Surface Roughness Characterization for Stress Concentration Factor Predictions: A Machine Learning Approach

Abstract

The surface roughness has an important influence on the fatigue life of the structures. The fatigue life reduces due to the stress concentration caused by surface roughness. The stress concentration governs the fatigue crack initiation and propagation. The accurate acquisition of the stress concentration factor of rough surfaces is a key issue in determining fatigue life. Nevertheless, semi-empirical models may be biased for various machining processes. Besides, finite element method simulations cannot give explicit expression of the stress concentration factor. In this paper, based on a number of data pairs, the correlation expression between the stress concentration factor and statistical roughness parameters of surfaces is established quickly through machine learning. The proposed method is illustrated using numerical database constructed by finite element method. Compared with some other semi-empirical models, the accuracy and stability of the proposed method are certified. This paper provides a simple and efficient approach to determine the stress concentration factor for rough surfaces under different processing conditions.

Keywords: surface roughness, stress concentration factor, statistical roughness parameters, MCMC, Gibbs sampling

1. Introduction

The fatigue life of structures is known to influence the service life of the structures highly. Moreover, the fatigue life depends on the surface quality, which will lead to the fatigue crack. There are three parameters to describe surface quality: (i) surface roughness, (ii) residual stress, (iii) microstructure. For some materials, especially aluminum alloy, surface roughness is the dominant parameter for fatigue life. Surface roughness introduces local stress concentration governing the crack initiation and propagation. The experiments show that the fatigue strength increases with a decrease in the surface roughness [1]. Therefore, the effect of surface roughness on the fatigue life cannot be neglected. Surface roughness is described through statistical roughness parameters. Traditional fatigue life predictions use empirical reduction factors to calculate the effect of surface roughness [2-5]. However, it is not a simple way due to many materials and machining processes. The empirical reduction factors need to be found for the same machining process through time-consuming testing. Beyond that method, the effect of statistical roughness parameters on the fatigue life is proposed in several approaches. In those methods, surface roughness is considered as causing stress concentration to reduce the crack initiation and propagation life. The effect can be described through the fatigue stress concentration factor $K_f$. It is defined as

$$K_f = \frac{\sigma_{D\text{ (smooth)}}}{\sigma_{D\text{ (rough)}}} \tag{1}$$

And it can be expressed by the stress concentration factor $K_t$ according to

$$K_f = 1 + q(K_t - 1) \tag{2}$$

This consideration requires the calculation of the stress concentration factor to evaluate the fatigue life of engineering components [6].

There are two methods to study the influence of surface roughness on the stress concentration factor. One method is a semi-empirical formula, and the other is a numerical simulation. Some researchers have treated surface roughness as a series of microscopic notches. A stress concentration factor $K_t$ is introduced. Neuber [7] considered that the surface topography is characterized by adjacent continuous gaps and proposed to estimate $K_t$ according to

$$K_t = 1 + n \sqrt{\lambda R_t \rho} \tag{3}$$

where $\lambda$ refers to the ratio between spacing and height of surface irregularities, and $n$ represents the conditions (where $n=1$ represents the shear and $n=2$ represents tensile). Nevertheless, the actual value of $\lambda$ is hard to define for generic surface textures. Arola and Ramulu [8] suggested a different formula, where

$$K_t = 1 + n \left( \frac{R}{\rho} \right) \left( \frac{R_t}{R} \right) \tag{4}$$

The surface topography is simplified as an ideal sinusoidal micro-notch, and for AISI 430 CR steel, this expression can estimate the fatigue stress concentration factor ($K_f$) more accurately. Also, Andrews and Sehitoglu [9] treated the surface topography as a more common semi-elliptical notch and gave the expression as
where the notch width and notch spacing are denoted as $a$ and $b$, the notch depth and tip radius are symbolized by $c$ and $\rho$. The above methods simplify the surface topography to microscopic notches, making it difficult to estimate the surfaces under different processing conditions accurately. Another method to study the problem calculates $K_t$ from finite element method (FEM) simulations of the measured surface topography. As et al. [10,11] used the FEM description of the surface profile to calculate $K_t$, the surface topography was measured using a white light interferometry microscope. Suraratchai et al. [12] fitted the measured machined surface topography and analyzed $K_t$ by FEM simulations. However, the FEM simulations cannot give the explicit expression of the stress concentration factor.

In this paper, $K_t$ is found from FEM simulations of surface topography, and the predicted results are used for training the machine learning models. Subsequently, Bayesian learning with Gibbs sampling, one of Markov Chain Monte Carlo (MCMC) algorithms, is employed to predict expression between stress concentration factor and statistical roughness parameters. The expression between $K_t$ and statistical roughness parameters of surfaces under different processing conditions can be established using this method. Furthermore, the predicted values are in good agreement with the FEM simulations.

2. Finite element method

2.1 Rough surfaces

The rough surfaces are generated randomly through an open-source code in MATLAB [13]. The usability of this code has been experimentally verified by the proposer. According to the study [5], machined surface roughness ($R_a$) values over 0.1 μm, which strongly influences fatigue life. With the surface roughness ($R_a$) being less than 0.1 μm, this effect diminishes as cracks initiate due to persistent slip bands or grain boundaries. Therefore, when the surface topography is generated, the value of $R_a$ is set to be greater than 0.1 μm. Moreover, the lower frequency cutoff and the upper frequency cutoff are incorporated through a slight modification. The surfaces can also be assigned from actual rough topographies through a white light interferometry microscope. The profile of the generated rough surface is shown in Fig. 1.

![Fig. 1 The profile of generated surface topography](image)

Surface roughness is characterized through statistical roughness parameters such as $R_a$ (average roughness), $R_y$ (peak-to-valley height roughness), $R_z$ (10-point roughness), and $\rho$ (the average profile valley radius). These parameters are defined in terms of the profile height distribution ($z$) recorded, in respect to the mean line, over an assessment length ($L$) according to

$$ R_a = \frac{1}{L} \int_0^L |z(x)| \, dx $$  \hspace{1cm} (6)

$$ R_y = |z_{\text{max}} - z_{\text{min}}| $$  \hspace{1cm} (7)

$$ R_z = \frac{1}{5} \left[ \sum_{i=1}^5 (z_i)_{\text{max}} + \sum_{i=1}^5 (z_i)_{\text{min}} \right] $$  \hspace{1cm} (8)

where $(z_i)_{\text{max}}$ and $(z_i)_{\text{min}}$ are the five higher local maxima and lower local minima, respectively, of the profile height distribution ($z$).

2.1 Finite element method simulations
In FEM simulations, the assumption of isotropic and homogeneous materials is introduced. When calculating the stress concentration factor, only the linear elastic behavior of the material should be considered. Firstly, a smooth finite element model is established by using the commercial software ABAQUS. Then, to realize the surface of model is the above profile generated by MATLAB, MATLAB programming is used to change the surface node coordinates of the generated model.

Fig. 2 shows the example of the final generated finite element model and simulation performed to determine $K_t$.

For the turning process, the specimens are rotated in a lathe. This process produces circumferential grooves through a repeated way, which can be modelled by an axisymmetric model. The height of the model is 5000 μm, and the nominal length of the rough surface is 20000 μm. Uniform tensile load is applied to both ends of the boundary. Young’s module $E$ is 72 GPa, and Poisson’s ratio $\nu$ is 0.33. In FEM simulations, axisymmetric quadrilateral elements (CAX4) are adopted to discretize the model, in which the meshes are gradually coarsened along the $r$-axis from top to bottom. For one of rough surfaces ($R_a=0.192818$, $R_y=1.191023$, $R_z=1.179942$, $\rho=569.1562$), the minimum mesh size of the adjacent surface topography is 0.4 μm. As preparation, the accuracy of our computational results has been guaranteed by convergence tests.

The stress concentration factor of each point on the surface topography is defined as

$$K = \frac{\sigma}{\sigma_{nom}}$$

where $\sigma$ is the Von Mises equivalent stress, and $\sigma_{nom}$ is the nominal Von Mises equivalent stress of the cross-section, as shown in Fig. 3.
And the maximal Von Mises equivalent stress obtained by the calculation is then divided by the nominal Von Mises equivalent stress due to the applied load to determine the stress concentration factor $K_t$, classically expressed as

$$K_t = \frac{\sigma_{\text{max}}}{\sigma_{\text{nom}}}$$  \hspace{1cm} (10)

where $\sigma_{\text{max}}$ is the maximal Von Mises equivalent stress along with the surface topography.

### 3. Machine learning

According to the existing models, several models propose the expression between $K_t$ and the statistical roughness parameters of surfaces, as shown in Table 1.

#### Table 1

<table>
<thead>
<tr>
<th>Empirical models</th>
<th>Literature</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_t = 1 + 2\lambda R_y \rho^{-1}$</td>
<td>Neuber</td>
</tr>
<tr>
<td>$K_t = 1 + 2R_y R_z (R_y \rho)^{-1}$</td>
<td>Arola-Ramulu</td>
</tr>
</tbody>
</table>

Those models can be rewritten as linear models. Therefore, the expression is assumed as

$$\ln(K_t - 1) = \omega_1 \ln(R_y / \rho) + \omega_2 \ln\left(R_z / R_y\right) + \omega_3 \ln\left(R_z / R_y\right) + \omega_4$$  \hspace{1cm} (11)

The Bayesian learning [14-18] with MCMC (Gibbs sampling) [19-23] is used to estimate $\omega$.

#### 3.1 Bayesian theorem

According to the Bayesian theorem and the proposed methods of early scholars, the expression can be written as a linear combination of function $\Phi$ of the form

$$y_i(X_i, \omega) = \sum_{m=0}^{M} \omega_m f_m(X_i) = \omega^T \Phi$$  \hspace{1cm} (12)

where $y_i(X_i, \omega)$ represents the function of predicted $K_t$, expressed as

$$y_i(X_i, \omega) = \ln\left|(K_t) - 1\right|$$  \hspace{1cm} (13)

$X_i$ represents the independent variables, and the $\Phi$ is a function of $X_i$. The $\omega_m$ are the parameters of the model and are generally called weights.

In Bayesian theorem, the basic idea is to convert a prior probability density function (PDF) for the parameters $\omega$ into a posterior distribution with the data $D$ in the form

$$P(\omega|D) = \frac{P(D|\omega)P(\omega)}{P(D)}$$  \hspace{1cm} (14)

where $D = [(X_1, y_1), (X_2, y_2), \ldots, (X_n, y_n)]$ represents the $n$ simulated stress concentration factor data pairs, $P(\omega|D)$ is the posterior PDF of $\omega$, $P(D|\omega)$ is the likelihood function, $P(\omega)$ is the prior PDF of $\omega$, and $P(D)$ is a normalizing constant.

To support variances between $y_i(X_i, \omega)$ and $\omega^T \Phi$, the Eq. (7) will be rewritten as

$$y_i(X_i, \omega) = \sum_{m=0}^{M} \omega_m \phi_m(X_i) + \varepsilon$$  \hspace{1cm} (15)

where $\varepsilon$ follow Gaussian distribution expressed as

$$P(\varepsilon|\tau) = N(\varepsilon|0, \tau^{-1})$$  \hspace{1cm} (16)

and $\tau$ follow Gamma distribution as

$$P(\tau) = \Gamma(\tau|a, b) = b^a \tau^{a-1} e^{-b\tau} / \Gamma(a)$$  \hspace{1cm} (17)
Also, an alternative framework based on hierarchical prior is used to constructed $P(\omega)$. Then the prior PDF of $\omega$ is expressed as

$$P(\omega|\alpha) = N(\omega|0, \alpha^{-1})$$

(18)

where $\alpha$ is taken to follow Gamma distribution, which is expressed as

$$P(\alpha) = \Gamma(\alpha|c,d) = \frac{\alpha^c}{\Gamma(c)} \frac{\exp{-\alpha d}}{\exp{-\alpha d}}$$

(19)

Thus, the posterior PDF is derived as below by Bayesian theorem,

$$P(\omega,\tau|\alpha,\tau,D) = \frac{P(D|\omega,\tau)P(\omega|\alpha)P(\tau)}{P(\alpha)}$$

(20)

It is not feasible to compute $P(\omega,\tau,\alpha|D)$ analytically or numerically because $P(D)$ does not have a tractable expression. Consequently, the Gibbs sampling, one of the Markov Chain Monte Carlo (MCMC) algorithms, is used.

3.2 Gibbs sampling

The process for Gibbs sampling is as below,

(i) select the training data pairs,
(ii) initialize the parameters $\alpha$ and $\tau$, and let $k=1$,
(iii) sample the parameter $\omega$ ~ $P(\omega|\alpha,\tau, D)$; sample the parameters $\alpha$ ~ $P(\alpha|\omega,\tau, D)$ and $\tau$ ~ $P(\tau|\omega,\alpha, D)$,
(iv) let $k=k+1$, go back to Step (iii) and repeat the simulation process until $\omega$, $\alpha$ and $\tau$ samples reach stationary states.

3.3 Burn-in period

In practice, an initial set of samples (burn-in) are often discarded to avoid starting biases [24]. The period required for the Markov chain to reach its stationary state is called the burn-in period. The burn-in period is usually determined by plotting the Markov chain samples through time. Moreover, the samples within the burn-in period are discarded. In this paper, by plotting the Markov chain samples, the burn-in period is obtained by visual inspection.

3.4 Data

By FEM simulations, the statistical roughness parameters and corresponding stress concentration factor of 41 different specimens have obtained, which constitutes the datasets, as shown in Table 2.

<table>
<thead>
<tr>
<th>Number</th>
<th>$R_a/\mu m$</th>
<th>$R_y/\mu m$</th>
<th>$R_z/\mu m$</th>
<th>$\rho/\mu m$</th>
<th>$K_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.555986</td>
<td>26.30374</td>
<td>26.11672</td>
<td>16.99651</td>
<td>2.153361439</td>
</tr>
<tr>
<td>2</td>
<td>4.2303</td>
<td>22.98283</td>
<td>22.69772</td>
<td>25.07418</td>
<td>2.105857375</td>
</tr>
<tr>
<td>3</td>
<td>5.440313</td>
<td>31.41201</td>
<td>31.00602</td>
<td>17.03947</td>
<td>2.638873261</td>
</tr>
<tr>
<td>4</td>
<td>5.772174</td>
<td>4.512271</td>
<td>4.475618</td>
<td>137.147</td>
<td>1.153786981</td>
</tr>
<tr>
<td>5</td>
<td>4.025152</td>
<td>25.22172</td>
<td>25.04139</td>
<td>23.22475</td>
<td>1.963668871</td>
</tr>
<tr>
<td>6</td>
<td>5.583996</td>
<td>30.33734</td>
<td>29.96099</td>
<td>19.03229</td>
<td>2.519425105</td>
</tr>
<tr>
<td>7</td>
<td>4.726791</td>
<td>30.61076</td>
<td>30.35464</td>
<td>26.28119</td>
<td>1.983688209</td>
</tr>
<tr>
<td>8</td>
<td>0.331772</td>
<td>2.402115</td>
<td>2.379683</td>
<td>246.3604</td>
<td>1.09061872</td>
</tr>
<tr>
<td>9</td>
<td>2.4</td>
<td>12.32988</td>
<td>12.24221</td>
<td>36.10878</td>
<td>1.509358561</td>
</tr>
<tr>
<td>10</td>
<td>3.693573</td>
<td>26.18617</td>
<td>25.79348</td>
<td>25.35032</td>
<td>2.040652054</td>
</tr>
<tr>
<td>11</td>
<td>0.192818</td>
<td>1.191023</td>
<td>1.179942</td>
<td>569.1562</td>
<td>1.03880591</td>
</tr>
<tr>
<td>12</td>
<td>6.000485</td>
<td>36.44235</td>
<td>36.27377</td>
<td>17.45057</td>
<td>2.673570711</td>
</tr>
<tr>
<td>13</td>
<td>3.252341</td>
<td>23.35157</td>
<td>23.11154</td>
<td>31.73201</td>
<td>1.863292314</td>
</tr>
<tr>
<td>14</td>
<td>1.92</td>
<td>9.863903</td>
<td>9.793769</td>
<td>45.11696</td>
<td>1.402105278</td>
</tr>
<tr>
<td>15</td>
<td>2.404957</td>
<td>14.19217</td>
<td>13.96398</td>
<td>39.43441</td>
<td>1.640159853</td>
</tr>
</tbody>
</table>
### Table 1: Simulation and Prediction Results

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>2.400194</td>
<td>14.57694</td>
<td>14.50951</td>
<td>38.80922</td>
<td>1.622350349</td>
</tr>
<tr>
<td>17</td>
<td>1.721862</td>
<td>11.27853</td>
<td>11.23405</td>
<td>18.84812</td>
<td>2.357763852</td>
</tr>
<tr>
<td>18</td>
<td>5.550449</td>
<td>33.70917</td>
<td>33.55324</td>
<td>56.88085</td>
<td>1.37554257</td>
</tr>
<tr>
<td>19</td>
<td>7.066363</td>
<td>44.00704</td>
<td>43.72666</td>
<td>16.13747</td>
<td>2.733849542</td>
</tr>
<tr>
<td>20</td>
<td>1.018625</td>
<td>6.101202</td>
<td>6.014407</td>
<td>93.90284</td>
<td>1.24962646</td>
</tr>
<tr>
<td>21</td>
<td>2.226748</td>
<td>11.31604</td>
<td>11.21579</td>
<td>41.2961</td>
<td>1.480055216</td>
</tr>
<tr>
<td>22</td>
<td>0.792587</td>
<td>3.752242</td>
<td>3.695435</td>
<td>101.3596</td>
<td>1.185491742</td>
</tr>
<tr>
<td>23</td>
<td>5.69759</td>
<td>40.40472</td>
<td>40.30994</td>
<td>16.74337</td>
<td>2.546558207</td>
</tr>
<tr>
<td>24</td>
<td>1.121265</td>
<td>7.949373</td>
<td>7.830165</td>
<td>83.26314</td>
<td>1.29358941</td>
</tr>
<tr>
<td>25</td>
<td>4.074499</td>
<td>24.40481</td>
<td>24.05763</td>
<td>41.9364</td>
<td>2.184433519</td>
</tr>
<tr>
<td>26</td>
<td>5.337771</td>
<td>34.96346</td>
<td>34.82555</td>
<td>18.41903</td>
<td>2.325390202</td>
</tr>
<tr>
<td>27</td>
<td>1.118098</td>
<td>7.006034</td>
<td>6.955942</td>
<td>83.35896</td>
<td>1.247479891</td>
</tr>
<tr>
<td>28</td>
<td>5.59575</td>
<td>32.3095</td>
<td>31.89191</td>
<td>16.57165</td>
<td>2.690652606</td>
</tr>
<tr>
<td>29</td>
<td>3.053939</td>
<td>17.03504</td>
<td>16.92747</td>
<td>39.79985</td>
<td>1.681130644</td>
</tr>
<tr>
<td>30</td>
<td>1.894048</td>
<td>12.40639</td>
<td>12.35745</td>
<td>51.71469</td>
<td>1.416110263</td>
</tr>
<tr>
<td>31</td>
<td>5.045286</td>
<td>36.30095</td>
<td>36.14158</td>
<td>18.61619</td>
<td>2.129087822</td>
</tr>
<tr>
<td>32</td>
<td>4.597006</td>
<td>23.19896</td>
<td>22.96311</td>
<td>33.14648</td>
<td>1.892614248</td>
</tr>
<tr>
<td>33</td>
<td>3.431884</td>
<td>21.76301</td>
<td>21.43352</td>
<td>17.53594</td>
<td>2.207043075</td>
</tr>
<tr>
<td>34</td>
<td>3.400587</td>
<td>20.05454</td>
<td>19.89164</td>
<td>30.91224</td>
<td>1.751842258</td>
</tr>
<tr>
<td>35</td>
<td>4.118261</td>
<td>24.06545</td>
<td>23.86996</td>
<td>25.78116</td>
<td>1.918798908</td>
</tr>
<tr>
<td>36</td>
<td>6.398332</td>
<td>44.80768</td>
<td>44.53094</td>
<td>16.74337</td>
<td>2.546558207</td>
</tr>
<tr>
<td>37</td>
<td>3.80513</td>
<td>36.3095</td>
<td>36.14158</td>
<td>18.61619</td>
<td>1.29087822</td>
</tr>
<tr>
<td>38</td>
<td>5.045286</td>
<td>36.30095</td>
<td>36.14158</td>
<td>18.61619</td>
<td>1.29087822</td>
</tr>
<tr>
<td>39</td>
<td>3.053939</td>
<td>17.03504</td>
<td>16.92747</td>
<td>39.79985</td>
<td>1.681130644</td>
</tr>
<tr>
<td>40</td>
<td>1.118098</td>
<td>7.006034</td>
<td>6.955942</td>
<td>83.35896</td>
<td>1.247479891</td>
</tr>
<tr>
<td>41</td>
<td>2.226748</td>
<td>11.31604</td>
<td>11.21579</td>
<td>41.2961</td>
<td>1.480055216</td>
</tr>
</tbody>
</table>

#### 3.5 The stability of the method

The coefficient of determination ($R^2$) and mean percentage error (MPE) is used to evaluate the accuracy of the proposed method, which are obtained as below. Where $(K_i)$ is the $i$th simulated data, $(K_{i,pre})$ is the $i$th predicted value. The best value for $R^2$ is 1.0, and the smaller the value of MPE, the better the predicted result.

$$R^2 = \left( \frac{n \sum \mu_{(K_i)} \times (K_i) - \sum \mu_{(K_i)} \sum (K_i)}{\sqrt{n \sum \mu_{(K_i)}^2 \left( \sum \mu_{(K_i)} \right)^2 \left( \sum (K_i)^2 \right) - \left( \sum (K_i) \right)^2}} \right)^2$$

$$\text{MPE} = \left( \frac{1}{n} \sum \left( K_{i,pre} - (K_i) \right) \right) \times 100\%$$

The procedure for a set of experiments is described below. Five pairs are randomly selected from the datasets as training samples, and the rest of the data pairs are testing sets. The predicted model of the stress concentration factor is obtained by the method mentioned in Eq. (20). Comparing the predicted stress concentration factor with the simulated value, $R^2$ and MPE are calculated.

A total of 1000 experiments are conducted. The frequency histogram of MPE and $R^2$ are plotted in Fig. 4 and Fig. 5, where “Mean” represents the mean value of 1000 experiments, and “SD” represents standard deviation. By the figures, for 1000 experiments, the mean value of MPE is 4.2%, and its standard deviation is 0.05. Meanwhile, for most experiments, the $R^2$ is around 0.94, and the standard deviation is 0.004. MPE and $R^2$ remain within the error tolerance range, which indicates the excellent stability of the method.
4. Results

Firstly, using the five data from datasets as training samples, the predicted model of stress concentration factor is obtained through the proposed method. The first 500 samples of 10,000 samples obtained by Gibbs sampling are discarded because of the burn-in period. Based on the remaining 9,500 samples, the predicted mean stress concentration factor is calculated from their means and standard deviations. The predicted formula between $K_t$ and statistical roughness parameters of the surfaces is expressed as

$$K_t = 1 + 1.915 \left( \frac{R_y}{R} \right)^{-0.223} + \frac{R_y}{R} + 0.549 \left( \frac{R_y}{\rho} \right)^{0.549}$$

(23)

To make the results more intuitive, each predicted $K_t$ in the 41 datasets are compared with the simulated $K_t$ in Fig. 6.

It is showed by open circles from mean $K_t$ and error bars for uncertainly in terms of $K_t + 1.96SD$, where SD is the standard deviation of each predicted $K_t$, to show the ranges of data. Fig. 6 also includes the true 1:1 line for comparison.

It shows that open circles fluctuate around the 1:1 line, and most samples have relative errors smaller than 10%, and all samples smaller than 20%. The maximum error is found to be 15.4%, and the average error is 3.7%. For quantitative the predicted model, $R^2$ and MPE between predicted and simulated $K_t$ are calculated, obtained as $R^2 = 0.95$, $MPE = 3.76\%$. The large $R^2$ and small MPE together indicate that the predicted model is reasonably accurate.
For comparison, Fig. 7 also includes the stress concentration factor predicted by the Neuber model and Arola-Ramulu model. It can be seen that $K_t$ predicted by the proposed method is closer to the 1:1 line, which indicates our predicted values are more consistent with the simulated values than the other two models. The average error of these two models is 9.87% and 21.23%, while for the proposed method, the average error is 3.76%.

To verify the influence of the number of data pairs ($n_s$) on estimation performance, the randomly 5, 15, 30 groups of datasets are selected as training samples to obtain the corresponding predicted expressions. The predicted $K_t$ are compared with simulated $K_t$, as shown in Fig. 8. The values predicted by the three correlation expressions all fluctuate around the 1:1 line. Furthermore, $R^2$ and MPE of three correlation expressions are plotted in Fig. 9. $R^2$ and MPE are kept around 0.94 and 4%, respectively.
Fig. 8 Predicted $K_t$ obtained by different $n_s$ ($n_s=5, 15, 30$) versus simulated $K_t$

Fig. 9 $R^2$ and MPE calculated by different $n_s$ ($n_s=5, 15, 30$)

It indicates no significant effect on the predicted results with decreasing the number of training samples in the appropriate range. In other words, the estimated error of the proposed method is on the reasonable range when the number of data pairs decreases. The results demonstrate that the proposed method performs well under limited training sets.

The above analysis proved that the proposed method could efficiently obtain the stress concentration factor explicit and has better accuracy than the previous models. Moreover, this method is quite convenient to be conducted through only a few FEM simulations or tests for certain machining processes.

5. Conclusions

This article proposes a novel approach to use machine learning to obtain correlation expression between the stress concentration factor and statistical roughness parameters. The training samples and testing samples are selected from the database, which is constructed by FEM simulations. Using the Bayesian learning with Gibbs sampling, one of Markov Chain Monte Carlo (MCMC) algorithm, the correlation expression between the stress concentration factor and statistical roughness parameters of the surfaces is estimated. The accurate correlation expression gives fairly good agreements with direct FEM simulations, which demonstrates the feasibility of using the proposed method. Compared to other models, the proposed method has the advantages of high accuracy and broad adaptability. In conclusion, this work demonstrates the viability of using machine learning to predict the stress concentration factor. It provides a simple and efficient approach to obtain the predicted stress concentration factor for the rough surfaces under a specific type of processing.

References

   https://doi.org/10.1016/0013-7944(94)00297-U.


