IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-1, Jan 2019 E-ISSN: 2347-2693

An Improved Version of Update Pheromone Rule of ACO algorithm for TSP

Ranjeet Savita[1*], Pankaj Sharma[2], Manish Gupta[3]

[1]Dept. Of Computer Science & Engineering, Vikrant Institute of Technology & Management, Gwalior, India

[*]Corresponding Author: ranjeet.savita100@gmail.com, Tel.: 9340673761

Available online at: www.ijcseonline.org

Accepted: 29/Dec/2018, Published: 31/Jan/2019

Abstract— Ant colony optimization algorithm is a popular meta-heuristic optimization algorithm that has been proven successful for solving travelling salesman problem. In this paper, modified version of ant colony optimization for solving travelling salesman problem has been proposed. In this modified version, update pheromone phase of ant colony optimization algorithm is updated. Here, best distance is calculated by comparing all the nodes distance and taken the best distance for find next node instead of taking ants one by one and keep updating later on. This modified version improves the total cost as well as total time of travelling salesman problem. Proposed algorithm is performed on 51 cities, 61 cities, 70 cities and 76 cities problem. Comparative study shows that proposed algorithm is better than standard ant colony optimization algorithm.

Keywords— Ant colony optimization, Travelling salesman problem, ACO, TSP, Update Pheromone Phase

I. INTRODUCTION

Introduction In Today's scenario, Optimization is needed in every sector of research. Various optimization algorithms have been proposed for solving different kind of problems like task scheduling problem, graph colouring problem, travelling salesman problem, problems related to mechanical engineering, problems related to electrical engineering and also problems related to social networks etc. Here ant colony optimization algorithm is used to improve the total cost as well as total time for solving travelling salesman problem.

Ant colony optimization algorithm is one of the best meta-heuristic algorithms that simulate the foraging behaviour of real ants that consistently optimize their path from their next to food.

L. Shufen, Et. al. [1] proposed Pheromone Model Selection in Ant Colony Optimization for the Travelling Salesman Problem. here two pheromone models, named as first order pheromone model and second order pheromone model are used and then compared and analysed. D. chitty [2] proposed a new improved algorithm that is used for solving large scale TSP problems. ACO algorithm can also be used to solve social network problem based on travelling salesman problem [3]. Z. A. Aziz [4] proposed ant colony algorithms based on generalized heuristic method, where two updating procedures (Local and Global) are used for solving travelling salesman problem. Various other version of modified ACO [5, 6, 7, 8] are also used for solving TSP.

The overall work in this paper is summarized as follows: Section 1 gives the introduction and also previous work that has been proposed for solving Travelling salesman problem. Section 2 introduces standard ACO algorithm for solving Travelling salesman problem. Section 3 gives the proposed ACO algorithm for solving Travelling salesman problem. Experimental results and analysis on various Cities is described in section 4. Section 5 concludes the overall work.

II. ANT COLONY OPTIMIZATION ALGORITHM FOR TSP

The travelling salesman problem (TSP) is a NP-complete problem where salesman has to travel every city once and reach to the starting city with minimum distance. TSP can be represented by a complete connected weighted graph and the objective to find a Hamiltonian cycle of minimum cost (distance).

The working principle of Ant colony algorithm, as a heuristic algorithm, is to simulate the foraging behaviour of real ants, where they will search the food on the basis of pheromone left by other real ants. In the TSP problem, the ants are randomly divided into nodes because each node can be accessed only once.
To access the next node, the following probability formula is used:

$$p_{ij}^{k}(t) = \begin{cases} \frac{[\tau_{ij}(t)]^{\alpha}[n_{ij}]^{\beta}}{\sum_{k\in allowed_k}[\tau_{ij}(t)]^{\alpha}[n_{ij}]^{\beta}} & \text{if } j \in a \\ 0 & \text{otherwise} \end{cases}$$

the pheromone trail values are updated according to following formula, After the ants completed their tours:

(t+n) =

Where = pheromone decrease parameter

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}$$

Where = quantity of per unit length of pheromone trail laid on edge (i, j). And calculated as

= if k^{th} ant uses (i, j) in tour

Here Q is constant and L_k= tour length of k^{th} ant.

The algorithm steps for solving TSP using ant colony algorithm is as follows:

1. Loop
2. Place m artificial ants on n cities randomly
3. For each city 1 to n
4. For each ant 1 to m
5. Each ant builds a solution by adding one city after the other city.

Parameters	Values
Alpha (α)	1
Beta (β)	2
Pheromone decrease factor [rho(σ)]	0.1
Pheromone increase factor [Q]	0.7
Number of Ants	51, 61, 70, 76

6. Select next city according to the probability equation
7. Apply local pheromone update
8. End for
9. End for
10. Apply Global pheromone update using best ant
11. Until end condition not reached.

III. PROPOSED METHODOLOGY

The proposed ant colony algorithm is based on the enhancement of updating pheromone rule. The enhanced version of update pheromone rule is given below, rest of the steps are same as original ACO algorithm for TSP. Here, best distance is calculated by comparing all the nodes distance and taken the best distance for find next node instead of taking ants one by one and keep updating later on. This modified version improves the total cost as well as total time of travelling salesman problem.

Enhanced Version of Update Pheromone Rule:

```
for (int i = 0; i < pheromones.Length; ++i)
{
for (int j = i + 1; j < pheromones[i].Length; ++j)
{
for (int k = 0; k < (ants.Length)-1; ++k)
{
double length = Length(ants[k], dists);           double
decrease = (1.0 - rho) * pheromones[i][j];
double increase = 0.0;
double bestdist=0.0;

if (EdgeInTrail(i, j, ants[k]) == true) increase = (Q /
length);
if(Length(ants[k],dists)<=Length(ants[k+1],dists))
bestdist=Length(ants[k],dists);
else
bestdist=Length(ants[k+1],dists);
pheromones[i][j] = decrease + increase+bestdist;

}
}
}
```

The proposed algorithm is performed on a system having 2 GB RAM, Core i3 processor. The following parameters are used to perform experiments on different number of cities [51 cities, 61 cities, 70 cities and 76 cities] for solving travelling salesman problem.

Table 1 shows the experimental results performed on 51 cities travelling salesman problem.

Algorithms	Total cost
Standard ACO algorithm	432
Proposed Algorithm	389

Standard ACO path:

4 43 3 6 17 5 18 28 27 23 14 21 37 12 35 40 30 46 11 42 50 32 44 49 41 25 8 2 39 13 9 20 38 34 16 22 24 29 36 47 10 19 26 15 48 33 31 0 1 45 7

Proposed Algorithm path:
8 0 5 36 17 37 48 35 26 22 50 7 46 6 9 44 33 3 20 21 47 32 28 31 34 1 19 43 45 49 14 4 29 15 24 42 13 38 25 2 10 11 23 30 18 12 27 39 41 16 40

Table 2 shows the experimental results performed on 61 cities travelling salesman problem.

Algorithms	Total cost
Standard ACO algorithm	621
Proposed Algorithm	585

Standard ACO path:
24 36 28 38 51 19 52 26 40 55 2 17 32 33 45 42 0 13 29 35 12 46 53 15 8 4 54 48 11 14 20 37 34 44 1 5 27 25 41 21 58 57 43 31 59 56 30 18 60 9 22 49 6 47 16 39 23 7 3 50 10

Proposed Algorithm path:
14 15 22 42 24 44 6 39 53 23 17 10 4 2 60 47 11 1 51 35 55 18 32 52 5 31 38 45 21 59 20 13 28 19 26 0 57 41 27 12 8 40 37 29 48 56 54 50 43 33 34 3 16 36 46 58 30 25 49 9 7

Table 3 shows the experimental results performed on 70 cities travelling salesman problem.

Algorithms	Total cost
Standard ACO algorithm	504
Proposed Algorithm	464

Standard ACO path:
17 26 63 40 38 7 52 10 31 57 47 59 29 49 41 60 43 0 22 37 34 4 39 19 53 8 66 6 25 64 51 69 30 42 23 16 5 2 20 46 45 54 36 3 35 44 48 62 33 61 12 28 14 55 15 13 65 27 67 24 11 50 56 18 1 58 21 68 9 32

Proposed Algorithm path:
34 50 40 22 32 53 6 63 30 9 36 65 20 14 42 64 18 15 52 58 44 5 62 68 54 13 45 43 0 31 35 41 21 10 27 55 59 8 33 16 26 61 25 69 12 23 29 2 51 4 38 48 66 47 28 37 49 39 19 17 3 46 67 60 56 57 11 1 24 7

Table 4 shows the experimental results performed on 76 cities travelling salesman problem

Algorithms	Total cost
Standard ACO algorithm	596
Proposed Algorithm	534

Standard ACO path:
51 8 33 72 22 17 16 52 53 71 69 38 7 27 61 57 10 14 55 3 63 19 23 11 26 50 75 30 20 70 15 47 64 65 67 36 68 54 9 29 18 58 66 62 60 35 59 25 4 42 56 12 44 48 1 31 46 24 40 39 6 73 32 45 41 13 49 2 34 21 43 5 74 28 37 0

Proposed Algorithm path:
31 40 29 58 23 41 39 43 69 75 19 30 38 56 2 57 13 50 68 63 6 37 26 5 28 21 73 11 7 33 55 65 8 9 62 32 42 66 36 61 59 18 44 60 70 45 72 74 20 25 3 35 24 46 10 53 12 16 15 71 49 14 47 48 17 34 27 22 67 51 1 4 0 52 54 64

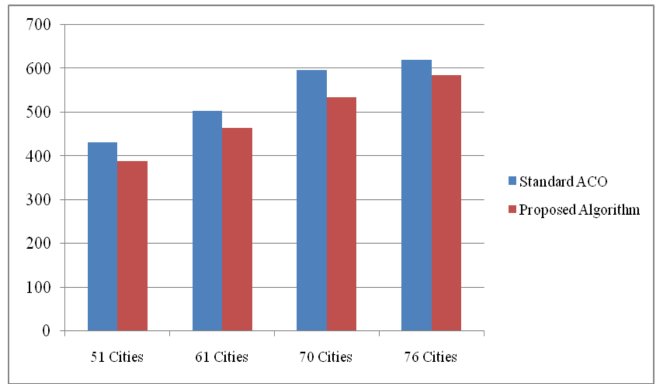

Figure 1 shows the overall analysis of all travelling salesman problems

V. CONCLUSION AND FUTURE SCOPE

Ant colony optimization algorithm is one of the optimization algorithms for solving travelling salesman problem efficiently. Here in this work, update pheromone phase of basic ant colony algorithm is updated, that improves the total cost as well as running time of an algorithm. Proposed algorithm is implemented on different cities to check the efficiency of an algorithm. Experimental results show that proposed algorithm is better than standard ACO algorithm. Future work is to implement this proposed algorithm on large TSP problems and also check the efficiency of proposed algorithm with the other modified versions of ACO algorithm.

REFERENCES

[1] L. Shufen, L. Huang and H. Lu," Pheromone Model Selection in Ant Colony Optimization for the Travelling Salesman Problem", Chinese Journal of Electronics, Vol.26, No.2, Mar. 2017.
[2] D. M. Chitty," Applying ACO to Large Scale TSP Instances," UK Workshop on Computational Intelligence, pp. 104-118. Springer, Cham, 2017.
[3] N. Xiong, W. Wu and C. Wu," An Improved Routing Optimization Algorithm Based on Travelling Salesman Problem for Social Networks" Sustainability 2017, 9, 985; doi:10.3390/su9060985.
[4] Z. A. Aziz," Ant Colony Hyper-heuristics for Travelling Salesman Problem", IEEE International Symposium on Robotics and Intelligent Sensors (IRIS 2015), Procedia Computer Science 76 (2015) 534 – 538.

[5] Jiang, Y. ,"The Application of an Improved Ant Colony Optimization for TSP", South-central University for Nationality: Wuhan, China, 2009.
[6] Chen, W.; Jiang, Y.," Improving ant colony algorithm and particle swarm algorithm to solve TSP problem", Inf. Technol. 2016, 2016, 162–165.
[7] Wang, Z.; Bai, Y.; Yue, L.," An Improved Ant Colony Algorithm for Solving TSP Problems", Math. Pract. Theory 2012, 42, 133–140.
[8] Sun, J.," Research on Ant Colony Algorithm for Solving Travelling Salesman Problem", Wuhan University of Technology: Wuhan, China, 2005.

Authors Profile

Mr Ranjeet Savita pursued master pf technology in computer science & engineering from Vikrant Institute of Technology & Management, Gwalior. His main research area is soft computing.

Mr. Pankaj Sharma is working as a assistant professor in Vikrant Institute of Technology & Management, Gwalior. He has completed his M.Tech degree in computer science & engineering stream. He has published more than 15 research papers in various journals and conferences. He has having 8+years of teaching and research experiences.

Mr. Manish Gupta is working as a assistant professor in Vikrant Institute of Technology & Management, Gwalior. He has completed his M.Tech degree in computer science & engineering stream. He has published more than 20 research papers in various journals and conferences. He has having 10+years of teaching and research experiences.

