TRAVELLING SALESMAN PROBLEM

Lokesh Nirania 170010009

Sudarshan Das 170010010

GROUP 4

SIMULATED ANNEALING

a. **Pertubation Method :-**
    a. Randomly select two cities in the path and swap them
    b. If the new path length is acceptable (using the prob function), we accept the new path and continue operate on the new path
b. **Cooling Schedules Tried :-**
    a. T = T * 0.99 , This is the most effective decreasing function
    b. T = T * 0.9 , Less optimal solution but faster than above function
    c. T = T – 1, It gives the least optimal solution
c. **Results**

T initial = n = number of cities

T = T * 0.99

Program	Peturbation	T saturation	Path Length
euc_100	Till first decrement	0.29404436871220024	3062.2909396091095
	n*n times	0.4395467595536361	1753.1535365482703
	n*n times	0.3595083440373008	1687.071822195381
euc_250	Till first decrement	0.09088105324281398	7600.9268091163685
	n times	0.472390807155341	5398.980572988858
	n times	0.15637698778708906	4647.2549600171615
noneuc_100	Till first decrement	1.301379814397157	5911.951338557698
	n times	8.696655909824688	6134.317684204799
	n*n times	3.183257323920795	5584.665589712602

T = T - 1

Program	Peturbation	T saturation	Path Length
euc_100	Till first decrement	1	6785.982759891252
	n*n times	1	2110.871473623961
	n*n times	1	2052.1278344105804
euc_250	Till first decrement	1	18684.132887890537
	n times	1	8927.85131987207
	n times	1	9356.518526350215

noneuc_100	Till first decrement	1	7704.250411858406
	n times	1	6530.2888919464
	n*n times	1	5765.760855450199

d. Effect of cooling schedule on tour found

Slow rate of decrease in temperature guarantees more optimal solution

GENETIC ALGORITHM

a. Representation Chosen

We represented tours as Path because we found it easier to implement and easier to get drawing/visual idea

b. Crossover Operator Chosen

We have chosen OX (ordered crossing) as it gave better results for our implementation

c. Results

k = 70 %

Test Case	Population Size	Generation number	Path Length
euc_100	n/10 = 10	1093	4167.344157979831
		2586	3035.972511064289
		5073	2528.93858622859
		10025	2099.9455592378395
		21896	1817.4326915817303
		51040	1625.3188135742705
	n/2 = 50	1041	3001.92066708097
		2614	2331.9301679896903
		5725	2187.7444122069705
		10794	2129.05790043601
euc_250	n/10 = 25	514	17439.864836035154
		1019	14322.074627478563
		2507	10113.498728815983
		5005	8096.357694063929
		10172	6301.187572229357
		25952	5589.012885003368
	n/2 = 125	511	13121.00835806214
		1004	9990.957089268202
		2521	6571.7507764144375
		5001	5064.7014993844605
		11397	3423.072440149471
		20051	2907.752941596558
noneuc_100	n/10 = 10	543	7065.2553688042
		1065	6526.334514038301

		2145	6225.898429617802
		5096	5874.680914749001
		10366	5746.363250719701
		27503	5564.443070894403
		54921	5519.237783334001
	n/2 = 50	504	6422.875656425398
		1008	6115.710257208798
		5058	5746.033614022199
		10035	5613.807342820898
		23871	5521.8093344627005

d. Effect of population size

i. Optimal solution is found early for larger population size

ii. Each generation takes longer time to generate for larger population size

ANT COLONY

a. Results

Program	No of Ants	Path Length
euc_100	100	1865.2489
	15	2592.95
	30	2236.4279
	50	2009.58473
noneuc_100	15	5623.8895
	30	5448.8745
euc_250	15	3899.3520
	30	3354.5695

CONCLUSION

a. Generally Simulated Annealing gives good solution but as number of cities increases, it becomes more slower. If we choose appropriate monotonically decreasing T function, it gives satisfiable solution.

b. Genetic Algorithm gives very good solution but it all depends upon our intuition of choosing optimal mutation function, crossover function, population size and k replacing members. It's very complicated to fine tune all these parameters

c. Ant Colony Optimisation depends on the number of Ants and the number of Iterations. Therefore it takes a lot of time for all the ants to complete their tours and update the pheremones. But with increase in iteration, the solutions tends to its optimal value. Also with decrease in number of Ants, not all its path are travelled. Hence, the solution thus obtained will not be as good as the previous case.

d. **PERSONAL PREFERENCE** :- According to the observation, genetic algorithm is out preferred one, but however if we get enough compute power, we will go with ant colony optimisation