Cutting Parameters Optimization by Using Particle Swarm Optimization (PSO)

J.G. Li¹,a, Y.X. Yao¹,b, D. Gao¹,c, C.Q. Liu¹ and Z.J. Yuan

¹School of Mechatronics Engineering, Harbin Institute of Technology, Harbin 150001, China
²mejgli@hit.edu.cn, bxyao@hit.edu.cn, cgaodong@hit.edu.cn

Keywords: Optimization, Cutting parameters, Particle swarm optimization, Cutting force

Abstract. Cutting parameters play an essential role in the economics of machining. In this paper, particle swarm optimization (PSO), a novel optimization algorithm for cutting parameters optimization (CPO), was discussed comprehensively. First, the fundamental principle of PSO was introduced; then, the algorithm for PSO application in cutting parameters optimization was developed; thirdly, cutting experiments without and with optimized cutting parameters were conducted to demonstrate the effectiveness of optimization, respectively. The results show that the machining process was improved obviously.

Introduction

Cutting parameters composed of cutting speed, feed rate, cutting depth and cutting width (for milling operation), have essential effects on the machining productivity and cost. The selection of cutting parameters has long depended on the skills and experience of machine tool operators or handbooks, and conservative cutting parameters are usually selected. This situation would cause significant productivity loses and lead to a costly machining operation.

The determination of optimum cutting parameters is a combinatorial optimization problem and is usually realized by applying optimization algorithms. These algorithms include neural network [1], geometric programming [2], simulated annealing [3], genetic algorithm (GA) [4], particle swarm optimization (PSO)[5], etc. GA was considered as a suitable algorithm for solving any type of machining process optimization problem [6]. However, encoding and decoding operations for each individual are involved and the complexity of genetic operations will be increased when variables to be optimized are more than two. It will decrease the process efficiency greatly.

PSO is discovered through simulation of the social behavior of bird flocking for food. It was used for optimization of continuous nonlinear functions. In PSO, the variables to be optimized need not to be encoded. Therefore, PSO will be more efficient when the number of variables to be optimized is more than two. Because of the convenience of realization and promising optimization ability in various problems, it has been paid more and more attention [7]. Opposite to the well-developed optimization algorithms, “PSO is still in its infancy and there are many associated problems that need further study” [8].

In this paper, CPO by using PSO will be discussed comprehensively in section 2 and section 3. In the following section, cutting experiments without and with cutting parameters optimization were conducted and analyzed. Conclusions were drawn in the last section.

Particle Swarm Optimization (PSO)

PSO is firstly introduced by Eberhart and Kennedy [9] and used for optimization of continuous nonlinear functions. The swarm is composed of volume-less particles with stochastic velocities, each of which represents a feasible solution. The algorithm finds the optimal solution through moving the particles in the solution space.

Each particle in PSO flies in the solution space with a velocity dynamically adjusted according to its own and its companions’ flying experience. Given a swarm of N particles, the ith particle is represented as \( X_i = (x_{i1}, x_{i2}, \ldots, x_{iD}) \) (D is the number of decision parameters of an optimum
problem. In PSO, not only the best particle among the swarm but also the best previous position for each particle needs to be recorded. The best previous position (the position giving the best fitness value) of the ith particle is represented as \( P_i = (p_{i1}, p_{i2}, \ldots, p_{iD}) \) and the best particle among the swarm is represented as \( P_g = (p_{g1}, p_{g2}, \ldots, p_{gD}) \). The rate of position change (velocity) for the ith particle is represented as \( V_i = (v_{i1}, v_{i2}, \ldots, v_{iD}) \). The particles are manipulated according to the following equation [10]

\[
\begin{align*}
    v_{id}(k+1) & = w \times v_{id}(k) + c_1 \times \text{rand}( ) \times (p_{id} - x_{id}(k)) + c_2 \times \text{Rand}( ) \times (p_{gd} - x_{id}(k)) \\
    x_{id}(k+1) & = x_{id}(k) + v_{id}(k+1)
\end{align*}
\]

where \( w \) is the inertia weight. \( c_1 \) and \( c_2 \) are two positive constants, and \( \text{rand}( ) \) and \( \text{Rand}( ) \) are two random functions with return value in range \([0, 1]\) independently. The first part of Eq.1 is the previous velocity of the particle. The second represents the exploiting of its own experience, where \( c_1 \) is the individual learning factor. And the third represents the shared information and mutual cooperation among the particles, where \( c_2 \) is the social learning factor.

Algorithm of Cutting Parameters Optimization by Using PSO

In NC programming, generally, cutting depth and cutting width are specified firstly according to the cutting tool and the workpiece; then spindle speed and feed rate are determined, which depends much on the programmer’s experience. Therefore, spindle speed and feed rate are considered as variables to be optimized. CPO is a process to searching for the optimal solutions in the solution space within the bounds defined by practical constraints [11] by using an optimization algorithm.

Fig.1 shows the framework of methodology of CPO. In Fig.1, the Given cutting parameter(s) is the parameter(s) that has been specified, such as cutting depth and cutting width. The Algorithm parameters are the initial value of parameters for the optimization algorithm, such as population size, the maximum number of iterations, etc. The Objective is a goal that the optimal cutting parameters can produce the extreme value. The Cutting database provides adequate information for the prediction of cutting force, tool-life, etc. Based-on the cutting parameters carried by each individual, the Prediction of machining performance is used to predict the machining performances so that optimal cutting parameters can be searched and evaluated. The Constraints, including power, tool-life, surface finish, etc. determines the solution space so that cutting parameters optimally determined must be limited in the bounds. The Optimization methodology is the kernel of optimization, by using which optimum cutting parameters are searched. Based-on the prediction of machining performance, the optimal cutting parameters are evaluated in the Evaluation module. The diagram of CPO by using PSO is sketched as shown in Fig.2.

![Fig. 1 Framework of methodology of cutting parameters optimization](image-url)
Input optimization parameters

Randomly initialize particles $X_i$ and $V_i$

Machining performance evaluation and refresh $P_i$ and $P_g$

$k$ reaches the max iterations? Output $P_g$

End

Calculate $V_i(k+1)$ and $X_i(k+1)$ with Eqs (1) and (2); $k++$

Fig. 2 Diagram of CPO by using PSO

Fig. 3 Snap shot of CPO tool by using PSO

Table 1 Test conditions of CPO by using PSO

<table>
<thead>
<tr>
<th>Material of workpiece</th>
<th>YL11</th>
<th>Range of spindle speed</th>
<th>300–4000[rev/min]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cutting depth</td>
<td>5[mm]</td>
<td>Range of feed rate</td>
<td>30–2000[rev/min]</td>
</tr>
<tr>
<td>Cutting width</td>
<td>10[mm]</td>
<td>Effective power</td>
<td>3.0[KW]</td>
</tr>
<tr>
<td>Material of tool</td>
<td>HSS</td>
<td>Permitted force</td>
<td>2000[N]</td>
</tr>
<tr>
<td>Diameter of tool</td>
<td>16[mm]</td>
<td>Ra</td>
<td>3.2[µm]</td>
</tr>
<tr>
<td>Number of flutes</td>
<td>3</td>
<td>Tool life</td>
<td>60[min]</td>
</tr>
<tr>
<td>Helix of flute</td>
<td>43°</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

According to Fig.1 and Fig.2, a cutting parameters optimization tool has been developed to determine optimal spindle speed and feed rate as shown in Fig.3. Fig.4 shows the average fitness of each searching cycle with an objective of constant cutting force under test conditions listed in Table 1. From Fig.4, the PSO on CPO can converge quickly to an optimal solution. A series of optimizations of cutting parameters have been made to demonstrate the consistency of results using PSO. The results tabulated in Table 2 shows that the deviation is within 3%.

Table 2 Result analysis of a series of optimizations

<table>
<thead>
<tr>
<th>Spindle speed[rev/min]</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>Average</th>
<th>Deviation (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1022</td>
<td>1019</td>
<td>1027</td>
<td>1031</td>
<td>1013</td>
<td>1022.4</td>
<td>0.92</td>
</tr>
<tr>
<td>Feed rate[mm/min]</td>
<td>1264.2</td>
<td>1260.0</td>
<td>1268.6</td>
<td>1273.7</td>
<td>1256.1</td>
<td>1264.5</td>
<td>0.73</td>
</tr>
<tr>
<td>Power[kW]</td>
<td>0.940</td>
<td>0.937</td>
<td>0.943</td>
<td>0.947</td>
<td>0.934</td>
<td>0.940</td>
<td>0.74</td>
</tr>
<tr>
<td>Ra[µm]</td>
<td>0.844</td>
<td>0.844</td>
<td>0.841</td>
<td>0.842</td>
<td>0.848</td>
<td>0.884</td>
<td>0.47</td>
</tr>
<tr>
<td>Tool life[min]</td>
<td>67.82</td>
<td>68.47</td>
<td>66.82</td>
<td>66.08</td>
<td>69.51</td>
<td>67.74</td>
<td>2.61</td>
</tr>
</tbody>
</table>

Fig. 4 Fitness of iterations: a) average fitness, b) fitness of the best particle
Experiments of Cutting Parameters Optimization with PSO

PSO is used to optimize the cutting parameters when a NC program is to be generated. To demonstrate the effectiveness of PSO in cutting parameters optimization, a controlled cutting experiment under the same cutting conditions (Table 3) was conducted without and with optimum parameters, respectively. The surface to cut is cylindrical so that cutting depth can continuously change (Fig. 5). Fig. 6 shows the cutting forces without optimization, respectively. The amplitudes of cutting forces change as that of the cutting depth. The cutting time is 42s.

<table>
<thead>
<tr>
<th>Table 3 Cutting conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Material of tool</strong></td>
</tr>
<tr>
<td><strong>Type of tool</strong></td>
</tr>
<tr>
<td><strong>Diameter of tool [mm]</strong></td>
</tr>
<tr>
<td><strong>Number of Flutes</strong></td>
</tr>
<tr>
<td><strong>Material of part</strong></td>
</tr>
<tr>
<td><strong>Radius of part [mm]</strong></td>
</tr>
<tr>
<td><strong>Dynamometer</strong></td>
</tr>
<tr>
<td><strong>Cutting width [mm]</strong></td>
</tr>
</tbody>
</table>

Based on the given cutting depth and cutting width, the spindle speed and feed rate were optimized by using PSO for an objective of constant cutting force as shown in Fig. 7. The NC program was modified according to the optimal spindle and feed rate. Then a cutting experiment was conducted. The cutting forces are shown in Fig. 8. The amplitudes of cutting forces are close to constant values, respectively. The cutting time is reduced to 35s.

Fig. 5 Cutting depth
Fig. 6 Measured cutting forces and predicted cutting forces

Fig. 7 Optimized cutting parameters.

Fig. 8 The measured and predicted cutting force with optimal cutting parameters
Summary

In this paper, the cutting parameters optimization by using particle swarm optimization was discussed comprehensively. According to the results, conclusions were drawn as follow:

1. PSO in cutting parameters optimization can converge quickly to a consistent combination of spindle speed and feed rate.
2. Machining process can be improved via cutting parameters optimization.
3. And machining efficiency can also be improved.

References

DOI References


doi:10.1007/s001700170146


pp.1756. 
doi:10.1109/TMAG.2005.848610

doi:10.1016/S0890-6955(96)00044-2


doi:10.1007/s001700170146

doi:10.1007/s001700170146

pp.1756. 
doi:10.1109/TMAG.2005.848610