

Algorithm for Multi- objective Traveling Salesman Problems based on Modified Transitive closure



T. Karthy, K. Ganesan

Abstract: *One of the challenging facts of the Multi Objective Traveling Salesman Problem (MOTSP) is to find the best compromised solution. In this paper, we have proposed a modified transitive closure algorithm to solve MOTSP using Genetic Algorithm (GA). Modified Transitive Closure method generates all the initial solutions of each objective. By applying Genetic Algorithm (GA), compromised solutions are obtained. Numerical examples are provided to show the efficiency of the proposed algorithm for MOTSP*

Keywords: *Traveling salesman problem, Route Conditions, modified transitive closure method, Genetic Algorithm.*

I. INTRODUCTION

The Traveling Salesman problem, a widely used mathematical approach to solve several advanced practical logistics and supply-chain scenarios. First discovered and put to use in 1950's, the objective was to find an optimal route for a salesman to visit each of the n cities exactly once and return to the starting city, without wasting the salesman's resources. There are various approaches like the Hungarian method, the Branch and Bound technique, Kruskal's algorithm, Tabu search, evolutionary algorithms, etc. available to solve traveling salesman problems.

Considering the Multi-Objective TSP for which the aim is to obtain or to approximate the set of efficient solutions. The solutions potential can be brought to its fullest when evolving the algorithm using a population of solutions and Multi-Objective Evolutionary Algorithms (MOEAs), thus able to approximate the Pareto optimal set.

Christof and Kenneth[1] described the major approaches to arrive at a solution for the users to choose from the partner efficiency approach and the coalition efficiency approach. While the former/latter benefits the logistics optimization model for interests in individual partner, however, here the individual partner has to set the objectives on its own set. Zutong Wang et. al. [14] reveals how to obtain the Pareto efficiency route through the uncertain approach. An Optimistic route to determine the TSP is prone to be poor in

UMTSP. The same can be benchmarked to newer UMTSP

case studies. Hence, a later ABC algorithm consisting of various operators such as crossover, reverse, and mutation is used in the problem design. Outperforming other algorithms on the basis of performance comparison as seen on three benchmark TSPs. Raquel and Ana[9] proposed heuristic procedures to improve the upper bound for the family of traveling salesman problems theoretical and practical comparisons are done. The authors solved the problem with 127 nodes in 70 seconds and 280 nodes in 3615 seconds. Thibaut and Jacques [10] explains the two-phase Pareto Local Search. The first phase is to generate the efficient solutions of extreme support from an initial available population. The second phase being an application of the Pareto local search to all available solutions from the initial population, without having to use any numerical parameter. It is shown that when combining the proposed two techniques almost always gives quality test results better than other existing state of the art algorithms. Iraklis-DimitriosPsychas et. al. [3] analysed the characteristics that are common for three hybrid evolutionary algorithms. The proposed holds true to all Multi objective Traveling Salesman Problems by modifying the constraint of the problem and the objective, making it possible to work with more than two objective functions. It is observed that the exploitation and exploration factors can be increased of said algorithm seamlessly giving highly optimized MTSP with an evolutionary model, just by using a hybridized global search algorithm. Kanimozhi Jayamoorthi et. al.[4] explained how the evaluation of the solution can be used in MOEAs (Multi Objective Evolutionary Algorithms) to approximate in just a single run using the Pareto optimal set. The time difference is significant when arriving at a nearest optimal solution as it involves optimizing many objectives.

The existing methods motivated us to think why only one of the paths is taken into consideration for solving MOTSPs, while there are many leading to the same initial solution wherein the decision maker can be given choices for selecting the appropriate initial solution which undergoes GA resulting in better compromised solution. reputed journal that published research articles globally.

In this research, modified transitive closure algorithm is coded in Java. MOTSPs will require the user to decide the initial population from the solutions of each objective.

Revised Manuscript Received on December 30, 2019.

* Correspondence Author

T. Karthy, Department of Mathematics, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chennai-603203, India. Email: karthy@srmist.edu.in

K. Ganesan*, Department of Mathematics, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chennai-603203, India. Email: ganesank@srmist.edu.in.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](http://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)



The results are then mutated through cross over to give a compromised path. This path is then fed into Python which displays every path that the user chose, scatter diagram and the compromised route graphically. By using proposed method, the decision maker will easily select the tour of each objective for the initial population of GA. After the selection of initial population, the tours are crossed over and the resultant is mutated randomly which provides the compromise solution for all the objectives.

The paper is organized as follows. In section 2, the basic definitions for a multi objective traveling salesman problems are stated. Section 3 explains the proposed method. Section 4 provides numerical illustrations. Section 5 is about discussions and comparison of results. In Section 6 we draw conclusions. Finally, Section 7 is about advantages and limitations.

II. PRELIMINARIES

For any given multi objective optimization problem, there may be an infinite number of Pareto optimal points constituting the Pareto optimal set. Considering the same from the researchers brings in much concerned to the extreme limitation of the solution spectrum, commonly known as the Pareto front. In practice, the Pareto front suggests a guideline to engineers to stay within said limit. Hence, by definition, the Pareto solution set is the nondominated solution among all optimum points. The Predominant concept in defining an optimal point can be explained by Pareto optimality which is defined as follows:

Definition[5]: If the objective functions are said to be conflicting, then there exists a Pareto optimal solution.

Definition[5]: A solution is called non dominated, pareto optimal, pareto efficient or non inferior, if none of the objective functions can be improved in value without degrading some of the other objective values.

Definition[5]: Pareto efficiency or pareto optimality is a state of allocation of resources in which it is impossible to make anyone individual better off without making atleast one individual worse off.

Definition[5]: If the compromised solution satisfies the decision maker’s preferences, then the solution is called the preferred compromised solution.

III. MULTIOBJECTIVE TRAVELING SALESMAN PROBLEM

Consider a TSP of n cities and distance dij (or cost zij or time tij) from city i to city j. The salesman starts from city i travels through all the other cities exactly once and returns to the city j. Initial route is selected from k objectives of the salesman. Selected route is crossed over and mutated.

$$\text{Min } Z(x) = Z_k(x) = C^k \mathbf{x} \text{ where } k = 1, \dots, p \text{ such that } \mathbf{x} \in X = \{\mathbf{x} \in Z_+^n; A\mathbf{x} = \mathbf{b}\}$$

$$X_{ij} = \begin{cases} 1; & \text{if salesman travels from city } i \text{ to city } j \\ 0; & \text{otherwise} \end{cases}$$

and

$$C = \begin{pmatrix} \infty & C_{12} & C_{13} & \dots & C_{1n} \\ C_{21} & \infty & C_{23} & \dots & C_{2n} \\ C_{31} & C_{32} & \infty & \dots & C_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ C_{n1} & C_{n2} & \dots & C_{n(n-1)} & \infty \end{pmatrix}$$

A. Modified Transitive Closure Algorithm

- Step 1: Divide all elements of the row by the minimum of respective row of the effectiveness matrix.
- Step 2: From the resultant matrix divide all the elements of the column by the minimum of respective column, denoted as W.
- Step 3: From W, write the positions of 1s in column 1 as i and also write the positions of 1s in row 1 as j. W₁ is obtained by replacing the position (i, j, i != j) of W by 1 and rest of the other elements are unaltered.
- Step 4: Similarly, perform n iterations to obtain W_n.
- Step 5: Select exactly one 1 in each row and column of W_n to get the tour.
- Step 6: If the route conditions of TSP are not satisfied, select the next higher value of the cost matrix to satisfy the route condition.

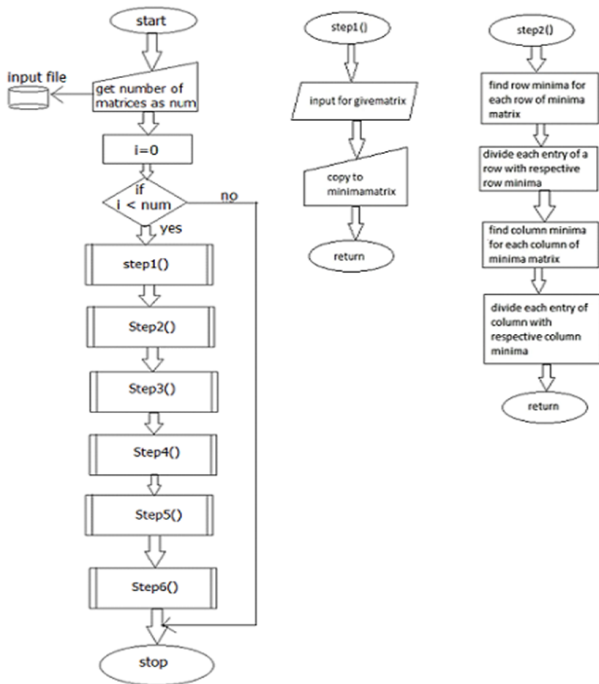
Java code has been used for implementing the modified transitive closure algorithm. The user uses the following syntax for the path of the algorithm. (“C://Users//*****//Desktop//sample.txt”); enter during the run-time causes human errors, hence the sample.txt is used as input to simplify the input data and can overcome the issues faced by the user while entering a large matrix. The first data to be entered in the sample.txt represents the number of matrices. The second data to be entered represents the size of the first matrix, say 3, representing a 3 matrix. Subsequently, enter all the elements of the first matrix by entering one value per line. If, there is a second matrix, specify its size, say 4 denoting a 4*4 right matrix after the values of the previous matrix have been completely entered. Continue until all the matrices have been entered. While entering the values of any given matrix, the leading diagonal element (∞) must be represented as ‘1’.

Once the java program is run, it will display all paths adding to the optimal solution if it is a single objective problem else the same becomes the initial solution in the case of the multi objective travelling salesman problem. If the problem is of the multi objective travelling salesman category, the data is entered.

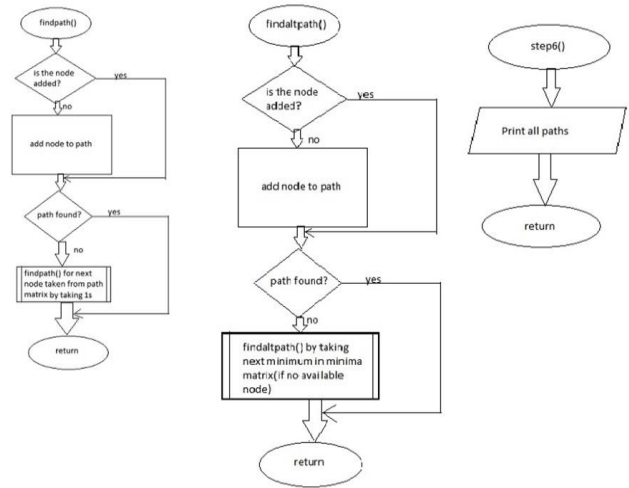
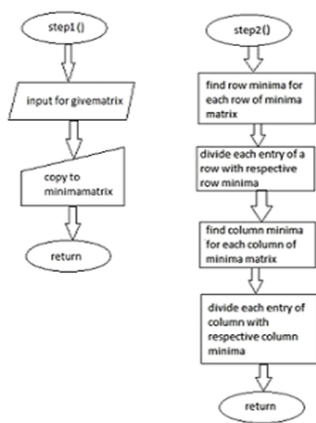
using Python to get a graphical representation of said solution. The dots represent the cities in the form of a scatter diagram, whereas the lines represent the initial path obtained. GA is applied to the initial paths.



B. Flow Chart



Flow chart 1 (a)



Flow chart 1(c)

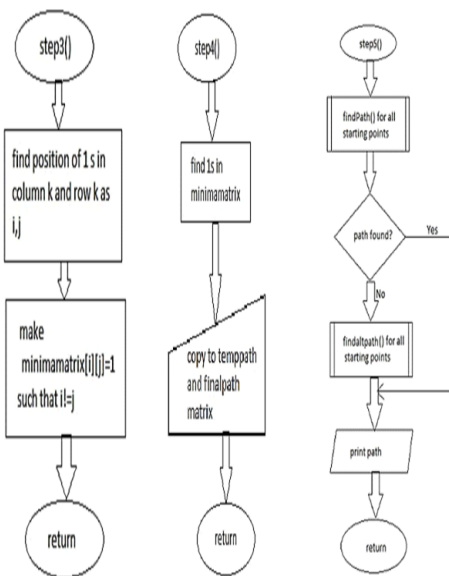
IV. ILLUSTRATIONS

Example 4.1 Consider the problem discussed by Matthias Ehrgott[5], Salesman travels six cities wherein his objective is to minimize the traveling time, CO2 emission from vehicle and also the expenditure of his travel. The tri objective represented as matrices.

$$\begin{bmatrix} \infty & 1 & 6 & 5 & 5 & 5 \\ 1 & \infty & 5 & 4 & 6 & 4 \\ 6 & 5 & \infty & 5 & 6 & 1 \\ 5 & 4 & 5 & \infty & 6 & 8 \\ 5 & 6 & 6 & 6 & \infty & 2 \\ 5 & 4 & 1 & 8 & 2 & \infty \end{bmatrix}; \begin{bmatrix} \infty & 57 & 55 & 24 & 19 & 46 \\ 57 & \infty & 151 & 126 & 121 & 137 \\ 55 & 151 & \infty & 121 & 90 & 117 \\ 24 & 126 & 121 & \infty & 34 & 61 \\ 19 & 121 & 90 & 34 & \infty & 27 \\ 46 & 137 & 117 & 61 & 27 & \infty \end{bmatrix};$$

$$\begin{bmatrix} \infty & 39 & 173 & 6 & 249 & 45 \\ 39 & \infty & 354 & 348 & 430 & 25 \\ 173 & 354 & \infty & 511 & 76 & 404 \\ 6 & 348 & 511 & \infty & 251 & 39 \\ 249 & 430 & 76 & 251 & \infty & 328 \\ 45 & 25 & 404 & 39 & 328 & \infty \end{bmatrix}$$

The input data has been entered using the 'sample.txt' file and is run on Java for the tri-objective optimization problem. When run, the program displays all paths leading to the initial solution under each objective and by using Python, the graphical illustration of the initial paths is given below.



Flow chart 1(b)

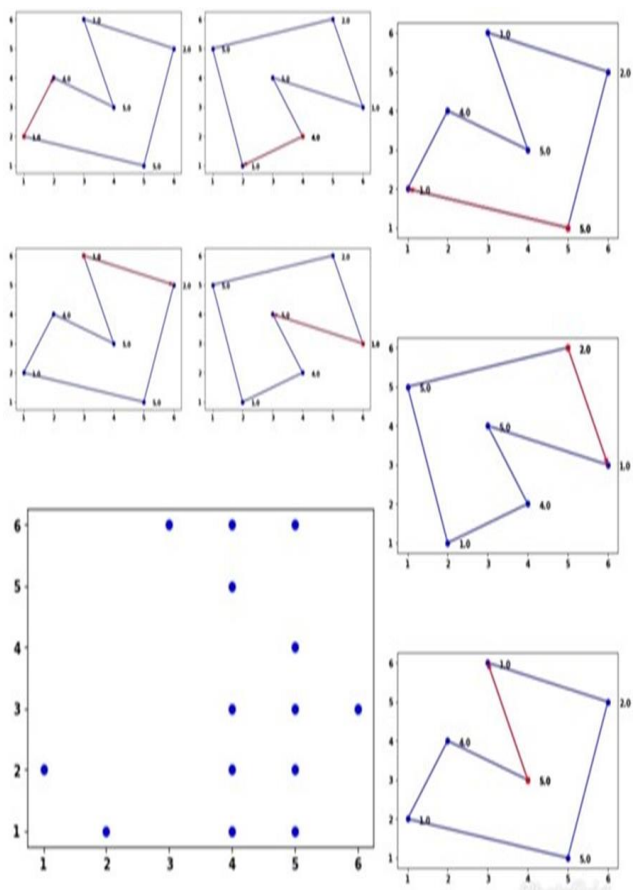


Fig. 1 Graphical illustration of Initial Population of Matrix 1

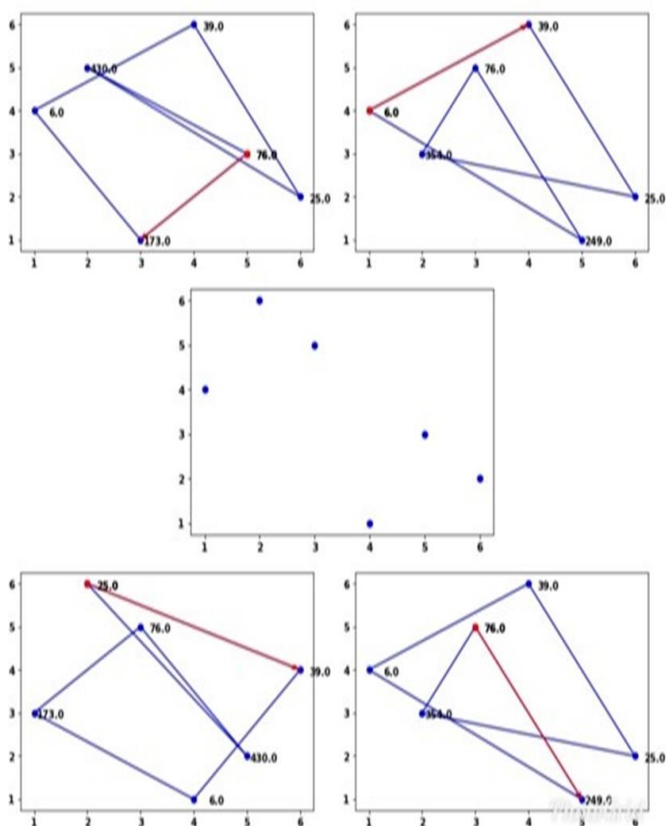


Fig. 2 Graphical illustration of Initial Population of Matrix 2

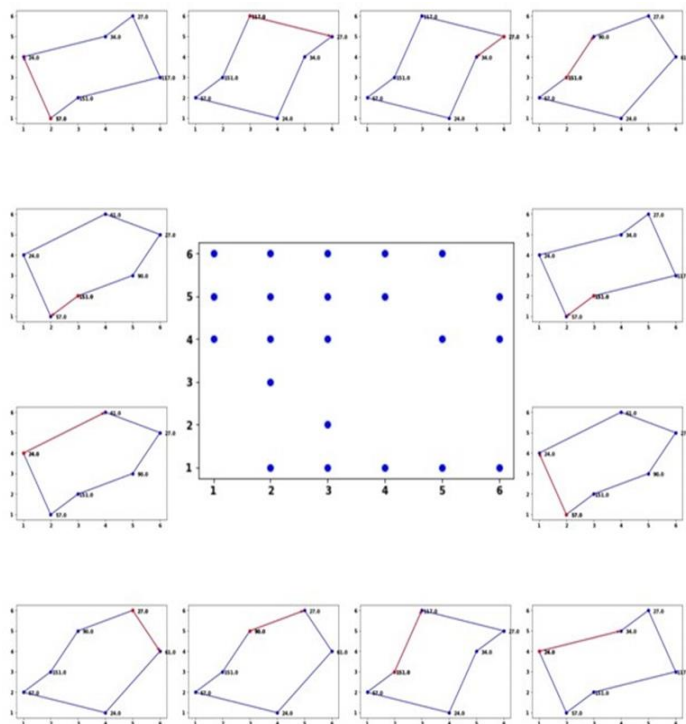


Fig. 3 Graphical illustration of Initial Population of Matrix 3

From the above graphical representation of Matrix 1, Matrix 2, Matrix 3 the decision maker can choose the path for his initial population which undergoes two point, crossover and mutation.

Initial Population

Parent 1: 5-6-3-4-2-1-5; $MinZ_1 = 18$; $MinZ_2 = 467$; $MinZ_3 = 1879$

Parent 2: 5-6-4-1-2-3-5; $MinZ_2 = 410$; $MinZ_3 = 842$; $MinZ_1 = 27$

Parent 3: 5-3-1-4-6-2-5; $MinZ_3 = 749$; $MinZ_1 = 35$; $MinZ_2 = 1558$

Two Point Cross Over

Child 1: 5-1-3-4-6-2-5; $MinZ_1 = 26$; $MinZ_2 = 514$; $MinZ_3 = 1427$

Child 2: 5-6-4-3-2-1-5; $MinZ_2 = 436$; $MinZ_3 = 1520$; $MinZ_1 = 26$

Child 3: 5-1-4-6-2-3-5; $MinZ_3 = 749$; $MinZ_1 = 33$; $MinZ_2 = 509$

Mutation

Child 4: 5-1-2-4-3-6-5; $MinZ_1 = 18$; $MinZ_2 = 467$; $MinZ_3 = 1879$

Child 5: 5-6-4-1-2-3-5; $MinZ_2 = 410$; $MinZ_3 = 842$; $MinZ_1 = 27$

Child 6: 5-1-4-6-3-2-5; $MinZ_3 = 493$; $MinZ_1 = 30$; $MinZ_2 = 25$

Example 4.2 A Salesman to travel five cities to promote company’s product. The Salesman travels all the cities with the objective to reduce the traveling time and expenditure simultaneously. First matrix represents the time taken to travel between the cities and second matrix represent the expenditures to travel between the cities. In short, this is a bi-objective traveling salesman problem, minimizing time and cost whose matrices are given by

$$\begin{bmatrix} \infty & 7 & 6 & 8 & 4 \\ 7 & \infty & 8 & 5 & 6 \\ 6 & 8 & \infty & 9 & 7 \\ 8 & 5 & 9 & \infty & 8 \\ 4 & 6 & 7 & 8 & \infty \end{bmatrix} \text{ and } \begin{bmatrix} \infty & 17 & 16 & 18 & 14 \\ 17 & \infty & 18 & 15 & 16 \\ 16 & 18 & \infty & 19 & 17 \\ 18 & 15 & 19 & \infty & 18 \\ 14 & 16 & 17 & 18 & \infty \end{bmatrix}$$

Elements of the two matrices are entered one after the other in sample.txt where in, the first entry represents the number of matrices, here 2, and the second entry represents the order of the square matrix. In this case, 5. The third entry (-) represents infinity values along leading diagonals. The 28th entry represents the order of the second matrix, here 5.

It is seen that the obtained initial paths in each objective are exact. Henceforth, further heuristic approach is not required.

TSP Tour 2-4-3-1-5-2 or 3-1-5-2-4-3 gives the value of the objective function as $Min Z_1 = 30$ and $Min Z_2 = 80$

V. DISCUSSION AND COMPARISON OF RESULTS

In the Illustration 4.1, the decision maker can choose any one of the compromised solutions according to his /her preference.

5-6-4-1-2-3-5; $MinZ_1 = 27$; $MinZ_2 = 410$; $MinZ_3 = 842$
or
5-1-4-6-3-2-5; $MinZ_1 = 30$; $MinZ_2 = 493$; $MinZ_3 = 1482$
or

5-1-2-4-3-6-5; $MinZ_1 = 18$; $MinZ_2 = 467$; $MinZ_3 = 1879$
which is same as the result obtained by Matthios Erghott [5], in his book multi criterion optimization (Second Edition, Springer) for his TSP tour (1, 2, 4, 3, 5, 6, 1). As the problem is about MOTSP, the decision maker can choose any of the compromised solutions depending upon his/her priority towards each objective.

Table 1

S.No	Optimal solution by the proposed method	Optimal solution obtained by [5]	Comparison
1	5-1-2-4-3-6-5; $MinZ_1=18$; $MinZ_2=467$; $MinZ_3 =1879$	1-2-4-3-5-6-1; $MinZ_1=18$; $MinZ_2=467$; $MinZ_3 =1879$	The proposed method gives the same as [5] .

In the Illustration 4.2, as all the initial paths are the same for both the objectives, we could see that the obtained path itself is the optimal (compromised) solution of each objective. Hence there is no need to develop further using GA and there by saves time. The existing methods gives only one of the initial paths. Further GA is applied for obtaining

compromised solution which consumes time.

VI. CONCLUSION

Through this paper, for MOTSPs, the proposed algorithm helps the decision maker to choose the tour for the initial population of GA. The selected initial population is allowed for cross over and then mutated randomly. The examples shown in this paper guarantees the efficiency of the algorithm. From the table 1, each objective function value obtained is same through this method when compared with the existing results[5] or provides better solutions while holding TSP conditions.

VII. ADVANTAGES AND LIMITATIONS

In MOTSPs, the proposed method gives all the tours leading to the initial solution wherein the decision maker has multiple choices to select the initial solution of each objective and by applying GA, finds the compromised solution.

The algorithm is near to perfection. The only thing holding it down would be the storage factor. When the Matrices are entered, they occupy a small storage in the system. A matrix of larger index might lead to a slight increase in the storage, whilst upholding the original optimization in the form of crossovers and mutations.

REFERENCES

- Christof Defryn, Kenneth Sörensen, “Multi-objective optimisation models for the travelling salesman problem with horizontal cooperation,” European Journal of Operational Research, vol.267, 2018, pp. 891–903.
- M. Dorigo, V. Maniezzo and A. Colomi, “The Ant System: Optimization by a Colony of Cooperating Agents,” IEEE Transactions on Systems, Man and Cybernetics Part B, vol.26,1996,pp.29-41. <http://dx.doi.org/10.1109/3477.484436>.
- Iraklis-Dimitrios Psychas, Eleni Delimpasi, Yannis Marinakis, “Hybrid evolutionary algorithms for the Multiobjective Traveling Salesman Problem,” Expert Systems With Applications, vol. 42, 2015, pp. 8956–8970.
- Kanimozhi Jayamoorthi, Dinesh Karunanidy, Amudhavel Jayavel, Subramanian Ramalingam, “A Survey On Multi-Objective Travelling Salesman Problem”, IIOABJ, Vol. 8, No. 2, pp. 223-233
- Matthias Ehrgott, Multicriteria Optimization. Second Edition, Springer Berlin Heidelberg, New York, 2005, pp. 282-284.
- Nosheen Qamar, Nadeem Akhtar, Irfan Younas, “Comparative Analysis of Evolutionary Algorithms for Multi-Objective Travelling Salesman Problem,” International Journal of Advanced Computer Science and Applications, vol. 9(2), 2018.
- E. Osaba, R Carballedo, F. Diaz, and A. Perallos, “Analysis of the Suitability of Using Blind Crossover Operators in Genetic Algorithms for Solving Routing Problems,” IEEE International Symposium on Applied Computational Intelligence and Informatics, 2013, pp. 17-22.
- J.Y. Potvin, 1993 “The Traveling Salesman Problem: A Neural Network Perspective,” ORSA J. Computing, vol. 5,2013, pp. 328-348. <http://dx.doi.org/10.1287/ijoc.5.4.328>
- Raquel Bernardino, Ana Paia. “Solving the family traveling salesman problem,” European Journal of Operational Research, vol. 267, 2018, pp. 453–466.
- Thibaut Lustand Jacques Teghem, “The Multiobjective Traveling Salesman Problem: A Survey and a New Approach,” Advances in Multi-Objective Nature Inspired Computing. Studies in Computational Intelligence, vol. 272,2010, pp 119-141.
- VelinKralev, “Different Applications of the Genetic Mutation Operator for Symmetric Travelling Salesman Problem,” International Journal On Advanced Science Engineering And Technology, vol. 8: No. 3 ,2018.



12. Y. Wu, T. Weise and R. Chiong, "Local Search for the Traveling Salesman Problem: A Comparartive Study," Proceedings of the 14th IEEE Int'l Conf. on Cognitive Informatics & Cognitive Computing, 2015, pp.213-220. <http://dx.doi.org/10.1109/icci-cc.2015.7259388>.
13. Zili Zhang, Chao Gao, Yuxiao Lu, Yuxin Liu, Mingxin Liang, "Multi-Objective Ant Colony Optimization Based on the Physarum-Inspired Mathematical Model for Bi-Objective Traveling Salesman Problems," 2016, doi.10.1371/journal.pone.014670, January 11.
14. ZutongWang, JianshengGuo, Mingfa Zheng, YingWang, "Uncertain multi objective traveling salesman problem," Decision Support. European Journal of Operational Research, vol. 241, 2015, pp. 478-489.

AUTHORS PROFILE



T. Karthy is an Assistant Professor in the Department of Mathematics, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chennai-603203, India. She received her M.Sc. and M.Phil. degrees from Bharathidasan University. Now, she is pursuing Ph. d. degree in Mathematics. Her area of research is multi objective fuzzy optimization. She published research articles in reputed national and international journals. Also she presented her research articles in both national and international conferences. Further she completed many online courses out of her interest. She is a life member in Indian Mathematical society (IMS) and Indian Science Congress Association (ISCA).



Dr. K. Ganesan is a Professor in the Department of Mathematics, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chennai-603203, India. He received his Doctor of Philosophy in Mathematics from University of Madras. His Area of research includes Fuzzy optimization and Interval Analysis. He published more than 100 articles in reputed national and international journals. He served as resource person for many national and international conferences. He organized many conferences, workshops and member in advisory committee of various conferences. He is reviewer and member of editorial board of many reputed journals. Further, he is life member in Indian Mathematical society (IMS) and Indian Science Congress Association (ISCA), Kerala Mathematical Association (KMA), ISTE etc.