International Journal of Scientific Research in Mathematical and Statistical Sciences
Volume-8, Issue-2, pp.01-04, April (2021)
DOI: https://doi.org/10.26438/ijsrmss/v8i2.14

Research Paper

E-ISSN: 2348-4519

Optimizing the Trapezoidal Fuzzy Travelling Salesman Problem Through Dhouib-Matrix-TSP1 Method Based on Magnitude Technique

Sa. Dhouib [1*], S. Dhouib[2]

[1]OLID Lab (Optimisation, Logistique et Informatique Décisionnelle), Sfax University, Tunisia
Higher Institute of Industrial Management, Sfax University, Sfax, Tunisia

** Corresponding Author : saima.dhouib10@gmail.com*

Available online at: www.isroset.org

Received: 12/Feb/2021, Accepted: 25/Mar/2021, Online: 30/Apr/2021

Abstract**—** In this paper, we propose to optimize the trapezoidal fuzzy Travelling Salesman Problem (TSP). First, we rank the fuzzy matrix of distance using the magnitude ranking technique. Second, the new method named Dhouib-Matrix-TSP1 is applied to easily and quickly search an optimal or a near optimal solution in polynomial computational time. A numerical example will be presented and discussed to prove the efficiency of the Dhouib-Matrix-TSP1 approach.

Keywords—Fuzzy Set, Travelling Salesman Problem, Optimization, Approximation Method, Dhouib-Matrix-TSP1, Magnitude Technique

I. INTRODUCTION

This paper presents the application of our new method, Dhouib-Matrix-TSP1, to solve the travelling salesmen problem in fuzzy environment.

Zadeh defines a fuzzy set in [1] as a mathematical construction by affecting each element in the universe of discourse to a value. This latter represents its membership grade. The number $\tilde{A}$ is a fuzzy set whose membership function $\mu_{\tilde{A}}(x)$ satisfies the following condition:

- $\mu_{\tilde{A}}(x)$ piecewise continuous
- $\mu_{\tilde{A}}(x)$ is convex
- $\mu_{\tilde{A}}(x)$ is normal $\mu_{\tilde{A}}(x_0)=0$

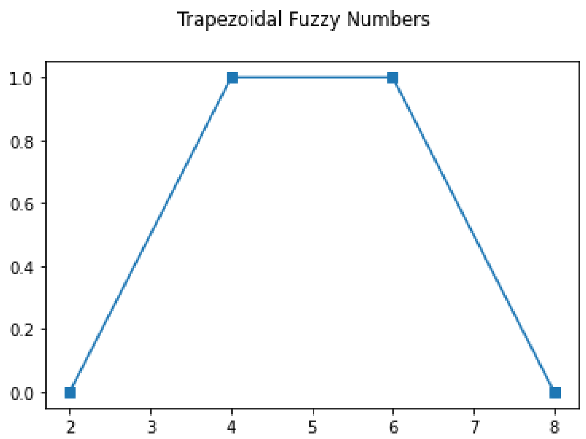

Figure 1. The trapezoidal fuzzy numbers graphical representation

A fuzzy number $\tilde{A}$ on R, as described by Figure 1, is said to be trapezoidal number, represented by (a_1, a_2, a_3, a_4), when the characteristics of its membership function $\tilde{A}: R \to [0,1]$ are:

$$\mu_{\tilde{A}}(x)\begin{cases} 0, & x < a_1 \\ w\left(\dfrac{x-a_1}{a_2-a_1}\right) & a_1 \le x \le a_2 \\ w & a_2 \le x \le a_3 \\ w\left(\dfrac{x-a_4}{a_3-a_4}\right) & a_3 \le x \le a_4 \\ 0 & x \ge a_4 \end{cases} \qquad (1)$$

For a given trapezoidal fuzzy number $\tilde{A}=(a_1,a_2,a_3,a_4;w)$, its magnitude is defined as follows:

$$Mag(u)=\frac{1}{2}\int_0^1 (\underline{u}(r)+\overline{u}(r)+x_0+y_0)\,f(r)dr$$

Well, the *Mag(u)* function is used to rank the fuzzy number and the function *f(r)* is considered as non-negative and increasing function on [0,1].

Where: $f(0)=0$, $\int_0^1 f(r)dr=\frac{1}{2}$ and $f(1)=1$.

Then, the magnitude is given by:

$$Mag(u)=\frac{a_1+5a_2+5a_3+a_4}{12}$$

This paper is organized as follows: section 2 describes some fuzzy preliminaries; section 3 presents the improved Dhouib-Matrix-TSP1 method for fuzzy TSP; section 4 discusses a numerical example in details and section 5 deals with the conclusion.

II. RELATED WORK

The Travelling Salesman Problem (TSP) is NP-Hard and looks to generate a Hamiltonian cycle where all cities are visited only once except the starting one. [2] presents the cutting-plane approach to solve the large TSP. [3] optimizes the multi-objective TSP in fuzzy environment using a genetic algorithm. Besides [4] solves the uncertain multi-objective TSP.

[5] optimizes the TSP for logistics distribution based on a geographical information system. [6] applies the reduce-optimize-expand technique in order to reduce instances of TSP in fuzzy logic classifier. [7] proposes the little algorithm to optimize the TSP.

In 2017, [8] solves the generalized TSP by the means of the neighborhood search algorithm and [9] applies hybrid self-organizing map approach on the TSP. [10] develops an ant system to solve TSP. [11] modifies the pheromone rule for the ant colony algorithm in order to enhance the optimization of the TSP.

In 2019, [12] provides a spiking neural system for TSP. [13] develops an original genetic algorithm using partially mapped crossover and proves its performance to optimize TSP.

Recently, [14] resolves the cost-balanced TSP using a variable neighborhood search algorithm. [15] defines a comprehensive survey on the multiple TSP and [16] solves the TSP using discrete Jaya algorithm.

III. METHODOLOGY

We recently design and develop the Dhouib-Matrix-TSP1 in [17, 18]. This method generates optimal or a near optimal solution in polynomial computational time: just only *n* simple iterations, where *n* denotes the number of cities.

Then, to adapt the Dhouib-Matrix-TSP1 to the fuzzy set theory, we convert the considered trapezoidal fuzzy distance matrix to a crisp distance matrix as a result of the magnitude ranking function. Then, we test if the generated crisp distance matrix is square, if not we will transform it as square matrix by adding crisp rows or columns (with high element).

Phase 1: Identify for each row its minimal element and select the smallest. The position of the smallest element will identify the first two cities to be inserted in the list of *Tour-Cities*. Let's say the cities *x* and *y*, so, *Tour-Cities* will be gathered in the order {*x*-*y*}. Before going to the next phase, discard columns of city *x* and city *y*.

Phase 2: Discover the smallest element for row *x* and row *y* and choose the smallest, let it be the city *z*.

Phase 3: Insert *z* before city *x* in the *Tour-Cities* {*z*-*x*-*y*}, if *z* is the small element in row *x*. Otherwise, add *z* after city *y* in the list *Tour-Cities* {*x*-*y*-*z*}, and discard the column of city *z*. Then, return to Phase 2 if there are still cities in the matrix, if not go to Phase 4.

Phase 4: Perform the generated route in *Tour-Cities* to be a feasible solution (tour). Well, if the first element in the list *Tour-Cities* is not the starting city so translate this element to the last position in the list *Tour-Cities*. Repeat that until the starting city will be at position number one in the list *Tour-Cities*. Finally, add the starting city at the last position in the list *Tour-Cities* and we obtain a feasible solution which can be the optimal solution or the near optimal one.

IV. RESULTS AND DISCUSSION

A truck driver needs to visit six cities in the following trapezoidal fuzzy TSP. So, he desires to visit all cities only once except the starting city with the minimum distance. Figure 2 presents the trapezoidal fuzzy distance matrix.

$$\begin{bmatrix} (0,0,0,0) & (1,2,4,5) & (3,7,10,14) & (2,3,5,6) & (0,1,4,5) & (2,6,7,17) \\ (0,2,4,6) & (0,0,0,0) & (1,4,6,9) & (1,2,3,4) & (1,7,15,21) & (0,5,7,12) \\ (3,7,10,14) & (3,4,6,7) & (0,0,0,0) & (2,5,15,18) & (1,2,5,6) & (1,4,6,9) \\ (1,3,5,7) & (0,2,3,5) & (2,5,15,18) & (0,0,0,0) & (0,2,16,18) & (1,9,16,18) \\ (0,1,4,5) & (1,7,15,21) & (2,3,4,5) & (0,2,16,18) & (0,0,0,0) & (0,4,7,17) \\ (2,6,7,17) & (0,4,7,17) & (0,4,6,10) & (0,5,7,12) & (0,4,7,17) & (0,0,0,0) \end{bmatrix}$$

Figure 2. The trapezoidal fuzzy distance matrix

Then, we convert the considered trapezoidal fuzzy distance matrix to a crisp distance matrix (Figure 3) by the use of the magnitude ranking function.

$$\begin{bmatrix} 0 & 3 & 8.5 & 4 & 2.5 & 7 \\ 3 & 0 & 5 & 2.5 & 11 & 6 \\ 8.5 & 5 & 0 & 10 & 3.5 & 5 \\ 4 & 2.5 & 10 & 0 & 9 & 12 \\ 2.5 & 11 & 3.5 & 9 & 0 & 6 \\ 7 & 6 & 5 & 6 & 6 & 0 \end{bmatrix}$$

Figure 3. The crisp distance matrix

Now, we can start our approximation method: the Dhouib-Matrix-TSP1 which will easily solve this problem in just 6 simple iterations.

Iteration 1: Identify the minimal element for each row, as depicted in Figure 4.

$$\begin{bmatrix} 0 & 3 & 8.5 & 4 & \boxed{2.5} & 7 \\ 3 & 0 & 5 & \boxed{2.5} & 11 & 6 \\ 8.5 & 5 & 0 & 10 & \boxed{3.5} & 5 \\ 4 & \boxed{2.5} & 10 & 0 & 9 & 12 \\ 2.5 & 11 & \boxed{3.5} & 9 & 0 & 6 \\ 7 & \boxed{6} & 5 & 6 & 6 & 0 \end{bmatrix} \begin{matrix} 2.5 \\ 2.5 \\ 3.5 \\ 2.5 \\ 3.5 \\ 6 \end{matrix}$$

Figure 4. Compute the minimal element for each row

Next, select the smallest value which is 2.5 for the position d_{15}. Thus, the cities 1 and 5 will be inserted in *Tour-Cities* {1-5} and their columns will be discarded (see Figure 5).

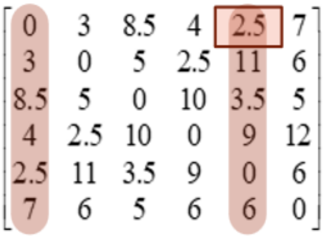

Figure 5. Select cities 1 and 5 and discard their columns

Iteration 2: Identify the minimal element for row 1 (which is 3 in position d_{12}) and for row 5 (which is 3.5 at position d_{53}). Then, select the smallest (3 in position d_{12}), insert city 2 at the left (because we found it in row 1) of the list *Tour-Cities* {2-1-5} and discard the column of city 2 (see Figure 6).

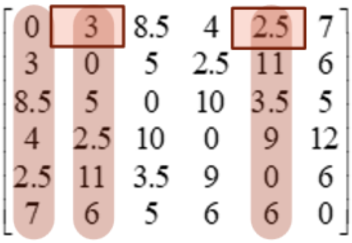

Figure 6. Select city 2 and discard its column

Iteration 3: Identify the minimal element for row 2 (which is 2.5 in position d_{24}) and for row 5 (which is 3.5 at position d_{53}). Then, select the smallest value (2.5 in position d_{24}), insert city 4 at the left (because we found it in row 2) of the list *Tour-Cities* {4-2-1-5} and discard the column of city 4 (see Figure 7).

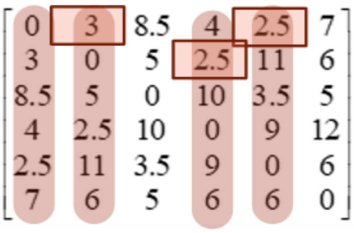

Figure 7. Select city 4 and discard its column

Iteration 4: Identify the minimal element for row 4 (which is 10 in position d_{43}) and for row 5 (which is 3.5 at position d_{53}). Then, select the smallest value (3.5 in position d_{52}), insert city 3 at the right (because we found it in row 5) of the list *Tour-Cities* {4-2-1-5-3} and discard the column of city 3 (see Figure 8).

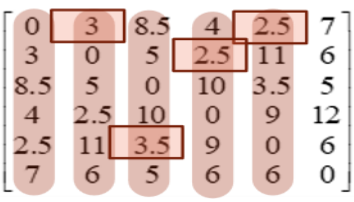

Figure 8. Select city 3 and discard its column

Iteration 5: Identify the minimal element for row 4 (which is 12 in position d_{45}) and for row 3 (which is 5 at position d_{36}). Then, select the smallest value (5 in the position d_{36}), insert city 6 at the right (because we found it in row 3) of the list *Tour-Cities* {4-2-1-5-3-6} and discard the column of city 6 (see Figure 9).

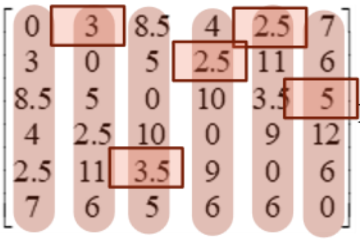

Figure 9. Select city 6 and discard its column

Iteration 6: The last phase is to generate a tour from the list *Tour-Cities* {4-2-1-5-3-6}. So, we need to translate the position in order to obtain a feasible solution starting and ending by city number 1. For that, translate the position of city 4 to the last position {2-1-5-3-6-4}. Then translate city 2 at the last position {1-5-3-6-4-2} and finally add city number 1 to the last position: {1-5-3-6-4-2-1}. Thus, our method Dhouib-Matrix-TSP1 found the optimal solution 22.5 in just 6 simple iterations.

V. CONCLUSION AND FUTURE SCOPE

In this research work, the trapezoidal fuzzy travelling salesman problem is reduced into the classical Travelling Salesman Problem by applying the suitable ranking magnitude technique. Then, the new method, named Dhouib-Matrix-TSP1, is applied to search the optimal or near optimal solution in simple and reduced number of iterations (just *n* iterations, where *n* denotes the number of cities). Additionally, our proposed method Dhouib-Matrix-TSP1 provides a better solution than those obtained by other methods.

REFERENCES

[1] L. A. Zadeh, "Fuzzy Sets," *Information and Control*, **Vol. 8, No.3, pp.338-353, 1965.**
[2] G. Dantzig, D. Fulkerson and S. Johnson, "Solutions of a large-scale traveling-salesman problem," *Operations Research*, **Vol.2 No.4, pp.393-410, 1954.**
[3] C. Changdar, S. G. Mahapatra and R. P. Kumar, "An efficient genetic algorithm for multi-objective solid travelling salesman problem under fuzziness," *Swarm and Evolutionary Computation*, **Vol.15, pp.27–37, 2014.**
[4] Z. Wang, J. Guo, M. Zheng and Y. Wang, "Uncertain multiobjective traveling salesman problem," *European Journal of Operational Research*, **Vol.241 No. 2, 2015.**
[5] W. Gu, Y. Liu Y., L. Wei and B. Dong, "A Hybrid Optimization Algorithm for Travelling Salesman Problem Based on Geographical Information System for Logistics Distribution," *International Journal of Grid Distribution Computing*, **Vol.8 No. 3, pp.359-370, 2015.**
[6] J. D. D. Francisco, M. Oscar and R. Sepúlveda, "Reducing the size of traveling salesman problems using vaccination by fuzzy selector," *Expert Systems with Applications*, **Vol.49, pp.20-30, 2016.**

[7] V. Vasilchikov,“On the Optimization and Parallelizing Little Algorithm for Solving the Traveling Salesman Problem,” *Modeling and Analysis of Information Systems*, **Vol. 23 No.4, pp.401-411, 2016.**

[8] L. S. Smith and F. Imeson, “GLNS: an effective large neighborhood search heuristic for the generalized traveling salesman problem,” *Computer Operatioanl Research*, **Vol.87, pp.1-19, 2017.**

[9] P. Tinarut and K. Leksakul, “Hybrid self-organizing map approach for traveling salesman problem”, *Chiang Mai University Journal of Natural Sciences*, **Vol.18 No.1, pp.27-37, 2019.**

[10] M. Divya, “Solving Travelling Salesman Problem Using Ant Systems: A Programmer’s Approach,” *International Journal of Applied and Computational Mathematics*, **Vol.5, pp.101, 2019.**

[11] S. Suriya and M. Rahul Kumar, "Solving Travelling Salesman Problem using an Enhanced Ant Colony Algorithm," *International Journal of Computer Sciences and Engineering*, **Vol.7, Issue-7, July 2019.**

[12] H. Zhang, L. Xiang and X. Liu, “A SN P System for Travelling Salesman Problem,” *International Conference on Human Centered Computing*, **pp.339-346, 2019.**

[13] M. S. Hardi, M. Zarlis, S. Effendi and S. M. Lydia, “Taxonomy Genetic Algorithm For Implementation Partially Mapped Crossover In Travelling Salesman Problem,” *International Conference on Advanced Information Scientific Development (ICAISD)*, **Vol.1641, 2020.**

[14] A. A. Mehmet, B. C. Kalayci, “A Variable Neighborhood Search Algorithm for Cost-Balanced Travelling Salesman Problem,” *Metaheuristics for Combinatorial Optimization*, **pp. 23-36, 2021.**

[15] O. Cheikhrouhoua and I. Khoufi, “A comprehensive survey on the Multiple Traveling Salesman Problem: Applications, approaches and taxonomy,” *Computer Science Review*, **vol.40, pp.100369, 2021.**

[16] M. Gunduza and M. Aslan, “DJAYA: A discrete Jaya algorithm for solving traveling salesman problem”, *Applied Soft Computing*, **Vol. 105, pp.107275, 2021.**

[17] S. Dhouib, “A New Column-Row Method for Traveling Salesman Problem: The Dhouib-Matrix-TSP1,” *International Journal of Recent Engineering Science*, **Vol. 8, No. 1, pp. 6-10, 2021.**

[18] S. Dhouib, “Stochastic Column-Row Method for Travelling Salesman Problem: the Dhouib-Matrix-TSP2,” *International Journal of Engineering Research & Technology*, **Vol. 10, Issue 3, pp. 524-527, 2021.**

AUTHORS PROFILE

Saima DHOUIB obtained her PhD in Quantitative Methods at the University of Sfax, Tunisia in 2016. She has published her research papers in several international journals such as: International Journal of Metaheuristics, International Journal of Manufacturing Technology and Management, etc. Her current research interests include Supply Chain Management and optimization.

Souhail Dhouib is a Full Professor at Sfax University, Tunisia. He received his BSc in Management Information System, his Master in Operations Research and his PhD in Quantitative Methods. His teaching and research interests are related to the areas of Decision Science, Computer Science, Supply Chain Management, Logistic etc. His publications have appeared in many journals.