AI - LAB
Assignment-4
TRAVELLING SALESMAN PROBLEM
Report

Arvind Kumar M - 200020008

Tarun Saini - 200010051

February 6, 2022

Contents

1 Introduction

Here we are finding optimal tour cost for Travelling Salesman Problem.
TSP is a problem where we have to find optimal cost by travelling each city only once and coming back to the start city.

We are finding that using greedy constructive method, Ant Colony Optimization, 2 and 3 city exchanges.

2 State space

In this problem for n cities, state space is a n size list where each element represents the index of that city.
The total possible state spaces are n! which is greater than exponential.
So TSP is an NP-Complete problem.

3 Pseudo code for Greedy()

def greedy():

for i in range(100)

CLOSED =[ ]

startcity $\rightarrow i$

$CLOSED.append(city)$

$for\ each\ j\ not\ in\ CLOSED:$

$nextcity = min(distance_{ij})$

$current \rightarrow j$

$tours.append(CLOSED)$

4 Ant Colony optimization

We are using Ant Colony Optimization only if there are 100 or less number of cities.
If there are more cities ACO is not giving optimal and taking time to find each tour .

So considering time constraint we are not using ACO for more than 100 cities. For initial phermone, we are using minimum cost greedy tour to get optimal convergence in ACO.

Pseudo code for ACO

def ACO():
while(time <vertices/4):
for n in range(no_of_ants):

tours =[ ]

cities = [i for i in range(vertices)]

startcity $\rightarrow$ *random.generate*()

cities.remove(*current*)

for j in range(*vertices* $-$ 1) :

for k in cities :

probability[*current*, *k*] = *pheromone*[*current*][*k*]∗∗*alpha*)((1/*v_dist*[*current*][*k*])∗
∗*beta*

next_city = *random.choice*(*cities*, *p* = *probability*)

tours.append(*next_city*)

current $\rightarrow$ *next_city*

for (*i*, *j*) *in tours*

pheromone[*i*, *j*] = *q*/*tour_cost*

for each edge :

pheromone[*i*, *j*] = *delta* ∗ *old_pheromone*[*i*, *j*] + *pheromone*[*i*, *j*]

5 ACO Parameters

Based on trials, we have found that below values for parameters give optimal value for tour cost.

Values for parameters

- Number of ants - 100
- Alpha - 1
- Beta - 5
- Evaporation rate - 0.7
- Delta - 1 - 0.7 = 0.3
- Q(pheromone) = 35*cities/100

6 Cities Exchange

- After doing ACO we call 2 city exchange to get minimum cost tour by exchanging two cities.
- After that to get more minimum tour cost we are doing 3 cities exchange.
- By doing these two we get better results.

Iterative Approaches

Ant Colony Optimization

- We did ACO with random start city and initial pheromone same in every city.
- We got 1800 for 100 cities in test case.

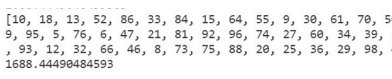
Greedy

- Then we tried only greedy constructive method
- It did not gave better results than ACO.
- We got 1879 as minimum tour cost for 100 cities.

Greedy + ACO

- Then we implemented both greedy and ACO together.
- We first run greedy and updated pheromone in than path and took it as initial pheromone.
- In that we got around 1700 as minimum cost tour for 100 cities test case.

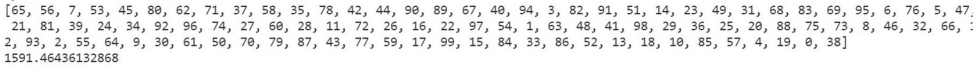
Below is the image of least cost got in test case got ACO + greedy.



Cities exchange

- After that we did 2 and 3 cities exchange for the minimum tour we got in ACO.
- In that we got around 1600 as minimum cost tour.

Below is the image of least cost got in test case got ACO + greedy + cities exchange.



We are running the whole program for 280 secs since only 300 secs is given in the constraint.