Chatter Detection in Milling Machines by Neural Networks classification and feature selection

Abstract
In modern industry, milling is an important tool when a high material removal rate is required. Chatter detection in this situation is a crucial step for improving surface quality and reducing both noise and rapid wear of the cutting tool. This paper proposes a new methodology for the chatter detection in CNC milling machines. This methodology is based on vibratory signal analysis and artificial intelligence. The methodology consists in five major steps: 1) data acquisition, 2) signal processing, 3) features generation, 4) features selection and 5) classification. As chatter components occur around system resonance frequencies, a multiband resonance filtering method is proposed at the processing step. The process is then followed by envelope analysis. This allows for increasing the signal-to-noise ratio and increasing the sensitivity of generated features. Extracted features are then ranked based on their entropy in which only best features are selected and presented to the system for classification. At the classification step, the selected features are classified into two classes: stable and unstable utilizing neural networks. Two neural network approaches, Radial Basis Function (RBF) and Multi-Layer Perceptrons (MLP), are tested. The developed approach is applied for chatter detection in Huron K2X10 milling machine. This approach is tested on a milling machine at different depths of cut and various rotational speeds. Discussions are made and the results confirm the accuracy of the proposed methodology.

Keywords
Chatter detection, Machining, Neural Networks, Envelope, Resonance filtering, Feature Selection.
Introduction

Chatter is a self-excited vibration that can arise in machining process at specific combinations of cutting parameters, depth of cut and spindle speed. This phenomenon affects the surface finish resultant from strong vibrations of the cutter. Furthermore, the machine, tool and bearings spindle wear out rapidly and a lot of noise is generated when chatter occurs. In machining field, chatter phenomenon is investigated extensively because manufacturing enterprises are turning to the automation system and the development of reliable and robust monitoring system for providing increased productivity, improved part quality and reduced costs (Liang et al., 2004). Chatter occurrence has several disadvantages (Quintana and Ciurana, 2011): a) poor surface quality, b) unacceptable inaccuracy, c) excessive noise, d) machine tool damage, e) reduced material removal rate, f) increase of costs in terms of production time, g) waste of material, h) environmental impact in terms of materials and energy. Moreover, chatter monitoring is not an easy task for various reasons. Firstly, the non linearity of machining processes and the time-varying of systems complicate this task. Secondly, the sensitivity and the dependency of acquired signals from sensors on different factors, such as machining condition (depth of cut, spindle speed, feed), cutting tool geometry and workpiece material (Vallejo et al., 2008). For these reasons, advanced automatic chatter detection and monitoring system is needed for optimizing and controlling machining processes and this becomes a topic of great interest. Several researches in this field are performed. Advanced monitoring and detection methods are developed mostly relying on time, frequency and time-frequency analysis, etc. Different methods have been applied for monitoring machining process (Abellán and Subiron, 2010). Nayfeh and Balanchandran (1994) analyzed the stability of milling operation using Poincaré section. They used bifurcation diagram for identifying the transition from stable cutting to chatter. Insperger (2002) analyzed the stability of system by studying differential equations of machining tool modeling. Bayly et al. (2002) used variance indicators in time domain to determine stability boundaries. Smith and Tlusty (1990) proposed the peak-to peak forces technique to identify the limits of stability. Li et al. (1997) proposed a frequency domain approach based on the cross-coherence of two perpendicular acceleration signals to detect wear and chatter. Bickraj et al. (2008) used Index Based Reasoning (IBR) for detection of the development of chatter in end milling operations. New time and frequency indicators based on cyclostationary approach are used for tool and chatter monitoring (Lamraoui et al., 2011a, 2011b). Many researchers developed monitoring process approach for the detection of chatter using the dynamic cutting forces. However, the use of a dynamometer is a drawback since not practical in normal industrial operation. These techniques consume much computational time to operate online in high speed milling and are more suitable for low speeds. Signal analysis can help to overcome these problems. Fault detection is conducted in the following phases: data collection, signal processing, feature extraction, selection and classification (Abellán and Subiron, 2010). The most effective features are investigated for the identification of chatter in their early stages. In the feature extraction process, a set of new features from the original data are extracted into a distinct feature space (Wyse and Jain, 1980). The time-domain vibration signal is useful to give a simple scalar indicator and in many cases, it is sufficient to diagnose the health of a machine. However, time signals are not always efficient to detect a defect at its early stage of degradation. In that case, the spectral domain obtained from Fourier analysis may be used to extract frequency-domain features of measured vibration signals (Samanta et al., 2006). Some approaches propose to combine time domain features with others extracted from the frequency domain (Li et al., 2000) for a better diagnosis. The adepts of data fusion use a large number of features for effective condition monitoring (Wen and Qiu, 1997; Malhi and Gao, 2004). However, this would affect
the training cost and time, as well as the classification accuracy due to the presence of irrelevant or redundant features (Widodo et al., 2007). Moreover, the sensitivity of features may vary considerably under different operating conditions (Wen and Qiu, 1997; Malhi and Gao, 2004). Also, in order to improve the classification accuracy and reduce computation time, some salient features need to be selected from the original feature set. There are also some feature selection methods such as conditional entropy (Lehrman et al., 1997), genetic algorithms (Samanta et al., 2006), distance evaluation technique (Lei et al., 2008), distance discriminant technique (Liang, 2007), principle component analysis (Barakat et al., 2012), etc. At the classification step, selected features are classified using classification tools. Neural Networks (NNs) present numerous advantages (Bishop, 1995; Haykin, 1999) to be used as such tool for many reasons: (a) NNs have the ability to learn from historical data; (b) they use parallel computation which is suitable for real time monitoring; (c) they can theoretically approximate any continuous input-output mapping to any desired degree of accuracy; (d) Neural networks can be used to extract and detect trends that are complex, and they may overcome the non-linear difficulty (Rangwala, 1990); (e) they have also the advantage of reducing the effects of noise in collected signal and increasing the pattern recognition in a wide parameter range (Chen et al., 2000). In the last decade, NNs have been applied to monitor machining processing, including turning, milling and grinding machines (Rangwala, 1990). Wavelet analysis and artificial neural networks may be applied to condition monitoring in high speed milling in order to detect the instability behavior (Ebner et al., 2006; Yao et al., 2010). Zhou et al. (2007) tried to predict chatter by establishing time series model of vibration acceleration signal in cutting process based on Hidden Markov Model. Regarding the non linear and the non stationary characteristics of milling process, neural networks seem the adequate classification approach to be used concerning vibration signals in order to separate stability from instability cases.

In this paper a procedure based on vibratory signal analysis and artificial intelligence for detecting chatter phenomenon is proposed. The approach is put forward to classify the accelerometer signals related to stable and unstable conditions. Signal processing strategy is mandatory after data acquisition and before generation and selection of features, due to the mechanical and electrical impact noises. In fact, chatter phenomenon is linked to the dynamic behavior of the machine-tool-workpiece system. Chatter components occur around system resonances frequencies and therefore, a multiband resonance filtering is proposed at the processing step. The process is then followed by envelope analysis. This allows for increasing the signal-to-noise ratio and increasing the sensitivity of generated features. Extracted features are ranked based on their entropy in which only best features are selected and presented to the system for classification. At the classification step, two neural network approaches, Radial Basis Function (RBF) and Multi-Layer Perceptrons (MLP), are tested. The two networks classify the selected features into stable or unstable classes. The developed approach is applied for chatter detection in Huron K2X10 milling machine. The originality of this application is to study and detect the chatter phenomenon regardless the operating condition of the machine (depth of cut, spindle speed …). In this situation, the classification problem becomes more complicated but it facilitates the automation and reduces both cost and time consumption. The paper is organized as follows. In section 2, data acquisition and taxonomy are presented. It covers the main stages involved in developing diagnosis tool for machinery system. Experimental data acquisition is then described and various parts of signal analysis are demonstrated. Finally, brief reviews of two pattern recognition classifiers (RBF and MLP) are presented. Classification results and discussion are exhibited in section 3. Conclusions and perspectives are put forward in the final section.
Methodology and taxonomy

Chatter is mainly classified in two classes: primary and secondary. Primary chatter can be caused by friction between tool and workpiece, thermodynamics of the process or even mode coupling (Tlusty, 2000). Secondary chatter, which is more significant, is due to regeneration of waviness on workpiece surface (Faassen et al., 2003). It can be explained as follows: as the tooth passes into the workpiece, a wavy surface finish is left. In the following passage, the second tooth encounters this wavy surface and generates its own ripples. Depending on the phase shift between these two waves, the chip thickness may vary rapidly leading to an amplitude variation of forces and consequently to vibrations (Fig. 1).

When the chip thickness exceeds a certain limit, this regenerative effect becomes dominant and chatter develops, producing poor and wavy surface finish. Advanced approaches that can automatically identify the stability of the system regardless cutting conditions (depth of cut, spindle speed) are then highly recommended. This paper proposes detection approach based on signal analysis and artificial intelligence that fulfills these tasks. This technique classifies sensitive features extracted from accelerometer signals to detect the chatter that is related to unstable conditions. The methodology consists in five major parts: 1) data acquisition, 2) signal processing, 3) features generation, 4) features selection and 5) classification. Each step is discussed in the following subsections. Figure 2 shows the developed methodology for chatter detection of milling machine process. The process is initiated by acquiring vibratory data. Accelerometers are implemented on the machine at different directions to collect multi-axial data. At the processing step, the vibratory signals are filtered in the frequency domain by using a multiband resonance filter and their envelopes are demodulated. Various statistical features in time domain are then extracted. They are ranked depending on their entropy factor and a group of them is selected for classification. At the classification step, the selected features are classified into two classes: stable and unstable utilizing neural networks. Finally an output decision is made for each tested sample. The different steps of the proposed methodology are discussed in detail in the following subsections.
Experimental data acquisition

All the experimental milling processes are performed by dry cutting using a vertical machine center Huron K2X10 which is able to operate at high rotational speeds (28000 rpm). Figure 3 shows the experimental setup used to achieve tests on slot milling at different cutting conditions. Aluminum samples # 7075-T6 # of size [194mm, 133mm, 50mm] are used and slot milling is created using 2 flutes micro grain solid carbide end mills characterized by a helix angle of 30°, maximum depth of cut of 38.1 mm and tool diameter of 25.4 mm. Three uni-axial piezoelectric accelerometers are used to capture the vibrations from 1 to 10 kHz. The measurement range is ±500g pk. Two of them are attached on the free end of the non-rotating part of spindle (spindle support) and oriented along the X and Y directions, in parallel to the machine axis. The third accelerometer is attached to the workpiece. It is oriented along the cutting direction and parallel to the Y axis of machine center. We denote its direction to be Y*. Data acquisition was conducted with a 24-bit accuracy. It is configured to obtain signals with a maximum sampling rate of 48 kHz. The acquisition is controlled by BETAVIB system. Dry slot milling tests are performed with 2 flutes end mill and feed per tooth $f_d$ of 0.03 mm.
It is known that machining and gyroscopic effects may affect the modal parameters when rotating at very high speeds (Gagnol et al., 2011; Badri et al., 2009). If rotating at very high speeds, it is preferable to measure the modal parameters by ARMA (Vu et al., 2013). However, the actual methodology for measuring modal parameters and computing stability lobes still use modal parameters as measured in static by tap tests (Altintas et al., 2004). It is why we have kept this method, since our speeds are not high enough. A tap test is applied on the tool tip in (X, Y) directions to measure the modal parameters of the spindle (including tool and tool holder). The system of measurements consists in:

1. a hammer impact as excitation source.
2. accelerometers to measure the vibratory responses.
3. data acquisition system with conditioner and analyzer to collect and analyze the measured data.

The H1 estimation technique was used to estimate the FRF, which is the ratio between the inter-correlation between force and response $S_{fx}$, and the auto-correlation of force $S_{ff}$. FRF in X and Y directions with real part, imaginary part, magnitude and coherence are shown in Figure 4.
Their magnitudes are used to compute the modeling parameters of the system. Four major peaks corresponding to resonance phenomenon appear near 860, 1600, 2400, and 3400 Hz. After identifying the resonance frequencies, manipulations are made to study the stability of the system. The stability lobe diagram (SLD) (Fig. 5) is calculated based on the analytical method.

**Analytical Stability Lobe Diagram**

![Analytical Stability Lobe Diagram](image)

The SLD is obtained using the following steps: (1) select chatter frequency around the dominant mode; (2) calculate the phase angle of the structure at the selected chatter frequency; (3) compute the critical depth of cut; (4) calculate the spindle speed for each stability lobe and (5) repeat the procedure by scanning the chatter frequencies around all the structure natural frequencies. Spindle speed is operated at 5 rotational speeds (3000, 5000, 8000, 10000, and 12000) rpm (Table 1). The parameters corresponding to experimental data acquisition are synthesized in Table 1 and Figure 5.

### Table 1. Summary of the cutting conditions for the chatter tests.

<table>
<thead>
<tr>
<th>Test #</th>
<th>$d$ cutting depth (mm)</th>
<th>S cutting speed (m/min)</th>
<th>R rotational speed (rpm)</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>239.4</td>
<td>3000</td>
<td>No chatter</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>239.4</td>
<td>3000</td>
<td>No chatter</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>239.4</td>
<td>3000</td>
<td>chatter</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>398.9</td>
<td>5000</td>
<td>No chatter</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>398.9</td>
<td>5000</td>
<td>chatter</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>638.3</td>
<td>8000</td>
<td>No chatter</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>638.3</td>
<td>8000</td>
<td>No chatter</td>
</tr>
<tr>
<td>8</td>
<td>6</td>
<td>638.3</td>
<td>8000</td>
<td>chatter</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>797.9</td>
<td>10000</td>
<td>No chatter</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>797.9</td>
<td>10000</td>
<td>chatter</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>957.5</td>
<td>12000</td>
<td>No chatter</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td>957.5</td>
<td>12000</td>
<td>No chatter</td>
</tr>
<tr>
<td>13</td>
<td>5</td>
<td>957.5</td>
<td>12000</td>
<td>No chatter</td>
</tr>
<tr>
<td>14</td>
<td>6</td>
<td>957.5</td>
<td>12000</td>
<td>No chatter</td>
</tr>
</tbody>
</table>
In each test, the depth of cut is increased until chatter occurred. Tests are stopped at a cutting depth of 6 mm if no chatter is observed. The number of experimental tests is 14, ten of them correspond to stable conditions and the others four emerge chatter phenomenon. Data are recorded at different rotational speeds for stable and unstable conditions.

**Processing step.**

The stability of machining highly depends on the dynamic behavior of the machine, which is related to tool-toolholder-spindle modes. However, dominant vibration frequencies that are usually associated with chatter frequencies in unstable conditions are in most of cases close to the natural frequencies (Dombovari, 2011). At the processing step, we propose that collected samples are first filtered by using multiband filters. The bandwidths are related to frequency resonance values. The envelopes of filtered samples are then analyzed and time features are extracted. The feature selection is made for both cases: (a) when the system operates at stable conditions and (b) when the system experiences chatter phenomenon. The aim is to obtain the more efficient features to discriminate the operating conditions. The multiband resonance filtering and the envelope analysis are detailed in the following subsections.

**Multiband resonance filtering.**

The multiband filter processing (Fig. 6) is a mandatory step before extracting and selecting significant features for two major reasons: (1) decreasing the high level of environmental noise and (2) keeping information that are correlated to the studied phenomenon.

![Multiband filter, difference between the designed filter and the desired one.](image)

Different steps are followed for implementing the multiband filter: 1) The natural frequencies are first extracted from real and imaginary part of the FRF as experimentally measured; 2) the natural frequencies that present the predominant modes are identified and selected. Four were selected and their values are approximately: 860Hz, 1600Hz, 2400Hz and 3400Hz; 3) a digital filter is implemented. The
FIR2 as designed by Matlab function is used because of the following advantages: 1) it can have linear phase 2) it is always stable and can be efficiently realized in hardware. However, the primary disadvantage is in requiring a high filter order. Figure 6 displays the designed filter in dashed red curve and the desired one in blue curve. The designed filter includes four bandwidths B (1 to 4) selected around the natural frequencies. The characteristic frequencies of the designed filter are appointed from f1 to f16. The filter order reaches 5000 to achieve good precision.

Envelope analysis.
After filtering the collected samples by mean of multiband filters, their envelope is computed. This step is required because envelope analysis has been found suitable for chatter detection. The envelope analysis allows for increasing the signal-to-noise ratio and increases the sensitivity of generated features. Chatter frequencies can modulate natural resonances so envelope analysis of multiband pass filtered signals around their high resonance frequencies can be an efficient tool for monitoring. The demodulation of vibratory signals is then applied by using the Hilbert transform (Thomas, 2011) in order to obtain the signal envelope. It has been shown that the envelope analysis is linked with the cyclic spectral density and considered as a valuable tool for the analysis of cyclostationary signals (Randal, 2001).

Feature generation
The third step is the extraction of statistical features from envelope signals. This extraction transforms the envelope signals into several descriptors that describe the signals. These indicators are correlated with milling states and influenced by chatter phenomenon. Different methods have been applied for features generation in time, frequency, or in time-frequency domains. In this work, nine scalar descriptors were investigated: Variance, Skewness, Kurtosis, Peak value, Root Mean Square (RMS), Clearance factor, Crest factor, Shape factor, Impulse factor (Teti et al., 2010; Thomas, 2011). Table 2 summarizes these features. It must be noticed that the definition of the peak value has not been considered has the maximum value, but as half the peak-to-peak value. This definition affects the Clearance, Crest factor and Impulse factor.

Features’ selection
After the features’ generation step, many descriptors are obtained. The features’ selection step is crucial in order to find which features are the most significant and reliable for chatter detection. In fact, if features are selected with little discrimination power, the ulterior design of a classifier would conduct to poor performance. On the other hand, if features with rich information are selected, the classification accuracy increases and the computation time is reduced. For this reason, a selection algorithm based on feature ranking is utilized. This algorithm is a salient approach among other selection techniques because it employs a metric to rank the features and excludes all ones that do not achieve appropriate accuracy. In this paper, relative entropy measure is proposed to rank features from best to worst and then, a set of only top listed features is presented for classification. Relative entropy measures the "distance" between the probability functions $P(x/w_1)$ and $P(x/w_2)$, where $w_1$, $w_2$ are two classes and $x$ is feature vector.
Table 2. Scalar descriptors

<table>
<thead>
<tr>
<th>Description</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variance (Var)</td>
<td>$\frac{1}{N} \sum_{n=1}^{N} (s_n - \bar{s})^2$</td>
</tr>
<tr>
<td>Skewness</td>
<td>$\frac{1}{N} \sum_{n=1}^{N} (s_n - \bar{s})^3 \sqrt{\text{Var}}^{3/2}$</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>$\frac{1}{N} \sum_{n=1}^{N} (s_n - \bar{s})^4 \sqrt{\text{Var}}^{-1}$</td>
</tr>
<tr>
<td>Peak value (Pv)</td>
<td>$\frac{1}{2} [\max(s_n) - \min(s_n)]$</td>
</tr>
<tr>
<td>Root Mean Square (RMS)</td>
<td>$\sqrt{\frac{1}{N} \sum_{n=1}^{N} s_n^2}$</td>
</tr>
<tr>
<td>Clearance factor (CLf)</td>
<td>$\frac{1}{N} \sum_{n=1}^{N}</td>
</tr>
<tr>
<td>Crest factor (Cf)</td>
<td>$\frac{P_v}{\text{RMS}}$</td>
</tr>
<tr>
<td>Shape factor (Sf)</td>
<td>$\frac{1}{N} \sum_{n=1}^{N}</td>
</tr>
<tr>
<td>Impulse factor (If)</td>
<td>$\frac{1}{N} \sum_{n=1}^{N}</td>
</tr>
</tbody>
</table>

where $N$ is the length of discrete signal $s_n$ and $n$ is a sampling point.

This is also known as Kullback-Leibler distance ($d$) and defined as (Theodoridis and Koutroumbas, 2008):

$$d = D_{12} + D_{21}$$  \hspace{1cm} (1)

$$D_{12} = \sum p(x/w_1) \log_2 \left( \frac{p(x/w_1)}{p(x/w_2)} \right)$$  \hspace{1cm} (2)

$$D_{21} = \sum p(x/w_2) \log_2 \left( \frac{p(x/w_2)}{p(x/w_1)} \right)$$  \hspace{1cm} (3)

Assuming now that the density functions are Gaussians: $N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$, respectively, the computation of the divergence $d$ becomes:

$$d = \frac{1}{2} \left( \frac{\sigma_2^2}{\sigma_1^2} + \frac{\sigma_1^2}{\sigma_2^2} - 2 \right) + \frac{1}{2} (\mu_1 - \mu_2)^2 \left( \frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2} \right)$$  \hspace{1cm} (4)

In case of completely overlapped classes, $d=0$. The values of $d$ increase as soon as the two distributions move far apart from each other. In other words, good features correspond to high values of $d$.

**Classification by mean of neural networks**

At the classification step, classical analytical techniques often cannot provide acceptable solutions to difficult diagnosis tasks. This explains why soft computing techniques such as neural networks become more and more popular in industrial applications of fault diagnosis. In this paper, two neural network
approaches, Radial Basis Function (RBF) and Multi-Layer Perceptrons (MLP), are investigated. We have chosen to work with neural networks (MLP and RBF) for many reasons. First, it is that we can learn these networks can learn by updating the weights and biases; this provides flexibility for the networks to adapt, recall and generalize from training data. Secondly, RBFs are nonlinear, they include kernel functions and have the ability of pattern recognition and they have the capabilities of processing data in parallel (rapid computations and decisions). Thirdly, MLP and RBF don’t need physical, geometrical or in-depth knowledge about the system which in industrial applications means reducing the number of sensors and no physical in-depth knowledge should be known. Consequently, MLP and RBF seem to be good choices for real and industrial applications. They are used to classify the selected features into two classes (stable and unstable). A brief theoretical review for the two networks is presented in the following sections.

Multi-layer perceptron (MLP).

The multi-Layer Perceptron (MLP) is a feed forward neural network which consists in units arranged in layers with only forward connections to units in subsequent layers (Theodoridis and Koutroumbas, 2008). The connections have weights associated with them. Each signal traveling along the link is multiplied by a connection weight. The first layer is the input layer and the input units distribute the inputs to units in subsequent layers. In subsequent layers, each unit sums its inputs, adds a bias or threshold term to the sum and transforms the sum to produce an output. This transformation is called the activation function of the unit; it can be linear or non linear depending on the chosen activation function. The output layers of computational nodes are related directly to total number of classes, in our case they correspond to two classes “stable and unstable”. Figure 7 shows the architecture of MLP with an input layer, two hidden layers and an output layer.

![Figure 7. Architecture of Multi-Layer Perceptron (MLP).](image)

The input layer consists of the inputs to the network (input parameters), then follows a hidden layer, which consists of any number of neurons, or hidden units placed in parallel. Each neuron performs a weighted summation of the inputs, which then passes a nonlinear activation function $T$ also called the transfer function. The transfer function of the network modifies the input to give a desired output. The number of nodes within the input and output layers are dictated by the nature of the problem to be solved and the number of input and output variables needed to define the problem however the choice of hidden layers number, hidden nodes and type of activation function plays an important role in model constructions (Komgong et al., 2007; Haykin et al., 1999). Let $x_i, i=1,...,Q$ be the input parameters. $w_{ji}$ are the network weights in which $j=1,...,\hat{N}$ and $\hat{N}$ is the number of nodes. The output vector of MLP is denoted by $f_k^{MLP}$ where $P$ is the number of classes. The weighted sum of the inputs $x_i$ and bias
term of each neuron $\hat{b}_j$ are passed to activation level through a transfer function $T$ to produce the output, and the units are arranged in a layered feed-forward topology. Each node in a layer (except the ones in the input layer) provides a threshold of a single value by summing up their input value $x_i$ with the corresponding weight value $w_{ji}$. Then the neuron’s net input value is formed by adding up this weighted value (sum), with the bias term $b_j$. The bias is added to shift the sum relative to the origin. The net input value then goes into transfer function $T$, which produces the neuron output $o_j$.

$$o_j = T\left(\sum_{i}^{Q} w_{ji} x_i + b_j\right)$$

(5)

The transfer function $T$ that transforms the weighted inputs into the output is usually a non linear function. The sigmoid (S-shaped) or logistic function is the most commonly used transfer function which restricts the nodes output between 0 and 1: \(\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}}\) (Howard S., 1994). The learning rule is applied to train the network to perform some particular task. Supervised learning is used which states that for every input $x_i$ there exists a target “desired output” $t_k$. The learning rule is provided with a set of examples (the training set) of proper network behavior: \(\left\{f_{1,M_{MLP}}, t_{1}\right\}, \left\{f_{2,M_{MLP}}, t_{2}\right\}, \ldots, \left\{f_{p,M_{MLP}}, t_{p}\right\}\) (Moody and Darken, 1989). As the input parameters are applied to the network, the network outputs are compared to the targets. The learning rule is then used to adjust the weights and biases in order to move the network outputs closer to the targets. MLPs learning rule falls in this supervised learning category. The objective is to reduce the error function '$e$', which is the mean square difference between network outputs and desired targets.

$$e = \frac{1}{P} \sum_{k=1}^{p} \left(t_k - f_{k,M_{MLP}}\right)^2$$

(6)

The perceptron learning rule calculates desired changes to the perceptron's weights and biases. Newton minimization and Levenberg-Marquardt Algorithms (LMA) may also be used in updating the weights and biases of network nodes. LMA is a very popular curve fitting algorithm [Marquardt, 1963]. However LMA finds a local minimum and not a global minimum. Moreover a damping parameter should be initiated and the algorithm converges only if the initial guess is already somewhat close to the final solution. However, we chose the gradient descent method because each node in the network should be updated quickly. In fact, when the classes are well separated, there will be no need to complicate the computations and go to higher orders as in Newtonian method. The adjusted weights and biases of $m^{th}$ layer at iteration $u$ are computed as follows:

$$w_{j,i}^{m}(u + 1) = w_{j,i}^{m}(u) - \alpha \frac{\partial e}{\partial w_{j,i}^{m}}$$

(7)

$$\hat{b}_j^{m}(u + 1) = \hat{b}_j^{m}(u) - \alpha \frac{\partial e}{\partial \hat{b}_j^{m}}$$

(8)
where $\alpha$ is the learning rate, $w_{ji}$ are the weights and $b_j$ are the biases.

**Radial basis function classifier.**

RBF is a feed-forward neural network trained using a supervised training algorithm (Haykin S., 1999). Its network architecture (Figure 8) is proven to be useful (Moody and Darken, 1989). It is typically configured with a single hidden layer of units whose activation function is selected from a class of functions called basis functions. The ability of the RBF network to recognize whether an input is near the training set or outside the trained region provides a significant benefit over MLP. The major difference between RBF networks and MLP is the behavior of the single hidden layer. Rather than using the sigmoid or other transfer functions, the hidden units in RBF networks use a Gaussian or some other basis kernel functions.

![Figure 8. Architecture of RBF network.](image)

- Input layer is made up of nodes, its dimension equal to the number of training parameters.
- Hidden layer of tuned neurons centered over receptive fields for non-linear mapping.
- Output layer with dimension equal to the total number of classes, provides the response of the network.

The RBF network has a feed forward structure. Its input layer is made up of source nodes whose number is equal to the dimension the input vector $x_i \in \mathbb{R}^Q$, where $Q$ is the total number of input parameters. The hidden layer consist of a non-linear mapping of the inputs in which each hidden unit acts as a locally tuned processor that computes a score for the match between the input vector and its connection weights or centers. RBF network performs a local mapping (i.e., only inputs near specific receptive fields will produce an activation). The units (in the hidden layer) receiving the direct input from a signal may see only a portion of the input pattern, which is further used in reconstructing a surface in a multidimensional space that furnishes the best fit to the training data. The basis units are highly specialized pattern detectors. Each hidden unit output $\phi_j$ is obtained by closeness of the input $X = \{x_i \in \mathbb{R}^Q\}$ to $j^{th}$ dimensional vector $\mu_j$ associated with $j^{th}$ hidden unit in which $\{j = 1, \ldots, \tilde{N}\}$ and $\tilde{N}$ is the number of nodes in hidden layer (Broomhead and Lowe, 1988). If an input vector $X$ lies near
the center of a receptive field $\mu_j$, then that hidden node will be activated. The response characteristics of the $j$th hidden unit is assumed to be a Gaussian hidden function,

$$\phi_j = \exp \left( -\frac{\|X - \mu_j\|^2}{2\sigma_j^2} \right)$$

(9)

where $\mu_j$ and $\sigma_j$ are mean and the standard deviation respectively, of the $j$th unit receptive field of Euclidean norm.

The RBF output is a linear mapping of the hidden layer outputs (i.e. the output units implement a weighted sum of hidden units output). Given an input vector $X$, the output of the RBF network

$$F_{\text{RBF}} = \left\{ f_k \right\}$$

vector is of dimension $P$, whose $k$th component $\{k = 1, \ldots, P\}$ is given by,

$$f_k (X) = \sum_{j=1}^{\tilde{N}} w_{kj} \phi_j (X)$$

(10)

where $w_{kj}$ are the network weights.

Finding the RBF weights of output layer is achieved by training the network. In supervised learning, the network is trained as input output pairs so the parameters of network are optimized in order to fit the network outputs to the actual outputs. A training set of $P$ different labeled pairs $\left\{ f_k, t_k \right\}$ represents associations of a given mapping and their corresponding targets. The sum of squared error criterion function can be considered as an error function $E$ to be minimized over the given training set in order to develop a training method that minimizes $E$ by adaptively updating the free parameters of the RBF network. The cost error $E$ is simply the squared error between the network output $f_k$ and the corresponding desired output $t_k$. Because of the differentiable nature of the RBF network transfer characteristics, a supervised gradient-descent method over $E$ is used to update the network parameters (Komgom et al., 2007).

$$\Delta \mu_j = -\rho_{\mu} \nabla_{\mu_j} E$$

(11)

$$\Delta \sigma_j = -\rho_{\sigma} \frac{\partial E}{\partial \sigma_j}$$

(12)

$$\Delta w_{kj} = -\rho_w \frac{\partial E}{\partial w_{kj}}$$

(13)

where $\rho_{\mu}$, $\rho_{\sigma}$ and $\rho_w$ are small positive constants.
These parameters are fixed in the gradient descent classical method and are equal to 0.1. These parameters adjust the convergence rate of the algorithm and are determined empirically. If they are too large, the algorithm is not stable and fluctuated around a solution, and if they are too small, a very large number of iterations will be required to converge to the solution, and thus the probability of convergence to a local solution increases. This method is able of matching or exceeding the performance of neural networks with back-propagation algorithm, but gives training comparable with those of sigmoid type of MLP (Haykin S., 1999).

**Application to milling machine for chatter detection**

The proposed approach is applied on CNC milling machine. The principle goal is to detect the chatter phenomenon regardless of the rotational speed and depth of cut. The problem is generalized and the solution becomes more difficult to obtain. The second objective is to determine the best position for a piezoelectric sensor to be placed. After that, parameters that can distinguish between stable and unstable (chatter phenomenon) situations, are investigated. Finally, discussions and comparisons of different approaches are carried out to confirm the effectiveness of the proposed methodology. During the manipulations, a total number of 23590 signals (one sample presents one spindle revolution) are collected from each piezoelectric sensor at different depths and various rotational speeds. 17219 signals were corresponding to stable case and the others 6371 samples include chatter phenomenon.

**Classification approach**

Figure 9 exhibits the classification process. Each signal is filtered using the band resonance filtering method and statistical features are generated from its envelope. Features are then selected depending on their relative entropy measurement. Selected features are classified using neural networks (RBF and MLP) into two classes: stable and unstable that represents the chatter phenomenon.

![Figure 9. Architecture of the proposed methodology.](image)

A good classification of data entered for training is important in order to get an efficient detection method. The stability diagram (Fig. 5) indicated the stable and unstable conditions. 40% of data (known classes: stable and unstable samples) are used to train the network. The rest (60% of data) are used for testing step. During the training step, all the extracted features are ranked from best to worst according to their capability to separate the two classes (stable and unstable), using relative entropy method. Then, best ranked features are selected. This feature vector is an input entry of the network.
Feature vectors of training samples are labeled to their desired class (stable or unstable). The input and the output targets may now be identified. The training process is then started by updating the weights and biases of the networks aiming to minimize the least mean square error between the network output and the desired output. Both network classifiers MLP and RBF have same input and output layers. They only differ in their hidden layers.

From previous works, it was shown that 2 hidden layers of MLP allowed for ensuring good classification with minimum computations. The number of hidden nodes depends on how much the classes of training data are separable (more classes are hardly separable, more hidden nodes should be added) and on the number of input features (more input features, more hidden nodes). Furthermore, an investigation of weights of different links allows for removing the links too weak. In our case, with four selected parameters and since the data collected at some speeds are hardly separable, the MLP has been designed with two hidden layers and 12 hidden nodes into each. On the other hand, the RBF classifier consists of one hidden layer with growing number of hidden nodes depending on the training conditions. At the end of the ranking process, features are ranked in decreasing order depending on their factor.

For selecting the input parameters, we made an algorithm that iteratively realize the classification first for one parameter then for two parameters then for three parameters ..., then till all parameters. We found that the optimum solution (best classification) is obtained after selecting the first four parameters. Table 3 exhibits the features collected from three piezoelectric sensors located in different locations (X), (Y) and (Y*), with corresponding entropy values. The four best indicators (Variance, RMS, Clearance and Peak) were selected since their entropy is greater than the others. They are the same at three directions (x, y, y*). However, it can be noticed that the order of their importance which is determined according to selective factor (relative entropy) differ. The variance is revealed as the more sensitive in the X direction, while the clearance was the more sensitive in the Y direction.

<table>
<thead>
<tr>
<th>Order</th>
<th>X</th>
<th>Entropy</th>
<th>Y</th>
<th>Entropy</th>
<th>X*</th>
<th>Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Variance</td>
<td>20</td>
<td>Clearance</td>
<td>9.8</td>
<td>Clearance</td>
<td>12.6</td>
</tr>
<tr>
<td>2</td>
<td>Peak</td>
<td>12.7</td>
<td>RMS</td>
<td>6.7</td>
<td>Peak</td>
<td>9.9</td>
</tr>
<tr>
<td>3</td>
<td>Clearance</td>
<td>7.6</td>
<td>Peak</td>
<td>5.8</td>
<td>RMS</td>
<td>9.6</td>
</tr>
<tr>
<td>4</td>
<td>RMS</td>
<td>7.1</td>
<td>Variance</td>
<td>5.0</td>
<td>Variance</td>
<td>9.2</td>
</tr>
<tr>
<td>5</td>
<td>IF</td>
<td>1.14</td>
<td>SF</td>
<td>0.34</td>
<td>Kurtosis</td>
<td>0.84</td>
</tr>
<tr>
<td>6</td>
<td>CF</td>
<td>1.12</td>
<td>CF</td>
<td>0.33</td>
<td>CF</td>
<td>0.42</td>
</tr>
<tr>
<td>7</td>
<td>SF</td>
<td>1.03</td>
<td>Impulse fac.</td>
<td>0.27</td>
<td>IF</td>
<td>0.24</td>
</tr>
<tr>
<td>8</td>
<td>Skewness</td>
<td>0.57</td>
<td>Kurtosis</td>
<td>0.18</td>
<td>SF</td>
<td>0.14</td>
</tr>
<tr>
<td>9</td>
<td>Kurtosis</td>
<td>0.12</td>
<td>Skewness</td>
<td>0.14</td>
<td>Skewness</td>
<td>0.12</td>
</tr>
</tbody>
</table>
The classification rate error of testing samples at different positions (X, Y and Y*) using RBF and MLP network classifiers are presented in the following diagram. Let’s denote stable error, unstable error and undefined error to be SE, UE and UN respectively.

- **SE** is the percentage number of unstable samples classified as stable with probability more than 55%.
- **UE** is the percentage number of stable samples classified as unstable with probability more than 55%.
- **UN** is the percentage number of stable and unstable samples which were difficult to classify since their classification probability was varying between 45% and 55%.

Figure 10 demonstrates that the percentage of error of stable and unstable samples don’t exceed 10% in both classifiers (RBF and MLP).

The simplest way is select a class is to choose a 50% level, but we decided to raise it to 55% in order to be more accurate in taking right decisions (classification with accuracy more than 55%). We have thus decided that samples belong to a certain class if their probability is greater than 55%. This creates a grey zone UN of unknown (third class) that varies from 45% to 55%. In this grey zone (UN), the testing samples are neither stable nor unstable. It can be noticed that RBF gives similar results than MLP in X and Y directions for SE and UN, but it provides worst result in the Y* direction. It may be noticed that UN classification errors UN are high in both types of neuron networks (MLP, RBF) and in the three positions (X, Y, Y*). This error (UN) is related to the fuzzy samples that do not belong to stable or unstable classes. This error informs us about the system complexity and the difficulty of discrimination when machining takes place in the limit of stability. The highest error in both networks (RBF and MLP) is related to stable samples that are classified as unstable with a probability greater than 55% (UE) since even in the stable situation, vibrations occur when the cutting tool edge enters and exits the workpiece. These vibrations are classified as unstable even if the system is normally operating. In general, results obtained at X and Y directions are satisfying and more than 97% of testing samples are well classified (normal or chatter) with an accuracy greater than 55% either using RBF or MLP. Fuzzy samples, which are presented in percentages of number of stable and unstable samples (UE) are about 4% at X and Y direction using RBF method. In other words, 97% of testing samples without the 4% of fuzzy samples are well classified. In order to confirm that every step in our proposed classification methodology is critical in providing accurate results, classifications are repeated in the absence of some steps.
Discussions

In this part, additional investigations are conducted in order to evaluate the proposed methodology. For this reason, three scenarios are considered. They are as follows:

1- The proposed approach as presented in figure 9.
2- The approach in absence of ranking/selection steps.
3- The approach in absence of processing and ranking/selection steps.

In the second scenario, all the extracted features from signal envelope are classified directly without ranking and selection, while in the third scenario, features are generated from collected data in the absence of processing (filtering and envelope analysis) and ranking/selection stages. These features are then classified with RBF and MLP. The three scenarios are applied to the milling machine for the chatter detection and their classification rate errors at different directions are displayed in Fig. 11.

![Figure 11](image-url) Classification rate error of milling machine in different scenarios and at various directions (X, Y, Y*).

The results confirm that the first scenario (our proposed approach) provides best chatter detection by utilizing RBF or MLP regardless of both rpm and depth of cut. The classification results by using RBF show that the second scenario is better than the third one and this is logically true because in the third scenario (in addition to the absence of ranking/selection step), the processing step is not regarded. In all these scenarios (1, 2 and 3), the data collected from X and Y direction are better from those collect in Y* direction. In the first scenario, RBF is better than MLP in X and Y direction while in the second and third scenarios, the opposite is true. The results illustrate that by using the proposed approach, more than 98 % of testing samples (normal and chatter) are well classified with accuracy greater than 55%. This reflects how important the processing and ranking/selection stages are.

Conclusion

This paper presents a new methodology for chatter detection in CNC milling machines based on signal analysis and using neural networks. The detection and monitoring algorithm is implemented in slot milling process where various Aluminum alloys and cutting conditions are used. It is well known that chatter phenomenon is linked to the dynamic behavior of the machine-tool-workpiece system. As chatter occur around resonances frequencies of CNC machine, a multiband filtering resonance is therefore proposed in the processing step. The process is followed up by an envelope treatment. This allows for increasing the signal to noise ratio and increasing the sensitivity of generated features. Features are then ranked according to their capability to separate between normal and chatter
phenomenon. Only best features are selected and classified by mean of two non linear classifiers that are the Radial Basis Function (RBF) and the Multi-Layer Perceptron (MLP). Manipulations are done at different depths of cut and various rotational speeds; moreover data is collected from three different positions. Classifications are done regardless of depth of cut and of rotational speed. Several techniques are investigated and the results show that the proposed technique offers good chatter detection wherever data is collected from. Although signals generated by CNC milling machine are non-linear and non-stationary, the results confirm how accurate the proposed methodology is. For future work, cyclostationarity properties (Lamraoui, 2012) and support vector machines (Komgom et al., 2007) will be investigated in the processing and classification stages respectively.

Acknowledgments

The support of NSERC (Natural Sciences and Engineering Research Council of Canada) and the Region Rhône-Alpes, through Research Cooperative grants are gratefully acknowledged. The authors would like to thank the École de technologie supérieure’s Products, Processes, and Systems Engineering Laboratory (P²SEL) in Montréal and Victor Songméne for their technical collaboration.

References


