Journal of Computer Science and Cybernetics, V.33, N.3 (2017), 213–228
DOI 10.15625/1813-9663/33/3/10802

A METHOD OF BEARING FAULT DIAGNOSIS USING SINGULAR SPECTRUM ANALYSIS, SPARSE FILTERING AND ANFIS

SY DZUNG NGUYEN[1], VAN HIEP NGUYEN[2]

[1] *Ton Duc Thang University, Ho Chi Minh City, Viet Nam*
[2] *Master student, Industrial University of Ho Chi Minh City*
nguyensydung@tdt.edu.vn

Abstract. Bearing is an important machine detail participating in almost all mechanical systems. Estimating online its operating condition to exploit actively the systems, therefore, is one of the most urgent requirements. This paper presents an online bearing damage identifying method named ASBDIM based on ANFIS (Adaptive Neuro-Fuzzy Inference System), Singular Spectrum Analysis (SSA) and sparse filtering. This is an online estimating process operated via two phases, offline and online one. In the offline period, by using SSA and sparse filtering, a database signed Off_DaB is built whose inputs are features extracted from the measured data stream typed big data, while its outputs are values encoding the surveyed bearing damage statuses. The ANFIS is then employed to identify the dynamic response of the mechanical system corresponding to the bearing damage statuses reflected by the Off_DaB. In the online period, first, at each estimating time, another database called On_DaB is established using the way similar to the one used for building the Off_DaB. The On_DaB participates as inputs of the ANFIS to generate its outputs which are then compared with the corresponding encoded outputs to specify bearing real status at this time. Survey results based on different data sources showed the effectiveness of the proposed method.

Keywords. Identifying bearing damage, AI for estimating damage, ANFIS based damage identification, SSA for identifying damage.

1. INTRODUCTION

Online identifying bearing fault to exploit, maintain or repair systems actively is really meaningful work [1, 2, 3, 4, 5, 6, 7]. Because mechanical faults cause changes in system dynamic response, actually this work is often carried out via vibration signal. Recently, artificial intelligence and solutions to extracting features from big data are widely paid attention to. A new tendency to damage manage based on measured data streams typed big data and data-driven models using Artificial Neural Networks (ANN), Fuzzy Logic (FL) or ANFIS [7, 8, 9, 10, 11, 12, 13, 14] has been taken form with a promising future. For this aim, in order to improve the ability to process, analyze and extract valuable information from the initial vibration data sets, wavelet and SSA are also often used [10, 11].

Mathematical models FL and ANN can remember and process information to establish complex and highly nonlinear relationships between the input and output data spaces of large data sets. They even can deal with inaccurate databases to deduce and continue learning in order to improve their efficiency via updated-training databases. In practice, for each specific application, an ANN's becoming structure needs to be established. The more reasonable the structure of the net is, the better the expression of the dynamic relationship of the system that can be obtained is. This increases the accuracy of applications and also increases the convergence rate in the training process [15]. However, there is no explicit and accurate answer for the question about how many layers and how many neurons in one layer should be used for each. Similarly, the effectiveness of FL significantly depends on the accuracy of fuzzy sets involving the number of fuzzy rules, the type of relationship functions, and the logical relationship between fuzzy sets in the input space and output space. In order to solve the above issues, several approaches have been proposed such as using the genetic algorithm (GA) [16] or a combination of ANN and FL to construct an ANFIS [17, 18, 19, 20]. Combination of FL and ANN in the form of ANFIS where the advantages of both ANN and FL can be exploited has been applied to many different fields, including estimating fault of mechanical systems [7, 10, 11, 12, 17, 18, 19, 20, 21, 22, 23].

SSA is a non-parametric technique of time series analysis based on the principles of multivariate statistics. It decomposes a given time series into a set of independent additive time series. Fundamentally, by using a procedure of principal component analysis, SSA projects the original time series onto a vector basis obtained from the series itself. The set of series obtained from the decomposition can be interpreted as a slowly varying trend representing the signal mean at each instant, a set of periodic series, and aperiodic noise [11] To extract information correlated with system response, the main steps are performed as follows. First, SSA builds a matrix, called the trajectory matrix, from the original time series in a process called embedding. This matrix consists of vectors obtained by means of a sliding window that traverses the series. The trajectory matrix is then subjected to singular value decomposition (SVD). The SVD decomposes the trajectory matrix into a sum of unit-rank matrices known as elementary matrices. Each of these matrices can be transformed into a reconstructed time series by a process known as diagonal averaging. The obtained time series is called principal components in which the sum of all the principal components is equal to the original time series. In next step known as grouping, the selection of the principal components that represent the trend of the signal is carried out [24, 25, 26].

In [10], a beam-damage-locating method constituted of ANFIS and an average quantity solution to wavelet transform coefficient of beam vibration signal was presented. The ANFIS was used to remember undamaged-beam dynamic properties while wavelet t took a part in signal analysis. In [11], Nguyen et al. proposed a method for identifying the undamaged status of a structure using ANFIS and SSA, because SSA could extract information correlated with system dynamic response state, via a database created from SSA the special features of the signal source could be specified more effectively than by using the original data. As a result, based on the database, the ANFIS could describe more objectively system dynamic response Another approach of ANFIS can be referred in [12], where a structure damage-locating method employing wavelet analysis and type-2 fuzzy logic system was represented. In spite of owning some advantages, the three methods as mentioned do not appropriately satisfy online applications due to high calculating cost. Recently, in order to make fault diagnosis less dependent on prior knowledge and diagnostic expertise when processing mechanical big data, an intelligent bearing fault diagnosis methodwas proposed in [14]. The method established a two-stage unsupervised learning process to extract features and estimate system. In the first one, sparse filtering and an unsupervised two-layer neural network were used to directly learn features from mechanical vibration signals. In the second stage, classifying the health conditions using the learned features and softmax regression was carried out. Although the method could reduce the need of human labor and made intelligent fault diagnosis handle big data more easily, preprocessing data was not paid attention pertinently. Deriving from our surveys, we can judge that combining the SSA's ability to extract information correlated with system dynamic response state and sparse filtering for processing and analyzing big data could bring us positive effectiveness.

Consequently, in this study we present a bearing damage identifying method named ASBDIM based on ANFIS, SSA, and sparse filtering. In the method, not only the combination of SSA and sparse filtering is considered but also the capability to approximate multi-dimensional and highly nonlinear mappings is exploited. This is an online checking method consisting of offline and online phases. In the offline period, firstly SSA and sparse filtering are coordinated to build a database, signed Off_DaB. For this work, preprocessing the original data source typed a big data stream and extracting meaningful features from it are carried out in succession. In the database, the input is the features, while its output is values encoding the surveyed bearing fault statuses. The ANFIS is then employed to identify the dynamic response of the mechanical system corresponding to the bearing damage statuses reflected by the Off_DaB. While the online phase is performed during the mechanical system operating process. During this period, at each checking time, another database called On_DaB is established using the way similar to the one used to build the Off_DaB. By comparing the encoded output with the output of the ANFIS corresponding to the input coming from the On_DaB, the bearing fault status can be addressed.

Three contributions of the study are as follows. Related to establishing and exploiting the ASBDIM, there are two proposed solutions consisting of 1) building the databases relied on SSA and sparse filtering, and 2) synthesizing the data-driven model ANFIS for identifying the surveyed bearing fault statuses via the Off_DaB as well as classifying the bearing real statuses during the system operating process based on the On_DaB. As the third contribution, an experimental system is fitted up to verify the effectiveness of the method.

2. RELATED THEORIES

2.1. Singular spectrum analysis

By using SSA, a set of independent additive time series can be generated from a given time series. Fundamentally, the set of series obtained from the decomposition can be interpreted as a slowly varying trend representing the signal mean at each instant, a set of periodic series, and aperiodic noise [24, 25, 26].

In this study we use the SSA decomposition algorithm presented [25] as follows.

Step 1: Embedding

From the raw data, a time series having Ndata points $\mathrm{F} = (z_0,\ z_1, ..., z_{N-1})$ is considered. By using a *window length* L which is an $1 < L < N$, one defines $K = N - L + 1$ sliding vectors $\mathrm{X_j} = (z_{j-1},\ z_j, ..., z_{j+L-2})^T$, j=1, 2, ..., $K$and a *trajectory matrix* X as follows

$$\mathrm{X} = (\mathrm{X}_1, \mathrm{X}_1, ..., \mathrm{X}_\mathrm{K}) = \begin{pmatrix} z_0 & z_1 & \cdots & \cdots & z_{N-L} \\ z_1 & z_2 & \cdots & \cdots & z_{N-L+1} \\ \vdots & \vdots & \ddots & \iddots & \vdots \\ z_{L-2} & z_{L-1} & \iddots & \ddots & z_{N-2} \\ z_{L-1} & z_L & \cdots & \cdots & z_{N-1} \end{pmatrix}, \tag{1}$$

X is a *Hankel matrix*, i.e., the elements of the diagonals $i+j=$ constant are equal.

Step 2: SVD of the trajectory matrix

One firstly calculates the eigenvalues and eigenvectors of the matrix $\mathrm{S} = \mathrm{XX}^T$ of dimension $L \times L$. Let $\lambda_1,\ \lambda_2, ..., \lambda_d$ be the non-zero eigenvalues of **S** arranged in the descending order, and $\mathbf{U}_1$, $\mathbf{U}_2$, ..., $\mathbf{U}_d$ be the corresponding eigenvectors. Vectors V_i are then constructed as below

$$\mathrm{V}_i = \mathrm{X}^T \mathrm{U}_i \Big/ \sqrt{\lambda_i}, \quad i = 1...d. \tag{2}$$

As a result, one obtains a decomposition of the trajectory matrix into a sum of matrices $\mathrm{X} = \mathrm{E}_1 + \mathrm{E}_2 + ... + \mathrm{E}_d$, where $\mathrm{E}_i = \sqrt{\lambda_i} \mathrm{U}_i \mathrm{V}_i^T$. These are the mutually orthogonal, unit-rank, elementary matrices.

Step 3: Reconstruction (diagonal averaging)

At this step, each elementary matrix is transformed into a principal component of length N by applying a linear transformation known as diagonal averaging or Hankelization. Let **Z** be a matrix of dimension $L \times K$ with elements $z_{i,j},\ 1\ \le i \le L,\ 1\ \le j \le K$. The diagonal averaging algorithm transforms **Z** into the reconstructed time series $g_0,\ g_1, ..., g_{N-1}$ by applying the expression

$$g_k = \begin{cases} \dfrac{1}{k+1} \sum_{m=1}^{k+1} z_{m,k-m+2}, & 0 \le k < L* -1 \\ \dfrac{1}{L*} \sum_{m=1}^{L*} z_{m,k-m+2}, & L* -1 \le k < K* \\ \dfrac{1}{N-k} \sum_{m=k-K*+2}^{N-K*+1} z_{m,k-m+2}, & K* \le k < N \end{cases} \tag{3}$$

where $L* = \min(L,\ K)$, $\quad K* = \max(L,\ K)$, and $N = L + K - 1$.

2.2. Sparse filtering

Sparse filtering optimizes the sparsity distribution of the features calculated by the collected data instead of modeling the distribution of the data. It tries to learn good features that satisfy the three principles consisting of population sparsity, lifetime sparsity, and high dispersal [14, 27]. Population sparsity means that each sample should be represented by only several active features, namely most of the features extracted from each sample should be zero. Lifetime sparsity means that each feature should be nonzero only for a few samples, which is beneficial to extract discriminative features. While high dispersal is to encourage the distribution of features having similar statistics to each other and it will improve the generalization ability of the features [28].

The result of sparse filtering is a data set whose inputs are collected samples while the outputs are the learned features. This can be summarized as follows [14].

From the original data set, a training data set of M samples $\{\mathrm{x}^i\}_{i=1}^{M}$, where $\mathrm{x}^i \in \Re^{N\times 1}$ is the i-the sample, is established. Sparse filtering maps the samples onto L their features $f^i \in \Re^{N\times 1}$, by which a weight matrix $\mathrm{W} \in \Re^{N\times L}$ is created. Thus, the l-th feature of the i-th sample signed f_l^i can be expressed $f_l^i = \mathrm{W}_l^T \mathrm{x}^i, i = 1...M, l = 1...L$.

The features f_l^i compose a feature matrix. Let l_p-norm of vector $\mathrm{t} = [t_1, ..., t_n]$ be $\|\mathrm{t}\|_p = \sqrt[p]{|t_1|^p + ... + |t_2|^p}$. First, normalizing each row of the feature matrix by l_2-norm across all the samples using $\tilde{\mathrm{f}}_l = \mathrm{f}_l / \|\mathrm{f}_l\|_2$, $l = 1...L$. Then, each column is normalized by its l_2-norm, so that the features lie on the unit l_2-ball $\hat{\mathrm{f}}^{\,i} = \tilde{\mathrm{f}}^{\,i} \Big/ \left\|\tilde{\mathrm{f}}^{\,i}\right\|_2$. Finally, the optimal weight matrix **W** could be solved by optimizing a cost function constraining l_1-norm for each sample such as $\underset{\mathrm{W}}{\text{minimize}} \sum_{i=1}^{M} \left\| \hat{\mathrm{f}}^{\,i} \right\|_1$.

2.3. Building ANFIS from a data set

As usual, a given initial data set (IDS) of input-output data samples $(\bar{\mathrm{x}}_i, y_i)$, $\bar{\mathrm{x}}_i = [x_{i1}, ..., x_{in}] \in \Re^n$, $y_i \in \Re^1, i = 1...P$, expressing an unknown mapping $f:\ \mathrm{X} \to \mathrm{Y}$ can be approximated by an ANFIS, by which the ANFIS undertakes the role of the mapping. In [15], an algorithm named FIN-ANFIS for building ANFIS via a recurrent mechanism and impulse noise filter was presented. Because the FIN-ANFIS will be employed in this paper to identify dynamic response relied on a matrix of distilled features, the filtering function should be hence extricated. The rest of this subsection briefly presents its version without the filter so-renamed the EFIN-ANFIS.

2.3.1. Structure of the ANFIS

First, from an IDS, a normalized initial data space signed $\overline{\mathrm{IDS}}$ is built. A cluster data space (CDS) is then created consisting of three steps: 1) building fuzzy clusters with centroids $(\bar{\mathrm{x}}_1^0, ..., \bar{\mathrm{x}}_C^0)$, 2) specifying the hard distribution of samples in each input data cluster, and 3) building the hyper-planes $w_k(.)$ in the output data space based on the hard distribution status in the input data space. Finally, the CDS is used to take shape of the ANFIS's structure.

The i-th data sample (also signed $(\bar{x}_i, y_i)$) in the $\overline{\mathrm{IDS}}$ to be $(\bar{\mathrm{x}}_i = [\tilde{x}_{i1}, ..., \tilde{x}_{in}]^T, y_i)$, is specified by $\tilde{x}_{ij} = x_{ij} \Big/ \max\limits_{k} |x_{kj}|, \quad i = 1...P, \quad j = 1...n, \quad k = 1...P$. Cluster centroids in the input data space $\bar{\mathrm{x}}_1^0, ..., \bar{\mathrm{x}}_C^0$ are set up such that objective function

$$J_{KFCM}(U, \bar{\mathrm{x}}^0) = \sum_{i=1}^{C} \sum_{j=1}^{P} \mu_{ij}^m \left\|\varphi(\bar{\mathrm{x}}_j) - \varphi(\bar{\mathrm{x}}_i^0)\right\|^2, \tag{4}$$

is minimized subjected to $\mu_{ij} \in [0,\ 1]\ \ \forall i,\ j$ and $\sum_{i=1}^{C} \mu_{ij} = 1\ \forall j$, where, $\bar{\mathrm{x}}_i^0 = [x_{i1}^0, ..., x_{in}^0] \in \Re^n$ is the i-th cluster center; $\mu_{ij} \in [0,\ 1]$ denotes the membership degree of the j-th data sample belonging to the i-th cluster; $U = U(\mu_{ij})$ is the distribution matrix; $\varphi(.)$ is kernel function; $m > 1$ is the fuzzy factor; and $\left\|\varphi\left(\bar{\mathrm{x}}_j\right) - \varphi\left(\bar{\mathrm{x}}_i^0\right)\right\|^2$ is the squared distance between $\bar{\mathrm{x}}_j$ and $\bar{\mathrm{x}}_i^0$ in the kernel space. By choosing Gaussian kernel function and the method of Lagrange multipliers, the optimal centroids (5), optimal membership degree of the j-th data sample belonging to the i-th cluster, μ_{ij}, (6), and the membership of $\tilde{x}_{il}$ belonging to A^k (7) can all be inferred from (4) (see [15, 17] for more detail).

$$\bar{\mathrm{x}}_i^0 = \sum_{j=1}^{P} \mu_{ij}^m \ \bar{\mathrm{x}}_j \ K(\bar{\mathrm{x}}_j, \bar{\mathrm{x}}_i^0) \Big/ \sum_{j=1}^{P} \mu_{ij}^m K(\bar{\mathrm{x}}_j, \bar{\mathrm{x}}_i^0), i = 1...C, \tag{5}$$

$$\mu_{ij} = \begin{cases} \left[\left(\sum\limits_{h=1}^{C} \dfrac{1 - K(\bar{\mathrm{x}}_j, \bar{\mathrm{x}}_i^0)}{1 - K(\bar{\mathrm{x}}_j, \bar{\mathrm{x}}_h^0)} \right)^{1/(m-1)} \right]^{-1} & \text{if } \bar{\mathrm{x}}_j \neq \bar{\mathrm{x}}_i^0 \\ 1 \quad (\text{and } \mu_{ik(k \neq j)} = 0) \quad \text{if } \bar{\mathrm{x}}_j = \bar{\mathrm{x}}_i^0 \end{cases} \\ i = 1...C; j = 1...P, \tag{6}$$

$$\bar{\mu}_{ki}(\tilde{x}_{il}) \equiv A_l^k(\tilde{x}_{il}) = \left[\left(\sum\limits_{h=1}^{C} \frac{1 - K(\tilde{x}_{il}, x_{kl}^0)}{1 - K(\tilde{x}_{il}, x_{hl}^0)} \right)^{1/(m-1)} \right]^{-1}, \\ (k = 1...C; i = 1...P; \ l = 1...n.) \tag{7}$$

where, $A^1, ..., \ A^C$ respectively are the input fuzzy sets corresponding to $\bar{\mathrm{x}}_1^0, ..., \bar{\mathrm{x}}_C^0$. From (7), membership of $\bar{\mathrm{x}}_q$ belonging to A^i and its normalized value are then specified as in (8) and (9), respectively.

$$\mu_{iq}(\bar{\mathrm{x}}_q) \equiv A^i(\bar{\mathrm{x}}_q) = \prod\nolimits_{l=1}^{n} \bar{\mu}_{iq}(\tilde{x}_{ql}), \quad i = 1...C, \ q = 1...P, \tag{8}$$

$$N_i(\bar{\mathrm{x}}_q) = \mu_{iq}(\bar{\mathrm{x}}_q) \Big/ \sum\nolimits_{h=1}^{C} \mu_{hq}(\bar{\mathrm{x}}_q), \quad q = 1...P, i = 1...C. \tag{9}$$

The hard distribution of the data samples in each input cluster is then estimated to calculate the ANFIS's output. The q-th data sample is called to distribute hardly into the i-th cluster if $N_i(\bar{\mathrm{x}}_q) = \max\limits_{h=1...C} (N_h(\bar{\mathrm{x}}_q))$, $q = 1...P, \ i = 1...C$. Let t_i be the number of data samples hardly distributed into the i-th cluster. By using the Least Mean Squares Method for t_i these data samples, index vector $\mathrm{a} = [a_0, a_1, ..., a_n]^T = [a_0, \bar{\mathrm{a}}]^T$ of output hyper-planes $w_i(.)$ is obtained. Namely, value of $w_i(.)$ corresponding to $\bar{\mathrm{x}}_q$ is

$$w_k(\bar{\mathrm{x}}_q) = a_0 + \bar{\mathrm{a}}^T \bar{\mathrm{x}}_q. \tag{10}$$

From the clustering results, a structure of the ANFIS is built with five layers as in Fig. 1. The data layer (D) with n nodes is the input of the cluster layer (CL). The product layer (Π) reflects Eq. (8), while the normalization layer (N) depicts Eq. (9). Finally, the specification layer (S) specifies the predicting output of the ANFIS. If the "winner takes all" law is used, the predicting output is specified by

$$\hat{y}_q(\bar{\mathrm{x}}_q) = w_k(\bar{\mathrm{x}}_q), \ \ q = 1...P, \tag{11}$$

where, k is the index of the output cluster $N_{(.)}(\bar{x}_q)$ (9) getting the maximum value

$$N_k(\bar{\mathrm{x}}_q) = \max_{h=1...C} \left(N_h(\bar{\mathrm{x}}_q)\right).$$

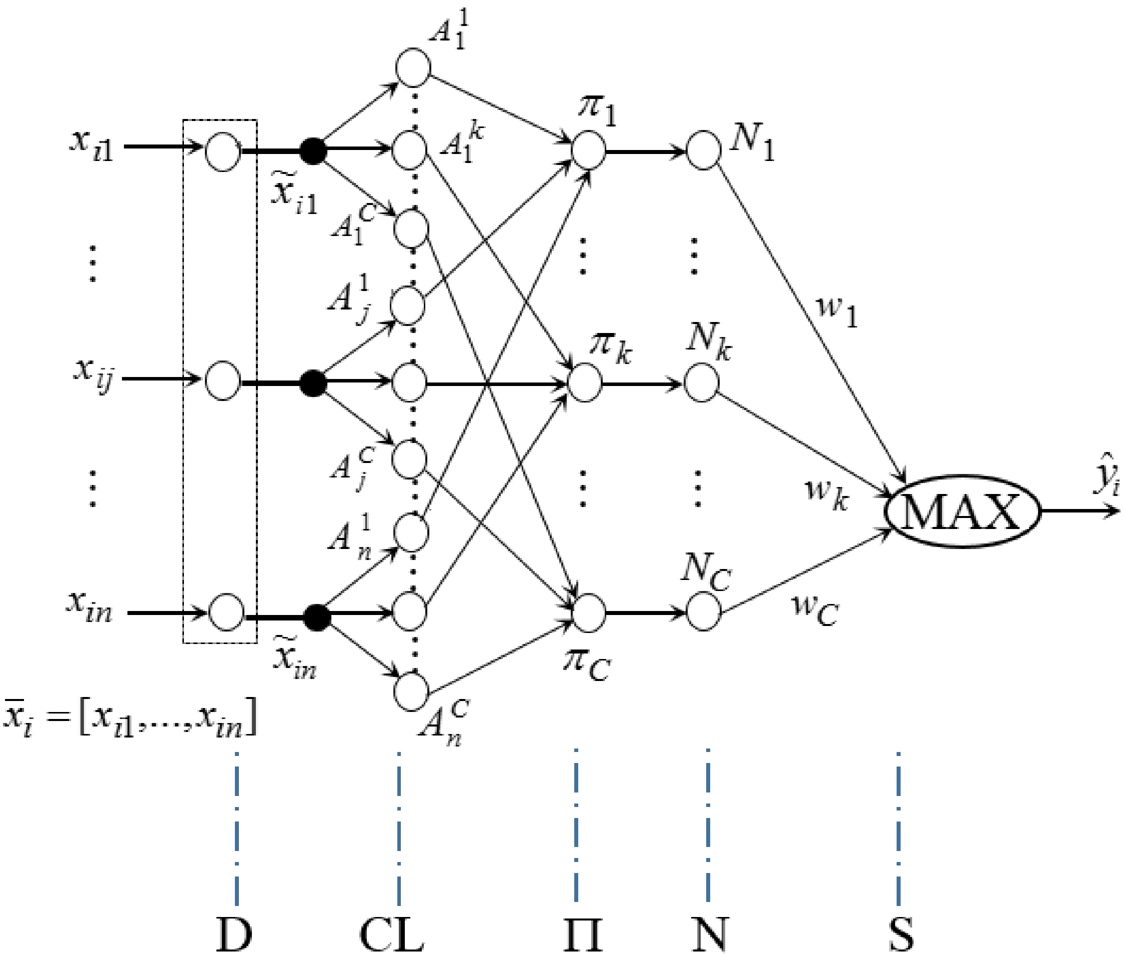

Figure 1. Structure of the ANFIS using "the winner takes all" method for defuzzification

2.3.2. The EFIN-ANFIS

The content of the EFIN-ANFIS is as follows.

Initializing: The initial index of the loops, r=1; the number of clusters $C << P - 1$; $J^{(r)}_{KFCM} = \Omega$, where Ω is a real number $\Omega > [ts]$; the initial cluster centroids are chosen randomly.

Building the CDS:

Step 1. Calculate μ_{ij} based on (6);

Step 2. Update cluster centroids $\bar{x}^0_i(r) = (x^0_{i1}, ..., x^0_{in}), \quad 1 \leq i \leq C$ by using (5);

Step 3. Check the stop condition of the clustering phase via coefficient

$$ts = \left(J^{(r)}_{KFCM} - J^{(r-1)}_{KFCM}\right) \Big/ J^{(r-1)}_{KFCM}$$

and its required value $[ts]$, where r denotes the r-th loop.

- If $ts \leq [ts]$: Go to Step 4;
- If $ts > [ts]$ and $r < [r]$: setup $r =: r + 1$, return to Step 1;

- If $ts > [ts]$ and $r = [r]$ and $C < P - 1$: Set$C =: C + 1, r =: 1$, and return to Step 1;
- If $ts > [ts]$ and $r = [r]$ and $C = P - 1$: Stop (the clustering process cannot converge to the desired distribution).

Forming and Estimating ANFIS:

Step 4. Build ANFIS as in Fig. 1 and calculate RMSE $= (P^{-1} \sum_{i=1}^{P} (\hat{y}_i - y_i)^2)^{0.5}$;
- If RMSE $\leq [E]$: Stop; the obtained ANFIS is the optimal one;
- If RMSE $> [E]$and$C < P - 1$: Go to Step 5;
- If RMSE $> [E]$and$C = P - 1$: Stop (the ANFIS cannot converge to the desired accuracy).

Step 5. Look for the worst data point WP $\equiv \left(\bar{\mathrm{x}}_i^{(WP)}, y_i^{(WP)}\right)$ satisfying (12)

$$\mathrm{WP} \equiv \left(\bar{\mathrm{x}}_i^{(WP)}, y_i^{(WP)}\right) \text{ such that } \left|y_i^{(WP)} - \hat{y}_i^{(WP)}\right| = \max_{h=1...P} |y_h - \hat{y}_h|. \tag{12}$$

Set $C =: C + 1$, $r =: 1$ add a new cluster centroid $\bar{\mathrm{x}}_C^0$ in the neighborhood of the WP, and return to Step 1.

3. PROPOSED METHOD

The proposed method is an online checking method consisting of offline and online phases as in Fig. 2. In the offline period, the Off_DaB is built using SSA and sparse filtering. Preprocessing the original data source and extracting meaningful features from it are carried out in succession. The Off_DaB's input data space (IDS) is the features while its output data space (ODS) is values encoding the surveyed bearing fault statuses. Based on the algorithm EFIN-ANFIS (see Subsection 2.3.2) and the Off_DaB, an ANFIS is then trained. During the mechanical system operating process, the online phase is being performed to estimate bearing fault status. Firstly, at each checking time, another database called On_DaB is built using the way similar to the one used to build the Off_DaB. Subsequently the bearing fault status can be addressed by comparing the encoded output with the output of the ANFIS corresponding to the input coming from the On_DaB. These contents are detailed as below.

3.1. Identifying system dynamic response

3.1.1. Building the database

Building the IDS:

For each bearing damage status, a measured vibration data set is built corresponding to each case coming from N considered fault statuses. Thus, we have N original data sets signed as below

$$[\mathrm{D}_1,\ \mathrm{D}_2, ..., \mathrm{D}_N]. \tag{13}$$

Let D_i correspond to the i-th bearing fault status ($1 \leq i \leq N$). By using SSA we get m subsets signed D_{i1}, D_{i2}, ..., D_{im}, where m is a parameter selected by the designer. As mentioned in [11, 25], the original mechanical vibration is prone to the low frequency range. Therefore, by putting the eigenvalues in the decreasing order, among the m subsets we delete $(m-k)$ last subsets which are considered as noise. Thus, we keep the first k subsets denoted as in (14) to build the database for the ASBDIM

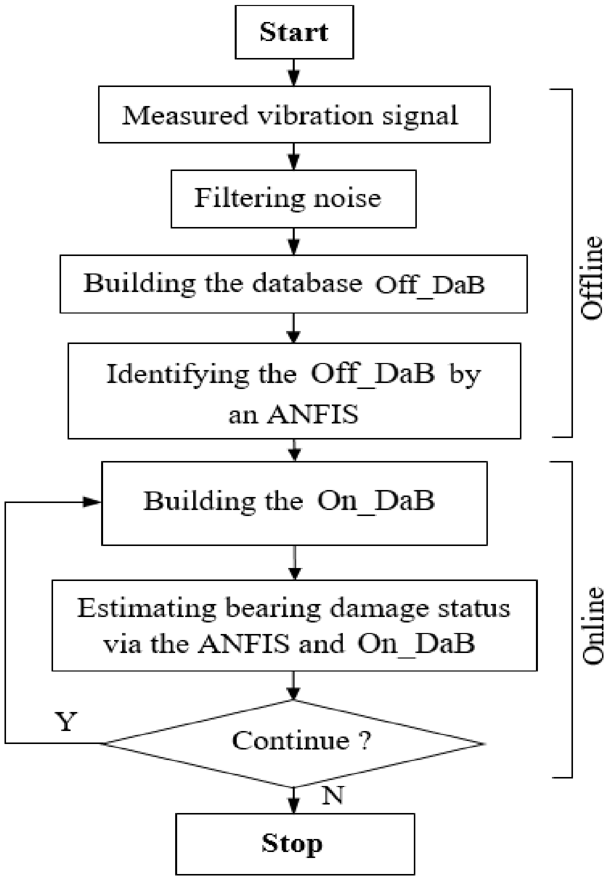

Figure 2. Flowchart of the ASBDIM

$$[\mathrm{D}_{i1},\ \mathrm{D}_{i2}, ..., \mathrm{D}_{ik}]. \tag{14}$$

It should be noted that k needs to be optimized using an objective function reflecting the accuracy rate of the method.

For each subset in (14), for example D_{ij}, by using sparse filtering presented in Subsection 2.2 we get a weight matrix signed $\mathrm{W}_{ij} \in \Re^{L\times M}$, $i = 1...N, j = 1...k$, in which L is the number of surveyed features while M is the number of data samples. A vector signed $\widehat{x}_j^i = [x_{1j}^i, x_{2j}^i..., x_{Lj}^i]^T$ is then generated by averaging out the elements in each row of W_{ij}. As a result, for the i-th damage status we obtain data matrix $\bar{\mathrm{D}}_i \in \Re^{L\times k}$ as in (15).

$$\overline{\mathrm{D}}_i = \begin{bmatrix} x_{11}^i & x_{12}^i & \cdots & x_{1k}^i \\ x_{21}^i & x_{22}^i & \cdots & x_{2k}^i \\ \vdots & \vdots & \cdots & \vdots \\ x_{L1}^i & x_{L2}^i & \cdots & x_{Lk}^i \end{bmatrix} \in \Re^{L\times k}. \tag{15}$$

By using repeatedly the way as mentioned above, for all the surveyed bearing fault statuses we get a database in the form of the matrix $\overline{\mathrm{D}} \in \Re^{NL\times k}$described as in (16)

$$\overline{\mathrm{D}} = \begin{bmatrix} x^i_{11} & x^i_{12} & \cdots & x^i_{1k} \\ x^i_{21} & x^i_{22} & \cdots & x^i_{2k} \\ \vdots & \vdots & \cdots & \vdots \\ x^i_{L1} & x^i_{L2} & \cdots & x^i_{Lk} \\ \vdots & \vdots & \cdots & \vdots \\ x^N_{11} & x^N_{12} & \cdots & x^N_{1k} \\ x^N_{21} & x^N_{22} & \cdots & x^N_{2k} \\ \vdots & \vdots & \cdots & \vdots \\ x^N_{L1} & x^N_{L2} & \cdots & x^N_{Lk} \end{bmatrix}^{(train)}_{(NL\times k)}. \tag{16}$$

Building the ODS:

Each of the N damage statuses is encoded by a real number. For example, the output of the database according to the j-th damage status is encoded by vector $\bar{\mathrm{y}}_j \in \Re^{L\times 1}$ whose elements to be y_j as in (17)

$$\bar{\mathrm{y}}_j = [y_j, ..., y_j]^T \in \Re^{L\times 1}, \quad j = 1...N. \tag{17}$$

The database:

As a result, thus, we obtain the database named Off_DaB for training the ANFIS as follows

$$\text{Off\_DaB} \equiv [\text{IDS} - \text{ODS}] \equiv [\bar{\mathrm{D}} - \bar{\mathrm{y}}], \tag{18}$$

where, $\bar{\mathrm{D}}$ comes from (16) while $\bar{\mathrm{y}}$ is constituted of $\bar{\mathrm{y}}_j \in \Re^{L\times 1}$ mentioned in (17) as below

$$\bar{\mathrm{y}} = [\underbrace{y_1, ...y_1}_{L}, ..., \underbrace{y_N, ...y_N}_{L}]^T. \tag{19}$$

3.1.2. Training the ANFIS

Based on the database Off_DaB (18) and the algorithm EFIN-ANFIS as presented in Subsection 2.3.2, we obtain cluster centroids $\bar{\mathrm{x}}^0_i = (x^0_{i1}, ..., x^0_{in})$, and index vectors $\mathrm{a} = [a_0, a_1, ..., a_n]^T = [a_0, \bar{\mathrm{a}}]^T$ of output hyper-planes $w_i(.)$, to which the ANFIS's output $\hat{y}_i$, $1 \le i \le C$, are specified.

3.2. Estimating bearing fault status

Bearing damage status is carried out online based on the ANFIS trained in the offline phase. By using the method of building the database Off_DaB (18) mentioned above, during the system operating process, after each default time period, a similar database named On_DaB is established. The input data space of the On_DaB is then utilized for the ANFIS trained in the offline phase as shown in Subsection 1.1.2 to calculate the output of the ANFIS $\hat{y}_i,\ i = 1...NL$. The compared error between the corresponding data and encoding outputs are the information for specifying the bearing real status. Namely, bearing status at this time is the one encoded by 'q' which satisfies the condition as in (20)

$$\sum\nolimits_{i=1}^{L} |\hat{y}_i - y_q| = \min_{h=1...N} \sum\nolimits_{i=1}^{L} |\hat{y}_i - y_h|. \tag{20}$$

4. VARIFYING THE METHOD

4.1. Experimental apparatus and estimating way

Fig. 3 shows the experimental apparatus for measuring vibration signal of the mechanical system used for surveys. In the apparatus, (1) is motor, (2) and (4) are acceleration sensors, (3) and (5) are the surveyed bearings, (6) is the module for processing and transforming series vibration signal incorporate software-selectable AC/DC coupling (Model: NI-9234), while (7) is computer for installing NI-9234 Driver as well as storing signal from the sensors.

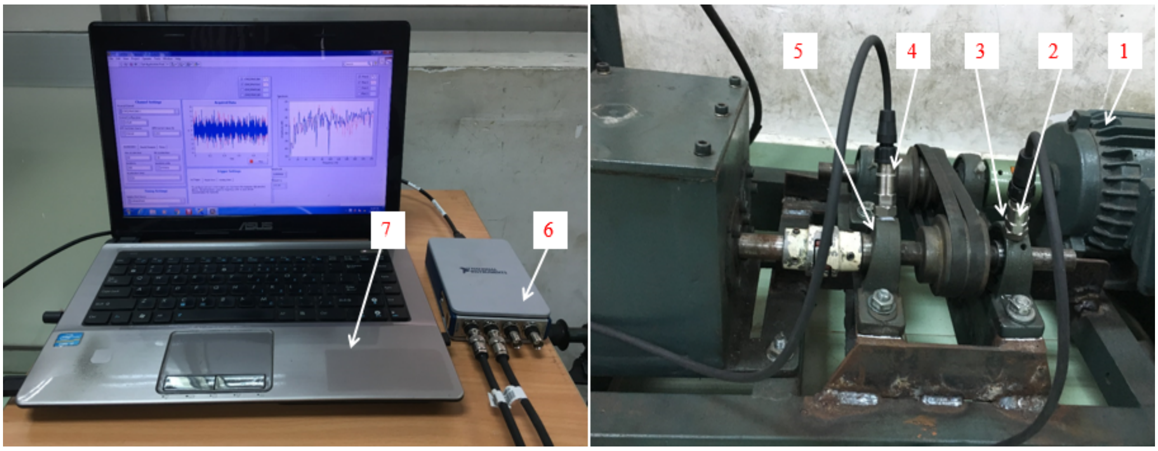

Figure 3. Experimental apparatus for measuring vibration signal

In order to estimate the effectiveness, the surveys using the proposed method along with two other methods were implemented. The first method was the one presented by Nguyen et al. (2013) in [10]. Related to this application, it should be noted that firstly we used the average quantity solution to wavelet transform coefficient (AQWTC) to build a database which reflects dynamic response of the mechanical system described in Fig. 3 corresponding to all considered bearing damage statuses; the next steps were followed the ASBDIM. The second shown by Lei et al. (2016) was the intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data shown in [14]. Besides, percentage of correct estimating samples calculated as in (21) and mean accuracy as in (22) were used

$$\text{Accuracy} = 100 \times cr\_samples_n / to\_samples_n \ \ (\%), \tag{21}$$

$$\text{Mean accuracy} = 100 \times \sum_{n=1}^{Q} cr\_samples_n \Big/ \sum_{n=1}^{Q} to\_samples_n \ \ (\%), \tag{22}$$

where, corresponding to the n-th damage level, $n = 1...Q$, $cr\_samples_n$ denoted the number of checking samples expressing correctly the real status of the bearings, while $to\_samples_n$ was the total of checking samples used in the survey; Q was the number of damage levels to be separated to consider.

Table 1. Four surveyed cases and the encoded values for expressing damage statuses

Surveyed Cases							
Case 1		Case 2		Case 3		Case 4	
Status	EV (y_i)	Status	EV (y_i)	Status	EV (y_i)	Status	EV (y_i)
L1Und	0	L2Und	0	L3Und	0	L1Und	0
L1D1Ba	1	L2D1Ba	1	L3D1Ba	1	L1D1In	1
L1D1In	2	L2D1In	2	L3D1In	2	L1D2In	2
L1D1Ou	3	L2D1Ou	3	L3D1Ou	3	L1D3In	3
L1D2Ba	4	L2D2Ba	4	L3D2Ba	4	L1D1Ou	4
L1D2In	5	L2D2In	5	L3D2In	5	L1D2Ou	5
L1D2Ou	6	L2D2Ou	6	L3D2Ou	6	L1D3Ou	6
L1D3Ba	7	L2D3Ba	7	L3D3Ba	7	-	-
L1D3In	8	L2D3In	8	L3D3In	8	-	-
L1D3Ou	9	L2D3Ou	9	L3D3Ou	9	-	-

4.2. Survey results

In this subsection we show the results obtained from four surveyed cases detailed in Tables 1 and 2. It is noted that in the tables, 'Encoding value' is abbreviated to 'EV'; LmDnBa, LmDnIn, and LmDnOu respectively define load degree to be m, damage level to be n, and damage location to be at the Ball, or Inner, or Outer of the bearing; while LmUnd shows load degree to be m, and the bearing is undamaged. Damage degree (n) and its location are provided in Table 2.

In each case we calculated Accuracy (21) and Mean accuracy (22) to establish Figs. 4-6 and Tables 3-4.

Table 2. Crack sizes

Surveyed Cases							
Case 1		Case 2		Case 3		Case 4	
Status	EV (y_i)	Status	EV (y_i)	Status	EV (y_i)	Status	EV (y_i)
L1Und	0	L2Und	0	L3Und	0	L1Und	0
L1D1Ba	1	L2D1Ba	1	L3D1Ba	1	L1D1In	1
L1D1In	2	L2D1In	2	L3D1In	2	L1D2In	2
L1D1Ou	3	L2D1Ou	3	L3D1Ou	3	L1D3In	3
L1D2Ba	4	L2D2Ba	4	L3D2Ba	4	L1D1Ou	4
L1D2In	5	L2D2In	5	L3D2In	5	L1D2Ou	5
L1D2Ou	6	L2D2Ou	6	L3D2Ou	6	L1D3Ou	6
L1D3Ba	7	L2D3Ba	7	L3D3Ba	7	-	-
L1D3In	8	L2D3In	8	L3D3In	8	-	-
L1D3Ou	9	L2D3Ou	9	L3D3Ou	9	-	-

Table 3. The compared results corresponding to each case as depicted in Table 1

Surveyed Cases	Mean accuracy (%)		
	[14]	[10]	**Proposed**
Case 1	90.99	89.98	**93.17**
Case 2	90.53	91.65	**92.19**
Case 3	96.18	94.52	**96.20**

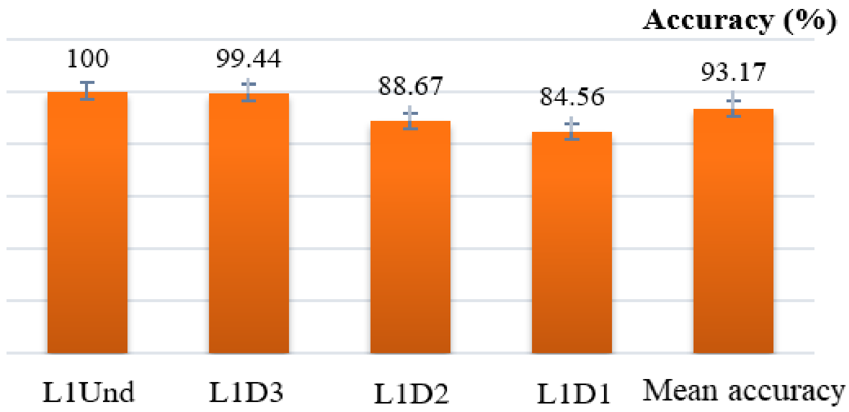

Figure 4. The proposed-method's accuracy in the first case which is depicted in Table 1 corresponding to the load degree 1

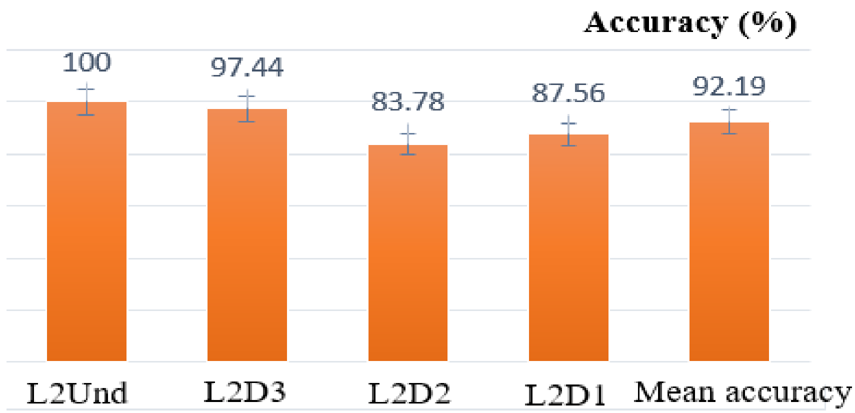

Figure 5. The proposed-method's accuracy in the second case which is depicted in Table 1 corresponding to the load degree 2

4.3. Discussion

It can observe that in these surveys the proposed method as well as the others can specify correctly whether existing fault in the bearings, such as in case the bearing is undamaged, the Accuracy (21) of all three methods are 100%. Otherwise, in case the bearing exists fault, especially when fault is lowest (D1, see Table 2), there often exist some incorrect conclusions related to the different damage degrees. However the comparative effectiveness in terms of the Mean accuracy estimated by (22) of the proposed method is always the best. For example, for case 1, this parameter can be seen from Table 3, corresponding to [10, 14], and the proposed ASBDIM respectively to be 90.99, 89.98, and 93.17%; or in case 4, the values of the Accuracy (21) related to [10, 14] and the proposed respectively are 97.53, 96.25, and 98.90 which are provided by Table 4.

Another aspect to be the relation between the accuracy and the load can be also observed via Figures 4-6. Although the accuracy of the proposed method is dependent on the load,

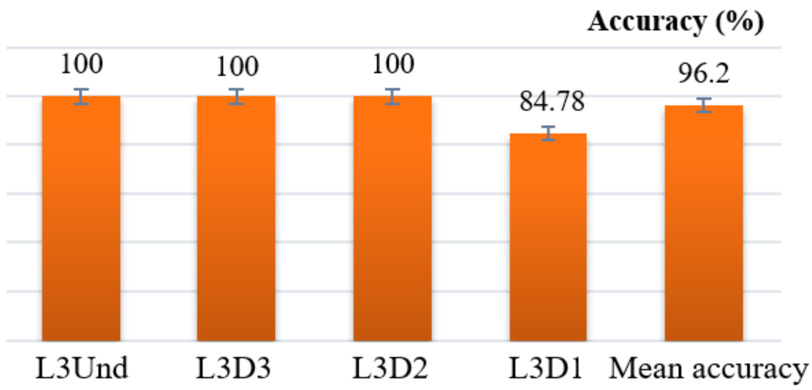

Figure 6. The accuracy of the proposed method in the third case which is depicted in Table 1 corresponding to the load degree 3

Table 4. The Accuracy (21) of surveyed methods related to the fourth case depicted in Table 1

Status	Accuracy (%)		
	[14]	[10]	Proposed
L1UnDad	100	100	100
L1D1In	97.28	100	98.12
L1D2In	100	98.15	97.35
L1D3In	96.57	94.55	100
L1D1Ou	97.91	92.22	98.88
L1D2Ou	96.42	94.87	98.95
L1D3Ou	94.54	93.98	98.99
Mean accuracy	**97.53**	**96.25**	**98.90**

this relation rate is not much.

5. CONCLUSION

The methodASBDIM for identifying online bearing fault base SSA, sparse filtering and ANFIS has been presented. The extracting features from a big data stream and the mechanism of two offline and online phases were described. SSA and sparse filtering were combined to build the database Off_DaB in the offline phase. Bearing fault statuses were then identified by the ANFIS using the Off_DaB. In order to check online bearing status, the online period was then deployed consisting of 1) building online the database On_DaB based on the way similar to the one for building the Off_DaB, and 2) comparing the output of the trained ANFIS corresponding to the input coming from the On_DaB and the encoded output. The survey results showed that the ASBDIM could address correctly the undamaged bearing situations, and quite correctly the fault degrees appearing in the bearing even if it was smallest (D1 as in Table 2). The results compared with that from [10] and [14] also reflect the ASBDIM's better ability. The proposed method could be exploited becomingly to the online applications, to which it was not too sensitive to load.

ACKNOWLEDGMENTS

This research was funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 107.01-2016.22.

REFERENCES

[1] W. Zhou, T. Habetler, R. Harley, "Bearing fault detection via stator current noise cancellation and statistical control," *IEEE Trans. Ind. Electron.*, vol. 55, no. 12, pp. 4260–4269, 2008.

[2] A. Giantomassi, "Electric motor fault detection and diagnosis by kernel density estimation and Kullback–Leibler divergence based on stator current measurements," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1770–1780, 2015.

[3] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal based approaches," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3757–3767, 2015.

[4] F. Filippetti, A. Bellini, and G. A. Capolino, "Condition monitoring and diagnosis of rotor faults in induction machines: State of art and future perspectives," in *Proc. IEEEWorkshop Electr.Mach. Des. Control Diagn.*, Paris, France, pp. 196–209, 2013.

[5] J. Pons-Llinares, J. A. Antonino-Daviu, M. Riera-Guasp, S. B. Lee, T. J. Kang, C. Yang "Advanced induction motor rotor fault diagnosis via continuous and discrete time–frequency tools," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1791–1802, 2015.

[6] D. Z. Li, W. Wang, and F. Ismail, "An enhanced bispectrum technique with auxiliary frequency injection for induction motor health condition monitoring," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 10, pp. 2279–87, 2015.

[7] Tahar Boukra, "Identifying new prognostic features for remaining useful life prediction using particle filtering and Neuro-fuzzy system predictor," 978-1-4799-7993-6/15/$31.00 ©2015 IEEE, 2015.

[8] Turker Ince, Serkan Kiranyaz, Levent Eren, Murat Askar, Moncef Gabbouj, "Real-time motor fault detection by 1-d convolutional neural networks," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, 2016.

[9] X. Dai and Z. Gao, "From model, signal to knowledge: A data-driven perspective of fault detection and diagnosis," *IEEE Trans. Ind. Informat.*, vol. 9, no. 4, pp. 2226–2238, 2013.

[10] Sy Dzung Nguyen, K. N. Ngo, Q. T. Tran, Seung-Bok Choi, "A new method for beam-damage-diagnosis using adaptive fuzzy neural structure and wavelet analysis , *Mechanical System and Signal Process*, vol. 39, pp. 181–194, 2013.

[11] Sy Dzung Nguyen, Quang Thinh Tran, Kieu Nhi Ngo, "A fuzzy logic system built based on fuzzy data clustering and differential evolution for fault diagnosis ," in *16th Asia Pacific Vibration Conference 24-26 November, HUST*, Ha Noi, Viet Nam, 2015.

[12] Sy Dzung Nguyen, Q. H. Nguyen, K. N. Ngo, X. P. Do, Seung Bok Choi, "A structure damage-locating method based on wavelet analysis and type-2 fuzzy logic system ," in *Proc. of SPIE,* vol. 9057 905727-1, USA, 2014.

[13] Silvio Simani, Saverio Farsoni, Paolo Castaldi, "Faultdiagnosis of a wind turbine benchmark via identified fuzzy models ", *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, 2015.

[14] Yaguo Lei, Feng Jia, Jing Lin, Saibo Xing, Steven X. Ding, "An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data ," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 5, 2016.

[15] Sy Dzung Nguyen, Seung-Bok Choi, Tae-Il Seo, "Recurrent mechanism and impulse noise filter for establishing ANFIS," *IEEE Transactions on Fuzzy Systems.* DOI 10.1109/TFUZZ.2017.2701313, 2017.

[16] J. Mao, J. Zhang, Y. Yue, H. Ding, "Adaptive-tree-structure-based fuzzy inference system", *IEEE Trans. Fuzzy Syst.*, vol. 13, pp. 1–12, 2005.

[17] Sy Dzung Nguyen, Quoc Hung Nguyen, Tae-Il Seo, "ANFIS deriving from jointed input-output data space and applying in smart-damper identification ," *Applied Soft Computing*, vol. 53, pp. 45–60, 2017.

[18] Sy Dzung Nguyen, Tae-Il Seo, "Establishing ANFIS and the use for predicting sliding control of active railway suspension systems subjected to uncertainties and disturbances ", *International J. of Machine Learning and Cybernetics*, DOI 10.1007/s13042-016-0614-z, 2016.

[19] Sy Dzung Nguyen, Seung-Bok Choi, "Design of a new adaptive neuro-fuzzy inference system based on a solution for clustering in a data potential field," *Fuzzy Sets and Systems*, http://dx.doi.org/ 10.1016/j.fss.2015.02.012, pp. 1–23, 2015.

[20] Sy Dzung Nguyen, Q. H. Nguyen, Seung-Bok Choi, "Hybrid clustering based fuzzy structure for vibration control – Part 1: A novel algorithm for building neuro-fuzzy system," *Mechanical Systems and Signal Processing*, vol. 50-51, pp. 510–525, 2015.

[21] W. Q. Wang, M. Farid Golnaraghi, F. Ismail "Prognosis of machine health condition using neuro-fuzzy systems," *Mechanical Systems and Signal Processing* , vol.18, pp. 813–831, 2004.

[22] F. Zhao, J. Chen, L. Guo, X. Lin, "Neuro-fuzzy based condition prediction of bearing health," *J. Vib. Control*, vol. 15, no. 7, pp. 1079–1091, 2009.

[23] C. Chen, B. Zhang, G. Vachtsevanos, , M. Orchard, "Machine condition prediction based on adaptive neuro-fuzzy and high- order particle filtering ", *IEEE Transaction on Instrumentation and Measurement*, vol. 61, no. 2, pp. 297–306, 2012.

[24] Golyandina, V. Nekrutkin, A. Zhigljavsky, *Analysis of Time Series Structure—SSA and Related Techniques*, Chapman & Hall/CRC, Boca Raton, Florida, 2001.

[25] Salgado D.R., F.J. Alonso, "Tool wear detection in turning operations using singular spectrum analysis," *Journal of Materials Processing Technology*, vol. 171, pp. 451–458, 2006.

[26] Bovic Kilundu, Pierre Dehombreux, Xavier Chiementin, "Tool wear monitoring by machine learning techniques and singular spectrum analysis," *Mechanical Systems and Signal Processing*, vol. 25, pp. 400–415, 2011.

[27] B. Willmore, D. J. Tolhurst, "Characterizing the sparseness of neural codes", *Netw.: Comput. Neural Syst.*, vol. 12, no. 3, pp. 255–270, 2001.

[28] J. Ngiam, Z. Chen, S. A. Bhaskar, P. W. Koh, and A. Y. Ng, "Sparse filtering", in *Proc. Neural Inf. Process. Syst.*, pp. 112–1133, 2011.

Received on October 14, 2017
Revised on December 06, 2017