The Effectiveness of Parameter Tuning on Ant Colony Optimization for Solving the Travelling Salesman Problem

Conference Paper · December 2019

DOI: 10.1109/CSNT.2018.17

2 authors:

Kush Shrivastava
Jaypee University of Engineering and Technology

Dr Shishir Kumar
Jaypee University of Engineering and Technology

The Effectiveness of Parameter Tuning on Ant Colony Optimization for Solving the Travelling Salesman Problem

Kush Shrivastava
Department of Computer Science Engineering
Jaypee University of Engineering & Technology
Guna(M.P.),India
kush.shri05@gmail.com

*Shishir Kumar
Department of Computer Science Engineering
Jaypee University of Engineering & Technology
Guna(M.P.),India
dr.shishir@yahoo.com

***Abstract*—Ant Colony Optimization(ACO) is a metaheuristic approach which tackles the combinatorial optimization problems such as Traveling Salesman Problem(TSP). ACO is highly motivated by the foraging behavior of ants and the way they utilize pheromone trail to discover the food source. The performance of an ACO algorithm is directy dependent on the choice of its parameters. The bad selection of parameters increases the computation time and accuracy of the algorithm. In this paper we studied the impact of the basic parameters that are used in ACO algorithms such as, α which is the relative importance of pheromone, β the relative importance of heuristics value and ρ the evaporation rate. We conducted an experiment for the better choice of the parameter values inorder to increase the effectiveness of algorithm for solving travelling salesman problem.**

***Index Terms*—ACO, hyper-parameters, Ant System, metaheuristic**

I. Introduction

The ant colony optimization(ACO) [1] algorithms are the family of algorithms that belong to swarm intelligence methods [2], in which the artificial agents collaborate among each other which leads them to form an intelligent system. It was proposed by an Italian scholar Marco Dorigo during his research study for solving the well-known traveling salesman problem. The characteristics and the behavior portrayed by artificial agents is the one that has been observed in nature, such as stimulating behavior [3] within the ant colonies, bee hives, flocks of birds and the different group of animals and insects. An observation by French entomologist Ernest Andre [4] gave rise to the origin of artificial ant study, the motive was to understand how almost blind insects like ants could manage to find the shortest route paths from their colony to feeding nest back. In his observation, he found that some species of insects use stigmergy, a form of indirect communication mediated by modifications of the environment by an individual and later on the other individuals responds to that change.

To acquire an optimal solution by solving combinatorial optimization problems [5] which include traveling salesman problem, vehicle routing problem, graph coloring problem, quadratic assignment problem, sequential ordering problem and routing in computer network etc. lead the foundation of ACO. The combinatorial problems are NP-hard i.e. it is believed that the problems cannot be solved to reach optimality within a polynomial bounded computation time.

The Combinatorial problem considered in this paper is well known Travelling Salesman Problem (TSP) defined as a problem in which salesman who wants to find the shortest path to his hometown by routing to a given set of cities, visiting one city exactly.The problem has been solved using one of the variants of ACO called Ant System [6]. The problem is instantiated with several parameters which have to be set manually. The parameter settings [11] and variations of ACO are still in the study stage and analysts effectively made a substantial number of investigations on parameter setting and the fundamental properties of ACO algorithms. Almost all publications in the area of ACO applications refer to Dorigo's seminal paper [7] when it comes to the selection of parameter values. This original paper describes the Ant System and analyses the relative merits of certain parameter settings but does not look at the interdependencies between parameters. It does however stress the importance of parameter settings for quick convergence towards the best known solution and mentions the dependency of the parameters on the problem.

We conducted an exhaustive, empirical analysis of the sensitivity of the Ant System algorithm to variations of parameters for different instances of the TSP. The remainder of this paper is organized as follows: Section II describes the overview of Ant System algorithm while section III presents the datasets and parameters used in the experiments. Then, section IV describes the experiments for finding the optimal parameters for Ant System algorithm and results are discussed in a brief manner in section V, and finally section VI concludes and shows the future work.

II. Ant system

In 1991 Dorigo et. al. [7] developed the first ant algorithm which was called Ant System [9] to solve the traveling salesman problem. Originally the term ANT system was used to refer to a range of Ant-based algorithms, as Ant Cycle. Now that Ant Cycle algorithm is popularly referred to as Ant System. The basic objective of the algorithm is to avail the heuristic information of the environment and history of ant's behavior in-order to

978-1-5386-5956-4/18/$31.00 ©2018 IEEE
DOI: 10.1109/CSNT.2018.17

construct the solution and later on expand the information that it has learned via constructing the solution into the history [12]. Dorigo et al proposed three different versions of AS and the difference lying among them was the way pheromone trails are updated. These algorithms are Ant density, Ant Quantity, and Ant Cycle [10]. The two former algorithms update the pheromone while building the solution I.e. while moving from one part of the city to another throughout the tour, however, the later one updates pheromone after completing the whole tour and the amount of pheromone updated becomes the function of the tour quantity. The ANT system algorithm has two phases, solution construction, and pheromone update [8]. The basic idea is to have a set of agents called artificial ants who search in a parallel way for a good solution and communicate through the pheromone to converge the optimal result. The first ACO algorithm, called Ant System was first tested on TSP and the reason being that TSP is an important NP-hard problem arising in various applications. The Ant system improves the efficiency when it is applied to symmetric and asymmetric TSP.

A. *Solution Construction*

Let G= (N, A), on arcs A having an initial amount of pheromone (τ), is available between the cities. The k artificial ants are assigned randomly to the city and then they travel from a current city to each city forming hamiltonian circuit until all cities are visited exactly one. The ant returns to the starting city after visiting the last city. At each movement, the ants modifies the pheromone trail on edges using pheromone trail update [13] shown in Eq 3. Each assigned ant k applies a probabilistic transition rule (Eq 1) for moves from city i to city j. The probability rule between two cities i and j is called random proportional rule which depends on two factors, pheromone and heuristic values. To each arc (i,j)ϵ A, a weight is assigned which represents the distance between the two cities i and j. τ shows what is the desirability of visiting city j just after city i. $\eta_{i,j}$ is chosen as a heuristic information and inverse of distance d_{ij}. Two essential parameters which effect the pheromone value and heuristic value are during the solution construction are α and β [14] Here α sets the amount of the pheromone on the edges would vanish after each cycle. This parameter sets how much memory of past arrangements we need. Parameter β sets the relative significance of pheromone versus heuristic value. $\alpha \geq 0$ and $\beta \geq 0$ & are adjustable parameters describing the weights of the pheromone trail and visibility when choosing the route. The role of parameters α and β are that if α=0, the cities that are closest are more likely to be selected. If β=0, the only pheromone is used without any heuristic. [15] Probability transition rule for ant k from city i to city j is:

$$p_{ij}^{k} = \frac{[\tau_{ij}]^{\alpha} * [\eta_{ij}]^{\beta}}{\sum_{u \epsilon N i^{k}} [\tau_{iu}]^{\alpha} * [\eta_{iu}]^{\beta}} \tag{1}$$

where,

N_i^k is the set of cities u not yet visited by an ant k. Each ant k maintains a memory to store the cities in the order they already visited. The memory is used to define the neighboring cities and also allows ant to compute the length of the tour.

B. *Pheromone Trail Update*

The ants communicate with each other via pheromone trails while searching for the food or nest and this is the main reason why ACO performs as a multi-agent system. Higher the intensity of pheromone a part of the path contains, higher the possibility for the ants to follow the path. In Ant System algorithm the k ant while moving from one city to another lay down some amount of pheromone substance [16]. Initially, a small amount of pheromone trail is presented over the path which is later decayed by a constant rate [17]. So inorder to reinforce the decayed pheromone trail the global best ant who found the shortest path after the solution construction updates the pheromone substance. The application of global update rule (Eq 3) is to make the ants search for the path in the environs of the best tour found so far.

C. *Pheromone Evaporation Rule*

Before implementing the update rule, a fraction of pheromone is allowed to evaporate on all edges which is done by applying the pheromone evaporation rule as shown in Eq 2 to avoid unlimited accumulation of the pheromone trails and also allows the other ants to forget the bad choices [15]. The coefficient of evaporation is denoted by $\rho \ \epsilon$ [0,1].
The evaporation helps to find the shortest path and provide that no other path will be assessed as the shortest. This process simulates the natural process of evaporation preventing the algorithm from converging too quickly and get trapped into the local optimum. The value of evaporation shows the importance of pheromone values from one iteration to the other [8].
Pheromone decay makes a real difference:

- Pheromone decay must wipe out obsolete paths quickly, but not remove newly formed paths before they are followed by other ants paths.
- Ideally, decay should cause closer food sources to be favored over existing, far ones.

At time t=0, $\tau_{ij} = \tau_0$ and after the time interval t the pheromone is decayed by a constant rate ρ :

$$\tau_{ij}(t) \leftarrow (1-\rho) * \tau_{ij} \tag{2}$$

The unlimited concentration of pheromone is removed by ρ and after evaporation the global pheromone trail update is applied as follows-

$$\tau_{ij}(t) \leftarrow \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \quad , \forall (i,j) \ \epsilon \ L \tag{3}$$

where

$$\Delta\tau_{ij}^{k} = \begin{cases} Q/C^k, & if \ arc(i,j) \ belongs \ to \ T^k; \\ 0 & otherwise > 0 \end{cases} \tag{4}$$

here T^k is the tour constructed by k^{th} ant, C^k is the cost of the k^{th} ant's tour (typically length L) and Q is a constant.

III. DATASETS

We benchmark our algorithm utilizing freely accessible dataset. The TSP dataset is the directory which contains sets of information about cities and the distances between them. In the following experimentation, GR17 TSP dataset [18] is employed to study the behavior of Ant System algorithm. GR17 is a set of 17 cities, from TSPLIB. The minimal tour has length 2085. The computational study on various ACO algorithms includes several parameters listed in Table I:

TABLE I: Parameters considered in computation

Parameters	Meaning
α	This parameter sets the amount of the pheromone on the edges would vanish after each cycle. This parameter really set how much "memory" of past arrangements we need.
β	This parameter sets the relative significance of pheromone versus heuristic value.
ρ	This parameter sets the amount of the pheromone on the edges would dissipate after each cycle. This parameter sets of the amount one ant impacted by different ants.

IV. EXPERIMENTAL STUDY FOR OPTIMAL PARAMETERS

The choice of parameter values exercises a great influence in the solution process and the effectiveness of an algorithm. The Ant System algorithm have a performance which strongly depends on the parameter setting under which they run [19]. The parameters are α (sensibility to trail), β (sensibility to distance) and ρ (evaporation or decay rate to pheromone trail). We are interested in studying that how parameter values affect the overall performance of the algorithms. We implemented the Ant System algorithm of ACO using Java programming language and tested several values for each parameter keeping all the other values of parameters constant, over ten simulations for each setting. The parameters mentioned in Table I are considered which directly or indirectly affects the computation of the probability to visit the neighboring city.

The number of ants has always been set equal to the number (n) of the cities. We tested several values of α β & ρ, on the GR17 tsp data instance with 17 cities. The values tested were: α $\epsilon\{0.2, 0.5, 0.8, 1, 2, 5\}$, β $\epsilon\{0.2, 0.5, 1, 3, 5, 10\}$ & ρ $\epsilon\{0.1, 0.3, 0.5, 0.7, 0.9\}$. All the tests have been carried out for iterations t ϵ $\{50, 100, 300, 500, 1000\}$ and were averaged over 10 trials.

At first, to understand the impact of parameter α on Ant System, an experiment was conducted by varying values of α from 0.2 to 5, keeping values of the parameter β=1 & ρ=0.1. For every t iteration, the algorithm was run for 10 trials and average tour length was calculated. The average tour length for every t iteration & parameter α is shown in Table II. In the same manner the value of β was varied from 0.2 to 10, keeping values of parameter α=0.5 & ρ=0.1. The average tour length for every t iteration & parameter β is shown in Table III. At last to understand the impact of parameter ρ, in the conducted experiment values of ρ was varied from 0.1 to 0.9, keeping values of the parameter α=0.5 & β=1. The Table IV shows the average tour length for every t iteration & parameter ρ. The optimal tour length for gr17 is 2085 which was obtained by the set of parameters in tables below. It was observed that when the number of iterations exceeds, despite the stochastic behavior all the ants follow the same tour. This happens because of higher deposition on the edges forming that tour than on all the others.

TABLE II: Best tour length for varying α values, β = 1 & $\rho = 0.1$

	α					
Iteration	0.2	0.5	0.8	1	2	5
50	2498.2	2326.3	2285.5	2197.7	2154.2	2139.6
100	2456.5	2235.7	2142.9	2098	2138.9	2158.2
300	2432.4	2184.7	2085	2085	2116	2155.8
500	2403.3	2085	2085	2085	2116	2154.8
1000	2386.3	2085	2085	2085	2116 Z	2148.6

TABLE III: Best tour length for varying β values, $\alpha = 0.5$ & ρ = 0.1

	β					
Iteration	0.2	0.5	1	3	5	10
50	3211.6	2938.5	2326.3	2263.7	2149	2149
100	2826.5	2745.4	2565.1	2123.5	2149	2149
300	2795.5	2566	2184.7	2085	2149	2149
500	2746.7	2546	2085	2085	2149	2149
1000	2635.7	2488	2085	2085	2149	2149

TABLE IV: Best tour length for varying ρ values, α = 0.5 & β = 1

	ρ				
No. of Iteration	0.1	0.3	0.5	0.7	0.9
50	2326.3	2161.3	2159.3	2203.6	2314.4
100	2235.7	2121.9	2162.4	2227	2301.8
300	2184.7	2132.2	2151.6	2197.4	2336.4
500	2085	2106.7	2136.6	2194.8	2314.8
1000	2085	2104.3	2136.6	2192.3	2286.4

V. RESULTS AND ANALYSIS OF EXPERIMENT

In this experiment we took the GR17 TSP instance and the important ACO parameters were tuned to observe the effect of finding the solution. Also, the best values of parameters were found for which the solution is optimal. For this, a deep analysis is done to understand the variation of parameters. The results obtained after conducting experiment 1 are shown in the Figure. 1, 2 & 3. In Figure 1 we illustrated the results of the best tour length versus number of iterations when value of α is varied, in Figure 2 the results of the best tour length versus number of iterations are shown when value of β is varied and in FIgure 3, we illustrated the results of the best tour length versus number of iterations when value of ρ is varied.

A. *Impact of Pheromone factor α*

The experiment was conducted to study the impact of pheromone factor α in the ACO algorithms. Different tests were conducted, varying value of a parameter at a time, keeping the others unchanged. In a first step, the value fo α was varied from 0.2 to 5, keeping β=1 & ρ=0.1. The results for best tour length versus number of iterations are analyzed in Figure 1 and discussed below:

if α**=0.2**, we see that when iterations are less the tour length calculated is very high but gradually decreases when the iteration reaches 1000. This means if enough importance is not given to the pheromone trail i.e. α is set to lower values then the algorithm is unable to find the good solution [20].

if α**=0.5**, we see that till 300 iterations a bad solution is found but after 500 iterations it gradually enters the stagnation behavior and the good solution is found every time [20] [7].

if α **=0.8 and 1**, very good solutions are found for α for these values. The algorithm starts finding the good solution even if the iterations are low and later when iterations were increased to 500 and more the shortest tour i.e. 2085 was obtained.

if α**=2 and 5** , when the values of α is kept higher than 1 the algorithm enters the stagnation behaviour very rapidly without finding very good solutions, this means a situation in which all the ants follow the same path and construct the same tour without considering the probability of choosing another city that is near. So it is better to use a bit high value of β if pheromone value is high as it will favor the probability of choosing another city that is near to high pheromone value [21].
As a conclusion we can state that for each examined value of α, 0.8 & 1 gives the similar results when this algorithm is in its optimal condition. The value 2 & 5 are very high and every time they constructs the same path and same tour which is sub optimal. The value 0.2 always remains the worst as it does not give enough emphasis to the pheromone value.

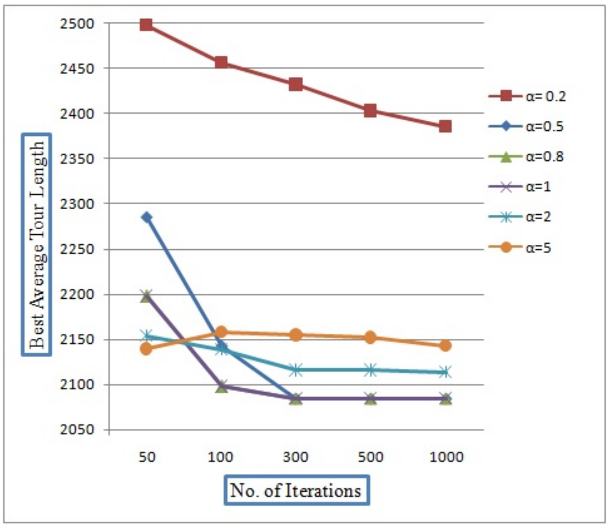

Fig. 1: Best tour length versus number of iterations, for α=0.2 to 10, β=1 and ρ=.1

B. *Impact of Heuristic factor β*

This experiment was conducted to study the impact of parameter β on the ACO algorithm. The values were varied from 0.2 to 10, keeping value of α=0.5 and ρ=0.1. Here it is worth noticing that we did not take the value of α as 1 because it may make the search bias. The results for best tour length versus number of iterations are analyzed in Figure 2 and discussed below:

if β**=0.2 and 0.5**, we see that the algorithm finds a poor solution either the iterations are less or more. This happens because if β is close to 0 it does not give enough emphasis to heuristic values i.e. only pheromone is used which generates the poor results [21].

if β**=1 and 3**, we see that when iterations are less the tour length found is not optimal but as soon as the iteration increases the algorithm is capable of finding the best tour length because it gives more emphasis to heuristic values which in turn finds the better tour. [22]

if β**= 5 and 10**, we see that whether the iterations are less or more, the same tour is found by the ants which means that ants give very high emphasis on heuristic value and it enters the stagnation behavior very rapidly without finding very good solutions [20]. It is seen that the ants show the uni- path behavior means the situation in which all the ants make the same tour: this would indicate that the system has ceased to explore new possibilities and therefore the best tour achieved so far will not be improved any more. As a conclusion in we consider β=3 to be best value because it finds the better solution in less iterations. It is also noticed that the high value of β and low values of α makes the algorithm very similar to stochastic multigreedy algorithm

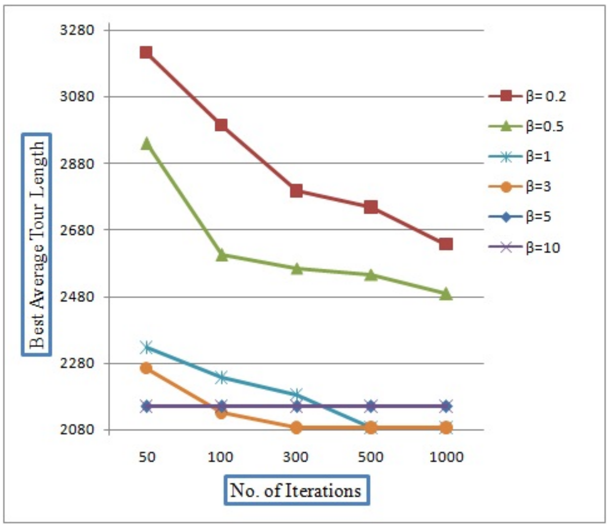

Fig. 2: Best tour length versus number of iterations, for β=0.2 to 10, α=0.5 and ρ=0.1

C. *Impact of decaying factor ρ*

The pheromone trail laid by ant along the tour evaporates with time. The parameter ρ directly effect the global search capability as well as randomness. The experiment was conducted to study the impact of decay factor ρ in the ACO algorithms. Different tests were conducted, varying value of a parameter at a time, keeping the others unchanged. In a first step, the value for ρ was varied from 0.1 to 0.9, keeping α=0.5 & β=1. In table IV the value of α is selected as 0.5 because for values of $\alpha > 1$ it leads to the rapid emergence of a stagnation situation, that is, a situation in which all the ants follow the same path and construct the same tour, which, in general, is strongly sub-optimal [20]. To get a good result algorithm should choose the appropriate range of α and β , generally α = 0.5~5 and β = 1~5 [21]. Therefore we selected α = 1 and β = 0.5 for our experimentation. The two parameters complement each other and closely related, therefore combinations of and are used to discus there impact on the performance of ant colony algorithm. The results for best tour length versus number of iterations are analyzed in Figure 3 and discussed below:

if ρ**=0.1 and 0.3**, we can see that it's effect is not seen when iterations are less, but as soon as the iterations increase the optimal solution is found. At first, the algorithm greedily finds the solution but later the impact of pheromone is lowered and the search finds optimal tour length. The algorithm forgets the part of the experience gained in the past in order to better exploit new incoming global information [23].

if ρ**=0.5**, we see that the behavior of algorithm is near about same in all iterations and if the iterations are increased it will reach the optimal solutions. It shows that keeping a strong memory of past experience is a good policy to explore the new information of path [23].

if ρ**=0.7 and 0.9**, this parameter shows some form of degradation to find the better solution. It is only suitable to use the large ρ values when the pheromone trail is high in value otherwise the evaporation of trail with higher rate will make the ants forget the better path of their tour [7].
The best value found for ρ is 0.1 and 0.3.

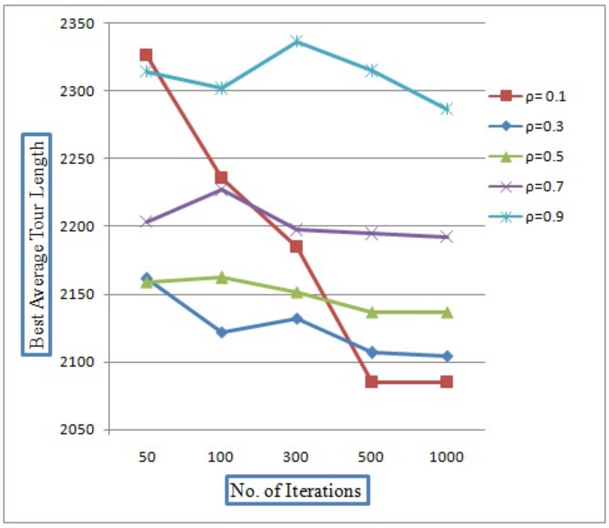

Fig. 3: Best tour length versus number of iterations, for ρ=0.1 to 0.9, α=0.5 and β=0.3

The results found in this examination demonstrates that a high value for α suggests that pheromone trail is fundamental and therefore ants tend to pick edges chosen by other ants previously. This is substantial until the point that the value of β ends up being high: for this situation regardless of the possibility that there is a high amount of trail on an edge, an ant always has a high possibility of choosing another city that is extremely close. So the optimal parameters after performing the tuning are recorded underneath in Table V.

TABLE V: Optimal Parameters

Parameters	Value
α	1
β	3
ρ	0.3

VI. Conclusion and Future Work

The combinatorial optimization problems are NP-hard problems but can be solved using some meta-heuristics. Ant Colony Optimization is a metaheuristic defined to solve the combinatorial optimization problems. In this work, we have presented an Ant Colony Optimization algorithm for solving travelling salesman problem. After giving a brief introduction of Ant system algorithm the experiments are conducted to study the impact of parameters on this algorithm. We have shown that the performance of Ant system algorithm depends on the appropriate setting of parameters which requires both human experience and luck to some extend. These parameters are dependent on the problem instance at hand and also on the required solution accuracy.

Parameter optimization in most of the machine learning and neural network techniques is one of the important steps in training the dataset without compromising the computational expensive. The manual tuning of parameters are in-effective and time-consuming as the number of parameters is increased. Hence, a well-sophisticated optimization algorithm is needed to be addressed. Further, this work opens up interesting opportunities in applying ant colony optimization to deep learning algorithms. The approach could be extended to evolve deep learning alogrithms for text classification, by allowing ants to find the optimal hyper parameters. Automatically searching for optimal hyperparameter configurations is of crucial importance for applying deep learning algorithms in practice. In deep learning architecture hyperparameters are the variables which determines the network structure(eg: number of hidden units) and the variables which determine how the network is trained(eg: learning rate). The hyper parameters such as number of filters, no. of filter sizes, activation functions, dropout rate, learning rate learning rate for an optimizer can be optimized using ACO algorithm.

References

[1] Dorigo, M., & Socha, K. (2006). An introduction to ant colony optimization. Handbook of Approximation Algorithms and Meta-heuristics, 26-1.
[2] Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). From natural to artificial swarm intelligence.
[3] Grassé, P. P. (1946). Les Insects Dans Leur Univers, Paris, France, Ed. Du Palais de la découverte.
[4] Wheeler, W. M. (1910). Ants: their structure, development and behavior (Vol. 9). Columbia University Press.
[5] Darquennes, D. (2005). Implementation and Applications of Ant Colony Algorithms. M. Sc. Tesis, University of Namur, Belgium.
[6] Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Transactions on evolutionary computation, 1(1), 53-66.
[7] Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 26(1), 29-41.
[8] Dorigo, M., & Gambardella, L. M. (1997). Ant colonies for the travelling salesman problem. biosystems, 43(2), 73-81.

[9] Gambardella, L. M., & Dorigo, M. (1996, May). Solving symmetric and asymmetric TSPs by ant colonies. In Evolutionary Computation, 1996., Proceedings of IEEE International Conference on (pp. 622-627). IEEE.
[10] Dorigo, M., Di Caro, G., & Gambardella, L. M. (1999). Ant algorithms for discrete optimization. Artificial life, 5(2), 137-172.
[11] Stützle, T., López-Ibánez, M., Pellegrini, P., Maur, M., De Oca, M. M., Birattari, M., & Dorigo, M. (2011). Parameter adaptation in ant colony optimization. In Autonomous search (pp. 191-215). Springer Berlin Heidelberg.
[12] Kim, D. (2011, April). Evolutionary multiobjective optimization for memory-encoding controllers in the artificial ant problem. In Computational Intelligence in Multicriteria Decision-Making (MDCM), 2011 IEEE Symposium on (pp. 73-80). IEEE.
[13] Do Duc, D., Dinh, H. Q., & Xuan, H. H. (2008, December). On the pheromone update rules of ant colony optimization approaches for the job shop scheduling problem. In Pacific Rim International Conference on Multi-Agents (pp. 153-160). Springer, Berlin, Heidelberg.
[14] Fidanova, S., Roeva, O., & Paprzycki, M. (2015). InterCriteria analysis of ant colony optimization application to GPS surveying problems. Issues in Intuitionistic Fuzzy Sets and Generalized Nets, 12, 20-38.
[15] Parsons, S. (2005). Ant Colony Optimization by Marco Dorigo and Thomas Stützle, MIT Press, 305 pp., $40.00, ISBN 0-262-04219-3. The Knowledge Engineering Review, 20(1), 92-93.
[16] Michelakos, I., Mallios, N., Papageorgiou, E., & Vassilakopoulos, M. (2011). Ant colony optimization and data mining. In Next Generation Data Technologies for Collective Computational Intelligence (pp. 31-60). Springer Berlin Heidelberg.
[17] Dorigo, M., & Di Caro, G. (1999). Ant colony optimization: a new meta-heuristic. In Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on (Vol. 2, pp. 1470-1477). IEEE.
[18] https://people.sc.fsu.edu/ jburkardt/datasets/tsp/tsp.html
[19] Li, P., & Zhu, H. (2016). Parameter Selection for Ant Colony Algorithm Based on Bacterial Foraging Algorithm. Mathematical Problems in Engineering, 2016.
[20] Dorigo, M., Stützle, T. (2005). "Ant Colony Optimization", MIT Press, Bradford Company, Scituate, MA, USA pp. 71, ISBN 0262042193
[21] Q. Wu, J. Zhang and X. Xu (1999), "Variability of ant colony algorithm", Computer Research and Development, vol. 36, no. 10.
[22] Randall, M. (2004, September). Near parameter free ant colony optimisation. In International Workshop on Ant Colony Optimization and Swarm Intelligence (pp. 374-381). Springer, Berlin, Heidelberg.
[23] Colorni, A., Dorigo, M., & Maniezzo, V. (1991). Ant system: An autocatalytic optimizing process. Dipartimento Di Elet-tronica, Politecnico Di Milano, Milan, Italy:[sn].