International Journal of Advanced Research in Engineering and Technology (IJARET)
Volume 12, Issue 5, May 2021, pp.1-12 Article ID: IJARET_12_05_001
Available online at https://iaeme.com/Home/issue/IJARET?Volume=12&Issue=5
ISSN Print: 0976-6480 and ISSN Online: 0976-6499
DOI: 10.34218/IJARET.12.5.2021.001

MINIMIZING THE TOTAL DISTANCE FOR THE SUPPLY CHAIN PROBLEM USING DHOUIB-MATRIX-TSP2 METHOD

Souhail Dhouib
Professor, Higher Institute of Industrial Management,
University of Sfax, Tunisia.

ABSTRACT

In this paper, we are looking for minimizing the total distance for supply chain management problems through the application of our new column-row methods: the deterministic Dhouib-Matrix-TSP1 and the stochastic Dhouib-Matrix-TSP2. Although these methods are composed of four simple steps, they find quickly the optimal or near optimal solution. Note that the Dhouib-Matrix-TSP1 is characterized by its rapidity, whereas, the Dhouib-Matrix-TSP2 is distinguish by its quality.

Stepwise application of the newest algorithm Dhouib-Matrix-TSP1 on the Milk-Run Delivery problem, which looks for finding the shortest route like in the well-known Travelling Salesman Problem, demonstrates the rapidity of the proposed method. Furthermore, the solution found is very competitive: deviation of 1.19% from the optimal solution. By enriching this method Dhouib-Matrix-TSP1 with a stochastic process, the Dhouib-Matrix-TSP2, the optimal solution is easily found. In addition, some instances from the literature are used to prove the efficiency of our methods.

Key words: Automobile Industry, Optimizing, Milk-Run delivery, Supply Chain Management, Travelling Salesman Problem, Dhouib-Matrix-TSP1.

Cite this Article: Souhail Dhouib, Minimizing the Total Distance for the Supply Chain Problem Using Dhouib-Matrix-TSP2 Method, *International Journal of Advanced Research in Engineering and Technology (IJARET)*, 12(5), 2021, pp. 1-12.
https://iaeme.com/Home/issue/IJARET?Volume=12&Issue=5

1. INTRODUCTION

In the automobile industry, the concept of Milk-Run delivery is widely used. The technique of Milk-run comes from the traditional milk sales system, small quantity deliveries with high frequency, which is based on the delivery of bottles full of milk and the collection of empty ones on the road. This method has become one of the most successful models. [1] generates a declarative model with interactive optimization and iterative formulation. [2] develops a two-layers heuristic search algorithm in order to minimize the total distance and the number of used vehicles.

Additionally, this technique is proposed in [3] for delivery vehicle scheduling modeling and optimization for automobile mixed milk-run mode involved indirect suppliers. [4] designs a simulated annealing metaheuristic to solve the Milk-Run routing problem. [5] presents an application of the Milk-Run concept to the express delivery industry.

Furthermore, the Milk-Run delivery is considered in [6] as tools for lean management and lean production environment. In [7], a genetic algorithm is used to solve the routing problem of automotive supply logistics. [8] presents a literature review about the Milk-Run logistics. [9] uses the tabu search metaheuristic in order to optimize the common frequency routing problem in automobile parts supply.

Technically, the Milk-Run delivery technique looks for minimizing the Supply Chain Total Cost (SCTC), which can be calculated as depicted in equation 1.

$$SCTC = Cost(Manufacturing + Delivery + Inventory) \quad (1)$$

Where the delivery cost is based on route distance *dr*, which needs to be optimized. Theoretically, the *dr* in the Milk-Run delivery problem has the same characteristics of finding the shortest route in the well-known Travelling Salesman Problem (TSP).

The TSP problem deals with Hamiltonian cycle: finding the shortest route in *n* cities, where the last visited city is the starting one and each city is visited exactly once. The TSP is mathematically formulated as follows, where d_{ij} denotes the distance between city *i* and city *j* and x_{ij} presents a binary variable (x_{ij} = 1 if city *i* is connected to city *j* otherwise x_{ij} = 0):

Minimize:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}x_{ij} \quad (2)$$

Subject to:

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1,...,n$$

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1,...,n$$

$$x_{ij} = 0 \text{ or } 1, \ i = 1,...,n, \ \ j = 1,...,n \quad (3)$$

In the next section, we will present the proposed methods: the Dhouib-Matrix-TSP1 and the Dhouib-Matrix-TSP2. Then, in section 3, we will show through a numerical example the stepwise algorithm of the Dhouib-Matrix-TSP1 method applied in the case study of Milk-Run delivery for automobile industry. Next, in section 4, we will present a comparative study of this problem and several other problems from the literature. Section 5, will give a conclusion.

2. THE PROPOSED METHODS

This section will present our two methods: The Dhouib-Matrix-TSP1 which can find a good solution in very reduced number of iterations (just *n* iterations, where *n* is the number of cities) while the stochastic version Dhouib-Matrix-TSP2 which can easily find the optimal or near solution.

2.1 The Deterministic Method: Dhouib-Matrix-TSP1

In [10], we introduced the new algorithm of Dhouib-Matrix-TSP1 for the travelling salesman problem as an approximation method that generates an optimal or near optimal solution in just *n* iterations.

In fact, our method Dhouib-Matrix-TSP1 is composed of 4 phases (see Figure 1). In addition, in this paper, we will enhance the first phase by the standard deviation metric instead of the minimum metric as presented in [10].

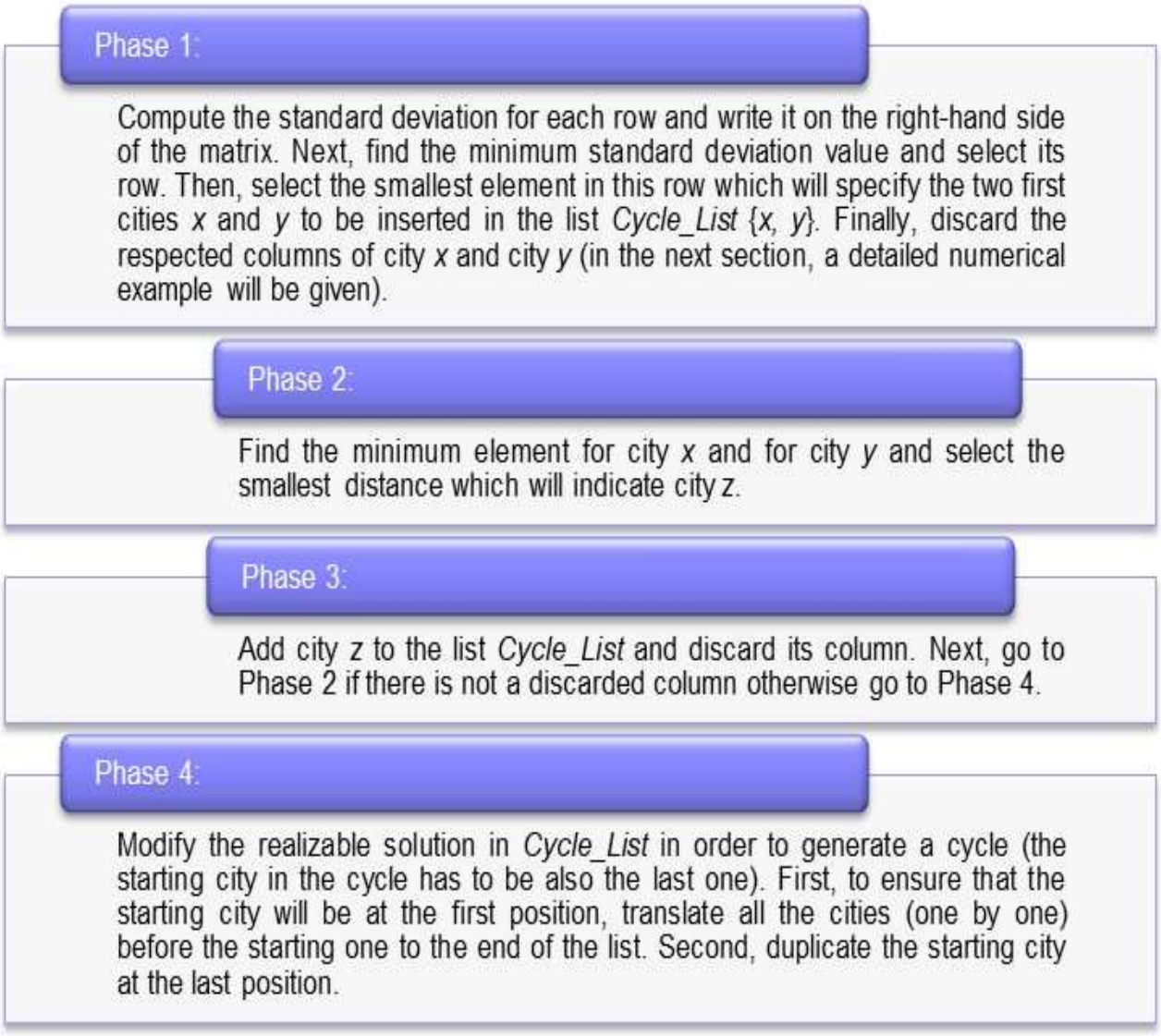


Figure 1 The Dhouib-Matrix-TSP1 Method

2.2 The Stochastic Method: The Dhouib-Matrix-TSP2

In our last research work [11], we have designed a stochastic version named Dhouib-Matrix-TSP2: instead of selecting the smallest element in a row, we select one of the *k* smallest elements (where *k* represents, the near neighbor nodes) in the row. The pseudo-code of Dhouib-Matrix-TSP2 method is presented in Figure 2.

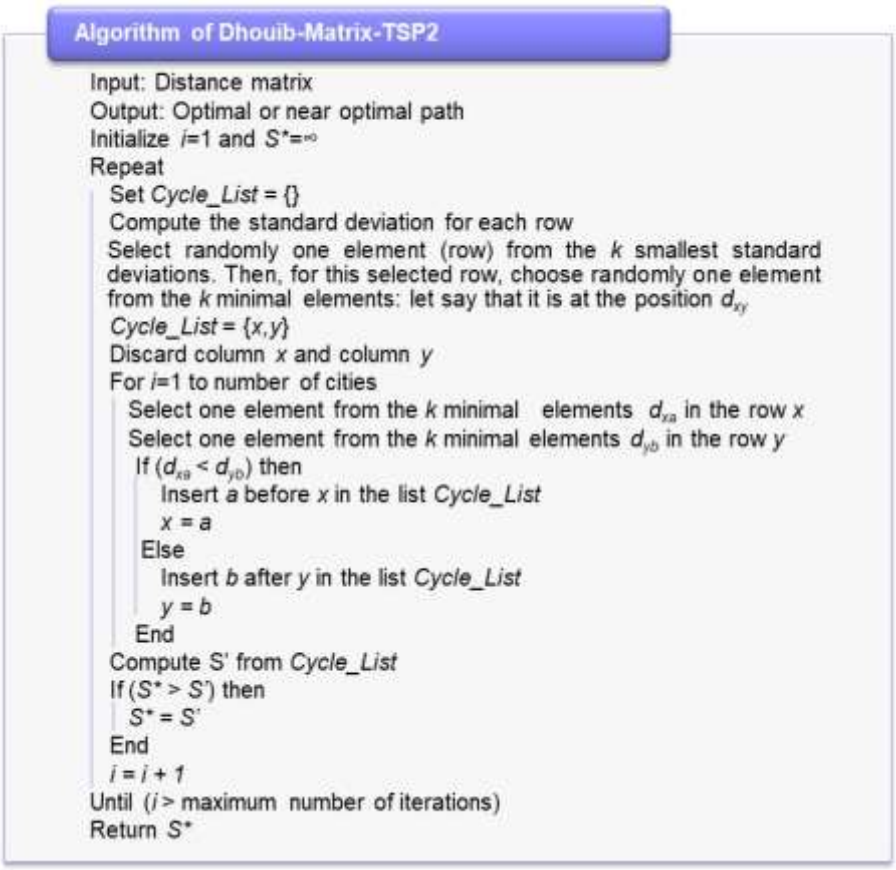


Figure 2 Dhouib-Matrix-TSP2 algorithme

3. RESULTS AND DISCUSSION

3.1 Numerical Example

This section presents a stepwise algorithm for a particular case of automobile industry presented in [12]. Table 1 presents the transportation distance matrix of nine nodes, one manufacturer, three suppliers and five customers. In the search process, our method Dhouib-Matrix-TSP1 will simply and quickly find a near optimal path in just nine iterations.

Table 1 Distance matrix of nine nodes

From / To	R0	S1	S2	S3	C1	C2	C3	C4	C5
R0	0	34.66	24.19	55.04	14.14	62.39	12.65	45.00	31.38
S1	34.66	0	58.41	84.12	20.52	95.38	31.38	60.03	13.93
S2	24.19	58.41	0	35.44	38.01	38.60	29.41	53.67	52.63
S3	55.04	84.12	35.44	0	66.10	21.21	52.89	88.77	73.82
C1	14.14	20.52	38.01	66.10	0	75.58	14.14	49.04	19.10
C2	62.39	95.38	38.60	21.21	75.58	0	64.35	85.99	87.20
C3	12.65	31.38	29.41	52.89	14.14	64.35	0	57.14	23.35
C4	45.00	60.03	53.67	88.77	49.04	85.99	57.14	0	66.71
C5	31.38	13.93	52.63	73.82	19.10	87.20	23.35	66.71	0

The first step is to compute the standard deviation for each row (red column in Figure 3). Then, select the smallest element which is 16.92 in row 3.

0	34.66	24.19	55.04	14.14	62.39	12.65	45.00	31.38	19.44
34.66	0	58.41	84.12	20.52	95.38	31.38	60.03	13.93	30.46
24.19	58.41	0	35.44	38.01	38.60	29.41	53.67	52.63	16.92
55.04	84.12	35.44	0	66.10	21.21	52.89	88.77	73.82	27.86
14.14	20.52	38.01	66.10	0	75.58	14.14	49.04	19.10	24.36
62.39	95.38	38.60	21.21	75.58	0	64.35	85.99	87.20	30.71
12.65	31.38	29.41	52.89	14.14	64.35	0	57.14	23.35	20.85
45.00	60.03	53.67	88.77	49.04	85.99	57.14	0	66.71	24.53
31.38	13.93	52.63	73.82	19.10	87.20	23.35	66.71	0	28.49

Figure 3 Compute the standard deviation for each row

Iteration number 1: Find the minimal element in row 3 (which is 24.19) in the position of d_{31}. Thus, insert in the respected order the node 3 and node 1 in the list *Cycle_List* {3-1} and discard column 3 and column 1 (see Figure 4).

0	34.66	24.19	55.04	14.14	62.39	12.65	45.00	31.38
34.66	0	58.41	84.12	20.52	95.38	31.38	60.03	13.93
24.19	58.41	0	35.44	38.01	38.60	29.41	53.67	52.63
55.04	84.12	35.44	0	66.10	21.21	52.89	88.77	73.82
14.14	20.52	38.01	66.10	0	75.58	14.14	49.04	19.10
62.39	95.38	38.60	21.21	75.58	0	64.35	85.99	87.20
12.65	31.38	29.41	52.89	14.14	64.35	0	57.14	23.35
45.00	60.03	53.67	88.77	49.04	85.99	57.14	0	66.71
31.38	13.93	52.63	73.82	19.10	87.20	23.35	66.71	0

3-1
Discard columns 3 et 1

Figure 4 Discard column 3 and column 1

Iteration number 2: Select the minimal element for row 3 (which is 29.41) and for row 1 (which is 12.65) and select the smallest one (which is at position d_{17}), insert node 7 in the *Cycle_List* {3-1-7} and discard its column (see Figure 5).

0	34.66	24.19	55.04	14.14	62.39	12.65	45.00	31.38
34.66	0	58.41	84.12	20.52	95.38	31.38	60.03	13.93
24.19	58.41	0	35.44	38.01	38.60	29.41	53.67	52.63
55.04	84.12	35.44	0	66.10	21.21	52.89	88.77	73.82
14.14	20.52	38.01	66.10	0	75.58	14.14	49.04	19.10
62.39	95.38	38.60	21.21	75.58	0	64.35	85.99	87.20
12.65	31.38	29.41	52.89	14.14	64.35	0	57.14	23.35
45.00	60.03	53.67	88.77	49.04	85.99	57.14	0	66.71
31.38	13.93	52.63	73.82	19.10	87.20	23.35	66.71	0

3-1-7
Discard column 7

Figure 5 Discard column 7

Iteration number 3: Select the minimal element for row 3 (which is 29.41) and for row 7 (which is 14.14) and select the smallest one (which is at position d_{75}), insert node 5 in the *Cycle_List* {3-1-7-5} and discard its column (see Figure 6).

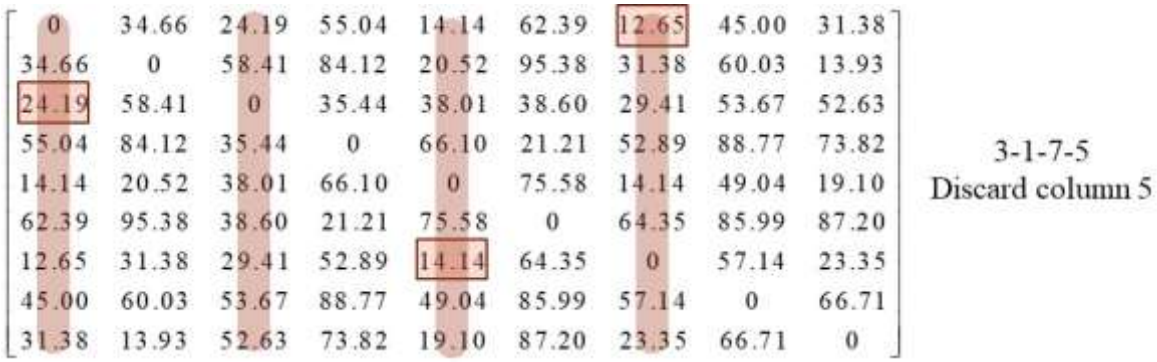


Figure 6 Discard column 5

Iteration number 4: Select the minimal element for row 3 and for row 5 and select the smallest one (which is at position d_{59}), insert node 9 in the *Cycle_List* {3-1-7-5-9} and discard its column (see Figure 7).

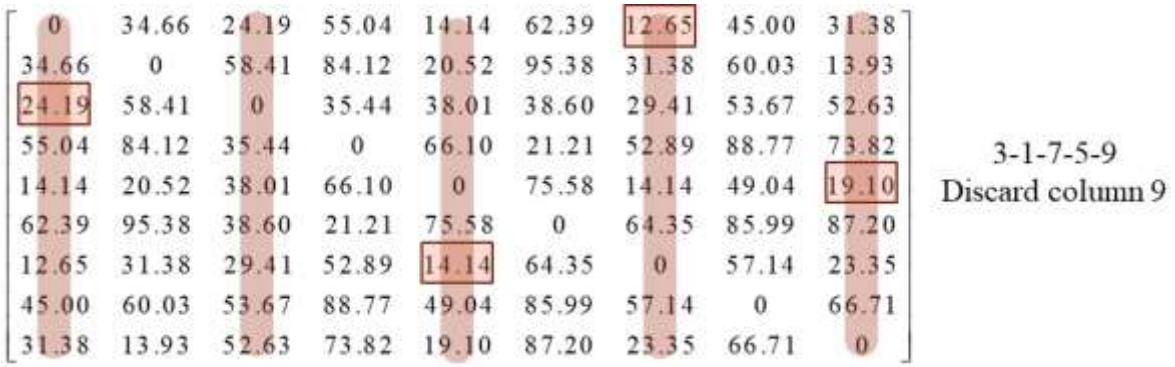


Figure 7 Discard column 9

Iteration number 5: Select the minimal element for row 3 and for row 9 and select the smallest one (which is at position d_{92}), insert node 2 in the *Cycle_List* {3-1-7-5-9-2} and discard its column (see Figure 8).

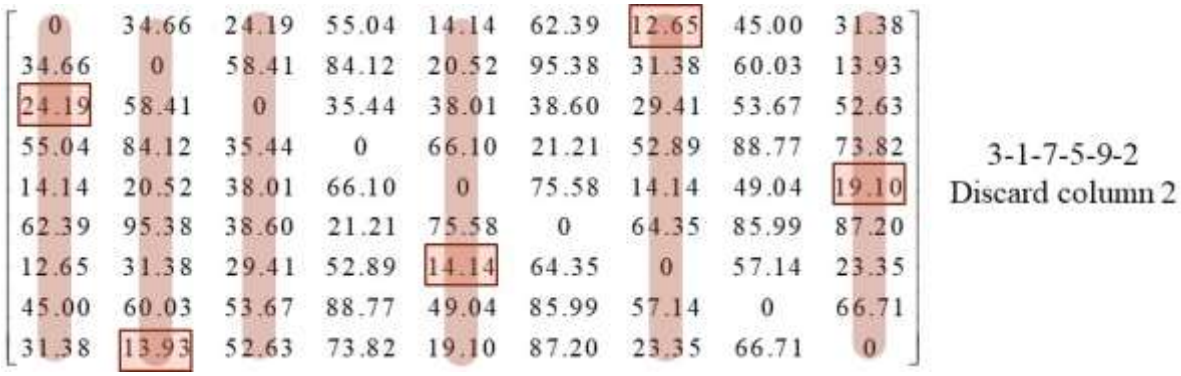


Figure 8 Discard column 2

Iteration number 6: Select the minimal element for row 3 and for row 2 and select the smallest one (which is at position d_{34}), insert node 4 in the *Cycle_List* {4-3-1-7-5-9-2} and discard its column (see Figure 9).

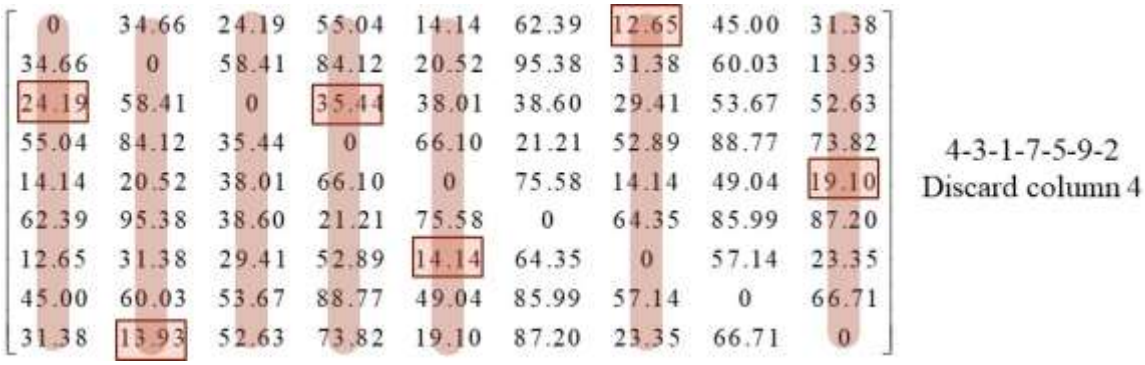


Figure 9 Discard column 4

Iteration number 7: Select the minimal element for row 4 and for row 2 and select the smallest one (which is at position d_{64}), insert node 6 in the *Cycle_List* {6-4-3-1-7-5-9-2} and discard its column (see Figure 10).

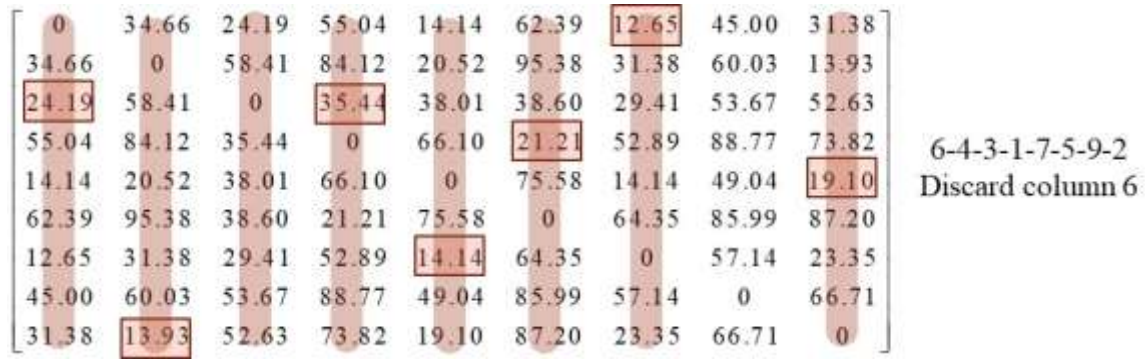


Figure 10 Discard column 6

Iteration number 8: Select the minimal element for row 6 and for row 2 and select the smallest one (which is at position d_{28}), insert node 8 in the *Cycle_List* {6-4-3-1-7-5-9-2-8} and discard its column (see Figure 11).

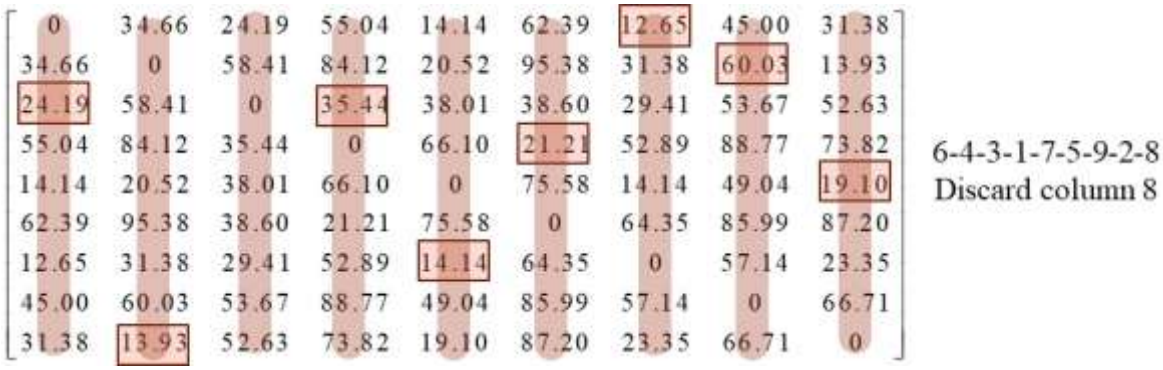


Figure 11 Discard column 8

Iteration number 9: All the columns are discarded, so, no more nodes to select. Then, the last step starts by transforming the found root in the list *Cycle_List* {6-4-3-1-7-5-9-2-8} to a cycle: select the first element in the list *Cycle_List* which is node 6 and translate it to become at the last position of the list {4-3-1-7-5-9-2-8-6}, repeat the same treatment for node 4 and node 3 and finally add the starting node at the last position: {1-7-5-9-2-8-6-4-3-1}.

3.2 Comparative study

The proposed method Dhouib-Matrix-TSP1 was developed with Python programming language (using NumPy and Matplotlib libraries). It can be seen from Figure 12 that the proposed method Dhouib-Matrix-TSP1 generates a satisfactory approximative solution in just 9 iterations.

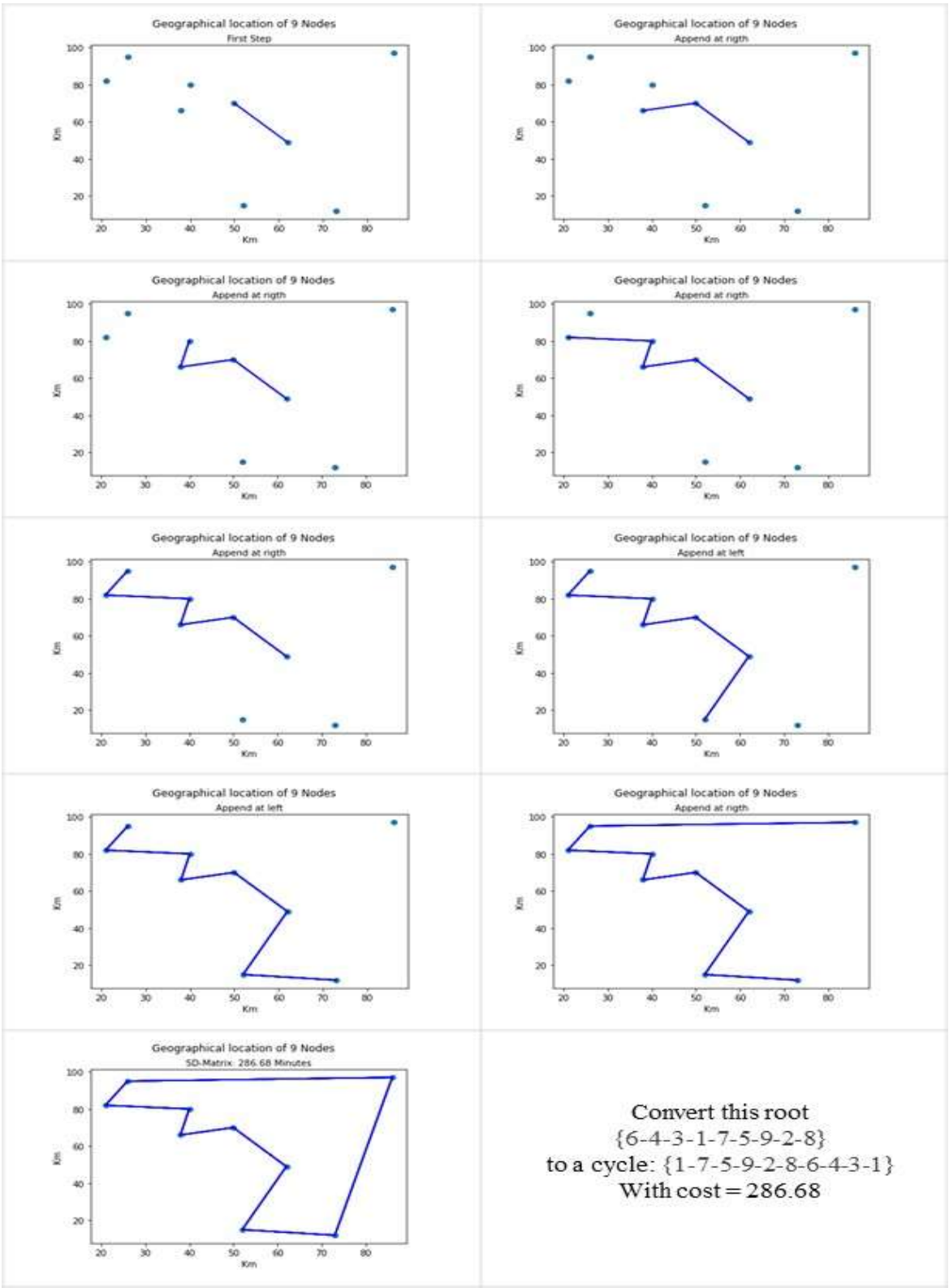


Figure 12 Solution found by Dhouib-Matrix-TSP1

Table 2 shows the comparison of results coming from methods applied on the Milk-Run delivery problem. Our method, Dhouib-Matrix-TSP1, finds a satisfactory near optimal solution (286.68 which represents a deviation of 1.19 % from the optimal solution).

The proposed stochastic method Dhouib-Matrix-TSP2 finds easily the optimal solution (See Figure 13).

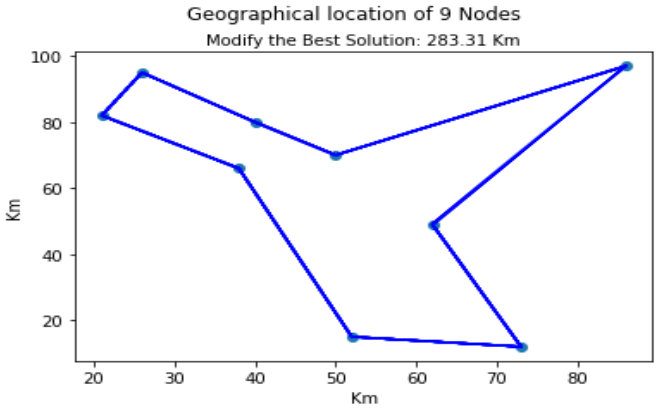


Figure 13 Optimal solution found by Dhouib-Matrix-TSP2

Whereas, [12] finds 286.69 using the Tabu Search algorithm (TS) and 286.22 using a hybrid Ant Colony Algorithm with Tabu Search (HAT). Furthermore, [13] uses the Genetic Algorithm (GA) and hybrid algorithm (HGACO) to find the optimal path.

Table 2 Final results

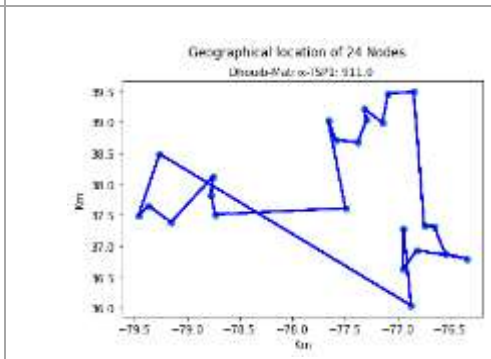
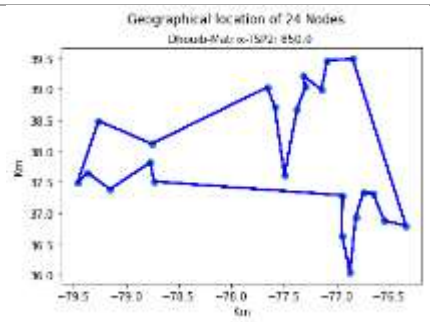
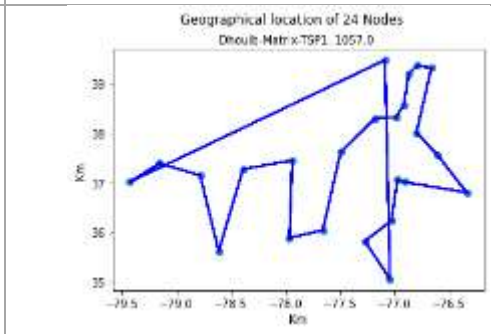
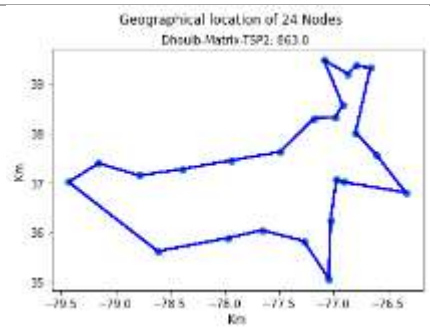
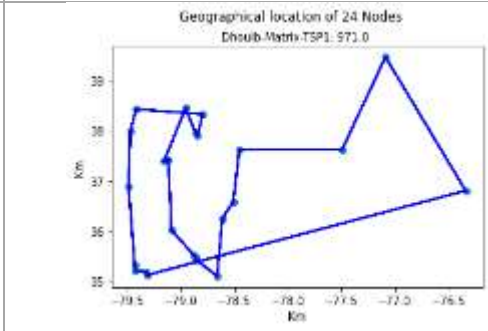
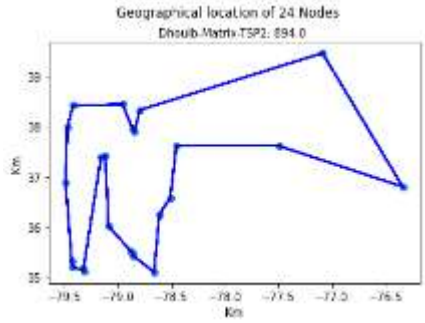
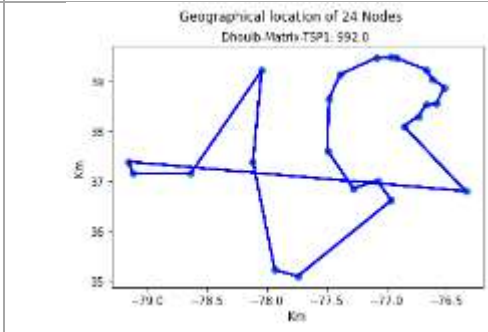
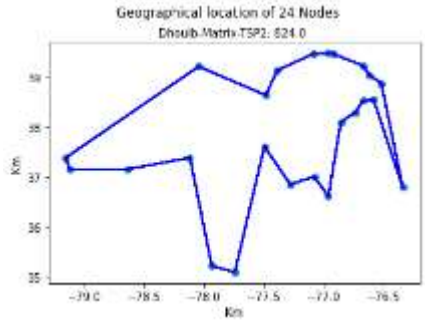
Methods	Distance
MIP	283.30
TS	286.69
HAT	286.22
ACO	286.22
GA	283.30
HGACO	283.30
Dhouib-Matrix-TSP1	286.68
Dhouib-Matrix-TSP2	283.31

Furthermore, the utility of the proposed stochastic method Dhouib-Matrix-TSP2 is demonstrated through another instance from [14] using k =3 (where k represents the nearest neighbors).

Table 3 Comparing Dhouib-Matrix-TSP1 to Dhouib-Matrix-TSP2

Instance from [14]	Dhouib-Matrix-TSP1	Dhouib-Matrix-TSP2
20c3sC1	911	850
20c3sC2	1057	863
20c3sC3	971	894
20c3sC4	992	824
20c3sC5	872	823
20c3sC6	988	923
20c3sC7	1813	1670
20c3sC8	1434	1322
20c3sC9	915	822
20c3sC10	923	872

Table 3. and Figure 14 depict the results coming from the deterministic method Dhouib-Matrix-TSP1 and the stochastic method Dhouib-Matrix-TSP2. It is important to take in consideration that Dhouib-Matrix-TSP1 method uses a very small number of iterations (only *n* iterations, *n* is the number of nodes) compared to the Dhouib-Matrix-TSP2.

Data	Dhouib-Matrix-TSP1	Dhouib-Matrix-TSP2
20c3sC1		
20c3sC2		
20c3sC3		
20c3sC4		

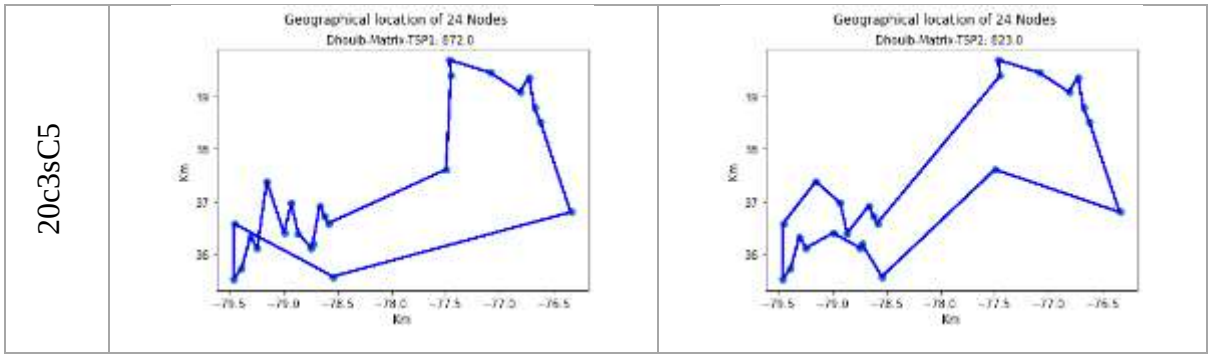


Figure 14 Comparing Dhouib-Matrix-TSP1 to Dhouib-Matrix-TSP2

4. CONCLUSION

In this paper, a case study of automobile industry and several other instances are used to perform the robustness of the newest methods Dhouib-Matrix-TSP1 and Dhouib-Matrix-TSP2 to find the optimal or the near optimal solution in polynomial time. The deterministic method Dhouib-Matrix-TSP1 is characterized by its rapidity whereas Dhouib-Matrix-TSP2 is considered for its quality. As an extension to this work, we aim to insert the Dhouib-Matrix-TSP1 with different statistical metric in multi-start structure.

REFERENCES

[1] G. Bocewicz, E. I. Nielsen, A. Gola and A. Z. Banaszak, "Reference model of milk-run traffic systems prototyping," International journal of production research, DOI: 10.1080/00207543.2020.1766717, pp. , 2020.

[2] X. Cai, L. Jiang, S. Guo and H. Huang, "A Two-Layers Heuristic Search Algorithm for Milk Run with a New PDPTW Model," International Conference on Combinatorial Optimization and Applications, vol.12577, pp. 379-392, 2020.

[3] T. Xiong, Q. Tang, T. Huang, Z. Shen, H. Zhou, K. Y. Hu and Y. Li, "Delivery vehicle scheduling modeling and optimization for automobile mixed milk-run Mode involved indirect suppliers," International Conference on Intelligent Computing for Sustainable Energy and Environment, pp.169-178, 2018.

[4] Y. Lin, Z. Bian, S. Sun and T. Xu, "A Two-Stage Simulated Annealing Algorithm for the Many-to-Many Milk-Run Routing Problem with Pipeline Inventory Cost," Mathematical Problems in Engineering, vol. 3, pp. 1-22, 2015.

[5] Y. Zhenlai and J. Yang, "Development and Application of Milk-Run Distribution Systems in the Express Industry Based on Saving Algorithm," Mathematical Problems in Engineering, vol. 2014, ID 536459, pp. 6-13, 2014.

[6] S. H. Kilic and B. M. Durmusoglu, "A mathematical model and a heuristic approach for periodic material delivery in lean production environment", International Journal of Advanced Manufacturing Technology, vol.69, pp.977-992, 2013.

[7] F. L. Wang, P. Liu, Y. Huang, C. Y. Wang and S. F. Kong, "GA: Research on milk-run routing problem of automotive supply logistics based on genetic algorithm," Proceedings of International Asia Conference on Industrial Engineering and Management Innovation: Core Areas of Industrial Engineering, IEMI, pp.517-526, 2013.

[8] B. S. Gurinder and S. Gagan, “Milk-run logistics: literature review and directions,” Proceedings of World Congress on Engineering, vol. 1, pp. 1-5, 2011.

[9] Z. Jiang, Y. Huang and J. Wang, “Routing for the milk-run pickup system in automobile parts supply,” Advances in Intelligent and Soft Computing, vol.66, pp. 1267-1275, 2010.

[10] S. Dhouib, "A New Column-Row Method for Traveling Salesman Problem: The Dhouib-Matrix-TSP1," International Journal of Recent Engineering Science, vol. 8, Issue 1, pp.6-10, 2021.

[11] S. Dhouib “Stochastic Column-Row Method for Travelling Salesman Problem: the Dhouib-Matrix-TSP2,” International Journal of Engineering Research & Technology, vol. 10, no. 3, pp. 524-527, 2021.

[12] T. Nguyen and T. Dao, “Supply Chain Milk-run Delivery Optimization,” The Journal of Management and Engineering Integration, vol. 8, no. 1, pp. 29-40, 2015.

[13] M. Hfeda, F. Marchand and T. Dao, “Optimization of Milk-Run Delivery Issue in Lean Supply Chain Management by Genetic Algorithm and Hybridization of Genetic Algorithm with Ant Colony Optimization: an Automobile Industry Case Study,” The Journal of Management and Engineering Integration, vol. 10, no. 2 pp 90-100, 2017.

[14] S. Erdogan, and E. Miller-Hooks, “A green vehicle routing problem,” Transportation Research Part E: Logistics and Transportation Review, vol. 48, no. 1 pp. 100-114, 2012.