Engineering Applications of Artificial Intelligence 35 (2014) 105–113

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Machine health condition prediction via online dynamic fuzzy neural networks

Yongping Pan [a,*], Meng Joo Er [b], Xiang Li [c], Haoyong Yu [a], Rafael Gouriveau [d]

[a] Department of Biomedical Engineering, National University of Singapore, Singapore 117575, Singapore
[b] School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore
[c] Singapore Institute of Manufacturing Technology, Singapore 638075, Singapore
[d] Automatic Control and Micro-Mechatronic Systems Department, FEMTO-ST Institute, Besancon 25000, France

ARTICLE INFO

Article history:
Received 5 October 2013
Received in revised form
29 May 2014
Accepted 29 May 2014
Available online 9 July 2014

Keywords:
Fuzzy neural network
Machine health condition
Non-stationary time series
Multi-step prognosis
Real-time learning
Self-organization

ABSTRACT

Machine health condition (MHC) prediction is useful for preventing unexpected failures and minimizing overall maintenance costs in condition-based maintenance. The neural network (NN)-based data-driven method has been considered to be promising for MHC prediction due to the adaptability, nonlinearity and universal approximation capability of NNs. This paper presents an online MHC prediction approach using online dynamic fuzzy NNs (OD-FNNs) with structure and parameters learning. To meet the requirement of real-time application, the original OD-FNN is simplified based on an extreme learning machine technique as follows: (1) initial fuzzy rules are randomly generated without the knowledge of training data; (2) fuzzy rules are added and pruned uniformly by fired strength-based criteria; (3) antecedent parameters are fixed after generation so that only consequent parameters are updated online. The modified OD-FNN is particularly suitable for MHC prediction since: (1) fuzzy rules can evolve as new training datum arrives, which enables us to cope with non-stationary processes in MHC; (2) learning mechanisms applied are simple and efficient for real-time implementation. The validity and superiority of the proposed MHC prediction approach has been evaluated by real-world monitoring data from the accelerated bearing life.

1. Introduction

Machine condition-based maintenance (CBM)[1] aims to prevent unexpected failures and minimize maintenance costs via condition monitoring information (Jardine et al., 2006). There are mainly six parts in the CBM, including data acquisition, data processing, feature extraction, fault diagnosis, fault prognosis, and decision making (Vachtsevanos et al., 2006). Fault prognosis is pivotal in the CBM since it provides accurate prediction of machine health condition (MHC) or even machine remaining useful life (RUL). Typically, existing prognosis methods can be divided into two categories, namely a model-based method and a data-driven method (Jardine et al., 2006). The model-based method directly utilizes physical models to predict the fault progression (Yu et al., 2011, 2014; Gasperin et al., 2011a,b), whereas the data-driven method employs the collected monitoring data to model fault propagation dynamics (Sokolowski, 2004). The fuzzy logic system (FLS) or neural network (NN)-based method, which falls under the category of the data-driven method, has been considered to be very promising for MHC prediction due to the adaptability, nonlinearity and universal approximation capability of FLSs or NNs (Zhao et al., 2009; Pan et al., 2011a,b). Batch learning and sequential learning are two major schemes for updating NN parameters (Rong et al., 2009). MHC prediction is essentially an online non-stationary time-series prediction problem which should perform real-time prediction while updating NN parameters. Thus, to save implementation cost while maintaining NN reliability, it is suggested that sequential learning should be employed in such a problem.

The most popular NNs applied to MHC prediction are recurrent NNs (RNNs) and fuzzy NNs (FNNs) (Tian and Zuo, 2010; Wang et al., 2004; Wang, 2007; Liu et al., 2009; Zhao et al., 2009; Wu et al., 2007; Lin and Tseng, 2005; Tran et al., 2009; Chen et al., 2011). The first NN-based MHC prediction approach was proposed in Wang et al. (2004), where an enhanced FNN with

* Corresponding author.
E-mail address: biepany@nus.edu.sg (Y. Pan).

[1] Refer to ISO 13374-1, Condition monitoring and diagnostics of machines – Data processing, communication and presentation – Part 1: General Guidelines. Geneva, Switzerland, 2002.

http://dx.doi.org/10.1016/j.engappai.2014.05.015
0952-1976/

recursive least squares (RLS) estimation was developed to forecast MHC. Next, a recurrent counterpart of Wang et al. (2004) and a multi-step counterpart of Wang (2007) were presented in Wang (2007) and Liu et al. (2009), respectively. An extended RNN that contains two types of feedback layers, namely Elman and Jordan context layers, was developed for gearbox MHC prediction in Tian and Zuo (2010). The classical FNN was applied to predict bearing MHC in Zhao et al. (2009). An NN-based decision support system for bearing maintenance was developed in Wu et al. (2007), where the classical feedforward NN was applied to estimate the life percentile and failure times of roller bearings. To improve the reliability of MHC prediction, reliability statistics (Lin and Tseng, 2005), regression trees (Tran et al., 2009) and Bayesian estimation (Chen et al., 2011) were also integrated into NN-based MHC prediction approaches. Note that batch learning was employed in Tian and Zuo (2010), Zhao et al. (2009), Wu et al. (2007), Lin and Tseng (2005), Tran et al. (2009), and Chen et al. (2011). Consistent conclusions from all these approaches are that the RNN usually outperforms the feedforward NN, and the FNN usually outperforms the stochastic model, the feedforward NN and the classical RNN. Yet, in spite of the effectiveness of these state-of-the-art MHC prediction approaches, the NNs applied therein have some common drawbacks as follows: (1) the fixed NN structures limit the flexibility and generalization capability of NNs; (2) the back-propagation (BP) learning algorithms applied to parameters learning increase the risk of local minimum; and (3) the adjustments of hidden node parameters lead to tedious designs of learning algorithms. In addition, all these drawbacks can inevitably increase the implementation cost of NNs.

The self-organizing FNN is a type of FNNs that contain two learning levels: structure learning and parameters learning. Structure learning includes the generation and/or deletion of fuzzy rules (i.e. neurons) while parameters learning includes updating antecedent (i.e. hidden node) and/or consequent (i.e. output weight) parameters. Roughly, the self-organizing FNN can be divided into two categories, namely a traditional self-organizing FNN (Juang and Lin, 1998; Lin et al., 2001; Tzafestas and Zikidis, 2001; Azeem et al., 2003; Leng et al., 2004; Rong et al., 2006; Lin and Lian, 2010; Chen, 2011; Wu et al., 2001; Er and Deng, 2004; Er et al., 2005; Er and Zhou, 2008; Pratama et al., 2013) and an evolving self-organizing FNN (Yao, 1999; Garcia-Pedrajas et al., 2003; Niska et al., 2004; Yu et al., 2008; Soleimani-B et al., 2010; Prado et al., 2010; Dovzan and Skrjanc, 2011; Lughofer et al., 2011). The evolving self-organizing FNN involves population-based evolutionary computation techniques, e.g. evolutionary programming, genetic algorithms and competitive learning, which is usually subject to the dilemma of time-consuming (Leng et al., 2004). The dynamic FNN (D-FNN) is a type of traditional self-organizing FNNs whose structure can be learned dynamically (Wu et al., 2001). The salient features of the D-FNN include: (1) fuzzy rules can be generated hierarchically based on accommodation boundary and system error criteria; (2) insignificant rules can be pruned dynamically according to their contribution to system performance; and (3) parameters learning can be achieved by least-squares estimation without iterative tuning. Despite of fast learning speed and favorable generalization ability, the original D-FNN is not suitable for online prediction since its error reduction ratio (ERR)-based pruning criterion needs to gather all historical data. To cope with this issue, an online D-FNN (OD-FNN) with fuzzy Q-learning and an enhanced OD-FNN with self-organizing mapping (SOM), all without rule pruning mechanisms, were developed in Er and Deng (2004) and Er et al. (2005), respectively. Next, a semi-supervised learning OD-FNN was developed in Er and Zhou (2008), where both structure and parameters learnings are performed by reinforcement learning online. Most recently, a parsimonious OD-FNN was developed in Pratama et al. (2013), where antecedent parameters are allocated by extended SOM, consequent parameters are updated by time localized least squares estimation, and insignificant rules are deleted by an online sequential pruning mechanism.

It is worth noting that the OD-FNN, even the self-organizing NN, has not been applied to MHC prediction yet. Owing to the potential of the OD-FNN for relaxing the drawbacks of the existing NN-based MHC prediction approaches, this paper focuses on the application of the OD-FNN to MHC prediction. Extreme learning machine (ELM) is an emergent technique for training generalized feedforward NNs with a key principle that all hidden node parameters are randomly generated without the knowledge of training data and are fixed after generation without online updating (Huang et al., 2006a). The result in Huang et al. (2006b) shows that a NN trained by the ELM can also guarantee universal function approximation capability. According to the previous analysis, some conclusions can be obtained as follows: (1) the ERR-based offline rule pruning criterion is not suitable for online MHC prediction; (2) current prediction errors are unavailable for rule generation during MHC prediction; and (3) the allocation mechanism of antecedent parameters is unnecessary by the ELM principle. From these conclusions, some efforts should be made so that the OD-FNN is suitable for MHC prediction.

Based on our previous works in Li et al. (2010b) and Pan et al. (2013), this paper presents an OD-FNN-based online MHC prediction approach. The applied FNN implements a Takagi–Sugeno–Kang (TSK) FLS based on ellipsoidal basis functions (EBFs). To meet the requirement of real-time application, the original OD-FNN is simplified based on the ELM technique as follows: (1) initial fuzzy rules are randomly generated without the knowledge of training data; (2) fuzzy rules are added and pruned uniformly by fired strength-based criteria; (3) antecedent parameters are fixed after generation such that only consequent parameters are updated online. From the artificial intelligence point of view, the contribution of this study is that a simplified OD-FNN with fast learning speed and low computational cost is proposed for the online prediction of non-stationary time series. From the engineering application point of view, the contribution of this study is that the OD-FNN is successfully applied to bearing MHC prediction based on real-world monitoring data.

The structure of the rest of the paper is as follows. The architecture of FNNs is described in Section 2. The simplified online learning strategy is proposed in Sections 3. The MHC prediction scheme is given by Section 4. Application studies using real-world monitoring data are provided in Section 5. Conclusions are summarized in Section 6. Throughout this paper, $\mathbb{R}$, $\mathbb{R}^+$, $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ denote the spaces of real numbers, positive real numbers, real n-vectors and $n \times m$ matrixes, respectively, $|\cdot|$ and $\|\cdot\|$ denote the absolute value and 2-norm, respectively, and $\min\{\cdot\}$, $\max\{\cdot\}$ and $\sup\{\cdot\}$ represent the functions of minimum, maximum and supremum, respectively.

2. Architecture of fuzzy neural networks

The applied FNN is built based on an EBF-NN and is functionally equivalent to a TSK FLS described by the following fuzzy rules (Mitra and Hayashi, 2000; Rong et al., 2009):

$$\text{Rule } j: \ \text{IF } x_1 \text{ is } A_{1j} \text{ and } \cdots \text{ and } x_n \text{ is } A_{nj} \quad \text{THEN } \hat{y} \text{ is } w_j \tag{1}$$

where $x_i \in \mathbb{R}$ are input variables, $\hat{y} \in \mathbb{R}$ is an output variable, A_{ij} is the antecedent of the ith input in the jth fuzzy rule, and w_j is the consequent of the jth fuzzy rule given by

$$w_j = \alpha_{0j} + \alpha_{1j}x_1 + \cdots + \alpha_{nj}x_n \tag{2}$$