Kursat Tanriver ✉
Mustafa Ay

https://doi.org/10.21278/TOF.483062023
ISSN 1333-1124
eISSN 1849-1391

EFFICIENT PATH PLANNING FOR DRILLING PROCESSES: THE HYBRID APPROACH OF A GENETIC ALGORITHM AND ANT COLONY OPTIMISATION

Summary

Efficiency in machining time during drilling is affected by various factors, with one key element being the machining path. Solving the machining path closely resembles the Travelling Salesman Problem (TSP). In this article, drilling on a sample model is simulated using a hybrid algorithm that is developed based on TSP. This hybrid algorithm (GACO) is created by combining the strengths of the Genetic Algorithm (GA) and Ant Colony Optimisation (ACO). Codes written to verify the stability of the algorithms were executed 10 times, and results were recorded indicating the shortest path and machining sequence. Accordingly, the performance of the hybrid GACO algorithm was observed to be 3.16% better than the ACO algorithm in terms of both total path length and total machining time. In terms of computation time, the ACO algorithm lagged behind the GACO algorithm by 6.46%. Furthermore, the hybrid GACO algorithm demonstrated enhanced performance in both total path length and total machining time when compared with the literature. This study aims to contribute to the industry, professionals, and practitioners in this field by providing cost and time savings.

Key words: *ant colony; drilling; machining; tool pathing optimisation; travelling salesman person*

1. Introduction

Optimisation is a method generally used to achieve the best possible result within determined goals and constraints. Optimisation techniques are often divided into two categories: mathematical and heuristic methods. Mathematical methods strive to find the most accurate analytical solution, while heuristic methods exhibit a more practical approach [1].

In scientific research and production, despite the rapid increase in the computational power of computers, the size of optimisation problems has increased correspondingly [2]. To overcome this difficulty, various algorithms are written by utilising heuristic optimisation techniques, depending on the problem. Heuristic algorithms are frequently used in solving the travelling salesman problem (TSP). TSP was first defined as a method to find the minimum distance tour where the starting and ending cities are the same, and each city is visited only once, initially in an inter-city problem [3]. Some heuristic algorithms are frequently used to solve path problems. These include ant colony optimisation (ACO) [4], genetic algorithms (GA) [5], particle swarm optimisation (PSO) [6], bat algorithms (BA) [7], and firefly algorithms (FA) [8]. Moreover, in recent years, many hybrid algorithms have been proposed by combining these algorithms to solve path problems [9]. Hybrid algorithms are created by taking the relatively better aspects of heuristic algorithms, depending on the characteristics and complexity of the problem being solved. Recently, hybrid algorithms used in path problems have also started to be employed in industrial robots. Industrial robots, which have a significant place in the manufacturing and aerospace industries, play an essential role in various production areas such as chip removal, welding, and painting. Industrial robots used in chip removal manufacturing types like drilling and milling have significant advantages in complex shapes and low-precision applications [10].

This study originated from the idea of solving path planning for drilling in industrial robots or machining tools using the Travelling Salesman Problem (TSP). The resolution of the TSP planning problem aims to minimise the machining time by determining the shortest path. An algorithm was written and run to achieve the shortest path between holes on a sample part. This algorithm, named GACO, is a hybrid algorithm created by considering the advantageous aspects of ACO and GA. Prior to machining on the piece, users can gain insights through simulation using this software. Thus, the goal is to contribute to cost and time savings for the industry, professionals, and users in this field.

In the manufacturing and aviation industries, the challenging nature of engineering problems and their wide areas of application have led to numerous studies seeking solutions [11]. It is known that as the number of variables in problems increases, finding solutions using traditional methods becomes more difficult. Efforts have been made to simplify the solution and find the best one, leading to the development of meta-heuristic algorithms. One such meta-heuristic algorithm, ACO [12], has been used to solve many application problems in different areas of daily life in conjunction with TSP [13]. TSP was first analytically described in 1930 and has been frequently put to use since the 1950s. It has been successfully used to solve various problems to this day [14].

However, hybrid studies created by combining GA and ACO algorithms are limited compared to the basic ACO algorithm. For example, Sezer et al. [15] conducted a study using a hybrid approach to solve the TSP problem with the idea of delivery using trucks or drones. They proposed a hybrid algorithm combining a genetic algorithm and ant colony optimisation algorithm. Unlike those in the literature, the proposed meta-heuristic method constructed both truck and drone routes simultaneously. Computational experiments show that the proposed algorithm can generate the optimal routes compared to those in the literature. Oleiwi et al. [16] developed a hybrid approach based on ACO and GA to solve the multi-objective path planning problem of mobile robots. The simulation results of this approach demonstrated that the mobile robot successfully navigated from one location to another in all tested environments, avoiding all obstacles along its path, thus validating the effectiveness of the proposed approach. Similarly, Wu et al. [17] conducted a hybrid ACO study aimed at solving the vehicle routing problem to reduce travel distance and transportation costs. A hybrid ant colony optimisation algorithm based on the ant colony algorithm and mutation operation was proposed for solving this problem. The proposed algorithm introduced three innovations: firstly, updating pheromones using a new method; secondly, introducing adaptive parameters; and thirdly, adding the mutation operation. The experimental results demonstrated the effectiveness of the proposed algorithm in solving the vehicle routing problem. Wang et al. [18] solved the Travelling Salesman Problem (TSP) for a resource robot's shortest path using a Monte Carlo-based hybrid ACO algorithm. Therefore, reflecting the advantages and disadvantages of the node in the first iteration was rather difficult and required multiple iterations to gradually demonstrate the advantages and disadvantages of the path. To address this limitation, Monte

Carlo-based ant colony optimisation (MC-IACO) was proposed to solve the path planning of resource robots. The simulation results indicated that the MC-IACO algorithm presented in this paper significantly improves the solution of the resource robot's path planning problem compared to basic ant colony optimisation and other advanced ant colony optimisation methods, demonstrating overall better performance. The solution of TSP problems can be applied to various problem domains, including algorithm selection based on feature normalisation [19], minimising total travel costs [20], maintenance scheduling of gas turbines [21], scheduling problems [22], vehicle routes [23], and locating ships with drones [24].

Some of the studies on machining parameters are as follows: Köklü et al. [25] investigated issues related to delamination, poor surface quality, and tool wear during the machining of composite materials. Their findings show the orientation of the material in the composite affected delamination by 89.5%, the cutting force by 0.1%, and vibration by 8.4%. Additionally, they conducted a study using artificial neural networks to predict drilling reactions or cutting parameters over a wider range. Morkavuk et al. [26] conducted experiments to investigate the machinability properties of carbon fibre-reinforced polymer composites (CFRP) and presented their results comparatively. Additionally, they achieved significant improvement in delamination damage behaviour due to drilling in composite pipes with the use of supports, even though there was no increase in cutting force.

Additionally, detailed examinations have been conducted on studies focusing on modelling machining parameters in drilling processes [27] [28] [29], conducting machinability experiments on difficult-to-machine steels [30], hybrid composite drilling methods [31], planning paths in robotic milling [32], optimising machinability parameters [33], and drilling processes in composites [34].

The studies mentioned above were individually examined, serving as inspiration and forming the basis for the present study. It was observed in this comprehensive research that the efficiency of drilling applications has not been considered from the perspective of tool path effectiveness. While the tool path exhibits similarities with studies conducted on TSP, no specific study focusing on tool path optimisation in drilling operations was encountered. Although comparisons have been made in some of the TSP examples, such as resource allocation, robotics, drones, and vehicle routing in the mentioned studies, there is a lack of comparison or validation with similar studies in the literature. It has also been noted that further improvements in parameters are necessary in studies that conduct comparisons. Taking into account the positive aspects, open topics, and shortcomings identified through comprehensive literature reviews, this article refines the drilling problem. The conception of the problem emerged from the idea of solving drilling path planning in industrial robots or machining tools using the Travelling Salesman Problem (TSP). For this purpose, a hybrid algorithm named GACO has been developed, aiming to solve the TSP planning problem on a sample model and to obtain the shortest path between the holes. The results of the GACO algorithm is presented not only through its comparison with basic ACO internally but also in comparison with similar studies in the literature. Further, the deficiencies in the parameters identified during the literature review have been eliminated and adapted to the GACO algorithm, and the algorithm has been enhanced accordingly.

In sum, this article aims at a solution to the TSP planning problem utilising the GACO algorithm, with the objective of minimising machining time by determining the shortest path. This contribution is geared towards enhancing production efficiency within competitive contexts and adding to the existing literature. Through simulation with this software, insights can be gained by users before machining the modelled part. Thus, the goal is to contribute to cost and time savings for the industry, professionals, and users working in this field.

2. Methodology

By utilising the information accumulated in the form of pheromone trails laid by artificial colony ants, consecutively shorter feasible tours are formed, and thus, at the end of the specified iteration, the determined shortest tours are recorded to arrive at the TSP solution. Computer simulations using programming languages have shown that they can produce good solutions to TSP examples [35].

TSP is a commonly identified optimisation problem with ACO and GA. In this study, the goal was to combine these two methods to achieve a more effective and reliable solution. Additionally, codes were written for this method, called hybrid GACO, and for the basic ACO method. The written codes were simulated using MATLAB software to test whether or not they worked.

2.1 ACO algorithm

The ACO algorithm is based on a random search that emulates the behaviour of ants. The decision-making process in the search is conducted using an artificial chemical compound called pheromone, which represents the quality of a solution. In the ACO algorithm, ants work together to find the most suitable path. Each ant traverses the existing paths randomly. The ants accumulate pheromones along the path, and the amounts of pheromones on the path are updated in each iteration. Thus, the goal is to select the path with the most pheromone traces, thereby choosing the most suitable path [36]. The fundamental equations for standard ACO can be defined as the probability of exploration and pheromone updating.

2.1.1 Transition probability

In the probability of exploration, the likelihood P of the ants selecting the next node is expressed in the following equation. In other cases, the value is taken as 0.

$$p_{i,j} = \frac{[\tau_{i,j}]^{\alpha} \ [\eta_{i,j}]^{\beta}}{\sum_{k \in N_i} [\tau_{i,k}]^{\alpha} \ [\eta_{i,k}]^{\beta}} \tag{1}$$

$$\eta_{i,j} = \frac{1}{d_{i,j}} \tag{2}$$

$$d = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \tag{3}$$

Here, i and j are the current node and a possible next node, respectively; α and β are the coefficients for pheromone accumulation and the heuristic parameter that controls the weighting of the distance; $\tau(i,j)$ is the amount of pheromone between nodes i and j; $\eta(i,j)$ is the heuristic value of visibility between nodes i and j; N refers to the nodes or cities that can be visited; $x,y(i,j)$ and $y(i,j)$ denote the coordinates of the nodes or visited cities relative to the starting point.

2.1.2 Transition probability

An update of the pheromone quantity is done after the ants complete their tours. In the initial phase, all the pheromones on the paths evaporate at a determined evaporation rate. In the next stage, the amounts of pheromones left on the paths by the ants are identified. The pheromone quantity is increased inversely proportionally to the total path length used by the ant. Thus, the paths used by ants with shorter distances show more increases in the amounts of pheromones.

The pheromone update can be performed in two ways: locally and globally. Local pheromone updating, denoted by $\tau_{ij}(t)$, is expressed in the equation below for an ant moving from the *i*th node to the *j*th node:

$$\tau_{i,j}\ (t) = (1 - \rho')\tau_{i,j}(t) + \ \rho'\tau_0 \tag{4}$$

Here, ρ represents the local evaporation coefficient, and τ_0 denotes a positive constant value.

Global pheromone updating refers to the updating of pheromones after all ants have completed the current iterative search. In this context, the accumulated pheromone amount $\tau_{ij}(t)$ up to iteration t, the amount of the pheromone increase in iteration t, $\Delta\tau_{ij}(t)$, and the evaporation coefficient ρ ($0 \le \rho \le 1$) are used to update the global pheromone quantity as expressed in the following equation:

$$\tau_{i,j}\ (t+1) = (1 - \rho)\tau_{i,j}(t) + \textstyle\sum_{k=1}^{m} \Delta\tau_{i,j}(t\ ) \tag{5}$$

The value of $\Delta\tau_{ij}\ (t + 1)$ for ant k can be calculated as expressed in the following equation:

$$\Delta\tau_{i,j}\ (t)) = \begin{cases} \frac{Q}{L^k}\ ,\ if\ j\ \in N\ and\ path\ ij\ \text{solutionarray}_k \\ 0,\ \ otherwise \end{cases} \tag{6}$$

$$L_k = \textstyle\sum_{i=1}^{N-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \tag{7}$$

Here, Q represents the strength of the pheromone, L_k represents the length of the path taken by ant *k* in the relevant iteration, and *N* represents the number of nodes. The ACO procedure is shown in Fig. 1a.

2.2 GA algorithm

GA (Genetic Algorithm) is a search algorithm based on evolutionary theory and is used to determine the most suitable individuals in a population. For this purpose, genetic operators called crossover and mutation are used. GA is generally used in the solution of complex optimisation problems. Before starting to write GA, the coding format must first be decided on. Binary coding, permutation coding, and real-valued coding are commonly used. Binary coding consists of 1 and 0 values, and the variables are represented in binary order, consisting of a number of genes determined according to the value range. Permutation coding is used in the solution of problems such as the shortest path, travelling salesman, and vehicle routing, where ordering is important, and repetition is not possible. Real-valued coding refers to the coding form in which the variables are represented directly by their values.

In GA, two basic operations called crossover and mutation are usually performed. In the crossover operator, two individuals (parents) are selected, and their genes are mixed at one or more crossover points. This mixing process is done to create a new individual (child). In the mutation operator, a specific gene of a selected individual is changed randomly. This helps prevent the algorithm from getting stuck in a local optimum and broadens the solution space.

2.3 Hybrid GACO algorithm

Hybrid algorithms can be formed by combining the advantages of each algorithm with different techniques. This way, the relatively weak aspects of the algorithms according to the problem being solved can be minimised, and the overall effect of the algorithm can be improved. In the solution of the TSP (Travelling Salesman Problem), ACO's (Ant Colony Optimisation) exploration capability usually provides an effective solution. The hybrid GACO

(Genetic Algorithm with Ant Colony Optimisation) algorithm is an approach that combines the ACO algorithm with the genetic operators of the GA (Genetic Algorithm). This approach provides some advantages aimed at enhancing the algorithm's performance and the quality of the solution. The hybrid GACO algorithm is formed by combining the GA's global search capability and the ACO's local search capability. In other words, the GA's crossover and mutation operators are used as part of ACO's pheromone update and local search steps.

Within the hybrid GACO, the genetic algorithm operators integrate the crossover and mutation processes into the ACO algorithm. This integration helps to provide genetic diversity within the population and aims to explore better solution spaces. In the hybrid GACO algorithm, in addition to the ACO algorithm, there is a genetic population. This population is directed by the genetic algorithm's crossover and mutation operations, enabling broader exploration in the solution space, and investigating better solution possibilities. The hybrid GACO algorithm performs parameter adjustments using genetic algorithm operators while seeking solutions using the ACO's fundamental mechanism. This allows for better optimisation of the algorithm parameters. The hybrid GACO algorithm is provided in Fig. 1b.

Algorithm 1 ACO Algorithm .	
1:	**input:** max_iterations, num_ants, evaporation_rate (ρ), alpha (α), beta (β)
2:	**output:** best_solution
3:	pheromone_matrix = initialize_pheromone_matrix()
4:	best_solution = None
5:	for iteration in range(max_iterations):
	ants = create_ants(num_ants)
6:	for ant in ants:
	ant.tour (pheromone_matrix, alpha, beta)
7:	update_pheromone (pheromone_matrix, ants, evaporation_rate)
8:	new_best_solution = find_best_solution(ants)
9:	if new_best_solution is better than best_solution:
	best_solution = new_best_solution
10:	**print**(best_solution)
11:	**return** best_solution

(a)

Algorithm 2 Hybrid GACO Algorithm.	
1:	**input:** max_iterations, ant_population, evaporation_rate (ρ), alpha (α), beta (β), ga_population_size (P), crossover_rate (C), mutation_rate (M)
2:	**output:** best_solution
3:	ants = initialize_ant_population(ant_population)
4:	ga_population = initialize_ga_population(ga_population_size)
5:	best_solution = None
6:	**for** iteration in range(max_iterations):
	for ant in ants:
	calculate(ant)
7:	ga_population = crossover(ga_population, crossover_rate)
8:	ga_population = mutate(ga_population, mutation_rate)
9:	update_pheromones(ants, evaporation_rate)
10:	new_best_solution = calculate_max(ants, ga_population)
11:	if new_best_solution is better than best_solution
12:	best_solution = new_best_solution
13:	**print** (best_solution)
14:	**return** best_solution

(b)

Fig. 1 Algorithms (a) ACO (b) GACO (hybrid)

Here, instead of specifying the calculate process and the calculate max function separately, the expression "calculate" is used to represent the general computation processes. Similarly, instead of the expression "list", the functions "initialize_ant_population" and "initialize_ga_population" are used to create the populations for the ants and the genetic algorithm, respectively. The statement "best solution = None" is used to indicate that the best solution was not determined at the start, and this value is altered within the algorithm once the best solution is found after an update. The parameter definitions are given in Table 1.

Table 1 Parameters definition

Parameters definition	
max_iterations	The maximum number of iterations for the algorithm
ant_population	The size of the ant population
evaporation_rate (ρ)	The rate at which the pheromone evaporates
alpha (α)	The parameter that controls the importance of pheromone in the ant's decision-making process
beta (β)	The parameter that controls the importance of distance in the ant's decision-making process.
ga_population_size (P)	The size of the genetic algorithm population
crossover_rate (C)	The probability of crossover operation in the genetic algorithm
mutation_rate (M)	The probability of mutation operation in the genetic algorithm

A flowchart illustrating the basic operation of the hybrid GACO algorithm is shown in Fig. 2.

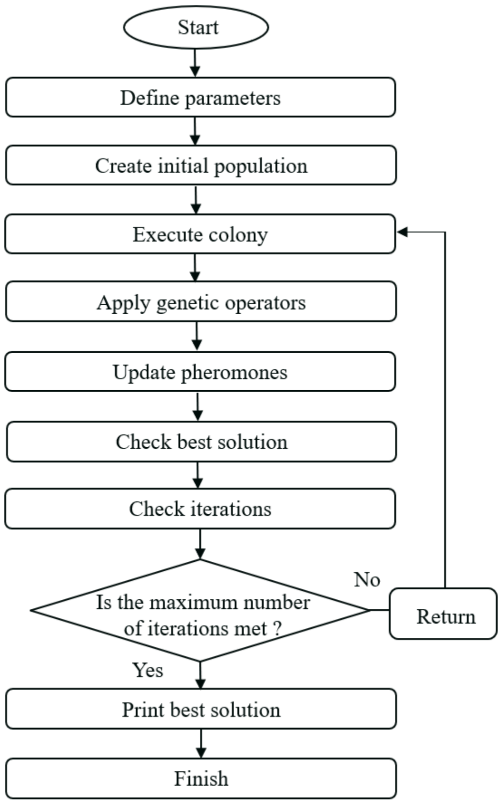


Fig. 2 GACO algorithm flowchart

The basic steps illustrating the operation of the hybrid GACO algorithm are listed below.

Step 1: Definition of Parameters. This step involves defining the necessary parameters for the algorithm to operate. These include the number of iterations, number of ants, evaporation rate, alpha, beta, size of the gene population, crossover rate, and mutation rate.

Step 2: Creation of the Initial Population. This covers information on the ant colony and the genetic algorithm population.

Step 3: Execution of the Colony. This expects the ants to visit all nodes and try to find the shortest path. Ants make choices as they move to a new node at each step. These choices consider pheromone trails and distances between nodes, often using a probability distribution. The ants remember the nodes they visit and the distances between them, which will be used for pheromone updating. They work independently to find a path but may be subject to some constraints when choosing the next unvisited node. These constraints can be adjusted according to specific criteria or rules related to the problem. If an ant reaches an unsuitable node (e.g., non-compliance with a constraint or revisiting a previously visited node), it backtracks to choose an alternative path. In this way, ants aim to discover the shortest path by exploring different routes.

Step 4: Application of Genetic Operators. At this stage, crossover and mutation operators are applied to the best paths.

Step 5: Updating of Pheromones. After the execution of all the ants, the pheromone update process is performed.

Step 6: Checking the Best Solution. If the solution found is better than the best solution so far, it is recorded as the best solution.

Step 7: Checking Iterations. Whether or not the determined maximum number of iterations has been reached is checked. If reached, the algorithm is terminated. If not, it returns to the step of applying genetic operators.

Step 8: Printing the Best Solution. When the algorithm is completed, the best solution found is printed.

This series of steps presents a comprehensive description of how the hybrid GACO algorithm operates, integrating aspects of Genetic Algorithms (GA) with Ant Colony Optimisation (ACO) to explore and optimise solutions to complex problems.

3. Result and discussion

3.1 Modelling of drilling with GACO

Carbon fibre-reinforced polymer composites (CFRP) are materials used in the manufacturing and aerospace industries, and are characterised by being low cost, lightweight, durable, and having reduced part usage. These materials often undergo drilling processes to facilitate the assembly of two parts. To achieve optimal performance in the drilling process, it is necessary to know certain criteria, including the cutting parameters, tool geometries, and the online and offline parameters. The steps of the study related to the performance criteria of composite materials in drilling operations are also shown in Figure 3 [37].

Since changes in production time are directly proportional to the costs, the optimisation of the machining parameters affecting this time has become increasingly important. In order to shorten the machining time, less machining length is required. Designers are searching for solutions to increase efficiency and to have prior knowledge of the machining time. For this purpose, virtual machining simulation is required, which assists in planning reasonable machining paths. Consequently, one of the performance criteria of composite materials is machining parameters. This article emphasises that the movement time in the drilling process with CNC machines or industrial robots affects machining performance and that machining time should be part of the performance criteria. Based on this and the fact that processes exhibit very similar features to the TSP, an algorithm considering machining time is presented. The presented algorithm, named "Operation time (GACO)", is added to revise the performance criteria, as shown in Figure 3b.

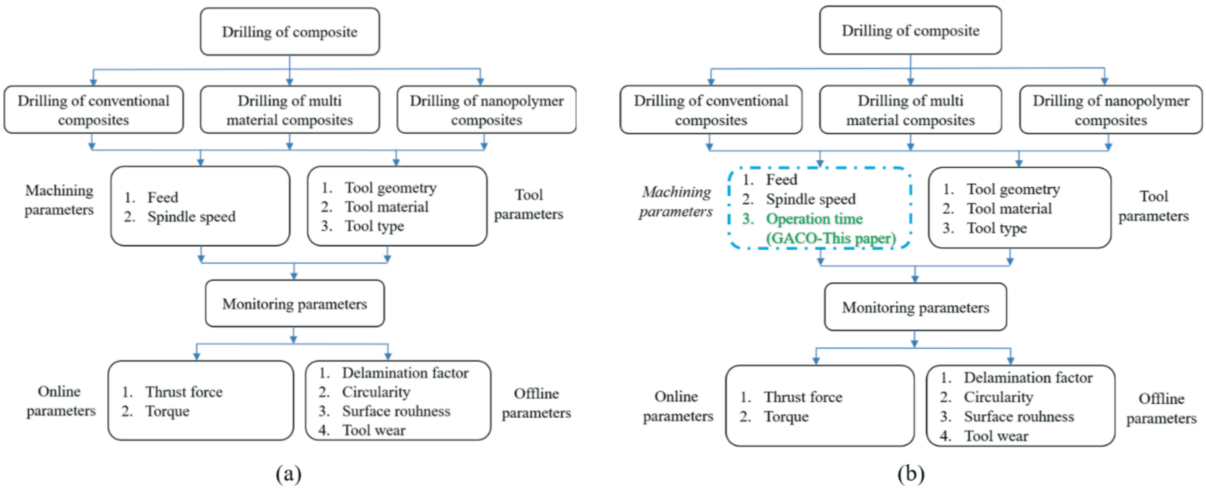


Fig. 3 Composite in drilling (a) material performance criteria [37] (b) material performance criteria with operation time (GACO- This paper)

When examining the figure, the study focuses on three types of composites: conventional composites, multi-material composites, and nano-polymer composites. From this, it emerges that the efficiency of the machining time will depend not only on the drilling time but also on the type of composite being processed. Therefore, the algorithm presented for the given problem is primarily based on the shortest path, movement speed, and drilling time, as expressed in Equation 8:

$$\sum operation\ time\ = \frac{best\ path}{moving\ speed} + \sum drilling\ time \tag{8}$$

In this algorithm, when the type of composite material changes, the machining time also changes. In this paper, the algorithm is run considering the properties of the CFRP material from the conventional composites material family as an example. When the composite material changes, the drilling time can be adjusted to obtain the calculation result according to the desired material. Under the same conditions, the drilling machining time is theoretically optimised for the tool path. Outputs such as delamination, tool wear, dimensional tolerances, etc., are not taken into account. The drilling time for the materials is calculated from the catalogues and studies [38] [39] [40] [41] and added to the algorithm, as these output parameters will be evaluated in another study.

The hybrid GACO algorithm can calculate the machining time on a two-axis CNC machine or a three-axis industrial robot machine, as illustrated in Figure 4. The proper definition of coordinates is sufficient within the algorithm. An industrial robot example is considered to demonstrate the algorithm's operation in three different planes. The model is scaled to a thickness of 10 mm and designed from CFRP material with dimensions of 250x250x250 mm. The anticipated parameters [26] and drill diameters (DIN 6537) for this example are compatible with the literature. The coordinates of the 25 holes on the sample part are presented in Table 2.

Some assumptions are made for each drilling operation in the algorithm. The drilling process starts by machining from a randomly selected drilling point for the initial hole. Subsequently, it traverses each of the 25-hole points on the part in sequence to perform the drilling operation. It can only pass through each drilling point once. After completing the tour of all hole points, the algorithm returns to the starting point. Other assumptions made for the generation of the machining time are summarised below.

- The machining time for each hole in the CFRP material is taken as 10 seconds based on the average feed rate calculated from the manufacturer catalogues and studies [38]

[39] [40] [41] (output parameters such as delamination, tool wear, dimensional tolerances, etc., are neglected and are planned to be evaluated in another study).

- To ensure clearer results, a constant average speed of 2 mm/s is assumed when moving to the next hole after drilling [26].
- A scaled model design is adopted to simplify the problem model during modelling. Additionally, some obstacles that may affect the operation of the robot under real working conditions are disregarded in this model.
- Online parameters (such as thrust force, torque) and offline parameters (such as tool wear, delamination, surface roughness, circularity) are neglected (the evaluation of the monitoring parameters is planned for another study).

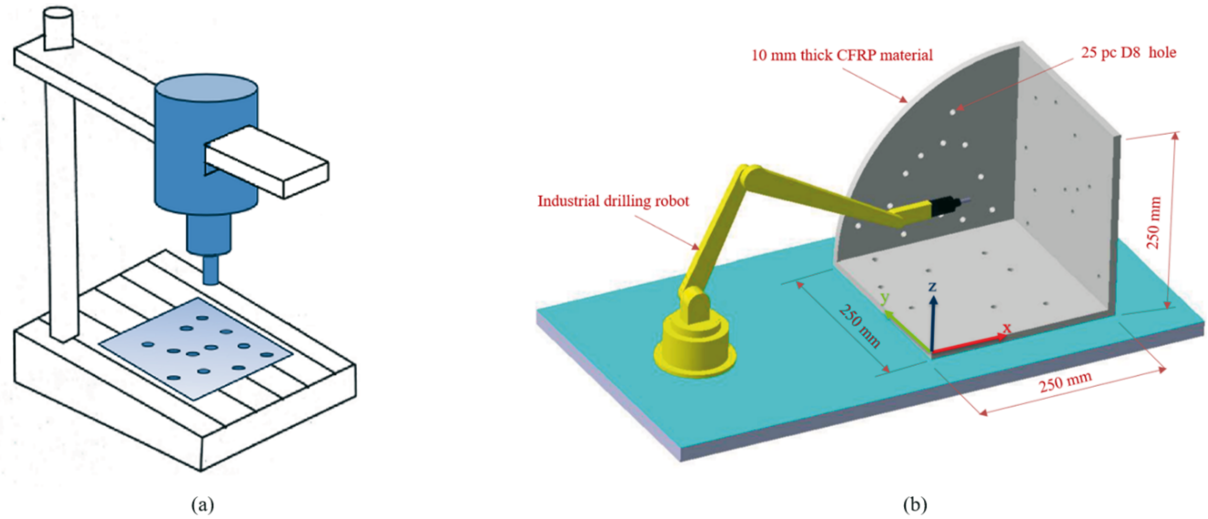


Figure 4 3D model (a) CNC drilling machine (b) industrial robot

Table 2 Coordinate data of drilling points

Coordinates	Number of hole positions (mm)																								
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
x	27	70	90	90	140	140	150	170	170	0	0	0	0	0	0	0	0	0	0	20	30	45	90	120	180
y	0	0	0	0	0	0	0	0	0	205	25	140	190	95	115	215	85	55	95	165	105	225	175	65	45
z	42	60	16	100	30	130	170	16	120	10	32	175	65	85	105	110	120	140	190	0	0	0	0	0	0

3.2 GACO algorithm simulation for drilling

The parameter values are taken as the same for both algorithms. Here, the maximum iteration number is 1000, the ant number 25, the evaporation rate (ρ) 0.5, alpha (α) 1, and beta (β) 2. Additionally, in the hybrid GACO algorithm, the mutation rate (M) is 0.1, the population size (P) 25, and the crossover rate (C) 0.8. The parameter values are shown in Table 3.

The codes were written in the MATLAB program to evaluate the performance of the algorithms. To verify the stability of the GACO and ACO algorithms, the algorithms were run 10 times, and the results were recorded. To better understand the obtained values, the results are given as computation time(s), total path length (mm), and total operation time (s) in Table 4.

Table 3 Parameter values

Parameter	ACO	GACO
max iteration	1000	1000
ant number	25	25
evaporation rate (ρ)	0.5	0.5
alpha (α)	1	1
beta (β)	2	2
mutation rate (M)	-	0.1
population size (P)	-	50
crossover rate (C)	-	0.8

Due to the nature of heuristic algorithms, different results emerged from each other in this example. The best sample values among these results were identified. In the ACO algorithm, the computation time is 26.4101 s, the total path length is 1456.8 mm, and the total operation time is 754.4146 s. In the GACO algorithm, the computation time is 28.2339 s, the total path length is 1412.1 mm, and the total operation time is 732.0635 s. Comparisons can be made between the best values identified in the ACO and GACO algorithm experiments. The percentage rate equations are given below for comparison:

$$\text{Rate}_{\text{computation time}} = \frac{\text{GACO}_{\text{value}} - \text{ACO}_{\text{value}}}{\text{GACO}_{\text{value}}} \text{ x } 100 \tag{9}$$

$$\text{Rate}_{\text{total path length}} = \frac{\text{GACO}_{\text{value}} - \text{ACO}_{\text{value}}}{\text{GACO}_{\text{value}}} \text{ x } 100 \tag{10}$$

$$\text{Rate}_{\text{operation time}} = \frac{\text{GACO}_{\text{value}} - \text{ACO}_{\text{value}}}{\text{GACO}_{\text{value}}} \text{ x } 100 \tag{11}$$

Using these equations, it can be observed that the performance of the hybrid GACO algorithm is 3.16% better than the ACO algorithm in both total path length and total machining time. In the MATLAB program, it is observed that the ACO algorithm is 6.46% behind the GACO algorithm in terms of computation time. The machining order for the ACO algorithm to find the shortest path between holes is [1 11 14 15 17 18 19 12 16 13 10 22 23 20 21 24 25 8 5 9 6 7 4 2 3 1]. In the GACO algorithm, which is better than the ACO algorithm in terms of the best path length, the machining order for the shortest path between holes is [23 20 22 10 13 16 12 19 18 17 15 14 11 1 2 4 7 6 9 3 5 8 25 24 21 23].

Table 4 Comparison of the GACO and ACO results

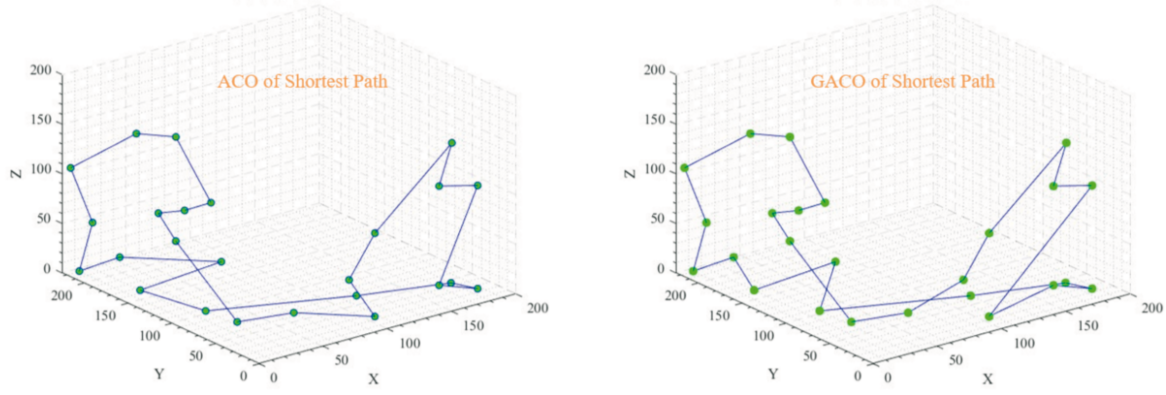
No	ACO (Best result in 10 runs)			GACO (Best result in 10 runs)		
	Computation time (s)	Total path length (mm)	Total operation time (s)	Computation time (s)	Total path length (mm)	Total operation time (s)
Run 1	28.1269	1484.6	768.2818	29.2568	1433.7	742.8537
Run 2	40.8015	1536.4	794.1846	29.0327	1510.0	780.9842
Run 3	26.5317	1526.0	788.9968	27.8547	1437.6	744.8111
Run 4	26.4101	1456.8	754.4146	28.7735	1489.7	770.8424
Run 5	27.7126	1484.8	768.4215	28.2339	1412.1	732.0635
Run 6	27.0493	1518.2	785.0979	28.8876	1479.3	765.6678
Run 7	27.0174	1550.8	801.3777	29.1493	1510.0	781.0044
Run 8	43.8033	1507.7	1456.8	28.3509	754.4146	781.0760
Run 9	35.7203	1487.0	769.4812	33.1687	1451.7	751.8546
Run 10	39.7664	1523.5	787.7366	29.3628	1532.0	792.0108
Best run	26.4101	1456.8	754.4146	28.2339	1412.1	732.0635

The shortest path machining order for ACO is shown in Figure 5a, and the shortest machining order for GACO in Figure 5b. Additionally, the ACO path length-iteration graph is shown in Figure 6a, and the GACO path length-iteration graph in Figure 6b. Additionally, a comparison was made with studies in the literature for the performance of the hybrid GACO algorithm, which was found to have a better performance in terms of both total path length and total machining time. In the literature research, the ratios between the algorithms developed in the studies themselves and the normal ant colony algorithm were found based on equation 10. The best path ratios were determined as -0.56% by Wang et al. [35], 3.17% by Song at al. [36], 5.96% by Wu et al. [17], and 16.60% by Song et al. in another study [36]. A comparison of the best path ratios is given in Table 5:

Table 5 Comparison of the GACO result with the literature

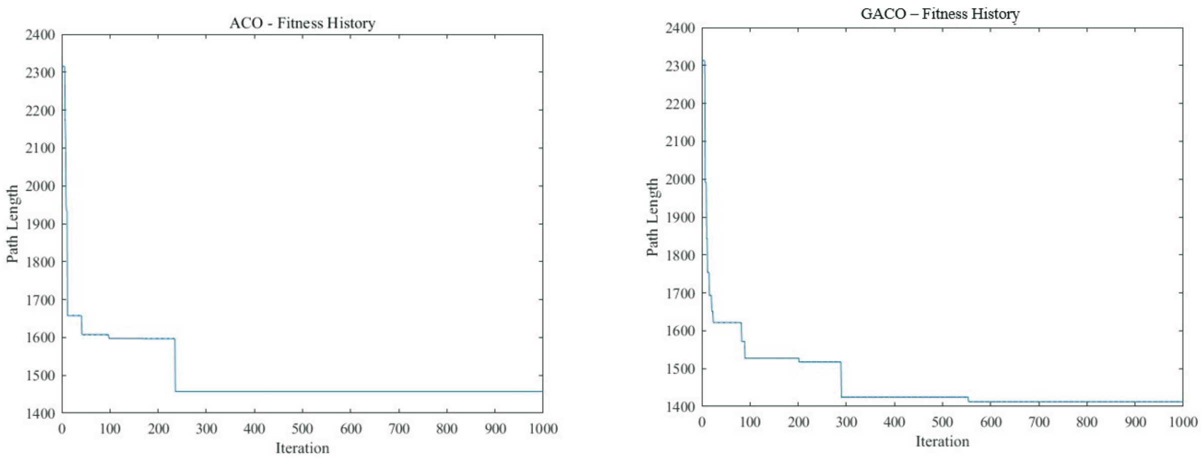
Definition	Total path length $_1$ (mm)	Total path length $_2$ (mm)	Rate $_{\text{total path length}}$ (%)	
ACO_1 / MC-ICO_2	2022.7901	2011.3316	-0.56	Wang et al.
ACO_1 / $GACO_2$	1456.80	1412.10	-3.16	This Article
ACO_1 / $MMAS_2$	67.69	65.61	-3,17	Song at al.
ACO_1 / $HACO_2$	1271.48	1199.90	-5.96	Wu at al.
ACO_1 / IPSO -ACO_2	67.69	58.05	-16,60	Song at al.

In conclusion, the MATLAB experiments reveal that the hybrid GACO algorithm offers a more effective solution in solving the TSP compared to the basic ACO algorithm, and demonstrate a success ratio that falls between the ratios of the hybrid ant colony and basic ant colony in studies developed in the literature.



(a) (b)

Fig. 5 Machining diagram of the shortest path **a** ACO **b** GACO



(a) (b)

Fig. 6 Path length iteration graph **a** ACO **b** GACO

4. Conclusion

According to extensive research, it is seen that the effectiveness of tool pathing is not taken into account in studies on application efficiency in the literature. Additionally, it is noted that while tool pathing exhibits similarities with the Travelling Salesman Problem (TSP) in studies, no specific study addressing tool pathing in drilling processes has been encountered. Although some comparisons have been made in studies in existing TSP examples, no comparisons with similar studies in the literature have been found. Some of the studies conducting comparisons also highlight the need for the further refinement of parameters. Taking into account these gaps and shortcomings, a hybrid algorithm named GACO was developed to resolve the drilling problem by considering the effective aspects of GACO, ACO, and GA. The hybrid GACO algorithm is capable of calculating the machining time on both two-dimensional CNC machines and three-dimensional industrial robot machines. An industrial robot example was employed to demonstrate the algorithm's operation in three different planes, and the results were compared with the literature. Users can acquire information through simulation using this software before machining their modelled parts, thus helping to save costs and time.

The codes were written in the MATLAB program to evaluate the algorithms' performance. When the codes are run, the results can be obtained as output in terms of computation time(s), total path length (mm), and total operation time (s). The GACO and ACO algorithms were run 10 times to verify their stability, and the results were recorded. Due to the heuristic nature of the algorithms, different results emerged in this example. The best example values among these results were identified, and the ratios between them were found. The performance of the hybrid GACO algorithm was seen to be 3.16% better than the ACO algorithm in terms of both total path length and total operation time. In the MATLAB program, it was observed that the ACO algorithm lagged behind the GACO algorithm by 6.46% in terms of calculation time. The ACO algorithm yields the machining sequence for the shortest path between holes as [1 11 14 15 17 18 19 12 16 13 10 22 23 20 21 24 25 8 5 9 6 7 4 2 3 1]. In terms of the best path length, the GACO algorithm, outperforming the ACO algorithm, provides the machining sequence for the shortest path between holes as [23 20 22 10 13 16 12 19 18 17 15 14 11 1 2 4 7 6 9 3 5 8 25 24 21 23]. Furthermore, the performance of the hybrid GACO algorithm, which showed better performance in terms of both total path length and total operation time, was compared with the studies in the literature. The best path ratios of Wang et al. were 0.56%, those of Song et al. 3.17%, those of Wu et al. 5.96%, and those of Song et al. in another study were 16.60%.

Ultimately, the MATLAB experiments revealed that the hybrid GACO algorithm provides a more effective solution in solving the TSP compared to the basic ACO algorithm. In comparison to the studies developed in the literature, the hybrid ant colony demonstrates a ratio of success among the hybrid ant colony and basic ant colony. It has been observed that the performance of the hybrid GACO algorithm is 3.16% better than the ACO algorithm in both total path length and total machining time. In the MATLAB program, it was observed that the ACO algorithm is 6.46% behind the GACO algorithm in terms of computation time. Future studies aim to conduct simulations and laboratory experiments of the hybrid GACO algorithm on CNC machining tools and to compare the results. Additionally, a study can be conducted to address monitoring parameters such as dimensional tolerance, online (thrust force, torque), and offline (tool wear, delamination, surface roughness, circularity) parameters, which have been neglected. This study can develop an algorithm that evaluates these parameters individually or collectively, and integrate them with GACO, presenting the results to the user. In addition to these suggestions, we encourage future studies to explore different optimisation techniques and for the findings to be shared.

REFERENCES

[1] Tanriver K, Ay M. Experimental, Software and Topological Optimization Study of Unpredictable Forces in Bolted Connections. Tehnički vjesnik **2023**, 30 (4), 1175-1184. https://doi.org/10.17559/TV-20221113121639

[2] Zhou X, Gui W, Heidari AA, Cai Z, Liang G, Chen H. Random following ant colony optimization: Continuous and binary variants for global optimization and feature selection. Applied Soft Computing **2023**, 144,110513. https://doi.org/10.1016/j.asoc.2023.110513

[3] Zhong J, Feng Y, Tang S. A collaborative neurodynamic optimization algorithm to traveling salesman problem. Complex Intell. Syst. **2023**, 9, 1809–1821. https://doi.org/10.1007/s40747-022-00884-6

[4] Zhao W, Liu Y, Li Y. et al. Multi-robot coverage path planning for dimensional inspection of large free-form surfaces based on hierarchical optimization. Int J Adv Manuf Technol **2023** 127, 5471–5486. https://doi.org/10.1007/s00170-023-11788-1

[5] Harikrishnan VK, Annamalai S, Sampath S, Paramasivam S. A Time-Performance Improvement Model with Optimal Ergonomic Risk Level Using Genetic Algorithm. Transactions of FAMENA **2023,** 47(4):109-128. https://doi.org/10.21278/TOF.474049022

[6] Qi Z, Guang W, Lan L, Rui T. Contact Stress Prediction Model for Variable Hyperbolic Circular Arc Gear Based on the Optimized Kriging-Response Surface Model. Transactions of FAMENA **2020**, 44(4):59-74. https://doi.org/10.21278/TOF.444011819

[7] Katiyar S, Dutta A. Comparative analysis on path planning of ATR using RRT, PSO, and modified APF in CG-Space. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science **2023**; 236(10): 5663-5677. https://doi.org/10.1177/09544062211062435

[8] Sivakumar A, Saravanakumar S, Sathiamurthi P, KarthiVinith KS. Forecasting the Equipment Effectiveness in Total Productive Maintenance Using an Intelligent Hybrid Conceptual Model. Transactions of FAMENA **2022**; 46(3):29-40. https://doi.org/10.21278/TOF.463042822

[9] Zhang D, Luo R, Yin Y-b, Zou S-l. (2023) multi-objective path planning for mobile robot in nuclear accident environment based on improved ant colony optimization with modified A∗. Nuclear Engineering and Technology **2023**, 55 (5), 1838-1854. https://doi.org/10.1016/j.net.2023.02.005

[10] Wang W, Guo Q, Yang Z, Jiang Y, Xu J. A state-of-the-art review on robotic milling of complex parts with high efficiency and precision. Robotics and Computer-Integrated Manufacturing **2023**, 79. https://doi.org/10.1016/j.rcim.2022.102436

[11] Hashemi A, Joodaki M, Joodaki N Z, Dowlatshahi M B. Ant colony optimization equipped with an ensemble of heuristics through multi-criteria decision making: A case study in ensemble feature selection, Applied Soft Computing **2022**,124. https://doi.org/10.1016/j.asoc.2022.109046

[12] Maniezzo V, Gambardella L, De Luigi F. Ant colony optimization. New optimization techniques in engineering **2004**, 1(5). https://doi.org/10.1007/978-3-540-39930-8_5

[13] Xiao Z, Wang Z, Liu D, Wang H. A path planning algorithm for PCB surface quality automatic inspection. J Intell Manuf **2022**, 33, 1829–1841. https://doi.org/10.1007/s10845-021-01766-3

[14] Das M, Roy A, Maity S, Kar. A Quantum-inspired Ant Colony Optimization for solving a sustainable four-dimensional traveling salesman problem under type-2 fuzzy variable. Advanced Engineering Informatics **2023**, 55. https://doi.org/10.1016/j.aei.2022.101816

[15] Gunay-Sezer N.S, Cakmak E, Bulkan, S. A Hybrid Metaheuristic Solution Method to Traveling Salesman Problem with Drone. Systems **2023**, 11, 259. https:// doi.org/10.3390/systems11050259

[16] Oleiwi B. K, Roth H, Kazem B. I. A Hybrid Approach Based on ACO and Ga for Multi Objective Mobile Robot Path Planning. In Applied Mechanics and Materials **2014**, 527, 203–212. https://doi.org/10.4028/www.scientific.net/amm.527.203

[17] Wu H, Gao Y, Wang W, Zhang Z. A hybrid ant colony algorithm based on multiple strategies for the vehicle routing problem with time Windows. Complex Intell. Syst. **2023**, 9, 2491–2508. https://doi.org/10.1007/s40747-021-00401-1

[18] Wang T, Wang L, Li D, Cai J, Wang Y. Monte Carlo-based improved ant colony optimization for path planning of welding robot. Journal of King Saud University- Computer and Information Sciences **2023**, 35(7). https://doi.org/10.1016/j.jksuci.2023.101603

[19] Heins J, Bossek J, Pohl J, Seiler M, Trautmann H, Kerschke P. A study on the effects of normalized TSP features for automated algorithm selection. Theoretical Computer Science **2022**,940, Part B,123-145. https://doi.org/10.1016/j.tcs.2022.10.019

[20] De Castro Pereira S, Solteiro Pires EJ, de Moura Oliveira PB. Ant-Balanced Multiple Traveling Salesmen: ACO-Bm TSP. Algorithms **2023**, 16, 37. https://doi.org/10.3390/a16010037

[21] Dong X, Xu M, Lin Q, Han S, Li Q, Guo Q. ITÖ algorithm with local search for large scale multiple balanced traveling salesmen problem. Knowledge-Based, Systems **2021**, 229. https://doi.org/10.1016/j.knosys.2021.107330

[22] Koulamas C, Kyparisis GJ. A classification of dynamic programming formulations for offline deterministic single-machine scheduling problems, European Journal of Operational Research **2023**,305 (3),999-1017. https://doi.org/10.1016/j.ejor.2022.03.043

[23] Uddin F, Riaz N, Manan A, Mahmood I, Song O-Y, Malik AJ, Abbasi AA. An Improvement to the 2-Opt Heuristic Algorithm for Approximation of Optimal TSP Tour. Appl. Sci. **2023**,13. https://doi.org/10.3390/app13127339

[24] Limafilho GML, Passaro A, Moura Delfino G, Monsuur H D, Monsuur H. A Novel Bias-TSP. Algorithm for Maritime Patrol in IEEE Access **2023**, 11,28190-28198. https://doi.org/10.1109/ACCESS.2023.3252013

[25] Köklü U, Mayda M, Morkavuk S, Avcı A, Demir O . Optimization and prediction of thrust force, vibration, and delamination in drilling of functionally graded composite using Taguchi, ANOVA and ANN analysis, Mater. Res. Express **2019**; 6(8), 085335. https://doi.org/10.1088/2053-1591/ab2617

[26] Morkavuk S, Köklü U, Aslantaş K. An Experimental Comprehensive Analysis of Drilling Carbon Fiber Reinforced Plastic Tubes and Comparison With Carbon Fiber Reinforced Plastic Plate, Journal of Reinforced Plastics and Composites **2023**, 42(7-8):323-345. https://doi.org/10.1177/07316844221122041

[27] Jangid YK, Goyal A, Dadhich M. Investigation of cutting time and tool wear rate on EN-24 steel alloy by the drilling process. Engineering Review **2022** ;42(2). https://doi.org/10.30765/er.1611

[28] Koklu, U. The drilling machinability of 5083 aluminum under shallow and deep cryogenic treatment, Emerging Materials Research **2020,** 9:(2), 323-330. https://doi.org/10.1680/jemmr.19.00127

[29] Zhu W, Fu H, Li F, Li Y, Bai F. Optimization of CFRP drilling process: a review. Int J Adv Manuf Technol **2022**, 123, 1403–1432. https://doi.org/10.1007/s00170-022-10112-7

[30] Tekaüt İ, Demir H, Şeker U. Experimental Analysis and Theoretical Modelling of Cutting Parameters in the Drilling of AISI H13 Steel with Coated and Uncoated Drills. Transactions of FAMENA **2018**, 42(2):83-96. https://doi.org/10.21278/TOF.42207

[31] Özerkan HB, Çoğun C. Electrochemical Small Diameter Deep Hole Drilling of Powder Metal Steel. Transactions of FAMENA **2020**, 44(4):47-58. https://doi.org/10.21278/TOF.444007919

[32] Peng JF, Ding Y, Zhang G, Ding H. Smoothness-oriented path optimization for robotic milling processes. Sci China Tech Sci, **2020**, 63: 1751–1763. https://doi.org/10.1007/s11431-019-1529-x

[33] Basmacı G, Kayacan M, Ay M, Etyemez A. Optimization of cutting forces and surface roughness via ANOVA and grey relational analysis in machining of In718. Open Chemistry **2023**, 21(1). https://doi.org/10.1515/chem-2022-0273

[34] Panchagnula KK, Palaniyandi, K. Drilling on fiber reinforced polymer/nanopolymer composite laminates: a review, Journal of Materials Research and Technology **2018**, 7(2), Pages 180-189. https://doi.org/10.1016/j.jmrt.2017.06.003

[35] Karimi F, Dowlatshahi MB, Hashemi A. SemiACO: A semi-supervised feature selection based on ant colony optimization. Expert Systems with Applications **2023**, 214. https://doi.org/10.1016/j.eswa.2022.119130

[36] Comert, SE, Yazgan HR. A new approach based on hybrid ant colony optimization-artificial bee colony algorithm for multi-objective electric vehicle routing problems. Engineering Applications of Artificial Intelligence **2023**, 123, Part B, 106375. https://doi.org/10.1016/j.engappai.2023.106375

[37] Jagadeesh P, Mavinkere Rangappa S, Suyambulingam I, Siengchin S, Puttegowda M, Selvi Binoj J, Gorbatyuk S, Khan A, Doddamani M, Fiore V, Moure Cuadrado MM. Drilling characteristics and properties analysis of fiber reinforced polymer composites: A comprehensive review. Heliyon **2023**, 9(3). https://doi.org/10.1016/j.heliyon.2023.e14428

[38] Malik K, Ahmad F, Keong WT, Gunister E. The effects of drilling parameters on thrust force, temperature and hole quality of glass fiber reinforced polymer composites. Polymers and Polymer Composites 2022, 30. https://doi.org/10.1177/09673911221131113

[39] Li H, Jiang X, Huo G, Su C, Zhou S, Wang B, Hu Y, g Zheng Z. An integrated trajectory smoothing method for lines and arcs mixed toolpath based on motion overlapping strategy. Journal of Manufacturing Processes **2023**, 95: 242-265. https://doi.org/10.1016/j.jmapro.2023.04.017

[40] Wang J, Yu H, Zheng Z, Lu G, Zhang K, Zheng T, Fang C. Autonomous robotic exploration with region-biased sampling and consistent decision making. Complex Intell. Syst.**2023**. https://doi.org/10.1007/s40747-023-01143-y

[41] Song Q, Yu L, Li S, Hanajima N, Zhang X, Pu R. Energy Dispatching Based on an Improved PSO-ACO Algorithm. International Journal of Intelligent Systems **2023**. https://doi.org/10.1155/2023/3160184

Submitted: 17.12.2023

Accepted: 26.02.2024

Kursat Tanriver *
Faculty of Engineering and Natural Sciences, Department of mechatronics Engineering, Istanbul Health and Technology University, Istanbul 34445, Turkey

Mustafa Ay
Faculty of Technology, Mechanical Engineering Department, Marmara University, Istanbul 34854, Turkey

*Corresponding author:
kursat.tanriver@istun.edu.tr