

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/236009452>

SOLVING TSP PROBLEM BY USING GENETIC ALGORITHM

Article in International Journal of Basic & Applied Sciences · August 2010

CITATIONS

22

READS

5,727

4 authors:



Fozia Hanif Khan

Sir Syed University of Engineering and Technology

28 PUBLICATIONS 99 CITATIONS

SEE PROFILE



Nasiruddin Khan

University of Karachi

16 PUBLICATIONS 52 CITATIONS

SEE PROFILE



Dr. Syed Inayatullah

University of Karachi

21 PUBLICATIONS 59 CITATIONS

SEE PROFILE



Shaikh Nizami

2 PUBLICATIONS 27 CITATIONS

SEE PROFILE

SOLVING TSP PROBLEM BY USING GENETIC ALGORITHM

FOZIA HANIF KHAN¹, NASIRUDDIN KHAN², SYED INAYATULLAH³,

And SHAIKH TAJUDDIN NIZAMI⁴

ABSTRACT:

The main purpose of this study 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 concept of the proposed method 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. Here 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.

KEY WORDS:

Genetic algorithm, fittest criteria, asymmetric travelling salesman problem.

INTRODUCTION:

As far as the artificial inelegance is concerned, the genetic algorithm is an optimization technique based on natural evolution that is the change over a long period of time. Genetic algorithm (GAs) has been used as a search technique of many NP problems. Genetic algorithms have been successfully applied to many different types of problems, though several factors limit the success of a GA on a specific function. Problem required are good, but optimal solutions are not ideal for GAs. The manner in which points on the search space are represented is an important consideration. An acceptable performance measure or fitness value must be available.

It must also be feasible to test many potential solutions. In nature the fittest individual is most likely to survive and mutate, therefore the next generation should be fitter and healthier because they were bred from healthy parents. The same idea can be applied on the TSP problem by first finding the different solutions and then combine those, which are the fittest solutions among them, in order to create a new and healthy solution and should be optimal or near optimal according to the problem.

¹ Fozia Hanif Khan, Fsazaia Degree college, Faisal, Phone:03333301050, email: mf_khans@hotmail.com.

² Prof. Dr Nasiruddin Khan, University of Karachi, Ph:o3333101945, email: drprof_khan@yahoo.com

³ Syad Inayatullah University of Karachi, Ph: 032221159022, email:inayt_ku@yahoo.com

⁴ Shaikh Tajuddin Nizami, NED University, Ph: 03002761859 email:

TSP is a well known problem for finding the optimal path which can be solved by various methods. Many algorithms have been developed for TSP but here we are using the concept of genetic algorithm. Other approximation techniques for finding near optimum solutions for TSP based on heuristics are proposed in the literature such as [1] simulated annealing [2], ant colonies [3], genetic algorithms (GA) [4] and [5]. John Holland's pioneering book "Adaptation in natural Artificial System (1975, 1992) showed how the evolutionary process can be applied to solve a wide variety of problems using a highly parallel technique that is now called genetic algorithm. Genetic Algorithms have been applied to a large number of real world problems. One of the first such applications was a gas pipeline control system created by [6], [7] mentions several GA applications including message routing, scheduling, and automated design. Entire conferences have been devoted to applications of Genetic Algorithms and evolutionary techniques to specific disciplines, such as Image Analysis, Signal Processing and Telecommunications [8].

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.

GENETIC ALGORITHMS:

The Genetic Algorithm involves the following basic steps.

- Evaluation
- Cross over
- Mutation

Here we are trying to describe the main function used by algorithm to find the shortest closed path through a group of two dimensional positions.

The Genetic Algorithm transforms a population of individual objects, each with an associated fitness value, into a new generation of the population using the Darwin principle of individual of reproduction and survival of the fittest and naturally occurring genetic operation such as a cross over (recombination) and mutation. Each individual in the population represents a possible solution to a given problem. The genetic algorithm attempts to find a very good or best solution to the problem by genetically breeding the population of individuals.

EVOLUTION:

The evolution function plays a very important role in genetic algorithm. Here we use a fittest function to decide how good a chromosome is. It also helps us in the selection of better chromosomes for the next cross over step, if we have relatively large number of initial population.

CROSS OVER:

After the completion of the selection process, the chromosomes chosen to be parents for the next generation are recombined to form children that are new chromosomes. This combination can take many forms and the crossover operator can greatly affects the result of the search.

MUTATION:

The operation of mutation allow new individual to be created. It begins by selecting an individual from the population based on its fitness. A point along the string is selected at random and the character at that point is randomly changed, the alternate individual is then copied in to the next generation of the population.

Mutation is performed after crossover by randomly choosing a chromosome in the new generation to mutate. We randomly choose a point to mutate and switch that point. Many types of mutation operators exist. Here we are using the bit flip method which is only used with a binary chromosome representation, that changes a particular point on the chromosome to its opposite.

Basic Definitions:**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.

Figure 1.

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

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

and city j in the tour. The above representation is for symmetric matrix TSP that the $d_{ij} = d_{ji}$. For asymmetric matrix that is when $d_{ij} \neq d_{ji}$ the above representation will be considered as,

Figure 2.

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

The left upper triangle (LUTM) represents the movement from left to right that is the forward movement. In (LUTM) we have the path from $A \rightarrow B$, $B \rightarrow C$, $C \rightarrow E$. where as the upper right triangular matrix (URTM) represents the movement from bottom to top, in (URTM) there is the path from $E \rightarrow D$, $D \rightarrow A$. In this way the complete tour will be,

$A \rightarrow B \rightarrow C \rightarrow E \rightarrow D \rightarrow A$

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

For symmetric case:

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

- 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.
- 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:

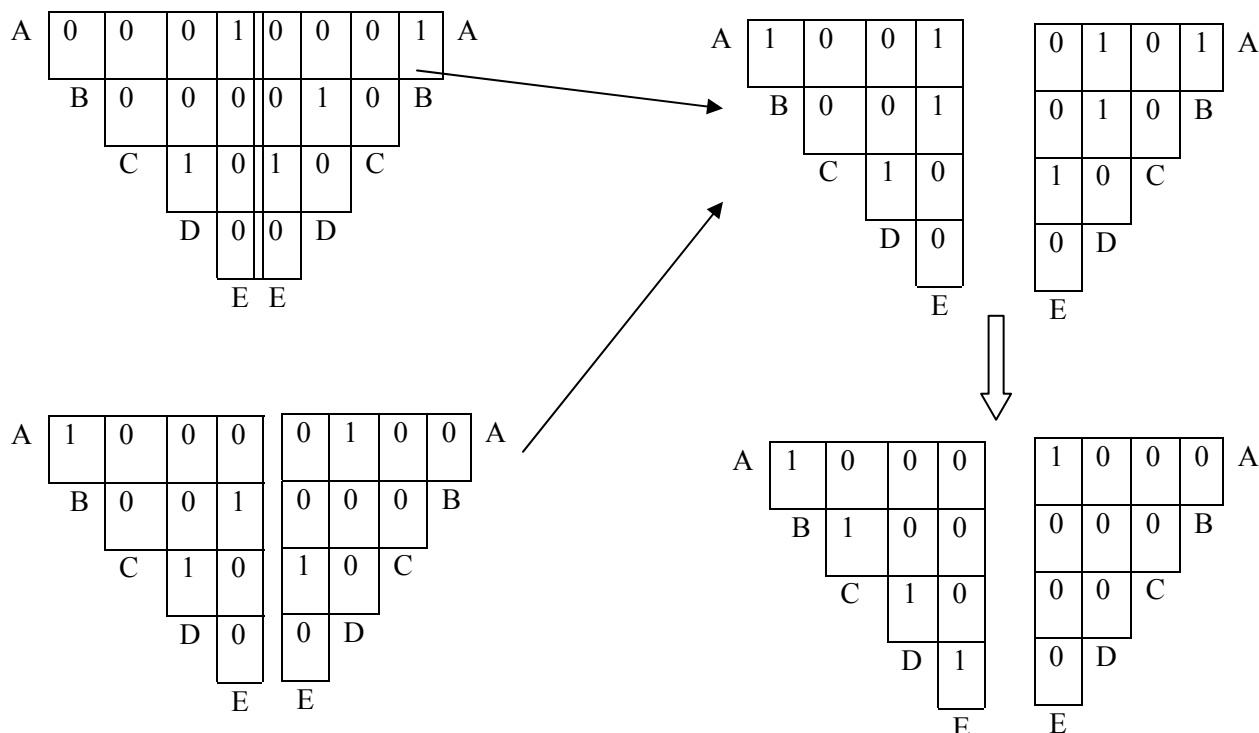
If the resultant tour (matrix) is an illegal tour (i.e does not satisfy the two condition mentioned above), then it must be repaired. This is done by counting the number of element with (1) value in each row and column for the city, if the number is greater than 2 then repeat deleting the longest edge 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 than 2 then add the missing edges in the tour by greedy algorithm. Considering two tours

$T_1: A \rightarrow E \rightarrow C \rightarrow D \rightarrow B \rightarrow A = 17$

$T_2: A \rightarrow B \rightarrow E \rightarrow C \rightarrow D \rightarrow A = 22$

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.

STEPS OF ALGORITHMS:

- Randomly create the initial population of individual strings of the given TSP problem and create a matrix representation of each, must satisfy the two basic conditions as mentioned earlier.
- Assign a fitness to each individual in the population using fitness criteria measure,

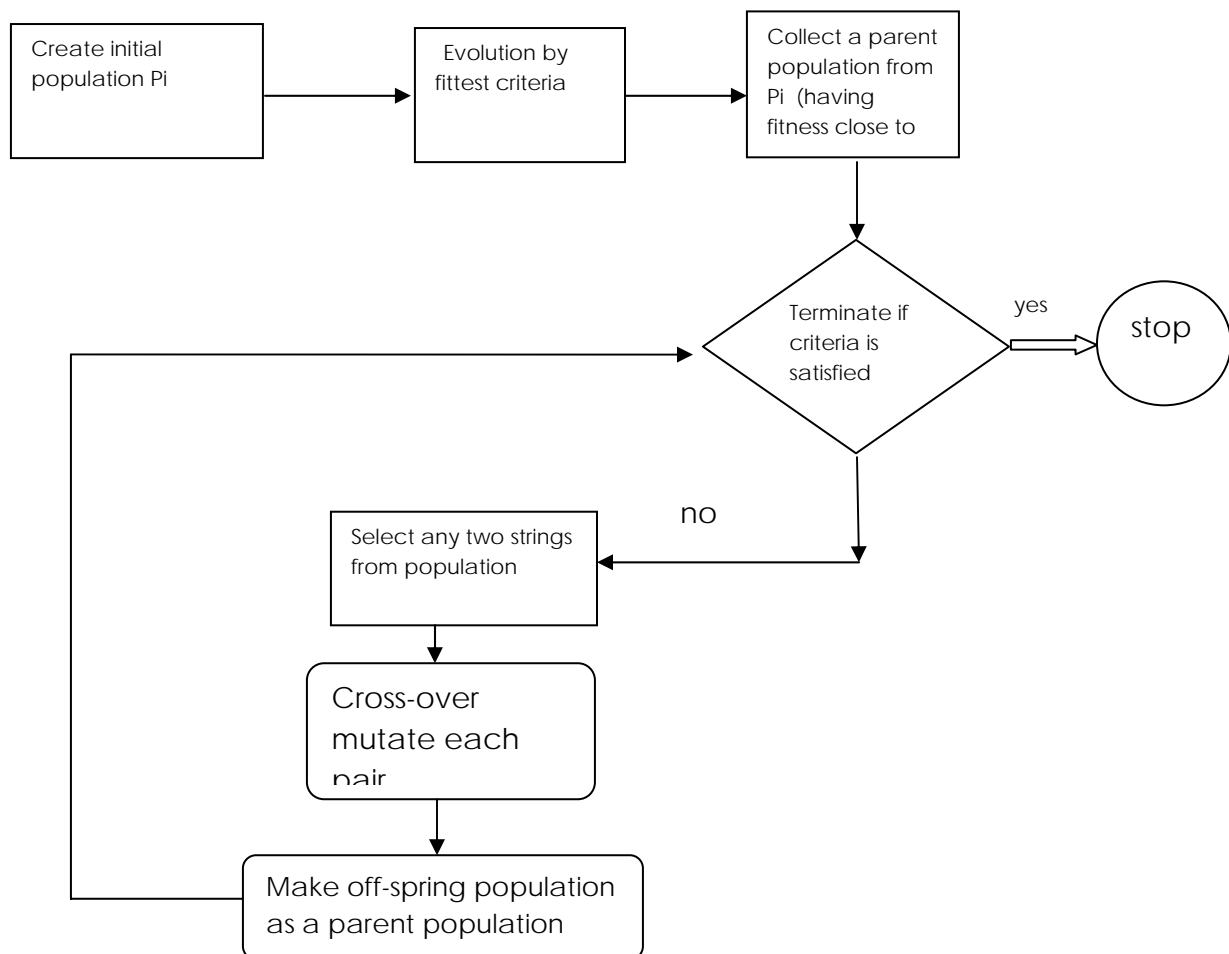
$$F(t) = \frac{\text{value of the assignments of the given problem}}{\text{value of the string}}$$

The selection criterion depends upon the value of the strings if it is close to 1.

- iii. Create new off-spring population of strings from the two existing strings in the parent population by applying crossover operation.
- iv. Mutate the resultant off-springs if required.

Note: After the cross-over and mutation off-spring population has the fitness higher than the parents.

- v. Call the new off-springs as parent population and continue the steps (iii) and (iv) until we get a single offspring that will be an optimal or near optimal solution to the problem.



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,

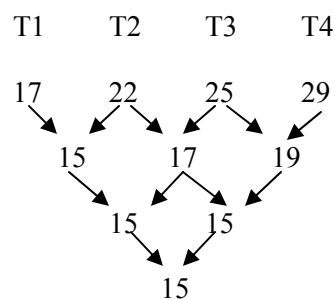
17,22,23,24

Initial population:

After 1st cross-over and mutation

After 2ND cross-over and mutation

After 3rd cross-over and mutation



The resultant tour will be,

$A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow A = 15$

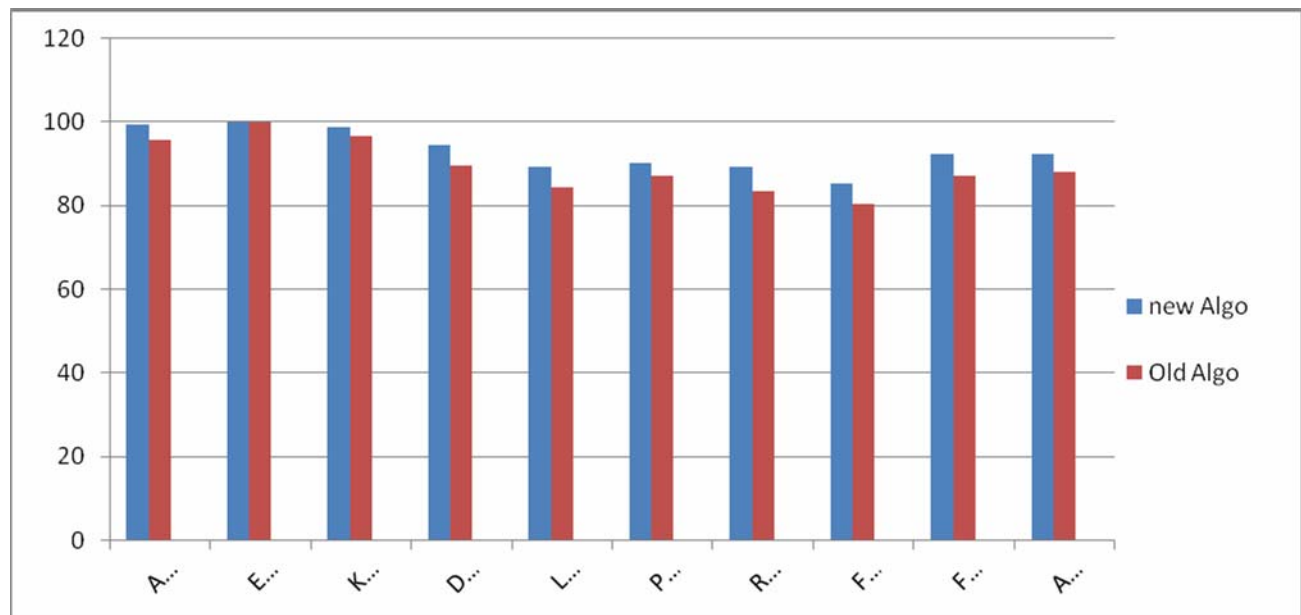
CONCLUSION:

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

EXPERIMENTAL RESULT:

s.no	Problem	Optimal solution	Sols by old algo %	sol by proposed algo%
1.	Att48	10628	93.61.	99.3
2.	Eli51	426	99.97	100
3.	KorA100	21282	96.73	98.7
4.	D198	15780	89.47	94.4
5.	Lin318	42029	84.51	89.32
6.	Pcb442	50778	87.5	90.1
7.	Rat873	8806	83.55	89.24
8.	Fll400	20127	80.31	85.21
9.	Fll577	22204, 22249	87.05	92.5
10.	A280	2579	88.18	92.5

Table 1. comparison between proposed algorithm and old algorithm [4],



Furthermore, in the algorithm given by [4] is for symmetric TSP and the off spring is done by OR operation, in the proposed algorithm here we are introducing the fittest criteria and applying it on the symmetric as well as asymmetric TSP.

REFERENCES:

- [1] B. Freisieben and P. Merz, New Genetic Local Search Operator for Travelling salesman Problem, Conference on Parallel Problem Solving From nature, app. 890-899, 1996.
- [2] P. van Laarhoven and E. H. L. Aarts, Simulated Annealing: Theory and Applications, Kluwer Academic, 1987,
- [3] M. Dorigo and L. M. Gambardella, Ant Colony System: A Cooperative Learning Approach to the Travelling Salesman Problem, JEEE Transaction on Evolutionary Computation, Vol. 1, pp. 53-66, 1997.
- [4] D.E. Goldberg, Genetic Algorithm in Search, Optimization and Machine Learning. Addison-Wesley, 1989.
- [5] Naef Taher Al Rahedi and Jalal Atoum, Solving TSP problem using New Operator in Genetic Algorithms, American Journal of Applied Sciences 6(8):1586-1590, 2009.
- [6] Goldberg D., Computer-Aided Pipeline Operations Using Genetic Algorithms and Rule Learning. Part I; Genetic Algorithms in pipeline Optimization, Engineering with Computer 3, 1987.
- [7] Greffenstette, J., Genetic Algorithms Made Easy, 1991.
- [8] Poli, R. ,el. al. (Eds), Evolutionary Image Analysis, Signal Processing and Telecommunications Springer, 1999.

- [9] 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.
- [10] 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.
- [11] K. E. Nygard and C. H. Yang, "Genetic Algorithm for the Travelling Salesman Problem With Time Window", in Computer Science and Operations Research: New Development in their Interface, O. Balci, R. Sharda and S. A. Zenios Eds, Pergamon Press, pp. 411-423.
- [12] H. Braun, "On Solving Travelling Salesman Problem by Genetic Algorithm", in Parallel Problem-Solving from Nature, Lecture Notes in Computer Science 496, H.P. Schwefel and R. Manner Eds, Springer-Verlag, app. 129-133.