*Journal of Network Intelligence* ©2025 ISSN 2414-8105
**Taiwan Ubiquitous Information** Volume **10**, Number **3**, August **2025**

# Neural Network-based Error Prediction Modeling for High-Speed CNC Machining

Tai-Xiang Li[1,*], Hong-Min Zhong[2]

[1]Intelligent Manufacturing College, Henan Polytechnic, Zhengzhou 450046, P. R. China
WXH202403@126.com

[2]College of Engineering, Mariano Marcos State University, Batac City 2906, Philippines
ue9770@163.com

[*]Corresponding author: Tai-Xiang Li

Received July 8, 2024, revised November 28, 2024, accepted March 2, 2025.

Abstract. *Traditional CNC machining error prediction methods mainly rely on physical models and empirical formulas, but it is difficult to deal with complex nonlinear relationships and multidimensional data. In order to solve this problem, this study proposes a high-speed CNC machining error prediction model based on neural network. In this paper, by introducing the Temporal Pattern Attention (TPA) mechanism combined with Temporal Convolutional Network (TCN) and Long Short-Term Memory (LSTM) network model (TPA-TCN-LSTM), which aims to improve the accuracy and efficiency of error prediction. The proposed model takes advantage of the efficient feature extraction capability of TCN and the temporal dependency modelling capability of LSTM, and further improves the model's focus on the important time steps through the TPA mechanism to achieve high-precision prediction of high- speed CNC machining errors. First, an input data set containing linear and nonlinear features is constructed by analysing the motion control characteristics and error generation mechanism of the CNC system. Then, through causal convolution and extended convolution, TCN effectively captures the long and short-term dependencies in the time series data. Secondly, the TPA mechanism further focuses on important time steps to improve feature extraction. The improved LSTM simplifies the structure and speeds up data processing. Finally, the prediction effect of the neural network under different parameters is compared, and a set of best experimental parameters is selected. The experimental results show that the TPA-TCN-LSTM model performs well in five-axis CNC machining error prediction and meets the expected results.*
**Keywords:** high-speed CNC machining; error prediction; TCN; LSTM; TPA

1. **Introduction.** High-Speed CNC Machining plays a crucial role in modern manufacturing industry, and its accuracy directly affects product quality and productivity [1, 2]. However, during high-speed machining, the generation of machining errors is inevitable due to the complexity of the mechanical system and the operating environment. These errors not only lead to a decrease in product quality, but also may increase production costs and delay delivery time. Therefore, effective prediction and control of high-speed CNC machining errors is of great research value and practical application [3]. Through the prediction model, it can be monitored and adjusted in real time during the machining process, so as to improve the machining accuracy, reduce the scrap rate, and enhance the production efficiency.

The application of neural networks in high-speed CNC machining error prediction has great potential. Traditional error prediction methods mainly rely on physical models and empirical formulas, but these methods are difficult to deal with complex nonlinear relationships and multidimensional data [4, 5]. Neural networks, with powerful data-driven characteristics and nonlinear mapping capabilities, can automatically extract features from a large amount of historical data and perform error prediction. In recent years, deep learning models such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory Networks (LSTMs) have performed well in the field of time-series data processing, and have gradually been applied to CNC machining error prediction, showing good results [6]. In recent years, deep learning techniques, especially Temporal Convolutional Network (TCN) [7, 8, 9], have shown great potential in the field of sequence prediction. TCN, through its unique causal convolution and dilation convolution structures, is able to efficiently capture long and short-term dependencies in time series data while maintaining efficient computational performance.

The research objective of this paper is to develop a high-speed CNC machining error prediction model (TPA-TCN-LSTM) based on Temporal Pattern Attention (TPA) [10, 11] combined with TCN and LSTM. This model aims to take advantage of the efficient feature extraction capability of TCN and the timing-dependent modelling capability of LSTM, and further improve the model's focus on important time steps through the TPA mechanism, so as to achieve high-precision prediction of high-speed CNC machining errors. In this paper, the construction process of the model, the feature selection method and the simulation experiment results are described in detail.

1.1. **Related work.** In recent years, the research on error prediction of high-speed CNC machining has received extensive attention and achieved a series of important results. Traditional error prediction methods mostly use physical models and empirical formulas, which have a certain degree of accuracy under simple working conditions, but are often inadequate under complex working conditions. In order to improve the prediction accuracy, scholars began to explore data-driven methods. Ramesh et al. [12] proposed an error prediction model based on support vector machine (SVM), which can achieve effective prediction of CNC machining error by training on historical data. However, SVM has high computational complexity when dealing with high-dimensional data, which makes it difficult to adapt to the real-time prediction needs in real production. Radhika et al. [13] introduced the Random Forest algorithm, which improves the stability and prediction accuracy of the model by integrating the learning method. Nevertheless, the Random Forest model has limitations in dealing with nonlinear and time series features. In contrast, neural networks show great potential in the field of error prediction due to their powerful nonlinear fitting and feature extraction capabilities. Saric et al. [14] proposed an error prediction method based on Multilayer Perceptron (MLP), which achieves good prediction results by modelling complex nonlinear relationships through the structure of multilayer neural networks. However, the MLP has limited performance in dealing with time series data and is difficult to capture long time dependencies. For this reason, Liu et al. [15] used LSTM for error prediction, which effectively solved the long time dependence problem and improved the prediction accuracy by introducing memory units.

Although neural networks show many advantages in CNC machining error prediction, there are still some problems and shortcomings in existing research. Firstly, traditional neural network models tend to require long training time when dealing with time series data and have limited ability to handle large-scale data. The CNN-based error prediction model proposed by Bhandari and Park [**?**] improves the computational efficiency of the model through convolutional operations, but it is still deficient in capturing long time dependencies. Secondly, the existing models generally ignore the importance differences of different time steps and fail to pay effective attention to critical time steps. To overcome these problems, Sun et al. [16] introduced the attention mechanism, which enhances the model's ability to capture important features by weighting different time steps, but the complexity and computational cost of this model increase significantly when dealing with multivariate time series data. In addition, most of the studies mainly focus on the optimisation of a single model and lack the exploration of multi-model fusion.

1.2. **Motivation and contribution.** In existing research, although LSTM networks perform better in capturing long-time dependencies, they are difficult to effectively handle large-scale and high-frequency data in practical applications due to long training time and high computational complexity. Many existing neural network models fail to fully consider the importance differences of different time steps when dealing with time-series data, resulting in deficiencies of the models in capturing key features. To remedy these deficiencies, a TPA-TCN-LSTM model is proposed in this paper. The main innovations and contributions of this work include.

(1) Aiming at the problem of insufficient capture of long-time dependencies, a model combining TCN and LSTM is proposed. In this case, TCN, as an efficient time series processing model, can effectively capture long time dependencies through the combination of causal convolution and null convolution. Meanwhile, the parallel computing feature of TCN significantly improves the training speed and computational efficiency of the model.

(2) To address the problem of ignoring the importance differences of different time steps, a TCN model considering TPA is proposed, which can effectively weight the importance of different time steps to ensure that the model can focus on the time steps that have the greatest impact on the prediction results. The TPA mechanism not only improves the model's ability to capture the key features, but also simplifies the structure of the model to a certain extent.

(3) Improvements were made to the LSTM network by simplifying its structure and retaining only one gate to reduce the number of parameters and speed up the learning of processed data, while ensuring the accuracy of the model. This improvement helps to process large-scale, high-frequency data and increase the efficiency of the model in practical applications.

## 2. Feature selection based on CNC system models.

2.1. **Typical motion control system.** The goal of a motion control system is to control the motion of the end mechanical part so that it follows a predetermined trajectory. It consists of the following main components: controller, driver, motor, end drive mechanism, and measuring elements.

(1) The controller receives the position command to be moved to and sends the processed position command to the driver. Usually, the controller uses PID control algorithms to process the input signals [17, 18] in order to reduce the error of the system. The output signals of the PID controller are used to control the motion of the motor through the actuator.

(2) The driver receives the instruction from the controller and converts the instruction into a pulse signal that can be understood by the motor. The driver plays the role of power amplification and signal conversion in the whole system. According to the position feedback signal of the motor, the driver can make real-time adjustments to ensure accurate control.

(3) The motor moves at a constant rate in accordance with the received pulse signal and is connected to the end screw guide through a coupling to drive the screw guide. There are various types of motors, including stepper motors and servo motors, the specific selection depends on the system requirements and performance indicators.

(4) Measuring elements (e.g., angle encoders, linear encoders, etc.) are used to monitor the position of the motor in real time and feed the measured data back to the driver and controller to form a closed-loop control system. In this way, high-precision motion control can be achieved.

In actual motion control systems, most of the motors at the end are controlled by three-loop control (current loop, speed loop, position loop). The current loop, speed loop, and position loop are in order from the inside out. Due to the faster response of the current loop, the current loop can usually be reduced to a proportional link. With this multi-level control structure, the motion control system is able to maintain high accuracy and stability in response to various disturbances.

In motion control systems, the transfer function (Transfer Function) is an important tool for describing the relationship between the inputs and outputs of the system. Assuming that the controller of the position loop uses proportional control, the transfer function of the system can be simplified as follow:

$$G(s) = \frac{K_p K_{rg}}{s} \tag{1}$$

where $K_p$ is the proportional gain and $K_{rg}$ is the motor gain.

Assuming that the position command is a ramp signal,

$$X(t) = vt \tag{2}$$

where $v$ is the desired velocity. With the transfer function, the tracking error can be obtained as follow:

$$E(s) = X(s) \cdot G_e(s) = \frac{v}{K_p K_{rg}} \tag{3}$$

This shows that in the case of proportional control, the tracking error is inversely related to the speed. In summary, a typical motion control system can effectively achieve precise control of the mechanical unterminal through the multi-loop control strategy and feedback mechanism. This kind of system is widely used in the fields of CNC machining and robot control.

2.2. **Neural network linear feature selection.** In the CNC system model, the selection of appropriate features is crucial for the training of the neural network model. Linear feature selection mainly consists of features related to key variables in the CNC system that can describe the dynamic behaviour and performance of the system to some extent. In order to accurately predict CNC machining errors, linear features of the CNC system need to be extracted and analysed.

In CNC systems, common linear features include position, velocity, and acceleration. These features can be measured or calculated by sensors and used as inputs to the neural network model. Specifically, position sensors (e.g., scales, encoders, etc.) in CNC systems provide accurate position information; velocity can be obtained by differential computation of position; and acceleration can be obtained by differential computation of velocity. By extracting and analysing these linear features, effective input data can be provided to the neural network model, thus improving the accuracy of error prediction.

For five-axis CNC machine tools, position and velocity are two key linear features [19, 20]. The position feature reflects the difference between the actual position and the target position of each axis of the machine, while the velocity feature reflects the motion of each axis of the machine. To capture these features, the following steps can be used:

The actual position of each axis of the machine is obtained using position sensors (encoders) and compared with the target position to calculate the position error [21].

This process can be expressed as:

$$e_p(t) = p_{target}(t) - p_{actual}(t) \tag{4}$$

where $e_p(t)$ is the position error, $p_{target}(t)$ is the target position, and $p_{actual}(t)$ is the actual position.

The velocity of each axis of the machine is calculated by numerically differentiating the position data [22]. The velocity characteristics are calculated as:

$$v(t) = \frac{dp_{actual}(t)}{dt} \tag{5}$$

where $v(t)$ is the velocity.

The acceleration of each axis of the computer bed is calculated by numerically differentiating the velocity data [23]. The acceleration characteristics were calculated as follows.

$$a(t) = \frac{dv(t)}{dt} = \frac{d^2 p_{actual}(t)}{dt^2} \tag{6}$$

where $a(t)$ is the acceleration.

Through the above steps, linear features of each axis of the machine tool can be obtained, and these features can be used as inputs to the neural network model to improve the accuracy of the error prediction. In addition, the training effect and stability of the model can be further improved by standardising these features.

2.3. **Neural network nonlinear feature selection.** In the process of CNC machining, nonlinear error is one of the important factors that lead to the degradation of machining accuracy. Common nonlinear factors include saturation, dead zone, friction and nonlinear interference. The existence of these nonlinear factors makes the system error difficult to be accurately predicted by a simple linear model. Therefore, the extraction and analysis for these nonlinear features are crucial to improve the performance of neural network error prediction models.

In practical CNC systems, nonlinear factors such as friction, backlash and saturation effects can significantly affect machining accuracy [24, 25]. For example, friction can be categorised as static friction, Coulomb friction and viscous friction, each of which affects the system differently under different operating conditions. In order to improve the accuracy of error prediction, these nonlinear features must be extracted and modelled.

According to the superposition method, the unit impulse response of the viscous force friction for the effect of tracking error can be obtained as:

$$G_{ef}(s) = h_f(s) T_f(s) \tag{7}$$

where $T_f(s)$ is the Laplace expression for viscous force friction, and $h_f(s)$ is the expression for the unit impulse response when the position input is zero.

Setting the viscous force friction as a constant term, tracking error due to friction can be obtained as:

$$e_f(t) = m \int_{-\infty} h_f(t - \tau) d\tau \tag{8}$$

where $m$ is the constant term of viscous force friction.

## 3. **TPA-TCN-LSTM network model.**

3.1. **Principle of TCN.** TCN is a neural network model specifically designed to process time series data. Compared with traditional Recurrent Neural Networks (RNN), TCN has higher computational efficiency and longer history information capturing ability. The basic structure of TCN includes causal convolution, dilation convolution, and residual connectivity [26].

TCN is a deep learning model designed for time series data, which captures local patterns and long-term dependencies in time series through convolutional operations. The core principle of TCN is to utilise the causal convolution structure, which ensures that the network is temporal in processing time series, i.e., the output of the network only relies on the current and past inputs, and not on the future inputs.

Due to the parallel nature of convolutional operations, TCNs can efficiently process large-scale data and are easier to train than RNNs. By expanding the convolution, TCN can capture long-range temporal dependencies without increasing the number of parameters. The combination of causal convolution and residual connectivity ensures causality and enhances the stability and prediction accuracy of the model.

3.2. **Causal convolution and dilation convolution.** Causal convolution is one of the core components of TCN. In traditional convolutional networks, the output of each layer shares information with future timesteps, which violates the causal relationship of time series data. Causal convolution ensures that the direction of information flow is from the past to the future by masking the information of future timesteps and using only past and current timestep data for convolution operations [27]. The representation of causal convolution is shown below:

$$H(T) = (X * f)(T) = \sum_{i=0}^{k-1} f(i) \cdot X(T-i) \tag{9}$$

where $X$ is the input sequence, $f$ is the convolutional kernel, $T$ is the time step, and $k$ is the size of the convolutional kernel.

To further expand the receptive field, TCN introduces Dilated Convolution. Dilated Convolution builds on Causal Convolution by introducing intervals when the convolution kernel is applied in order to expand the receptive field without increasing the computational effort. Dilated Convolution skips a certain number of time steps during the convolution process by specifying a dilation rate, enabling the network to capture dependencies over a longer time span with fewer layers. The representation of dilated convolution is shown as follow:

$$H(T) = (X *_d f)(T) = \sum_{i=0}^{k-1} f(i) \cdot X(T - d \cdot i) \tag{10}$$

where $d$ is the dilation rate. By adjusting the dilation rate, dependencies over longer time spans can be captured efficiently.

In TCN, causal and null convolution are often used in combination for more efficient and accurate temporal data processing. Causal convolution ensures the correctness of temporal dependencies, while null convolution captures long time dependencies by extending the sensory field. This combination allows TCNs to capture more useful information while maintaining efficient computation when processing complex temporal data.

3.3. **Considering TPA for TCN.** The traditional attention mechanism can effectively improve the performance of neural network model training, however, it only focuses on the temporal weight of a single sequence, and lacks the understanding of the intrinsic relationship between different temporal sequences when facing problems such as cyclic patterns of cross-domain multi-events and temporal patterns of multivariate time series data.

For the nonlinear relationships, the temporal pattern attention mechanism is used to enhance the algorithm's understanding of the multi-temporal connotations and further explore the intrinsic correlations of the sequence data. The temporal pattern attention mechanism uses multiple one-dimensional convolutional filters to extract feature vectors from TCN hidden states, which is able to learn multivariate dependencies from different time steps, thus improving the learning efficiency and training capability.

Assume that the hidden layer information generated by the TCN network is $H = \{h_{t-\omega}, h_{t-\omega+1}, \ldots, h_{t-1}\}$, where $h_t$ is the hidden layer information at the $t$-th moment, and $\omega$ is the length of the temporal sliding window. The original hidden layer information $H$ is convolved with $l$ convolution kernels to produce an $n \times l$ dimensional matrix:

$$HC_{i,j} = \sum_{l=1}^{n} H_{i,(t-\omega-1+l)} \times C_{j,t-\omega+l} \tag{11}$$

where $\alpha_i$ denotes the strength of the influence of each row of $c$ on the upcoming prediction of $h_t$, i.e., the magnitude of the influence of each time series on $h_t$. The weights $\alpha_i$ are calculated by

$$f(HC, h) = (HC)^T W_h \tag{12}$$

$$\alpha = sigmoid(f(HC, h)) \tag{13}$$

where $c$ denotes the vector of the $i$-th row in $HC$. Weighted summation of $c$ yields the combined time impact of all rows of $c$ on $h_t$:

$$v_t = \sum_{i=1}^{n} \alpha_i HC_i \tag{14}$$

The actual predicted values are obtained by weighting the impacts as well as $h_t$:

$$h_t' = W_h h_t + W_v v_t \tag{15}$$

The structure of the TPA-TCN network constructed from this is shown in Figure 1, and the neural network consists of an input layer, a TCN network layer, a TPA layer, a Flatten layer, a Dense layer and an output layer. The input layer inputs historical time series data, influencing factors and other time series data into the network sequentially; the TCN layer convolution unit consists of a one-dimensional dilated causal convolution layer, an activation function and a Dropout layer, which is used for the extraction of features of the input data; the TPA layer receives the feature vectors output from the TCN layer and obtains the temporal pattern attention weights through computation; the Flatten layer transforms the multidimensional feature matrix of TPA layer into one-dimensional feature matrix, and maps the feature matrix to the sample labelling space and outputs the predicted values through Dense layer and activation function.

3.4. **Improvement of LSTM.** The LSTM has three special gates, namely the forgetting gate $f_t$, the input gate $i_t$ and the output gate $O_t$, and the hidden unit $C'$, which is used to regulate the information flow in the network. Where $C_{t-1}$ is the output of the previous hidden cell, $C_t$ is the output of this hidden cell, $h_{t-1}$ is the output of the previous cell, $h_t$ is the output of this cell, $x_t$ is the input at this moment, $\sigma$ is the sigmoid function, and tanh is the hyperbolic tangent function. However, although the traditional LSTM network performs well, the large amount of data for 5-axis machining leads to a large number

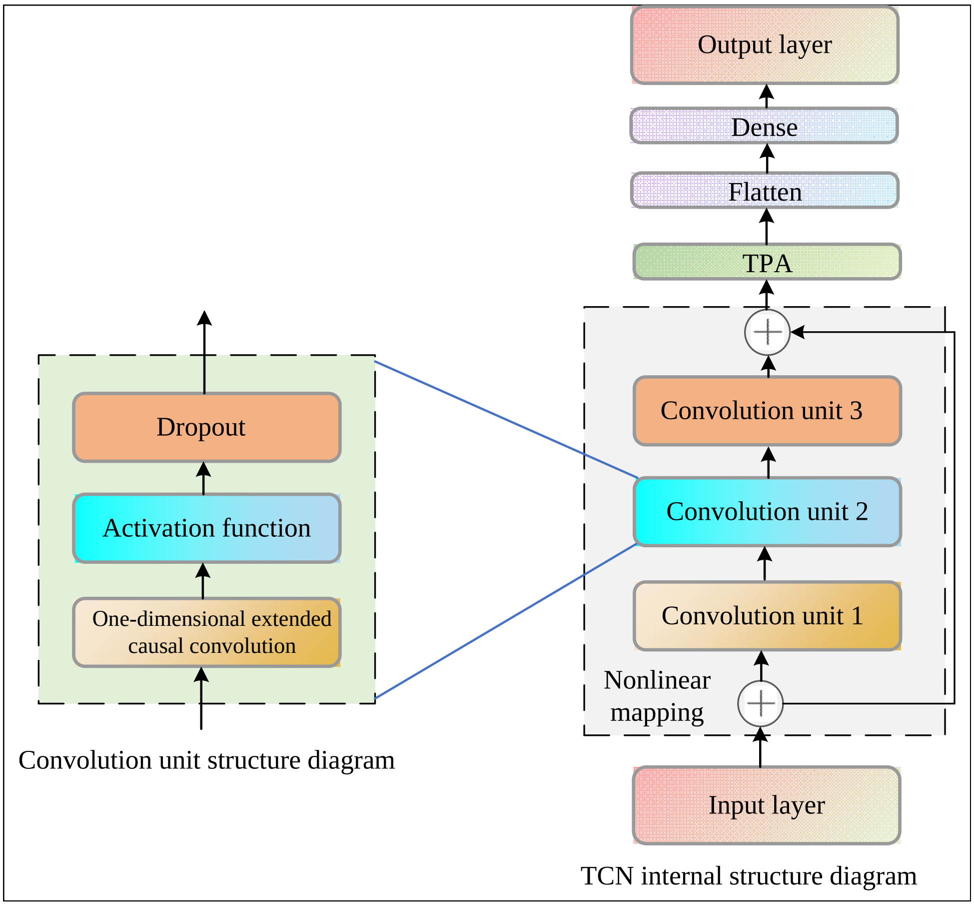


Figure 1. TPA-TCN network architecture

of parameters need to be learnt for each gate, which results in a long running time of the LSTM. Therefore, this paper proposes a simplified LSTM network to accelerate the learning speed of machining data.

The improved LSTM structure retains only 1 gate, which can effectively reduce the number of parameters while extracting the hidden timing information to ensure the accuracy of the model.

$$\begin{cases} \tilde{C}_t = \tanh\left(\mathbf{W}_{x\tilde{c}} \cdot x_t + \mathbf{W}_{h\tilde{c}} \cdot h_{t-1} + \mathbf{b}_{\tilde{c}}\right); \\ \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}; \\ g_t = \sigma\left(\mathbf{W}_{xg} \cdot x_t + \mathbf{W}_{hg} \cdot h_{t1} + \mathbf{P}_{cg} \cdot C_{t-1} + \mathbf{b}_g\right); \\ \sigma(x) = \frac{1}{1+e^{-x}}; \\ C_t = \left(\tilde{C} + C_{t-1}\right) \times g_t; \\ h_t = g_t \times \tanh\left(C_t\right) \end{cases} \tag{16}$$

where $x_t$ is the input; $h_{t-1}$ is the previous moment output; $W_{xc'}$, $W_{hc'}$, $W_{xg}$, $W_{hg}$, and $P_{cg}$ are the corresponding weight matrices; $b_{c'}$ and $b_g$ represent bias vectors; $g_t$ is the output gate.

## 4. **5-axis tracking error prediction based on TPA-TCN-LSTM.**

**4.1. Generation and analysis of training data.** In the CNC machining error prediction model, the quality and quantity of training data are crucial to the accuracy of the model. In order to effectively train neural network models, diverse and representative training data need to be generated. This section describes the five-axis CNC machine trajectory generation method used to train the TPA-TCN-LSTM model and analyses the characteristics of the generated data.

Generating training trajectories for 5-axis machine tools is fitted with Non-uniform Rational B-spline (NURBS) curves, which are able to accurately describe complex paths, and diversified 5-axis trajectories can be constructed by generating sequences of random control points and independent variables.

Firstly, the control point $P_i$ is generated through a normal distribution, setting the mean $E$ and standard deviation $\sigma$:

$$P \sim N(E, \sigma^2) \tag{17}$$

where $E$ is usually set to the centre of the machine travel and $\sigma$ depends on the complexity of the trajectory.

The sequence of independent variables $U$ is generated by random numbers with non-uniform distribution:

$$U = [u_0, u_0 + \Delta u_1, \ldots, u_0 + \sum_{i=0}^{n} \Delta u_i] \tag{18}$$

where $u_0$ is the initial value and $\Delta u_i$ is a random increment.

Then, using the generated control points $P_i$ and the sequence of independent variables $U$, the five-axis trajectories are fitted by the NURBS curve formulation:

$$C(u) = \sum_{i=0}^{n} N_{i,p}(u) P_i, \quad u \in [a, b] \tag{19}$$

where $N_{i,p}(u)$ is the B-spline basis function; $a$ and $b$ are the range of independent variables.

To ensure the validity of the training data, the generated trajectories need to be characterised, mainly including the distribution of velocity and acceleration features. The Coefficient of Variation (CV) was used to assess the homogeneity of the sample points:

$$CV = \frac{\sigma_D}{D} \tag{20}$$

where $D_i$ is the number of sample points in the $i$th interval and $D$ is the average number.

By calculating the coefficient of variation for trajectories with different numbers of sample points, it is found that the coefficient of variation decreases gradually when the number of sample points increases, indicating that the trajectories are better able to traverse the entire space of motion, and thus stimulate the properties of the system more fully.

4.2. **Neural network input and output data.** When constructing a five-axis CNC machining error prediction model based on TPA-TCN-LSTM, the selection of appropriate input and output data is crucial for improving the prediction accuracy of the model.

**(1) Selection of input data.** The input data for the neural network model should be able to adequately reflect the dynamic characteristics of the 5-axis CNC system. The input data used include linear features such as position, velocity and acceleration, as well as non-linear features such as friction, backlash and saturation effects. Position, velocity and acceleration characteristics can be measured or calculated by sensors, and are the basic variables reflecting the motion state of the system. For a time step $t$, its input characteristics can be expressed as $x(t) = [p(t), v(t), a(t)]$ where $p(t), v(t)$ and $a(t)$ denote the position, velocity and acceleration, respectively. Nonlinear features include friction $f(t)$, backlash $b(t)$ and saturation effect $s(t)$. These features can be obtained from experimental measurements or data analysis and are important variables in describing the nonlinear behaviour of the system.

By combining linear and nonlinear features, a more accurate and comprehensive input feature set can be constructed. Assuming that the input feature vector at time step $t$ is $x(t)$, the input data to the neural network model can be expressed as:

$$X = [x(t-\tau), \ldots, x(t-1), x(t)] \tag{21}$$

where $\tau$ is the length of the time window.

**(2) Selection of output data.** The output data of the neural network model is the predicted value of the future time step error. For a 5-axis CNC system, the output data mainly includes position error ($e_p(t)$), velocity error ($e_v(t)$) and acceleration error ($e_a(t)$). Specifically, the output data can be expressed as follows:

$$y(t) = [e_p(t), e_v(t), e_a(t)] \tag{22}$$

where $e_p(t)$ denotes the position error, $e_v(t)$ denotes the velocity error, and $e_a(t)$ denotes the acceleration error.

The input and output data need to be normalised in order to improve the training efficiency of the neural network and the stability of the model:

$$x_{\text{norm}}(t) = \frac{x(t) - x_{\min}}{x_{\max} - x_{\min}}, \qquad y_{\text{norm}}(t) = \frac{y(t) - y_{\min}}{y_{\max} - y_{\min}} \tag{23}$$

where $x_{\min}, x_{\max}$ are the minimum and maximum values of the input data; $y_{\min}, y_{\max}$ are the minimum and maximum values of the output data, respectively.

For model training, the normalised input data $X_{\text{norm}}$ and output data $y_{\text{norm}}$ are fed into the neural network model. The model parameters are iteratively updated by a loss function (e.g. mean square error loss function) and an optimisation algorithm (e.g. Adam's optimiser) until the loss function converges or a preset number of training rounds is reached:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_{i,\text{norm}} - \hat{y}_{i,\text{norm}})^2 \tag{24}$$

where $n$ is the number of samples and $\hat{y}_{i,\text{norm}}$ is the predicted value for the $i$th sample.

## 5. Tracking error prediction simulation experiment.

### 5.1. Simulation description.

In order to quantitatively evaluate the results of the five-axis tracking error prediction, the following performance metrics are defined, where $e\%$ denotes the prediction deviation of the tracking error:

$$e\% = |e - \hat{e}| \tag{25}$$

where $e$ denotes the true value of the tracking error and $\hat{e}$ denotes the predicted value.

Define $e\%_{\max}$ to denote the maximum prediction deviation of the tracking error, and $e\%_{\text{rms}}$ to denote the root mean square value of the prediction deviation of the tracking error:

$$e\%_{\max} = \max|e - \hat{e}| \tag{26}$$

$$e\%_{\text{rms}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(e_i - \hat{e}_i)^2} \tag{27}$$

5.2. **Comparison of different neural network parameters.** When constructing and training the TPA-TCN-LSTM based 5-axis CNC machining error prediction model, choosing the appropriate neural network parameters is crucial to improve the performance of the model. In order to find out the best combination of parameters, a comparative analysis of the models under different parameter configurations is needed.

Firstly, the prediction effect of the neural network is compared when using different activation functions. The learning rate is set to 0.01, the number of neural network layers is set to 3, and the expansion rate is set to 2. Figure 2 shows that among the four activation functions, the Sigmoid activation function has the worst prediction effect, and the three activation functions, namely ReLU, LeakyReLU, and GELU, have relatively close prediction effects, among which, the GELU activation function has the best prediction effect.

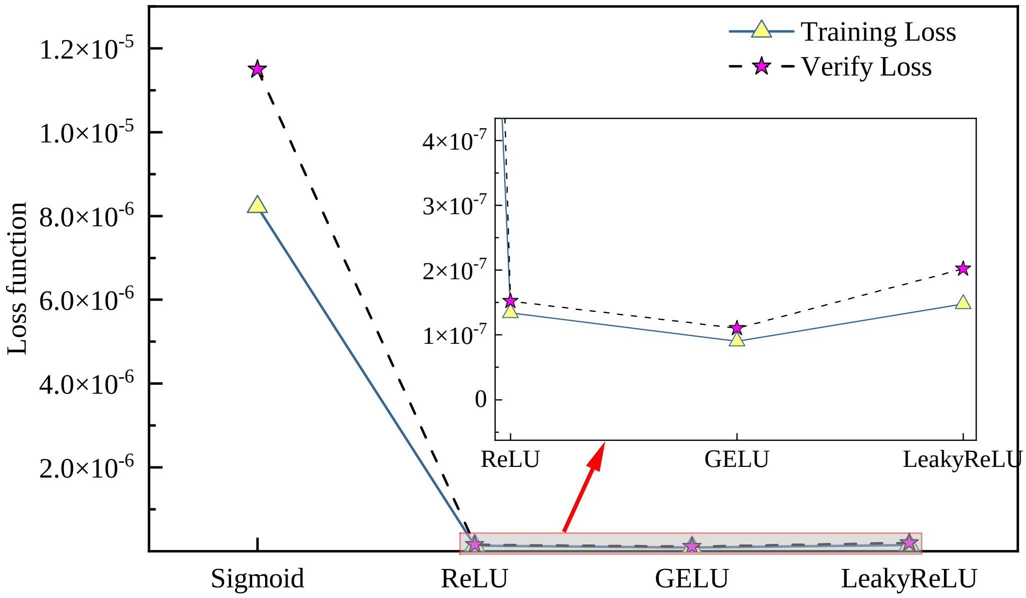


Figure 2. Comparison of different activation functions

The dilation rate determines the number of steps between convolution kernel applications in the dilation convolution. In the experiments, the model effects at dilation rates of 1, 2, 3, and 4 were compared, as shown in Figure 3. The results show that the TPA-TCN-LSTM model can achieve a better balance between computational complexity and prediction accuracy when the expansion rate is 3.

By comparing the model performance with different parameter configurations, it can be concluded that the optimal activation function of the TPA-TCN-LSTM model is GELU and the optimal dilatation rate is 3. In the optimal parameter scenario, the prediction of the tracking error of the trajectory translation axis of the NURBS test set is shown in Table 1.

Table 1. The prediction of the tracking error of the trajectory translation axis

| **Performance indicators** | **X-axis** | **Y-axis** | **Z-axis** |
|---|---|---|---|
| Maximum prediction bias $e\%_{\max}$ ($\mu$m) | 0.968 | 0.628 | 0.625 |
| Root mean square value of prediction bias $e\%_{\text{rms}}$ ($\mu$m) | 0.308 | 0.137 | 0.152 |

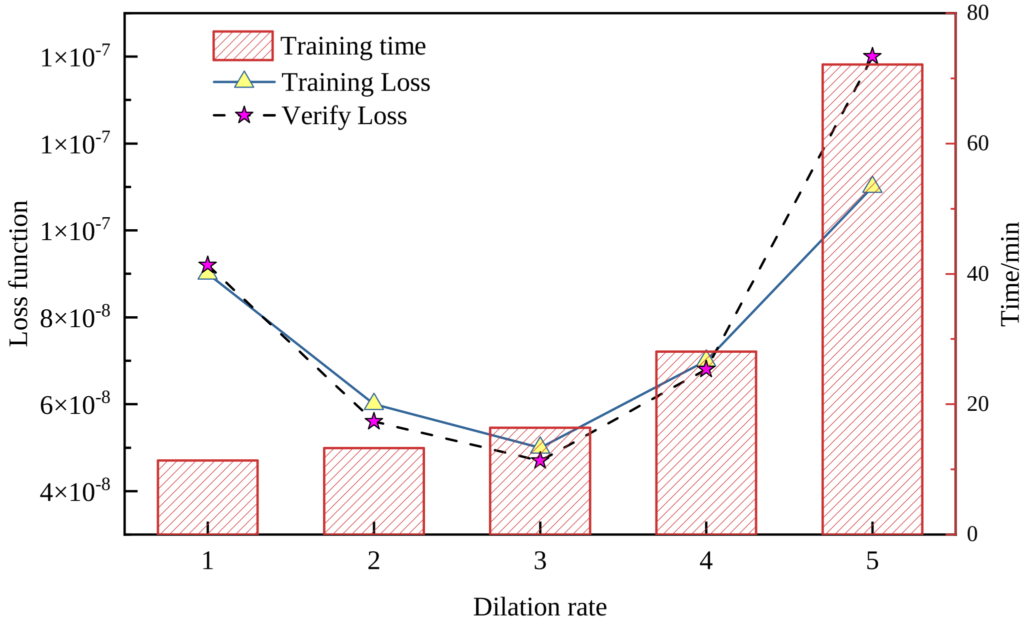


Figure 3. Comparison of different expansion rates

6. **Conclusions.** In this study, a high-speed CNC machining error prediction model (TPA-TCN-LSTM) was proposed and validated by detailed experiments. The adoption of TCN can effectively capture the long and short-term dependencies in time series data through causal convolution and extended convolution, which ensures the prediction performance of the model when dealing with complex nonlinear data. TPA is introduced to model the intrinsic relationships between different time series variables, thus further improving the model's ability to focus on important time steps and feature extraction. The traditional LSTM network is improved to simplify its structure and reduce the number of parameters, which accelerates the learning speed of large-scale and high-frequency data and ensures the accuracy and real-time performance of the model. The experimental results show that the TPA-TCN-LSTM model has significant advantages in five-axis CNC machining error prediction. Future research can further optimise the model structure, explore more nonlinear feature extraction methods and applications in different CNC machining scenarios to further enhance the versatility and practicality of this model.

## REFERENCES

[1] T. Schmitz, M. Davies, B. Dutterer, and J. Ziegert, "The application of high-speed CNC machining to prototype production," *International Journal of Machine Tools and Manufacture*, vol. 41, no. 8, pp. 1209-1228, 2001.
[2] S. Gordon and M. T. Hillery, "Development of a high-speed CNC cutting machine using linear motors," *Journal of Materials Processing Technology*, vol. 166, no. 3, pp. 321-329, 2005.
[3] M. Essid, B. Gassara, M. Baili, M. Hbaieb, G. Dessein, and W. Bouzid Saï, "Analytical modeling of the CNC machine axis motion in high-speed milling with local smoothing," *The International Journal of Advanced Manufacturing Technology*, vol. 105, pp. 457-470, 2019.
[4] Y. Dai, X. Tao, Z. Li, S. Zhan, Y. Li, and Y. Gao, "A review of key technologies for high-speed motorized spindles of CNC machine tools," *Machines*, vol. 10, no. 2, 145, 2022.
[5] D. Lyu, Q. Liu, H. Liu, and W. Zhao, "Dynamic error of CNC machine tools: a state-of-the-art review," *The International Journal of Advanced Manufacturing Technology*, vol. 106, pp. 1869-1891, 2020.
[6] S. Raghavendra, P. S. Sathyanarayana, T. Vs, and M. Kn, "High speed machining of titanium Ti 6Al4V alloy components: study and optimisation of cutting parameters using RSM," *Advances in Materials and Processing Technologies*, vol. 8, no. 1, pp. 277-290, 2022.

[7] J. Bi, X. Zhang, H. Yuan, J. Zhang, and M. Zhou, "A hybrid prediction method for realistic network traffic with temporal convolutional network and LSTM," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 1869-1879, 2021.
[8] A. Mezina, R. Burget, and C. M. Travieso-González, "Network anomaly detection with temporal convolutional network and U-Net model," *IEEE Access*, vol. 9, pp. 143608-143622, 2021.
[9] S. Li, Y. A. Farha, Y. Liu, M.-M. Cheng, and J. Gall, "Ms-tcn++: Multi-stage temporal convolutional network for action segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 6, pp. 6647-6658, 2020.
[10] H. Hu, Q. Wang, M. Cheng, and Z. Gao, "Trajectory prediction neural network and model interpretation based on temporal pattern attention," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 3, pp. 2746-2759, 2022.
[11] W. Chen, L. Cheng, Z. Chang, B. Wen, and P. Li, "Wind turbine blade icing detection using a novel bidirectional gated recurrent unit with temporal pattern attention and improved coot optimization algorithm," *Measurement Science and Technology*, vol. 34, no. 1, 014004, 2022.
[12] R. Ramesh, K. Ravi Kumar, and G. Anil, "Automated intelligent manufacturing system for surface finish control in CNC milling using support vector machines," *The International Journal of Advanced Manufacturing Technology*, vol. 42, pp. 1103-1117, 2009.
[13] N. Radhika, S. B. Senapathi, R. Subramaniam, R. Subramany, and K. Vishnu, "Pattern recognition based surface roughness prediction in turning hybrid metal matrix composite using random forest algorithm," *Industrial Lubrication and Tribology*, vol. 65, no. 5, pp. 311-319, 2013.
[14] T. Saric, G. Simunovic, K. Simunovic, and I. Svalina, "Estimation of machining time for CNC manufacturing using neural computing," *International Journal of Simulation Modelling*, vol. 15, no. 4, pp. 663-675, 2016.
[15] Y.-C. Liu, K.-Y. Li, and Y.-C. Tsai, "Spindle thermal error prediction based on LSTM deep learning for a CNC machine tool," *Applied Sciences*, vol. 11, no. 12, 5444, 2021.
[16] B. Bhandari and G. Park, "Non-contact surface roughness evaluation of milling surface using CNN-deep learning models," *International Journal of Computer Integrated Manufacturing*, vol. 37, no. 4, pp. 423-437, 2024.
[17] J. Sun, Z. Liu, C. Qiu, Z. Jiang, H. Mao, L. He, and J. Tan, "An axial attention-BiLSTM-based method for predicting the migration of CNC machine tool spindle thermal error under varying working conditions," *The International Journal of Advanced Manufacturing Technology*, vol. 130, no. 3, pp. 1405-1419, 2024.
[18] H. Gai, X. Li, F. Jiao, X. Cheng, X. Yang, and G. Zheng, "Application of a new model reference adaptive control based on PID control in CNC machine tools," *Machines*, vol. 9, no. 11, 274, 2021.
[19] X. Dong, Z. Jian-qu, and W. Feng, "Fuzzy PID control to feed servo system of CNC machine tool," *Procedia Engineering*, vol. 29, pp. 2853-2858, 2012.
[20] W. Lei and Y. Hsu, "Accuracy test of five-axis CNC machine tool with 3D probe–ball. Part I: design and modeling," *International Journal of Machine Tools and Manufacture*, vol. 42, no. 10, pp. 1153-1162, 2002.
[21] R.-S. Lee and C.-H. She, "Developing a postprocessor for three types of five-axis machine tools," *The International Journal of Advanced Manufacturing Technology*, vol. 13, pp. 658-665, 1997.
[22] A. Lasemi, D. Xue, and P. Gu, "Recent development in CNC machining of freeform surfaces: A state-of-the-art review," *Computer-Aided Design*, vol. 42, no. 7, pp. 641-654, 2010.
[23] B. Sencer, Y. Altintas, and E. Croft, "Feed optimization for five-axis CNC machine tools with drive constraints," *International Journal of Machine Tools and Manufacture*, vol. 48, no. 7-8, pp. 733-745, 2008.
[24] V. Kvrgic, Z. Dimic, V. Cvijanovic, J. Vidakovic, and N. Kablar, "A control algorithm for improving the accuracy of five-axis machine tools," *International Journal of Production Research*, vol. 52, no. 10, pp. 2983-2998, 2014.
[25] W. Lei and Y. Hsu, "Accuracy enhancement of five-axis CNC machines through real-time error compensation," *International journal of machine tools and manufacture*, vol. 43, no. 9, pp. 871-877, 2003.
[26] R.-S. Lin and Y. Koren, "Ruled surface machining on five-axis CNC machine tools," *Journal of Manufacturing Processes*, vol. 2, no. 1, pp. 25-35, 2000.
[27] R. Wan, S. Mei, J. Wang, M. Liu, and F. Yang, "Multivariate temporal convolutional network: A deep neural networks approach for multivariate time series forecasting," *Electronics*, vol. 8, no. 8, 876, 2019.

[28] J. Yan, L. Mu, L. Wang, R. Ranjan, and A. Y. Zomaya, "Temporal convolutional networks for the advance prediction of ENSO," *Scientific Reports*, vol. 10, no. 1, 8055, 2020.