Proceedings of the Second International Conference on Inventive Research in Computing Applications (ICIRCA-2020)
IEEE Xplore Part Number: CFP20N67-ART; ISBN: 978-1-7281-5374-2

Applying Improved Genetic Algorithm to Solve Travelling Salesman Problem

Mayank Agrawal
Department of Computer Engineering and Applications
GLA University
Mathura, India
mayank.agrawal@gla.ac.in

Vinod Jain
Department of Computer Engineering and Applications
GLA University
Mathura, India
vinod.jain@gla.ac.in

Travelling Salesman Problem is a very popular problem in graph theory and it is applicable to solve many problems of science and engineering. It has its applications in electronics, transportation, navigation, computer networks etc. As the number of cities increases the complexity of the algorithm to find the optimal solution increases. TSP is a maximization optimization problem. In this paper genetic algorithm is used to solve TSP problem. Genetic Algorithm is a nature inspired algorithm which is very suitable to solve optimization problems. An improved genetic algorithm is proposed and applied on standards TSP instances. The achieved results are found better as compared to other recent research works in this area.

Keywords—Genetic Algorithm; NP-Hard Problem; Travelling Salesman Problem

I. INTRODUCTION

In Travelling Salesman Problem a salesman visit given cities. It starts its journey from one city and visit each and every city exactly once and came back to the starting city. The solution must visit each and every city exactly once and should follow the path with minimum path length. The problem can be represented using graph theory where we have to find Hamiltonian Circuit for a graph with given number of nodes. The graph is taken as a fully connected graph in which there is an edge from one node to every other node.

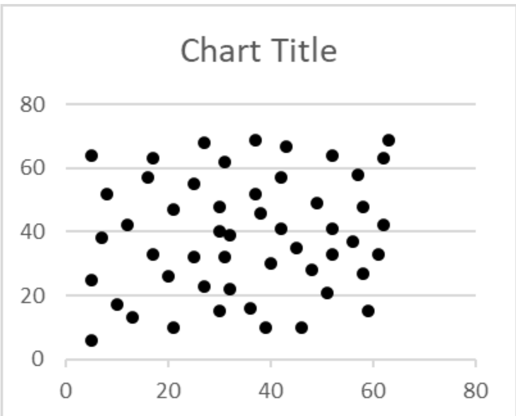

Fig. 1. EIL51 Travelling Salesman Problem

Figure 1 shows a standard TSP problem EIL51. It shows the location of cities in the graph. The solution of the TSP problem is represented by a path that visits each and every city exactly once. The nature inspired algorithm are found better for solving TSP instance which are having large number of cities. Figure 2 shows the steps of standard genetic algorithm. It starts from generation of initial population and then fitness calculation and then selection, cross over and mutation are performed. The review of recent research works in this area are shown in next section.

II. LITERATURE SURVEY

In [1] Chao Li, Xiaogeng Chu, Yingwu Chen and Lining Xing proposed a new Genetic Algorithm for the Travelling Salesman Problem. They proposed new algorithm for improving the performance of Genetic Algorithm by applying a knowledge-based initialization technique. Yu Zhu and lin wu [2] proposed the solution of travelling salesman problem using their new approach using quadratic partitioning. The quadratic partitioning approach provide a solution that was suited well for complex optimization problems. The proposed algorithm was implemented and the algorithm provide good results for the optimization problem. Ameera Jaradat et al. [3] proposed a solution for travelling salesman problem using firefly algorithm. The author proposed a solution using K-mean clustering approach for its solution. The major steps involved in the approach includes creating cluster of nodes, then finding the best path for each cluster then find a global solution by combining these clusters. Sahib Singh Juneja [4] et al. proposed an optimized solution of travelling salesman problem using genetic algorithm. The optimization capability of genetic algorithm to find optimized solution using selection, cross over and mutation was used to solve travelling salesman problem. The work was able to find more optimal solution to some standard TSP instances. Maaki Sakai et al. [5] proposed an optimized cross over operator for the genetic algorithm. Edge assembly operator was a new cross over operator proposed in that work. It improves the diversity in the chromosomes of the population by using EAX cross over. The performance of the EAX operator was checked in small instances of TSP i.e. in which number of cities are less. Gao Ying and Ye Jianwei [6] proposed a normal distribution based genetic algorithm for solving travelling salesman problem. The sequence of the normal distribution was used as a library of random numbers.

In the process of genetic algorithm some random numbers are used to perform swapping in mutation and cross over operations. The location of these positions were generated using random numbers taken from sequence. This method improved the convergence speed of the genetic algorithm. Kureichik V. M et al. [7] proposed an improved genetic algorithm for solving travelling salesman problem. To analyze the quality of the population, the paper introduces new indicators to find the quality of the chromosomes in the population. The work was implemented on some instances of the TSP problem and its results were found good. Junjun Liu et al. [8] proposed a new mutation operation for genetic algorithm using greedy approach and apply that operator on travelling salesman problem. The population of the genetic algorithm was generated using greedy permutation method. This performance of the new method was checked on some instances of the travelling salesman problem and its results were found good. Juju Liu and Wenzheng li proposed parallel solution of large scale Travelling Salesman Problem by using clustering and evolutionary algorithms. It is aimed to solve the Traveling Salesman Problem, by partitioning the problem with a clustering technique, K-Means, and solving these pieces with Genetic Algorithm and finally adding these solutions into one. As results suggest after experiment, in comparison to solving them by partitioning them yields more convincing result than solving by large scale optimization problems as single problems, in both solution quality and time. Many other researchers [9-15] also proposed a lot of techniques of this area. After going through this literature, it is observed that the genetic algorithm solves the TSP problem very well. However, there is a need for improving the performance of genetic algorithm to solve TSP problem. The next section discusses some proposed work in this direction.

III. PROPOSED WORK

In this paper a new genetic algorithm with improved genetic operators is proposed. The genetic algorithm is having selection, reproduction and mutation three genetic operators. The performance of genetic algorithm depends on the performance of all these operators. Better are the genetic operators better is the performance of genetic algorithm. In this paper the cross over operator is improved. The proposed cross over operator is based on a greedy approach. The nearest neighbor first is a very popular greedy approach. This approach states that the next node selected in a TSP path is the nearest node from the current node. When we select a starting city then we will move to the next city which is at minimum distance from the current city.

To explain the proposed approach let the TSP problem is having 5 nodes. The distance between the nodes is shown in Table 1.

Table 1 is showing the distance matrix of a sample network of five nodes. The first row and first column in the table 1 are showing the node numbers from node1 to node5. The remaining cells defines the distance between the two nodes. The table is showing the symmetric matrix i.e. the distance from node1 to node4 is 70 and the distance between node4 to node1 is also 70. So it is symmetric matrix.

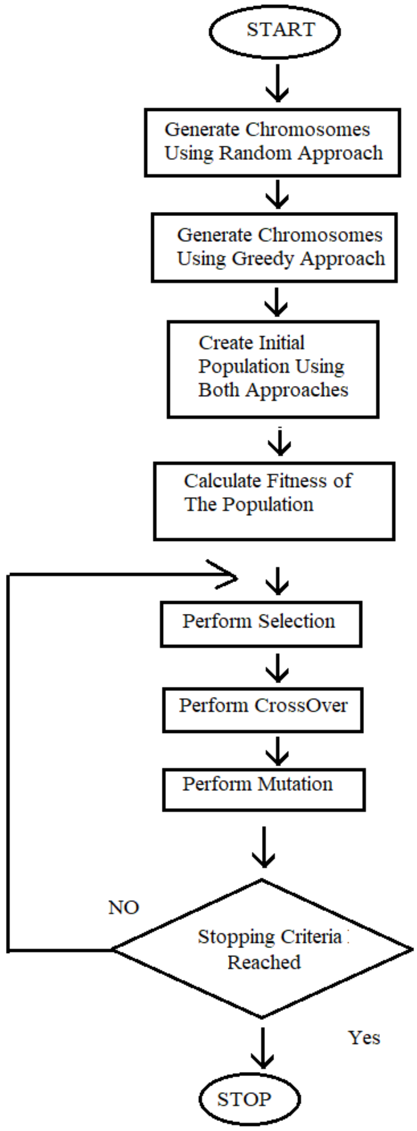

Fig. 2. Flowchart of Greedy genetic algorithm

TABLE I. THE DISTANCE BETWEEN THE NODES OF THE NETWORK

TABLE II.

Node No	1	2	3	4	5
1	0	50	40	70	60
2	50	0	30	20	80
3	40	30	0	50	60
4	70	20	50	0	40
5	60	80	60	40	0

Let we have to generate 5 chromosomes using random approach. The table 2 is showing the chromosomes of the random paths.

TABLE III. THE PATHS GENERATED USING RANDOM APPROACH

Path1	1	2	3	4	5	1
Path2	1	4	3	2	5	1
Path3	1	2	5	4	3	1
Path4	1	3	2	4	5	1

The table 3 is showing the paths generated using greedy approach. From the starting node, all the cities are selected just after the node1. So after, all the remaining nodes are sleeted one by one. For example, in the Greedy Path 1, we select node2 just after the node1. The remaining nodes are 3,4, and 5. The node which is nearest to node2 i.e. node4 is selected. After node4, the node which is nearest to node4 is selected as the next node. Because node5 is nearer than node3 that's why node5 is the next node after node4. Then the only remaining node is node3 which is selected as the last node. Then starting node 1 is selected as the last node and the GreedyPath is over. In this way the other three GreedyPaths shown in table 3 are also generated.

TABLE IV. THE PATHS GENERATED USING GREEDY APPROACH

GreedyPath1	1	2	4	5	3	1
GreedyPath2	1	3	2	4	5	1
GreedyPath3	1	4	2	3	5	1
GreedyPath4	1	5	4	2	3	1

To select the final initial population, the paths generated using random approach i.e. Path1 to Path4 and the paths generated using greedy approach i.e. GreedyPath1 to GreedyPath4 are generated. The other steps of genetic algorithm i.e. selection, cross over and mutation are remaining unchanged and are taken from the standard genetic algorithm. The results obtained are explained in next section.

IV. RESULT AND ANALYSIS

The proposed genetic algorithm is implemented in JAVA programming language and applied on two TSP instances EIL51 and EIL76. These are the instances which are available on TSPLIB standard data set. EIL51 instance is having 51 nodes in the graph and EIL76 instance have 76 nodes in the graph.

The position of all the nodes is given and we have to find a path that traverses each nodes in these graph exactly once. Table 3 is sowing the results of the algorithm proposed by A. Jaradat et al. The results of the proposed algorithm are shown in Table 4. The results shown in Table 4 indicates that the proposed genetic algorithm performs better as compared to other exiting techniques.

TABLE V. RESULT COMPARISON OF PROPOSED ALGORITHM WITH OTHER TECHNIQUES.

Data Set	Optimal Path	Algorithm	Min	Avg
Eil 51	426	Ant Colony using Clustering	484	
		Genetic Algorithm using Clustering	484	
		Firefly Algorithm	435	451
		Firefly using Clustering	485	531
		Proposed Greedy Genetic Algorithm	468	487
Eil76	538	Ant Colony using Clustering	624	-
		Genetic Algorithm using Clustering	624	-
		Firefly Algorithm	-	-
		Firefly using Clustering	613	633
		Proposed Greedy Genetic Algorithm	571	586

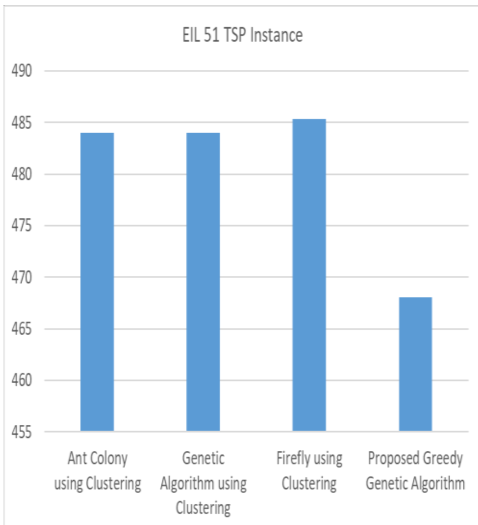

Fig. 3. Result comparison of Eil51 TSP Instance

Figure 3 shows a bar chart of the minimum path length of the tour by proposed algorithm and other algorithm [3]. It also shows that the proposed algorithm performs better as it is having the less path length. Figure 3 shows the result comparison of EIL51 instance and Figure 4 shows the result comparison for the EIL76 TSP instance.

Fig. 4.

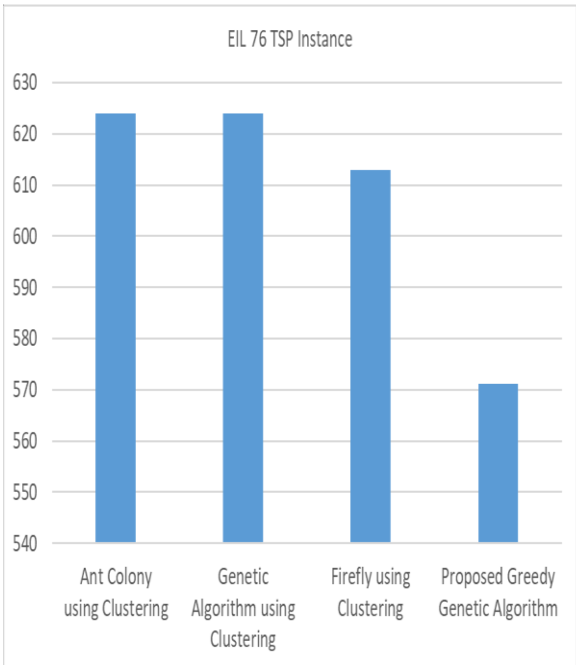

Fig. 5. Result comparison of Eil76 TSP Instance

V. CONCLUSION AND FUTURE WORK

This paper proposed an improved genetic algorithm to solve TSP problem. The TSP problem have many applications in science and engineering. It is an optimization problem in which the path length to traverse all the nodes of a graph is minimized. The genetic algorithm is very useful in optimization problems. An improved genetic algorithm was proposed in this paper and applied on standard TSP instances. It is concluded that the genetic algorithm solves TSP minimization problem very efficiently. The performance of genetic algorithm to solve TSP problem is improved by applying improved genetic operators. So there is a further scope of research to use different techniques on genetic operators of genetic algorithm so that its performance can be made better.

REFERENCES

[1] Chao Li, Xiaogeng Chu, Yingwu Chen and Lining Xing, "A knowledge-based initialization technique of genetic algorithm for the travelling salesman problem," *2015 11th International Conference on Natural Computation (ICNC)*, Zhangjiajie, 2015, pp. 188-193. doi: 10.1109/ICNC.2015.7377988

[2] Y. Zhu and L. Wu, "Structure Study of Multiple Traveling Salesman Problem using Genetic Algorithm," *2019 34rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, Jinzhou, China, 2019, pp. 323-328. doi: 10.1109/YAC.2019.8787633

[3] A. Jaradat, B. Matalkeh and W. Diabat, "Solving Traveling Salesman Problem using Firefly algorithm and K-means Clustering," *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, Amman, Jordan, 2019, pp. 586-589. doi: 10.1109/JEEIT.2019.8717463

[4] S. S. Juneja, P. Saraswat, K. Singh, J. Sharma, R. Majumdar and S. Chowdhary, "Travelling Salesman Problem Optimization Using Genetic Algorithm," *2019 Amity International Conference on Artificial Intelligence (AICAI)*, Dubai, United Arab Emirates, 2019, pp. 264-268. doi: 10.1109/AICAI.2019.8701246

[5] M. Sakai, Y. Hanada and Y. Orito, "Edge Assembly Crossover using Multiple Parents for Traveling Salesman Problem," *2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS)*, Toyama, Japan, 2018, pp. 474-477. doi: 10.1109/SCIS-ISIS.2018.00088

[6] Y. Gao and J. Ye, "An Improved Genetic Algorithm Based on Normal Distribution for Solving the Traveling Salesman Problem," *2018 International Conference on Virtual Reality and Intelligent Systems (ICVRIS)*, Changsha, 2018, pp. 360-362. doi: 10.1109/ICVRIS.2018.00094

[7] V. M. Kureichik and J. A. Logunova, "Some of the New Indicators in Genetic Algorithms for the Traveling Salesman Problem," *2018 IEEE East-West Design & Test Symposium (EWDTS)*, Kazan, 2018, pp. 1-5. doi: 10.1109/EWDTS.2018.8524713

[8] J. Liu and W. Li, "Greedy Permuting Method for Genetic Algorithm on Traveling Salesman Problem," *2018 8th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, Beijing, 2018, pp. 47-51. doi: 10.1109/ICEIEC.2018.8473531

[9] D. R. Singh, M. K. Singh and T. Singh, "Multiple traveling salesman problem using novel crossover and group theory," *2017 International Conference on Computing, Communication and Automation (ICCCA)*, Greater Noida, 2017, pp. 368-372. doi: 10.1109/CCAA.2017.8229844

[10] B. K. Widiartha, S. E. Anjarwani and F. Bimantoro, "Traveling salesman problem using multi-element genetic algorithm," *2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)*, Lombok, 2017, pp. 1-4. doi: 10.1109/TSSA.2017.8272917

[11] Y. Tan, L. Tan, G. Yun and W. Zheng, "Bilevel Genetic Algorithm with Clustering for Large Scale Traveling Salesman Problems," *2016 International Conference on Information System and Artificial Intelligence (ISAI)*, Hong Kong, 2016, pp. 365-369. doi: 10.1109/ISAI.2016.0084

[12] J. Stastný, V. Skorpil and L. Cizek, "Traveling Salesman Problem optimization by means of graph-based algorithm," *2016 39th International Conference on Telecommunications and Signal Processing (TSP)*, Vienna, 2016, pp. 207-210. doi: 10.1109/TSP.2016.7760861

[13] X. Meng, J. Li and X. Dai, "Selection of solution strategies for colored traveling salesman problems with different city distribution," *2016 13th International Workshop on Discrete Event Systems (WODES)*, Xi'an, 2016, pp. 338-342. doi: 10.1109/WODES.2016.7497869

[14] G. Singh, N. Gupta and M. Khosravy, "New crossover operators for real coded genetic algorithm (RCGA)," *2015 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, Okinawa, 2015, pp. 135-140. doi: 10.1109/ICIIBMS.2015.7439507

[15] Y. Gao and J. Ye, "An Improved Genetic Algorithm Based on Normal Distribution for Solving the Traveling Salesman Problem," *2018 International Conference on Virtual Reality and Intelligent Systems (ICVRIS)*, Changsha, 2018, pp. 360-362. doi: 10.1109/ICVRIS.2018.00094