A Multi-Bias Recurrent Neural Network for Modeling Milling Sensory Data

Guodong Wang*, Mohamed Amin Ben Sassi† and Radu Grosu*
*Institute of Computer Engineering, Vienna University of Technology, Austria
†Department of Computer Systems Engineering, Mediterranean Institute of Technology, Tunis, Tunisia
{guodong.wang, radu.grosu}@tuwien.ac.at
mohamed.bensass@medtech.tn

***Abstract*—Modeling high dimension sensory data is a key issue for Cyber-Physical Manufacturing Systems especially for milling process due to: (a) Sophisticated characteristics of input signals and (b) The complex procedure of processing sensory data. In this paper, we provide an End-to-End data modeling platform i.e., a multi-bias randomly connected recurrent neural network that makes use of recurrent structure and multi-bias to achieve efficient and accurate modeling performances. In order to tune the parameters of the proposed recurrent neural network (RNN), we apply a sampling method called Zoom-In-Zoom-Out (ZIZO) that helps RNN to quickly find a set of appropriate weights. We apply our technique to an empirical data set collected from NASA data repository and show that our method provides more precise and efficient results than existing methods.**

I. Introduction

Milling is widely used for shaping metals in industries. Various factors e.g., depth of cut, types of materials, vibration and acoustic signals and so forth seriously effect the sharp degree of milling tools and lead to unqualified products. Exploring the relation among tool wear states and various factors is essential for improving production quality and reducing cost. With the development of Cyber-Physical Systems technology, manufacturers can easily set up amount of sensors around working platforms and monitor the health status of production systems. When massive data, generated by Cyber-Physical Production Systems, flows into database, how to effectively analyze gathered data is becoming seriously.

A batch of methods have been proposed for modeling sensory data. Generally, existing solutions can be classified into two categories i.e., model based methods and data-driven methods [1]. Zhu et al. [2] propose a morphological component based region growing algorithm to optimize the image analysis of tool-wear. Garcia Nieto et al. [3] and Li et al. [4] combine particle swarm optimization (PSO) with support vector machines (SVM) developing a practical hybrid regression model to predict milling tool. Benkedjouh et al. [1] and Kimotho et al. [5] provide SVM based methods for monitoring machining processes. The tool wear prediction model is constructed by learning the correlation among Principle Component Analysis (PCA) method extracted features and actual tool wear status. Hong et al. [6] suggest to wavelet packet transforms and Fishers linear discriminant to model the tool wear process in micro-milling and estimate the tool wear states by given cutting force features. Later on hidden Markov Model (HMM) is adapted to determine the tool states. Chen et al. [7] suggest to use Gaussian Mixture HMM Models select application sensitive features, then feed extracted features to neural network and model the tool wear states. The similar work has also been done by applying Bayesian network [26]. Pati et al. [8] suggest to use wavelet transform to analyze vibration signals and take the advantage of artificial neural networks to classify last steps generated features.

Neural network based methods are promising for industrial data analysis and predictive maintenance [9, 10, 33]. Hasani et al. [9] provide an Autoencoder based method modeling a bearing data set with 20480 input-dimension of every instance and automatically extracting essential features (i.e health indicators) [10]. Bangalore et al. [11] and Drouillet et al. [12] describe feed forward neural networks based methods for monitoring gearbox bearing. Whereas, Guo et al. [13] present a recurrent neural network based technique for predicting the health indicator of bearing machines. Corne et al. [14] apply neural network for tool wear detection in real time. Xiao et al. [15] propose to use recurrent neural networks model intensity function of a point process. It proves that the issue of randomly generating time steps in predictive maintenance could be solved by RNN.

Jaouher et al. [16] combine Simplified Fuzzy Adaptive Resonance Theory Map Neural Network (SFAMNN) with Weibull distribution. SFAMNN is capable of processing time series data, which makes it work perfectly for modeling bearing degradation in time sequence. Addona et al. [17] mix the DNA-based computing technique with artificial neural network for tool wear pattern recognition.

The performance of other methods e.g., genetic algorithms, evolution algorithms, statistical analysis based methods, hybrid neural networks models and so forth, are also demonstrated by experimental simulations and industrial applications [18–20, 22–25, 27, 28].

As aforementioned, a mount of neural networks based techniques applied in industry. In order to improve the modeling performance of neural network based approaches, existing techniques mostly take advantages of optimization methods and feature extraction algorithms. Even though these methods work fine in practical applications, but the problem is when we try to solve the original problems, the applied optimization methods introduce additional hyper parameters which needed

978-1-5386-6531-2/18/$31.00 ©2018 IEEE

to be solved as well e.g., the number of principle components, the number of neighbors for K-means algorithm. In summary, the main drawback of these mentioned techniques is the difficulty of implementation, as plenty of parameters of models and optimization methods need to be adjusted for different scenarios. To tackle this issue, we propose a multi-bias randomly connected recurrent neural network (MBRC-RNN) for modeling multi-sensor signals collected from Cyber-Physical Production Systems.

MBRC-RNN provides an End-to-End systems modeling and tool-wear degree prediction service. The only effort for applying MBRC-RNN is changing the size of the hidden layer, based on different application scenarios. More precisely, we first randomly generate a 3-layer recurrent neuron network constituted by an input layer, a randomly generated hidden layer and a readout layer. In this work, the size of the input layer is 5400 augmented by bias whereas the hidden recurrent layer is randomly connected. Every hidden neuron receives signals from the input layer and other neurons around it at the same time. This type of connection is performed to reduce the size of neural networks. In oder to tackle the over fitting issue, we initialize all neural weights from the range of $[-0.001, 0.001]$. Then, apply the so called ZIZO sampling method for exploring the weights of RNN [30].

Multi-bias is another important property of MBRC-RNN. In addition to the common bias, we also take other inputs as biases e.g., the type of metals, the cutting depth for one operation on metal. These data usually do not get changed in a specific period, so we call them static features in this work. It does not make sense if one merges the static sensory data with quickly changed sensory information e.g., vibration signals, noise signals and so forth. Static features working together with dynamically changed signals finally effect the tool wear status. Hence, in this work, static features are considered as multi-bias of neural networks. After searching process, the readout layer is trained by linearly combining the neural states which is collected from hidden layer, until systems reach predefined conditions. In oder to evaluate the performance of MBRC-RNN, we applied our method on a milling sensory data set collected from NASA data repository [29]. Experimental results demonstrate that our method can precisely and easily model the sensory data generated by complex milling processes.

The main contributions of this paper are as follows: (1) We design a novel neural network for modeling milling process generated data and (2) We apply ZIZO for searching neural weights, instead of using back-propagation based methods, due to the fact that back-propagation is not applicable for complex neural networks in the real world problems. Accordingly, the paper is organized as follows: Section II introduces the structure of MBRC-RNN and describes the steps of applying MBRC-RNN on modeling milling sensory data. In section III, we perform various experiments with applying MBRC-RNN and presenting our results correspondingly. Within Section III, we qualitatively and quantitatively explain the properties of MBRC-RCNN.

II. Multi-Bias Randomly Connected Recurrent Neural Network (MBRC-RNN) Based Milling Sensory Data Modeling Method

In this section, we describe the mechanism of MBRC-RNN for modeling milling process generated sensory data. Figure 1 illustrates the structure of proposed neural network. A MBRC-RNN model is trained by learning milling sensory data. The input u has two parts, one is directly feed into the colorful (blue & red) and randomly connected recurrent layer, the other part is simply copied to the bias $[b_1, ..., b_p]$ (all the dash lines in Figure 1 represent the copy operation). After feeding data into the hidden recurrent layer, we use ZIZO [30] to generate one group of weights for RNN, then applying linear combination to train the readout layer and validating the prediction error E - Mean Squared Error (MSE) between the targets y^{target} and the predicted outputs y. If E reaches predefined threshold, then terminating the program immediately.

The main contribution of the colorful hidden recurrent layer is shrinking the redundant input information. MBRC-RNN is specially designed for merging dynamic and static sensory data. Modern sensors could generate a batch of data in a very short period e.g., 20480 samples in one microsecond generated by a vibration sensor [9]. These data points contain important information of systems status. One can not directly input these data collected from various sensors into a neural network, as the data is corrupted and noisy [2]. Additionally, high dimensions inputs for neural network is highly time consuming. In the followings, we present the mathematical model of MBRC-RNN. Then, introduce the parameters-searching method called ZIZO.

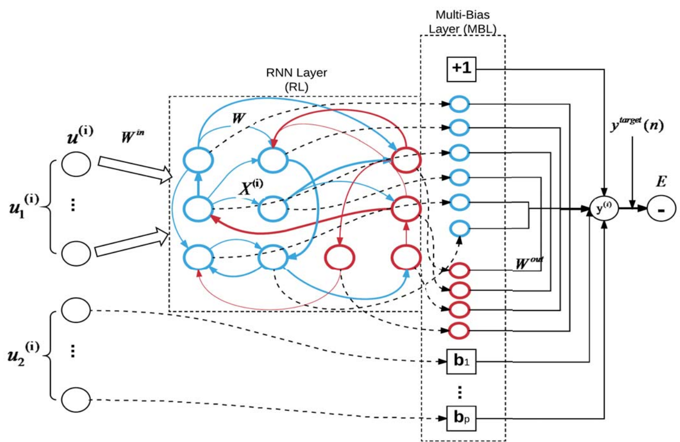

Fig. 1. Multi-Bias Randomly Connected Recurrent Neural Network Structure. Blue neurons stand for excited neurons. Red ones represent inhibited neurons. Bold connections means neurons have stronger impact on others. Thick ones mean less weights values. This figure is only for demonstrating how MBRC-RNN works, it is unnecessary that one places all inhibitory neurons in the right part as depicted in this figure.

A. Formalism of MBRC-RNN

A MBRC-RNN model tries to model sensory data and predict tool wear degree, especially for multi-sensor inputs.

The input signals $u^{(i)} = [u_1^{(i)}, u_2^{(i)}]^T$ (i, the ith instance). u_1 represents different types of sensory data e.g., spindle vibration, acoustic emission and so forth. It contains plenty of critical underlying information of cutters, so it is necessary to process this part of information first. Specifically, mixing the input signals into much lower dimensions and extracting useful information via small size of hidden layer. u_2 is static input for some specific periods. It presents systems' configuration for different scenarios. The output of Recurrent Layer (RL) is governed by Equations 1 and Equation 2.

$$X^{(i)} = f(W^{in} \cdot u(n)^{(i)}) \tag{1}$$

$$x_n = f(W \cdot X^{(i)}) \tag{2}$$

where, $X^{(i)} = \{x_1, x_2, ..., x_n\}$ is the states of RL, x_n is the nth neuron's state. W is the weights of recurrent connection, W^{in} is the weights from input layer to RL. W in Equation 2 is a key component in MBRC-RNN. It is used for obtaining neighbor neurons status. The states of one neuron represent partial information of inputs. f is the activation function, one could use sigmoid function, hyperbolic tangent function or rectified linear unit(Relu). In this work, we use sigmoid as the activation function, because we normalize all inputs into the range [0,1] in following experiments, and the flank tool-wear degree is great or equal to 0 in our study cases. The recurrent connection used here is for efficiently modeling high dimension inputs. We call this type of network as biological recurrent connection, as the way how neurons connect with each other is mimicking the neural networks of animals. It takes the advantage of its neighbor neurons and acquires additional information with less effort compared with other structure e.g., the deep forward structure. Every neuron shares its own information with others via the biological connection. Generally, the benefit of applying biological recurrent connection is that it could improve information presentation capability of every single neuron, and shrink the size of entire neural networks at same time. It is worth to note that if the number of hidden layer neurons greater than some threshold, the neural network will quickly adapt to high accuracy states while the generalization performance is much worse than training. Specifically, it indicates that one adds too many neurons to hidden layer and leads to information-overlapping among hidden neurons.

In order to accurately model the tool-wear status, we need to stack static features together with the outputs of RL. The mathematic operation is formalized as Equation 3. The multi-bias is essential for modeling physical process generated data, as it decides the ground picture of sensory data e.g., type of materials or feed speed are static parameters for systems in one specific period and they represent the background of collected data. Therefore, we model these static properties as multi-bias within the neural network.

$$X_{mbl} = [X^{(i)}; b_1, ..., b_p] \tag{3}$$

The readout layer makes use of the reformed outputs X_{mbl} as inputs. Accordingly, RL layer extracts essential and relative independent information from inputs. Every neurons in the MBL layer represents one key component of entire inputs. Hence, in the readout layer, we apply linear combination technique, specifically ridge regression, for modeling the W^{out}. The outputs are acquired by Equation 4.

$$W_{out} = (R^{(i)} + \alpha^2 \cdot I)^{-1} \cdot P^{(i)} \tag{4}$$

$$R^{(i)} = (X_{mbl}^{(i)})^T \cdot X_{mbl}^{(i)} \tag{5}$$

$$P^{(i)} = (X_{mbl}^{i})^T \cdot (y^{(i)})^{target} \tag{6}$$

where, α^2 is a non-negative constant value. We set it as $1.0e{-8}$ in this work. I is an identity matrix. R is the correlation matrix of X_{mbl}. P is the cross-correlation matrix between X_{mbl} and the desired outputs.

Finally, in order to terminate searching and training process, we measure the MSE and stop the program when it lesses than predefined threshold. MSE is calculated as Equation 7.

$$E(y, y^{target}) = \frac{1}{M} \sum_{i=1}^{M} (y^{(i)} - (y^{(i)})^{target})^2 \tag{7}$$

$$y = g(W_{out} \cdot X_{mbl}) \tag{8}$$

where, M is the number of instances, y is the predicted outputs. g is the readout activation function. One should select it based on practical applications. In this work, we choose linear function $g(X) = X$, because the tool-wear degree of some experimental cases is great than 1, so it is not going to work if one applies sigmoid or tangent function as the activation function of readout layer. Another possible option is using Relu while it is not efficient at all since we already know all the damage-degrees great than or equal to 0. Therefore, applying linear function for the readout layer is reasonable.

B. Searching Appropriate Values for Input Weights W^{in} & RL Weights W

Training complex RNN is a tough work, especially for a completely random connected recurrent neural network. Potentially, we have three possible solutions for finding a set of appropriate weights for MBRC-RNN i.e., gradient search with back propagation, completely random generated weights which works for Echo-State Networks [31], or one can apply the technique called Zoom-In-Zoom-Out searching method for RNN.

Gradient search based back propagation algorithm is highly time consuming for complex recurrent neural networks as the main idea is calculating accumulated errors. Firstly, one needs to feed forward inputs through entire neural network. Then, computing the errors between y and y^{target}. Following applies derivative operations and back propagation techniques on the neural network. The structure of MBRC-RNN's RL layer is similar to Echo State Networks (ESNs) while MBRC-RNN does not have memory of historical inputs, compared with ESNs. It does not have wash out operations either [31]. Those make MBRC-RNN significantly different from ESNs. However, we do use the principle of readout layer and train

TABLE I
THE QUALITATIVE COMPARISON AMONG BACK-PROPAGATION BASED GRADIENT SEARCH, RANDOM METHOD AND ZIZO. ++: HIGHLY SATISFIES. +: SATISFIES. −: MUCH LESS SATISFIES.

	Gradient Search	Random Search	ZIZO
Complexity	++	-	+
Time Consuming	++	-	+-
Accuracy	+	-	+-

the output layer by linear combination. Randomly generating W^{in} and W works good for ESN. In this work, we take the same idea and randomly initialize W^{in}, W, and train W^{out} by Equation 4. Usually it takes long time to adjust hyper-parameters of ESNs and MBRC-RNN through manually tuning or grid searching methods[32]. Especially, the manual tuning requires users equipped with professional experience of adjusting hyper-parameters for neural networks.

ZIZO is a heuristic random searching method [30]. The principle of it is, firstly, generating a batch of possible weights for neural networks in predefined ranges. Since we do not have enough information for predicting the probability distribution of neural weights, so we first randomly generate the weights based on uniform distribution. Secondly, go through all candidates, if E reaches the threshold, then terminate the program. Otherwise, re-generate new ranges based on the sub-optimal results generated by previous steps i.e., regarding these points as centers of possible solutions and re-generating a new box, accordingly. The final step is repeating aforementioned two steps until the program hits predefined conditions. In order to avoid th over fitting issue for neural networks, we define one of the stop conditions as $E_{generalization} \geq 3 * E_{training}$ when ZIZO reaching a group of weights. Here, $E_{generalization}$ and $E_{training}$ are the prediction and training errors, respectively. The qualitative comparison among gradient search, random method and ZIZO is presented (Table I). One can apply gradient search based back-propagation training algorithm while it needs to conduct derivative operations on neurons states among different hidden layers. In order to update every neuron's weights, back-propagation is applied on entire neural network in the form of Jacobian matrix. With larger neural network, the calculation complexity increase exponentially and usually it is time expensive. Random searching algorithm does not have any logic inside, the only effort is keeping randomly generated samples fall into predefined probability density distribution. ZIZO is a heuristic way for searching appropriate weights of MBRC-RNN in this work. It does not have any derivative operations. The core idea of ZIZO for updating neural weights is evolving systems based on searching results, generated by its previous step i.e., if current training performance getting better, then applying the weights generated by this trial and re-generating a new solution space. Otherwise, keeping previous step generated weights and reproduce another group of possible weights. In the following experiments, one can see that the size of hidden layer is much smaller than input layer. In order to tackle over shooting issue (i.e the outputs of neurons always equal to 1 or 0), it is necessary to limit the searching space of neural weights into a small range.

One of the disadvantage of ZIZO is that it can not guarantee the global optimal states of neural networks while the fast searching speed saves plenty of time for users and allows us to do more trials for searching optimal solution.

III. EXPERIMENTS WITH MBRC-RNN

In this section, we evaluate the performance of our proposed model by employing it on a milling data set collected from NASA data repository [29]. We first explain the data set and related experiments setup. Then, we illustrate the performance of MBRC-RNN by modeling the complex milling sensory data. Finally, we compare our results with existing work and present the difference.

A. Milling Tool Wear Data Set

The data set is collected from a milling machine under various operating conditions. The milling tool wear is measured in a regular cut as well as entry cut and exit cut. The sampled data is gathered by three different types of sensors i.e., acoustic emission sensor, vibration sensor and current sensor. Signals from all sensors are amplified and filtered firstly, then fed through two Root-Mean-Square operation before saved to industrial computer. Signals from a spindle motor current sensor are fed into the computer without further processing. Some other factors may effect the tool wear are also collected i.e., duration of experiment, depth of cut and material. There are 16 cases with various number of runs. The number of runs dependents on the degree of flank wear which is measured among runs at irregular intervals, up to the wear limit. The basic setup encompasses the spindle and the table, which are provided by Matsuura machining center MC-510V. The matrix of parameters chosen for the experiments is guided by industrial applicability and recommended manufacturers-settings. For the details of experimental setup, one can reference the work [3].

B. Data Preprocessing

Figure 2 depicts the contour of normalized 6 types of sensory data. X-axis represents different runs, Y-axis stands for different instances in different study cases. One can clearly see that spindle motor current signals and spindle vibration signals are very noisy and corrupted. How to fuse these massive data into single neural network and provide an End-to-End service is a crucial challenge. In this work, MBRC-RNN is designed for such scenarios. In order to find appropriate weights for MBRC-RNN, we firstly map sensory data into [0, 1], which is collected from same sensors except flank wear VB. Flank wear is used as the indicator of health status of cutters. 0 and 1 as inputs fed into neural network could introduce fluctuation to final outputs. Hence, we reset all the inputs with the value of 0 to 0.01 and replace 1 to 0.99. Additionally, we simply delete instances marked with NaN, so there are totally 145 instances selected for our experiments. The way of mapping the data into range [0, 1] is presented as Equation 9.

$$Y = (Y_{max} - Y_{min}) \cdot (X - X_{min}) / (X_{max} - X_{min}) + Y_{min} \quad (9)$$

where Y_{max} and Y_{min} are user defined up bounder and lower bounder, respectively. X_{max} and X_{min} are the maximum and minimum values of collected signals.

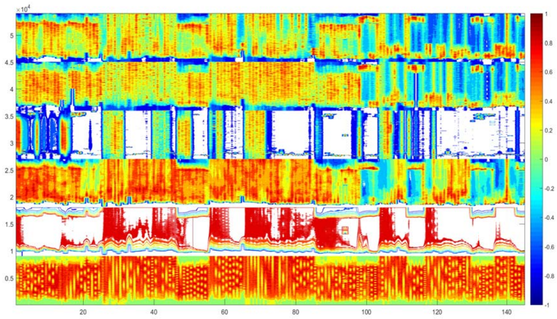

Fig. 2. Contour of normalized experimental data set. Horizontal axis is the index of various runs, vertical axis constituted by spindle motor current, spindle motor current, table vibration, spindle vibration, acoustic emission from table and spindle, respectively.

To demonstrate our method, we select 6 different study cases (Figure 3). One can observe that these 6 different study cases basically stand for 6 types of milling tool wear trend. The experimental goal is applying MBRC-RNN on these sensory data and modeling cutters' tool-wear status.

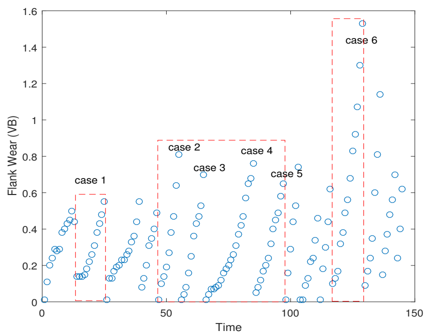

Fig. 3. 6 different study cases. Each case represents one independent milling process under various operation conditions.

C. Experimental Results

We implemented our MBRC-RNN model in MATLAB, as it is widely used in modeling industrial manufacturing systems. The entire experimental process is performed on a Microsoft Azure NC-Series virtual machine powered by one NVIDIA Tesla K80 GPU. We demonstrate our method's functionality in three aspects where firstly we train the MBRC-RNN model with part of the data, the obtained result is presented in high accuracy. In the second aspect, we model the complex milling process with limited a mount of neurons in hidden layer.

TABLE II
EXPERIMENTS WITH DIFFERENT MBRC-RNN STRUCTURES((INPUT NEURONS)-(HIDDEN NEURONS)-(OUTPUT NEURON)). TRAINING AND PREDICTION ERRORS ARE CALCULATED BY MEAN-SQUARE-ERROR AS EQUATION 7

Case No.	Structure	Training Err	Prediction Err
Case 1	$54001 - 11 - 1$	$2.09e - 4$	$3.38e - 4$
Case 2	$54001 - 12 - 1$	$0.439e - 4$	$6.38e - 4$
Case 3	$54001 - 9 - 1$	$1.46e - 4$	$7.78e - 4$
Case 4	$54001 - 20 - 1$	$3.54e - 4$	$4.9e - 3$
Case 5	$54001 - 11 - 1$	$9.14e - 4$	$1.61e - 4$
Case 6	$54001 - 17 - 1$	$2.1e - 4$	$6.56e - 4$

Third, MBRC-RNN could be a generic modeling method of milling process under various operation conditions. Experimental results demonstrate that, with much less neurons in RL layer, MBRC-RNN could achieve high prediction accuracy. The consumed time for ZIZO-searching and neural network training totally lesses than 20 minutes for every experiment.

a) Results: We discretize the deterioration trend of tool-wear in experiments and treat every instance as an independent run. Randomly selecting 80% of experimental data for training, the remaining part is applied for testing. Figure 4 - 9 depict the correlation efficient between prediction and desired outputs among 6 different cases. The correlation calculated here is the all correlation i.e., after training the model on experimental data set, we use the whole data set (includes training data set and testing data set) test the trained model and calculate the correlation (Equation 10). The corresponding experiments results are presented in Table II.

$$R = \frac{\sum_m \sum_n (A_{mn} - \overline{A})(B_{mn} - \overline{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \overline{A})^2)(\sum_m \sum_n (B_{mn} - \overline{B})^2)}} \quad (10)$$

One can conclude that MBRC-RNN is capable of providing End-to-End complex sensory data modeling and prediction service. It automatically merges 6 different types of sensory data and precisely fits the tool-wear states. It worths to note that, in this work, the experimental data set is highly corrupted while the results are still accuracy, which demonstrates the robustness of MBRC-RNN.

Table III comprehensively illustrates a qualitative and quantitative comparison among various modeling methods for milling sensory data. The assessment on performance of each method is carefully performed based on provided results and specifications of methods in their corresponding reports. Under such evaluations, results suggest the superiority of MBRC-RNN over existing methods where it can easily and automatically model noisy milling-process generated data.

b) Predict Tool-Wear with Small Size of Hidden Layer: Conventionally, if one uses deep neural network to tackle the issue of this work, it will need several layers with hundreds neurons totally for hidden layers, since the data is corrupted and retained in such high dimensions. Observing the experimental results, one can see that the number of hidden neurons are 11, 12, 9, 20, 11 and 17, respectively (Table II). The reason for this is neurons share their own states

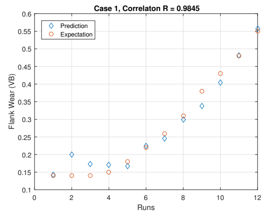

Fig. 4. Case 1 experiment results. The correlation efficient between prediction and desired outputs is $R = 0.9845$.

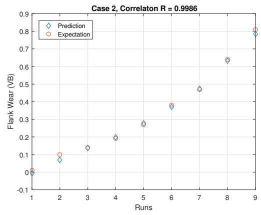

Fig. 5. Case 2 experiment results. The correlation efficient between prediction and desired outputs is $R = 0.9986$.

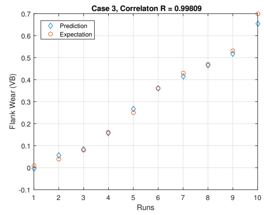

Fig. 6. Case 3 experiment results. The correlation efficient between prediction and desired outputs is $R = 0.99809$.

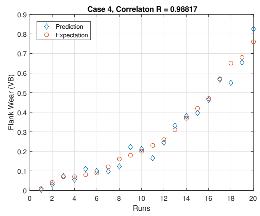

Fig. 7. Case 4 experiment results. The correlation efficient between prediction and desired outputs is $R = 0.98817$.

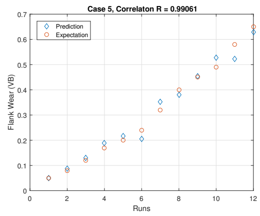

Fig. 8. Case 5 experiment results. The correlation efficient between prediction and desired outputs is $R = 0.99061$.

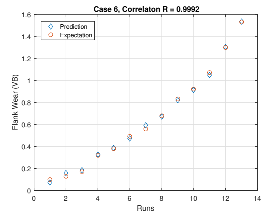

Fig. 9. Case 6 experiment results. The correlation efficient between prediction and desired outputs is $R = 0.9992$.

with others through complete recurrent connections. With the help of ZIZO, one can quickly check the modeling accuracy, then adjust the number of neurons accordingly and avoid over fitting issue. Potentially, one can integrate the step of searching appropriate size for neural networks with ZIZO, and the program will automatically select the best structure based on practical applications. This is going to be conducted in our next work.

TABLE III
QUALITATIVE AND QUANTITATIVE COMPARISON AMONG BMRC-RNN AND EXISTING METHODS. ++: HIGHLY SATISFIES. +: SATISFIES. −: MUCH LESS SATISFIES.

Method	Complexity	Generalization	Accuracy
MBRC-RNN	–	++	0.99±0.01
PSO-SVM [3]	++	+	0.98
Stochastic [34]	++	+-	0.9
SA-ESN [21]	++	+-	0.99±0.1
S-Transform [35]	+++	+-	0.95±0.04

c) Explainable MBRC-RNN: In order to clearly show how the MBRC-RNN works, especially how every neuron contributes to the final prediction, we visualize the input weights W^{in} and RL weights W for the first 3 different study cases (Figure 10 - Figure 12). Different color-blocks are the connection weights among neurons. It worths to note that, in our work, neurons within hidden layer have self-connections. One can see that neural weights, obtained through ZIZO searching, follow well organized statistical distribution i.e., they are gradually distributed without big jump.

IV. CONCLUSIONS

We proposed a novel neural network called MBRC-RNN for modeling the milling process generated sensory data and

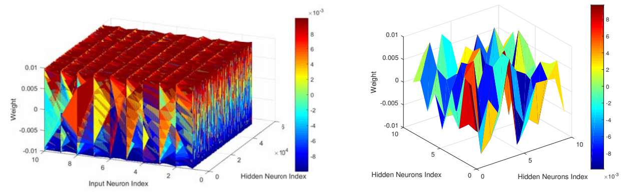

Fig. 10. Case 1. Weights of Input-Hidden Layers & Weights of Hidden-Hidden Recurrent Layer.

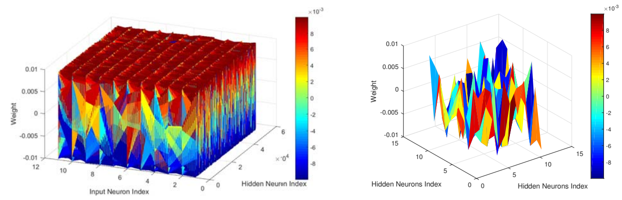

Fig. 11. Case 2. Weights of Input-Hidden Layers & Weights of Hidden-Hidden Recurrent Layer.

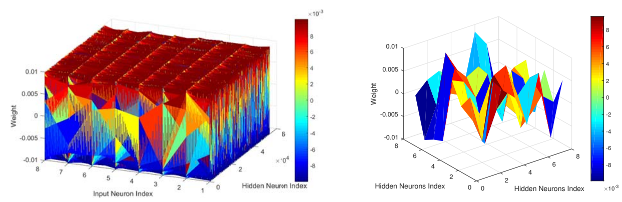

Fig. 12. Case 3. Weights of Input-Hidden Layers & Weights of Hidden-Hidden Recurrent Layer.

predicting the tool-wear degree. The new structure adds multi-bias to the recurrent hidden layer, specifically it treats the none sensory features (static features) as the additional biases. With the help of ZIZO sampling method, one can quickly obtain a group of appropriate weights for input layer W^{in} and RL layer W. The readout layer uses linear combination technique to train the W^{out}. MBRC-RNN totally has three different layer i.e., input layer, RL layer and readout layer. RL is a bio-inspired layer for transforming the inputs to a either high dimension space or reducing the redundancy information of inputs, according to different application scenarios. We demonstrate that MBRC-RNN can successfully model the tool wear status of milling process under various operation conditions (the data set is collected with different operation parameters). In this work, we also show that with much less neurons in hidden recurrent layer, one can precisely predict the tool wear status in different scenarios and obtain very stable and robust results.

For the future work, we are planning to improve the performance of weights searching process via automatically detecting the optimal or suboptimal structure for MBRC-RNN. The core idea is keeping the size of RL layers stay as small as possible and improving the prediction accuracy as much as possible at the same time. To generalize the proposed method, it is also very important to apply MBRC-RNN on other key machining processes e.g., the gearing process, drilling process and so forth.

Acknowledgment

This work is partially supported by DC-Cyber-Physical Production Systems Project, funded by TU Wien, and EU-ECSEL under grant agreement n^o 737459 (project Productive4.0). The authors would like to acknowledge the great suggestions from Mr. Ramin M.Hasani during the preparation of this paper.

References

[1] Benkedjouh T., Kamal Medjaher, Noureddine Zerhouni, and Said Rechak, *Health assessment and life prediction of cutting tools based on support vector regression*. Journal of Intelligent Manufacturing 26, no. 2 (2015): 213-223.

[2] Zhu Kunpeng, and Xiaolong Yu, *The monitoring of micro milling tool wear conditions by wear area estimation*. Mechanical Systems and Signal Processing 93 (2017): 80-91.

[3] Garcia Nieto, Paulino Jose, E Garcia Gonzalo, JA Vilan Vilan, and A Segade Robleda, *A new predictive model based on the PSO-optimized support vector machine approach for predicting the milling tool wear from milling runs experimental data*. The International Journal of Advanced Manufacturing Technology 86, no. 1-4 (2016): 769-780.

[4] Li Dongdong, Boping Xiao, Haiping Huang, and Aoqing Wang, *A method of predicting demand for aircraft follow-up spare based on discrete particle swarm optimization algorithm and RBF neural network*. In Industrial Engineering and Engineering Management (IEEM), 2016 IEEE International Conference on, pp. 1488-1492. IEEE, 2016.

[5] Kimotho James K Christoph Sondermann-Woelke, Tobias Meyer, and Walter Sextro, *Machinery prognostic method based on multi-class support vector machines and hybrid differential evolution-particle swarm optimization*. Chemical Engineering Transactions 33 (2013): 619-624.

[6] Hong Young-Sun, Hae-Sung Yoon, Jong-Seol Moon, Young-Man Cho, and Sung-Hoon Ahn, *Tool-wear monitoring during micro-end milling using wavelet packet transform and Fishers linear discriminant*. International Journal of Precision Engineering and Manufacturing 17, no. 7 (2016): 845-855.

[7] Kong Dongdong, Yongjie Chen, and Ning Li, *Force-based tool wear estimation for milling process using Gaussian mixture hidden Markov models*. The International Journal of Advanced Manufacturing Technology (2017): 1-13.

[8] Patil Aditi B., JitendraA Gaikwad, and Jayant V Kulkarni, *Bearing fault diagnosis using discrete Wavelet Transform and Artificial Neural Network*. In Applied and Theoretical Computing and Communication Technology (iCATccT), 2016 2nd International Conference on, pp. 399-405. IEEE, 2016.

[9] Hasani Ramin M., Guodong Wang, and Radu Grosu, *An Automated Auto-encoder Correlation-based Health-Monitoring and Prognostic Method for Machine Bearings*. arXiv preprint arXiv:1703.06272 (2017).

[10] Hasani, Ramin M., Dieter Haerle, Christian F. Baumgartner, Alessio R. Lomuscio, and Radu Grosu, *Compositional neural-network modeling of complex analog circuits*. In Neural Networks (IJCNN), 2017 International Joint Conference on, pp. 2235-2242. IEEE, 2017.

[11] Bangalore Pramod, and Lina Bertling Tjernberg, *An artificial neural network approach for early fault detection of gearbox bearings*. IEEE Transactions on Smart Grid 6, no. 2 (2015): 980-987.

[12] Drouillet Cyril, Jaydeep Karandikar, Chandra Nath, Anne-Claire Journeaux, Mohamed El Mansori, and Thomas Kurfess, *Tool life predictions in milling using spindle power with the neural network technique*. Journal of Manufacturing Processes 22 (2016): 161-168.

[13] Guo Liang, Naipeng Li, Feng Jia, Yaguo Lei, and Jing Lin, *A recurrent neural network based health indicator for remaining useful life prediction of bearings*. Neurocomputing 240 (2017): 98-109.

[14] Corne Raphael, Chandra Nath, Mohamed El Mansori, and Thomas Kurfess, *Enhancing Spindle Power Data Application with Neural Network for Real-time Tool Wear/Breakage Prediction During Inconel Drilling*. Procedia Manufacturing 5 (2016): 1-14.

[15] Xiao Shuai, Junchi Yan, Xiaokang Yang, Hongyuan Zha, and Stephen M Chu, *Modeling the Intensity Function of Point Process Via Recurrent Neural Networks*. In AAAI, pp. 1597-1603. 2017.

[16] Ali Jaouher Ben, Brigitte Chebel-Morello, Lotfi Saidi, Simon Malinowski, and Farhat Fnaiech, *Accurate bearing*

remaining useful life prediction based on Weibull distribution and artificial neural network. Mechanical Systems and Signal Processing 56 (2015): 150-172.

[17] DAddona Doriana M., AMM Sharif Ullah, and D Matarazzo, *Tool-wear prediction and pattern-recognition using artificial neural network and DNA-based computing*. Journal of Intelligent Manufacturing 28, no. 6 (2017): 1285-1301.

[18] Bazdar Aliasghar, Reza Baradaran Kazemzadeh, and Seyed Taghi Akhavan Niaki., *Fault diagnosis within multistage machining processes using linear discriminant analysis: a case study in automotive industry*. Quality Technology Quantitative Management 14, no. 2 (2017): 129-141.

[19] Feng Yi, Baochun Lu, and Dengfeng Zhang, *Multifractal manifold for rotating machinery fault diagnosis based on detrended fluctuation analysis*. Journal of Vibroengineering 18, no. 8 (2016).

[20] Cao Hongrui, Denghui Li, and Yiting Yue, *Root Cause Identification of Machining Error Based on Statistical Process Control and Fault Diagnosis of Machine Tools*. Machines 5, no. 3 (2017): 20.

[21] Wang Guodong, and Radu Grosu., *Milling-Tool Wear-Condition Prediction with Statistic Analysis and Echo-State Networks*. In Proceedings of S2M'16, (2016), the International Conference on Sustaniable Smart Manufacturing.

[22] Cerrada Mariela, Grover Zurita, Diego Cabrera, Rene-Vinicio Sanchez, Mariano Artes, and Chuan Li., *Fault diagnosis in spur gears based on genetic algorithm and random forest*. Mechanical Systems and Signal Processing 70 (2016): 87-103.

[23] Liao Linxia, *Discovering prognostic features using genetic programming in remaining useful life prediction*. IEEE Transactions on Industrial Electronics 61, no. 5 (2014): 2464-2472.

[24] Yin Shen, and Xiangping Zhu, *Intelligent particle filter and its application to fault detection of nonlinear system*. IEEE Transactions on Industrial Electronics 62, no. 6 (2015): 3852-3861.

[25] Szydlowski Michal, Bartosz Powalka, Marcin Matuszak, and Pawe Kochmanski, *Machine vision micro-milling tool wear inspection by image reconstruction and light reflectance*. Precision Engineering 44 (2016): 236-244.

[26] Wang Guodong, Ramin M. Hasani, Yungang Zhu, and Radu Grosu, *A novel Bayesian network-based fault prognostic method for semiconductor manufacturing process*. In Industrial Technology (ICIT), 2017 IEEE International Conference on, pp. 1450-1454. IEEE, 2017.

[27] Li Zhimeng, Guofeng Wang, and Gaiyun He, *Milling tool wear state recognition based on partitioning around medoids (PAM) clustering*. The International Journal of Advanced Manufacturing Technology 88, no. 5-8 (2017): 1203-1213.

[28] Xia Min, Teng Li, Lin Xu, Lizhi Liu, and Clarence W de Silva, *Fault Diagnosis for Rotating Machinery Using Multiple Sensors and Convolutional Neural Networks*. IEEE/ASME Transactions on Mechatronics (2017).

[29] A Agogino, K Goebel, *Milling Data Set*. BEST lab, UC Berkeley, NASA Ames Prognostics Data Repository, NASA Ames Research Center, Moffett Field, CA.

[30] Wang Guodong, Mohamed Amin Ben Sassi, and Radu Grosu, *ZIZO: A Novel Zoom-InZoom-Out Search Algorithm for the Global Parameters of Echo-State Networks*. Canadian Journal of Electrical and Computer Engineering 40, no. 3 (2017): 210-216.

[31] Xue Fangzheng, Qian Li, and Xiumin Li, *The combination of circle topology and leaky integrator neurons remarkably improves the performance of echo state network on time series prediction*. PloS one 12, no. 7 (2017): e0181816.

[32] Lukosevicius Mantas, *A practical guide to applying echo state networks*. In Neural networks: tricks of the trade, pp. 659-686. Springer Berlin Heidelberg, 2012.

[33] Hasani Ramin M., Dieter Haerle, and Radu Grosu, *Efficient modeling of complex analog integrated circuits using neural networks*. In Ph. D. Research in Microelectronics and Electronics (PRIME), 2016 12th Conference on, pp. 1-4. IEEE, 2016.

[34] Wang Peng, and Robert X Gao, *Stochastic tool wear prediction for sustainable manufacturing*. Procedia CIRP 48 (2016): 236-241.

[35] Rad Javad Soltani, Ensieh Hosseini, Youmin Zhang, and Chevy Chen, *Online tool wear monitoring and estimation using power signals and S-transform*. In Control and Fault-Tolerant Systems (SysTol), 2013 Conference on, pp. 234-238. IEEE, 2013.