A Neural Network Based Approach To Determine Chatter Stability In Milling For A Variable Pitch Cutter

Pritam Ghoshal
pritam20@iitk.ac.in

Department of Mechanical Engineering, Indian Institute of Technology Kanpur, Kalyanpur, Uttar Pradesh,208016, India

Abstract

The aim of this paper is to investigate a artificial neural network based approach to determine the chatter stability limit of a milling process with a variable pitch cutter. The variable pitch cutter provides an advantage over the uniform pitch cutter as it shows increased stability behaviour in a small range of spindle speed and depth of cut. The training data for the neural network is obtained by time domain simulations for a given combination of cutting parameters like cutting coefficients, feed, machine dynamic parameters like stiffness and damping. This data is fed to the multi-layer perceptron neural network to solve a classification problem using the scaled conjugate gradient method. The network performance is compared for several network architectures and the best combination is chosen. Once trained, the neural network is able to predict the stability of any given cutting condition with an accuracy of 99%.

Keywords: Neural Network, Artificial Intelligence, Milling, Variable Pitch, Chatter Stability, Time Domain Simulation

1. Introduction

Machining processes like milling have wide scale applications in various industries. The process has undergone several advancements over the past few decades. However, the central problem with milling, as is with any machining operation is the phenomenon of chatter. Chatter is nothing but the unwanted vibrations of the cutting tool which lead to poor surface finish and decreased productivity.

One of the pioneering works in determining chatter stability of milling was done by Altintas and Budak[1] who developed a general formulation by considering the cutter and work-piece as a multi-degree of freedom system. The dynamics of this system was modelled using delay differential equations. As a further continuation of their work, they developed simple analytical models for face milling and peripheral milling [2]. Altintas [3] presented the frequency and discrete time domain chatter stability laws.

A methodology for in-process predictions of work-piece dynamics based on a finite element model was proposed by Altintas et al. [4]. Zhongqun and Qiang [5] analyzed the chatter stability of end milling in the time domain. Numerical methods were used to solve the differential equations and it was shown that the time domain simulation results agreed closely with that obtained from the analytical model. Graham and Mehrpouya [6] investigated the effect of changing natural frequency, stiffness and cutting coefficients on chatter stability. A robust chatter stability model based on the analytical stability model had been developed. The model was found to agree closely with experimental data.

Milling with variable pitch and variable helix tools have been analyzed by Sims et al. [7].Comak and Budak [8] analyzed the dynamic stability of a variable pitch and helix cutter both in the frequency domain as well as using semi-discretisation method using multiple delays. Several other authors have worked on establishing chatter stability for irregular cutting tools namely Yusoff et al. [9],Huang et al. [10].

Artificial neural networks have been previously used to predict chatter stability. Cherukuri et.al [11] applied a neural network approach to chatter stability in turning. The analytical stability limit was used to train a neural network which then predicted whether a cutting condition is stable or unstable. Lamraoui et al. [12] used artificial intelligence for chatter prediction in milling using radial basis function and multi layer perceptrons. Deng et al. [13] described a reliability analysis of a milling process with uncertainties using neural networks. Tran et al. [14] proposed a method of chatter detection in milling based on the scalogram of the continuous wavelet transformation and the deep convolutional neural network.

All previous studies of stability of milling process using artificial neural networks(ANN) have been limited to milling cutters where the teeth are equally spaced. However, cutters with non-proportional spacing have been shown to have increased stability behaviour in a certain regime due to interruption of regeneration of surface waviness. Such type of cutters are especially used where changing the spindle speed is inconvenient like in transfer lines or in applications where a high spindle speed will lead to tool wear and machinability issues. However, not all variable tooth spacing are accompanied by an increase in stability. The region which shows an increased stability behaviour depends on the system dynamics, teeth spacing and spindle speed.This paper aims to determine the chatter stability limit of a milling process for a variable pitch cutter using ANN which may then be used to determine the stability for any given cutting condition. Several neural network architecture has been compared based on the number of hidden layers and neurons used to determine the stability boundary.

2. Background

Consider a milling cutter having radius r and N_t teeth, being used in a milling process with depth of cut a mm and linear feed f mm/min. The cutter is rotating with a speed of Ω rpm. The feed per tooth is hence given by,

$$f_t = \frac{f}{\Omega N_t} \tag{1}$$

Let at any instant, the cutter has rotated by an angle ϕ. The cutter is said to be cutting, only when it is in contact with the work-piece. This happens when ϕ is between ϕ_s and ϕ_e,the start and exit angle which are defined by the cutter geometry and type of milling, that is up milling or down milling. More details about ϕ_s and ϕ_e can be found in [15].

Let k_n and k_t be the cutting force coefficients in the normal and tangential direction and b be the width of cut. Then the cutting force is given in these directions is given by,

$$F_n = k_n bh \qquad F_t = k_t bh \tag{2}$$

where h is the instantaneous chip thickness.The cutting forces in the x and y directions can hence be found as,

$$F_x = F_t \cos\phi + F_n \sin\phi \tag{3}$$

and

$$F_y = F_t \sin\phi - F_n \cos\phi \tag{4}$$

A detailed description of the time domain simulation of a milling problem with a constant pitch cutter is given below which is based on the formulation as given by Smith and Tlusty[16].This formulation will later be modified for a variable pitch cutter.

The simulation has four basic steps which are

- Determining the instantaneous chip thickness using the vibrations of the current and previous tooth at a given engagement angle.
- Calculating cutting force using cutting force coefficients
- Calculating displacements using force
- Increasing engagement angle and repeating all the steps

Due to vibrations of the tool(assuming the work-piece to be rigid), wavy profiles are created on the original surface by the first tooth in contact. When the next tooth comes in contact with the work-piece, it does not encounter the original surface, but the wavy surface left behind by the previous tooth. Thus the instantaneous chip thickness depends not only upon the the current vibration, but also on the surface developed by the previous tooth. This variable chip thickness that governs the cutting forces affects the subsequent vibrations as well. Thus a feedback mechanism is produced.

The chip thickness is measured along the surface normal. The normal direction changes as a function of the engagement angle ϕ. Projecting the vibrations in the x and y directions, in the direction of the normal,the following equation is obtained.

$$n = x \sin\phi - y \cos\phi \tag{5}$$

where $\phi = 6\Omega t$ and t is in seconds.The instantaneous chip thickness can thus be written as

$$h(t) = f_t \sin\phi + n(t-\tau) - n(t) \tag{6}$$

where f_t is the feed per tooth, $f_t \sin\phi$ is the mean chip thickness and τ is the tooth period in seconds and is given by

$$\tau = \frac{60}{\Omega N_t} \tag{7}$$

Here $n(t)$ is the current normal direction vibration and $n(t-\tau)$ is the normal direction vibration when the previous tooth was in cut. Care must be taken to calculate $n(t-\tau)$ by considering whether the previous tooth was indeed cutting or not. For a more detailed description on how to calculate $n(t-\tau)$,one must refer to [15].

Once the chip thickness is calculated, the cutting force in the x and y direction can be calculated using Eqs.(3) and (4). Considering a single degree of freedom in the x and y directions, the equations of motion can be written as

$$m_x\ddot{x} + c_x\dot{x} + k_x x = F_x \tag{8}$$

and

$$m_y\ddot{y} + c_y\dot{y} + k_y y = F_y \tag{9}$$

Here m_i, c_i, k_i are the mass, damping and stiffness in the i direction. If there are more than one degrees of freedom in each direction,Eqs.(8) and (9) will become a system of equations represented in matrix form.

Assuming the initial displacement and velocity in both the x and y directions to be zero, the equations of motions can be solved for $\ddot{x}$ and $\ddot{y}$ iteratively, which can then be used to find the displacement and velocity for the subsequent time steps. One such iterative method is the Euler forward integration method, the algorithm for which states that,

$$\dot{x} = \dot{x} + \ddot{x}(dt) \qquad \dot{y} = \dot{y} + \ddot{y}(dt) \tag{10}$$

and

$$x = x + \dot{x}(dt) \qquad y = y + \dot{y}(dt) \tag{11}$$

where dt is the small incremental time step. This completes the derivation of the time domain simulation algorithm for a constant pitch cutter.

In case of cutters with uniform teeth spacing, there is a periodic delay between two engagements. However, when the spacing is varied, this periodicity is interrupted.Hence the first modification from the previous derivation is that τ is no longer constant. The second modification will be for the feed per tooth, which will now change with every tooth. Feed per tooth for each tooth each is given by

$$f_t = \bar{f}_t \theta \frac{N_t}{360} \tag{12}$$

where $\bar{f}_t$ is the mean feed per tooth and θ is the angle between the current and previous teeth and is different for each tooth. Rest of the derivation remains the same. For a more detailed derivation, one must again refer to [15].

3. Artificial Neural Networks

There are two types of machine learning algorithms, supervised and unsupervised. Supervised learning algorithms are trained using labelled data which means that the output for a given input is known. On the other hand if the output for each corresponding input is not known or in other words the data is unlabelled, then the learning algorithm is unsupervised. Two classes of problems arise in machine learning-classification and regression. If the output data is to used to categorise the input into different classes, it is called a classification algorithm. In this case, the output is usually discrete. If on the other hand, the output is continuous and real valued, then it becomes a regression problem. For the problem at hand, we aim to classify a given cutting condition specified by the depth of cut and spindle speed as stable or unstable. Hence the problem is of the classification type.

The ANN tries to simulate the functioning of the brain using a network of neurons and synapses. The neurons represented by circles are arranged in a column called layers. The leftmost layers is called the input layer. The rightmost layer is called the output layer. Any layer in between is called the hidden layer. There may be one or more hidden layers. If there is only one hidden layer, it is called a shallow neural network. If there are more than one hidden layers, it is called a deep neural network.There are four basis components in a neural network-input to the neurons denoted by x_i,$i = 1, 2, ...n$,where n is the number of neurons in a layer, weights assigned to each synapse w_i, a bias b and an activation function f.

The input h_i to a neuron of the next layer which happens to be the output of all the neurons in the current layer is given by $f(z_i)$, where

$$z_i = b + \sum_{i=1}^{n} w_i x_i \tag{13}$$

The activation function is usually a nonlinear function like logarithmic or hyperbolic tangent sigmoid function. This process is continued till the output layer. There can be more than one neurons in the output layer. If there is only one neuron, it is called a binary classification problem (although two neurons can also be used), whereas for more than one neurons, it is called a multi-class classification problem. For the problem at hand, the objective is to classify a cutting condition as stable or unstable. So the classification is binary.

The neural network is trained using a given data set. For the first pass, the initial weights are assumed to be some random quantities. Using these random weights, the output is calculated. The error between the actual output and the predicted output is calculated using a loss function which is a function of the weights and biases. The goal is to minimise this loss function by adjusting the weights appropriately. This can be done using several methods, but the most common method is by using the steepest descent method. The updated weights are again used to predict the output. This process is continued until the error is less than a given tolerance level. In this paper, the scaled conjugate gradient method is used to train the network. Once trained, the network is validated and tested using an independent set of data points.

4. ANN Model

The training data is obtained by time domain simulations for several combinations of spindle speed and depth of cut using the algorithm described in Section 2. Here it is considered that there are two modes each in the x and y directions.Data for both the modes is assumed to be as follows: $k_1 = 2 \times 10^7$N/m, $f_{n1} = 800$Hz, $\zeta_1 = 0.05$,$k_2 = 1.5 \times 10^7$N/m, $f_{n1} = 1000$Hz, $\zeta_1 = 0.03$, where f is the natural frequency and ζ is the damping ratio. The tangential and radial cutting coefficients are 520N/mm^2 and 300N/mm^2. The cutter has 4 teeth which are placed at $95, 180, 275$ degrees from the first tooth. The radius of the cutter is such that the starting and exit angles are 0 and 66.4 degrees.

The builtin neural net toolbox of MATLAB is used to construct the neural net architecture. There are two input features, the spindle speed and the limiting depth of cut. So the input layer has two nodes. A total of 19200 data points is used to train and validate the ANN.The blue dots signify stable cutting condition while the red dots represent unstable cutting conditions. The spindle speed ranges from 2000rpm to 6000rpm and the depth of cut ranges from 1mm to 24mm. Since the scale of both the features is different, first a feature scaling is performed using the min-max method to bring the input features within the range $[-1, 1]$. For the training data, the output is either 0 or 1 for unstable and stable points respectively. The normalised data is then randomly divided into training set,validation set and the testing set in the ratio $70 : 15 : 15$. So the training set has 13440 data points, and the validation and test set contains 2880 data points each.The training,validation and test data is shown in Fig.1.

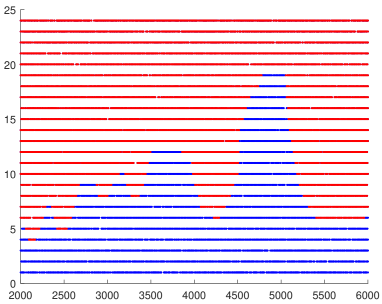
The ANN consists of two hidden layers with seven neurons each as shown in Fig.2. The input layer consists of two neurons corresponding to the limiting depth of cut and spindle speed. The output layer consists of one neuron, the output of which is the probability $p(x)$ such that the conditions as defined by the input is stable. The probability is defined as

$$p(x) = \{Y = 1|(\Omega, b_{lim})\} \tag{14}$$

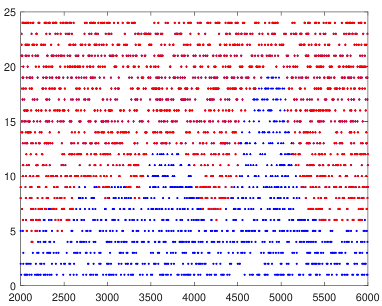
If $p(x) \geq 0.5$, then the cutting conditions is considered to be stable, otherwise unstable.The activation function for the input and hidden layers has been taken as the hyperbolic tangent sigmoid function given by,

$$y = \text{tansig}(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{15}$$

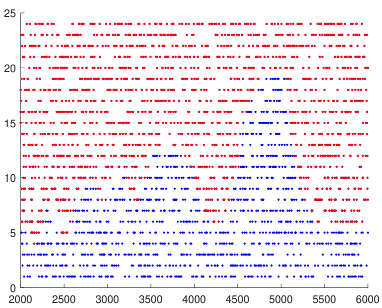
while that for the output layer is taken as a purely linear function. The bias has been taken as one for both the input and hidden layers.



(a) Training Data Set



(b) Validation Data Set



(c) Test Data Set

Figure 1: Data Set For The ANN

The cross-entropy error function is used in this paper. However there are other error functions available which might also be used. There are four stopping conditions for training. The first condition is when the number of epochs exceed 500000. The second condition is when the gradient(derivative with respect to the weights) reaches 10^{-6}. The third is when error as evaluated by the error function is less than than 0.001. The final condition is when the number of validation checks or the number of successive iterations for which the validation performance has failed to increase exceeds 6.

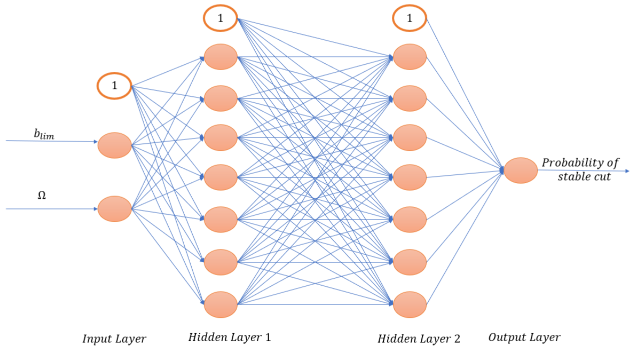


Figure 2: The ANN architecture with 2 hidden layers and 7 node each

5. Results

The confusion matrix for the training,validation and test data is shown in Fig.3. The target class represents the actual value while the output class represents the predicted value. 0 and 1 represent the chatter and stable case respectively. As can be seen in Fig.3c, the ANN predicts 2863 (1787 true positives and 1076 true negatives) out of the 2880 test data points accurately with an accuracy of 99.4% which is very accurate. Overall the ANN predicts the data correctly with an accuracy of 99.7% as can be seen in Fig.3d.

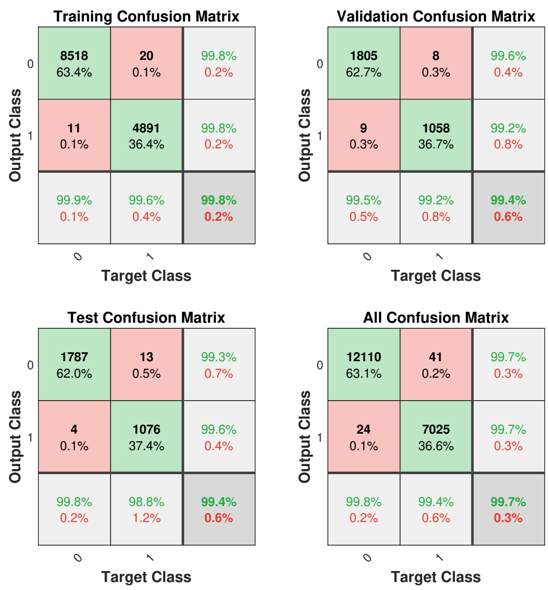


Figure 3: Confusion Matrix

The network performance as evaluated by the cross-entropy function versus the epoch for the training,validation and test data is shown in Fig.4. As can be seen, the error in the predicted output for the training data reduces with number of epochs and ultimately goes below 0.01. For the validation and test data, the curve starts rising upwards after a certain epoch which suggests over-fitting. Hence instead of running the algorithm till the error reaches below a certain value(which may lead to overfitting), the algorithm is run upto a 500000 epochs. At this point the error in prediction for the training data is below 0.01 and in the testing and validation data is below 0.1 which is within acceptable range.

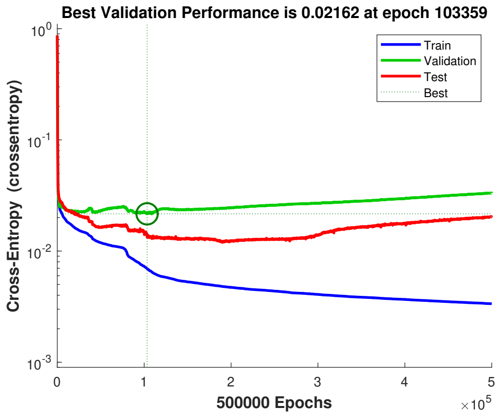


Figure 4: ANN Performance Plot

The error histogram for the training,validation and testing data is shown in Fig.5. The x axis represents the error in the prediction while the y axis represents the number of samples which are in the same error range. The error ranges from −0.95 to 0.95 which has been divided into 20 ranges or bins. The width of each bin is thus $(0.95 + 0.95)/20 = 0.095$. It can be seen that most of the sample data lies within a small range of error from $(0.0499-0.0950/2) = 0.0024$ to $(0.0499+0.0950/2) = 0.0974$.Almost 18000 of the total 19200 points have been predicted with an error of less than 0.05 which suggests that the classification problem has been solved very accurately.

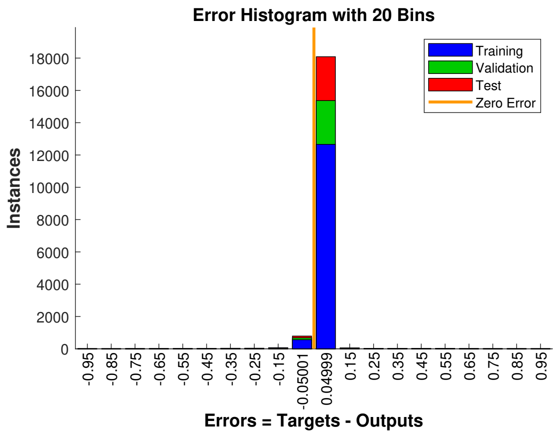


Figure 5: Error Histogram

The variation of the gradient with epochs is shown in Fig.6. As can be seen the gradient has a decreasing trend on average with number of iterations which means the error function is approaching its minimum. The gradient after 500000 iterations was 0.000249 which signifies that the algorithm has stopped very close to the minimum.

The receiver operating characteristic(ROC) curve is shown in Fig.7. The ROC is a measure of the performance of a binary classifier and is created by plotting the true positive rate versus the false positive rate.The more each curves hugs the top and left edge of the graph, better the classifier.As can be seen, the curves for all three data sets-training,validation,testing are almost along the edges which signify a very good prediction of stability. Hence from all these measures it can be concluded that the ANN is able to predict the stability of a given cutting condition very accurately.

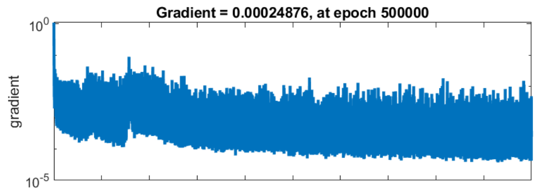


Figure 6: Variation of gradient with epochs

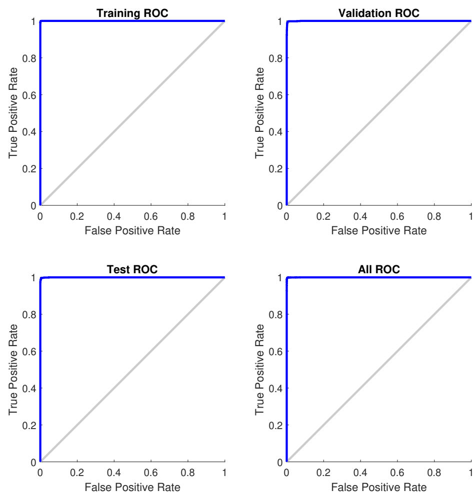


Figure 7: Receiver Operating Characteristic Curves

The decision boundary as obtained by the ANN is shown in Fig.8. It is a pretty accurate representation of the stability boundary as obtained from the time domain simulation. It is also evident from Fig.8, that there is a region around 4000rpm to 4600rpm where the stability has increased as compared to the uniform pitch cutter, which is in line with the time domain simulation predictions.To give some perspective, the stability limit at 4000rpm for a uniform pitch cutter is around 4.6mm, for that of a variable pitch cutter in the configuration considered in this paper is 10mm while that predicted by ANN is around 9.8mm. The stability limit at 4600rpm for a uniform pitch cutter is around 7mm, for that of a variable pitch cutter is 16mm while that predicted by ANN is around 16.2mm.

The sensitivity of the decision boundary on the number of hidden layers and neurons has been investigated. The decision boundary has been plotted for several combinations of layers and neurons. Some such combinations are shown in Table 1.

The first column represents the number of hidden layers and number of neurons in each layer. The second column shows the decision boundary as obtained from the ANN and the third column shows the original decision boundary obtained from time domain simulation. As can be seen, for less number of neurons, although the increase in stability around 4000rpm is predicted, the ANN misses out on the smaller lobes. However, on increasing the number of neurons, the ANN is able to predict the stability lobes very closely.

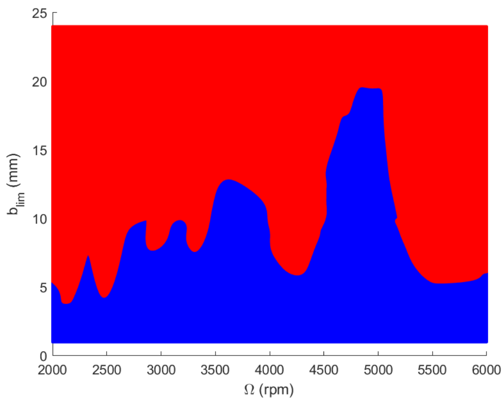


Figure 8: Stability Limit As Predicted By ANN

6. Conclusions

This paper aims to apply a neural network based approach to milling chatter stability analysis for a cutter with variable pitch. The ANN predicts that for the cutting parameters and machine dynamics considered in this paper, there is a small regime around 4000rpm to 4600rpm where improved stability behaviour can be obtained which confirms the predictions of the time domain simulations. For less number of neurons in the hidden layer, the ANN is not able to predict stability near the smaller lobes very accurately. On increasing the number of nodes, even the smaller lobes are taken into account. The ANN with two hidden layers and seven neurons each is able to predict the stability with an accuracy of about 99%. Once trained, this network can be used to predict the stability for any given cutting condition for the parameters considered in this paper without performing a complex time domain simulation and observing the behaviour of the displacements and forces.

7. Future Scope

The data in this paper has been generated using the time domain simulations. An improvement over this would to generate the data from actual experiments itself for a wide combination of cutting coefficients, material, machine dynamics. The data thus obtained, could be then used to train a much larger neural network. The input layer in this case, would not only consist of the spindle speed and depth of cut, but also several other parameters like feed, cutting coefficients, machine stiffness and damping, number of modes etc. The trained network thus obtained can then be used to predict chatter stability in milling on any machine for any material work-piece combination.

Improved stability behaviour may also be obtained using a helical tooth that has a varying angle of helix along the axis or a cutter that has a different angle of helix for each tooth. Such cases may also be investigated in future using ANN.

References

[1] E. Budak, Y. Altintas, Analytical prediction of chatter stability in milling—part i: general formulation, ASME Digital Collections (1998).

[2] E. Budak, Y. Altintas, Analytical prediction of chatter stability in milling—part ii: application of the general formulation to common milling systems, ASME Digital Collections (1998).

[3] Y. Altintas, G. Stepan, D. Merdol, Z. Dombovari, Chatter stability of milling in frequency and discrete time domain, CIRP Journal of Manufacturing Science and Technology 1 (1) (2008) 35–44.

[4] E. Budak, L. T. Tunç, S. Alan, H. N. Özgüven, Prediction of workpiece dynamics and its effects on chatter stability in milling, CIRP annals 61 (1) (2012) 339–342.

[5] L. Zhongqun, L. Qiang, Solution and analysis of chatter stability for end milling in the time-domain, Chinese Journal of Aeronautics 21 (2) (2008) 169–178.

[6] E. Graham, M. Mehrpouya, S. Park, Robust prediction of chatter stability in milling based on the analytical chatter stability, Journal of Manufacturing Processes 15 (4) (2013) 508–517.

[7] N. Sims, B. Mann, S. Huyanan, Analytical prediction of chatter stability for variable pitch and variable helix milling tools, Journal of Sound and Vibration 317 (3) (2008) 664–686.

[8] A. Comak, E. Budak, Modeling dynamics and stability of variable pitch and helix milling tools for development of a design method to maximize chatter stability, Precision Engineering 47 (2017) 459–468.

[9] Yusoff, A. Razlan, M. H. Mansor, H. C. Hussain, M. A. Z. Abidin, Analytical chatter stability prediction for irregular types of milling tools, Materials Science Forum 773 (2014) 420–426.

[10] P. Huang, J. Li, J. Sun, M. Ge, Milling force vibration analysis in high-speed-milling titanium alloy using variable pitch angle mill, The International Journal of Advanced Manufacturing Technology 58 (1–4) (2012) 153–160.

[11] H. Cherukuri, E. Perez-Bernabeu, M. Selles, T. L. Schmitz, A neural network approach for chatter prediction in turning, Procedia Manufacturing 34 (2019) 885–892.

[12] M. Lamraoui, M. Barakat, M. Thomas, M. E. Badaoui, Chatter detection in milling machines by neural network classification and feature selection, Journal of Vibration and Control 21 (7) (2015) 1251–1266.

[13] C. Deng, J. Miao, Y. Ma, B. Wei, Y. Feng, Reliability analysis of chatter stability for milling process system with uncertainties based on neural network and fourth moment method, Journal of Production Research 58 (9) (2020) 2732–2750.

[14] M.-Q. Tran, M.-K. Liu, Q.-V. Tran, Milling chatter detection using scalogram and deep convolutional neural network, The International Journal of Advanced Manufacturing Technology 107 (3) (2020) 1505–1516.

[15] T. L. Schmitz, K. S. Smith, Machining dynamics, Springer, 2014.

[16] S. Smith, J. Tlusty, An overview of modeling and simulation of the milling process, Journal of Engineering for Industry 113 (2) (1991).

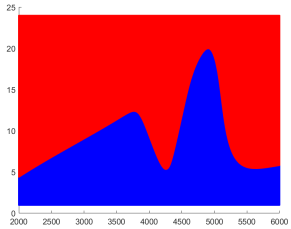
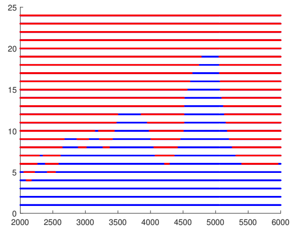
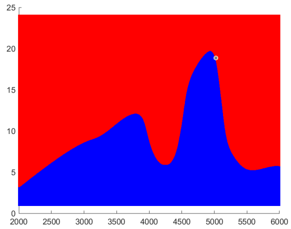
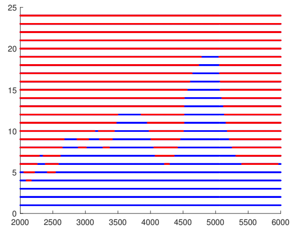
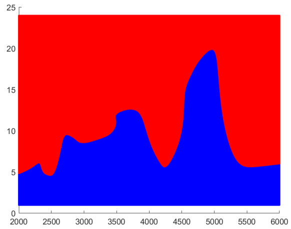
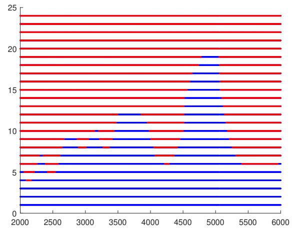
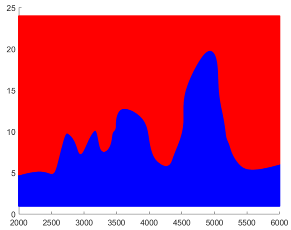
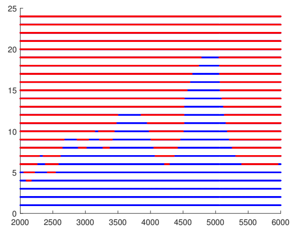
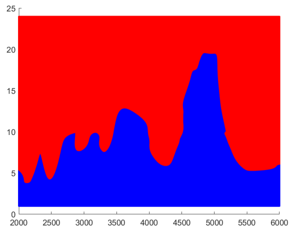
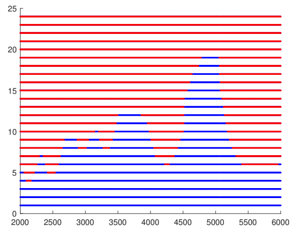
ANN Architecture	ANN Decision Boundary	Time Domain Simulation
2 layers 3 neurons		
2 layers 4 neurons		
2 layers 5 neurons		
2 layers 6 neurons		
2 layers 7 neurons		

Table 1: Comparison Of Sensitivity Of Decision Boundary To ANN Architecture