Journal of Advanced Computer Science & Technology, 8 (2) (2019) 32-37

Journal of Advanced Computer Science & Technology

Website: www.sciencepubco.com/index.php/JACST

Research paper

Comparative study between elephant herding optimization (EHO) and U-turning ant colony optimization (U-TACO) in solving symmetric traveling salesman problem (STSP)

Saman M. Almufti [1] *, Ridwan Boya Marqas [2], Renas R. Asaad [1]

[1] College of Computer Science & Information Technology, Nawroz University, Duhok-kurdistan Region, Iraq
[2] Department of Information Technology, Duhok Private Technical Institute Duhok-kurdistan Region, Iraq
**Corresponding author E-mail: Saman.Almofty@gmail.com*

Abstract

Swarm Intelligence is an active area of researches and one of the most well-known high-level techniques intended to generat, select or find a heuristic that optimize solutions of optimization problems.
Elephant Herding optimization algorithm (EHO) is a metaheuristic swarm based search algorithm, which is used to solve various optimization problems. The algorithm is deducted from the behavior of elephant groups in the wild. Were elephants live in a clan with a leader matriarch, while the male elephants separate from the group when they reach adulthood. This is used in the algorithm in two parts. First, the clan updating mechanism. Second, the separation mechanism.
U-Turning Ant colony optimization (U-TACO) is a swarm-based algorithm uses the behavior of real ant in finding the shortest way between its current location and a source of food for solving optimization problems. U-Turning Ant colony Optimization based on making partial tour as an initial state for the basic Ant Colony algorithm (ACO).
In this paper, a Comparative study has been done between the previous mentioned algorithms (EHO, U-TACO) in solving Symmetric Traveling Salesman Problem (STSP) which is one of the most well-known NP-Hard problems in the optimization field. The paper provides tables for the results obtained by EHO and U-TACO for various STSP problems from the TSPLIB95.

Keywords: *Swarm Intelligence (SI), Elephant Herding Optimization (EHO); U-Turning Ant Colony Optimization (U-TACO); Optimization; Symmetric Traveling Salesman Problem (STSP).*

1. Introduction

Generally the complexity of the optimizations problems have been increased in a manner that it becomes difficult for the traditional mathematical programming methods to solve and optimize them [1].
In the past few years, Swarm-based Metaheuristics Optimization Algorithms widely used by researches for optimizing solutions and providing a sufficiently better fittest solutions for objective functions in optimization problems [2,3].
Swarm Intelligence is the field of studying and developing nature-inspired metaheuristics algorithms that inspires the behaviors of real living animal groups such as a clan of elephants, school of fish, colony of ants,… etc. for finding and improving problem solutions [2-5].
This paper compares two swarm based algorithms EHO and U-TACO in solving STSP problem which is one of most known NP-Hard problems, Elephant Herding Optimizations (EHO) algorithm is a metaheuristic swarm based method inspired from the natural herding behaviors of real elephant in their clans [6]. U-Turning Ant colony optimization (U-TACO) is a swarm based algorithm uses the behavior of real ant in finding the shortest way between its current location and a source of food [2,3,4].

2. Symmetric traveling salesman problem STSP

Traveling Salesman Problem (TSP) is a an NP-hard combinatorial optimization problems, which is widely used for testing, improving and evaluating optimizations algorithms [4,6]. In a graph, a complete TSP tour requires visiting all nodes once and return to the starting node [4,6,7]. Symmetric TSP (STSP) is a type of TSP such that in graph G(N,A) the distance d(node i, node j) = d(node j, node i) and the number of possible tours is (n-1)!/2 for n node [6,7]. The fitness of optimal tour can be found by measuring the summation of the distances between nodes as shown in Eq. (1) [4].

$$\mathrm{Btour} = \left(\sum_{i=1}^{n-1} d_{\pi(i)\,\pi(i+1)}\right) + d_{\pi(n)\,\pi(1)} \qquad (1)$$

Where π is a probability list of nodes with minimum distance between city (π and π +1) [6].

3. Swarm intelligence (SI)

Swarm intelligence is the field of Studying, designing, developing and improving intelligent swarm-based systems that cooperate to optain a specific goal [2,3,6]. Swarm intelligence firstly has been determined by Dorigo as "The emergent collective intelligence of groups of simple agents" [8]. Generally all Swarm-based algorithms are inspired from collective behaviors of social living beings that cooperate to achieve a goal [2,6].

different optimization techniques based on SI principles have been inspired from real collective behavior systems in the nature [2,4]including Elephant herding optimizations (EHO) by Wang at 2015, U-Turning Ant colony optimization (U-TACO) by Almufti in 2015 [2,3,4], and other optimization algorithms [5].

4. Elephant herding optimization

Elephant herding optimizations (EHO), is an swarm based metaheuristics search algorithm defined by Wang at the end of 2015 [11] for solving optimization problems. The algorithm models the herding behavior of real elephants in clan [11, 12].

The herding behavior can be summarized as follows:

- Elephants swarms involves a number of sub-swarms, called clans, which consists of Calves and a number of Female elephants [11], as shown in Fig. 1.
- Every clan is under the supervision of a matriarch (adult female) [11] Fig. 1.
- In a clan, whenever a male calf reaches to adulthood, it leave the clan [11-12] as shown in Fig. 2.

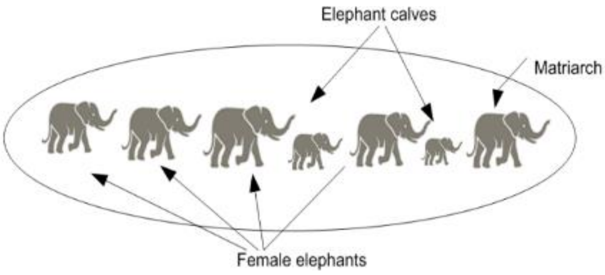

Fig. 1: Elephants Clan.

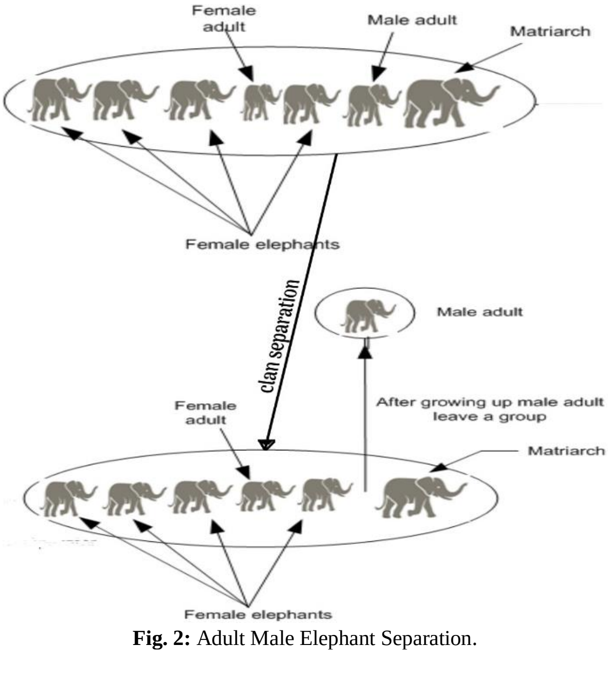

Fig. 2: Adult Male Elephant Separation.

EHO models the herding behaviors in two steps:

a) Clan update.
b) Separation.

a) Clan update:

In each clan, female elephants lives under the supervision of a matriarch. The position of clan elephants are influenced by the matriarch positions, in a way that in ci clan the position of j elephant is updating using Eq. (2) [11,20].

$$E_{new,ci,j} = E_{ci,j} + \alpha\left(E_{best\ ci} - E_{ci,j}\right) r \qquad (2)$$

Where $E_{ci,j}$ represent the old position whereas the $E_{new,ci,j}$ represent the new updated position for j elephant in ci clan, α represent a scale operators ∈ [0, 1] to determine the effect of matriarch ci on $E_{ci,j}$[11]. $E_{best,ci}$ is the matriarch of ci clan [12], and finally r is a type of stochastic distribution ∈ [0, 1] that can improves the diversity of elephant populations in the next search phase [13].
In Eq. (2) the matriarch elephant $E_{best,ci}$ in clan ci is not effected. $E_{best,ci}$ can be updated by Eq. (3).

$$E_{new,ci,j} = \beta E_{center,ci} \tag{3}$$

Where $E_{center,ci}$ represents the center of the ci clan, obtained from the information's achieved by the ci clan elephants, and β is an operator determines the influences of $E_{center,ci}$ on $E_{new,ci,j}$ it ∈ [0, 1]. $E_{center,ci}$ for the d_{th} dimension can be found by Eq. (4).

$$E_{center,ci,d} = \frac{1}{n_{ci}} \sum_{j=1}^{n_{ci}} E_{ci,j,d} \tag{4}$$

Where d_{th} is a dimension between 1 and total number of dimensions $(1 \le d \le D)$. n_{ci} denotes the population number in ci clan. $E_{ci,j,d}$ denotes the d_{th} of elephant $E_{ci,j}$, $E_{center,ci}$ denotes the center of ci clan [12].The clan update operation is shown in Algorithm I[11], [14].

Algorithm I: Clan Updating Operation [11,14].

```
Begin
For ( $Ci = 1 :  nClan$ )
For ($j = 1 :  n_{ci}$ )
Updates ($E_{ci,j}$) & find ($E_{new,ci,j}$) Eq. (2).
If ($E_{ci,j} == E_{best,ci}$)
Updates ($E_{ci.,j}$) & find ($E_{new.,ci,j}$)Eq. (3)
End
 End
 End
```

a) Separation:

In all elephant clans, male elephant leaves the group to live alone after it reaches Adult age. In optimization problems this separating process is called separating operator .In EHO method, the adult male with the worst efficiency separates the clan in each generation using Eq. (5) [11-12], [20].

$$E_{worest,ci} = E_{min} + (E_{max} - E_{min} + 1) * r \tag{5}$$

Where $E_{worest,ci}$ denotes the worst male elephant in the ci clan [12]. E_{min} & E_{max} denotes the Lower & Upper bounds of elephants positions. r is a type of stochastic and uniform distribution ∈ [0, 1] [11,12,13].The separating operation is shown in Algorithm II [12,20].

Algorithm II: Separating Operator [15].

```
Begin
For ($Ci = 1\ to\ nClan$)
Exchange worest elephants individual in $Ci$ Eq. (5).
End
End
```

Based on the EHO operations that are modeled on algorithms I and II, The EHO can be developed as shown in Algorithm III [11].

5. EHO to solve the STSP

The EHO method was defined for solving continued optimization problems in which a number represent the solution [11,12].
STSP is a combinatorial optimization problem in which the solution is denoted by series that can be modeled as a vector [2,3,4].
so EHO that was proposed by Wang cannot be applied to solve STSP directly [16]. In this paper we will adapt EHO operators to solve STSP by respecting the real behavior of elephants as shown bellow:

- Random initialization of elephant position.
- Position of each elephant denotes a node in solution cycle of STSP.
- Distance between each two Elephant positions (E1 – E2) denotes the set permutation (probability) apply to E2 to obtain E1.
- Addition between the set of permutations and a position (E + sp) applies the set of permutations to the E position.

Fig. 3: shows the procedure of Using EHO algorithm in solving STSP.

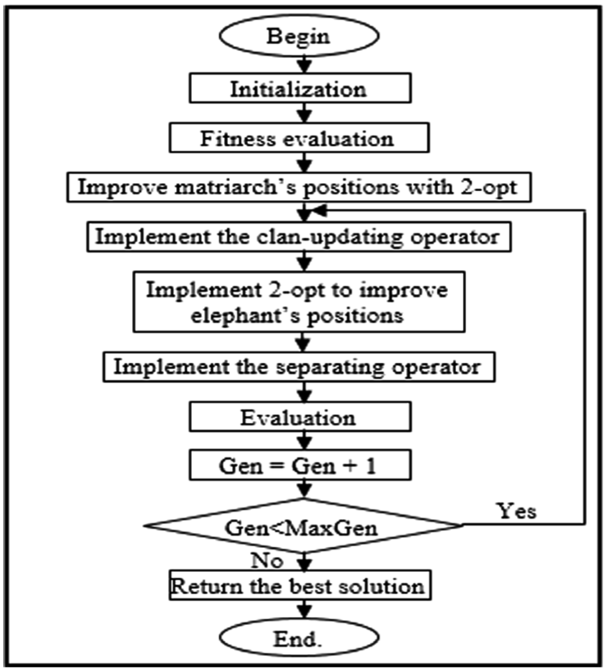
Begin
Initialization
Fitness evaluation
Improve matriarch's positions with 2-opt
Implement the clan-updating operator
Implement 2-opt to improve elephant's positions
Implement the separating operator
Evaluation
Gen = Gen + 1
Gen<MaxGen
Yes
No
Return the best solution
End.

Fig. 3: EHO Flowchart.

6. U-turning ant colony optimization (U-TACO)

U-Turning Ant Colony Optimization (U-TACO) is metaheuristic swarm based algorithm designed by Saman M. Almufti in his master thesis in (2015), U-TACO generally based on the procedure of Ant System (AS) that was designed in 1992 by Marco Dorigo in his PhD thesis as a nature-inspired metaheuristics for solving hard combinatorial optimization (CO) problems [3], [4], [17], [18]
U-TACO algorithm is a modification .of Ant Colony Algorithm that inspires the behavior of Ant in finding food source, basically it depends on updating pheromone trail and the following of other swarm ants to the smell of pheromone: ants deposit pheromones in their way to the food source which takes attention of other swarm ants to the food source and the way that they should take to the food, gradually the way that have more pheromone is the shortest way to the food [3], [4], [17- 18] as shown in Fig. 4.

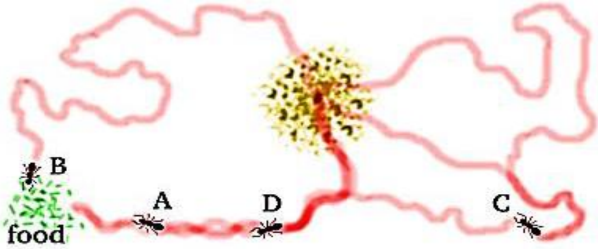

Fig. 4: Behavior of Real Ants.

U-TACO involves two phases:
First Step: initialization of pheromone matrix by U-Ant. U-Ants makes an incomplete tour for the given graph and returns to the start position feedback in the same way that it takes; those partial tours updates the pheromone trail in the path that seems to be useful in obtaining an optimal tour in second step [4, 17].
Second Step: obtaining the best (fittest) tour for the problem by ACO Ants, that makes complete tours as shown in Fig. 5.

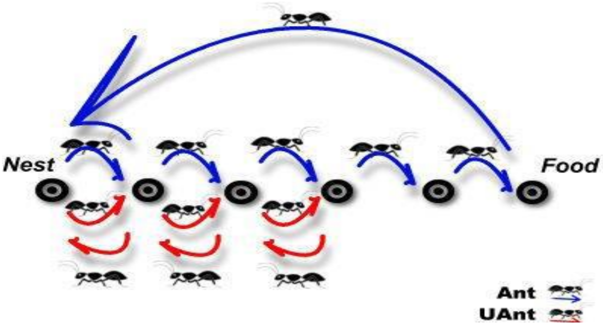

Fig. 5: U-Ant and Ant [17].

U-Ants are not responsible of finding an optimal solution for a given problem, where U-Ant mission is only initializing the pheromone trail matrix, which will be use by ant to construct an optimal solution [4], [17]

7. U-taco to solve the STSP

U-Turning Ant Colony Optimization Algorithm uses Ant and U-Ant in finding the fittest tour for the Salesman. In U-TACO number of Ants equals to the number of cities in the graph, whereas the number of U-Ants ∈ [1, number of cities]. Both Ants and U-Ants follows the same technique in initializing, updating evaporation pheromone trail, the deferens arises in the length and the process of going back to the starting city [4, 17]

U-TACO consist of two steps;

First step is constructing partial tours by U-Ant which consist of the following steps.

1) Initialize the partial tour length l ∈ [1, number of cities].
2) Random allocation of U-Ants in cities:

Based on random proportional rule which is a probability function that depend on pheromone value and heuristic; each U-Ant choses the next city.

$$P_{ij}^{U} = \frac{[\tau_{ij}]^{\alpha}.[\eta_{ij}]^{\beta}}{\sum_{1\in\mathcal{N}_i^U}[\tau_{ij}]^{\alpha}.[\eta_{ij}]^{\beta}} \qquad (6)$$

Eq.(6) shows the probability of U-Ant (U) in city (i) to move to city (j).Where (η_ij) denotes a current heuristic value, the distance between cities i and j is denoted by (dij), (τ_ij) denotes the matrix of pheromone trail, α is a value to regulate the influence of τ_ij, β is a value to regulate the influence of η_ij, and $(\mathcal{N}_i^k)$ is the set of the neighborhood Cities that U-Ant (U) must visit them [17]

3) U-Ant visit l cities.
4) Updating pheromone matrix: A local pheromone update occur once a new city is added to the U-Ant tour as shown in Eq.(7) τ_{ij}new=τ_{ij}old+$\Delta\tau_{ij}$ (7)
5) Pheromone evaporates: at every iteration the pheromone evaporates which leads to gradually decreases of pheromone in the path as shown in Eq.(8).

$$\tau_{ij} new = (1-\psi).\tau_{ij}\text{old} \qquad (8)$$

Where $(0 < \psi < 1)$ is a value used for decreasing pheromone trail.

Once all U-Ants complete their partial tour, the U-TACO first step completes and the second step starts.

In the second step the Ant follow the same procedure (initializing, pheromone trail updating and evaporating) in the as in first step; but here the ant construct a complete tour and directly goes to the initial city as shown in Fig. 5.

Fig. 6, shows the procedure of using U-TACO to solve STSP.

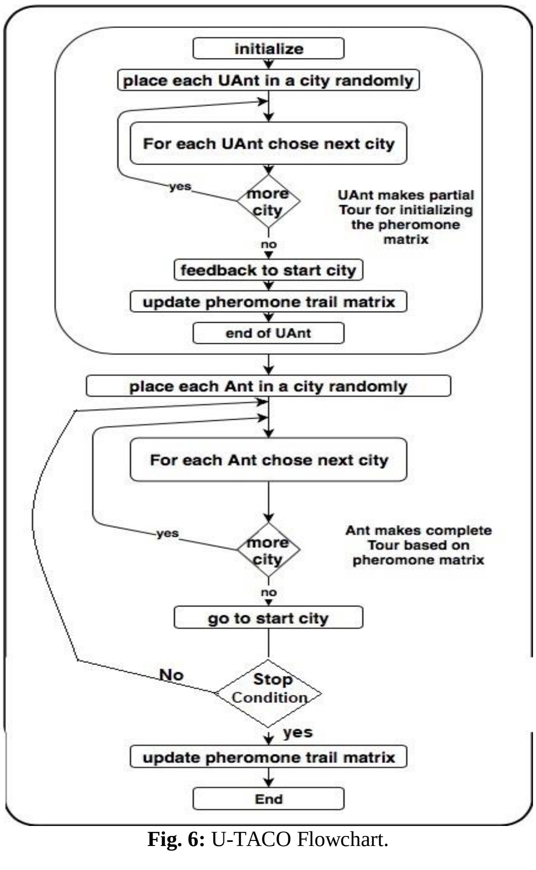
initialize
place each UAnt in a city randomly
For each UAnt chose next city
yes
more city
UAnt makes partial Tour for initializing the pheromone matrix
no
feedback to start city
update pheromone trail matrix
end of UAnt
place each Ant in a city randomly
For each Ant chose next city
yes
more city
Ant makes complete Tour based on pheromone matrix
no
go to start city
No
Stop Condition
yes
update pheromone trail matrix
End

Fig. 6: U-TACO Flowchart.

8. Experimental results

This section presents the comparison of using U-TACO and EHO in solving different STSP problems selected from the TSPLIB library [19] as shown in Table VI.

Table 1: Excremental Results of Using EHO and U-TACO to Solve STSP

STSP problem	Optimal Tour	EHO	U-TACO
$Eil51$	426	426	427
$berlin52$	7542	7542	7542
$St70$	675	675	679
$Eil76$	538	538	538
$Pr76$	108159	108159	108487
$kroA100$	21282	21282	21282
$kroB100$	22141	22141	22200
$kroC100$	20749	20749	20798
$kroD100$	21294	21294	21455
$Eil101$	629	630	629
$Lin105$	14379	14379	14379
$Pr107$	44303	44303	44303
$Pr124$	59030	59030	59030
$Pr136$	96772	96772	100702
$Ch150$	6528	6550	6547
$kroA150$	26524	26524	26618
$kroB150$	26130	26130	26242

In the table of experimental results the best value are written in Bold style, and the table shows that the EHO algorithms is better than U-TACO in finding optimal solutions for STSP.

9. Conclusion

This paper presents a Compassion between two swarm inspired algorithms Elephant herding optimizations (EHO) and U-Turning Ant Colony Optimization (U-TACO) in solving Symmetric Traveling Salesman Problem(STSP) and the results of simulations are compared with the optimal solutions from TSPLIB library.

In the Experimental results both algorithms are adapted to solve same STSP problem in same experimental conditions. Based on the Tables of solutions from 17 STSP problem EHO in 9 problem has a better solution than U-TACO, where as U-TACO has better solution in 2 problem, and they have same solution 6 problem. EHO has reached the optimal solution in 15 problem, U-TACO has reached the optimal solution in 7 problem.

In general the result obtained by applying EHO are fittest than those obtained from U-TACO.

References

[1] X. Yang, "Metaheuristic Optimization" Scholarpedia, 6(8), p.11472, 2011. https://doi.org/10.4249/scholarpedia.11472.

[2] S. Almufti, "U-Turning Ant Colony Algorithm powered by Great Deluge Algorithm for the solution of TSP Problem", Hdl.handle.net, 2018. [Online].

[3] S. Almufti, "Using Swarm Intelligence for solving NPHard Problems," Academic Journal of Nawroz University, vol. 6, no. 3, pp. 46–50, 2017. https://doi.org/10.25007/ajnu.v6n3a78.

[4] S. Almufti and A. Shaban, "U-Turning Ant Colony Algorithm for Solving Symmetric Traveling Salesman Problem", Academic Journal of Nawroz University, vol. 7, no. 4, pp. 45-49, 2018. https://doi.org/10.25007/ajnu.v6n4a270.

[5] J. Rajpurohit, T. Sharma and A. Abraham, "Glossary of MetaheuristicAlgorithms", International Journal of Computer Information Systems and Industrial Management Applications, vol. 9, pp. 181-205, 2017.

[6] R. Asaad and N. Abdulnabi, "Using Local Searches Algorithms with Ant Colony Optimization for the Solution of TSP Problems", Academic Journal of Nawroz University, vol. 7, no. 3, pp. 1-6, 2018. https://doi.org/10.25007/ajnu.v7n3a193.

[7] S. Chibani and A. Tari," Elephant Herding Optimization for Service Selection in QoS-Aware Web", International Journal of Computer and Information Engineering, 2017.

[8] Y. Li, "Solving TSP by an ACO-and-BOA-based Hybrid Algorithm". International Conference on Computer Application and System Modeling, pp. 189–192. IEEE Press, New York ,2010.

[9] M. Dorigo," Optimization, Learning and Natural Algorithms", PhD thesis, Politecnico di Milano, Italy, 1992.

[10] C. Kahraman and G. Kayakutlu, Energy management - collective and computational intelligence with theory and applications. Cham, Switzerland: Springer, 2018. https://doi.org/10.1007/978-3-319-75690-5.

[11] G. Wang, L. Dos Santos Coelho, X. Gao and S. Deb, "A new metaheuristic optimisation algorithm motivated by elephant herding behaviour", International Journal of Bio-Inspired Computation, vol. 8, no. 6, p. 394, 2016. https://doi.org/10.1504/IJBIC.2016.10002274.

[12] G. Wang, S. Deb and L. Coelho, "Elephant Herding Optimization", 2015 3rd International Symposium on Computational and Business Intelligence (ISCBI), 2015. https://doi.org/10.1109/ISCBI.2015.8.

[13] G. Wang, S. Deb, X. Zhao and Z. Cui, "A new monarch butterfly optimization with an improved crossover operator", Operational Research, vol. 18, no. 3, pp. 731-755, 2016. https://doi.org/10.1007/s12351-016-0251-z.

[14] L. Barolli, F. Xhafa and J. Conesa, Advances on Broad-Band Wireless Computing, Communication and Applications. New York: Springer, 2017. https://doi.org/10.1007/978-3-319-49106-6.

[15] P. Suganthan, N. Hansen, J. Liang, K. Deb, Y. Chen, A. Auger and S. Tiwari, "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization", Nanyang Technological University, Singapore, 2005.

[16] 25A. Hossam, A. Bouzidi and M. Riffi, "Elephants Herding Optimization for Solving the Travelling Salesman Problem", Advances in Intelligent Systems and Computing, pp. 122-130, 2019. https://doi.org/10.1007/978-3-030-12065-8_12.

[17] S. Almufti, "U-Turning Ant Colony Algorithm powered by Great Deluge Algorithm for the solution of TSP Problem" (2015). [online] Hdl.handle.net. Available at: http://hdl.handle.net/11129/1734.

[18] M. Dorigo, "Optimization, Learning and Natural Algorithms", PhD thesis, Politecnico di Milano, Italy. (1992).

[19] G. Reinelt, "TSPLIB", Elib.zib.de, 1997. [Online]. Available: http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsplib.html. [Accessed: 18- Feb- 2019].

[20] S. Almufti, R. Asaad and B. Salim, "Review on Elephant Herding Optimization Algorithm Performance in Solving Optimization Problems". International Journal of Engineering & Technology, [S.l.], v. 7, n. 4, p. 6109-6114, may 2019. ISSN 2227-524X. Available at: <https://www.sciencepubco.com/index.php/ijet/article/view/28473/15566>.