Hindawi
Applied Computational Intelligence and Soft Computing
Volume 2021, Article ID 7761993, 10 pages
https://doi.org/10.1155/2021/7761993

Research Article

Novel Metaheuristic Based on Iterated Constructive Stochastic Heuristic: Dhouib-Matrix-3 (DM3)

Souhail Dhouib

Higher Institute of Industrial Management, University of Sfax, Sfax, Tunisia

Correspondence should be addressed to Souhail Dhouib; souh.dhou@gmail.com

Received 25 June 2021; Revised 12 October 2021; Accepted 3 December 2021; Published 27 December 2021

Academic Editor: Babak Daneshvar Rouyendegh (B. Erdebilli)

This paper presents a new metaheuristic named Dhouib-Matrix-3 (DM3) inspired by our recently developed constructive stochastic heuristic Dhouib-Matrix-TSP2 (DM-TSP2) and characterized by only one parameter: the number of iterations. The proposed metaheuristic DM3 is an iterative algorithm in which every iteration is based on two relay hybridization techniques. At first, the constructive stochastic heuristic DM-TSP2 starts by generating a different initial basic feasible solution and then each solution is intensified by the novel procedure Far-to-Near which exchanges far cities by closer ones using three perturbation techniques: insertion, exchange, and 2-opt. Experimental results carried out on the classical travelling salesman problem using the well-known TSP-LIB benchmark instances demonstrate that our approach DM3 outclasses the simulated annealing algorithm, the genetic algorithm, and the cellular genetic algorithm. Furthermore, the proposed DM3 is statistically concurrent to the hybrid simulated annealing cellular genetic algorithm. Nevertheless, DM3 is easier to implement and needs only one parameter to identify (the maximum number of iterations).

1. Introduction

In real-world industry, decision makers are daily confronted with several complicated combinatorial optimization problems such as the scheduling problem, the transportation problem, the knapsack problem, the travelling salesman problem (TSP), and so on. In order to help the decision maker to solve these kinds of problems, there are two major categories of optimization algorithms: the exact and the approximative methods. The first category of algorithms is unable to solve complex problems in that it is characterized by the huge consumption of computational time, whereas the second category, namely, the approximative methods, tries to find a balance between time consumption and quality of the generated solution by partially searching the space of solutions.

Actually, the approximative methods gather heuristics as well as metaheuristics. The heuristics are dedicated to specific optimization problems, whereas the metaheuristics are more flexible in a way that they can solve several complex optimization problems (by offering acceptable solutions but without guaranteeing finding the optimality).

In this context, tremendous research works are developed in order to prove the efficiency of metaheuristics in the resolution of NP-hard problems. The majority of metaheuristics are stimulated from the collective behavior of creatures in nature. The genetic algorithm (GA) was designed in [1] and was proven as a very successful optimization technique especially with its various variants such as those in [2, 3]. The harmony search (HS) algorithm is based on the search of the best of harmonics, which was firstly introduced in [4]. The artificial bee colony (ABC) is inspired by honey bee mating, which was proposed in [5]. The firefly algorithm (FA) is an optimization algorithm depicted in [6] and invigorated from lights and radiance of fireflies. In [7], the spider monkey algorithm was proposed, which is a swarm-based method that takes benefits from spider monkeys in nature.

There are many other population-based algorithms, namely,the grey wolf optimizer (GWO) illustrated in [8] which was inspired from the hierarchy and the hunting process of grey wolves in forests. An improved version of GWO was proposed in [9] in which the biological

evolution and the survival of the strongest in the nature's updating are included in the basic WGO. The bird mating optimizer (BMO) is an evolutionary algorithm designed in [10] and dedicated for continuous optimization problems and invigorated by bird species strategies along the mating season. In 2016, the whale optimization algorithm (WOA) was firstly introduced in [11], and it is based on the behavior of whales in nature. An improved version of WOA was depicted in [12]. The farmland fertility algorithm (FFA) inspired from farmland behavior in nature was presented in [13], and improved FFA was proposed in [14]. The African vultures optimization algorithm (AVOA) was illustrated in [15], which is based on the navigation and foraging behaviors of African vultures.

Nevertheless, all these metaheuristics have several parameters that must be fixed before starting the resolution process such as the size of the tabu search memory, the number of neighbourhoods, the initial and the final temperatures, the size of populations, the number of iterations, and so on. In reality, the performance of any metaheuristic is dependent on the best equilibration between all its parameters, and for that, an exhaustive sensitive analysis needs to be done for each parameter. Consequently, in order to simplify this issue, we propose in this paper a new metaheuristic entitled Dhouib-Matrix-3 (DM3) which is characterized by only one parameter: the number of iterations.

The proposed metaheuristic DM3 hybridizes in iterated structure a stochastic heuristic with a local search method. For that, our stochastic heuristic Dhouib-Matrix-TSP2 (DM-TSP2) is used to explore different initial basic feasible solutions [16]. DM-TSP2 is a column-row method, and it is characterized by its simplicity and rapidity. Furthermore, our novel local search method Far-to-Near (FtN) is designed in order to exploit each solution generated by DM-TSP2 (see Figure 1).

Actually, FtN helps to guide the search by affecting far cities to their neighbours. It allows for each city to test its connected nodes and generates a new solution using three perturbation techniques: insertion, exchange, and 2-opt. The FtN procedure accepts any new solution which is not so far from the optimal solution.

Hence, DM3 drives the search space through two steps in an iterated structure (see Figure 2). At first, it rapidly generates a good initial basic feasible solution using the stochastic heuristic DM-TSP2; then, it intensifies this initial solution using the original procedure, namely, the FtN.

The performance of the proposed metaheuristic DM3 is proved through its application on the standard travelling salesman problem (TSP). In fact, TSP deals with finding a shortest distance d between n cities that are visited once except the starting city which will be visited twice because it has been already the last visited city. TSP is mathematically formulated as in equation (1) which indicates that d_{ij} denotes the distance between city i and city j while x_{ij} presents a binary variable ($x_{ij} = 1$ if city i is connected to city j; otherwise, $x_{ij} = 0$):

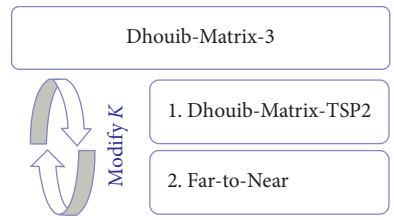


Figure 1: The proposed metaheuristic DM3.

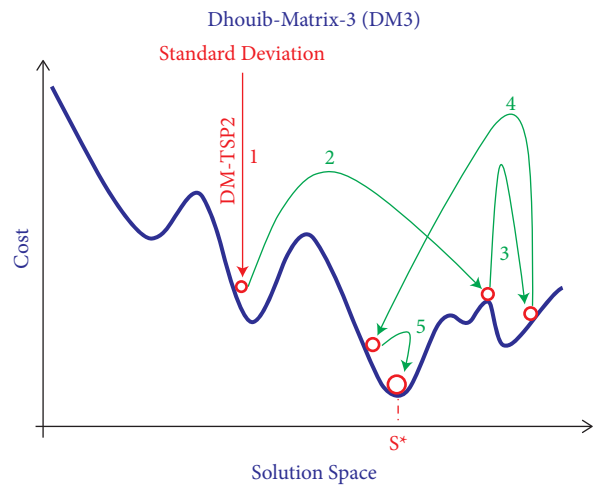


Figure 2: The exploration of the solution space through DM3.

$$\begin{aligned} \text{minimize}: & \sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}x_{ij} \\ & \sum_{j=1}^{n} x_{ij} = 1,\ i = 1,\ldots,n, \\ \text{subject to}: & \sum_{i=1}^{n} x_{ij} = 1,\ j = 1,\ldots,n, \\ & x_{ij} = 0 \text{ or } 1,\ i = 1,\ldots,n,\ j = 1,\ldots,n. \end{aligned} \tag{1}$$

This paper is organized as follows. The next section provides all the details about the stochastic constructive heuristic DM-TSP2. Section 3 describes the new proposed metaheuristic DM3. Section 4 depicts the numerical application of the proposed method DM3 on TSP-LIB instances with comparison to other developed metaheuristics from the literature. Finally, the conclusion and further work will be given in Section 5.

2. The Dhouib-Matrix-TSP2 Method (DM-TSP2)

In [16], we designed and developed the novel heuristic DM-TSP2 which is a stochastic constructive algorithm based on several rules in order to orient the search path. DM-TSP2 is inspired from our deterministic heuristic, namely, Dhouib-Matrix-TSP1 (DM-TSP1), proposed in [17]. The robustness

of DM-TSP1 was proven under fuzzy environments in [18–20] and neutrosophic domains in [21, 22].

DM-TSP2 is composed of four simple phases and characterized by its rapidity: it needs only n iterations to find a good initial basic feasible solution (even the optimal). Figure 3 depicts the general body of the DM-TSP2 algorithm in which K represents the number of the nearest nodes (K is a positive integer). All the details concerning the stepwise application of DM-TSP2 on several examples are exposed in [16, 23].

In order to explain the stochastic heuristic DM-TSP2, here is a stepwise application on TSP with $K = 2$ and 4×4 distance matrix presented in Figure 4. To solve this problem, the heuristic DM-TSP2 needs only four simple iterations to generate the optimal or near-optimal solution.

$$\begin{pmatrix} \infty & 12 & 13 & 14 \\ 12 & \infty & 9 & 7 \\ 13 & 9 & \infty & 10 \\ 14 & 7 & 10 & \infty \end{pmatrix}. \tag{2}$$

The first step is to compute the standard deviation which will be presented at the last column (see Figure 5). Next, select the K highest elements ($K = 2$, so the values 5.67 and 5.12 are selected) and randomly choose one of them. Let us assume that 5.12 is randomly selected; then, find the K minimal elements in row 4 (which correspond to 7 and 10) and randomly choose one of them (let us assume the element at position d_{42} with the value of 7).

Next, insert cities 4 and 2 in List-cities {4-2} and discard their corresponding columns (see Figure 6).

Hence, select the K smallest elements for rows 4 and 2 which are equal to 10 and 14 for row 4 and equal to 12 and 9 for row 2. Then, for row 4, choose randomly a value, say 10. Similarity, for row 2, select 12. Now, find the minimal element which is equal to 10 at position d_{43}. So, insert city 3 in List-cities {3-4-2} and discard column 3 (see Figure 7).

There is only one element in column 3 and in column 2, so select the smallest which is 12 at position d_{21}, insert city 1 in List-cities {3-4-2-1}, and discard column 1 (see Figure 8).

Now, there is no more column to discard. So, to generate the initial basic feasible solution, a cycle is needed to be obtained from List-cities {3-4-2-1}. Therefore, we must translate one by one the positions of all cities before city 1 at right with the respected order. Hence, city 3 is translated after city 1. Then, similarly, city 4 and city 2 are added to List-cities {1-3-4-2}. Finally, complete List-cities with city 1 at the end position {1-3-4-2-1}.

DM-TSP2 is a very simple and fast stochastic heuristic. It needs only 4 iterations (where 4 represents the number of cities) to generate an initial basic feasible solution. In order to perform DM-TSP2, a guided repetitive structure is needed and for that we design and develop the metaheuristic DM3.

3. The Proposed Metaheuristic: Dhouib-Matrix-3 (DM3)

The proposed metaheuristic DM3 iterates in a sequential manner two techniques: the stochastic constructive method DM-TSP2 and the novel procedure FtN. In fact, the iterated algorithms were widely used in literature. The author [24] implemented the variable neighbourhood descent with an iterated local search algorithm in order to solve the TSP with hotel selection. The author [25] also optimized the weighted vertex colouring problem using an iterated local search algorithm with two new neighbour structures. In [26], the author solved the capacitated vehicle routing problems using the iterated local search algorithm with path-relinking method. In [27] generated an iterated local search metaheuristic with large-scale neighbour technique to solve the TSP. In [28], a hybrid multistart iterated local search algorithm was designed with two new compact mixed integer programming formulations with the aim of minimizing the weighted feedback vertex set problem. In [29], a new mathematical programming formulation was generated, and an iterated local search method was in order to minimize the sum of the travelling and the waiting times at the depot. Cacchiani et al. [30] developed an iterated local search algorithm to solve the pollution TSP that looks for generating a Hamiltonian cycle which optimizes a function of driver costs and fuel consumption (which depends on load, speed, and distance). In [31], the iterated local search was used with the variable neighbourhood descent to solve the multivehicle inventory routing problem by minimizing the total cost of storage and transportation. In [32], an iterated local search metaheuristic was designed using a novel perturbation technique for the split delivery vehicle routing problem. In [33], the periodic vehicle routing problem was solved with time windows using a multistart iterated local search algorithm.

This paper is focused on proposing a novel metaheuristic entitled DM3 in an iterated structure. DM3 starts by generating a constructive realizable solution using DM-TSP2. Subsequently, the procedure FtN will intensify the solution generated by DM-TSP2 using three different perturbation techniques. Next, the value of K will be modified (where K represents the number of the nearest nodes, so K is always a positive integer), and the process will be iterated again (see Figure 9).

In reality, the procedure FtN is proposed with the intention of organizing the intensification research path. The main idea of this procedure is to eliminate every far neighbour node. As it is detailed in [19, 20], DM-TSP2 is a constructive method which uses in the beginning the rule of the nearest cities with some perturbations. Therefore, after the affectation of some cities, this rule (nearest cities at first) cannot be applied since the nearest cities have been already affected in the first steps of the process. Thus, after the application of DM-TSP2, the proposed procedure FtN will be launched with three perturbation techniques, namely, insertion, exchange, and 2-opt, in order to generate a neighbour solution from the current one.

FtN subsequently applies three perturbation techniques for all the nodes and not randomly (see Figure 10). It starts by selecting one by one the last inserted cities (in the proposed realizable solution generated by DM-TSP2) and then applying for each selected city the three perturbation

Dhouib-Matrix-TSP2 (DM-TSP2)

Input: Distance-matrix, K
Output: Optimal or near optimal path

```
Set i = 1 and S*=∞
Repeat
   Set Cycle_List = {}
   Compute the standard deviation for each row
   Select randomly one element (row) from the K highest standard deviations. Then, for
   this selected row, choose randomly one element from the K minimal elements: let us say
   that it is at position d_xy
   Cycle_List = {x,y}
   Discard column x and column y
   For i=1 to number of cities
       Select one element from the K minimal elements d_xa in row x
       Select one element from the K minimal elements d_yb in row y
       If (d_xa < d_yb) then
           Insert a before x in the list Cycle_List
           x = a
       Else
           Insert b after y in the list Cycle_List
           y = b
       End
   Compute S' from Cycle_List
   If (S* > S') then
      S* = S'
   End
   i = i + 1
Until (i > maximal number of iterations)
Return S*
```

Figure 3: The general body of the DM-TSP2 algorithm.

$$\begin{pmatrix} \infty & 12 & 13 & 14 \\ 12 & \infty & 9 & 7 \\ 13 & 9 & \infty & 10 \\ 14 & 7 & 10 & \infty \end{pmatrix}$$

Figure 4: The 4×4 distance matrix.

$$\begin{array}{c} \begin{bmatrix} \infty & 12 & 13 & 14 \\ 12 & \infty & 9 & 7 \\ 13 & 9 & \infty & 10 \\ 14 & \boxed{7} & 10 & \infty \end{bmatrix} \end{array} \begin{array}{c} 5.67 \\ 4.42 \\ 4.85 \\ 5.12 \end{array}$$

Figure 5: The 4×4 distance matrix.

$$\begin{bmatrix} \infty & 12 & 13 & 14 \\ \boxed{12} & \infty & 9 & 7 \\ 13 & 9 & \infty & 10 \\ 14 & \boxed{7} & \boxed{10} & \infty \end{bmatrix}$$

Figure 6: Discarding columns 2 and 4.

$$\begin{bmatrix} \infty & 12 & 13 & 14 \\ \boxed{12} & \infty & 9 & 7 \\ \boxed{13} & 9 & \infty & 10 \\ 14 & \boxed{7} & 10 & \infty \end{bmatrix}$$

Figure 7: Discarding column 3.

$$\begin{bmatrix} \infty & 12 & 13 & 14 \\ 12 & \infty & 9 & 7 \\ 13 & 9 & \infty & 10 \\ 14 & \boxed{7} & 10 & \infty \end{bmatrix}$$

Figure 8: Discarding column 1.

Dhouib-Matrix-3 (DM3)

Input: Distance-matrix
Output: Optimal or near optimal path

```
Set S* = +∞
Set K = 1
S1 = Dhouib-Matrix-TSP2 (Distance-matrix, K)
S* = Far-to-Near (K, S1)
K = K + 1
Repeat
    S2 = Dhouib-Matrix-TSP2 (Distance-matrix, K)
    S3 = Far-to-Near (Distance-matrix, K, S2)
    If ((S* - S3) > = 0) then
        S* = S3
    Else
        K = K + int (random (1,-1))
    End
Until termination condition is met
Return S*
```

Figure 9: The general body of the proposed metaheuristic DM3.

techniques (insertion, exchange, and 2-opt) using the nearest K nodes. For example, if the insertion technique is selected, $K = 3$ and the node 5 is chosen so the FtN procedure will insert city 5 at position of one city from its three nearest cities ($K = 3$).

Regarding the acceptation criterion, FtN selects the best neighbour solution and then accepts it only if it is not far than phi (φ) percent from the best solution where $\varphi = (1 + \sqrt{5})/2 = 0.1618$ represents the golden ratio (GR). φ represents the unique solution for the equation $x^2 = x + 1$ with the characteristic of $\varphi^2 = \varphi + 1$ and $1/\varphi = \varphi - 1$.

In nature, patterns of GR appear especially in the proportions of body dimensions, the spiral arrangement of shell, the arrangement of leaves (phyllotaxis), and additional plant parts. GR has been also used by artists, architects, music composers, decision makers, and scientists in order to analyse a model proportion of natural objects as systems such as investing in financial markets (several technical analyses use GR to handle price level).

4. Results and Discussion for a TSP-LIB Problem

In order to confirm the performance of the proposed metaheuristic DM3, twelve standard benchmark instances taken from TSP-LIB (with up to 200 nodes) are used which are the most challenging instances for the travelling salesman problem (olivre30, chn31, eil51, berlin52, eil76, ktpa100, pr107, bier127, ch130, pr136, pr152, and kroa200). Note that the results found by the proposed method DM3 are compared to the results found by the simulated annealing (SA), the genetic algorithm (GA), the cellular genetic algorithm (CGA), and the hybrid simulated cellular genetic algorithm (SCGA) presented in [34].

The numerical experiments are conducted using a laptop computer with the following characteristics: Windows 10 Pro, Intel Core i5 2.5 GHz, and 8 GB in RAM. The program is coded in Python programming language using NumPy and Matplotlib libraries. With the intention of comparing the results, we execute DM3 in a multiple run composed of

```
Far-to-Near (FtN)

Input: Distance matrix, K, S
Output: Optimal or near optimal path

Set S* = S
Repeat
   S0 = S
   j = size (S) - 1
   Repeat
      v = select one city from the K nearest neighbors of S (j)
      S1 = insert (v, S(j))
      S2 = exchange (v, S(j))
      S3 = 2-opt (v, S(j))
      S4 = select the best solution between (S1, S2, S3)
      if ( (S4 – S*)<GR)
         S = S4
         if (S* < S4)
            S* = S4
         end
      end
   j = j - 1
   Until (j < 2)
Until (S0 = S)
Return S*
```

Figure 10: The general body of the proposed local search metaheuristic FtN.

thirty iterations with a time limit equal to 60 seconds if the number of nodes is less than 100 and 180 seconds otherwise.

Table 1 summarizes the results obtained by the proposed method DM3 and the other metaheuristics for the TSP-LIB instances. The deviation error (DE) percentage is calculated by $\mathrm{DE} = (\text{best–optimal})/\text{optimal}) \times 100$ where best reports the best solution obtained by DM3 and optimal presents the optimal solution reported in TSP-LIB. The deviation error (DE) ranges from 00.00% to 01.01% when the number of cities varies from 30 to 200.

Figure 11 depicts the variation of the DE indicator for DM3 and other metaheuristics when the TSP size (number of nodes) increases from 30 to 200 nodes. We can conclude that when the number of nodes is less than 100, the proposed method can generate the optimal or the near-optimal solution in a computational time less than 60 seconds. However, the percentage of deviation error (DE) will increase depending on the TSP-LIB instance size (number of nodes).

As shown in Figure 11, we can easily conclude that SCGA and DM3 are significatively better than SA, GA, and CGA for all the instances. Furthermore, the total RE for all instances confirm that the generated solutions by DM3 and SCGA are very close (where their corresponding RE values are 3.39 and 3.54) and better than those generated by SA, GA, and CGA (where their corresponding RE values are, respectively, 55.17, 127.22, and 33.43).

Numerically, DM3 is so close to SCGA (respectively, 3.39 and 3.54), and we cannot make comparison between them. However, we can compare their distribution through the Mann–Whitney U nonparametric test using the IBM SPSS Statistics software. This test will take into account the variation of the results using the RE indicator. We will test the following null hypothesis: "the distribution of RE is the same, so DM3 and SCGA are equal," and the alternative hypothesis: "their distribution is not equal, so DM3 is different from SCGA." We will fix also the significance risk level at 0.05%. The corresponding p value is equal 0.514 (see Figure 12) which is not less than 0.05, and we fail to reject the null hypothesis. Thus, DM3 and SCGA are statistically equal (same distribution).

As shown in Figure 13, a comparison of DM3 with SA, GA, CGA, and SCGA metaheuristics is given. These results show that we can affirm that for the RE indicator, the results found by DM3 (on all the instances tested in this work) are better than those found by the other methods except the SCGA metaheuristic which is statistically equal to DM3.

Table 1: Results found by DM3 and other metaheuristics applied on TSP-LIB instances.

Instance	Optimal	SA		GA		CGA		SCGA		DM3	
		Best	RE	Best	RE	Best	RE	Best	RE	Best	RE
oliver30	420	425	1.12	424	1.22	424	0.89	424	0.89	420	0.00
chn31	15377	15487	0.72	15451	4.69	15378	0.00	15378	0.00	15377	0.00
eil51	426	438	2.73	440	7.63	439	2.95	429	0.67	427	0.23
berlin52	7542	7899	4.73	7685	9.26	7813	3.59	7544	0.03	7542	0.00
eil76	538	571	6.13	575	15.15	562	4.46	547	1.70	540	0.37
kroa100	21282	21985	3.30	22094	15.89	21544	1.23	21285	0.02	21282	0.00
pr107	44303	45983	3.79	45162	11.49	44579	0.62	44302	0.00	44303	0.00
bier127	118282	128515	8.65	126213	12.21	123578	4.48	118294	0.01	118839	0.47
ch130	6110	6542	7.07	6493	12.95	6350	3.93	6183	1.20	6169	0.97
pr136	96772	104785	8.28	105372	14.21	102048	5.45	96795	0.02	96790	0.02
pr152	73682	75801	2.88	76993	12.68	74889	1.64	73684	0.00	74089	0.55
kroa200	29368	32406	10.34	34197	23.38	31725	8.03	29533	0.56	29666	1.01

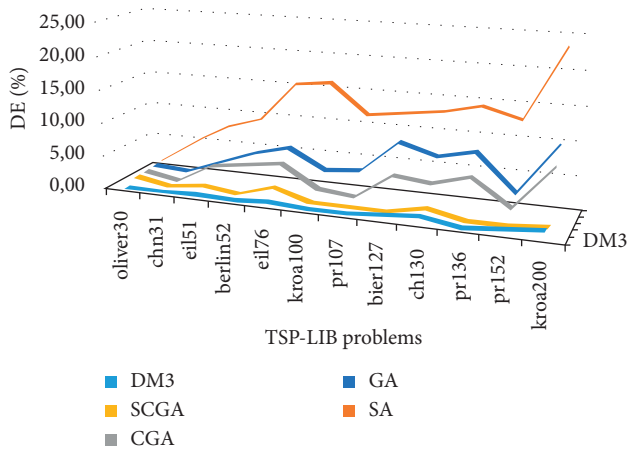


Figure 11: Trend of the metaheuristic DM3 when the number of nodes increases.

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
1	The distribution of DE is the same across categories of Group.	Independent-Samples Mann-Whitney U Test	.514[1]	Retain the null hypothesis.

Asymptotic significances are displayed. The significance level is .05.

[1]Exact significane is displayed for this test.

Figure 12: Comparing DM3 with SCGA using the Mann–Whitney U test through IBM SPSS Statistics software.

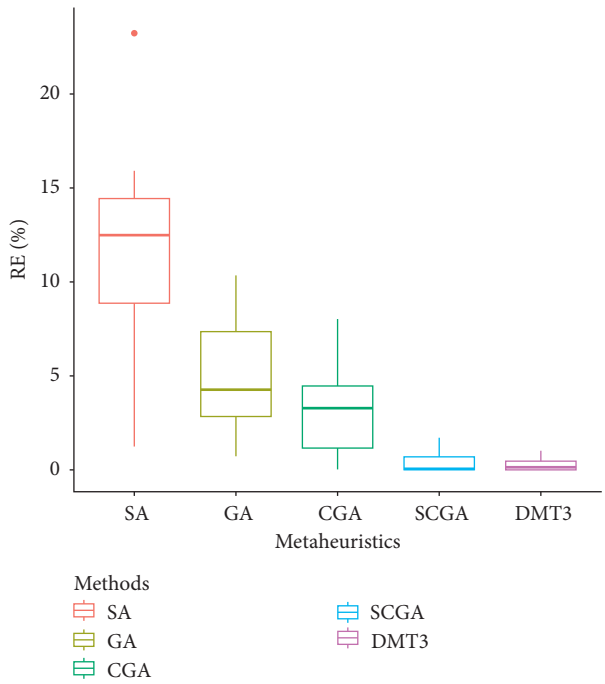
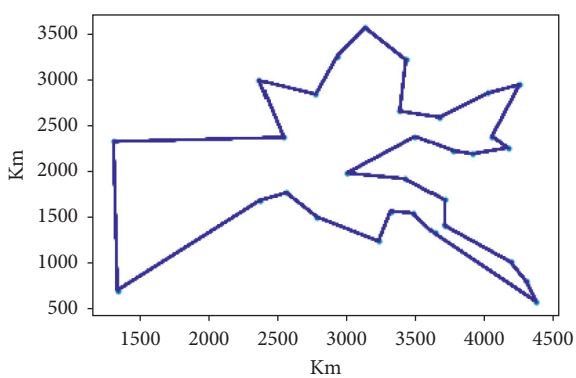


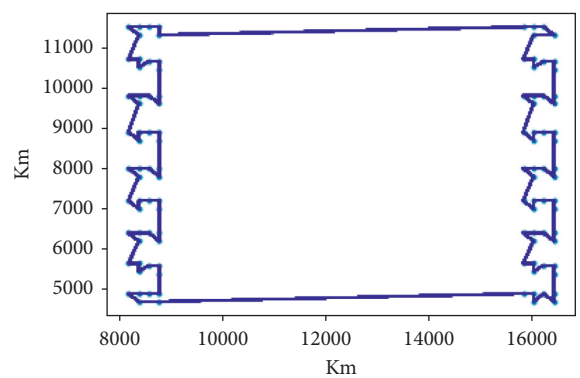
Figure 13: Box plots for comparing DM3 with SA, GA, CGA, and SCGA.

Table 2: Solutions found by the proposed metaheuristic DM3.

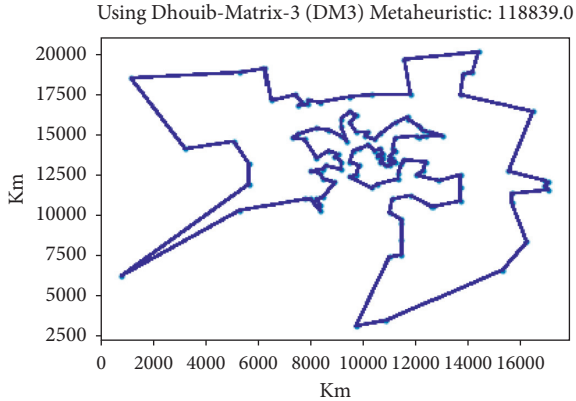
(a) chn31
Geographical location of 31 Nodes
Using Dhouib-Matrix-3 (DM3) Metaheuristic: 15377.0



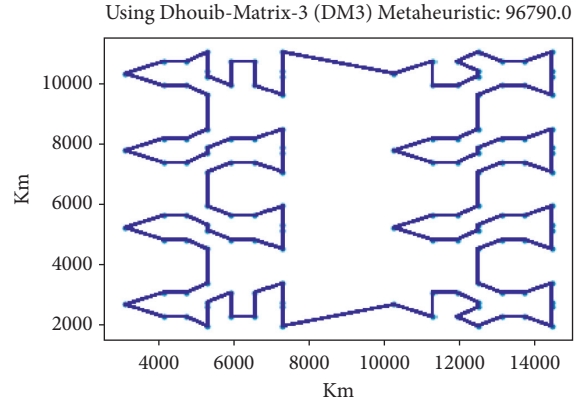
(b) pr107
Geographical location of 107 Nodes
Using Dhouib-Matrix-3 (DM3) Metaheuristic: 44303.0



(c) bier127
Geographical location of 127 Nodes
Using Dhouib-Matrix-3 (DM3) Metaheuristic: 118839.0



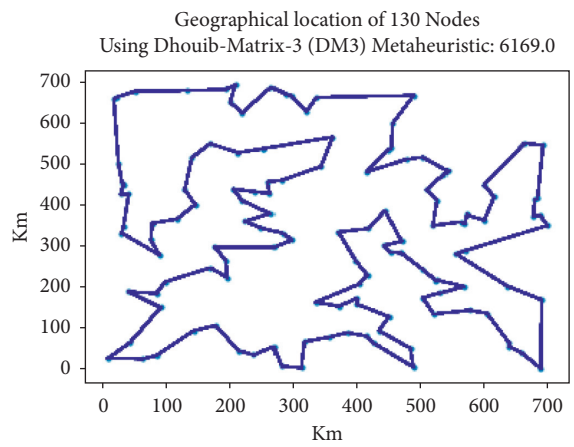
(d) pr136
Geographical location of 136 Nodes
Using Dhouib-Matrix-3 (DM3) Metaheuristic: 96790.0



(e) ch130

(f) pr152

Table 2: Continued.



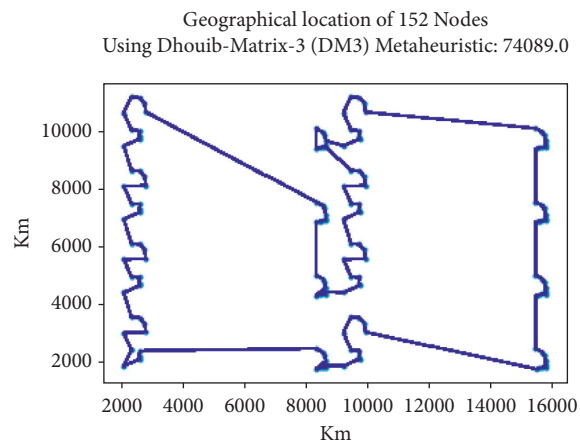


Table 2 shows the graphical representation of the best solution found by DM3 for several TSP-LIB problems: (a) chn31, (b) pr107, (c) bier127, (d) pr136, (e) ch130, and (f) pr152.

5. Conclusion

This paper describes a new metaheuristic named Dhouib-Matrix-3 (DM3) characterized by only one parameter: the number of iterations. This method hybridizes the stochastic constructive method Dhouib-Matrix-TSP2 (DM-TSP2) with an original method named the Far-to-Near in an iterated structure. In fact, each iteration in DM3 starts by generating a good initial solution through the method DM-TSP2; then, this realizable solution will be intensified using the novel procedure Far-to-Near by the means of three different perturbation techniques.

The performance of the proposed metaheuristic DM3 is tested on well-known TSP-LIB instances. The computational results show that the novel method DM3 is very competitive, and it outclasses the simulated annealing algorithm, the genetic algorithm, and the cellular genetic algorithm and is statistically concurrent to the hybrid simulated cellular genetic algorithm. Further research will deal with the adaptation of the proposed method DM3 to generate Pareto nondominated set solutions for multiobjective travelling salesmen problem.

Data Availability

The TSP-LIB data used to support the findings of this study are publicly available at http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/tsp/.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Supplementary Materials

Graphical abstract. (*Supplementary Materials*)

References

[1] J. H. Holland, "Genetic algorithms," *Scientific American*, vol. 267, no. 1, pp. 66–72, 1992.

[2] G. D'Angelo and F. Palmieri, "GGA: a modified genetic algorithm with gradient-based local search for solving constrained optimization problems," *Journal of Information Science*, vol. 547, pp. 136–162, 2021.

[3] J. Xu, "Improved genetic algorithm to solve the scheduling problem of college English courses," *Complexity*, vol. 2021, Article ID 7252719, 11 pages, 2021.

[4] Z. W. Zong Woo Geem, J. H. Joong Hoon Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: harmony search," *Simulation*, vol. 76, no. 2, pp. 60–68, 2001.

[5] D. Karaboga and B. Basturk, "Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems," *Lecture Notes in Computer Science*, vol. 4529, pp. 789–798, 2007.

[6] X. Yang, *Nature-Inspired Metaheuristic Algorithms*, pp. 81–96, Luniver Press, Bristol, England, UK, 2010.

[7] J. C. Bansal, H. Sharma, S. S. Jadon, and M. Clerc, "Spider monkey optimization algorithm for numerical optimization," *Memetic Computing*, vol. 6, no. 1, pp. 31–47, 2014.

[8] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[9] J. S. Wang and S. X. Li, "An improved grey wolf optimizer based on differential evolution and elimination mechanism," *Scientific Reports*, vol. 9, no. 7181, pp. 7181–7221, 2019.

[10] A. Askarzadeh, "Bird mating optimizer: an optimization algorithm inspired by bird mating strategies," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 4, pp. 1213–1228, 2014.

[11] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.

[12] G.-Y. Ning and D.-Q. Cao, "Improved whale optimization algorithm for solving constrained optimization problems," *Discrete Dynamics in Nature and Society*, vol. 2021, Article ID 8832251, 13 pages, 2021.

[13] H. Shayanfar and F. S. Gharehchopogh, "Farmland fertility: a new metaheuristic algorithm for solving continuous optimization problems," *Applied Soft Computing*, vol. 71, pp. 728–746, 2018.

[14] Y.-J. Wang and Y. Chen, "An improved farmland fertility algorithm for global function optimization," *IEEE Access*, vol. 8, pp. 111850–111874, 2020.

[15] B. Abdollahzadeh, F. S. Gharehchopogh, and S. Mirjalili, "African vultures optimization algorithm: a new nature-inspired metaheuristic algorithm for global optimization problems," *Computers & Industrial Engineering*, vol. 158, pp. 1–37, 2021.

[16] 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.

[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 and S. Dhouib, "Solving the single-valued trapezoidal neutrosophic transportation problems through the novel dhouib-matrix-TP1 heuristic," *Mathematical Problems in Engineering*, vol. 2021, no. 2, 11 pages, Article ID 3945808, 2021.

[19] M. Miledi, S. Dhouib, and T. Loukil, "Dhouib-Matrix-TSP1 method to optimize octagonal fuzzy travelling salesman problem using α-cut technique," *International Journal of Computer and Information Technology*, vol. 10, no. 3, pp. 130–133, 2021.

[20] S. Dhouib, "Haar dhouib-matrix-TSP1 method to solve triangular fuzzy travelling salesman problem," *Research Journal of Recent Sciences*, vol. 10, no. 3, pp. 18–20, 2021.

[21] S. Dhouib, "Neutrosophic triangular fuzzy travelling salesman problem based on dhouib-matrix-TSP1 heuristic," *International Journal of Computer and Information Technology*, vol. 10, no. 5, pp. 180–183, 2021.

[22] S. Dhouib, "Optimization of travelling salesman problem on single valued triangular neutrosophic number using dhouib-matrix-TSP1 heuristic," *International Journal of Engineering*, vol. 34, no. 12, pp. 2642–2647, 2021.

[23] S. Dhouib, "Minimizing the total distance for the supply chain problem using dhouib-matrix-TSP2 method," *International Journal of Advanced Research in Engineering and Technology*, vol. 12, no. 5, pp. 1–12, 2021.

[24] M. M. DeSousa, P. H. González, L. S. Ochia, and S. L. Martins, "A hybrid iterated local search heuristic for the traveling salesperson problem with hotel selection," *Computers & Operations Research*, vol. 129, pp. 1–16, 2021.

[25] B. Nogueira, E. Tavares, and P. Maciel, "Iterated local search with tabu search for the weighted vertex coloring problem," *Computers & Operations Research*, vol. 125, pp. 1–11, 2021.

[26] V. R. Máximo and M. C. V. Nascimento, "A hybrid adaptive iterated local search with diversification control to the capacitated vehicle routing problem," *European Journal of Operational Research*, vol. 294, no. 3, pp. 1108–1119, 2021.

[27] S. Haddadi, "Iterated local search for consecutive block minimization," *Computers & Operations Research*, vol. 131, pp. 1–11, 2021.

[28] R. A. Melo, M. F. Queiroz, and C. C. Ribeiro, "Compact formulations and an iterated local search-based matheuristic for the minimum weighted feedback vertex set problem," *European Journal of Operational Research*, vol. 289, no. 1, pp. 75–92, 2021.

[29] C. Archetti, D. Feillet, A. Mor, and G. Speranza, "An iterated local search for the Traveling Salesman Problem with release dates and completion time minimization," *Computers & Operations Research*, vol. 98, pp. 24–37, 2018.

[30] V. Cacchiani, C. Contreras-Bolton, J. W. Escobar, L. M. Escobar-Falcon, R. Linfati, and P. Toth, "An iterated local search algorithm for the pollution traveling salesman problem," *New Trends in Emerging Complex Real Life Problems*, AIRO Springer Series, vol. 1, , pp. 83–91, 2018.

[31] E. Santos, L. S. Ochi, L. Simonetti, and P. H. González, "A hybrid heuristic based on iterated local search for multivehicle inventory routing problem," *Electronic Notes in Discrete Mathematics*, vol. 52, pp. 197–204, 2016.

[32] M. M. Silva, A. Subramanian, and L. S. Ochi, "An iterated local search heuristic for the split delivery vehicle routing problem," *Computers & Operations Research*, vol. 53, pp. 234–249, 2015.

[33] J. Michallet, C. Prins, L. Amodeo, F. Yalaoui, and G. Vitry, "Multi-start iterated local search for the periodic vehicle routing problem with time windows and time spread constraints on services," *Computers & Operations Research*, vol. 41, pp. 196–207, 2014.

[34] Y. Deng, J. Xiong, and Q. Wang, "A hybrid cellular genetic algorithm for the traveling salesman problem," *Mathematical Problems in Engineering*, vol. 2021, Article ID 6697598, 16 pages, 2021.