International Journal of Computer Science
Oddity...Probe...Reviste...

http://www.ijcsjournal.com Volume 1, Issue 2, No 3, 2013. ISSN: 2348-6600
Reference ID: IJCS-020 PAGE NO: 105-110

SOLVING TRAVELLING SALESMAN PROBLEM EFFICIENTLY USING GENETIC ALGORITHM

Dr.S.S.Dhenakaran #1, S.Sangari Devi *2

#1 Assistant professor, Department of Computer science and Engineering,
Alagappa University, Karaikudi.
India.

*2 Department of Computer science and Engineering,
Alagappa University, Karaikudi.
India.

Abstract— The Genetic Algorithm is a popular optimization technique which is bio-inspired and is based on the concepts of natural genetics and natural selection theories.The main idea is to propose a new representation method of chromosomes using binary matrix and new fittest criteria to be used as method for finding the optimal solution for TSP. The idea is taken from genetic algorithm of artificial inelegance as a basic ingredient which has been used as search algorithm to find the near-optimal solutions is proposed. Now we are introducing the new fittest criteria for crossing over, and applying the algorithm on symmetric as well as asymmetric TSP, also presenting asymmetric problem in a new and different way.

KeyWords: Genetic Algorithm, Chromosomes, Fittest criteria.

I. INTRODUCTION

Genetic Algorithms are search algorithms based on natural selection and natural genetics. They combine survival of fittest among structures with structured yet randomized information exchange to form a search algorithm. Genetic Algorithm has been developed by John Holland and his co-workers in the University of Michigan in the early 60‟s. Genetic algorithms are theoretically and empirically proved to provide robust search in complex spaces. Its validity in –*Function Optimization* and *Control Applications* is well established. Genetic Algorithms (GA) provide a general approach for searching for global minima or maxima within a bounded, quantized search space. Since GA only requires a way to evaluate the performance of its solution guesses without any apriori information, they can be applied generally to nearly any optimization problem. GA does not guarantee convergence nor that the optimal solution will be found, but do provide, on average, a "good" solution. GA is usually extensively modified to suit a particular application. As a result, it is hard to classify a "generic" or "traditional" GA, since there are so many variants. However, by studying the original ideas involved with the early GA and studying other variants, one can isolate the main operations and compose a "traditional" GA. An improvement to the "traditional" GA to provide faster and more efficient searches for GAS that does not rely on average chromosome convergence (i.e. applications which are only interested in the best solution).therefore will be referred to as the fixed-rate GA.

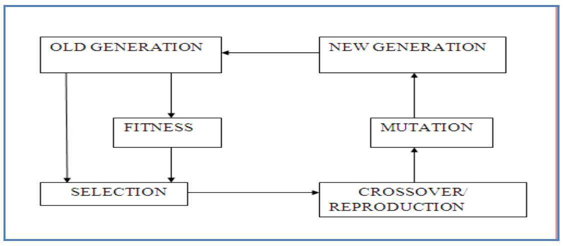


Figure 1 Genetic Loop

The Travelling Salesman Problem (TSP)

The Travelling Salesman Problem (TSP) is a classic combinatorial optimization problem, which is simple to state but very difficult t o solve. This problem is known to be NP-hard, and cannot be solved exactly in polynomial time. Many exact and heuristic algorithms have been developed in the field of operations research (OR) to solve this problem. The problem is to find the shortest possible tour through a set of n vertices so that each vertex is visited exactly once.

The proposed genetic algorithm in this paper build on much work done by previous researchers [4], but we introduces additional improvements, providing an algorithm for symmetric as well as Asymmetric TSP, here we are implementing the new fittest criteria as well as new representation of asymmetric matrix and improving our solution by applying the crossover and mutation again and again in order to get the optimal solution.

II. Genetic Algorithm(GA)

Genetic algorithm (GA) as a computational intelligence method is a search technique used in computer science to find approximate solutions to combinatorial optimization problems. The genetic algorithms are more appropriately said to be an optimization technique based on natural evolution. They include the survival of the fittest idea algorithm. The idea is to first ‘guess’ the solutions and then combining the fittest solution to create a new generation of solutions which should be better than the previous generation. We also include a random mutation element to account for the occasional mishap.

The genetic algorithm process consists of the following:

The population is defined to be the collection of all the chromosomes. A generation is the population after a specific number of iterations of the genetic loop. A chromosome is composed of genes, each of which reflects a parameter to be optimized. Therefore, each individual chromosome represents a possible solution to the optimization problem. The dimension of the GA refers to the dimension of the search space which equals the number of genes in each chromosome.

FITNESS

The fitness function provides a way for the GA to analyze the performance of each chromosome in the population. Since the fitness function is the only relation between the GA and the application itself, the function must be chosen with care. The fitness function must reflect the application appropriately with respect to the way the parameters are to be minimized.

SELECTION

The selection operator selects chromosomes from the current generation to be parents for the next generation. The probability of each chromosomes selection is given by:

$$\mathbf{Ps(i)=f(i)/\sum^{N} J=1f(J)}$$

where *ps* ***(i)*** and *f (i)* are the probability of selection and fitness value for the ith chromosome respectively. Parents are selected in pairs. Once one chromosome is selected, the probabilities are renormalized without the selected chromosome, so

International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 1, Issue 2, No 3, 2013. ISSN: 2348-6600
Reference ID: IJCS-020 PAGE NO: 105-110

that the parent is selected from the remaining chromosomes. Thus each pair is composed of two different chromosomes. It is possible for a chromosome to be in more than one pair.

CROSSOVER

Crossover is the GA's primary local search routine. The crossover/reproduction operator computes two offspring for each parent pair given from the selection operator. These offspring, after mutation, make up the new generation. A probability of crossover is predetermined before the algorithm is started which governs whether each parent pair is crossed-over or reproduced. Reproduction results in the offspring pair being exactly equal to the parent pair. The crossover operation converts the parent pair to binary notation and swaps bits after a randomly selected crossover point to form the offspring pair.

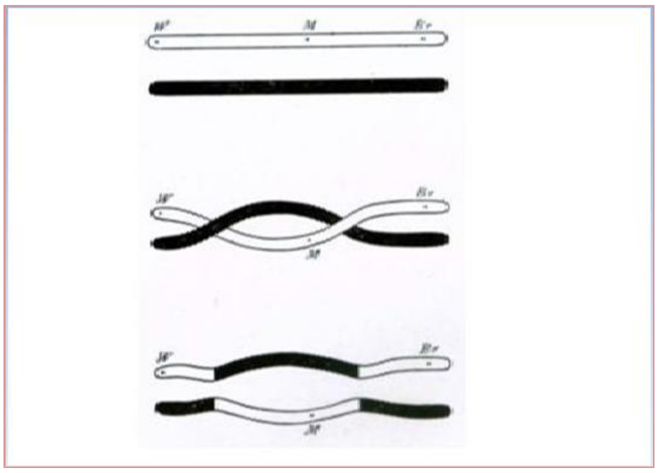


Figure.2 Crossover of Two Strands of Chromosome

MUTATION

Mutations are global searches. A probability of mutation is again predetermined before the algorithm is started which is applied to each individual bit of each offspring chromosome to determine if it is to be inverted

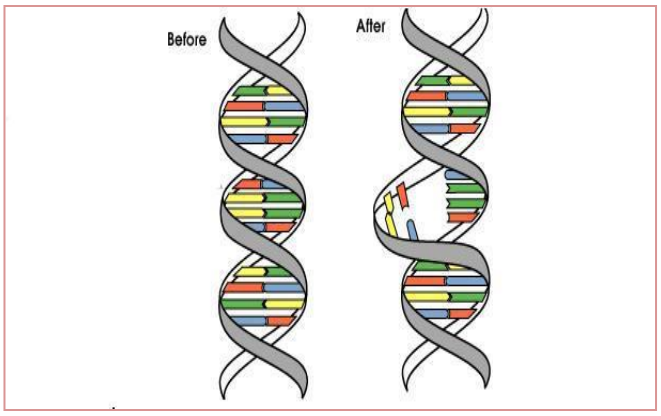


Figure.3 Mutation of a Chromosome

III. Matrix Representation

According to the algorithm we are presenting the tour as binary matrix. In figure (A,B,C,D,E) every gene is presented as binary bits, if the element (i ,j) in the matrix is a set to 1 its mean there is an edge (directed path) between the city I to city j in the tour.

A	1	0	1	0
	B	1	0	0
		C	0	1
			D	1
				E

Figure.3

According to the algorithm we are presenting the tour as a binary matrix. In fig 1 gene is represented as a binary bit, if the element (i,j) is set to (1) its mean that there is an edge (directed path) between the city i

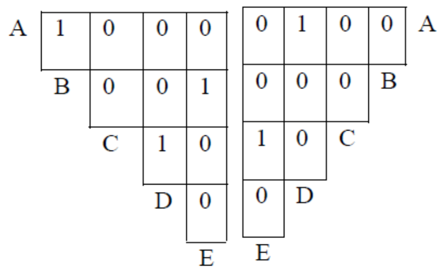


Figure 4

and city j in the tour. The above representation is for symmetric matrix TSP that the dij = dji . For asymmetric matrix that is when dij ≠ dji the above representation will be considered as,

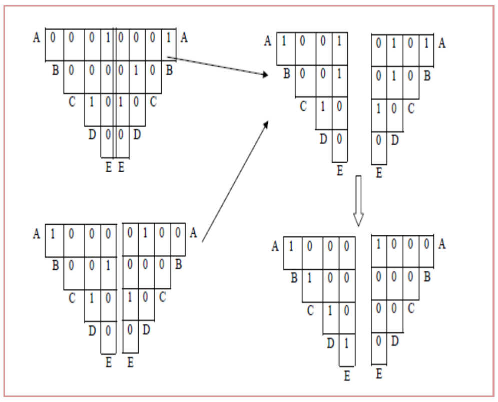


Figure 5 Cross and Mutation

The left upper triangle (LUTM) represents the movement from left to right that is the forward movement.

In (LUTM) we have the path from

A→B, B→C, C→E

Whereas the upper right triangular matrix (URTM) represents the movement from bottom to top, in (URTM) there is the path from E→D, D→A. In this way the complete tour will be,

A→B→ C→E → D→A

The matrix representation must satisfy the two conditions to satisfy a legal tour:

For symmetric case:

(i) The number of matrix element that has the value (1) must equal to the number of vertices in the tour.

(ii) The number of matrix elements that have the value of (1) in each row and each column of the same vertex
must be equal to two;

For asymmetric case:

(i) The total number of the element that has the value (1) in both (LUTM) and (RUTM) must be equal to the number of vertices in the tour.

(ii) For the same vertex the sum of both matrix elements that has the value (1) must be equal to 2.

Cross-over Operation

Here we are using the cross-over operator by applying the **OR** operation on the two parent matrices to get a single matrix.

Mutation Operation

If the resultant tour (matrix) is an illegal tour (i.e does not satisfy the two condition mentioned above),
then it must be repaired. This is done by counting the number of element with (1) value in each row and column for the city, if the number is greater than 2 then repeat deleting the longest edge from the resultant tour until the number of element in the resultant tour is equal to 2. However, if the number of elements in the resultant tour is less then 2 than

International Journal of Computer Science
Oddity...Probe...Reviste...
http://www.ijcsjournal.com Volume 1, Issue 2, No 3, 2013. ISSN: 2348-6600
Reference ID: IJCS-020 PAGE NO: 105-110

add the missing edges in the tour by greedy algorithm. Considering two tours

T1: A→E→ C→D → B→A=17
T2: A→B→ E→C → D→A=22

Then cross-over and mutation of these two tours will be,

Value of the assignment:

Consider the weighted matrix of the given problem and solve it by using assignment algorithm and called the optimal total cost as a value of the assignment problem.

IV. ALGORITHM

Step-1: Randomly create the initial population of individual string of the given TSP problem and
Create a matrix representation of the cost of the path between two cities.

Step-2: Assign the fitness to each chromosome in the population using fitness criteria measure.
F(x) = 1/x
Where, x represents the total cost of the string. Selection criteria depend upon the value of string if it is close to some threshold value.

Step-3: Create new off-spring population from two existing chromosomes in the parent population by applying crossover operator.

Step-4: Mutate the resultant off-springs if required.
NOTE: After the crossover off spring population has the fitness value higher than the parents.

Step-5: Repeat step 3 and 4 until we get an optimal solution to the problem.

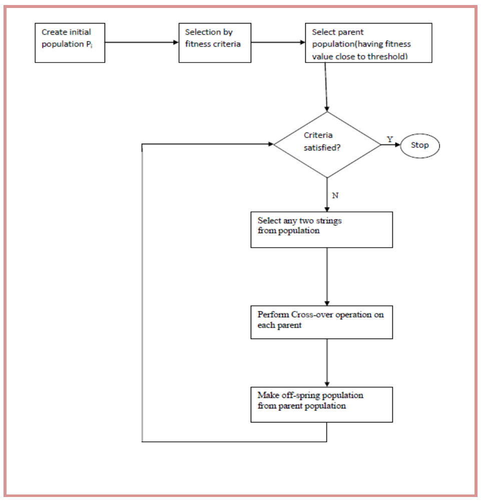

Create initial population P$_i$
Selection by fitness criteria
Select parent population(having fitness value close to threshold)
Criteria satisfied?
Y
Stop
N
Select any two strings from population
Perform Cross-over operation on each parent
Make off-spring population from parent population

Figure:6 Flow chart

EXAMPLE:

Consider the weighted matrix,

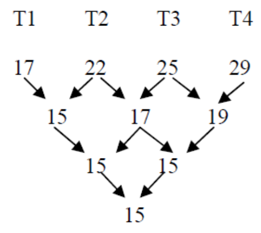
	A	B	C	D	E
A	∞	2	5	7	1
B	6	∞	3	8	2
C	8	7	∞	4	7
D	12	4	6	∞	5
E	1	3	2	8	∞

The value of the assignment of the above problem is 13.

Initial population:

A→E→ C→D → B→A=17
A→B→ E→C → D→A=22
A→C→ D→E → B→A=23
A→B→ D→C → E→A=24
A→E→ B→C → D→A=25
A→B→ C→E → D→A=26

A→E→ D→C → B→A=28
A→D→ C→E → B→A=29
A→E→ C→B → D→A=30
According to the fitness criteria we are selecting the tour having values,
Initial population: 17 22 25 29
After 1st cross-over and mutation 15 17 19
After 2ND cross-over and mutation 15 15
After 3rd cross-over and mutation 15.



The resultant tour will be,
A→B→ C→D→ E→A=15

V. CONCLUSION

While this exposition has covered the basic principles of GAs, the number of variations that have been suggested is enormous. Probably everybody's GA is unique. Many variations in population size, in initialization methods, in fitness definition, in selection and replacement strategies, in crossover and mutation are obviously possible.In this paper we have given a very effective procedure for TSP by using the genetic algorithm of artificial intelligence. Also providing the fittest criteria, the proposed algorithm is for symmetric as well as asymmetric TSP with a different representation for asymmetric TSP which is very useful in solving the problem easily.

REFERENCES

[1] Bineet Mishra, Rakesh Kumar Patnaik "Genetic Algorithm and Its Variants: Theory And Applications", Department of Electronics and Communication Engineering. NIT ROURKELA.

[2] Phillip David Power, "Non Linear Multi Layer Perceptron Channel Equalisation",Chapter 4 „Genetic Algorithm Optimisation", *in IEEE Transactions*,The Queen University of Belfast,April.

[3] *Tzung-Pei Hong,"*Evolution of Appropriate Crossover andMutation Operators in a Genetic Process" Department of Information Management I-Shou University,Kaohsiung, 84008, Taiwan, R.O.C.

[4] *Tzung-Pei Hong,"*Evolution of Appropriate Crossover andMutation Operators in a Genetic Process" Department of Information Management I-Shou University,Kaohsiung, 84008, Taiwan, R.O.C.

[5] T. B&a&ck, "Optimal mutation rates in genetic search," in *Proceedings of the Fifth International Conference on Genetic Algorithms*, pp. 2-8, 1993.

[6] B. Freisleben and P. Merz, "A Genetic local search Algorithm for Solving Symmetric and Asymmetric Travelling Salesman Problems," International Conference On Evolutionary Computation, pp. 616-621, 1996.

[7] Seniw, D., A Genetic algorithm for the Travelling Salesman Problem, MSc Thesis, University of North Carolina, at Charlotte. http://www.heatonresearch.com/articales/65/page1.html. 1996.

[8] B. Freisleben and P. Merz, "A Genetic local search Algorithm for Solving Symmetric and Asymmetric Travelling Salesman Problems," International Conference On Evolutionary Computation, pp. 616-621,1996