

IMPACT OF EVAPORATION PARAMETER ON QUALITY OF SOLUTION TO TRAVELLING SALESMAN PROBLEM BY ANT COLONY OPTIMIZATION ALGORITHM

Ondřej Míča¹

Abstract: Ant colony optimization is metaheuristic algorithm inspired by nature. It has several parameters, which have to be set before the first run of algorithm and their correct setting is important for finding a satisficing solution. In this paper, it is shown how different value of one of these parameters – the evaporation factor – can affect the quality of given solution.

Key words: ant colony optimization, metaheuristic, optimization, travelling salesman problem.

INTRODUCTION

Travelling salesman problem (TSP) belongs in basic problems of operations research. It is a NP-hard problem. The number of possible solutions of this problem is very high – it increases with the factorial of the number of the nodes at the graph. So even with nowadays computers, it takes very large amount of time to solve TSP with exact methods. Therefore, TSP is now usually solved with a heuristic (or metaheuristic) techniques, which provides a satisfactory solution in real-time.

One of the metaheuristic methods for solving Travelling salesman problem is Ant colony optimization algorithm (ACO). In this algorithm, several parameters are used, which have to be set manually. These parameters determine the behavior of each ant and are critical for fast convergence to near optimal solutions of a given problem instance.

1. TRAVELLING SALESMAN PROBLEM

Travelling salesman problem is very well known and popular optimization problem. The goal of TSP is to visit all nodes (cities) of the graph exactly once with the possible shortest route and return to the origin node (i.e. to find Hamiltonian cycle called after the Irish mathematician William Rowan Hamilton, who is considered to formulate TSP in 19th century). (1)

This seeking the Hamiltonian cycle is situated on a transportation network, which can be described as graph $G = (\mathbf{N}, \mathbf{E})$, where \mathbf{N} is set of n nodes (cities) and \mathbf{E} is set of m edges (paths) between these nodes and each edge has its own length. The network is a complete graph (i.e. each pair of nodes is connected by an edge) very often. Or when the graph is not complete, fictive edge between two unconnected nodes with length of shortest path between these two nodes can be added without affecting the optimal solution. (2)

¹ Ing. Ondřej Míča, University of Pardubice, Jan Perner Transport Faculty, Department of Informatics in Transport, Studentská 95, 532 10 Pardubice 2, Czech Republic. GSM.: +420 466 036 428, E-mail: on.mica@email.cz

TSP in general can be represented with following mathematical model:

$$\min \sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq i}}^n d_{ij} x_{ij} \quad (1)$$

$$\sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1 \text{ for } j = 1 \dots n \quad \sum_{\substack{j=1 \\ j \neq i}}^n x_{ij} = 1 \text{ for } i = 1 \dots n \quad (2)$$

$$y_i - y_j + n x_{ij} \leq n - 1 \text{ for } 2 \leq i \neq j \leq n \quad (3)$$

$$x_{ij} \in \{0,1\} \text{ for } i, j = 1 \dots n \quad y_i \in \mathbb{N}_0 \text{ for } i = 1 \dots n \quad (4)$$

where n is a number of nodes, d_{ij} is a distance between nodes i and j , and variable x_{ij} is 1 when the edge between nodes i and j belongs to the Hamiltonian cycle, otherwise it is 0.

The equalities (2) provide that each node can be entered and left only once. The constraint (3) provides that Hamiltonian cycle is only one tour, not a number of smaller cycles. (3)

There are several cases of TSP (not all are mentioned): (4)

- Symmetric: distance between nodes i and j is the same as distance between j and i .
- Asymmetric: distance between nodes i and j is not the same as distance between j and i .
- With time windows: each node can be visited only in given amount of time.
- Sequential Ordering Problem: nodes can be visited only in given order.
- Etc.

Only a symmetric travelling salesman problem is considered further in this paper.

Since TSP is NP-hard problem (1) – the number of possible solutions for graph with n nodes is $(n - 1)!/2$ – the exact algorithm can be used only for small number of nodes. Table Tab. 1 contains duration of finding solution with exact algorithm when the computer would be able to evaluate 1 000 000 000 solutions per second. Therefore, only heuristics or metaheuristics are used for solving large TSP.

Tab. 1 – Duration of exact algorithm

Number of nodes	Number of possible solutions	Duration of exact algorithm [year]
18	177 843 714 048 000	0,006
19	3 201 186 852 864 000	0,102
20	60 822 550 204 416 000	1,929
21	1 216 451 004 088 320 000	38,573
22	25 545 471 085 854 700 000	810,042

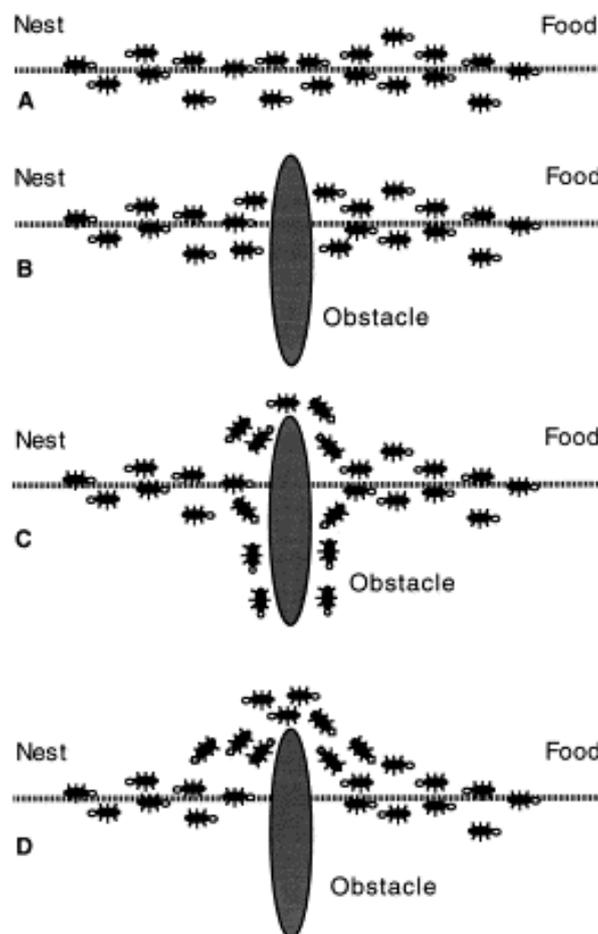
Source: Author

2. ANT COLONY OPTIMIZATION ALGORITHM

Ant colony optimization algorithm (ACO) is young probabilistic technique of discrete optimization. It was first presented by Marco Dorigo in his doctoral thesis in 1992 (5). It was originally designed for finding the shortest path between two nodes of graph. Later, the ACO method was extended and now is used for solving a large amount of optimization problems. e.g. (6)

ACO belongs in nature-based algorithms, as e.g. Bat algorithm, Bees colony, Particle swarm optimization, etc. It is based on examining ant colonies and studying cooperation and communication of ants when they are searching for food. Ants belong to social insect; it means that the prosperity of whole colony is much more important than surviving of individual ant.

Ants communicate with each other using pheromone trails. The pheromones are chemical substances used by ants to mark their paths. Ants leave pheromone trails on a ground and other ants can scent the direction and intensity of those pheromones. Each ant, which uses a marked path, renews a pheromone trail because it evaporates during a time (loses its attractive strength). When the path is not used for some time, the pheromone trail slowly disappears. (5)



Source (6)

Fig. 1 – Reaction of ants to obstacle at their path

The communication between ants by pheromones is shown at figure 1. At first (figure 1A), there is a pheromone marked path between the colony and the source of food. When a new obstacle appeared at the path (figure 1B), ants start to find a new path randomly choosing turn left or right (we can assume with the same probability) (figure 1C). At the shorter path, there is less time for pheromones to evaporate, so the density becomes higher and during a time, all ants will choose the shorter path (figure 1D). (6)

3. THE DIGITAL IMPLEMENTATION OF ACO

Since the ACO is not a simulation of real ant colony, but it is used for seeking the shortest path in a graph, for implementation is used artificial ant, which has three main attributes. They are:

- The ant remembers its path in the graph.
- The ant moves discretely only from one node to another.
- The ant is able to determine the attractiveness of each edge η_{ij} according to formula

$$\eta_{ij} = \frac{1}{d_{ij}} \quad (5)$$

where d_{ij} is a distance between nodes i and j .

At the beginning of the algorithm, the m artificial ants are placed into randomly selected nodes of the graph. At each step, they move to the new node with probability given by the random proportional rule defined as

$$p_{ij} = \frac{(\tau_{ij})^\alpha (\eta_{ij})^\beta}{\sum_{i \in N_k} (\tau_{ii})^\alpha (\eta_{ii})^\beta} \quad (6)$$

where τ_{ij} is amount of pheromone on the edge between nodes i and j , α and β are parameters and N_k is set of unvisited nodes for k^{th} ant. After visiting all nodes (the set of unvisited nodes is empty), the ants return to their original nodes and the amount of pheromone on each edge is updated according to the following formula

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (7)$$

where ρ is evaporation parameter and $\Delta\tau_{ij}^k$ is defined as

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{if } k^{\text{th}} \text{ ant uses edge } (i,j) \text{ in its Hamiltonian cycle} \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where Q is a constant and L_k is a length of found Hamiltonian cycle of k^{th} ant. (7)

4. RESULTS

In ACO algorithm, there are several parameters described in previous section, which have to be set before starting the optimization algorithm – α , β , ρ and m .

Parameters α and β are used to weigh the relative influence of the pheromone and the attractiveness of the edge. Parameter ρ is evaporation parameter from interval $(0,1)$, which represents the speed of evaporation of pheromones and avoids unlimited accumulation of pheromones on edges. And m is number of artificial ants used for ACO algorithm. (7)

TSPLIB (9) was used for benchmarking and testing examples. TSPLIB is a library of sample instances for the TSP (and related problems) from various sources and of various types. This library nowadays contains 112 instances of symmetric TSPs with its optimal solutions. For this paper, 8 instances of TSP with different numbers of nodes were chosen.

Parameter setting chosen for empirical tests was according to (7) and (8) $\alpha = 1$, $\beta = 5$ and m was set as number of nodes of each tested example.

The ACO algorithm ran for each tested instance of TSP with following values of parameter ρ : 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 0.999. For each combination of tested instance and value of parameter ρ , the ACO algorithm ran 500 times and each run was limited to 500 iterations. So together 40 000 runs of ACO algorithm were made for testing.

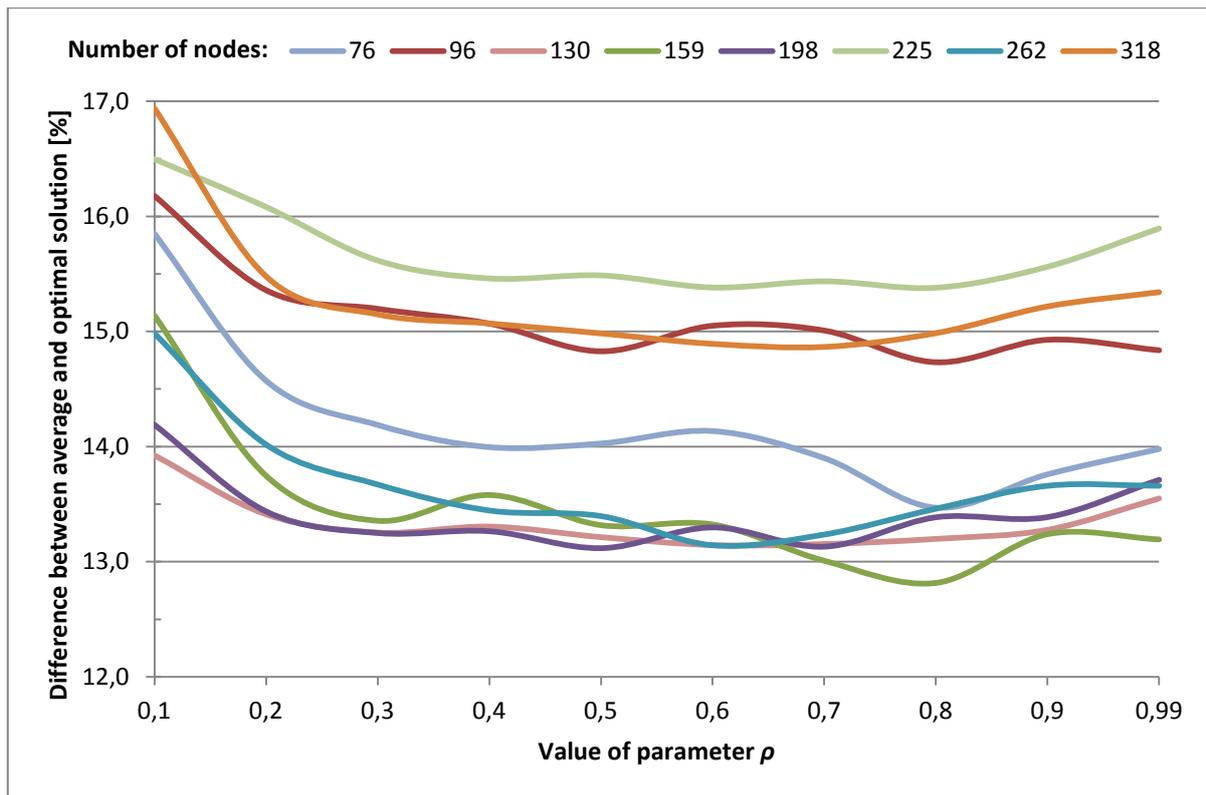
An average solution for each tested combination was counted from 500 given solutions and this average solution was compared with the optimal solution of instance taken from (9) and the difference between average and optimal solution was determined.

Tab. 2 – Difference between average and optimal solution for each tested combination [%]

		Value of parameter ρ									
		0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,99
Number of nodes	76	15,85	14,57	14,19	14,00	14,03	14,14	13,90	13,47	13,76	13,98
	96	16,18	15,36	15,20	15,07	14,83	15,05	15,01	14,73	14,93	14,84
	130	13,92	13,41	13,25	13,31	13,21	13,14	13,15	13,20	13,28	13,55
	159	15,14	13,75	13,36	13,58	13,32	13,32	13,01	12,82	13,24	13,19
	198	14,19	13,44	13,25	13,26	13,14	13,30	13,13	13,39	13,39	13,71
	225	16,50	16,08	15,62	15,46	15,49	15,38	15,44	15,38	15,56	15,90
	262	14,98	14,02	13,67	13,44	13,40	13,14	13,24	13,46	13,66	13,66
	318	16,94	15,48	15,15	15,07	14,98	14,89	14,87	14,99	15,22	15,34

Source: Author

The differences between optimal and average solution are shown in table 2 and figure 2. All 8 instances of TSP reach the lowest difference using parameter ρ from value 0.6 to 0.8. To be specific, two instances have the lowest difference using $\rho = 0.6$, also two examples in 0.7, in three cases is the lowest difference in 0.8 and one instance reaches the minimum difference both in 0.6 and 0.8.



Source: Author

Fig. 2 – Difference between average and optimal solution

5. CONCLUSION

In this paper, it is shown that the performance of ACO algorithm depends on the suitable setting of evaporation parameter, which requires both human experience and luck. The possible range of this parameter is from 0 to 1. But all 8 instances of TSP used for benchmarking in this paper gain the best average solution using evaporation parameter ρ from value 0.6 to 0.8.

REFERENCES

- (1) CENEK, P., JÁNOŠÍKOVÁ, Ľ.: Models and optimisation in transports and logistics. EDIS – vydavateľstvo ŽU: Žilina. 2008. ISBN 978-80-8070-951-8.
- (2) VOLEK, J.: Operační výzkum I. Vydání 2., nezměn. Pardubice: Univerzita Pardubice, 2008, ISBN 978-80-7395-073-6.
- (3) MILLER, C. E., TUCKER, A. W. and ZEMLIN, R. A.: Integer Programming Formulation of Traveling Salesman Problems. In: Journal of the ACM. p. 326-329. DOI: 10.1145/321043.321046. ISSN 00045411.
- (4) APPLGATE, D. L.: The traveling salesman problem: a computational study. Oxford: Princeton University Press, c2006, ix, 593 s. Princeton series in applied mathematics. ISBN 0-691-12993-2.

- (5) DORIGO, M.: Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italy, 1992.
- (6) DORIGO, M., GAMBARDELLA, L. M. and VECCHI, M. P.: Ant colonies for the travelling salesman problem. In: Biosystems. 1997, p. 73-81. DOI: 10.1016/S0303-2647(97)01708-5. ISSN 0303264
- (7) DORIGO, M., MANIEZZO, V. and COLORNI, A.: Ant system: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*. 1996, Vol. 26: p. 29-41 DOI: 10.1109/3477.484436
- (8) GAERTNER, D. and CLARK, K.: On Optimal Parameters for Ant Colony Optimization Algorithms. In: *Proceedings of the International Conference on Artificial Intelligence 2005*. CSREA Press, p. 83-89.
- (9) TSPLIB. Ruprecht-Karls-Universität Heidelberg [online]. 2001 [accessed 2015-10-05]. Available from: <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>