An improved hybrid genetic algorithm for holes machining path optimization using helical milling operation

Z Q Li, X Liu, L S Duan and L Liu
School of Mechanical Engineering, Hunan University of Technology, Zhuzhou, China
Zhqunli@hut.edu.cn

Abstract. Helical milling is capable of fabricating a lot of holes with only one cutting tool under one clamping. As the holes machining efficiency is determined by the holes machining time and the airtime, and when the cutting conditions of helical milling operation is determined, holes machining path optimization will have a great effect on holes machining efficiency. Based on the TSP model, an improved hybrid genetic algorithm is proposed to solve holes machining path optimization model to get the optimized processing path. By comparing the path length of the optimized machining path and the simulation time with that obtained using the basic GA and the basic ACO, the optimization algorithm proposed in this paper can effectively reduce the machining path length and thus improve the efficiency of holes machining operation.

1. Introduction
Helical milling is a new type of hole-making technology developed in recent years, which can fabricate holes of different diameters using only one cutting tool [1-2]. It is able to overcome the shortcomings of traditional processing methods, and thus it is particular suitable for making holes in hard-to-cut materials such as titanium alloy and carbon fibre reinforced polymer materials [3-5]. When helical milling is used to make holes, the machining time mainly consists of the cutting time and the airtime. The cutting time is determined by the cutting conditions, whereas the airtime is determined by the cutting order of holes. Accordingly, the machining efficiency can be improved by cutting conditions optimization to reduce the cutting time and tool path optimization to reduce the airtime. Study shows that when a large number of holes are machined at one time, the airtime may account for a large proportion of the overall machining time, and sometimes even up to 70% [6]. However, at present holes machining path in workshop is mainly determined by experience or designer’s intent, and far from optimal. Therefore, the optimization of holes machining path is very important in improving holes machining efficiency.

Holes machining path optimization can be simplified as a typical Traveling Salesman Problem (TSP). TSP is a classical combinatorial optimization problem and can be described as follows: Given a set of cities and the distances between each pair of them, what is the shortest possible tour that visits each city exactly once, and returns to the starting city [6]? From the perspective of graph theory, the problem can also be described as: Given a complete undirected graph \( G = (V, E) \) that has nonnegative integer cost \( c(u, v) \) associated with each edge \( (u, v) \) in \( E \), the problem is to find a Hamiltonian cycle of \( G \) with minimum cost. Since the feasible solution of the problem is the full permutation of all vertices, it will result in a combinatorial explosion with increasing number of vertices. Therefore, it is a NP complete problem and no polynomial-time algorithm can solve this kind of problem. NP-complete problems are generally considered as that the optimal solution cannot be found within limited time and space, and nonconventional solving techniques are commonly used to solve this kind of problems [7].
In early times, exact algorithms such as branch-and-bound method, cutting plane method, linear programming method and dynamic programming method were used to solve small-scale TSPs. However, when they are used to solve large-scale TSPs, the effect of these methods is not particularly satisfactory. Therefore, scholars are focusing on using intelligent algorithms such as genetic algorithm (GA), ant colony optimization (ACO), simulated annealing (SA), tabu search, greedy algorithm and artificial neural network (ANN) to solve this kind of problems.

Recently, these intelligent algorithms are widely used in solving optimization problems in manufacturing fields and good results are achieved. Applications of GAs in solving the multi-pass turning, multi-pass face milling, surface grinding and machining fixture design have been reported in literatures [8-11]. Liu [12] modified GA by defining and changing the operating domain and used for optimization of milling parameters. Wang [13] proposed a new hybrid approach, named genetic simulated annealing (GSA), based on GA and SA to find optimal machining parameters in milling operations. Vijayakumar [14] proposed a new optimization technique based on the ant colony algorithm for solving multi-pass turning optimization problems.

Through thoroughly analysis of the nature of different intelligent optimization algorithms, the cutting tool is treated as salesperson and holes to be machined as cities the salesperson want to visit in this paper. On the basis of mathematical modelling of TSP, an improved hybrid genetic algorithm was proposed to obtain the shortest machining path. As a result, the machining efficiency can be improved.

2. Establishment of mathematical model

When helical milling technology is applied to making holes in a part, its operation steps are as follows:
1) The cutting tool sets out from the start point;
2) Process each hole in a predetermined order;
3) Return to the start point.

In order to reduce the complexity of the problem, the following assumptions are made for the model:
1) Control accuracy of tool position can meet the design requirements;
2) Moving speed of the cutting tool between holes to be machined is kept unchanged;
3) Holes are machined under given cutting parameters.

The optimization of holes machining consists of machining path optimization and cutting conditions optimization of single hole. The former is the topic of this paper whereas the latter will be considered and presented in our next paper. In path planning of holes machining, the information of holes to be machined can be represented by hole’s position coordinates \((x, y)\), hole’s diameter \(D_h\) and hole’s depth \(h\), and the cutting parameters can be represented by the diameter of the cutting tool \(D_t\), the spindle speed \(n\), the tangential feed per tooth \(f_{zt}\) and the axial feed per tooth \(f_{za}\).

For a known weighted graph \(G = (V, A)\), where \(V = \{v_1, v_2, \ldots, v_n\}\) is the set of vertices which represent \(n\) holes to be machined, and \(A = \{(r, s): r, s \in V\}\) is the set of edges interconnecting each vertex which represents the combination interconnection of the holes in set \(V\). Supposing \(\delta(r, s)\) is the Euclidean distance between \(r\) and \(s\) and \(\delta(r, s) = \delta(s, r)\). Therefore, it is a symmetric TSP. Holes machining path optimization problem is to find a minimum cost closed tour that visits each hole once.

\[
\min L = \sum_{i=1}^{n} \delta_{a_i a_{i+1}}
\]  \hspace{1cm} (1)

3. Solution to TSP using IHGA

3.1. Classical genetic algorithm

GA is a search heuristic that mimics the process of natural selection and was first developed by Holland. This heuristic is routinely used to generate useful solutions to optimization and search problems. Genetic algorithms belong to the larger class of evolutionary algorithm (EA), which generates solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection and crossover. In a genetic algorithm, a population of candidate solutions to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties which can be
mutated and altered; traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. GAs consist of the following basic elements:

1) Genetic representation: use linear binary representation to code appearance, behavior, physical qualities of individuals;
2) Initialization: generate the initial population size based on the nature of the problem;
3) Define fitness function: the fitness function is always problem dependent and used to measure the quality of the represented solution;
4) Genetic operations: by producing a "child" solution using crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents".
5) Control parameters: define reasonable tuning parameters such as the mutation probability, crossover probability, population size, and generation number to find reasonable settings for the problem.

In practical applications, GA is very easy to cause premature convergence. How to determine the selection method to ensure both the best individual can be kept down and the diversity of population can be retained is very difficult [15].

3.2. Ant Colony Optimization
ACO is a probabilistic technique for solving optimization problems which can be reduced to finding good paths through graphs based on the behaviour of ants seeking a path between their colony and a source of food [16-18]. Advantages of ACO lie in that it has the features of self-adaption, self-organization, nature of positive feedback and intrinsic parallelism. However, due to the influence of better solutions found in early time of optimization process, ACO may has a long search time, easily fall into local optima and thus make the optimization stagnant.

The probability with which ant $k$ in city $r$ chooses to move to the city $s$ can be given by the following state transition rule used by ant system:

$$P_k(r,s) = \begin{cases} 
\frac{[\tau(r,s)]^\alpha [\eta(r,s)]^\beta}{\sum_{u \in J_k(r)}[\tau(r,u)]^\alpha [\eta(r,u)]^\beta} & \text{if } s \in J_k(r) \\
0 & \text{otherwise}
\end{cases}$$

(2)

where $\tau$ is the pheromone, $\eta = 1/\delta$ is the inverse of the distance $\delta(r,s)$, $J_k(r)$ is the set of cities that remain to be visited by ant $k$ located on city $r$, and $\beta$ is a parameter which determines the relative importance of pheromone versus distance ($\beta > 0$).

As shown in Eq. (2), the pheromone on edge $(r,s)$ is multiplied by the corresponding heuristic value $\eta(r,s)$ which means that the shorter edge is easily be selected and also has a greater amount pheromone.

In an ant system, the global updating rule is implemented as follows. Once all $m$ ants have completed their tours of all $n$ cities, the pheromone should be updated on all edges according to

$$\tau(r,s) = (1-\alpha) \cdot \tau_{ij}(r,s) + \sum_{k=1}^{m} \Delta \tau_k(r,s)$$

$$\Delta \tau_k(r,s) = \begin{cases} 
\frac{1}{L_k} & \text{if } (r,s) \in \text{tour done by ant } k \\
0 & \text{otherwise}
\end{cases}$$

(3)

where $0<\alpha<1$ is a pheromone decay parameter, $L_k$ is the length of the tour conducted by ant $k$, and $m$ is the number of ants. The flow chart of basic ACO is shown in Figure.1.

3.3. Improved Hybrid Genetic Algorithm
Based on the deep analysis of the characteristics of basic GA and ACO, combined with the advantages of modified fitness function and GA to find better initial population, an improved optimization solution method is proposed to improve the solution efficiency. In terms of fitness function, the constraints of problem are integrated into the fitness function, and as a result the fitness function with a variable penalty term is achieved which can be used to guide the genetic search.
As for the traveling salesperson problem, if the objective function is a problem of extreme value, the fitness function can be expressed as:

\[ f(x) = \frac{1}{u(x)} \]  (4)

where \( u(x) \) is the individual path length.

When there are individuals with superordinary fitness value in the initial population, these special individuals may affect the evolution direction of the whole population, results in the algorithm converges to local optimal solution, and thus the global optimal solution will be replaced by the local optimal one. Therefore, it is necessary to restrict the propagation of these exceptional individuals. Moreover, at the end of the iteration, there is no distinct difference between the superior and the inferior individuals, resulting in the excellent individuals cannot be optimized further and only swing around the optimal solution. As mentioned above, the fitness value will be modified according to the following equations:

\[ g(x) = \frac{f(x)}{f_{max} - f_{min} + \delta} \]

\[ f(x) = \begin{cases} 
C_{max} - u(x) & u(x) < C_{max} \\
0 & u(x) \geq C_{max}
\end{cases} \]  (5)

where \( f \) and \( g \) are fitness value before and after modification respectively, \( f_{max}, f_{min} \) are the upper and lower bounds of fitness value before modification respectively, \( \delta \) is a positive real number in the open interval \((0, 1)\) in order to avoid the denominator is zero and thus to increase the randomness of genetic algorithm, and \( C_{max} \) is the longest path in the current generation.

Figure.1 The flow chart of basic ACO  Figure.2 Modification of fitness value

As shown in Figure.2, with the increase of the difference between \( f_{max} \) and \( f_{min} \), the inclination angle \( \alpha \) of the modified fitness function becomes small; the variation range of the modified fitness value is reduced, and thus it can effectively prevent the local convergence resulting from the effect of special individuals on the whole population. Otherwise, the smaller the difference between \( f_{max} \) and \( f_{min} \), the larger the inclination angle \( \alpha \), and thus the larger the variation range of the modified fitness value. At late stage of iteration, the individual differences on the whole population are increased, and therefore the swing around the optimal solution can effectively be avoided. The modified fitness value is changed with population fitness value and alters the selection pressure [19].
In terms of how to generate the initial population, the basic ACO is used to generate the initial population for IHGA, in which m ants will tour all the n holes and different iteration times are used as termination conditions. The walking path of each ant under different iteration times can represent a gene \((a_0, a_1, \ldots, a_n)\), which indicates that the ant starting from \(a_0\), via \(a_1, a_2, \ldots, a_n\), and finally return to \(a_0\). To increase the diversity of the initial population, when the ACO is used to generate the initial population, the genes under different iteration times should be selected. As a result, at the beginning of selection and crossover in the genetic algorithm, the individual with the shortest path can be selected with the largest probability, and as a result the efficiency and accuracy of the algorithm can be improved. When IHGA is used in solving the holes machining path problem, the flow chart is shown in Figure 3.

### 4. Holes machining path optimization example and discussions

In holes machining, the geometrical feature can be represented using \((x, y, D_h, h, \text{num})\) where \(x, y\) is the coordinates of the hole centre, \(D_h\) is the hole diameter, \(h\) is the hole depth, and \(\text{num}\) is the number of the holes [25]. A part with 30 holes will be fabricated using helical milling operation, of which the geometrical parameters are: \{(20, 60, 20, 20, 1); (50, 30, 24, 20, 2); (80, 60, 20, 20, 3); (100, 20, 20, 4); (150, 60, 24, 16, 5); (200, 60, 24, 18, 6); (200, 20, 20, 16, 7); (240, 60, 20, 16, 8); (280, 20, 16, 16, 9); (320, 16, 18, 10); (335, 24, 16, 20, 11); (365, 23, 16, 20, 12); (380, 49, 16, 20, 13); (365, 75, 16, 20, 14); (335, 75, 16, 20, 15); (350, 50, 22, 20, 16); (350, 140, 22, 18, 17); (380, 170, 22, 18, 18); (350, 170, 22, 16, 19); (320, 170, 22, 16, 20); (240, 140, 20, 16, 21); (240, 100, 20, 16, 22); (200, 180, 20, 18, 23); (200, 140, 24, 16, 24); (150, 140, 24, 16, 25); (50, 100, 24, 16, 26); (80, 180, 24, 20, 27); (80, 140, 24, 20, 28); (20, 140, 24, 20, 29); (80, 110, 16, 20, 30)\}. The basic GA, the basic ACO and the proposed IHGA are all used to solve the holes machining path optimization problem respectively.

| Algorithm | Schematic diagram of the optimized tool path | Optimal path length /mm | Simulation time /s |
|-----------|---------------------------------------------|-------------------------|-------------------|
| GA        | ![Schematic diagram of the optimized tool path](image) | 1902.8 | 4.1 |

Figure 3 Flow chart of the improved hybrid genetic algorithm

![Schematic diagram of the optimized tool path](image)
In both the basic GA and the IHGA is used to obtain the optimal holes machining path, the population size is 100, the crossover probability is 0.8, and the mutation probability is 0.1. When the basic ACO is used to obtain the optimal holes machining path or to get the initial population for the IHGA, the number of ants \( m = 31 \), the parameter to control the influence of pheromone \( \alpha = 1 \), the parameter to control the influence of heuristic factor \( \beta = 5 \), the pheromone evaporation coefficient \( \rho = 0.5 \), and the pheromone intensity constant \( Q = 100 \). All the programs in this paper are executed under the platform of MATLAB 7.6.0 on a laptop (Intel ® Core (TM) 2 Duo CPU, 2.4 GHz, 4G). Simulation result comparisons of the above three algorithms are shown in table 1.

Table 1 shows that under the same simulation parameters, the shortest path length for the same machined holes obtained using GA, ACO and IHGA are 1902.8 mm, 1695.0 mm and 1410.6 mm respectively, and the simulation times are 4.1 s, 7.2 s and 3.9 s. As the path length is concerned, the path length obtained using IHGA is 26% shorter than that obtained using GA, and 18% shorter than that obtained using ACO. As the simulation time is concerned, the time of IHGA is almost the same as that of GA, but the time of ACO is much longer than the two genetic algorithms. In the above example, IHGA will save 46% of simulation time compared with ACO. Generally speaking, IHGA can obtain the shortest path length with less iteration times and shorter simulation time. Instead of generating the initial population randomly as the basic GA, IHGA makes full use of the advantages of ACO such as positive feedback and self-adaption, and selects the initial population from the walking path of ACO. As a result, the optimal solution can be achieved and the efficiency of the algorithm can be improved.

5. Conclusions
The paper proposed an improved hybrid genetic algorithm focusing on solving the holes machining path optimization problem using helical milling operation, and the following conclusions can be drawn.

1) In order to overcome the weakness of both the basic GA and the basic ACO, an IHGA was proposed by constructing a fitness function with variable penalty term as well as by introducing a new initial population generating mechanism resulting from the basic ACO.

2) Simulation results show that when the proposed IHGA was used to solve the classical TSP problem, both the obtained path length and the simulation time is the shortest. Compared with GA and ACO, the shortest path is reduced by 18.0% and the simulation time is reduced by 5.1%.

3) The proposed IHGA can be used to guide path planning for holes machining using helical milling operation in engineering applications.

Acknowledgement
The authors would like to acknowledge the support of Natural Science Foundation of Hunan Province (Grant No. 2020JJ4270)
References

[1] Li Z Q, Zheng M, Wang X 2013 J Hunan Univ Techno 27 38
[2] Brinkmeier E, Fangmann S, Meyer I 2008 Prod Eng 2 277
[3] Denkena B, Boehnke D, Dege J H 2008 CIRP J Manuf Sci Techno 1 64.
[4] Bullnheimer B, Kotsis G, Strauss C 1998 Applied Optimization (Wien: Springer)
[5] Khan W A (1999) Euro J Oper Res 117 310
[6] Lin W, Delgado-firas J G, Gause D C, Vassiladis S 1995 Cybernet Syst 26 387
[7] Holland J 1992 Adaptation in natural and artificial systems (Cambridge: A Bradford Boo)
[8] Bhaskara Reddy S V, Shunmugam M S, Narendran T T 1998 J Mat Process Tech 76 101
[9] Shunmugam M S, Bhaskara Reddy S V, Narendran T T 2000 Int J Mach Tool Manu 40 401
[10] Saravanan R, Asokan P, Sachidanandam M 2002 Int J Mach Tool Manu 42 1327
[11] Necmettin Kaya 2006 Comput Ind 57 112
[12] Liu Y M, Wang C J 1999) Int J Adv Manuf Technol 15 796.
[13] Wang Z G, Wong Y S, Rahman M 2004 Int J Adv Manuf Technol 24 727
[14] Vijayakumar K, Prabhaharan G, Asokan P, Saruvanan R 2003 Int J Mach Tool Manu 43 1633
[15] Mak K L, Wong Y S, Wang X X 2000 Int J Adv Manuf Technol 16 491
[16] Dorigo M, Gambardella LM 1999 IEEE T Evolut Comput 1 53
[17] Dorigo M, Stützle T 2004 Ant Colony Optimization (Cambridge: MIT Pres)
[18] Dorigo M, Blum C 2005 Theor Comput Sci 344 243
[19] Li Z Q, Wang X, Dong Y F 2013 Adv Mater Res 834: 1386-