The assessment of straightness and flatness errors using particle swarm optimization

C. Cui\textsuperscript{a,b,*}, T. Li\textsuperscript{a}, L. A. Blunt\textsuperscript{a}, X. Jiang\textsuperscript{a}, H. Huang\textsuperscript{b}, R. Ye\textsuperscript{b} and W. Fan\textsuperscript{b}

\textsuperscript{a}EPSRC Innovative Manufacture Research Centre in Advanced Metrology, Centre for Precision Technologies, School of Computing and Engineering, University of Huddersfield, Queensgate, Huddersfield, HD1 3DH, UK
\textsuperscript{b}College of Mechanical Engineering and Automation, Huaqiao University, Xiamen, 361021, China

Abstract

The straightness and flatness errors are generally assessed by using the Least Squares Method (LSM). However, the results obtained from LSM often overestimate the tolerances, and are not consistent with the ISO standards' definitions. To this end, this paper presents a method to evaluate those errors by using particle swarm optimization (PSO). The realization technique is detailed. The experimental data is utilized to verify this algorithm, together with a comparison with some typical optimization algorithms.

© 2012 The Authors. Published by Elsevier B.V. Selection and/or peer-review under responsibility of Professor Xiangqian Jiang.

Keywords: form error; straightness; flatness; assessment; Particle Swarm Optimization (PSO); optimization

1. Introduction

The evaluation of form errors, such as straightness and flatness, are of importance for the precision mechanical manufacturing. In practice, it is impossible (and also unnecessary in many cases) to obtain the variation over the whole surface of a workpiece. Only finite points, therefore, are collected from the surface to represent its features. To assess the tolerance errors, it is important to select an appropriate algorithm to extract the feature from the measured dataset. Note that an inappropriate algorithm may overestimate the tolerance and lead to unnecessary rejection.

The definitions of straightness and flatness have been specified by the International Standard Organization (ISO) in detail [1] and they have been improved greatly and rapidly with the development of science and technology worldwide. Those errors are determined by both the location and orientation of a reference datum. The datum is unknown in advance before assessing the tolerance, and their assessments are non-linear issues. Generally, they are evaluated by the Least Squares Method (LSM) and minimum zone algorithms. The LSM is a traditional method which is to find the result under the condition, that the sum of the squares of the residuals (between the sampled values and calculated values) is minimized. It has been widely accepted in many fields such as in the form errors assessment due to the uniqueness of its results and the simplicity on its computation. The problem is that the LSM is an approximative method, which could lead an overestimation the tolerance and results in an unnecessary rejection [2,3]. To replace the LSM, thus, many algorithms have been proposed, e.g. the optimization algorithms. And most of them conform to the minimum zone principle [4,5]. However, some of them have difficulty on its understanding, interpreting and implementing. And some of them cannot assess all items of the geometrical errors at the same time. Thus, researchers introduce the optimization algorithm, such as

* Corresponding author. Tel.: +44-1484-471284; fax:+44-1484-472161.
E-mail address: cuichc@hqu.edu.cn
the genetic algorithm (GA) and particle optimization algorithm (PSO). The GA is a little more complex than PSO in the principle for the same work [6,7]. Some PSO-based algorithms have already been developed to evaluate the cylindricity errors [8-10].

This paper documents the PSO-base algorithm for the evaluation of the straightness and flatness errors. Section 2 presents the modeling of those errors. The implementation is detailed in Section 3. Section 4 presents the verification of this method, together with a comparison between some of the typical methods. Section 5 concludes that PSO has advantages in the assessment on the straightness and flatness.

2. Computation models for straightness and flatness errors

2.1. Straightness error

Adhering to the definition given by ISO 1101 [1], the datum of an evaluated straight line can be expressed as:

\[ x \cos \beta - y \cos \alpha - c = 0 \]  

(1)

where \( \alpha \) and \( \beta \) are the direction angles \( (\alpha + \beta = 90^\circ) \), \( x \) and \( y \) are the coordinates of the line, and \( c \) is a constant (see Fig. 1). Now suppose \( X = (\alpha) \) is the unknown parameter vector, and \( \{P_i\} = \{x_i, y_i\}, (i = 1, 2, \ldots, n) \) are the coordinates of samples of the assessed line. The straightness error is,

\[ h(X) = \max \{d_i\} - \min \{d_i\} = d_{\text{max}} - d_{\text{min}}, i = 1, 2, \ldots, n \]  

(2)

where \( d_i = x_i \cos \beta - y_i \cos \alpha - c \). Then the error \( \delta \) of the straight line is,

\[ \delta = \min [h(X)] \]  

(3)

2.2. Flatness error

Adhering to ISO’s definition [1], the datum plane of an evaluated plane can be expressed as:

\[ x \cos \alpha + y \cos \beta + z \cos \gamma - c = 0 \]  

(4)

where, \( \alpha, \beta \) and \( \gamma \) are the direction angles of the plane, \( x, y \) and \( z \) are the coordinates of the plane, and \( c \) is a constant (see Fig. 2). Suppose \( X = (\alpha, \beta, \gamma) \) is the unknown parameter vector, and \( \{P_i\} = \{x_i, y_i, z_i\}, (i = 1, 2, \ldots, n) \) are the coordinates of samples of the plane. Then the objective function of flatness error is:

\[ h(X) = \max \{d_i\} - \min \{d_i\} = d_{\text{max}} - d_{\text{min}}, i = 1, 2, \ldots, n \]  

(5)

where \( d_i = x_i \cos \alpha + y_i \cos \beta + z_i \cos \gamma - c \). Then the error \( \delta \) of the plane is the minimum of \( h(X) \), i.e.

\[ \delta = \min [h(X)] \]  

(6)

3. Realization technique of PSO in form errors optimization

The Particle Swarm Optimization is a stochastic evolutionary method first proposed by Kennedy and Eberhart in 1995 [11,12]. PSO is made up of a swarm of particles. Particle represents a potential solution, and will move within a multidimensional search space in order to find the best position.

Suppose that the search space is \( D \)-dimensional, the \( i \)th particle of the swarm is represented by a \( D \)-dimensional vector \( X_i = (x_{i1}, \ldots, x_{id}, \ldots, x_{iD}) \), and the velocity of this particle is represented by \( V_i = (v_{i1}, \ldots, v_{id}, \ldots, v_{iD}) \). This two dimensional searching space is shown in Fig.3. The velocity and new position
of the $i$th particle are updated by the following equations:

$$v_{id}(t+1) = \omega(t) \times v_{id}(t) + c_1 \times r_1 \times (x_{gd}(t) - x_{id}(t))$$ \hspace{1cm} (7)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$ \hspace{1cm} (8)

where, $\omega$ is the inertia weight to control the impact of velocity of previous particles. $r_1$ and $r_2$ are independently uniformly distributed random variables within range $(0,1)$. $c_1$ and $c_2$ are positive constant parameters, called acceleration coefficients, which control the maximum step size. $x_{gd}$ is the $d$th dimensional parameter of the best point which the swarm can find out. $x_{imd}$ is the $d$th dimensional parameter of the best point which the $i$th particle can get. $t$ is the evolution generation.

The particles are encoded by using real numbers. For each particle, $x_{id}$ corresponds to the $d$th dimensional variable or optimization parameter of the optimization problem, and $v_{id}$ is the increment or evolutionary step correspondingly.

The fitness function bridges the problem and the optimization algorithm. Optimization problems can be grouped into two classes, maximization and minimization. For the evaluation of straightness and flatness, the fitness functions of PSO is given as:

$$f(X) = \frac{1}{\varepsilon + h(X)}$$ \hspace{1cm} (9)

where, $h(X)$ is the objective function of straightness and flatness error as shown in section 2, $\varepsilon$ is a small parameter to keep $f(X)$ meaningful.

Inertia weight is used to adjust the impact of the velocity of the previous generation of particles to the new particles. It changes with the evolution to achieve a finer adjustment. It is expected that the inertia weight decreased gradually with the evolution from a bigger value to a smaller one, and an adjustment strategy [13] is expressed as:

$$\omega(t) = (\omega_0 - \omega_e) \left( \frac{T - t}{T} \right) + \omega_e$$ \hspace{1cm} (10)

where, $t$ is the current generation of the evolution, $T$ is the total generations of evolution, $\omega_0$ is the initial inertia weight, $\omega_e$ is the ultimate inertia weight.

According to the search principle of PSO, the new generation of particles comes from the best particle of the current generation and found by the whole swarm. Suppose that the search space is $D$-dimensional, and the best particle of the swarm and the velocity of this partial can be represented by $X_m=(x_{m1}, x_{m2}, \ldots, x_{mD})$, and $V_m=(v_{m1}, v_{m2}, \ldots, v_{mD})$ respectively. On the basis of Equ. (7), the new velocity of a particle is generated as follow:

$$v_{id}(t+1) = \omega(t) \times v_{id}(t) + c_1 \times r_1 \times (x_{gd}(t) - x_{id}(t))$$ \hspace{1cm} (11)

where, $r_1$ is a randomly distributed variable with range $(0,1)$, $c$ is a positive constant parameter.

In terms of Equ. (11), the variation of the random parameter $r_1$ can produce a group of increments that their center is the best particle of the former generation swarm, and generate a group of new particles. Based on Equ. (8), the $D$-dimension new position of the $i$th particle is generated as follow:

$$x_{id}(t+1) = x_{imd}(t) + v_{id}(t+1)$$ \hspace{1cm} (12)

4. Experimental validation and discussion

To validate this proposed PSO algorithm, a comparison has undertaken by using datasets in
reference [14]. Table 1 & 2 list the results obtained from different algorithms, i.e. Least Squares Method (LSM), Optimization Technique Zone (OTZ) [14], Linear Approximation Technique (LAT) [14], Genetic Algorithm [7] and PSO. The condition for PSO is listed as follows. The constant \( c \) in Equ (11) is set to 2. The inertia weight changes from 0.9 to 0.4. The particle number \( S \) is 20. The particle dimension \( D \) is 1 and 2 for straightness and flatness respectively. The initial particles are produced based on the result given by the LSM. The maximum velocity \( V_{max} \) is set as the distribution range of the measurement data, and the termination condition is the maximum evolution generations 40.

Compared with LSM, the results in Table 1 & 2 show that the PSO is an effective optimization algorithm which assesses the flatness and straightness errors with improved precision. And the precision of the results of obtained from PSO algorithms are at the same level as that of OTZ, LAT, and GA (see Table 1 & 2). The advantages of PSO are its relative simple principle and the ease of its realization.

5. Conclusions

In this paper, a particle swarm optimization algorithm has been developed to solve the optimization problem of straightness and flatness evaluation. The given examples show the improved precision of the proposed algorithm than that of LSM. The effectiveness of those algorithms has been illustrated via a comparison with other optimization algorithms.

Acknowledgements

The Project is supported by the Program for Changjiang Scholars and Innovative Research Team in University (No.PCSIRT1063), Program for New Century Excellent Talents in University (No. NCET-10-0116). The authors would like to thank the EPSRC Centre for Innovative Manufacturing in Advanced Metrology, UK.

References


<table>
<thead>
<tr>
<th>Examples</th>
<th>LSM</th>
<th>OTZ (^{[14]})</th>
<th>LAT (^{[14]})</th>
<th>GA (^{[7]})</th>
<th>PSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.001672</td>
<td>0.001711</td>
</tr>
<tr>
<td>2</td>
<td>0.0015</td>
<td>0.0014</td>
<td>0.0014</td>
<td>0.001428</td>
<td>0.001401</td>
</tr>
<tr>
<td>3</td>
<td>0.0049</td>
<td>0.0047</td>
<td>0.0047</td>
<td>0.004687</td>
<td>0.004706</td>
</tr>
<tr>
<td>4</td>
<td>0.0025</td>
<td>0.0023</td>
<td>0.0023</td>
<td>0.002266</td>
<td>0.002291</td>
</tr>
<tr>
<td>5</td>
<td>0.0011</td>
<td>0.0011</td>
<td>0.0011</td>
<td>0.001079</td>
<td>0.001104</td>
</tr>
</tbody>
</table>

\(^{a}\) Optimization technique zone 
\(^{b}\) Linear approximation technique 
\(^{c}\) Genetic algorithm


