Modified Ant Colony Optimization with Pheromone Mutation for Travelling Salesman Problem

C. Ratanavilisagul, "Modified Ant Colony Optimization with pheromone mutation for travelling salesman problem," 2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Phuket, Thailand, 2017, pp. 411-414, doi: 10.1109/ECTICon.2017.8096261.

Chiabwoot Ratanavilisagul

Department of Technology Computer Faculty of Technical Education

Rajamangala University of Technology (RMUTK)

Bangkok, Thailand

Chaibwoot@hotmail.com

***Abstract*—Ant Colony Optimization (ACO) algorithm is a stochastic algorithm that is used for solving combinational optimization problem. It is inspired by the foraging behavior of ant colony. The ant colony walks along density of pheromone from ant's nest to feeding sources. It leads to create shortest path from ant's nest to feeding sources. ACO is normally troubled with the problems of trapping in local optimum. This paper proposed an improved ACO algorithm by mutation is applied with pheromone of ants when ant colony traps in local optimum. The proposed technique is tested on twenty-two maps from the Traveling Salesman Problem Library (TSPLIB) and gives more satisfied search results in comparison with ACOs.**

Keywords- ant colony optimization; mutation operator; Travelling salesman problem; optimization

I. INTRODUCTION

Dorigo M. [1] introduced Ant Colony Optimization algorithm (ACO) in 1991. It is motivated by the foraging behavior of ant colony [2], [3], [4]. ACO has been widely used for solving optimization problems [5] such as job scheduling problem, traveling salesman problem (TSP), network routing and vehicle routing problem etc.

ACO attempts to search for shortest paths from ant's nest to feeding sources. Each ant walks along density of pheromone and releases pheromone among walking. This action causes the pheromone that has much density in short paths. On the other hand, pheromone has a little density in long paths [6], [7].

The advantages of ACO [5], [8], [9], [10], [11] are rapid discovery of good solutions, efficient for Traveling Salesman Problem (TSP) and similar problems. However, disadvantages of ACO [5], [8], [9], [12] are long interval in operation and easily chances on trapping in the local optimum.

To overcome disadvantages of ACO, many researchers [13], [14], [9], [15] increased searching diversity in the ant colony of ACO by adding the mutation technique in the process of ACO. The experiment results of these researches showed that mutation techniques can increase searching performance of standard ACO and obtain better solutions than standard ACO.

Many algorithms such as Particle Swarm Optimization (PSO) [16], [17] add the mutation technique to improve searching performance.

The experiment results of these researches showed that mutation techniques can increase searching performance and obtain better solutions.

Reference [9] proposed mutation strategy is similar to mutation operator of the Genetic Algorithms (GA). The mutation strategy can enhance diversity of ant colony. Reference [14] proposed mutation strategy apply to ACO. This algorithm is called that ant colony optimization algorithm with mutation mechanism (MACO). This algorithm, mutation applied with the best ant. Then, if the result from mutation is better than the best ant, the best ant is replaced by it. Reference [13] proposed mutation strategy apply to ACO. This algorithm is called that ant colony optimization algorithm with uniform mutation using self-adaptive approach (ACOMS). This algorithm, mutation applied with some ants. Then, if the result from mutation is better than the mutated ant, the mutated ant is replaced by it. From the experiment results show the performance of MACO and ACOMS in solving combinatorial optimization problems is better than that of the previous modified ACO.

However, the weak point of ACOMS and MACO, number of evaluation calls of them per an iteration are more than that of ACO. If number of evaluation call is equal, ACOMS and MACO may obtain solutions worse than ACO. Moreover, the results from mutation cannot guide searching in next iteration. Hence, mutation technique which is added seldom increase diversity of ant colony.

To increase diversity of ant colony in next iteration and constant number of evaluation call per an iteration. This paper proposed a novel mutation technique apply with ACO to solve TSP. The proposed technique, mutation are applied with pheromone of ants when ant colony traps in local optimum. This technique can increase diversity of ant colony but it does not increase number of evaluation call when it is compared by standard ACO.

A set of maps in the TSPLIB [18], [19] is used to compare the standard ACO [6] by source code from [20], MACO [14], ACOMS [13], and the proposed technique. The results show that the quality of solutions of the proposed technique is better than other compared algorithms in the TSPLIB.

The rest of this paper is organized as follows. Section 2 explains TSP, the basic ACO, the mutation operator, and ACO with mutation. Section 3 explains modified ACO with pheromone mutation (the proposed PSO algorithm). Section 4 explains the case studies of TSP, the experiment setup and presents the experiment results. Section 5 concludes the paper with a brief summary.

II. Related Work

A. Travelling Salesman Problem

Traveling salesman problem (TSP) [21] is well-known combinational optimization problems and an NP-hard problem. The TSP is the problem of finding a minimal cost closed path that visits each city once.

A complete weighted graph G = (V, A) can be used to represent a TSP. Where V is the set of n cities and A is the set of paths fully connecting all cities. Each edge (d_{ij}) is the distance between cities i and j. d_{ij} is as follows:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{1}$$

V = {1,. ., n}, A = {(i, j): i, j ∈ V}, d(i, j) = d(j, i), Edge (i, j) ∈ A

B. Ant colony optimization

This paper shows applying ACO with TSP. An ant (A) is the set of paths fully connecting all cities. The fitness function or the objective function of TSP is as follows:

$$fitness\ of\ ant = \sum_{i=1}^{n-1} d_{i,i+1} + d_{n,1} \tag{2}$$

ACO can be described shortly as follow. Initially, each edge has an initial pheromone τ_{ij} (0) between two cities. The next step is select city of ant. The first city of each ant is randomly selected, and then each ant selects next city according to probability function as follows:

$$P_{i,j}^{k} = \begin{cases} \dfrac{[\tau_{i,j}(t)]^{\alpha}[\eta_{i,j}]^{\beta}}{\sum_{t\epsilon allowed}[\tau_{i,t}(t)]^{\alpha}[\eta_{i,t}]^{\beta}} & , j \epsilon\ allowed \\ 0 & , otherwise \end{cases} \tag{3}$$

Where $P_{i,j}^{k}$ is the probability of ant k chooses to move from city i to city j, τ_{ij} is the pheromone, $\eta_{ij} = 1/d_{ij}$ is the inverse of the distance, β is a parameter which determines the relative importance of pheromone versus distance (β > 0). The result from formula (3) that causes selecting edges which are shorter and have a greater amount of pheromone. After selection cities of ant has completed, the fitness of ant is calculated by (2). The fitness of each ant is used to update pheromones according to (4). The next step, repeat this until the stop condition is reached.

$$\Delta\tau_{ij}(t+n) = \rho\tau_{i,j} + \Delta\tau_{ij} \tag{4}$$

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \tau_{i,j}^{k} \tag{5}$$

Where $\Delta\tau_{ij}$ is the sum of new enhanced pheromones at this edge is degree of dissipate for pheromones, τ_{ij}^{k} is the amount of pheromones of the k^{th} ant at its edge between t and t+n, ρ is a coefficient such that (1 - ρ) represents the evaporation of trail between time t and t+n. Q is a const, L_k is fitness of ant k. $\Delta\tau_{ij}^{k}$ is calculated by (6).

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{L_K}, & if\ the\ k^{th}\ ant\ uses\ edge\ (i,j) in\ its\ tour \\ 0, & otherwise \end{cases} \tag{6}$$

C. Ant colony optimization with mutation

Reference [14] proposed MACO. This paper uses concept of simple mutation operation of Genetic Algorithm (GA) [22], [23]. The main concept of MACO is can be summarized as follows: After update pheromone step, randomly choose one or more elements in the local best solution (S_{nbest}), change them. The result from mutation is called that S'_{nbest}. If S'_{nbest} is better than S_{nbest} , replace S_{nbest} by S'_{nbest}.

Reference [13] proposed ACOMS. The main concept of MACO is can be summarized as follows: The mutation operator is applied with ant. The new ant is generated from mutation. If the fitness of the new ant is better than that of ant, replace ant with the new ant. From the experiment results show that both ACOMS and MACO have better performance in solving TSP than the previous modified ACO algorithms.

However, the weak point of ACOMS and MACO are using over evaluation call because ants of these techniques are updated by mutation technique and ACO technique in each iteration. Each iteration, they can guess solutions more than ACO. Hence, they can obtain better solutions than ACO. On the other hand, if amount of evaluation calls of ACO, ACOMS and MACO are the same amount, ACOMS and MACO may obtain worse solutions than ACO in some maps. Moreover, the mutation cannot change guessing of solution in next iteration. Hence, Ant colony has still moved along previous path. They are hardly distributing to search for new area. Hence, the diversity of ant colony is hardly increased.

III. Modified ACO with the Pheromone Mutation

As previously mentioned the main problem of applying mutation technique is using over evaluation call and hardly increase diversity of ant colony. To avoid these problems, this paper proposed applying mutation with pheromone because pheromone causes distribution of ant colony. Moreover, the proposed algorithm does not increase evaluation call.

The trapping in local optimum is possible for ACO, which leads to the stagnation of the searching in which the solution obtained at the point of trapping, is repeatedly produced irrespective of the length of searching time. Therefore, the unchanged number of consecutive solution can indicate state of ant colony when ant colony traps in local optimum. Normally, the mutation should be used when colony trap or trend to trap in local optimum [16], [17] to distribute ant colony to search in other areas. The easy techniques to indicate the trapping state of the ant colony is to monitor the unchanged number of the best consecutive solution. Thereby, this paper proposed the mutation period considered from the unchanged number of consecutive solution.

In begin time of searching, pheromone occurs very changing. When time passes, pheromone decreases changing. Finally, pheromone hardly occurs changing. It causes to create solutions which are similar to previous solutions. The obtained solutions are not improved or Ant colony trapped in local optimum. This event should take the previous pheromones, which still have diversity of ant colony, replace the current pheromones in some part to enhance diversity of ant colony and chance of selection new paths. This technique is called that mutation operation. The pheromones that are replaced should cause to obtain the better solution because this pheromone can find the better solution. The number of mutated pheromones or the mutation rate depends on amount of cities. The mutation rate is one-third of number of all cities. Therefore, this paper proposed the novel mutation technique which can increase performance searching solutions of ACO and get better solutions. This technique is not increase evaluation calls but it can increase diversity of population Moreover, the results from mutation can affect to search of the next iteration. It is called modified ACO with Pheromone Mutation (ACOPM). Pseudo code of ACOPM is shown below:

```
Initialize edges (i, j) and pheromone (τij)
While termination condition ≠ true do
    For each number of ants
        Random the start city
        For until cites in list of an ant is full
            Choose the next city with probability according to formula (3)
            The chose city is not repeat city in list of this ant
            Insert chose city into list of this ant
        End For
        Evaluate fitness of each ant
        If fitness of each ant is better than fitness of solution
            Update fitness of solution = fitness of this ant
            Save pheromones (τ'ij)
        End If
        If fitness of each ant is better than that of the best solution
            Update fitness of the best solution = fitness of this ant
        End If
    End For
    For every edges on the pheromone do
        Update pheromone according to formula (4)
    End For
    Times of solution consecutive unchanged ++
    If the times of solution consecutive unchanged >= mutation period (MP)
        For number of mutation
            Random edges (i, j) which is not repeat random previous edges.
            pheromone (τij) = pheromones (τ'ij)
        End for
        Reset solution
        Set the times of solution consecutive unchanged to 0
    End If
End While
```

IV. Experiments and Results

TABLE I. COMPARATIVE RESULTS OF ACO, MACO, ACOMS, AND ACOPM ON TSP.

Techniques		**ACO**			**MACO**		
Problem name	**optimum point**	**MBF**	**SD**	**SR**	**MBF**	**SD**	**SR**
burma14	3323	3323.00	0.00	50	3323.00	0.00	50
ulysses16	6859	6859.00	0.00	50	6859.00	0.00	50
gr17	2085	2085.00	0.00	50	2085.00	0.00	50
gr21	2707	2707.00	0.00	50	2707.00	0.00	50
ulysses22	7013	7038.48	6.34	0	7018.84	7.00	18
fri26	937	937.00	0.00	50	937.00	0.00	50
bayg29	1610	1657.48	11.18	0	1662.36	19.39	0
bays29	2020	2069.80	15.77	0	2045.92	14.64	0
Oliver30	420	420.88	1.34	34	420.20	0.49	41
dantzig42	699	724.52	14.24	2	715.52	9.18	0
swiss42	1273	1331.40	13.72	0	1294.60	21.42	9
gr48	5046	5320.28	60.41	0	5355.16	78.66	0
att48	10628	11319.10	118.62	0	11429.20	139.82	0
eil51	426	455.08	7.09	0	448.56	7.79	0
berlin52	7542	7962.80	116.52	0	8001.80	143.08	0
brazil58	25395	26873.40	237.86	0	26509.10	314.21	0
st70	675	745.20	10.94	0	731.20	9.50	0
pr76	108159	119483.00	1698.26	0	118282.00	1657.98	0
eil76	538	594.60	9.67	0	588.68	12.71	0
rat99	1211	1407.36	42.78	0	1405.28	26.64	0
eil101	629	739.88	10.38	0	728.84	12.78	0
lin105	14379	15963.40	245.10	0	15750.90	277.74	0

TABLE I. (CONTINUED).

Techniques		**ACOMS**			**ACOPM**		
Problem name	**optimum point**	**MBF**	**SD**	**SR**	**MBF**	**SD**	**SR**
burma14	3323	3323.00	0.00	50	3323.00	0.00	50
ulysses16	6859	6859.00	0.00	50	6859.00	0.00	50
gr17	2085	2085.24	0.81	45	2085.00	0.00	50
gr21	2707	2707.00	0.00	50	2707.00	0.00	50
ulysses22	7013	7049.64	28.65	8	7013.00	0.00	50
fri26	937	937.00	0.00	50	937.00	0.00	50
bayg29	1610	1672.92	18.13	0	1622.60	9.12	11
bays29	2020	2071.60	27.18	0	2033.05	7.95	4
Oliver30	420	422.24	1.94	11	420.00	0.00	50
dantzig42	699	734.04	10.15	0	705.80	4.67	5
swiss42	1273	1336.68	20.09	0	1279.00	8.66	21
gr48	5046	5399.20	72.21	0	5248.50	38.33	0
att48	10628	11521.20	176.76	0	11192.00	87.83	0
eil51	426	456.00	9.07	0	439.20	5.60	3
berlin52	7542	8073.16	156.79	0	7817.20	68.33	0
brazil58	25395	27162.40	333.83	0	26401.80	187.75	0
st70	675	752.24	12.77	0	723.60	7.74	0
pr76	108159	120480.00	1673.55	0	115730.00	1106.49	0
eil76	538	596.48	11.90	0	572.60	4.22	0
rat99	1211	1434.40	29.98	0	1355.60	23.54	0
eil101	629	744.20	13.62	0	716.80	4.71	0
lin105	14379	16167.60	403.70	0	15407.80	141.32	0

A. Parameters Setting

Parameters are as follows for all experiments: $\beta = 1$, $Q = 1$, $\alpha = 1$, $\rho = 0.05$ as suggested by [20], $\tau_0 = (n \times L_{nn})^{-1}$ where L_{nn} is the length of list of ant that is produced by the nearest neighbor heuristic [24] and n is the number of cities. The number of ants used is 20. The number of experiments of each map is 50 runs. To guarantee fairness of performance measurement, evaluation calls of all the algorithms are set equation. The number of evaluation of each run in experiments is 100,000 evaluations. The non-ACO parameters are as follows: The compared algorithm parameters (MACO, ACOMS) are set according to suggested by the original papers. For proposal algorithm, mutation period = 50. This research is conducted by a personal computer of Intel Core i5 2450 with a 2.5-GHz CPU and 8 GB RAM. All experimented algorithms are not used by well known path improvement heuristics such as 2-opt [25], and Lin–Kernighan [26]. All maps are used in experiment are from TSPLIB [18], [19].

B. Experiment of proposed algorithm

The measures of algorithm performance in the experiments are as follows: The mean best fitness value (MBF) is the mean of best fitness in the final iteration from all running (50 runs). MBF indicates the solution searching efficiency of an algorithm. The closer the MBF to the optimum point of a method, the better the method. SD is the standard deviation. SR is the success round. SR is the number of running rounds that yields the optimum solution. The higher the SR is, the better the algorithm is. SR and SD indicate the solution searching reliability of an algorithm.

From the experimental results of Table 1 show that the quality solution of ACOPM is better than that of ACO, MACO, and ACOMS because of its lowest MBF all tested maps. The reliability of ACOPM is better than that of ACO, MACO, and ACOMS because of its lowest SD in all tested maps. Moreover, SR of ACOPM is better than that of ACO, MACO, and ACOMS in all tested maps. For small-scale TSP problems (less than 26 cities), ACOPM can locate the optimum points of all tested maps. For medium-scale and large-scale TSP problems (more than 26 cities), ACOPM cannot locate the optimum points but both quality solution and reliability are better than compared other algorithms. Hence, ACOPM outperforms ACO, MACO, and ACOMS in case of reliability and quality solution. In some maps, MBF and SD of MACO and ACOMS are higher than that of ACO, such as BAYG29, GR48, and ATT48. In this case show that if number of evaluation call is equal, MACO and ACOMS may obtain worse solutions than ACO in some cases.

V. CONCLUSION

The mutation operator is applied with ACO to enhance diversity of ant colony and solve trapping in local optimum. Normally, the mutation operation is applied with ant. These techniques cause number of evaluation calls that are increased. Moreover, these techniques rarely increase diversity of ant colony. This paper proposed mutation operation is applied with pheromone. The pheromone affects distribution of ant colony and improves solutions. Moreover, mutation is applied when ant colony trapped in local optimum. The proposed technique is called that ACOPM. ACOPM can improves solutions by ACOPM can enhance diversity of ant colony. In addition, ACOPM do not enhance evaluation calls. From the experimental results show that the proposed ACOPM outperforms ACO, MACO, and ACOMS with regard to the reliability and quality of solutions in all maps.

REFERENCES

[1] A. Colorni, M. Dorigo, V. Maniezzo, “Distributed optimization by ant colonies,” Proceedings of European Conference on Artificial Life, Paris, France, pp. 134-142, 1991.

[2] J. L. Denebourg, J. M. Pasteels, and J. C. Verhaeghe, “Probabilistic behavior in ants: A strategy of errors?,” J. Theoret. Biol., vol. 105, pp. 259-271, 1983.

[3] J. L. Denebourgand S. Goss, “Collective patternsand decision-making,” Ethology, Ecology & Evolution, vol. 1, pp. 295-311, 1989.

[4] S. Goss, R. Beckers, J. L. Denebourg, S. Aron and J. M. Pasteels, “How trail laying and trail following can solve foragingproblems for ant colonies,” in Behavioral Mechanisms of Food Selection, R. N. Hughes Ed., NATO-AS1 Series. Berlin: Springer-Verlag, vol. 20, 1990.

[5] V. Selvi, and R. Umarani, “Comparative Analysis of Ant Colony and Particle Swarm Optimization Techniques,” International Journal of Computer Applications, vol. 5., pp. 1–6, August 2010.

[6] M. Dorigo, V. Maniezzo, A. Colorni, “Ant system: optimization by a colony of cooperating agents,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 26, pp. 693–698, pp.1-13, February 1996.

[7] M. Dorigo, L.M. Gambardella, “Ant colony system: a cooperative learning approach to the traveling salesman problem,” IEEE Transactions on Evolutionary Computation, vol 1, pp. 53 - 66, April 1997.

[8] Y. Zhang, Z. Pei, J. Yang, Y. Liang, “An Improved Ant Colony Optimization Algorithm Based on Route Optimization and Its Applications in Travelling Salesman Problem,” IEEE, BIBE, pp. 693–698, 2007.

[9] X. Song, B. Li, and H. Yang, “Improved Ant Colony Algorithm and its Applications in TSP,” Sixth International Conference on Intelligent Systems Design and Applications, october 2006.

[10] M. Dorigo and, LM. Gambardella, “Ant colony system: A cooperative learning approach to the traveling salesman problem,” IEEE Transactions on Evolutionary Computation, pp. 53–66, 1997.

[11] M. Dorigo and, LM. Gambardella, “A study of some properties of ant-Q . In: Voigt H-M, Ebeling W, Rechenberg I, Schwefel H-S, eds. Proceedings of the 44th International Conference on Parallel Problem Solving from Nature,” Berlin: Springer-Verlag, pp. 656–665, 1996.

[12] A. Colorni and, M. Dorigo, “Heuristics from nature for hard combinatorial optimization problems,” International Trans Operational Research, pp. 1–21, 1996.

[13] R. S. Jadon, U. Datta, “Modified ant colony optimization algorithm with uniform mutation using self-adaptive approach for travelling salesman problem,” Computing, Communications and Networking Technologies, 2013.

[14] N. Zhao, Z. L. Wu, Y. Q. Zhao, and T. F. Quan, “Ant colony optimization algorithm with mutation mechanism and its applications,” Expert Syst. Appl. , vol. 37, pp. 4805-4810, July 2010.

[15] Q. WU, and J. ZHANG, “ACO With the Characteristic of Mutation”, Computer research and development, pp. 240-245, 1999.

[16] C. Ratanavilisagul, and B. Kruatrachue, “A modified particle swarm optimization with dynamic mutation period,” Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, July 2014.

[17] C. Ratanavilisagul, and B. Kruatrachue, “A modified particle swarm optimization with mutation and reposition,” International Journal of Innovative Computing, Information and Control, vol. 10, pp. 2127–2142, December 2014.

[18] http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95

[19] http://stevedower.id.au/blog/research/oliver-30

[20] http://yarpiz.com/53/ypea103-ant-colony-optimization

[21] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy-Kan and D. B. Shmoys Eds., “The Travelling Salesman Problem,” New York: Wiley, 1985.

[22] J. H. Holland, “Adaptation in Natural and Artificial Systems,” University of Michigan press, pp. 211, 1979.

[23] D. E. Goldberg, “Genetic Algorithm in Search,” Optimization and Machine Learning, Addison-Wesley, pp. 372, 1989.

[24] D. J. Rosenkrantz, R. E. Stearns, and P. M. Lewis, “An analysis of several heuristics for the traveling salesman problem,” SIAM J. Comput.,vol. 6, pp. 563–581, 1977.

[25] S. Lin., “Computer solutions of the traveling salesman problem,” Bell Syst. J., vol. 44, pp. 2245–2269, 1965.

[26] S. Lin and B. W. Kernighan, “An effective heuristic algorithm for the traveling salesman problem,” Oper. Res., vol. 21, pp. 498–516, 1973.