Neutrosophic Sets and Systems, Vol. 51, 2022
University of New Mexico

Novel Heuristic for New Pentagonal Neutrosophic Travelling Salesman Problem

Souhail Dhouib *

OLID laboratory,

Higher Institute of Industrial Management,

University of Sfax, Tunisia

* Correspondence: dhouib.matrix@gmail.com.

Abstract: This paper presents a new variant of Travelling Salesman Problem (TSP) and its first resolution. In literature there is not any research work that has presented the TSP under pentagonal fuzzy neutrosophic environment yet. TSP is a critical issue for manufacturing companies where all cities need to be visited only once except the starting city with a minimal cost. In real life, information provided (cost, time … etc.) are generally uncertain, indeterminate or inconsistent that's why in this paper parameters of the TSP are presented as neutrosophic pentagonal fuzzy numbers. To solve this problem, the novel heuristic Dhouib-Matrix-TSP1 (DM-TSP1) is applied using a ranking function in order to transform the fuzzy set to crisp data and range function to select cities. To prove the efficiency of the proposed DM-TSP1 in solving the new variant of TSP, we create novel benchmark instances. Then, a stepwise application of DM-TSP1 is illustrated in details.

Keywords: Fuzzy Optimization Techniques; Neutrosophic Applications; Travelling Salesman Problem; Operational Research; Combinatorial optimization; Dhouib-Matrix Optimization Methods, Dhouib-Matrix-TSP1.

1. Introduction

In real life, data of industrial companies are most of the time uncertain. That's why these data can be suitably presented by the neutrosophic concept in which the imprecision, the uncertainty and the indetermination are flexibly explored. This philosophy was firstly announced by Smarandache in [1] via three membership functions: Truth (T), Indeterminacy (I) and Falsity (F) with values belonging to]-0,1+[.

In literature, few research papers deal with combinatorial optimization under pentagonal neutrosophic number. In fact, Chakraborty *et al.* studied the Transportation Problem under single value pentagonal neutrosophic numbers for all parameters (supply, demand and transportation cost) in [2]. Also, Das and Chakraborty optimized the linear programming problem with pentagonal neutrosophic environment [3]. Radhika and Prakash considered the Assignment Problems with pentagonal neutrosophic number using a new magnitude ranking function for defuzzification [4]. Das unraveled the Transportation Problem where all parameters are presented using pentagonal neutrosophic numbers [5]. Then, Chakraborty hackled the networking problem in single valued pentagonal neutrosophic environment and introduced a new score function for defuzzification [6]. In addition, Chakraborty *et al.* studied the mobile communication system under pentagonal neutrosophic domain with multi-criteria group decision-making problem [7]. Besides, Chakraborty designed a job-sequencing model in pentagonal neutrosophic area [8]. Kane *et al.* involved the pentagonal and hexagonal fully fuzzy Transportation Problems [9].

To the best of our knowledge, in literature there is no research work which solved the Travelling Salesman Problem (TSP) under pentagonal fuzzy neutrosophic environment. In fact, many applications of TSP with five numbers of variable for each of the three components *T*, *I* and *F* can be found in real life such as the useful of pentagonal membership functions under multi objective environment. Also, the representation of verbal phrase with five different information and even the dynamic variation of the information with time can be presented by five membership functions.

From the above discussion carried on pentagonal neutrosophic problems, there are no current methods for solving pentagonal TSP under Neutrosophic condition. Thus, we generate new benchmark instances for pentagonal TSP with adapting the novel heuristic Dhouib-Matrix-TSP1 (DM-TSP1) to solve this problem. Correspondingly, this paper presents also the first application of DM-TSP1 on pentagonal domain.

Hence, this paper supports the theory and practical efficiency with several novel contributions which can be enumerated as follows:

- First resolution of TSP under pentagonal neutrosophic domain
- Introduce new instances for the pentagonal neutrosophic TSP
- Enhancing the novel heuristic DM-TSP1
- Step wise application of DM-TSP1

This paper is structured as follows: section 2 introduces one of the most important problems in operational research: the TSP. Section 3 presents the concept of the pentagonal neutrosophic environment. Section 4 presents the proposed novel heuristic DM-TSP1. Section 5 illustrates several numerical examples in order to clarify the application of optimization technique DM-TSP1. Finally, section 6 concludes the manuscript with the presentation of the future works.

2. The Travelling Salesman Problem

Materials and Methods should be described with sufficient details to allow others to replicate and build on published results. Please note that publication of your manuscript implicates that you must make all materials, data, computer code, and protocols associated with the publication available to readers. Please disclose at the submission stage any restrictions on the availability of materials or information. New methods and protocols should be described in detail while well-established methods can be briefly described and appropriately cited.

In real-world, the Travelling Salesman Problem (TSP) is extensively used. It deals with generating the shortest round between all nodes (cities, customers, suppliers, … etc.) namely the Hamiltonian cycle: each node is visited only once except the starting node which will be also the last visited node. The TSP is mathematically formulated as described in Equation 1.

Minimize:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij}x_{ij} \tag{1}$$

Subject to:

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1,...,n$$
$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1,...,n \qquad (2)$$
$$x_{ij} = 0 \; or \; 1, \; i = 1,...,n \; , \; \; j = 1,...,n$$

The binary variable x_{ij} is used to indicate either city i is connected to city j (then x_{ij} = 1) or city *i* and *j* are not connected (x_{ij} = 0). The parameter t represents the time (cost, distance, … etc.) while t_{ij} represents the time between city i and city j.

Panwar and Deep proposed the Grey Wolf metaheuristic for the symmetric TSP [10]. Gunduz and Aslan developed the stochastic Jaya algorithm in order to solve the TSP [11].

Mosayebi *et al.* generated a new type of hybrid TSP with Scheduling Problem in order to minimize the time of completion of the last job [12]. Wang and Han combined the Symbiotic Organisms Search with the Ant Colony Optimization algorithms to optimize the standard TSP [13]. Krishna *et al.* designed a new optimization method for the TSP namely the Spotted Hyena-based Rider Optimization by integrating the Rider Optimization with the Spotted Hyena Optimizer methods [14].

Çakir *et al.* integrated the Dijkstra algorithm with the Minimum Vertex Degree method in order to find the minimal transportation network [15]. Hu *et al.* designed a Bidirectional Graph Neural Network as an element of Deep Learning to solve the TSP [16]. Pandiri and Singh adapted the Artificial Bee Colony metaheuristic for the generalized covering TSP [17]. Luo *et al.* developed a Multi-start Tabu Search metaheuristic for Multi-visit TSP with Multi-drones [18]. Cavani *et al.* used the Branch-and-Cut technique for TSP with multiple drones for last-mile delivery [19]. Pereira *et al.* integrated the Branch-and-Cut with Valid Inequalities method for pickup and delivery TSP with multiple stacks [20]. Huerta *et al.* proposed a new spatial representation of nodes in TSP and used the Anytime Automatic Algorithm [21]. Hougardy *et al.* computed for the metric TSP an approximative ratio of the 2-Opt method [22]. Baniasadi *et al.* presented an application on two modern logistic problems with a description of how to transform the clustered generalized TSP to classical TSP [23]. Chen *et al.* introduced the Branch-and-Price algorithm for a multiple TSP [24].

3. Preliminaries

Several definitions and basic concepts are presented in this section in order to introduce the fuzzy and neutrosophic concepts.

Definition 1:

Let X be a space of points with its generic elements denoted by *x*. The neutrosophic set *N* has the form $N = \{\langle x : T_N(x), I_N(x), F_N(x) \rangle, x \in X\}$ where the functions T, I, F: $X \to]^{-}0,1^{+}[$ verifying the condition $^{-}0 \leq T_N(x) + I_N(x) + F_N(x) \leq 3^{+}$.

Definition 2:

Obviously the pentagonal neutrosophic number $N^N = \langle (t_1,t_2,t_3,t_4,t_5),(f_1, f_2, f_3, f_4, f_5),(i_1,i_2,i_3,i_4,i_5);(p,q,r) \rangle$, with $p,q,r \in [0,1]$, presents three membership functions: Truth (T), Indeterminacy (I) and Falsity (F) defined by:

$$T_N(x)=\begin{cases}0, & x\le t_1\\ \dfrac{p(x-t_1)}{t_2-t_1}, & t_1\le x\le t_2\\ \dfrac{p(x-t_2)}{t_3-t_2}, & t_2\le x\le t_3\\ 1, & x=t_3\\ \dfrac{p(t_4-x)}{t_4-t_3}, & t_3\le x\le t_4\\ \dfrac{p(t_5-x)}{t_5-t_4}, & t_4\le x\le t_5\\ 0, & x\ge t_5\end{cases}$$

$$I_N(x)=\begin{cases}1, & x\le i_1\\ \dfrac{i_2-x+q(x-i_1)}{i_2-i_1}, & i_1\le x\le i_2\\ \dfrac{i_3-x+q(x-i_2)}{i_3-i_2}, & i_2\le x\le i_3\\ q, & x=i_3\\ \dfrac{x-i_3+q(i_4-x)}{i_4-i_3}, & i_3\le x\le i_4\\ \dfrac{x-i_4+q(i_5-x)}{i_5-i_4}, & i_4\le x\le i_5\\ 1, & x\ge i_5\end{cases}$$

$$F_N(x)=\begin{cases}1, & x\le f_1\\ \dfrac{f_2-x+r(x-f_1)}{f_2-f_1}, & f_1\le x\le f_2\\ \dfrac{f_3-x+r(x-f_2)}{f_3-f_2}, & f_2\le x\le f_3\\ r, & x=f_3\\ \dfrac{x-f_3+r(f_4-x)}{f_4-f_3}, & f_3\le x\le f_4\\ \dfrac{x-f_4+r(f_5-x)}{f_5-f_4}, & f_4\le x\le f_5\\ 1, & x\ge f_5\end{cases}$$

Here is an example of a graphical representation (see Figure 1) for a pentagonal neutrosophic number $N^N=\langle(2,4,8,10,11),(1,2,5,9,11),(3,7,9,12,13);1,0,0\rangle$.

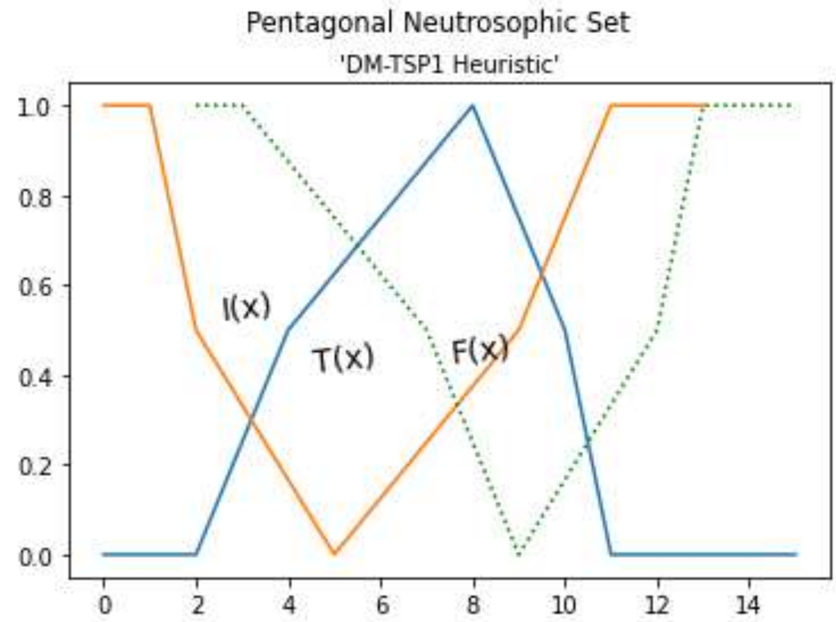

Figure 1. Graphical representation of the neutrosophic pentagonal fuzzy set.

Definition 3:

Let define $N^N = \langle (t_1, t_2, t_3, t_4, t_5); (f_1, f_2, f_3, f_4, f_5); (i_1, i_2, i_3, i_4, i_5) \rangle$ as a pentagonal neutrosophic number. From [2], the score and accuracy functions can be described as follows:

$$S(N^N) = \left(2 + \frac{t_1 + t_2 + t_3 + t_4 + t_5}{5} - \frac{i_1 + i_2 + i_3 + i_4 + i_5}{5} - \frac{f_1 + f_2 + f_3 + f_4 + f_5}{5}\right) \Big/ 3, \quad (3)$$

$$AC(N^N) = \left(\frac{t_1 + t_2 + t_3 + t_4 + t_5}{5} - \frac{f_1 + f_2 + f_3 + f_4 + f_5}{5}\right), \quad (4)$$

Definition 4:

To define the order between several pentagonal neutrosophic numbers, the score and accurate functions can be agreeably used. Let us assume two arbitrary pentagonal neutrosophic numbers N_a^N and N_b^N where:

$$N_a^N = \langle (t_{a1}, t_{a2}, t_{a3}, t_{a4}, t_{a5}), (i_{a1}, i_{a2}, i_{a3}, i_{a4}, i_{a5}), (f_{a1}, f_{a2}, f_{a3}, f_{a4}, f_{a5}) \rangle,$$

$$N_b^N = \langle (t_{b1}, t_{b2}, t_{b3}, t_{b4}, t_{b5}), (i_{b1}, i_{b2}, i_{b3}, i_{b4}, i_{b5}), (f_{b1}, f_{b2}, f_{b3}, f_{b4}, f_{b5}) \rangle.$$

The first step is to compute the score function for each number, so $S(N_a^N)$ and $S(N_b^N)$ verify:

1. if $S(N_a^N) > S(N_b^N)$, then $N_a^N > N_a^N$
2. if $S(N_a^N) < S(N_b^N)$, then $N_a^N < N_a^N$
3. if $S(N_a^N) = S(N_b^N)$, then:
    a. if $AC(N_a^N) > AC(N_b^N)$, then $N_a^N > N_a^N$
    b. if $AC(N_a^N) < AC(N_b^N)$, then $N_a^N < N_a^N$
    c. if $AC(N_a^N) = AC(N_b^N)$, then $N_a^N = N_a^N$

Here is a numerical example that illustrates the previous order definition between two pentagonal numbers:

$N_a^N = \langle (4,5,6,7,9),(1,3,5,6,7),(0,1,2,3,4) \rangle$ then $S(N_a^N) = 0.60$

$N_b^N = \langle (5,6,7,8,9),(0,1,2,3,4),(3,4,5,6,7) \rangle$ then $S(N_b^N) = 0.67$

if $S(N_a^N) < S(N_b^N)$ then $N_a^N < N_b^N$

4. The Novel Heuristic Dhouib-Matrix-TSP1 (DM-TSP1)

The novel deterministic heuristic Dhouib-matrix-TSP1 (DM-TSP1) was firstly designed by Dhouib in order to rapidly find an initial basic feasible solution for the TSP [25]. Then, it was followed by a stochastic version entitled Dhouib-Matrix-TSP2 in [26]. Also, an application of those two methods on automobile industry was presented in [27]. Furthermore, an application of these methods on TSP under uncertain environment was illustrated with triangular fuzzy numbers in [28], trapezoidal fuzzy numbers in [29] and octagonal fuzzy numbers in [30]. Moreover, the TSP was solved with DM-TSP1 under intuitionistic environment in [31] and with neutrosophic area in [32,33,34]. Besides, a new metaheuristic entitled Dhouib-Matrix-3 (DM3) was invented in [35] and a novel multi-start metaheuristic namely Dhouib-Matrix-4 (DM4) was introduced in [36].

The heuristic DM-TSP1 is composed of four steps (see Figure 2) where step 1 and step 4 are executed once. Nevertheless, step 2 and step 3 are repeated *n* iterations (*n* is the number of cities). DM-TSP1 is characterized by its flexibility to use different descriptive statistical metrics (Dhouib, 2021g). In this current research work, we will use the range (max-min) metric.

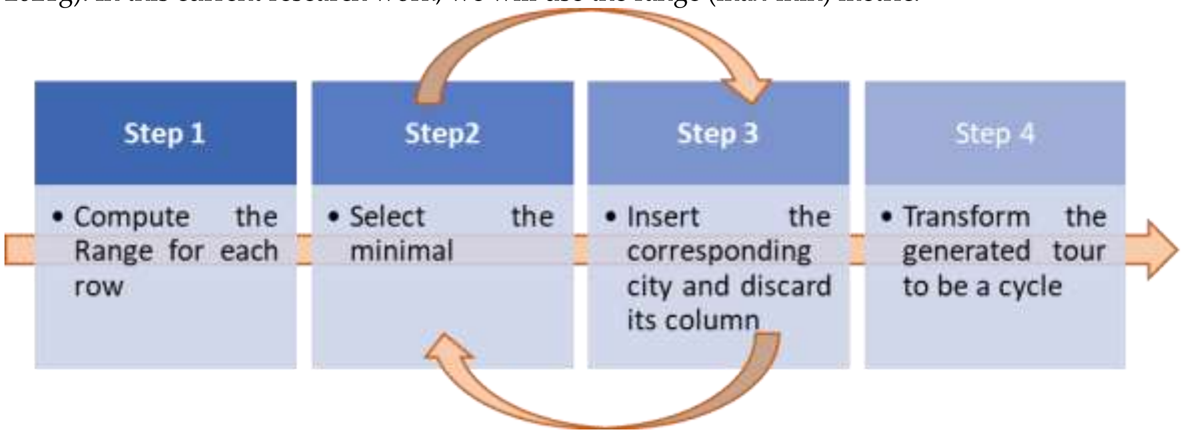

Figure 2. The flowchart of the proposed DM-TSP1.

In this paper, all parameters of the TSP are presented as pentagonal neutrosophic number. So, the function described in Equation 3 is used to convert these numbers into crisp numbers. Besides, the four steps can be started:

Step1: Compute the range function (max-min) for each row and write it on the right-hand side of the matrix. Next, find the minimal range and select its row. Then, select the smallest element in this row which will specify the two first cities *x* and *y* to be inserted in the list *List-cities {x, y}*. Finally, discard the respected columns of city *x* and city *y*.

Step 2: Find the minimal element for city *x* and for city *y* and select the smallest distance which will indicate city *z*.

Step 3: Add city *z* to the list *List-cities* and discard its column. Next, if there is no column go to step 4 otherwise go to step 2.

Step 4: Modify the realizable solution in *List-cities* in order to generate a cycle (the starting city in the cycle has to be also the last one). First, to ensure that the starting city will be at the first position, translate all the cities (one by one) before the starting one at the end of the list. Second, duplicate the starting city at the last position.

5. Application of DM-TSP1 heuristic in Neutrosophic Pentagonal Travelling Salesman Problem

This section will describe the stepwise application of the novel heuristic DM-TSP1 on pentagonal neutrosophic TSP. We generate two new instances because in literature there is not any work that solved this problem under pentagonal neutrosophic environment.

5.1. Illustration example 1

Let consider a travel salesman who needs to generate a Hamiltonian cycle between 5 cities namely A, B, C, D and E (each city is visited only once except the starting city which will also be the last visited city). The estimated time between all cities is presented as pentagonal neutrosophic number as presented in table 1.

Table 1. The pentagonal neutrosophic time between 5 cities.

	A	B	C	D	E
A	∞	<5,7,8,9,11; 2,3,4,5,6; 0,1,2,3,4>	<11,13,14,15,16; 2,4,6,8,10; 1,3,5,7,9>	<7,8,9,10,11; 2,3,4,5,6; 0,2,3,4,5>	<11,12,13,14,15; 3,4,6,8,9; 1,2,3,4,5>
B	<5,7,8,9,11; 2,3,4,5,6; 0,1,2,3,4>	∞	<10,11,12,13,14; 1,2,3,4,5; 6,7,8,9,10>	<8,9,10,11,13; 3,5,6,8,9; 1,2,4,5,6>	<5,7,8,9,14; 3,4,5,6,7; 0,1,2,3,4>
C	<11,13,14,15,16; 2,4,6,8,10; 1,3,5,7,9>	<10,11,12,13,14; 1,2,3,4,5; 6,7,8,9,10>	∞	<5,9,11,13,14; 4,6,7,8,9; 1,2,3,4,5>	<7,8,9,14,15; 0,1,2,3,4; 2,3,4,5,6>
D	<7,8,9,10,11; 2,3,4,5,6; 0,2,3,4,5>	<8,9,10,11,13; 3,5,6,8,9; 1,2,4,5,6>	<5,9,11,13,14; 4,6,7,8,9; 1,2,3,4,5>	∞	<8,9,14,15,16 1,2,3,5,7; 2,5,6,7,8>
E	<11,12,13,14,15; 3,4,6,8,9; 1,2,3,4,5>	<5,7,8,9,14; 3,4,5,6,7; 0,1,2,3,4>	<7,8,9,14,15; 0,1,2,3,4; 2,3,4,5,6>	<8,9,14,15,16 1,2,3,5,7; 2,5,6,7,8>	∞

At first, convert the pentagonal neutrosophic number to crisp number through to Equation (3). Figure 3 depicts the generated crisp time matrix.

$$\begin{pmatrix} \infty & 1.33 & 1.60 & 1.40 & 2.00 \\ 1.33 & \infty & 1.00 & 0.80 & 1.20 \\ 1.60 & 1.00 & \infty & 0.87 & 2.20 \\ 1.40 & 0.80 & 0.87 & \infty & 1.73 \\ 2.00 & 1.20 & 2.20 & 1.73 & \infty \end{pmatrix}$$

Figure 3. The crisp time matrix.

Now, DM-TSP1 can start by computing the range function (max-min) for each row (See figure 4).

$$\begin{pmatrix} \infty & 1.33 & 1.60 & 1.40 & 2.00 \\ 1.33 & \infty & 1.00 & \boxed{0.80} & 1.20 \\ 1.60 & 1.00 & \infty & 0.87 & 2.20 \\ 1.40 & 0.80 & 0.87 & \infty & 1.73 \\ 2.00 & 1.20 & 2.20 & 1.73 & \infty \end{pmatrix} \begin{matrix} 0.67 \\ (0.53) \\ 1.33 \\ 0.93 \\ 1.00 \end{matrix}$$

Figure 4. Compute the range (max-min) of each row.

Apparently, the minimal range is 0.53, so we look for the minimal element in the second row: it is 0.80 at position d_{24}. Thus, city 2 and city 4 are inserted in *List-cities* {2-4} and their corresponding columns are discarded (see figure 5).

$$\begin{pmatrix} \infty & 1.33 & 1.60 & 1.40 & 2.00 \\ 1.33 & \infty & 1.00 & 0.80 & 1.20 \\ 1.60 & 1.00 & \infty & 0.87 & 2.20 \\ 1.40 & 0.80 & 0.87 & \infty & 1.73 \\ 2.00 & 1.20 & 2.20 & 1.73 & \infty \end{pmatrix}$$

Figure 5. Discard columns 2 and 4.

Besides, find the smallest element between row 2 and row 4 which is 0.87 at position d_{43}. Then, insert city 3 at the last position (after city 4) in *List-cities* {2-4-3} and discarded its corresponding column (see Figure 6).

$$\begin{pmatrix} \infty & 1.33 & 1.60 & 1.40 & 2.00 \\ 1.33 & \infty & 1.00 & 0.80 & 1.20 \\ 1.60 & 1.00 & \infty & 0.87 & 2.20 \\ 1.40 & 0.80 & 0.87 & \infty & 1.73 \\ 2.00 & 1.20 & 2.20 & 1.73 & \infty \end{pmatrix}$$

Figure 6. Discard column 3.

Next, find the smallest element between row 2 and row 3 which is 1.20 at position d_{52}; then, insert city 5 before city 2 in *List-cities* {5-2-4-3} and discard its corresponding column (see Figure 7).

$$\begin{pmatrix} \infty & 1.33 & 1.60 & 1.40 & 2.00 \\ 1.33 & \infty & 1.00 & 0.80 & 1.20 \\ 1.60 & 1.00 & \infty & 0.87 & 2.20 \\ 1.40 & 0.80 & 0.87 & \infty & 1.73 \\ 2.00 & 1.20 & 2.20 & 1.73 & \infty \end{pmatrix}$$

Figure 7. Discard column 5.

Find the smallest element between row 3 and row 5 which is 1.60 at position d_{13}, insert city 1 after city 3 in *List-cities* {5-2-4-3-1} and discard its corresponding column (see Figure 8).

$$\begin{pmatrix} \infty & 1.33 & 1.60 & 1.40 & 2.00 \\ 1.33 & \infty & 1.00 & 0.80 & 1.20 \\ 1.60 & 1.00 & \infty & 0.87 & 2.20 \\ 1.40 & 0.80 & 0.87 & \infty & 1.73 \\ 2.00 & 1.20 & 2.20 & 1.73 & \infty \end{pmatrix}$$

Figure 8. Discard column 1.

Obviously, all columns are discarded and a tour is generated. The final step is to generate a cycle starting and ending by city 1. So, translate city by city, from the left to the right until city 1 will become at the first position: {1-5-2-4-3}. Finally, add city one at the last position in order to generate a cycle: {1-5-2-4-3-1}. Thus, the optimal solution generated using DM-TSP1 is: $x_{24} = 1;\ x_{43} = 1;\ x_{52} = 1;\ x_{31} = 1;\ x_{15} = 1$. With a total crisp cost $z = 0.80 + 0.87 + 1.20 + 1.60 + 2.00 = 6.47$. Consequently, the minimal pentagonal neutrosophic cost is: $z^N = < 40, 50, 56, 62, 72; 15, 23, 30, 38, 44; 4, 10, 17, 23, 29 >$. The graphical representation of the optimal solution obtained by DM-TSP1 is depicted in Figure 9.

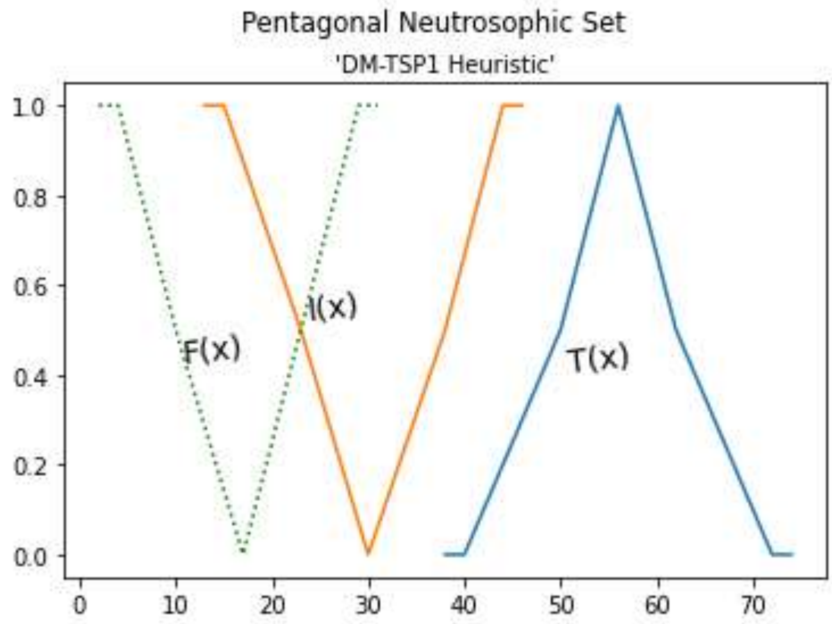

Figure 9. Graphical representation of the optimal neutrosophic solution.

Thus, the decision maker can deduce the minimal pentagonal neutrosophic cost with its truth, indeterminacy and falsity degrees. The truth membership function for the generated solution is denoted by Equation 5.

$$T_N(x) = \begin{cases} 0, \ x \le 40 \\ \dfrac{x-40}{50-40}, 40 \le x \le 50 \\ \dfrac{x-50}{56-50}, 50 \le x \le 56 \\ 1, x = 56 \\ \dfrac{62-x}{62-56}, 56 \le x \le 62 \\ \dfrac{72-x}{72-62}, 62 \le x \le 72 \\ 0, \ x \ge 72 \end{cases} \tag{5}$$

Similarly, the indeterminacy membership function is presented by Equation 6.

$$I_N(x) = \begin{cases} 1, \ x \le 15 \\ \dfrac{23-x}{23-15}, 15 \le x \le 23 \\ \dfrac{30-x}{30-23}, 23 \le x \le 30 \\ 0, x = 30 \\ \dfrac{x-30}{38-30}, 30 \le x \le 38 \\ \dfrac{x-38}{44-38}, 38 \le x \le 44 \\ 1, \ x \ge 44 \end{cases} \tag{6}$$

Also, the falsity membership function is presented by Equation 7.

$$F_N(x) = \begin{cases} 1, \ x \le 4 \\ \dfrac{10-x}{10-4}, 4 \le x \le 10 \\ \dfrac{17-x}{17-10}, 10 \le x \le 17 \\ r, x = 17 \\ \dfrac{x-17}{23-17}, 17 \le x \le 23 \\ \dfrac{x-23}{29-23}, 23 \le x \le 29 \\ 1, \ x \ge 29 \end{cases} \quad (7)$$

3.2. Illustration example 2

Let consider a second numerical example for a travel salesman man who needs to generate a Hamiltonian cycle between 6 cities namely A, B, C, D, E and F (see Table 2).

Table 2. The pentagonal neutrosophic time between 6 cities

	A	B	C	D	E	F
A	∞	<10,11,12,13,14; 4,5,6,7,9; 2,5,6,8,12>	<19,21,24,27,30; 8,9,14,17,18; 6,7,8,9,10>	<4,5,6,7,8; 0,1,2,3,4; 0,1,1,2,3>	<15,22,23,26,29; 5,7,9,11,15; 5,6,8,9,15>	<7,11,12,16,17; 2,4,5,6,7; 1,5,8,9,11>
B	<10,11,12,13,14; 4,5,6,7,9; 2,5,6,8,12>	∞	<9,10,15,18,19; 5,6,7,8,9; 1,2,6,8,12>	<6,13,14,15,16; 4,5,8,9,11; 0,3,5,6,7>	<1,6,11,16,21; 3,4,5,6,7; 1,2,3,4,5>	<9,16,17,18,25; 3,4,6,11,12; 2,3,4,5,6>
C	<19,21,24,27,30; 8,9,14,17,18; 6,7,8,9,10>	<9,10,15,18,19; 5,6,7,8,9; 1,2,6,8,12>	∞	<9,10,15,18,19; 5,6,7,8,9; 1,2,6,8,12>	<5,7,14,15,16; 5,6,7,8,9; 0,1,2,3,4>	<13,16,17,21,27; 7,9,10,11,12; 2,5,6,7,8>
D	<4,5,6,7,8; 0,1,2,3,4; 0,1,1,2,3>	<6,13,14,15,16; 4,5,8,9,11; 0,3,5,6,7>	<9,10,15,18,19; 5,6,7,8,9; 1,2,6,8,12>	∞	<11,13,15,24,25; 3,4,6,13,14; 2,5,8,11,14>	<14,17,21,22,23; 3,4,6,11,21; 1,3,7,9,16>
E	<15,22,23,26,29; 5,7,9,11,15; 5,6,8,9,15>	<1,6,11,16,21; 3,4,5,6,7; 1,2,3,4,5>	<5,7,14,15,16; 5,6,7,8,9; 0,1,2,3,4>	<11,13,15,24,25; 3,4,6,13,14; 2,5,8,11,14>	∞	<8,9,11,17,19; 2,3,4,10,11; 1,3,4,5,7>
F	<7,11,12,16,17; 2,4,5,6,7; 1,5,8,9,11>	<9,16,17,18,25; 3,4,6,11,12; 2,3,4,5,6>	<13,16,17,21,27; 7,9,10,11,12; 2,5,6,7,8>	<14,17,21,22,23; 3,4,6,11,21; 1,3,7,9,16>	<8,9,11,17,19; 2,3,4,10,11; 1,3,4,5,7>	∞

At first convert the pentagonal neutrosophic number to crisp number using Equation (3). The generated crisp matrix is presented in Figure 10.

$$\begin{pmatrix} \infty & 0.40 & 1.67 & 1.53 & 2.33 & 1.00 \\ 0.40 & \infty & 1.13 & 1.07 & 1.67 & 2.60 \\ 1.67 & 1.13 & \infty & 2.00 & 1.47 & 1.80 \\ 1.53 & 1.07 & 2.00 & \infty & 1.20 & 1.73 \\ 2.33 & 1.67 & 1.47 & 1.20 & \infty & 1.60 \\ 1.00 & 2.60 & 1.80 & 1.73 & 1.60 & \infty \end{pmatrix}$$

Figure 10. The crisp time matrix.

Next, compute the range function (max-min) for each row (see figure 11).

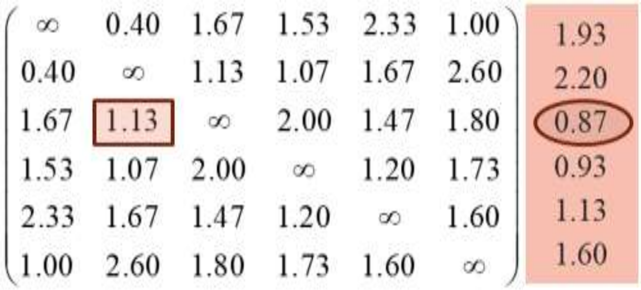

Figure 11. Compute the range of each row.

The minimal range is 0.87, so we look for the minimal element in the third row: it is 1.13 at position d_{32}. Thus, city 3 and city 2 are inserted in *List-cities* {3-2} and their corresponding columns are discarded (see Figure 12).

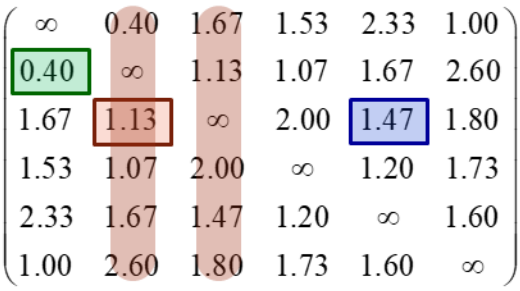

Figure 12. Discard columns 2 and 3.

Besides, find the smallest element between row 3 and row 2 which is 0.40 at position d_{21}. Then, insert city 1 at the last position (after city 2) in *List-cities* {3-2-1} and discarded its corresponding column (see Figure 13).

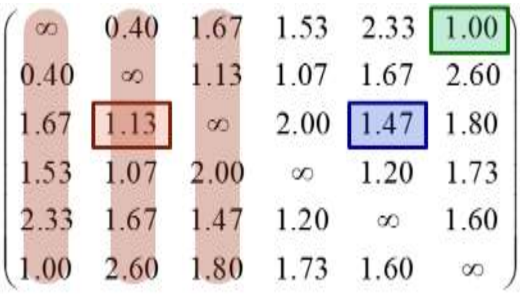

Figure 13. Discard column 1.

Next, find the smallest element between row 3 and row 1 which is 1.00 at position d_{16}; then, insert city 6 after city 1 in *List-cities* {3-2-1-6} and discard its corresponding column (see Figure 4).

$$\begin{pmatrix} \infty & 0.40 & 1.67 & 1.53 & 2.33 & 1.00 \\ 0.40 & \infty & 1.13 & 1.07 & 1.67 & 2.60 \\ 1.67 & 1.13 & \infty & 2.00 & 1.47 & 1.80 \\ 1.53 & 1.07 & 2.00 & \infty & 1.20 & 1.73 \\ 2.33 & 1.67 & 1.47 & 1.20 & \infty & 1.60 \\ 1.00 & 2.60 & 1.80 & 1.73 & 1.60 & \infty \end{pmatrix}$$

Figure 14. Discard column 6.

Succeeding, find the smallest element between rows 3 and 6 which is 1.47 at position d_{35}; then, insert city 5 before city 3 in *List-cities* {5-3-2-1-6} and discard its corresponding column (see Figure 15).

$$\begin{pmatrix} \infty & 0.40 & 1.67 & 1.53 & 2.33 & 1.00 \\ 0.40 & \infty & 1.13 & 1.07 & 1.67 & 2.60 \\ 1.67 & 1.13 & \infty & 2.00 & 1.47 & 1.80 \\ 1.53 & 1.07 & 2.00 & \infty & 1.20 & 1.73 \\ 2.33 & 1.67 & 1.47 & 1.20 & \infty & 1.60 \\ 1.00 & 2.60 & 1.80 & 1.73 & 1.60 & \infty \end{pmatrix}$$

Figure 15. Discard column 5.

Subsequent, select the smallest element between rows 5 and row 6 which is 1.20 at position d_{54}, insert city 4 before city 5 in *List-cities* {4-5-3-2-1-6} and discard its corresponding column (see Figure 16).

$$\begin{pmatrix} \infty & 0.40 & 1.67 & 1.53 & 2.33 & 1.00 \\ 0.40 & \infty & 1.13 & 1.07 & 1.67 & 2.60 \\ 1.67 & 1.13 & \infty & 2.00 & 1.47 & 1.80 \\ 1.53 & 1.07 & 2.00 & \infty & 1.20 & 1.73 \\ 2.33 & 1.67 & 1.47 & 1.20 & \infty & 1.60 \\ 1.00 & 2.60 & 1.80 & 1.73 & 1.60 & \infty \end{pmatrix}$$

Figure 16. Discard column 4.

The final step is to generate a cycle starting and ending by city 1. So, translate city by city, from the left to the right until city 1 will become at the first position: {1-6-4-5-3-2}. Finally, add city one at the last position in order to generate a cycle: {1-6-4-5-3-2-1}.

Thus, the optimal solution generated using DM-TSP1 is: $x_{32}=1;\ x_{21}=1;\ x_{16}=1;\ x_{53}=1;\ x_{45}=1; x_{64}=1$. With a total crisp cost $z=1.13+0.40+1.00+1.47+1.20+1.73=6.93$. Consequently, the minimal pentagonal neutrosophic cost is: $z^N =< 56,69,89,108,114;22,29,37,53,69;7,21,37,48,69 >$. The graphical representation of the optimal solution obtained by DM-TSP1 is depicted in Figure 17.

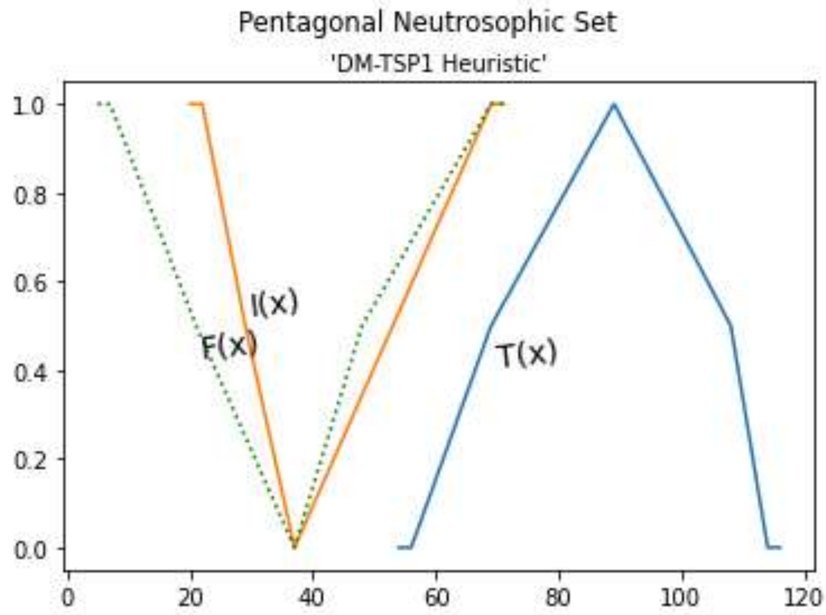

Figure 17. Graphical representation of the optimal neutrosophic solution.

All examples presented in this section are presented as type-1 neutrosophic number. For next research, the proposed constructive heuristic DM-TSP1 can be as well employed to optimize TSP under type-2 neutrosophic number which is an advancement of neutrosophic number presented in [37].

5. Conclusions

The neutrosophic concept is a new philosophy and representing the Travelling Salesman Problem (TSP) under pentagonal neutrosophic environment has not been earlier considered by any other author in the literature. Consequently, in this paper we describe the first resolution of the TSP with pentagonal neutrosophic number using the novel heuristic Dhouib-Matrix-TSP1 (DM-TSP1). Viewing that there is no instance for this problem; so, we generate novel instances. The method DM-TSP1 has been demonstrated by a suitable numerical example. Furthermore, DM-TSP1 easily generates the optimal Hamiltonian cycle after only n iterations, where n is the number of cities. As future work, the DM-TSP1 will be improvised for optimizing the TSP under type-2 neutrosophic domain.

Funding: This research received no external funding.

References

1. Smarandache, F. A Unifying Field in Logics: Neutrosophic Logic. Neutrosophy, Neutrosophic Set, Neutrosophic Probability and Statistics. *Infinite Study* **2005**, DOI: 10.5281/zenodo.8792.

2. Chakraborty, A.; Broumi, S.; Singh, P.K. Some properties of Pentagonal Neutrosophic Numbers and its Applications in Transportation Problem Environment. *Neutrosophic Sets and Systems* **2019**, Volume 28, 200-215, DOI: 10.5281/zenodo.3382542.

3. Das, S.K.; Chakraborty, A. A new approach to evaluate linear programming problem in pentagonal neutrosophic environment. *Complex & Intelligent Systems* **2021**, Volume 7, 101–110, DOI: 10.1007/s40747-020-00181-0.

4. Radhika, K.; Prakash, K.A. Ranking of Pentagonal Neutrosophic Numbers and its Applications to solve Assignment Problem. *Neutrosophic Sets and Systems* **2020**, Volume 35, DOI: 10.5281/zenodo.3951704.

5. Das, S.K. Application of Transportation Problem under Pentagonal Neutrosophic Environment. *Journal of Fuzzy Extension & Applications* **2020**, Volume 1, 27–41, DOI: 10.22105/jfea.2020.246633.1001.

6. Chakraborty, A. A New Score Function of Pentagonal Neutrosophic Number and its Application in Networking Problem. *International Journal of Neutrosophic Science* **2020**, Volume 1, No. 1, 40-51.

7. Chakraborty, A.; Banik, B.; Monda, S.P.; Alam, S. Arithmetic and Geometric Operators of Pentagonal Neutrosophic Number and its Application in Mobile Communication Service Based MCGDM Problem. *Neutrosophic Sets and Systems* **2020**, Volume 32, 61-79.

8. Chakraborty, A. A job-sequencing problem using modified score function in PNN environment. *International Journal of Hybrid Intelligence*, **2021**, Volume 2, No. 1, 67-78, DOI: 10.1504/IJHI.2021.117818.

9. Kane, L.; Diakite, M.; Kane, S.; Bado, H.; Diawara, D. Fully Fuzzy Transportation Problems with Pentagonal and Hexagonal Fuzzy Numbers. *Journal of Applied Research on Industrial Engineering* **2021**, Volume 8, No. 3, pp. 251–269, DOI:10.22105/jarie.2021.288186.1331.

10. Panwar, K.; Deep, K. Discrete Grey Wolf Optimizer for symmetric travelling salesman problem. *Applied Soft Computing* **2021**, Volume 105, Paper ID: 107298, DOI: 10.1016/j.asoc.2021.107298.

11. Gunduz, M.; Aslan, M. DJAYA: A discrete Jaya algorithm for solving traveling salesman problem. *Applied Soft Computing* **2021**, Volume 105, Paper ID: 107275, DOI: 10.1016/j.asoc.2021.107275.

12. Mosayebi, M.; Sodhi, M.; Wettergren, T.A. The Traveling Salesman Problem with Job-times (TSPJ). *Computers & Operations Research* **2021**, Volume 129, Paper ID: 105226, DOI: 10.1016/j.cor.2021.105226.

13. Wang, Y.; Han, Z. Ant colony optimization for traveling salesman problem based on parameters optimization. *Applied Soft Computing* **2021**, Volume 107, Paper ID: 107439, DOI: 10.1016/j.asoc.2021.107439.

14. Krishna, M.M.; Panda, N.; Majhi, S.K. Solving traveling salesman problem using hybridization of rider optimization and spotted hyena optimization algorithm. *Expert Systems with Applications* **2021**, Volume 183, Paper ID: 115353, DOI: 10.1016/j.eswa.2021.115353.

15. Çakir, E.; Ulukan, Z.; Acarman, T. Shortest Fuzzy Hamiltonian Cycle on Transportation Network Using Minimum Vertex Degree and Time-dependent Dijkstra's Algorithm. *IFAC-PapersOnLine* **2021**, Volume 54, Issue 2, 348-353, DOI: 10.1016/j.ifacol.2021.06.048.

16. Hu, Y.; Zhang, Z.; Yao, Y.; Huyan, X.; Zhou, X.; Lee, W.S. A bidirectional graph neural network for traveling salesman problems on arbitrary symmetric graphs. *Engineering Applications of Artificial Intelligence* **2021**, Volume 97 Paper ID: 104061, DOI: 10.1016/j.engappai.2020.104061.

17. Pandiri, V.; Singh, A. An artificial bee colony algorithm with variable degree of perturbation for the generalized covering traveling salesman problem. *Applied Soft Computing* **2019**, Volume 78, 481-495, DOI: 10.1016/j.asoc.2019.03.001.

18. Luo, Z.; Poon, M.; Zhang, Z.; Liu, Z.; Lim, A. The Multi-visit Traveling Salesman Problem with Multi-Drones. *Transportation Research Part C: Emerging Technologies* **2021**, Volume 128, Paper ID: 103172, DOI: 10.1016/j.trc.2021.103172.

19. Cavani, S.; Iori, M.; Roberti, R. Exact methods for the traveling salesman problem with multiple drones. *Transportation Research Part C: Emerging Technologies* **2021**, Volume 130, Paper ID: 103280, DOI: 10.1016/j.trc.2021.103280.

20. Pereira, A.H.; Mateus, G.R.; Urrutia, S.A. Valid inequalities and branch-and-cut algorithm for the pickup and delivery traveling salesman problem with multiple stacks. *European Journal of Operational Research* **2021**, DOI: 10.1016/j.ejor.2021.07.051.

21. Huerta, I. I.; Neira, D.A.; Ortega, D.A.; Varas, V.; Godoy, J.; Asín-Achá, R. Improving the state-of-the-art in the Traveling Salesman Problem: An Anytime Automatic Algorithm Selection. *Expert Systems with Applications* **2022**, Volume 187, Paper ID: 115948, DOI: 10.1016/j.eswa.2021.115948.

22. Hougardy, S.; Zaiser, F.; Zhong, X. The approximation ratio of the 2-Opt Heuristic for the metric Traveling Salesman Problem. *Operations Research Letters* **2020**, Volume 48, Issue 4, 401-404, DOI: 10.1016/j.orl.2020.05.007.

23. Baniasadi, P.; Foumani, M.; Smith-Miles, K.; Ejov, V. A transformation technique for the clustered generalized traveling salesman problem with applications to logistics. *European Journal of Operational Research* **2020**, Volume 285, Issue 2, 444-457, DOI: 10.1016/j.ejor.2020.01.053.

24. Chen, C.; Lee, C.; Chen A. A Building Information Model enabled Multiple Traveling Salesman Problem for building interior patrols. *Advanced Engineering Informatics* **2021**, Volume 47, Article ID: 101237, DOI: 10.1016/j.aei.2020.101237.

25. Dhouib, S. A New Column-Row Method for Traveling Salesman Problem: The Dhouib-Matrix-TSP1. *International Journal of Recent Engineering Science* **2021**, Volume 8, Issue 1, 6-10, 10.14445/23497157/IJRES-V8I1P102.

26. Dhouib, S. Stochastic Column-Row Method for Travelling Salesman Problem: The Dhouib-Matrix-TSP2. *International Journal of Engineering Research & Technology* **2021**, Volume 10, Issue 3, 524-527.

27. Dhouib, S. Minimizing the Total Distance for the Supply Chain Problem Using Dhouib-Matrix-TSP2 Method. *International Journal of Advanced Research in Engineering and Technology* **2021**, Volume 12, Issue 5, 1-12, DOI: 10.34218/IJARET.12.5.2021.001.

28. Dhouib, Sa.; Dhouib, S. Optimizing the Trapezoidal Fuzzy Travelling Salesman Problem Through Dhouib-Matrix-TSP1 Method Based on Magnitude Technique. *International Journal of Scientific Research in Mathematical and Statistical Sciences* **2021**, Volume 8, Issue 2, 1-4, DOI: 10.26438/ijsrmss/v8i2.14.

29. Miledi, M.; Dhouib S.; Loukil T. Dhouib-Matrix-TSP1 Method to Optimize Octagonal Fuzzy Travelling Salesman Problem Using α-Cut Technique. *International Journal of Computer and Information Technology* **2021**, Volume 10, Issue 3, 130-133, DOI: 10.24203/ijcit.v10i3.105.

30. Dhouib, S. Haar Dhouib-Matrix-TSP1 Method to Solve Triangular Fuzzy Travelling Salesman Problem. *Research Journal of Recent Sciences* **2021**, Volume 10, Issue 3, 18-20.

31. Dhouib, S. Novel Heuristic for Intuitionistic Triangular Fuzzy Travelling Salesman Problem. *International Journal of Applied Evolutionary Computation* **2021**, Volume 12, Issue 4, In Press.

32. Dhouib, S. Neutrosophic Triangular Fuzzy Travelling Salesman Problem Based on Dhouib-Matrix-TSP1 Heuristic. *International Journal of Computer and Information Technology* **2021**, Volume 10, Issue 5, 180-183, DOI: 10.24203/ijcit.v10i5.154.

33. Dhouib, S. Optimization of Travelling Salesman Problem on Single Valued Triangular Neutrosophic Number using Dhouib-Matrix-TSP1 Heuristic. *International Journal of Engineering* **2021**, Volume 34, Issue 12, pp. 2642-2647, DOI: 10.5829/IJE.2021.34.12C.09.

34. Dhouib, S.; Broumi, S.; Lathamaheswari, M. Single Valued Trapezoidal Neutrosophic Travelling Salesman Problem with Novel Greedy Method: The Dhouib-Matrix-TSP1 (DM-TSP1). *International Journal of Neutrosophic Science* **2021**, Volume 17, Issue 2, 144 - 157, DOI: 10.54216/IJNS.170205.

35. Dhouib S. Novel Metaheuristic Based on Iterated Constructive Stochastic Heuristic: Dhouib-Matrix-3 (DM3). *Applied Computational Intelligence and Soft Computing* **2021**, Volume 2021, Article ID 7761993, 1-10, DOI: 10.1155/2021/7761993.

36. Dhouib, S. Multi-Start Constructive Heuristic Through Descriptive Statistical Metrics: The Dhouib-Matrix-4 (DM4) Metaheuristic. *International Journal of Operational Research* **2022**, DOI: 10.1504/IJOR.2021.10045069, In Press.

37. Abdel-Basset, M.; Saleh M.; Gamal A. and Smarandache F. An approach of TOPSIS technique for developing supplier selection with group decision making under type-2 neutrosophic number. *Applied Soft Computing* **2019**, Volume 77, 438-452, DOI: 10.1016/j.asoc.2019.01.035.

Received: July 18, 2022. Accepted: September 23, 2022.