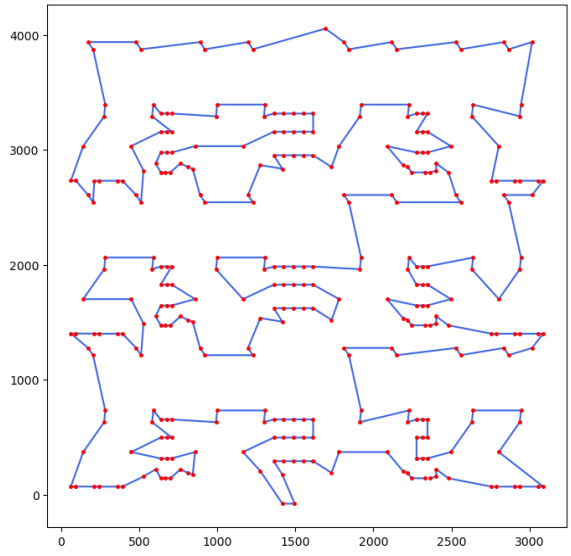


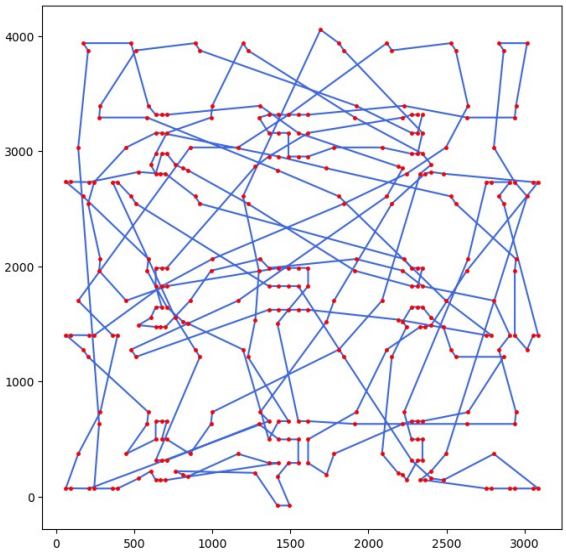
VNS Graph



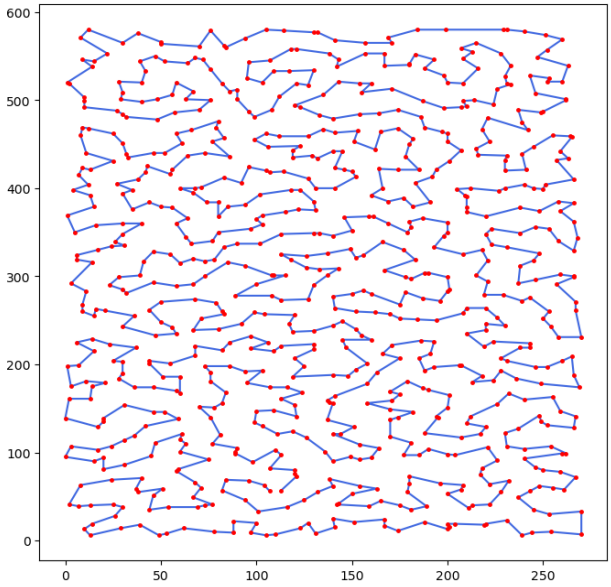
SA Graph

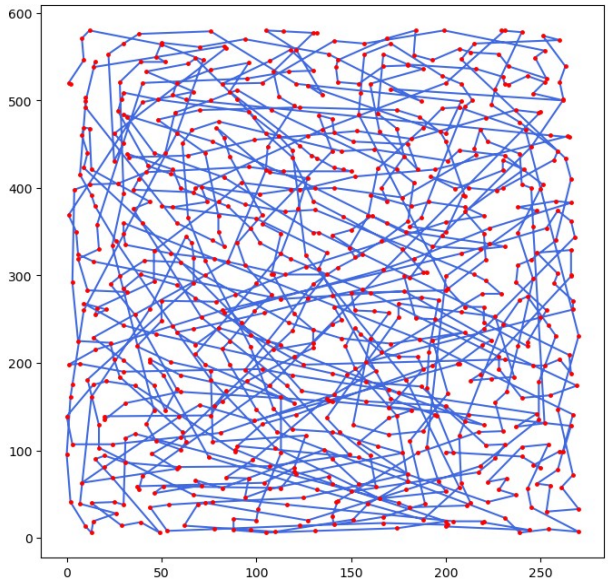
for lin318.tsp test case





for rat783.tsp test case





--Thank You--