CHAPTER 6

CHATTER SUPPRESSION USING ARTIFICIAL NEURAL NETWORKS

The central idea of this chapter is to present methodology of active chatter control using ANN, which is the adopted chatter control strategy for boring. Having discussed the concept of active damping using LR and SVR, the chapter proceeds to implement the active chatter suppression and validation of the results (effectiveness) in boring operations using MR damper. The summary is presented at the end of the chapter.

6.1 ARTIFICIAL NEURAL NETWORKS

Artificial Neural Networks (ANN) is modeled on biological neurons and nervous systems. They have the ability to learn, and possess the processing elements known as neurons which perform their operations in parallel. ANN’s are characterized by their topology, weight vector and activation functions. They have three layers namely an input layer, which receives signals from the external world, a hidden layer, which does the processing of the signals and an output layer, which gives the result back to the external world.

6.2 MULTI-LAYER PERCEPTRON (MLP)

This is an important class of neural networks, namely the feed forward networks. Typically, the network consists of a set of input parameters
that constitute the input layer, one or more hidden layers of computation nodes and an output layer of computation nodes. The input signal propagates through the network in a forward direction on a layer-by-layer basis.

MLPs have been applied to solve some difficult and diverse problems by training them in a supervised manner with a highly popular algorithm known as the error back-propagation algorithm. Each neuron in the hidden and output layer consists of an activation function, which is generally a non-linear function like the logistic function, which is given by

\[ f(x) = \frac{1}{1 + e^{-x}}, \tag{6.1} \]

Where \( f(x) \) is differentiable and

\[ x = \sum_{i=1}^{I} W_{ij} \xi_i + \theta_j, \tag{6.2} \]

Where, \( W_{ij} \) is the weight vector connecting the \( i^{th} \) neuron of the input layer to the \( j^{th} \) neuron of the hidden layer, \( \xi_i \) is the input vector and \( \theta_j \) is the threshold of the \( j^{th} \) neuron of the hidden layer. Similarly, \( W_{ij} \) is the weight vector connecting \( j^{th} \) neuron of the hidden layer with the \( k^{th} \) neuron of the output layer. \( i \) – represents the input layer, \( j \) represents the hidden layer and \( k \) represents the output layer. The weights that are important in predicting the process are unknown. The weights of the network to be trained are initialized to small random values. The choice of value selected obviously affects the rate of convergence. The weights are updated through an iterative learning process known as ‘Error Back Propagation (BP) algorithm’. Error BackPropagation process consists of two passes through the different layers of the network; a forward pass in which input patterns are presented to the input layer of the network and its effect propagates through the network layer by layer. Finally, a set of outputs is produced as the actual response of the
network. During the forward pass the synaptic weights in the networks are all fixed. The error value is then calculated, which is the mean square error (MSE) given by

$$E_{tot} = \frac{1}{n} \sum_{n=1}^{n} E_n$$  \hspace{1cm} (6.3)

Where,

$$E_n = \frac{1}{2} \sum_{k=1}^{m} (\vec{z}_k - \vec{O}_k)^2$$  

Where, \(m\) is the number of neurons in the output layer,

\(\vec{z}_k\) is the \(k^{th}\) component of the desired or target output vector and

\(\vec{O}_k\) is the \(k^{th}\) component of the output vector.

The weights in the links connecting the output and the hidden layer \(W_{jk}\) are modified as follows:

$$\Delta W_{jk} = \eta \left( -\frac{\partial E}{\partial W_{jk}} \right) = \eta \delta_j y_j,$$  \hspace{1cm} (6.4)

Where,

$$\delta_j = y_j (1 - y_j) \sum_{k=1}^{m} \delta_k W_{jk}.$$  

$$W_{ij}^{new} = W_{ij}^{old} + \Delta W_{ij}$$  \hspace{1cm} (6.5)

$$\delta_k = (\vec{z}_k - \vec{O}_k) \vec{O}_k (1 - \vec{O}_k)$$  \hspace{1cm} (6.6)
6.7 \hspace{1cm} \delta_j = y_j(1 - y_j)\sum_{k=1}^{m} \delta_k W_{jk} \quad \text{for hidden neurons.}

The training process is carried out until the total error reaches an acceptable level (threshold). If \( E_{\text{tot}} < E_{\text{min}} \) the training process is stopped and the final weights are stored, which is used in the testing phase for determining the performance of the developed network. The training mode adopted was ‘batch mode’, where weight updating was performed after the presentation of all training examples that constitute an epoch (Wasserman 1993, Smith 1993, Sergios Theodoridis et al 2009, Siegelmann et al 1994, Ripley et al 1996, Lawrence and Jeanette 1994, Hertz J. Palmer et al 1990, Haykin 1999). Artificial neural networks can be used for classification problems, regression problem and optimization problems. It is widely used for classification problems (Egmont-Petersen et al 2002, Duda et al 2001, Cybenko 1989). ANN is also used for regression purpose as found in literature (Bishop 1995, Bhadeshia 1999, Wang 2007, Zhang 2005, Kumar 2005, Hill et al 1996, Altun et al 2007).

6.3 RESULTS AND DISCUSSION

The experimental study was carried out for various cutting conditions as shown in Table 3.1. In regression model, current is the dependant variable. Mean, min (minimum value) and max (maximum value) are independent variables representing the vibration signals. The feature extraction and regression analysis using artificial neural networks were performed. For each cutting condition a regression model was built to quantify the amount of current required to provide enough damping by MR damper in order to suppress tool chatter. It is desirable to have one regression model for all possible cutting conditions; however, in practice, this leads to an unacceptable error in the predicted value. In order to have high model accuracy, a separate model was built for each cutting condition. The problem
was modeled using artificial neural networks and the architecture of the
developed ANN regression model is shown in Figure 6.1. The regression
model generated by ANN is presented in Figure 6.2 (for 47 m/min), Figure
6.3 (for 72 m/min), Figure 6.4 (for 119 m/min) and Figure 6.5 (for all speeds).
For building the ANN regression model a freely downloadable tool called
‘WEKA’ was used. The output of the ANN regression model looks like the
one shown in Figures 6.2 to 6.5. The method of interpreting the result is as
follows:

The node 0, node1 and node 2 are shown in Figure 6.1. From
Figure 6.2, node1 value is obtained as below:

\[
node_1 = (-0.1848 \times \text{max}) + (-0.3955 \times \text{min}) + (-1.4662 \times \text{mean}) - 0.1941
\]

(6.8)

The coefficient values of corresponding independent variables
are multiplied with variables and added together with the intercept value to
get expression for node1. Similarly, one can have expressions for node2 as
shown below.

\[
node_2 = (2.3056 \times \text{max}) + (4.1068 \times \text{min}) + (4.5113 \times \text{mean}) - 8.1781
\]

(6.9)

Then, the expression for current is obtained from the expressions of
node1 and node 2.

\[
current = (-1.0641 \times node_2) + (-3.0476 \times node_1) + 2.1667
\]

(6.10)
Figure 6.1 Architecture of ANN Model

Linear Node 0
Inputs  Weights
Threshold  2.1667018166714756
Node 1  -3.047675560651728
Node 2  -1.0641969544325145

Sigmoid Node 1
Inputs  Weights
Threshold  -0.19412763933809113
Attrib Mean  -1.46629225382033383
Attrib Min  -0.3955199883470132
Attrib Max  -0.18480647496673205

Sigmoid Node 2
Inputs  Weights
Threshold  8.178148334719156
Attrib Mean  4.511390754861274
Attrib Min  4.106892749188562
Attrib Max  2.305667214619961

Figure 6.2 ANN Model for 47 m/min
### Linear Node 0
- Inputs: Weights
  - Threshold: 2.8823894903900475
  - Node 1: -3.4176982332588177
  - Node 2: -1.6494170522545368

### Sigmoid Node 1
- Inputs: Weights
  - Threshold: 0.0698271495542658
  - Attrib Mean: -2.0945641267856145
  - Attrib Min: 0.08290558484967162
  - Attrib Max: 0.02000892937673048

### Sigmoid Node 2
- Inputs: Weights
  - Threshold: 5.59782475993681
  - Attrib Mean: 9.078910175818253
  - Attrib Min: 0.10618243848293399
  - Attrib Max: -0.16744844462968722

---

**Figure 6.3** ANN Model for 72 m/min

### Linear Node 0
- Inputs: Weights
  - Threshold: 1.4288841935336085
  - Node 1: 1.41779532041843
  - Node 2: -3.8021356264122224

### Sigmoid Node 1
- Inputs: Weights
  - Threshold: -7.4311158035082245
  - Attrib Mean: -9.0249148038515
  - Attrib Min: -0.5732360536962807
  - Attrib Max: 0.1817677252132839

### Sigmoid Node 2
- Inputs: Weights
  - Threshold: -0.22937453441336275
  - Attrib Mean: -1.5227164407440195
  - Attrib Min: -0.05228763919785024
  - Attrib Max: -0.01647370067676503

---

**Figure 6.4** ANN Model for 119 m/min
Figure 6.5 ANN Model for all speeds (47 m/min, 72 m/min and 119 m/min)

The error estimates of the ANN regression model are presented in Figures 6.6 to 6.10. ‘CC’ represents correlation co-efficient, ‘MAE’ means mean absolute error, ‘RMS error’ stands for root mean square error, ‘RA error’ represents relative absolute error and ‘RRS error’ represents root relative squared error. The value of correlation coefficient is close to 1 for most of the models. If the ‘CC’ value is close to 1, it means that the regression model built has high resemblance with the training data points and hence, the prediction performance of these models will be good. To support this argument, wherever the CC (correlation coefficients) are close to 1, the error rates will be less. One can observe from Figure 6.10 that RRS error rate is the minimum in ANN regression model compared to linear regression model and SVR models that are presented in previous chapters. As a matter of fact, other error rates also follow the same trend.
Figure 6.6 Correlation coefficient Vs Speed

Figure 6.7 Mean absolute error Vs Speed
Figure 6.8 RMS error Vs Speed

Figure 6.9 RA error Vs Speed
Figure 6.10 RRS error vs Speed

6.4 IMPLEMENTATION

The regression model built using ANN was implemented in labVIEW software. Three separate models were implemented using ‘switch-case’ option where user needs to choose the speed to choose the corresponding regression equation. The input was taken from accelerometer through DAQ card as explained in chapter 3. The statistical measures namely, mean, minimum value, maximum value (in a given signal) were then computed and they act as input to the regression model. The ANN regression model gives the amount of current to be supplied to the MR damper. The current varies the damping co-efficient of the MR damper and the stiffness of the medium changes offering different damping to the vibration. While building the model, the range of vibration signals are mapped to the range of current for the MR damper. Due to dynamic damping the boring tool chatter is expected to become stable leading to reduced vibration. The developed
LabVIEW program for implementing ANN based regression controller is shown in Figure 6.11.

Figure 6.11 LabVIEW implementation of developed ANN regression controller

6.5 VALIDATION

The vibration signals of boring operation with MR damping and without damping was plotted for the purpose of comparison and shown in Figures 6.12 to 6.20. Referring to Figure 6.12 and Figure 6.13, the vibration amplitude without damper is much higher compared to the vibration signals with damper.

In Figures 6.12 to 6.20 the plots of vibration signals acquired is presented for visual comparison. The y-axis shows amplitude with respect to sample number. The amplitude of acceleration of vibration (tool chatter) was measured in ‘g’, a unit of acceleration. The x-axis shows time information. In analog domain the time is continuous and normally measured in ‘s’. Here, the
discrete samples are used through DAQ (data acquisition card) and hence the sample number along with sampling rate will represent the time axis.

**Figure 6.12** Vibration signal during machining speed at 47 m/min

**Figure 6.13** Vibration signal during machining speed at 47 m/min
Figure 6.14 Vibration signal during machining speed at 47 m/min

Figure 6.15 Vibration signal during machining speed at 72 m/min
Figure 6.16 Vibration signal during machining speed at 72 m/min

Figure 6.17 Vibration signal during machining speed at 72 m/min
Figure 6.18 Vibration signal during machining speed at 119 m/min

Figure 6.19 Vibration signal during machining speed at 119 m/min
Table 6.1 Effect of MR damping using ANN regression model

<table>
<thead>
<tr>
<th>Speed (rpm)</th>
<th>Speed (m/min)</th>
<th>RMS value of vibration (g)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Trial 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WOD</td>
</tr>
<tr>
<td>300</td>
<td>47</td>
<td>0.841</td>
</tr>
<tr>
<td>460</td>
<td>72</td>
<td>0.332</td>
</tr>
<tr>
<td>755</td>
<td>119</td>
<td>0.458</td>
</tr>
</tbody>
</table>

The effect of damping is presented in time domain plots from Figures 6.12 to 6.14 for 47 m/min. The similar plots are depicted from Figures 6.15 to 6.17 for 72 m/min and from Figure 6.18 to Figure 6.20 for 119 m/min respectively. Observing these plots one can see that there is significant reduction in boring tool chatter. At 47 m/min and 72 m/min the effect of damping is very high; it leads to effective dynamic damping. At 119 m/min the effect of damping is very low.
To further investigate the effect quantitatively, three representative signals of the same cutting conditions are taken without damper. At the same cutting condition, three more signals are taken with damper. To quantify the amount of vibration, RMS level is commonly used and same is calculated and presented in Figure 6.23. This exercise was repeated for 72 m/min and 119 m/min. The difference between RMS level of vibration signals in without damping condition and with damping condition is taken as a measure of effectiveness of the damping. The percentage of reduction in RMS level in Figure 6.23 was computed using the following formula:

\[
\% \text{ of Reduction} = \left( \frac{WOD - WD}{WOD} \right) \times 100
\]

Where,  
WOD – RMS value of vibration signals without damping (g)  
WD- RMS value of vibration signals with damping (g)

![Figure 6.21 Effect of MR damping using ANN regression controller](image-url)
Figure 6.22 Speed Vs reduction in RMS value

Figure 6.23 Speed Vs % of reduction in RMS value

The damping effect of MR damper using ANN regression model is presented in the form of a graph in Figure 6.21. Now it becomes evident that
the damping effect is close to 60% -80% of its initial vibration at lower speed and the effect falls sharply at higher speed (119 m/min). The effect of damping is less at high speed, because the vibration level itself is low compared to that of lower speed vibration level. As the vibration level is small at higher speeds, the ANN regression model will give an output corresponding to a lower current. The small current increases the stiffness of the MR damper only to a smaller extent, thus leading to a small reduction in boring tool chatter level. On the other hand, at lower speeds, the boring tool chatter is very high. If the vibration level is high the regression model will give high current output that increases the stiffness of the damper greatly. This gives a larger reduction in boring tool chatter at lower speeds.

6.6 SUMMARY

Self excited vibration or chatter is the most important type of vibration in machining process. Chatter occurs when the cutting forces are modulated by changes in the uncut chip thickness, which in turn results in greater variations in the uncut chip thickness of the next tooth (pass). It limits cutting depth (as a result, productivity), adversely affects surface finish and causes premature tool failure. This chapter has presented an ANN based regressive model based approach to reduce the tool chatter using magneto rheological fluid, whose properties are controlled through the magnetizing current. The regressive model relates the parameters of vibration to the current so as to reduce the boring tool chatter based on the present level of vibration. The results of ANN regression model was validated by computing reduction in RMS level of the vibration with MR damping. The practical results are in line with the results predicted by the mathematical model.