Proceedings of 3rd National Conference on Trends In Mechanical Engineering, Dec 29-30, 2006

Detection of Tool Breakage in Milling Operations using Neural Networks (RCE)

S. MADHAVA REDDY[1] and Dr. A. CHENNAKESAVA REDDY[2]

Associate Professor
Department of Mechanical Engineering
MGIT, Gandipet, Hyderabad –75.
E-mail: Deepak_madhav2003@yahoo.co.in

Professor
Department of Mechanical Engineering
JNTU College of Engineering, (Autonomous)
Anantapur – 515 002.

Abstract: In this study supervised neural network system was used to detect tool breakage in milling operations. The Restricted Coulomb Energy (RCE) type neural network was used for supervised learning. The effectiveness of the encoding method was tested using the RCE network on both simulated and experimental cutting force signals. The RCE network correctly categorized more than 98% of the presented data sets after training, which included simulated and experimental cutting force data.

Key words: Restricted coulomb energy, neural network, cutting force, supervised learning

INTRODUCTION

One of the most important monitoring requirements is a system capable of detecting tool breakages on-line. Unless recognized in time, tool breakage can lead to irreparable damage to the work piece and possibly to the machine tool itself. The cutting force variation characteristics of normal and broken tools are different. With the normal and broken tool cutting force variation signals is possible to train neural networks. The milling operations can be monitored with the neural network, after training. The use of restricted coulomb energy (RCE) type neural network was evaluated for detections of tool breakage, in this study [2]. Also simulation-based training is proposed to reduce the cost of preparing the systems that monitor the real cutting signals.

Many different neural network paradigms have been developed to achieve different learning and processing speed capabilities. Neural networks are also classified as supervised and unsupervised according to their learning characteristics. Neural network based systems have been applied to pattern recognition problems in many fields. RCE type supervised neural network paradigm was used for detection of tool breakage in this study. Among the available neural network paradigms, one of the most commonly used is the back propagation technique. Back-propagation is a supervised learning technique and is a perfect choice for many applications such as forecasting and analog output generation. In this paper, RCE supervised neural network paradigm was used for detection of tool breakage. RCE paradigm was used for the following reasons.

(a) The training of paradigm is much faster than the back-propagation technique:
(b) The back-propagation technique generalizes the given information in order to store it inside the initially selected hidden layers. The back propagation technique cannot give reliable decisions on the sufficiency of previous training; and
(c) RCE networks can be used to evaluate the characteristics of the data and the performance of the encoding method.

Supervised RCE networks have three layers consisting of input, hidden and output layers. The neural network inspects presented cases during the training session. For each input pattern, the neural network first checks it can accurately classify the case, based on previous training. If it classifies that input pattern correctly, it will not change the parameters of the model. If it classifies it incorrectly, based on the previous information, or it was not satisfactorily trained for the presented case, the neural network modifies the previously selected parameters, and/or assigns a new hidden node, and then selects the parameters. RCE networks do not generalize the presented cases, since only the parameters of only the related hidden nodes are changed during training. After training, the neural network can classify the given input patterns. Another important issue is the training of the neural network. It is extremely expensive and time consuming to collect cutting force data at different cutting conditions with normal and broken tools. To overcome this problem, simulation-based training of neural networks was introduced.

RESTRICTED COULOMB ENERGY (RCE) NEURAL NETWORK

The RCE network architecture is a feed-forward arrangement. This arrangement allows the network to classify pattern signals in real time without any special hardware. RCE is a parallel neural network modeled after the human learning and classification process. The network is composed of three layers of cells: the input layer; the internal or hidden layer; and the output layer. The nodes of the input layer are connected to every node of the internal layer. The nodes in the internal layer are connected selectively to the output nodes during the training process. The output nodes correspond to different pattern classes. The internal connections occur in such a way that the correct output cell will be fired when an appropriate pattern class is given to the system. To use the neural networks, sensory signals are processed and a set of features is created.

The RCE networks use two learning mechanisms. When new patterns are presented to the network, the response of the neural network is tested without any modification of the weight matrix. If the classification of the network matches the required output, the weight matrix is not changed. Otherwise, the influence of the existing nodes is modified and/or a new node will be created.

The stability of such a network may be proven through an elegant mathematical technique. A function can be found that always decreases each time the network changes state. Eventually this function must reach a minimum and stop thereby ensuring that the network is stable. The function that follows is called a Liapunov or energy methods, which have been applied to the analysis of neural networks. The central idea of the Coulomb energy network is he definition of the potential energy of a collection of memory sites. Each memory site can either attract or repel other memory sites. Memory sites, $x_1 \ldots x_m$, can be grouped into pattern classes in the feature space. The electrostatic potential energy of the configuration of a collection of charges with the same and different charges is defined by the equation for ψ :

$$\psi = (\frac{1}{2}L)\sum_{i=1}^{M} \sum_{j=1}^{M} Q_i Q_j \mid x_i - x_j \mid -L \quad (1)$$

Where ψ = an artificial network energy

L=parameter related to the memory size

M= basin of attraction

Q = output of neuron

For supervised learning, Qi (c) is defined as the charge of the memory site, class, and c, in such a way that the following criteria are satisfied:

$\text{sign}(Q_i(c)) \neq \text{sing}\ (Q_i(c')) \text{ for } c = c'$ (2)

$\text{sign}(Q_i(c)) = \text{sing}\ (Q_i(c')) \text{ for } c \neq c'$ (3)

For unsupervised learning, all patterns are assumed different and the definition of the attractive potential energy must be modified to:

$\text{sign}(Q_i(c)) = \text{sing}\ (Q_i(c')) \text{ for } c \neq c'$ (4)

A non –linear sigmoidal logistic function is then defined:

$$F_n(\sum_{m=1}^{k} \omega_{nm} = f_{mi}) = (1 + \exp(-1/kt)(\sum_{m=1}^{k} \omega_{nm} f_m + \theta))^{-1} \quad (5)$$

Where ω_{nm} is the synaptic weight. F_n and f_i indicate the afferent activity, k and t are parameters that control the steepness of the sigmoid, and θ is the threshold. Using the logistic function, the vector of the network activity that results from mapping the afferent pattern, f_i via the weight matrix, w_{nm} can be defined as:

$$x_i = \sum_{n=1}^{N} e_n F_n (\sum_{m=1}^{k} \omega_{nm} f_{mi}) \quad (6)$$

where e_n is the unit vector of the nth cell of activity.

When the synaptic weight matrix changes, motion of the memory sites in the activity space occurs. The electrostatic energy, ψ, can be minimized by the correct selection of the synaptic weight matrix. To minimize the potential energy, the gradient, with respect to the weight matrix, is determined:

$$\delta\omega_{nm} = - \delta\psi / \delta w_{nm} \quad (7)$$

$$\delta w_{nm} = \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M} Q_i Q_j |x_i - x_j|^{-(L+2)} |x_i - x_j|^{\delta/\delta w_{nm}} |x_i - x_j| \quad (8)$$

As the weight matrix develops, the potential energy for a set of M patterns is minimized.

RESULTS AND DISCUSSION

The effectiveness of the developed RCE neural network-based tool breakage detection system was tested in two stages. In the first stage, the effectiveness of he encoding approach, number of inputs, and selected paradigms were tested on simulated data. In the second stage, the system was trained on the simulated data and tested on different cutting force patterns and experimental data.

The cutting forces of the end milling operations were simulated at different conditions with perfect and broken tools. The simulated data of the broken tool were prepared by considering that one of the teeth removed 80% less material compared to the other teeth. The cutting tool was a four-flute end mill, moving at a 0.0002 m per tooth feed rate. In the simulations, the axial depths of cut were selected between 0.000127 and 0.00254 m. The RCE network was trained on nine different simulation sets. Each set had 200-400 simulated cutting force profiles. In these situations, feed rates were selected between 0.0001 and 0.001 m per tooth. The axial depths of cut ranged from 0.000127 to 0.00254 m. The broken tooth removed 80% less material than the other teeth. The other parameters of the simulations are presented in table 1. The RCE network used minimum 2, maximum 5 prototypes to classify the good and broken too signals. The neural network will classify the input patterns without any hesitation between two possible classes.

Table 1. The experimental data collected at the cutting conditions to evaluate the performance of the selected encoding and the RCE neural networks

Axial depth of cut (mm)	Feed rate (mm/min)	Spindle speed (rpm)
1.016	50.8	500
1.016	50.8	500
1.016	50.8	500
1.016	50.8	500
1.524	50.8	500
1.524	101.6	500
1.524	203.2	500
1.524	254	500
1.016	50.8	500
1.016	50.8	500
1.016	50.8	500
1.016	50.8	500
1.524	50.8	500
1.524	101.6	500
1.524	203.2	500
1.524	254	500
0.508-1.016	50.8	500
0.508-1.016	101.6	500
0.762-1.625	50.8	500
0.762-1.625	101.6	500

CONCLUSION

The resultant cutting force of milling operations was monitored by RCE type neural network to detect tool breakage. The effectiveness of the proposed encoding and selected paradigms was tested on simulated and experimentally collected data. On small training sets the RCE neural network established two o five prototypes to represent the perfect and broken tool.

The RCE neural networks were able to learn characteristics of the training cases after a few inspections, while the back propagation approach 10. needed hundreds of presentations. The learning speed and accuracy was found to be very convenient and helpful for monitoring of milling operations with single processor microcomputers. The neural network could identify more than 98% of the presented test data with confidence following computer simulation-based raining. The proposed encoding method method and RCE type neural networks were found to be convenient and beneficial for detection of tool breakage when a low cost system to monitor repetitive cutting operations in which the cutting conditions are constant is going to be developed.

REFERENCE

1. Laurene Fauselt, "Fundamentals of Neural Networks, Architectures, Algorithms, and Applications", Pearson Education, 2nd Edition, 2004.
2. Ibrahim N. Tansel, "Modelling 3-D Cutting Dynamics with Neural Networks" , Int. J. Mach tools Manufact, Vol-32, No.6, pp. 829-853, 1992.
3. Ibrahim Nur Tansel and charles Mclaughlin, "Detection of tool breakage in milling operations -I, The time series analysis approach", Int.J.Mach.Tools Manufact, vol.33, No.4, pp-531-544, 1993.
4. Yusuf Altintas, "In-processs detection of tool breakages using time series monitoring of cutting forces", Int.J.Mach.Tools Manufact, vol.28, No.2, pp-157-172, 1998.
5. Nestler.A and Schulz.G, "Cutting values prediction with Neural Network", Institute of Production Engineering, Nommsenstr, Germany.
6. Freeman.A. James and Skapuera.M.David, "Neural Networks", Pearson Education, 1991.
7. Haykin Simon, "Neural Network", Eastern Economy Edition, PHI 2nd Edition, 1999.
8. Zurada.M.Jacek, "Artificial Neural Systems", A Jaico Puclishers, 2000.
9. Philip D. Wasserman "Neural Computing" Theory and Practice, ANZA Research, Inc, VAN NOSTRAND REINHOLD, 1989.