ISSN 22773061

# Genetic algorithm based approach to solve travelling salesman problem with one point crossover operator:

Prof. Sharadindu Roy[1]
Asst. Professor, Dept. of computer science,
Sonarpur Mahavidyalaya, University of Calcutta, India.
Email: sharadinduroysonarpur@gmail.com

**Abstract:**

In this paper, the travelling salesman problem using genetic algorithm has been attempted. In this practical paper solution is easy and we can easily apply genetic operator in this type of problem. Complexity is both in time and space, provided size of the problem an as integer (count is infinite). The solution of the traveling salesman problem is global optimum. There are cities and given distances between them. Traveling salesman has to visit all of them. TSP main objective is to find traveling sequence of cities to minimize the traveling distance.* traverse one time*initially we select parent1 & parent2 by Roulette wheel concept. Apply one point crossover operator on parents and produce the offspring. Again we apply the mutation operator on offspring and created child. But the no. of bits (cities) will be inverted by the mutation operator, that is depended on mutation probability ($p_m$). So one generation contain 6 individual. Then count fitness of the individuals in each generation. For the next generation (for parent1 & parent2) two individuals will be selected whose fitness is best in generation. Here we see crossover between two good solution may not always yield a better or as good a solution. Since parents are good, so the probability of the child being good is high. Every time we have to do, identity the good solution in the population and make multiple copies of the good solution.

**Index terms:** Genetic algorithm, Crossover, Mutation, Travelling salesman problem, NP-complete, Cost matrix, Fitness.

**Academic Discipline And Sub-Disciplines**: computer science, soft computing.

**SUBJECT CLASSIFICATION**: computer science, genetic algorithm.

## Genetic algorithm:

The genetic algorithm involves the following basic steps-
Evaluation.
Crossover.
Mutation.

## Crossover:

**One point crossover :-**

Part of the first parent copied rest is taken in the same order as in second parent.

**Two point crossover :-**

Two part of the first parent are copied and the rest between is taken in the second parent in same order.

## Mutation:

A few genes are chosen and exchanged. But the number of bit will be changed depends on mutation probability.
Mutation probability (Pm):-

Mutation rate is the probability of mutation which is used calculates number of bits to be muted. The mutation operator preserves the diversity among the population which is also very important for the search. Mutation probabilities are smaller in natural population leading us to conclude that mutation is appropriately considered a secondary mechanism of genetic algorithm adoption.

Individual→011101001011 and Pm = 0.25

Then we say, 3 bit will be inverted.

## Cost matrix:

Cost matrix contains distances between one city to another cities. Cost matrix contain zero ('0') value on the main diagonal.

## Fitness:

Fitness means distances between cities

## Introduction:

Traveling salesman problem (TSP) is one of the old problems in computer science and operations Research. This problem is:A graph with 'n' nodes (or cities), with 'node 1' as a 'headquarters' and travel cost (or distances, or travel time etc.,) matrix C=[$c_{ij}$] of order n associated with ordered node pairs (i, j) is given. The problem is to find a minimum cost Hamiltonian cycle. The search space contains N! Permutations and since TSP is NP-complete and the corresponding optimization problems are therefore NP-hard. The problem with this representation is obvious. Starting with a population of valid chromosomes, ordinary crossover and mutation operators cause problems. In this algorithm we use the one point crossover operator but mutations are not performed at single points. Here, simple bit-string crossover and **Ideas related to the TSP have been around for a long time: In 1736, Leonard Euler studied the problem of finding a round trip through seven bridges in K¨onigsberg. In 1832, a handbook was published for German travelling salesmen, which included examples of tours. In the 1850s, Sir William Rowan Hamilton studied Hamiltonian circuits in graphs. He also marketed his 'Icosian Game', based on finding tours in a graph with 20 vertices and 30 edges.**

## Solution methodology:

For the TSP, solution is typically represented by chromosome of length as the number of nodes in the problem. Each gene of a chromosome takes a label of node such that no node can appear twice in the same chromosome. There are mainly two representation methods for representing tour of the TSP **-** adjacency representation and path representation. We consider the path representation for a tour, which simply lists the label of nodes. For example, let {1,2,3,4,5} be the label of nodes in 5 node instances, then a tour {1→3→4→2→5→1} may be represented as (1,3,4,2,5).
At first the parent1 & parent2 will be selected via roulette wheel selection concept. Here we apply crossover operator that will be – part of the first parent is copied and the rest is taken in the same order as in the second parent.

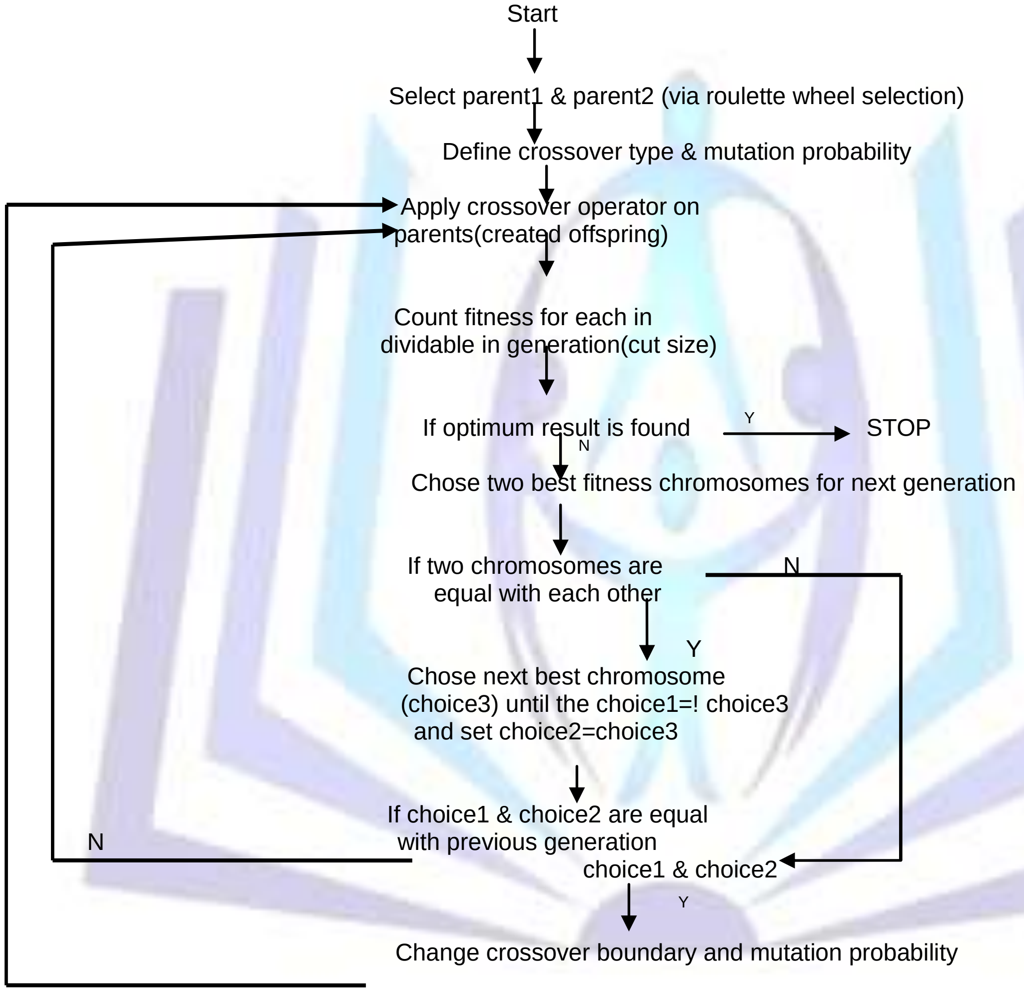
For example:
The 6 cities- 1) Mumbai, 2) Chennai, 3) Delhi, 4) Bengaluru, 5) Hydra bad, 6) Pune.
We take the any sequence of the 6 cities, that can be-

- 1 3 6 2 4 5
- 3 2 1 5 6 4
- 2 5 3 4 6 1
- 1 5 3 2 4 6

If, fitness of the chromosomes are respectively 45, 43, 11,78 then roulette wheel select 3 and 2 no. chromosome. Suppose the chromosome "1 3 6 2 4 5" , For this chromosome the fitness will be calculated as a following process------------

Fitness=1 to 3 distance + 3 to 6 + 6 to 2 + 2 to 4 + 4 to 5 + 5 to 1 distance.

## Algorithm:



## Experimental result:

There are 8 cities. The travelling cost between cities shows is in table that is called cost matrix. Cost matrix =

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 28 | 75 | 99 | 9 | 35 | 63 | 10 |
| 2 | 51 | 0 | 46 | 85 | 88 | 29 | 20 | 22 |
| 3 | 100 | 5 | 0 | 16 | 28 | 35 | 27 | 32 |
| 4 | 20 | 45 | 11 | 0 | 59 | 53 | 49 | 11 |
| 5 | 86 | 63 | 33 | 65 | 0 | 76 | 72 | 22 |
| 6 | 36 | 53 | 89 | 31 | 21 | 0 | 52 | 81 |
| 7 | 58 | 31 | 43 | 67 | 52 | 60 | 0 | 99 |
| 8 | 22 | 32 | 41 | 58 | 65 | 88 | 72 | 0 |

The genetic algorithm has been used the minimized the travelling cost between many cities. The coding has been done using MATLAB r2012a (7.3 versions). The above example contains 8 cities, and the proposed algorithm is applied by the MATLAB r2012a on above cities and run the programme and takes the result after 10 generation.

The genetic algorithm has been used the minimized the travelling cost between many cities. The coding has been done using MATLAB r2012a (7.3 versions). The above example contains 8 cities, and the proposed algorithm is applied by the MATLAB r2012a on above cities and run the programme and takes the result after 10 generation. The result are shown in tabular form

| No_of_iteration | Min_cut | Avg_cut | Max_cut |
|---|---|---|---|
| 10 | 228 | 288 | 504 |
| 20 | 228 | 279 | 470 |
| 30 | 226 | 292 | 515 |
| 40 | 222 | 271 | 403 |
| 50 | 222 | 273 | 403 |
| 60 | 208 | 248 | 379 |
| 70 | 208 | 252 | 379 |
| 80 | 208 | 250 | 379 |
| 90 | 191 | 268 | 377 |
| 100 | 191 | 274 | 377 |

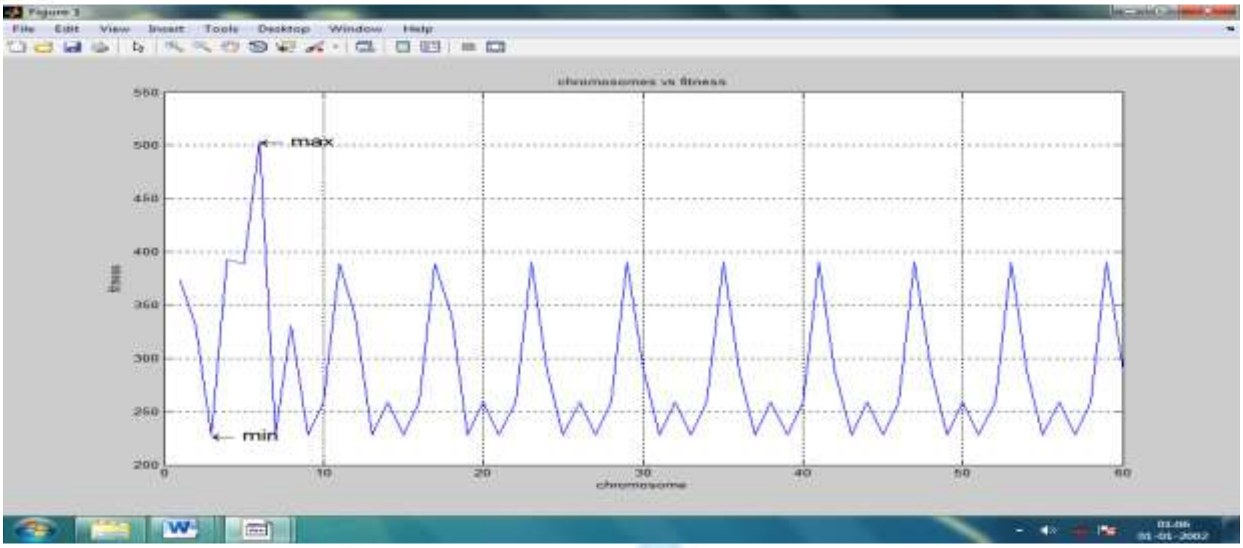



Fig.1 plot for chromosome vs fitness in 10 generation.

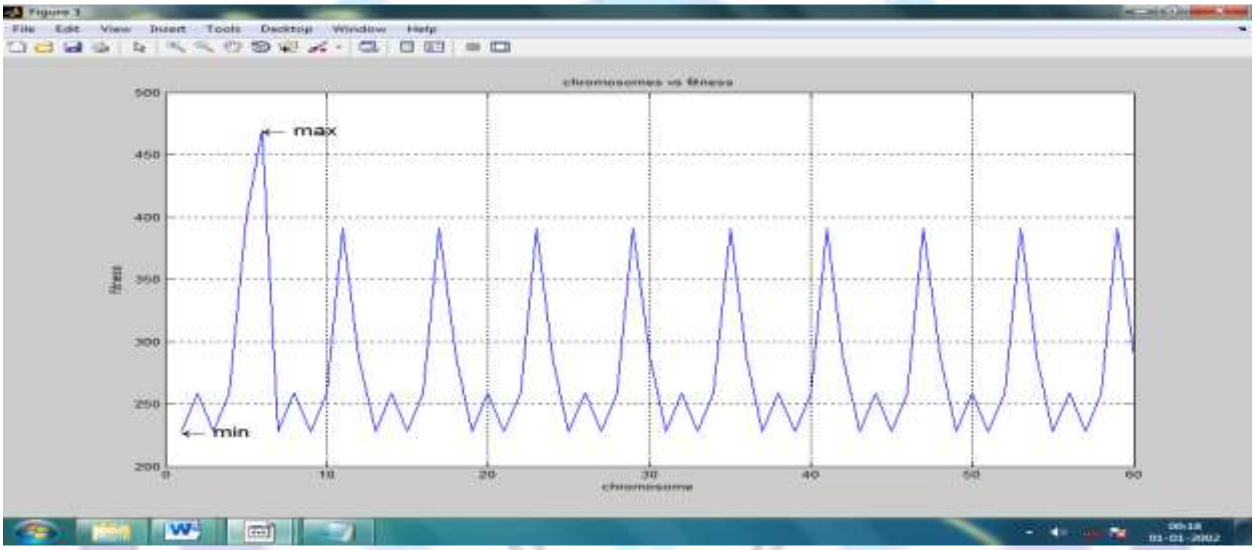



Fig2: plot for chromosome vs. fitness in generation 11 to 20

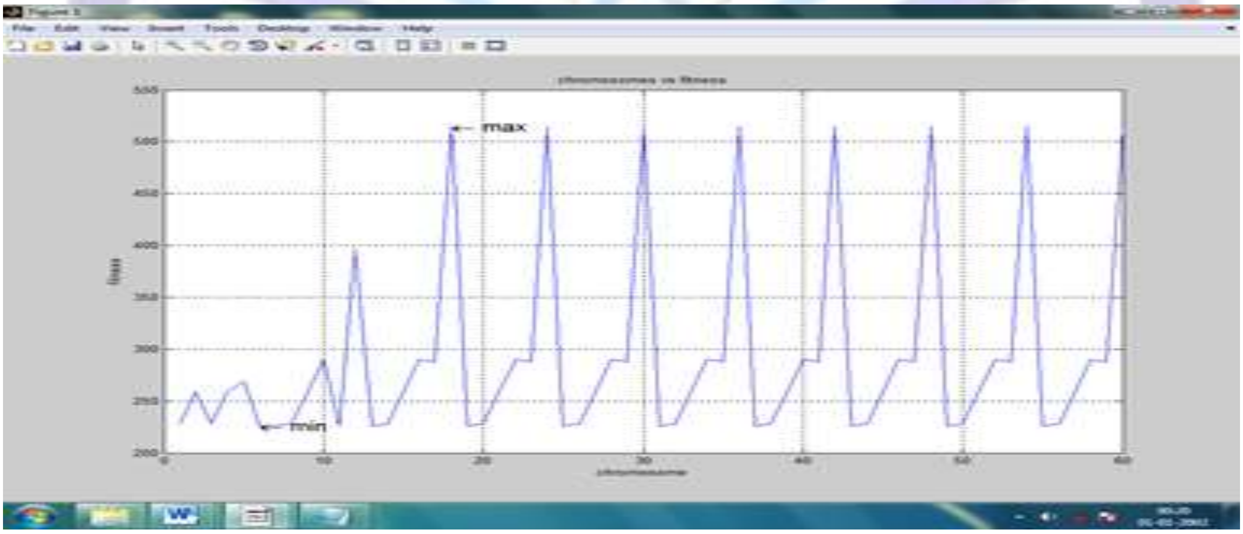



Fig3: plot for chromosome vs. fitness in generation 21 to 30

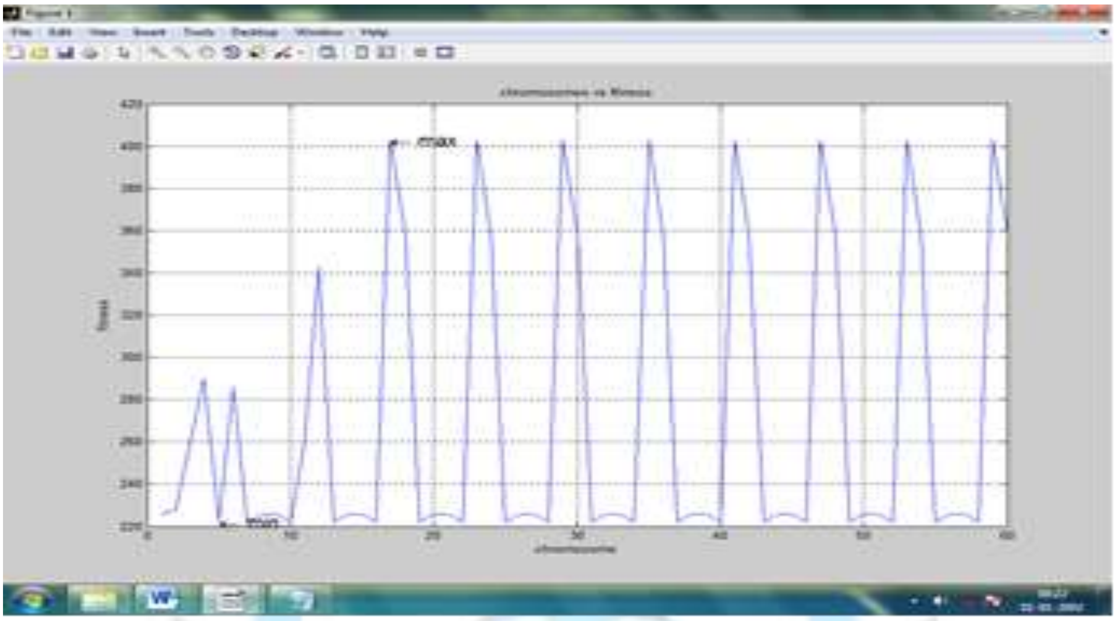



Fig4: plot for chromosome vs. fitness in generation 31 to 40

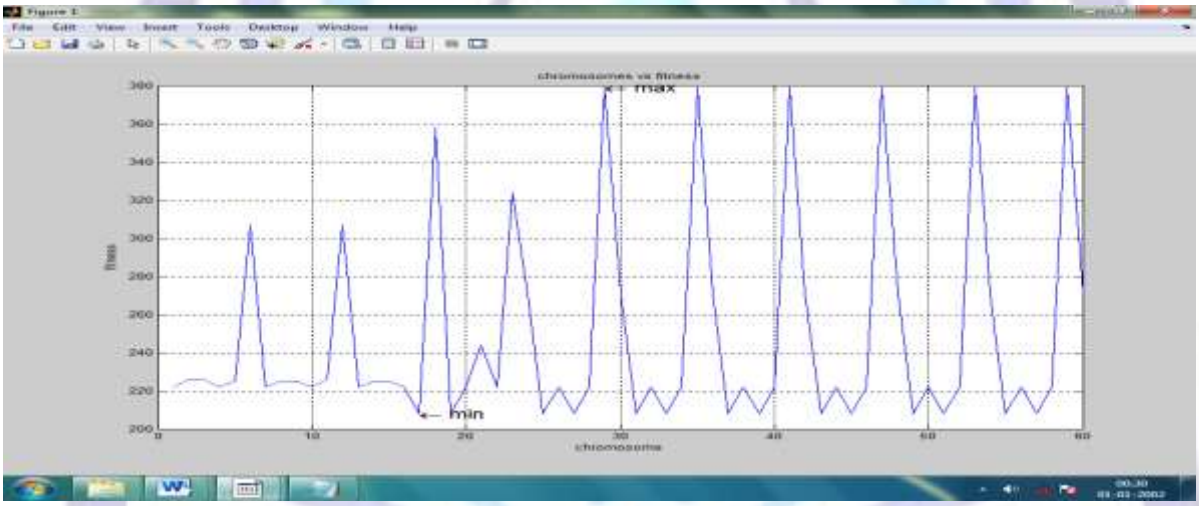



Fig5: plot for chromosome vs. fitness in generation 51 to 60

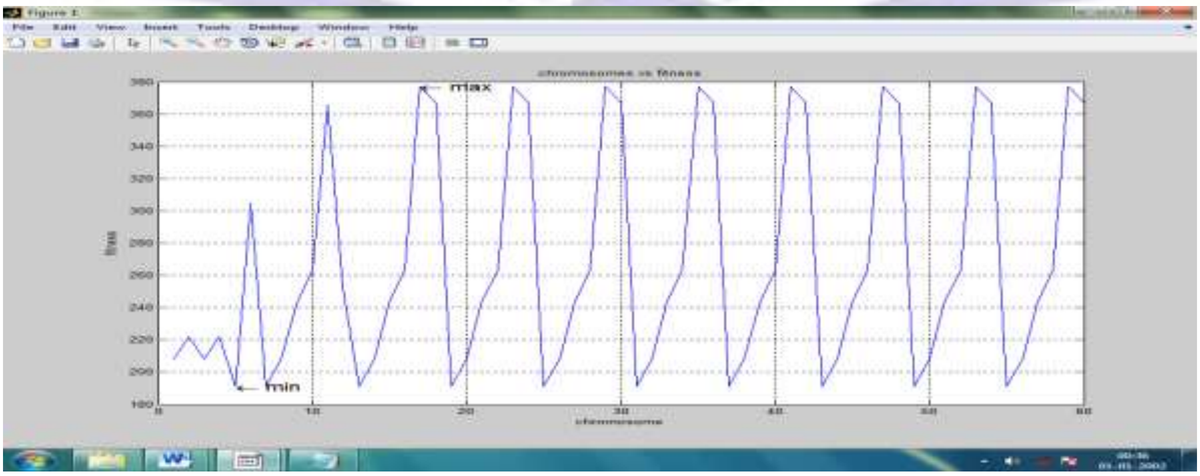



Fig6: plot for chromosome vs. fitness in generation 81 to 90

ISSN 22773061

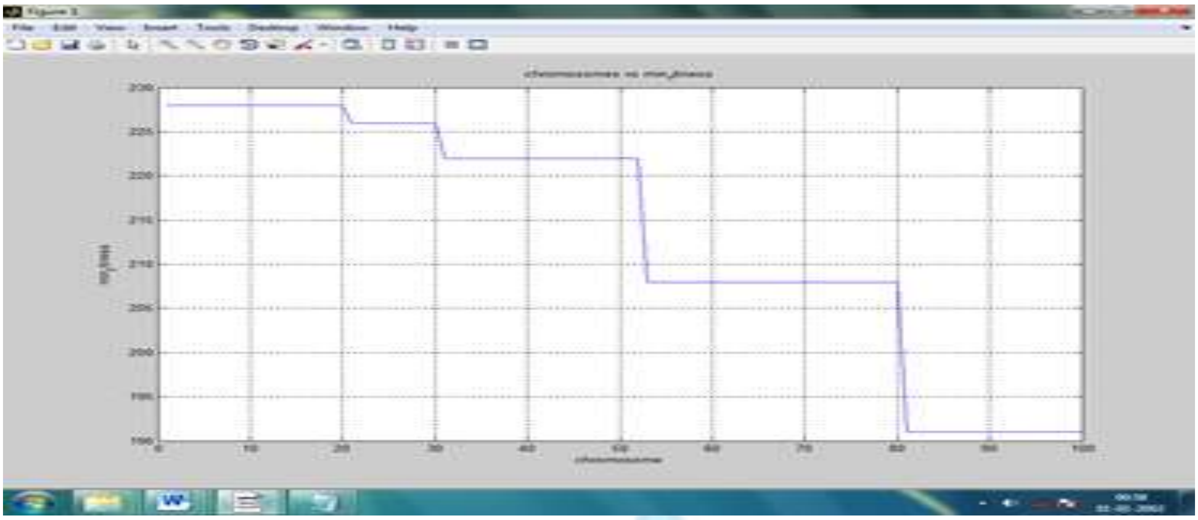



Fig7: plot for chromosome vs. minimum_ fitness in 100 generation.

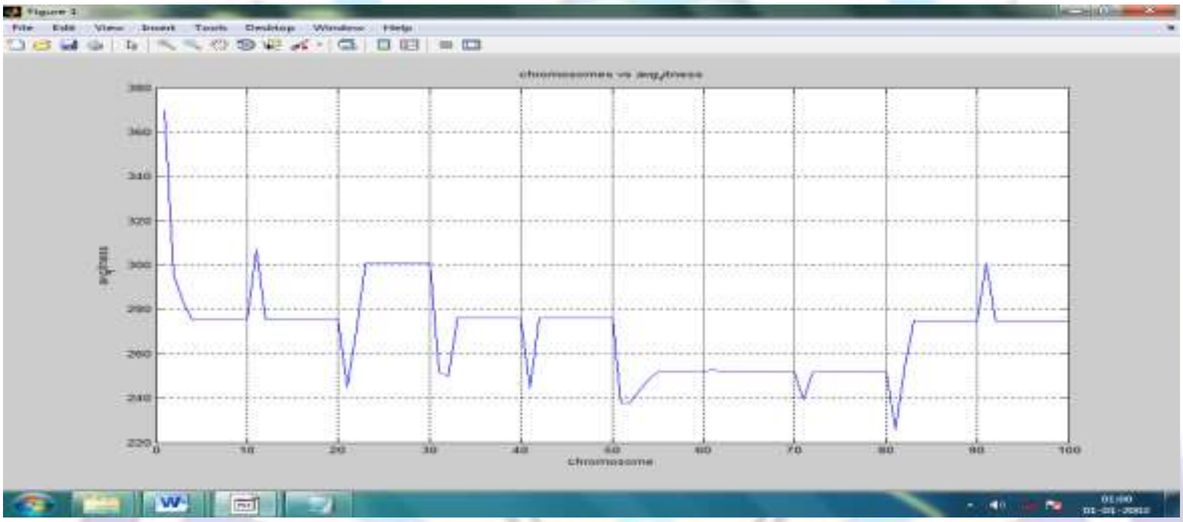
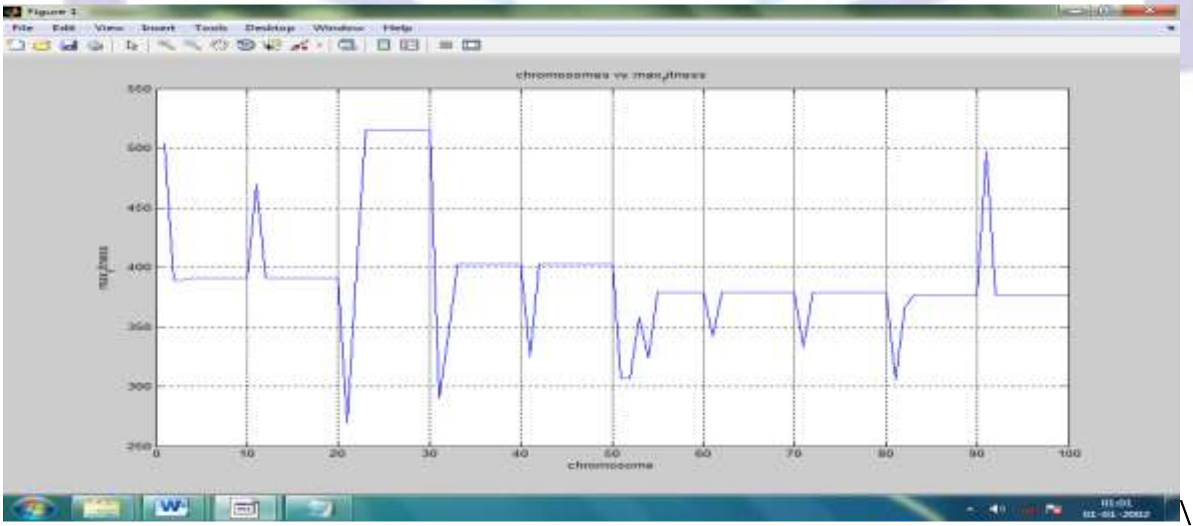



Fig8: plot for chromosome vs. average_ fitness in 100 generation.




\

Fig9: plot for chromosome vs. maximum_fitness in 100 generation.

As seen from the results in Fig.1 when increasing the no.of itaration 1:10 we get desired minimum fitness in first generetion that is 228 , maximum fitness with value 504 and avarage cost is 288. When the no. of iteration are 11 to 20, change the crossover boundary and mutation probability (Pm=0.5) then the minimum fitness is same as previous itaration 1to10.But avarage fitness and max fitness are better then the previous iteration.Fig.2 reflex the each chromosome in each generation in iteration number 11 to 20.Fig.3 shows chromosome vs. fitness in no. of iteration 21 to 30. Here, we see recent desired fitness is 226 better than the previous iteration, But max fitness and avarage fitness are increase than the previous iteration.In case of Fig.1,Fig.2 . Here we see that, when we change the crossover boundary and mutation probability minimum dersired fitness found but avarage cost & max cost are high.when we decrease the mutation probability (Pm=0.375) in iteration.no 31 to 40 we get desired minimum fitness, avarage fitness, max cost that is respectively 222, 271, 403 better than the previous generation.The iteration no. 31 to 40 shows in fig.4. After a certain generation, when we change the crossover boundary and mutation probability (Pm=0.25) in generation 51 to 60, we get desired minimum fitness better then the previous generation. The fig.5 reflex the iteration no. 51 to 60.So, we see that, fitness is not dependent on crossover boundary and mutation probability but there are probability. The fig6. Reflex the iteration no. 81 to 90.when iteration number 81 to 90 , then we change the crossover boundary and mutation probability (Pm=0.75), we get the minimum cost between 1 to 100 generation, that is 191. Minimum fitness,avarage fitness and max fitness are shown respectively in Fig.7, Fig.8, Fig.9 whose iteration number between 1to 100.

## Conclusion:

We have followed survival of fittest of Charles Darwin's theory. Main philosophy of genetic algorithm is followed to Holland. We apply one-point crossover operator for a genetic algorithm for the travelling salesman problem. In this method, when we get the minimum cost in generation then we see, minimum cost is also in next generation or reduce the minimum cost. So, minimum cost is easily found in many generations.

## Reference:


[1]. D.E. Goldberg and R. lingle. "Alleles, Loci and the travelling salesman problem". In J.J. Grefenstette (ed) proceedings of the 1st International Conference on genetic algorithms and their applications. Lawrence Erlbaum, Hilladale, NJ, 1985.

[2]. Davis. "job-shop Scheduling with Genetic algorithms". Proceeding of International Conference on genetic algorithms and their applications,pp. 136-140, 1985.

[3] I.M. Oliver, D. J. Smith and J.R.C. Holland. *"A Study of Permutation Crossover Operators on theTravelling Salesman Problem".* In J.J. Grefenstette (ed.). Genetic Algorithms and TheirApplications: Proceedings of the 2nd International Conference on Genetic Algorithms. LawrenceErlbaum Associates, Hilladale, NJ, 1987.

[4] D.E. Goldberg. *"Genetic Algorithms in Search, Optimization, and Machine Learning".* Addison-Wesley, New York, 1989.

[5] R.G. Bland and D.F. Shallcross. *"Large Travelling Salesman Problems arising form Experimentsin X-ray Crystallography: A Preliminary Report on Computation".* Operations Research Letters 8,pp. 125-128, 1989.

[6] D. Whitley, T. Starkweather and D. Shaner. *"The Traveling Salesman and Sequence Scheduling:Quality Solutions using Genetic Edge Recombination".* In L. Davis (Ed.) Handbook of GeneticAlgorithms. Van Nostrand Reinhold, New York, pp. 350-372, 1991.

[7] K. Deb. *"Optimization For Engineering Design: Algorithms And Examples".* Prentice Hall Of India Pvt. Ltd., New Delhi, India, 1995.

[8] N.J. Radcliffe and P.D. Surry. *"Formae and variance of fitness".* In D. Whitley and M. Vose (Eds.) Foundations of Genetic Algorithms 3. Morgan Kaufmann, San Mateo, CA, pp. 51-72,1995.

[9] P. Poon and J. Carter. *"Genetic algorithm crossover operations for ordering applications"*.Computers and Operations Research 22, pp. 135–47, 1995.

[10] C.H. Papadimitriou and K. Steglitz. *"Combinatorial Optimization: Algorithms and Complexity"*.Prentice Hall of India Private Limited, India, 1997

[11] Z.H. Ahmed and S.N.N Pandit. "The travelling man salesman problem with precedence constraints". Opsearch 38,pp. 299-318, 2001.

[12] [C. Moon, J. Kim, G. Choi and Y. Seo. *"An efficient genetic algorithm for the traveling salesmanproblem with precedence constraints".* European Journal of Operational Research 140, pp. 606-617, 2002.

[13] I. Choi, S. Kim and H. Kim. *"A genetic algorithm with a mixed region search for the asymmetric traveling salesman problem"*. Computers & Operations Research 30, pp. 773 – 786, 2003.