OPTIMIZING ANT COLONY ALGORITHM PARAMETERS USING TAGUCHI METHOD

Van Hiep Dao[1,*]

[1]*Le Quy Don Technical University*

Abstract

In the present study, Taguchi Method (TM) is applied to determine the optimum Ant Colony Optimization (ACO) parameters. The Tool Path Optimization (TPO) for drilling a hole pattern on the Printed Circuit Board (PCB) with 220 holes was examined as a case study. Experimental verification shows that, with the optimal ACO parameters the tool path length is reduced about 2 times and the convergence rate is significantly approved against their average values in the initial data. This setting variant is also confirmed to be actually the best in comparison with available ones. This allows to reduce not only time for machining but also the one for program execution.

Keywords: *Taguchi method; Ant Colony Optimization; Tool Path Optimization; Traveling Salesman Problem; Printed Circuit Board.*

1. Introduction

Ant Colony Optimization (ACO) belongs to the group of metaheuristic approaches for solving a wide range of optimization problems. The inspiring source of ACO is the foraging behavior of real ant colonies. The first ACO algorithm, i.e. Ant Systems (AS), addresses the well-known Traveling Salesman Problem (TSP).

Mathematically, TSP is classified to the class of combinatorial NP-hard optimization problems, that cannot be exactly solved in polynomial time. Exact methods work well only for TSPs with not more than 80 vertices [1]. For large-scale TSPs, it is more practical to use heuristic algorithms that have relatively short running time and they often give solutions that only slightly differ from the true optimum.

As a nature-inspired metaheuristic, ACO comprises some algorithms with different levels of complexity and performance. From the initial algorithm AS to recent ones, ACO is improved with much better computational performance [2], and enlarged variety of different problems, such as TPO in machining [3, 4], operation sequence optimization in machining process planning [5, 6], telecommunication network routing, vehicle routing, sequential ordering, scheduling, and so on. Some other metaheuristic based methods, such as Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are foremost candidates for a wide range of optimization problems, and several researchers have also

* Email: hdaovan@gmail.com
https://doi.org/10.56651/lqdtu.jst.v14.n01.478

successfully used them for TSP-type problems [3], but for the last case, Dorigo et al. [7] suggested ACO as one of the most efficient candidates for choice. However, similar to other heuristic-based optimization algorithms, ACO has notable disadvantages, such as:

- Gives only near-optimal but not exact solution;
- Not guaranteed for repeatability of results with the same initial condition settings;
- The optimization performance is highly dependent on parameter setting and difficult to control.

Despite this fact, ACO users usually determine parameters based on their own experience [5] or referring to some suggestions, such as the one from Dorigo's original publications. Unfortunately, while analyse the relative merits of certain parameter settings the authors don't look at the interdependencies between parameters themselves [8]. More carefully, some authors [4, 6] examine the effect of some parameters on the optimal solution, but only for small problems with a few nodes (6-12); while some others try to experimentally tune parameters [9, 10], but each single parameter was analyzed and tuned separately, i.e. in each experiment, one parameter is changed while others are fixed. Such a one-factor-at-a-time (OFAT) experimental model doesn't allow to determine the effect of several factors at the same time. The most important question that interests every researcher is how to set the ACO parameters for convergence toward the best solution with a few iterations?

In the last two decades appeared many optimizing techniques that help to answer this type of questions, among which TM is said to be the robust design technique to optimize the most common controllable parameters, such as machining conditions in material processing or structural parameters in mechanical design. In order to make a leap in effectiveness for ACO application, in this study, we propose an experimental method using TM for optimizing, or more precisely speaking, for fine tuning the ACO parameters to predetermine the optimal performance indices (or criteria).

2. ACO and possibility of predetermining its performance

ACO algorithms simulate foraging behavior of real ant colonies, and are described in many text books such as [7] and articles. In [11], we briefly introduced one instance of ACO and its application to solve the TPO problem for drilling a hole pattern on PCB. In this article the same TPO problem is used as a case study to express the importance of parameter setting for convergence towards the best solution as quick as possible.

ACO's performance is highly dependent on parameters, including m - number of ants; Q - update constant; α and β - exponents of pheromone concentration and heuristic information; and ρ - coefficient of evaporation. Experiments show that the influence of m and Q is not noticeable, but of α, β and ρ is strong [7].

Metaheuristic algorithms need to achieve an appropriate trade-off between the exploitation of the search experience gathered so far and the exploration of search space regions. In ACO, the systematic variation of α and β could be useful measure to get the balance. Let us consider the first parameter, α. For $\alpha > 0$, the larger value of α results in the stronger exploitation of the search experience. For $\alpha = 0$, the pheromone trails are not taken into account, then the selection probabilities are proportional to $\eta_{i,j}^{\beta}$, and the closest cities will more likely be selected. For $\alpha < 0$, the most probable choices done by the ants are less desirable from the point of view of pheromone trails. Hence, varying α could be a measure to shift from exploration to exploitation and vice versa. Parameter β determines the influence of the heuristic information in a similar way. If $\beta = 0$, only pheromone amplification is at work. This will lead to the rapid emergence of a *stagnancy* situation where all the ants follow the same path and construct the same solution, that may correspond to sub-optimal solutions.

Parameter ρ is used to avoid unlimited accumulation of the pheromone trails and enables the algorithm to *"forget"* previously done bad decisions. In other words, this is needed to avoid a much rapid convergence towards a sub-optimal region, while favoring the exploration of new areas in the search space.

As the performance criterion, we usually consider the exactness of solution. In minimization problems, the optimal solution corresponds to the minimal objective function, so the smaller the objective function value L_{min} is, the more exact the optimal solution is produced. In principle, functional relationships between the ACO's optimization performance criterion L_{min} and parameters α, β and ρ may exist, allowing predetermine the performance criterion by tuning parameters. Unfortunately, at the moment the required explicit functions have not been found. So the feasible way to tune ACO parameters is by experiments.

In the next sections, we are going to promote how to use TM for finding optimal ACO parameters and their effect to the performance indices.

3. Taguchi method and its application for tuning ACO parameters

3.1. Taguchi method and optimization problem

Genichi Taguchi (1924-2012) developed a methodology to improve the quality of manufactured goods by using a fractional factorial approach whenever there are several factors involved and it is accomplished with the aid of orthogonal arrays [12]. Taguchi's philosophy is that quality should be achieved through a system design, parameter design and tolerance design by reduction of the deviation from a target. This deviation is influenced by the two types of factors: the controllable factors (called Signal), and the uncontrollable ones (Noise). The *S/N* (called *Signal-to-Noise*) ratio is the indicate of the solution's goodness. In other words, in TM, optimization problem becomes the maximization of *S/N* ratio. Taguchi considered three following performance indices:

Nominal is the best (target problem)

$$S/N_T = 10\log(\frac{\overline{y}^2}{s_y^2}) \tag{1}$$

Larger is the better (maximization problem)

$$S/N_L = -10\log(\frac{1}{n}\sum_{i=1}^{n}\frac{1}{y_i^2}) \tag{2}$$

Smaller is the better (minimization problem)

$$S/N_S = -10\log(\frac{1}{n}\sum_{i=1}^{n}y_i^2) \tag{3}$$

In the equations above, y_i is the i-th value of the performance characteristic and n is the number of trials for a given experiment; $\overline{y}$ is the mean value and s_y is the variance:

$$\overline{y} = \frac{1}{n}\sum_{i=1}^{n}y_i; \quad s_y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \overline{y})^2 \tag{4}$$

The S/N_S is selected in this study, since the objective is the minimum tool path length.

3.2. Case study: The tool path optimization problem

As a case study, the same TPO problem as in [11] is examined, where ACO is applied for minimizing the non-productive (or "air-time") when drilling a hole pattern on a PCB. PCBs usually consist of the most complicated hole pattern because of the large number (may be thousands) of holes and their non-uniform arrangement. Layout of the PCB used in this study is presented in Fig. 1, where holes are numbered only for readability.

In CNC technology, multi-hole drilling operation is related to the so-called *point-to-point control*, under which the path segments connecting each pair of neighbouring points are straight lines. The path from start point to end point is called *total tool path*, or simply *tool path*. The set of points (or nodes, vertices) and segments (or edges, arcs) connecting them establish a circuit (or graph).

In such a way, the TPO problem can be stated as follows: *Find a shortest tool path to drill every hole exactly once*. The meaning of the “shortest” can be minimum path length, moving time or cost. Mathematically, TPO problem is similar to the TSP, which is stated as: *A salesman has to visit N cities, each city exactly once and return back to the starting city with minimum cost (or path length, traveling time)* [13].

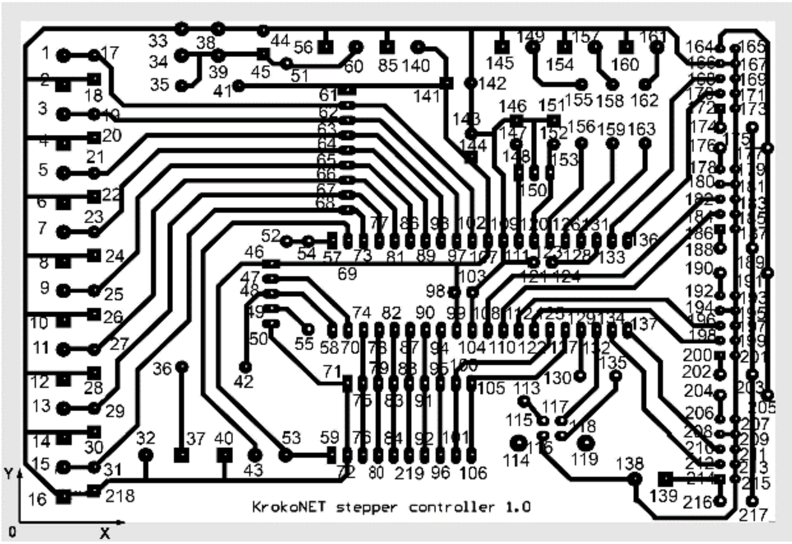


Fig. 1. Hole pattern on a printed circuit board

First, input data, i.e. coordinates of center points from CAD data was exported to a worksheet, that can be read by the ACO program. Such a data sheet looks like the one in Tab. 1, where columns consist the hole numbers and their corresponding coordinates.

Tab. 1. Point data exported from CAD to a worksheet

Input data (Coordinates of center points)

Pt	0	1	2	3	4	5	6	...	213	214	215	216	217	218	219
X	0	7.0	7.0	7.0	7.0	7.0	7.0	...	114.0	111.0	114	111	116.5	12.0	62.0
Y	0	75.0	70.3	65.6	60.9	56.2	51.5	...	8.5	6.0	6.0	3.0	3.0	4.5	10.0

3.3. Experimental research

The main steps for tuning ACO parameters using TM will be described as follows:

Step 1. Taguchi experiment design: As seen in section 2, the ACO parameter tuning problem consists of 3 factors: α, β, ρ and response L_{min}.

In order to cover a sufficiently wide range for parameters (refer to [7]), their levels are set as shown in Tab. 2; that is: α = [0.5 2.5 4.5]; β = [0.5 2.5 4.5]; ρ = [0.1 0.4 0.7]. For TM, the most suitable design is the L9, represented by orthogonal array.

Tab. 2. Factors and their levels

Factors		Level 1	Level 2	Level 3
A	α	0.5	2.5	4.5
B	β	0.5	2.5	4.5
C	ρ	0.1	0.4	0.7

Step 2. Conduction of experiments:

For each experiment, we run a .m-file for 3 times to compute the shortest tool path L_{min}. Other ACO parameters are pheromone update Q = 1, number of ants n = 80, and maximum iterations It_{max} = 120. Number of holes m = 220 is automatically counted from the input data. The complete experiment design is shown in Tab. 3, where the shady columns are the mean of triplicate observations on L_{min} and its corresponding *S/N* ratio, computed by (3).

Randomly all of the observations give the same tool path length L_{min}. Statistics at the bottom of the table shows a wide spread of responses. Some variants, such as the 1st or 7th are very bad, while some others, like 3rd and 9th are quite good. But we still don't know whether any of them may be optimal one.

Tab. 3. Experimental data

Test	Factors			Shortest tool path (L_{min})					It_m
	α	β	ρ	L_1	L_2	L_3	L_m	S/N	
1	1	1	1	5948.5	5948.5	5948.5	5949	-75.5	111
2	1	2	2	1244.8	1244.8	1244.8	1245	-61.9	38
3	1	3	3	1097.4	1097.4	1097.4	1097	-60.8	6
4	2	1	2	2421	2421	2421	2421	-67.7	14
5	2	2	3	1180.7	1180.7	1180.7	1181	-61.4	3
6	2	3	1	1094.4	1094.4	1094.4	1094	-60.8	4
7	3	1	3	3366.7	3366.7	3366.7	3367	-70.5	6
8	3	2	1	1174	1174	1174	1174	-61.4	4
9	3	3	2	1086.4	1086.4	1086.4	1086	-60.7	2
L_m	Max:	5949	Min:	1086	Mean:	2068	Stdev:	1658	

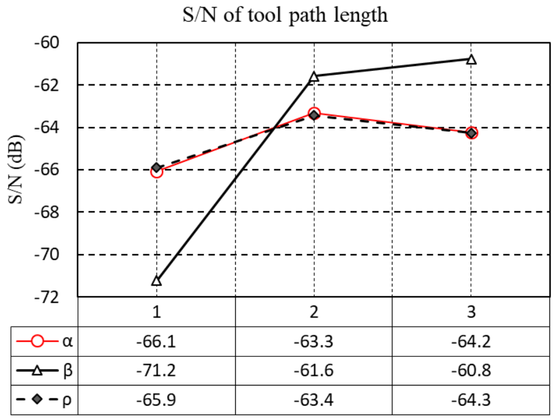
Step 3. Taguchi design analysis:

Taguchi analysis procedure is described in many text-books. This section is focused on how to find out the optimal solution by analysis of *S/N* ratio.

Analysis of *S/N* ratio allows discriminating the effect of each ACO parameter for the different levels. The mean *S/N* ratio for each parameter at levels 1, 2 and 3 can be computed by averaging the *S/N* ratios for correspondent experiments and presented in the *S/N* ratio response table (Fig. 2a). The optimal set of parameters corresponds to factor levels with maximum *S/N* that is $\alpha 2\beta 3\rho 2$, i.e. $\alpha = 2.5$; $\beta = 4.5$; $\rho = 0.4$. Estimated value *S/N* = -58.4 dB and L_{min} = 836 mm. The result is visually presented in the *S/N* ratio response graph in Fig. 2b. The short ANOVA results show that β is the most influencing factor to path length (61.8%), followed by α (20.2%) and the last is ρ (18%). This rank-order agrees with statements in the well-known Dorigo's original text book [7].

S/N ratio response table			
	Mean S/N ratio for L_{min}		
	α	β	ρ
1	-66.1	-71.2	-65.9
2	**-63.3**	-61.6	**-63.4**
3	-64.2	**-60.8**	-64.3
Mean	-64.5		
Max	-63.3	-60.8	-63.4
Max-Mean	1.2	3.8	1.1
Influence (%)	20.2	61.8	18.0
Opt. variant	Variant	S/N	L_{min}
Predicted Value	α2β3ρ2	-58.4	**836**

(a)



(b)

Fig. 2. Taguchi analysis results

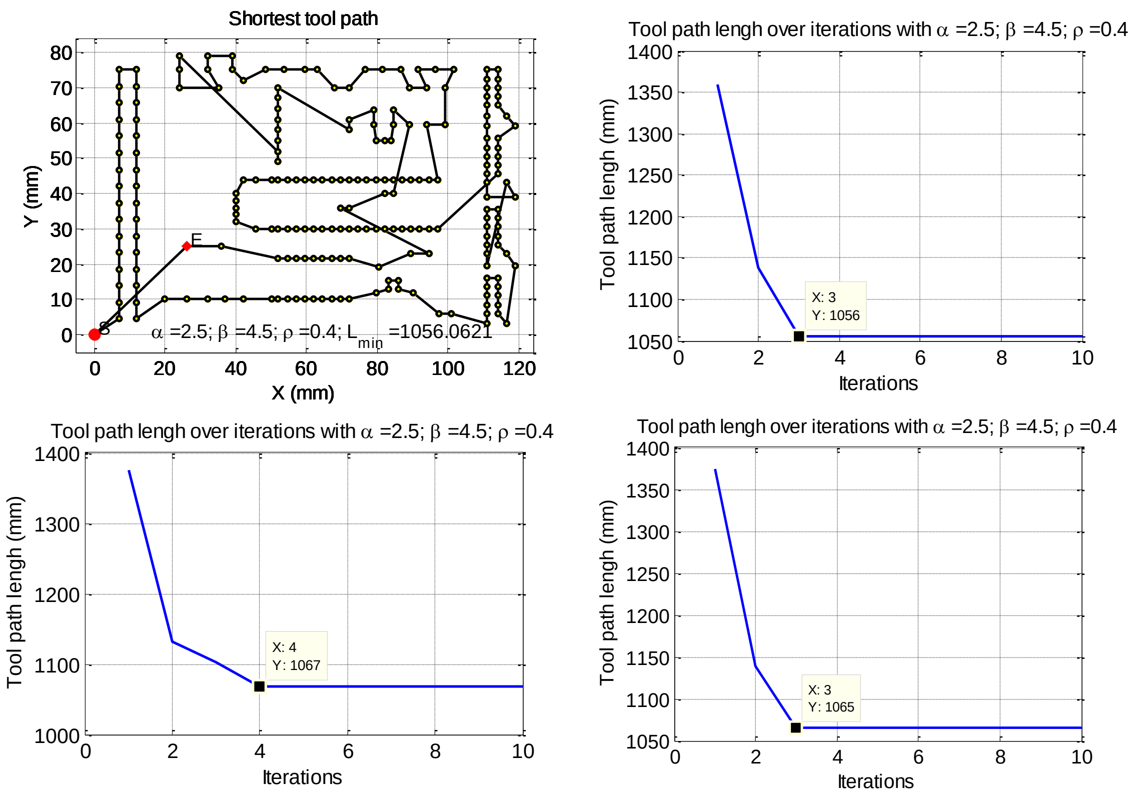


Fig. 3. Confirmation test results

Step 4. Verification experiment: This step involves performing the experiments to confirm the optimization results. The .m-file runs 3 times with the optimal parameters, i.e. $\alpha = 2.5$; $\beta = 4.5$; $\rho = 0.4$. The obtained results are presented in Fig. 3, where the first graph shows one of the real tool path, and remains display the tool path length after iterations.

The results also are summarised in Tab. 4. According to the table, the average fitness value L_{min} = 1063 mm obtained after It_{op} = 3 iterations. Comparison to input data in Tab. 3 shows that the obtained tool path is improved significantly, i.e. L_{min} is even smaller than the smallest of all initial values and is reduced about 2 times in comparison to the mean value (2068 mm). Moreover, the remarkable improvement of convergence rate, i.e. the reduction of iterations (It_{op}) allows us to set the program stop condition *MaxIt* to 10 or less instead of initial one (120) reducing the time for program execution.

Tab. 4. Summary of the confirmation test results in Fig. 3

Summary	α	β	ρ	It_{op1}	It_{op2}	It_{op3}	It_{op}	L_{min1}	L_{min2}	L_{min3}	L_{min}
	2.5	4.5	0.4	3	4	3	3	1056	1067	1065	1063

4. Result Analysis

In order to see how good the obtained optimal solution is, we compare some parameter setting variants for to the same TPO problem, that summarized in Tab. 5. These variants come from available sources. The first one is our article [11], the second is based on the suggestion from [7], that almost all publications in the area of ACO applications refer to, and the last is our optimal solution in this article (Fig. 3).

Tab. 5. Setting variants of ACO parameters

Data sourse	Parameters			Convergence rate (It_{op})				Shortest tool path (L_{min})			
	α	β	ρ	It_1	It_2	It_3	It_m	L_1	L_2	L_3	L_m
[7]	1.0	3.5	0.5	12	13	10	12	1054	1080	1079	1071
[11]	1.0	1.0	0.5	80	-	-	80	1067	-	-	1067
Opt	2.5	4.5	0.4	4	3	3	3	1056	1067	1065	1063

As seen in the table, the last variant is actually the best, which gives the shortest path (1063 mm) while requires the least number of iterations (3).

5. Conclusion

ACO is shown to be the most effective algorithm to solve TSP based combinatorial NP-hard optimization problems, that cannot be exactly solved in polynomial time. However, its performance is very sensitive to parameter settings and difficult to control. In order to overcome such difficulties, TM was applied for tuning ACO parameters α, β, and ρ to obtain the predetermined minimum path length. As a case study, the TPO for drilling a hole pattern on a PCB with 220 holes has been solved. The obtained optimal parameters α = 2.5; β = 4.5; ρ = 0.4 was experimentally confirmed, that ensure the shortest tool path of L_{min} = 1063 mm and a predefined convergence rate of It_{op} = 3 iterations. It is actually the best in comparison to available parameter setting variants. Remarkable improvement of tool path length and convergence rate allows us to reduce not only time for drilling but also the one for program execution.

References

1. Brezina I. J., Čičková Z. (2011). Solving the Travelling Salesman Problem Using the Ant Colony Optimization. *Management Information Systems, 6*, 10-14.
2. Blum C. (2005). Ant Colony Optimization: Introduction and Recent Trends - Review. *Physics of Life Reviews*, 353-373.
3. Narooei, K. D. and Ramli, R. (2014). Application of Artificial Intelligence Methods of Tool Path Optimization in CNC Machines: A Review. *Research Journal of Applied Sciences, Engineering and Technology, 8*(6), 746-754.
4. Narooei K. D., et al. (2014). Tool Routing Path Optimization for Multi-Hole Drilling Based on Ant Colony Optimization. *World Applied Sciences Journal, 32*(9), pp. 1894-1898.
5. Srinivas P. S., Raju V. R., Rao C. S. P. (2012). An Ant Colony Optimization Algorithm for Optimization of Process Plans. *International Journal of Mechanical Engineering and Robotics Reseach*, *1*(3), 31-42.
6. Wang J. F., Fan X. L., Ding S. (2014). An Improved Ant Colony Optimization Approach for Optimization of Process Planning. *The Scientific World Journal*, Vol. 2014, 1-15.
7. Dorigo M., Stützle T. (2004). *Ant Colony Optimization.* MIT Press.
8. Gaertner D. and Clark K. (2005). On Optimal Parameters for Ant Colony Optimization Algorithms. *Proceedings of the 2005 International Conference on Artificial Intelligence (ICAI)*, 83-89.
9. Shweta K. M., Singh A. (2013). An Effect and Analysis of Parameter on Ant Colony Optimization for Solving Travelling Salesman Problem. *International Journal of Computer Science and Information Technology, 2,* 222-229.

10. Stodola P., Mazal J., and Podhorec M. (2015). Parameter Tuning for the Ant Colony Optimization Algorithm used in ISR Systems. *International Journal of Applied Mathematics and Informatics*, *9*, 123-126.

11. Dao Van Hiep, Dao Trung Kien (2018). Tool Path Optimization for Hole Pattern Making on Printed Circuit Boards by Combination of TSP and ACO. *Journal of Science and Technique - Military Technical Academy*, *191* (6-2018), 40-49.

12. Cavazzuti, M. (2013). *Optimization Methods to Design Scientific and Technological Aspects in Mechanics*, Springer.

13. Applegate D. L., Bixby R. E., Chvátal V. & Cook W. J. (2007). *The Traveling Salesman Problem*, Princeton University Press.

TỐI ƯU HÓA CÁC THAM SỐ CỦA THUẬT TOÁN ĐÀN KIẾN NHỜ PHƯƠNG PHÁP TAGUCHI

Tóm tắt: *Trong bài báo này, phương pháp Taguchi được áp dụng để tối ưu hóa các tham số của giải thuật tối ưu hóa (TƯH) đàn kiến (ACO). Bài toán TƯH đường chạy dao khi khoan mảng lỗ trên bảng mạch in 220 lỗ được lấy làm ví dụ. Kiểm chứng bằng thực nghiệm cho thấy, với các tham số tối ưu của ACO, chiều dài đường chạy dao giảm khoảng 2 lần và tốc độ hội tụ được cải thiện đáng kể so với giá trị trung bình trong dữ liệu ban đầu. Phương án thiết đặt tham số này cũng được xác thực là tốt nhất so với các phương án thông thường. Điều đó cho phép giảm không chỉ thời gian gia công mà cả thời gian chạy chương trình.*

Từ khoá: Phương pháp Taguchi; tối ưu hóa đàn kiến; tối ưu hóa đường chạy dao; bài toán người bán hàng; bảng mạch in.

Received: 02/11/2018; *Revised:* 30/12/2018; *Accepted for publication:* 27/02/2019

❑