ONLINE TOOL WEAR MONITORING IN TURNING USING TIME-DELAY NEURAL NETWORKS

Bernhard Sick

University of Passau, Chair of Computer Architectures (Prof. Dr.-Ing. W. Grass),
Innstr. 33, D – 94032 Passau, Germany (email: sick@fmi.uni-passau.de)

ABSTRACT

Wear monitoring systems often use neural networks for a sensor fusion with multiple input patterns. Systems for a continuous online supervision of wear have to process pattern sequences. Therefore recurrent neural networks have been investigated in the past. However, in most cases where only noisy input or even noisy output patterns are available for a supervised learning, success is not forthcoming. That is, recurrent networks don't perform noticeably better than non-recurrent networks processing only the current input pattern like multilayer perceptrons. This paper demonstrates on the basis of an application example (online tool wear monitoring in turning) that results can be improved significantly with special non-recurrent feedforward networks. The approach uses time-delay neural networks which consider the position of a single pattern in a pattern sequence by means of delay elements at the synapses. In the mentioned application example, the average error in the estimation of a characteristic wear parameter could be reduced by about 24.2% compared with multilayer perceptrons.

1. INTRODUCTION

Online tool monitoring in turning has to deal with the detection of collisions, the identification of tool breakage and the determination of a tool's wear. The latter task is the most difficult in this area, but the possible economic advantages are important: on the one hand tool costs can be reduced with a good exploitation of the tool's lifetime and on the other hand products with a higher surface quality can be produced exchanging worn tools in time. Without using monitoring systems, tools have to be exchanged precautionary depending on the operator's observations and experience. Commercial systems have some serious disadvantages like causing false alarms and many reactions not being transparent to the operator. Scientific approaches use neural networks, fuzzy systems or combinations of both (see e.g. [1, 3, 4, 5, 6, 7, 10, 15, 17]). However, due to insufficient generalization capabilities or simply a lack of precision even promising methods are not marketable up to now.

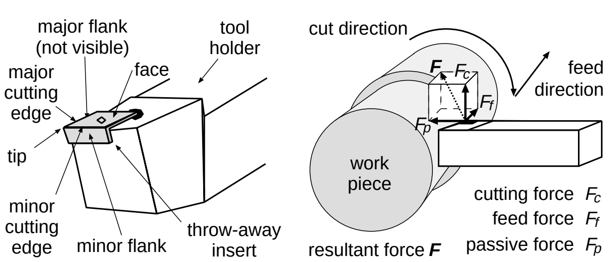

Figure 1: Tool holder with insert and cylindrical turning process

A detailed description of metal cutting processes is given for example in [12]. Modern NC-lathes usually provide tools with throw-away inserts attached on a rotatable turret (see fig. 1, left side). In this paper, results are given for cylindrical turning processes, the most frequently used process type (see fig. 1, right side). The wear of an insert can be classified into different types (e.g. fractures, diffusion, erosion or abrasion) or manifestations (e.g. crater wear or flank wear) [12]. The parameter ww describing the width of wear land at the major flank mainly caused by abrasion is usually considered as an expressive quantity for the overall wear state of an insert (see fig. 2). Values up to 2.6 mm occured in our experiments.

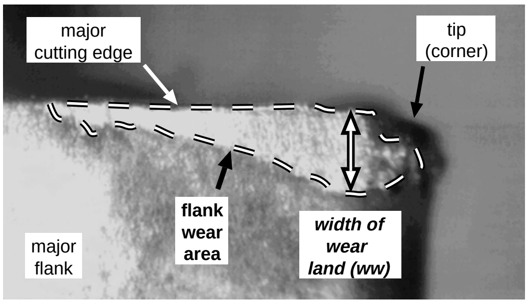

Figure 2: Definition of the wear parameter ww

Neural Networks used for a continuous supervision of wear have to process pattern sequences. Therefore recurrent networks have been investigated by some authors (see e.g. [3, 5, 15]). However, in most cases where only noisy input or even noisy output patterns are available for a supervised learning, success is not forthcoming. That is, recurrent networks don't perform noticeably better than non-recurrent networks processing only the current input pattern like multilayer perceptrons. This paper shows how the continuous estimation of the wear parameter ww can be improved significantly using special non-recurrent feedforward networks which consider the position of a single pattern in a pattern sequence by means of delay elements at the synapses. Comparable investigation are not known (the 'time-delay networks' used in [3] are special recurrent networks).

2. INPUT AND OUTPUT PARAMETERS

Signals from sensors in machine tools are disturbed for many reasons: outbreaks at cutting edges, chatter (i.e. self-exited vibrations), sensor nonlinearity, noise of digitizers etc. Only by means of a multisensor data fusion technology it is possible to determine the tool's wear (although there are additional disturbances e.g. in form of crosstalk effects between sensor channels). Therefore three sensor elements (piezo-electric) for the measurement of forces in the three orthogonal directions have been used (sampling rate 10 kHz). With this sensor system, a large number of experimental cylindrical turning processes with more than 30 inserts has been carried out. The wear parameter has been determined periodically by means of a microscope. It must be noticed, that ww could be determined with a precision of about 5 μm only and that a removal of the insert in order to measure ww can noticeably disturb any of the three forces in the following cut. Five static and dynamic process parameters (type of the insert (coating and substrate), depth of cut, feed rate, cutting speed, workpiece diameter) have been varied in these experiments. Other parameters, e.g. those which describe the tool geometry (corner radius, clearence and cutting angle, cutting edge inclination etc.) or the workpiece material (steel Ck45), have been identical in all experiments.

Output parameter of the neural network is the width of wear land ww; *input parameters* are the average values of the three forces in a certain short time window. Before computing these inputs, the force signals are aligned with respect to a force model. For this reason, each measured force value is divided by a correction factor which has been derived from a physical / mathematical model of machining processes with defined cutting edges originally developed by KIENZLE, VICTOR et al. (see e.g. [9]). Compared with neural networks using 'non-aligned' force signals and process parameters as additional inputs, the precision of the estimation could be improved by about 15.8% with pre-processed forces only [14].

The problem is now to find a relationship between extremely noisy input data (see above) and output data which also can be called 'noisy', because ww describes the overall wear in a quite simplified way and cannot be determined exactly. However, based on a sufficient number of training patterns, neural networks are able to ignore disturbed or noisy information, to detect fundamental interdependencies and to approximate a sought non-linear function (see e.g. [18]).

3. NEURAL NETWORK PARADIGMS, STRUCTURE AND TRAINING

Time-delay neural networks (TDNN) are non-recurrent networks which use more than one connection between two nodes in successive layers [16, 8]. Each connection is able to delay the propagation of values and has its own weight (see fig. 3, a delay element D_{X} delays its input by x time steps). Therefore the inputs to a node consist of the outputs of previous nodes not only during the current time step t, but during some previous time steps as well (usually but not necessarily a time sequence without gaps which is mostly identical for all connections between two layers of nodes, e.g. $t, t-1, ..., t-N$). The output (or activation) of a node i is consequently given by

$$y_i(t) = A\left(\sum_{a=1}^{M}\sum_{b=0}^{N} y_a(t-b)\cdot w_{a,i,b}\right),$$

where $y_i(t)$ is the output of node i at time t, $w_{a,i,b}$ is the weight of a connection between node a in layer x and i in layer $x+1$ with time delay b, and A is an activation function.

With the given definition multilayer perceptrons (MLP) are a subclass of TDNNs without 'real' delays (i.e. $N = 0$). Another subclass of TDNNs with delays only at the input nodes can be shown – on the assumption of certain start conditions for a pattern sequence – to be equivalent to multilayer perceptrons with a sliding window (MLP-sw) [18]. The mentioned subclass of TDNNs uses an internal representation of the temporal information, whereas the MLP-sw networks need input patterns which have been delayed externally.

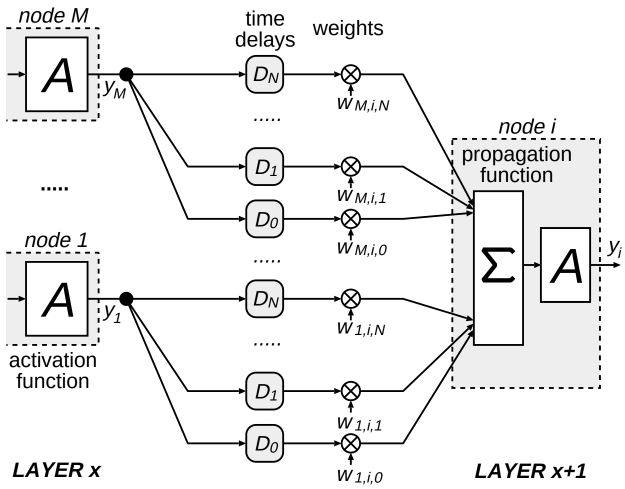

Figure 3: Time-delay neural networks

Networks with one hidden layer (12 nodes, fully connected) have been trained with the resilient backpropagation (RPROP) supervised learning algorithm (batch learning, 50000 epochs) [11]. The activation function has been the non-linear sigmoid activation function. The experiments in the following section compare MLP, TDNN and MLP-sw networks with different delays. In previous investigations it has been shown that MLPs perform better than Classifying SOMs, FuzzyARTMAPs and NEFCLASS networks [13].

4. RESULTS

To assess the generalization capability of a trained network, results for learning (training) patterns have been compared with results for unknown test patterns ('extrapolation'). In order to demonstrate the ability to reproduce the result of a training starting with randomly initialized weights, each experiment has been repeated 25 times. The mean average error ('mean' refers to the 25 repetitions and 'average' to the set of test patterns in one experiment), the standard deviation divided by this mean average error, the minimum and maximum average error and the median of the 25 average errors have been determined.

It must be noticed, that the test patterns belong to three chipping experiments with combinations of process parameters which lie in an area of the paramenter space covered quite well with chipping experiments. For the available data this measure turned out to be necessary as a result of the uneven and sparse distribution of parameter combinations in the parameter space. That's why results for test patterns are better than results for training patterns. However, the relative improvements with different network paradigms and structures are obvious and the conclusions are valid in any way.

Table 1 gives the estimation results for the continuous estimation of ww. In each experiment the number of learning and test

experiment number	1		2		3		4		5	
neural network configuration										
paradigm	MLP		TDNN		TDNN		MLP-sw		MLP-sw	
structure	$3 \xrightarrow{1} 12 \xrightarrow{1} 1$		$3 \xrightarrow{2} 12 \xrightarrow{3} 1$		$3 \xrightarrow{2} 12 \xrightarrow{2} 1$		$3 \cdot 3 \xrightarrow{1} 12 \xrightarrow{1} 1$		$3 \cdot 2 \xrightarrow{1} 12 \xrightarrow{1} 1$	
number of weights	48		108		96		120		84	
average errors										
learning / testing	L	T	L	T	L	T	L	T	L	T
mean average error $\mu_\varnothing$ in μm	95.26	54.10	78.04	43.51	79.48	43.71	78.09	45.15	82.65	45.25
standard deviation $\sigma_\varnothing$ / $\mu_\varnothing$ in %	1.71	2.27	2.53	3.18	2.20	2.15	2.54	3.11	1.45	1.74
minimum average error $\min_\varnothing$ in μm	92.55	51.88	74.46	40.50	75.46	41.46	74.52	43.45	80.36	43.95
maximum average error $\max_\varnothing$ in μm	97.28	55.45	81.36	46.85	83.11	45.37	85.68	50.36	85.54	46.97
median of the average errors $\text{med}_\varnothing$ in μm	96.47	54.78	77.84	43.55	79.64	43.77	77.65	45.01	82.60	45.16

Table 1: Estimation results for the continuous estimation of ww

patterns has been 769 and 225, respectively. The description of the structure of a network has to be interpreted as follows: $3 \xrightarrow{3} 12 \xrightarrow{3} 1$ is a network with 3 inputs, 12 hidden nodes and 1 output, and $\xrightarrow{3}$ represents delays of 0, 1 and 2 from one layer of neurons to the next. In the case of a MLP-sw, $3 \cdot 4$ inputs means 12 inputs divided into 3 sliding windows of length 4.

Experiment 1 shows the results for a common MLP which doesn't consider pattern sequences. The best result (regarding the test patterns) with a TDNN is given in experiment 2. This network uses delays of 2 and 3 in the first and second layer of weights, respectively. The first 'runner-up' is presented in experiment 3. In both cases, the mean average error for test patterns is about 20% better compared with the MLP in experiment 1. The best results with a MLP-sw are given in experiment 4 and 5. As a remark, a fully connected TDNN with 3 inputs, 12 hidden nodes, 1 output and equal delays in each layer of weights operating on a receptive window of length x has only $w_{\text{TDNN}} = 24 \cdot x + 24$ weights, whereas the corresponding MLP-sw (i.e. operating on the same receptive window) has $w_{\text{MLP-sw}} = 36 \cdot x + 12$ weights.

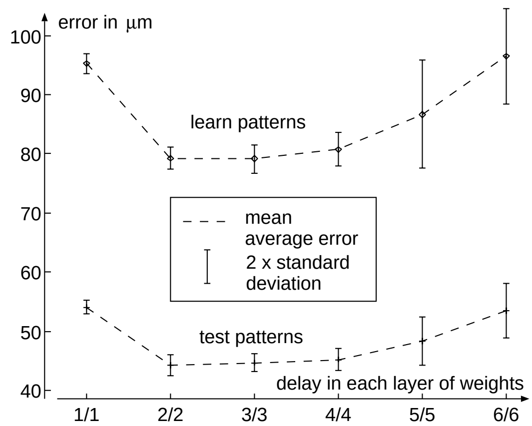

Figure 4: Results with different number of delays (TDNN)

Fig. 4 shows the results of experiments using TDNNs with different delays (two layers of weights with equal delay, i.e. $3 \xrightarrow{x} 12 \xrightarrow{x} 1$). With increasing delays the networks are not able to learn the input/output relationship correctly. However, the optimum in figure 4 is quite wide. Fig. 5 presents comparable results for a MLP-sw ($3 \cdot y \xrightarrow{1} 12 \xrightarrow{1} 1$). With an increasing length of the sliding window, the network begins to 'overfit' (the average error for learning patterns decreases, whereas the average error for test patterns and particularly the standard deviation increases significantly). For example, a MLP-sw with a sliding window of length 7 (264 weights) shouldn't be used for the estimation. The corresponding TDNN with delays 4/4 (192 weights) operating on the same receptive window, however, performs quite well. Comparing fig. 4 and fig. 5, it can be stated that for the TDNNs with equal delays in the two layers (in contrast to the MLP-sw networks) the optimum for learning and test patterns is the same. Therefore it is quite easy to decide on a certain network structure.

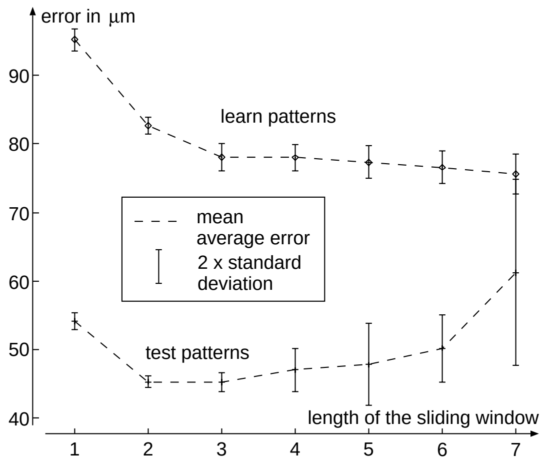

Figure 5: Results with different sliding windows (MLP-sw)

An interesting question in the area of neural networks is, whether a variation of the number of nodes in a hidden layer changes the results significantly. Fig. 6 shows some results with a TDNN ($3 \xrightarrow{3} z \xrightarrow{3} 1$). Using between 6 and even 43 nodes in the hidden layer leads to almost the same mean average error.

In any of the described experiments test patterns with values for ww up to 1.5 mm have been evaluated so far. However, such worn inserts wouldn't be used in a chipping process. In accordance with ISO recommendations [2], a wear limit of 0.5 mm is usually considered sensible. Taking this limit into account, the mean average error for test patterns in experiment 2 is only 31.79 μm which is an improvement of about 24.2% compared to 41.96 μm in experiment 1 under the same conditions. Tab. 2 shows the mean identification rates of test patterns ($ww \leq 0.5$ mm) with a given acceptable maximum error and the mean maximum error with the corresponding standard deviation for the experiments 1, 2 and 4.

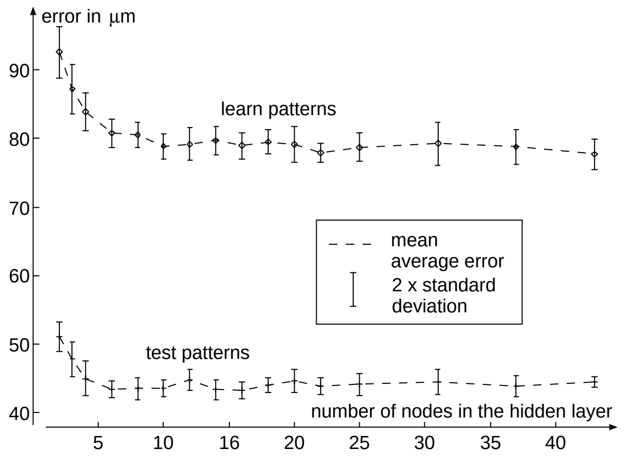

Figure 6: Different number of nodes in the hidden layer (TDNN)

experiment number	**1**	**2**	**4**
mean identification rates			
error $\leq 10\ \mu m$ in %	9.71	21.37	19.62
error $\leq 25\ \mu m$ in %	30.75	47.44	46.47
error $\leq 50\ \mu m$ in %	68.06	79.64	77.69
error $\leq 75\ \mu m$ in %	88.60	93.45	92.42
error $\leq 100\ \mu m$ in %	96.08	98.00	97.23
error $\leq 200\ \mu m$ in %	100.00	100.00	100.00
maximum errors			
mean max. error $\mu_{\max}$ in μm	143.65	127.45	132.70
std. dev. $\sigma_{\max}$ / $\mu_{\max}$ in %	1.40	5.70	12.14

Table 2: Identification rates and maximum errors

5. CONCLUSIONS AND OUTLOOK

As a general result, it can be concluded that neural networks are an excellent method to estimate the wear of turning tools provided that the input signals are aligned with respect to a force model (about 15.8% increase of the precision, described in [14]) and time-delay neural networks are used which consider the position of a single pattern in a pattern sequence by means of delay elements at the synapses (about 24.2% additional increase of the precision, described here). Slight modifications of delays and the number of neurons in the hidden layer don't change the results significantly. Comparable investigations with TDNNs and/or the mentioned preprocessing measures are not known. The ideas and solutions presented in this paper could be transfered to other online wear monitoring problems, particularly problems which have to cope with extremely noisy data, e.g. other machining processes using tools with defined cutting edges (e.g. milling or drilling).

Our actual and future research also deals with tests of other learning algorithms and the use of energy parameters as inputs of the networks. Genetic algorithms will be used to find the optimal structure of a neural network and the optimal time delays. Additionally, we investigate variations of the process model in order to improve the pre-processing step.

6. REFERENCES

[1] D. Barschdorff, L. Monostori, T. Kottenstede, G. Warnecke and M. Müller, "Cutting tool monitoring in turning under varying cutting conditions; an artificial neural network approach", in *Proc. of the 6th Int. Conf. on Industrial & Eng. Appl. of Artif. Intelligence & Expert Systems*, 1993.

[2] G. Boothroyd and A.W. Knight, *Fundamentals of Machining and Machine Tools*, Marcel Dekker Inc., New York, 1989.

[3] S.T.S. Bukkapatnam, S.R.T. Kumara and A. Lakhtakia, "Fractal estimation of flank wear in turning using time-delay neural networks", in *Intelligent Eng. Systems through Artificial Neural Networks*, C.H. Dagli et al. Eds. ASME Press, New York, 1994.

[4] S. Elanayar and Y.C. Shin, "Robust tool wear monitoring using radial basis function neural networks", in *J. of Dynamic Systems, Measurement and Control*, vol. 117, dec., 1995.

[5] S.V. Kamarthi, *On-line tool wear estimation in turning through sensor data fusion and neural networks*, PhD thesis, Pennsylvania State Univ., 1994.

[6] C.S. Leem, D.A. Dornfeld and S.E. Dreyfus, "A customized neural network for sensor fusion in on-line monitoring of cutting tool wear", in *J. of Eng. for Industry*, vol. 117, may, 1995.

[7] S. Li and M.A. Elbestawi, "Tool condition monitoring in machining by fuzzy neural networks", in *J. of Dynamic Systems, Measurement and Control*, vol. 118, dec., 1996.

[8] D.-T. Lin, J.E. Dayhoff and P.A. Ligomenides, "Trajectory production with the adaptive time-delay neural network", in *Neural Networks*, vol. 8 (3), 1995.

[9] E. Paucksch, *Zerspantechnik*, Friedrich Vieweg & Sohn Verlagsges., Braunschweig, 9th ed., 1992.

[10] M. Rahman, Q. Zhou and G.S. Hong, "Application of kohonen neural network for tool condition monitoring", in *Proc. of SPIE*, vol. 2620, 1995.

[11] M. Riedlmiller and H. Braun, "A direct method for faster backpropagation learning: The rprop algorithm", in *Proc. of the Int. Conf. on Neural Networks*, vol. 1, 1993.

[12] M.C. Shaw, *Metal Cutting Principles*, Oxford University Press, Oxford, 1989.

[13] B. Sick, "Classifying the wear of turning tools with neural networks", in *Proc. of the 7th Int. Conf. on Artificial Neural Networks (ICANN '97)*, 1997.

[14] B. Sick, "Wear estimation for turning tools with process-specific pre-processing for time-delay neural networks", in *Smart Eng. Systems Design: Neural Networks, Fuzzy Logic, and Evolutionary Programming*, C.H. Dagli et al. Eds. ASME Press, New York, 1997.

[15] I. Suwalski and H. Weber, "Mit neuronalen Netzen die Überwachungsgüte verbessern", in *ZWF (Zeitschrift für wirtschaftlichen Fabrikbetrieb)*, vol. 90 (10), 1995.

[16] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano and K.J. Lang, "Phoneme recognition using time-delay neural networks", in *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 37 (3), 1989.

[17] G. Warnecke and M. Müller, "Application of artificial neural networks for the identification of cutting processes", in *Annals of the German Academic Society for Production Eng.*, vol. 3 (1), 1994.

[18] A. Zell, *Simulation Neuronaler Netze*, Addison-Wesley Publishing, Bonn, 1994.