iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 **Issue:** V **Month of publication:** May 2014

DOI:

www.ijraset.com Vol. 2 Issue V, May 2014

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

Enhanced Genetic Algorithm for solving Travelling Salesman Problem

Jyoti Girdhar[#1] , Kapil Chawla[*2]

[#]CSE,DCRUST

***Abstract*— TSP is optimization problem which is used to find minimum path for salesperson. The Actual use of tsp is routing in network. Minimum path helps to reduce the overall receiving time and improves system performance. The work proposed here intends to test the performance of different Crossover used in GA and compare the performance for each of them and compare to others. Since there are other methods traditionally adopted to obtain the optimum distance for TSP. This work aims at establishing the superiority of Genetic Algorithms in optimizing TSP. Since precise minimum path remains a great challenge, the objective of this paper is to develop some new and practical model with computational intelligence algorithms.**

***Keywords*— Genetic algorithm, tavelling salesman problem , crossover, operators , optimization**

I. INTRODUCTION

The traveling salesman problem (TSP) is a typical combinatorial optimization problem which is perhaps the best-studied NP-hard combinatorial problem[1].Given a set of cities and the cost of travel (or distance) between each possible pairs, the TSP, is to find the best possible way of visiting all the cities and returning to the starting point that minimize the travel cost (or travel distance). There are some near-optimal or approximate approaches to solve this problem, such as branch-and-bound[2], cutting planes[3], 2- opt[4], simulated annealing[5], neural networks[6], and tabu search[7]. General search methods such as genetic algorithms (GAs) [8],[9] have also been applied to the TSP.The advantages of the GA over other heuristic methods for solving combinatorial optimization problems include massive parallelism and its ability to solve "non-linear" optimization problems where the search space is extremely large. Another area in which genetic algorithms excel is their ability to manipulate many parameters simultaneously.In this paper, a Genetic algorithm is also proposed with the goal of solving the optimization problems, and has been applied to the TSP with varying degree of success.

This paper is organized as follows. Section 2 discusses the travelling salesman problem, Section 3 describes the genetic algorithm. Section 4 discusses the problem description. Section 5 and 6 includes the genetic algorithm relating our problem domain and the results comparing the three operators

II. TRAVELLING SALESMAN PROBLEM

The traveling salesman problem (TSP) is one of the most important and representative combinatorial optimization problems because it is simple to state but difficult to solve. The TSP can be stated as follows. The salesman must visit a list of cities, all of whose locations are given. The salesman's task is to find the cheapest tour connecting them all, visiting each city only once, and return to the city of origin. Cost here can be distance, time, money, etc. If all the costs between any two cities are equal in both directions, the problem is called symmetric TSP; otherwise, it is called asymmetric.[10].

Definition: Traveling Salesman Problem (TSP), there is n cities and a salesman, and the salesman has to visit those n cities by starting from a certain city, exactly once each city and then return to the city where he starts. For each two cities, the distance between them is prescribed. The objective is to select the order in which he visits the cities to make sure that the total length of the distances in his tour, called a tour length, is as small as possible.

III. GENETIC ALGORITHM

The most popular technique in evolutionary computation research has been the genetic algorithm. In the traditional genetic algorithm, the representation used is a fixed-length bit string. Each position in the string is assumed to represent a

ISSN: 2321-9653

particular feature of an individual, and the value stored in that position represents how that feature is expressed in the solution.

Usually, the string is evaluated as a collection of structural features of a solution that have little or no interactions. The similarity may be drawn directly to genes in biological organisms. Each gene represents an entity that is structurally independent of other genes.

The main reproduction operator used is bit-string crossover, in which two strings are used as parents and new individuals are formed by swapping a sub-sequence between the two strings.

The different types of operators on which genetic algorithm is made are selection operators, reproduction operators (i.e. crossover and mutation operators). These operators are the main foundations over which a genetic algorithm is based.

The flow chart is shown in fig.1

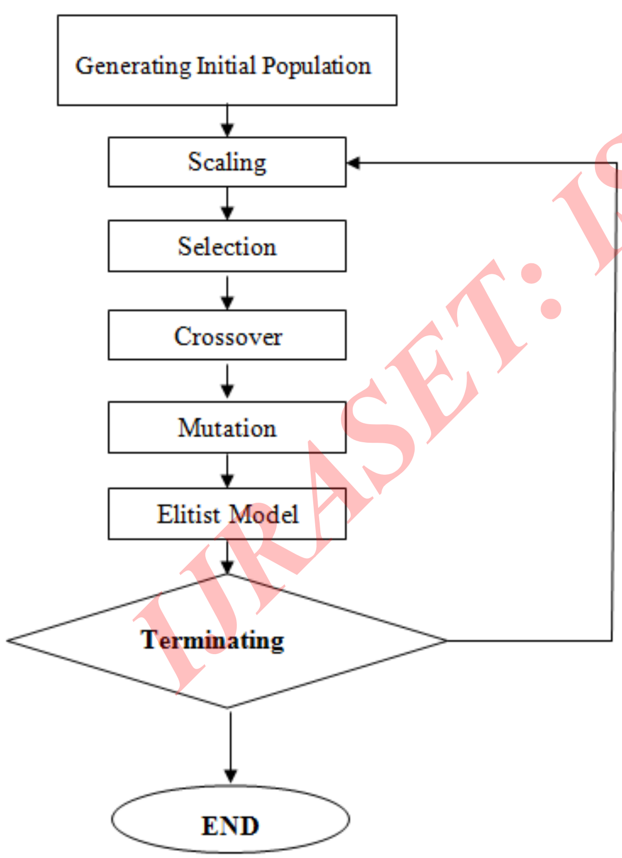


Fig. 1 Flowchart showing the process of a traditional Genetic Algorithm

IV. PROBLEM DISCRIPTION

In this, there are a number of cities. We have been given the distance between different cities. We have to find in which order we must traverse the city so that each city is traversed exactly once and the total distance traveled is minimum.
Here the problem is comparing the performance of different genetic algorithms with different crossover operators. As already discussed previously, genetic algorithm is one of the optimization techniques that can be used to solve the problems of function maximization. It can be said as a search procedure inspired by principles from natural selection and genetics. It is often used as an optimization method to solve problems where very little is known about the objective function. The operation of the genetic algorithm is very simple. It starts with a population of random individuals, each corresponding to a particular candidate solution to the problem to be solved. Then, the best individuals survive, mate, and create offspring, originating a new population of individuals. This process is repeated a number of times, and usually leads to better and better individuals.

V. GA RELATING OUR PROBLEM DOMAIN

The steps of applying GA relating our problem at hand are:

1. Choosing an Encoding scheme.
2. Choosing Fitness function.
3. Choosing Operators.
4. Choosing Parameters.
5. Choosing an Initialization method and Stopping criteria

Step1: Encoding scheme

The first step for solving a problem in GA is to encode the problem at work into a state which can be solved using GA. The basis of genetics in nature is a chromosome. Each individual in the search space, i.e. each solution to the problem worked upon, needs to be encoded so that it can be modeled as a chromosome. The application of a genetic algorithm to a problem starts with the encoding.

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

The encoding specifies a mapping that transforms a possible solution to the problem into a structure containing a collection of decision variables that are important to the problem worked upon.

For our problem value encoding is required.

Real/Value encoding -Every chromosome is a string of values and the values can be anything related to the problem . Direct value encoding can be used in problems where some complicated value such as real numbers are used. Use of binary encoding for this type of problems would be very difficult. In value encoding, every chromosome is a string of some values. Values can be anything connected to problem, form numbers, real numbers or chars to some complicated objects

Step2: Fitness function

The next step is to specify a function that can assign a score to any possible solution. The score is a numerical value that indicates how well a particular solution solves the problem. The score is the fitness of the individual solution. It represents how well the individual suits to the environment. The task of the GA is to discover solutions that have higher fitness values among the set of all possible solutions.

Our aim is to find out individual having minimum total waiting time. So the individual who has minimum total waiting time is fittest as compared to other. So, The fitness function of a Solution S_i is given by

$$\text{Fitness}(S_i) = \frac{\sum_{i,j=1}^{n} D_{ij}}{n}$$

Where D_{ij} is distance between city i and j.

n is the total no. of cities.

4.1.3 Operators

Once the encoding and the fitness function are specified, the implementer has to choose selection and genetic operators to evolve new solutions to the problem being solved. The selection operator simulates the "survival-of-the-fittest". There are various mechanisms to implement this operator, and the idea is to give preference to better individuals.

Selection- After deciding on an encoding, the next step is to decide how to perform selection i.e., how to choose individuals in the population that will create offspring for the next generation and how many offspring each will create. The purpose of selection is to emphasize fitter individuals in the population in hopes that their off springs have higher fitness Selection is a method that randomly picks chromosomes out of the population according to their evaluation function. The higher the fitness function, the more chance an individual has to be selected.

For our case, selection process is through Roulette wheel selection.

The Figure 4.3 shows the basic selection process.

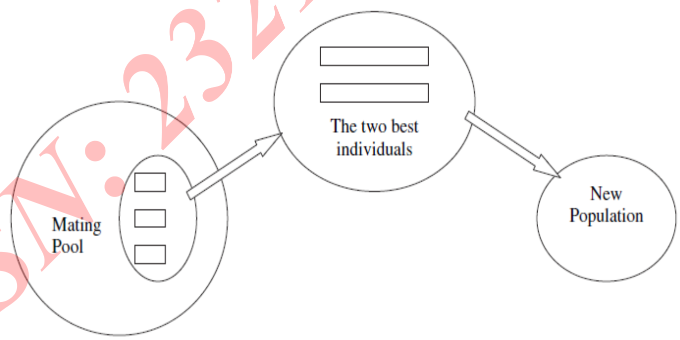


Figure 3 Selection

Roulette wheel selection

Roulette selection is one of the traditional GA selection techniques. The commonly used reproduction operator is the proportionate reproductive operator where a string is selected from the mating pool with a probability proportional to the fitness. The principle of roulette wheel selection is a linear search through a roulette wheel with the slots in the wheel weighted in proportion to the individual's fitness values.

This method is implemented as follows:

1. Sum the total expected value of the individuals in the population. Let it be T.

2. Repeat N times:

- Choose a random integer 'r' between o and T.

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

- Loop through the individuals in the population, summing the expected values, until the sum is greater than or equal to 'r'. The individual whose expected value puts the sum over this limit is the one selected.

Roulette wheel selection is easier to implement but is noisy. The rate of evolution depends on the variance of fitness's in the population.

Step 4: Parameters

With an encoding, a fitness function, and operators in hand, the GA is ready to enter in action. But before doing that, the user has to specify a number of parameters such as population size, no. of processes. For crossover the probability is 100% i.e. with every iteration/algorithm run crossover is performed always. Whereas the mutation operation performed after every five consecutive algorithm runs.

Step5: Initialization method & stopping criteria

The last steps of applying a GA are the specification of an initialization method and stopping criteria. Following the initialization step, each individual is evaluated according to the user's specified fitness function.

A number of criteria can be chosen for this purpose, including among others, a maximum number of generations or time has elapsed, some predefined fitness value is reached, or the population has converged considerably . For our problem, stopping criteria chosen is no. of iterations and initialization method is random generation of population.

VI. RESULTS

After appling the above stated steps in our implementation, The procedure followed while implementing it is first started with randomly generating the population of chromosomes, for this implementation population chosen is 50 only with each chromosome having length of 50 genes. This process in GA has been called as encoding the input population.

Then a selection technique has been employed over the encoded population and after that a three types of crossover is performed, i.e. cyclic crossover. After the crossover operation, mutation is performed, but after every 5 iterations. Then this operation is repeated for the required number of iterations. The results generated for our problem at hand are shown in Figure 4.

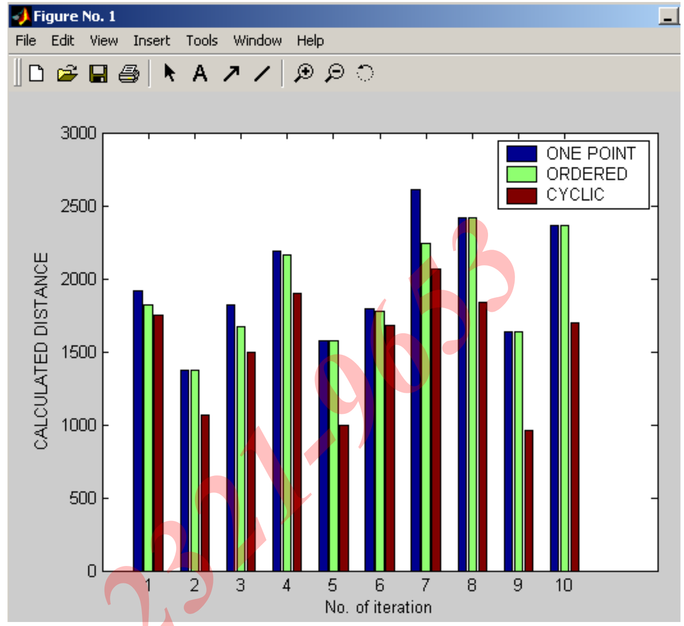


Fig.4. Results of various crossover operators over a no. of iterations

The results that came out are in favour of algorithm with cyclic crossover operator. The parameters which used are as follows:

Population—Chosen Randomly

Encoding scheme employed—Value/Real encoding

Crossover—one point, ordered, cyclic crossover

Mutation—interchanging mutation

VII. CONCLUSION

This dissertation is based on implementing different Genetic algorithm based on different crossover operators and comparison of various crossover operators based on the total distance of these algorithms at different number of iterations.

Basically GA is one of the better function optimization methods generally employed now days. Population is randomly generated. The encoding scheme for this problem of function maximization is value/real encoding. And so a different type of crossover method is applied for crossing the chromosomes in the population for producing better

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

offspring's. And finally interchanging mutation is used while implementing this algorithm.

In this dissertation mainly we solve the problem of CPU scheduling using the genetic algorithm. And cyclic gives provably optimal solution.

REFERENCES

[1] Korte B.H., Jens V, *Combinatorial optimization: theory and algorithms*, Springer, New York, 2008

[2] G. Finke, A. Claus, E. Gunn, "A two-commodity network flow approach to the traveling salesman problem", Congress Numerantium

41, 1984, pp. 167–178.

[3] P. Miliotis, "Using cutting planes to solve the symmetric traveling salesman problem", Mathematical programming 15, 1978, pp. 177–

188.

[4] S. Lin, B.W. Kernighan, "An effective heuristic algorithm for traveling salesman problem", Operations Research, 1973, pp. 498– 516.

[5] S. Kirkpatrick, C.D. Gelatt Jr., M.P, Vecchi, "Configuration space analysis of traveling salesman problem", Journal Physique 46 , 1985,

pp. 1277–1292.

[6] B. Shirrish, J. Nigel, M.R. Kabuka, "A Boolean neural network approach for the traveling salesman problem", IEEE Transactions on

Computers 42 ,1993, pp. 1271–1278.

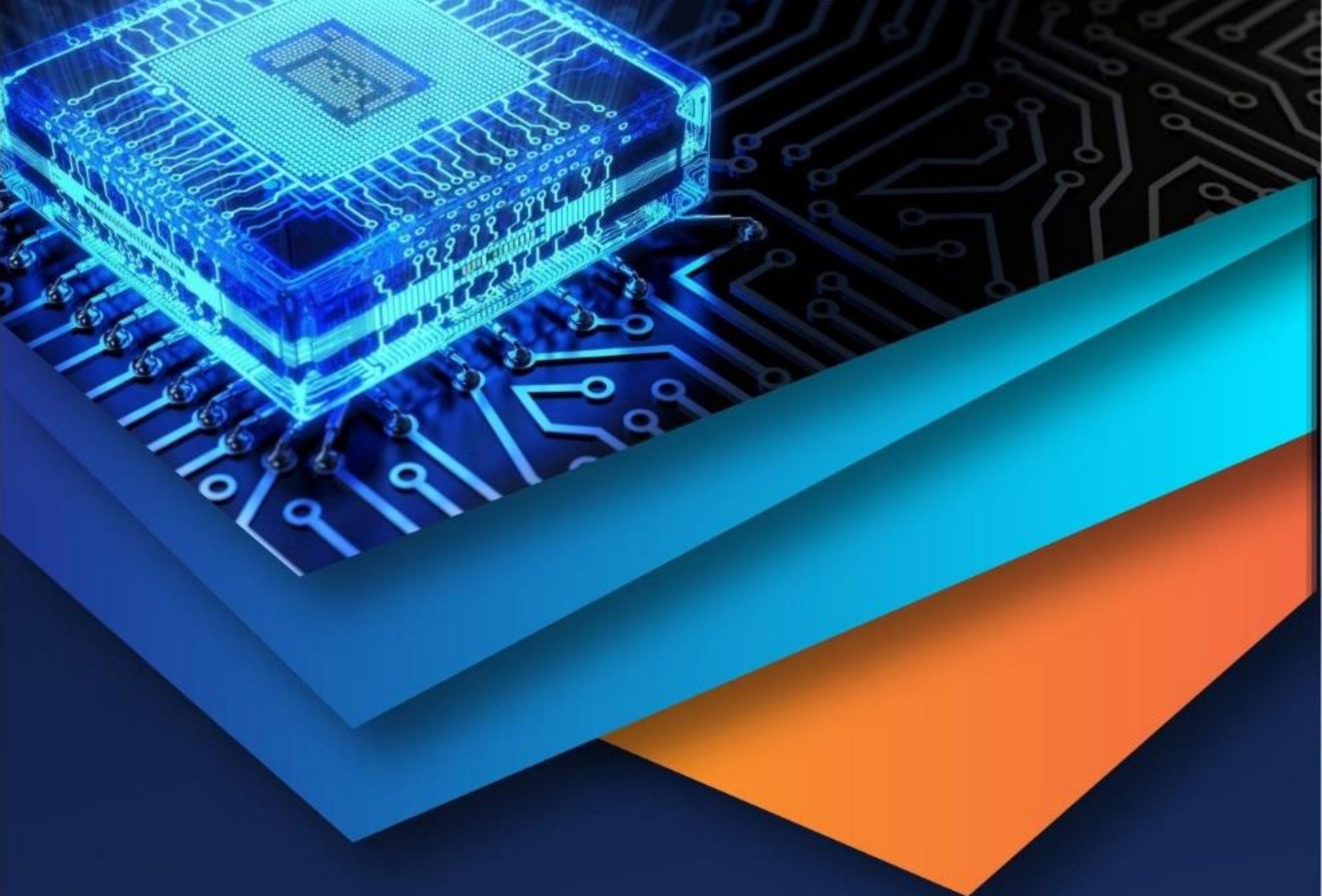
[7] F. Glover, Artificial intelligence. "heuristic frameworks and tabu search", Managerial & Decision Economics 11 , 1990, pp. 365–378.

[8] D.E.Goldberg, Genetic Algorithms in Search Optimization,and Machine learning, MA : Addison Wesley, 1989

[9] J.H. Holland, , *Adaptation in Natural and Artificial SystemsAnn Arbor*. Univ. of Michigan Press, 1975.

[10] Hung Dinh Nguyen, Ikuo Yoshihara, Kunihito Yamamori and Moritoshi Yasunaga "Implementation of an Effective Hybrid GA for Large-Scale Traveling Salesman Problems" IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS, VOL. 37, NO. 1, FEBRUARY 2007

.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY