

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# ANFIS: Establishing and Applying to Managing Online Damage

*Sy Dzung Nguyen*

## Abstract

Fuzzy logic (FL) and artificial neural networks (ANNs) own individual advantages and disadvantages. Adaptive neuro-fuzzy inference system (ANFIS), a fuzzy system deployed on the structure of ANN, by which FL and ANN can interact to not only overcome their limitations but also promote the ability of each model has been considered as a reasonable option in the real fields. With the vital strong points, ANFIS has been employed well in many technology applications related to filtering, identifying, predicting, and controlling noise. This chapter, however, focuses mainly on building ANFIS and its application to identifying the online bearing fault. First, a traditional structure of ANFIS as a data-driven model is shown. Then, a recurrent mechanism depicting the relation between the processes of filtering impulse noise (IN) and establishing ANFIS from a noisy measuring database is presented. Finally, one of the typical applications of ANFIS related to online managing bearing fault is shown.

**Keywords:** fuzzy logic, artificial neural networks, adaptive neuro-fuzzy inference system

## 1. Introduction

As well known, the mathematical tools FL and ANN possess both the advantages and disadvantages as their specific characteristics. The hybrid structure ANFIS, where ANN and FL can interact to not only overcome partly the limitations of each model but also uphold their strong points [1–25], is, hence, considered as a reasonable option in many technology applications such as identifying [1–2, 4, 6, 12], predicting [9, 11, 17, 25], controlling [3, 5, 7, 18–26], and filtering noise [14–16, 27–29].

To build an ANFIS from a given database, firstly, an initial data space (IDS) expressing the mapping  $f : X \rightarrow Y$  must be created. A cluster data space (CDS) is then built from the IDS to form the ANFIS via a training algorithm. Being viewed as a popular technique for unsupervised pattern recognition, clustering is an effective tool for analyzing and exploring data structures to build CDSs [30–37]. Reality shows that the accuracy and training time of the ANFIS depend deeply on the features of both the IDS and CDS [2–6]. In the process of building ANFIS, the two issues as follows should be considered: (1) What is the essence of the interactive relation between ANFIS's convergence capability and CDS' attributes? (2) How to exploit this essence for increasing ANFIS's ability to converge to the desired accuracy with the improved calculating cost?

Many different clustering approaches have been discovered [2–4, 10, 30–34, 37]. Separating data in X and in Y distinctly with a mutual result reference, step by step,

was described in [10]. The method, however, could not solve appropriately the above issues. Besides, the difficulty in deploying fuzzy clustering strategies along with the high calculating cost was their disadvantage. Generally, a hard relation could not reflect fully database attributes [31, 34]. The well-known method of fuzzy C-means clustering was seen as a better option in this case. It, however, was not effective enough for the “non-spherical” general datasets [30, 37]. Therefore, the idea of fuzzy clustering in a kernel feature space was then developed to deal with these cases [30–34, 37]. A weighted kernel-clustering algorithm could be referred to [30], or a method of weighted kernel fuzzy C-means clustering based on adaptive distances was detailed in [31]. In spite of owning considerable advantages, the identification and prediction accuracy of the ANFIS based on the CDS coming from [30–31] are sensitive to attributes of the CDS due to the negative influence of noise.

Reality has shown that noise status including IN always exists in the measured IDSs [2, 4, 9, 16], which degrades violently the accuracy of ANFISs deriving from them. There are many reasons resulting in this, such as the lack of precision of the measurement devices, tools, measurement methods, or the negative impact of the surrounding environment. In [7], an ANFIS took part in the system in the form of an inverse MR damper (MRD) model to specify the time-verifying desired control current. To maintain the accuracy of the inverse MRD model, the ANFIS was retrained after each certain period due to the dynamic response of the MRD depending quite deeply on temperature. Another more active approach is filtering noise or preprocessing data [7, 9, 11, 17, 21, 38–40]. In [11, 17], where ANFISs were employed to predict the health of mechanical systems, vibration signal was always measured and filtered to update the ANFISs. Related to the preprocessing data to set up ANFIS, it can observe that to maintain the stability of the above online ANFIS-based applications, reducing time delay is really meaningful. One of the becoming solutions for this can be referred to [16] where filtering IN and building ANFIS were carried out synchronously via a recurrent mechanism. A recurrent strategy for forming ANFIS was carried out, in which the capability to converge to a desired accuracy of the ANFIS training process could be estimated and directed online. As a solution, increasing the quality of both the IDS and CDS was paid attention. Building an ANFIS via a filtered database and exploiting the ANFIS as an updated filter to refilter the database were depicted via an online and recurrent mechanism. The process was upholden until the ANFIS-based database approximation convergent to the desired accuracy or a stop condition appears.

Inspired by the ANFIS’s capability, in order to provide the readers with the theoretical basis and application direction of the model, this chapter presents the formulation of ANFIS and one of its typical applications. The rest of the chapter is organized as follows. Section 2 shows a structure of ANFIS as a data-driven model deriving from fuzzy logic and artificial neural networks. Setting up the CDS consisting of the input data clusters, output data clusters, and the CDS-based ANFIS as a jointed structure is all detailed. Deriving from this relation, a theoretical basis for building ANFIS from noisy measuring datasets is presented in Section 3. An online and recurrent mechanism for filtering noise and building ANFIS synchronously is clarified via algorithms for filtering IN and establishing ANFIS. A typical application of ANFIS related to online managing bearing fault status is shown in Section 4. Finally, some general aspects are mentioned in the last section.

## 2. Structure of ANFIS

Let’s consider a given IDS having  $P$  input–output data samples  $(\bar{x}_i, y_i)$ ,  $\bar{x}_i = [x_{i1}, \dots, x_{in}] \in \mathfrak{R}^n$ ,  $y_i \in \mathfrak{R}^1$ , and  $i = 1 \dots P$ . With a data normalization solution and a

used certain clustering algorithm, a CDS is then created. The  $k$ th cluster, signed  $\Gamma^k$ ,  $k = 1...C$ , consists of one input cluster and one output cluster signed  $\Gamma^{k(A)}$  and  $\Gamma^{k(B)}$ , respectively. The CDS can be seen as a framework for establishing ANFIS. This section presents how to build the CDS as well as the CDS-based ANFIS structure.

## 2.1 Some related notions

Some notions shown in [16] are used in this chapter as follows.

Definition 1. Normalizing a given IDS to set up a normalized initial data space signed  $\overline{\text{IDS}}$  is performed as follows:

$$\tilde{x}_{ij} = x_{ij} / \max_k |x_{kj}|, \quad i, \quad k = 1...P, j = 1...n. \quad (1)$$

By this way, the  $i$ th data sample (also signed  $(\bar{x}_i, y_i)$ ) in the  $\overline{\text{IDS}}$  is constituted as follows:

$$(\bar{x}_i = [\tilde{x}_{i1}, \dots, \tilde{x}_{in}]^T, y_i) \quad i = 1...P. \quad (2)$$

Definition 2. The root-mean-square error (RMSE) in Eq. (3) is used to evaluate accuracy rate of ANFIS. The required RMSE value is signed  $[E]$ . The absolute error,  $\bar{e}_i$ ,  $i = 1...P$ , between the data output  $y_i = f(\bar{x}_i)$  and the corresponding ANFIS-based output  $\hat{y}_i(\bar{x}_i)$  is defined in Eq. (4). The desired value of  $\bar{e}_i$  is signed  $[\bar{e}]$ :

$$\text{RMSE} = \sqrt{P^{-1} \sum_{i=1}^P (\hat{y}_i(\bar{x}_i) - f(\bar{x}_i))^2}, \quad (3)$$

$$\bar{e}_i = |\hat{y}_i(\bar{x}_i) - f(\bar{x}_i)|, \quad i = 1...P. \quad (4)$$

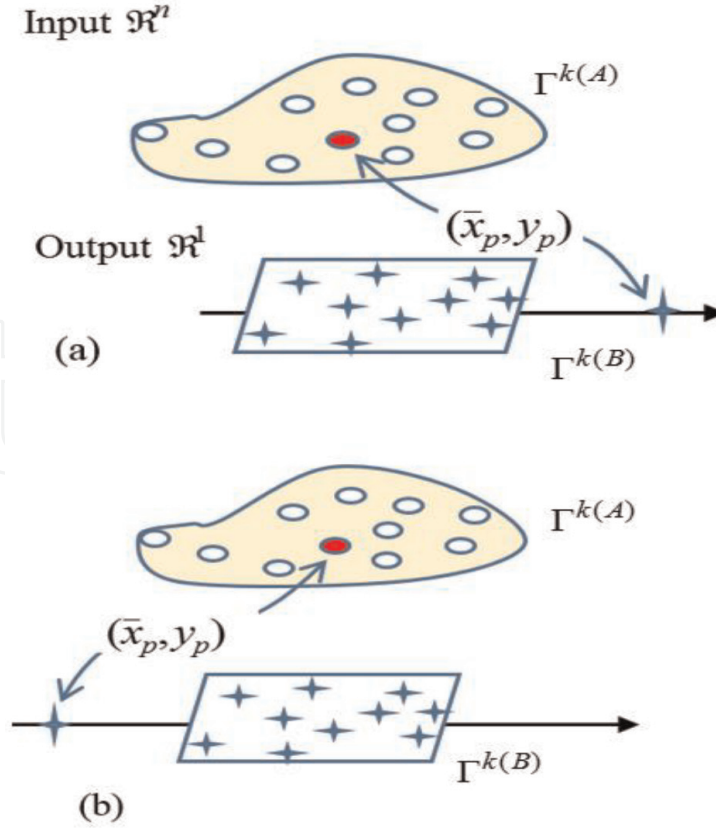
Definition 3. Let's consider  $\bar{x}_i \in \overline{\text{IDS}}$  in which  $\overline{\text{IDS}}$  depicts an unknown mapping  $f : X \rightarrow Y$ . The ANFIS-based approximation of  $f : X \rightarrow Y$  is called to be continuous at  $(\bar{x}_p, y_p) \in \overline{\text{IDS}}$  if.

$$\hat{y}_i(\bar{x}_i) \rightarrow y_p \pm [\bar{e}] \text{ when } \bar{x}_i \rightarrow \bar{x}_p. \quad (5)$$

Definition 4. Let's consider an ANFIS-based approximation of a mapping expressed by an  $\overline{\text{IDS}}$ . The ANFIS is said to be a uniform approximation with a required error  $[\bar{e}]$  if at  $\forall \bar{x}_i \in X$ , by choosing any small constant  $\varepsilon \geq [\bar{e}]$ , corresponding data sample  $\bar{x}_j \in \overline{\text{IDS}}$  always exists such that.

$$\forall \bar{x}_h \in \overline{\text{IDS}}; \text{ if } \|\bar{x}_h - \bar{x}_i\| \leq \|\bar{x}_j - \bar{x}_i\|, \text{ then } |\hat{y}_h(\bar{x}_h) - f(\bar{x}_i)| \leq |\hat{y}_j(\bar{x}_j) - f(\bar{x}_i)| = \varepsilon. \quad (6)$$

Definition 5. Data cluster  $\Gamma^k$  and data sample  $(\bar{x}_p, y_p) \in \Gamma^k$  in a CDS derived from an  $\overline{\text{IDS}}$  are depicted in **Figure 1**. Let sign  $\Gamma^k \setminus p$  to be a subset consisting of the data samples belonging to  $\Gamma^k$  except  $(\bar{x}_p, y_p)$ . The subset contains  $Q_{kp}$  data samples. It is assumed that all of data samples in  $\Gamma^{k(A)}$  are distributed closely, while in  $\Gamma^{k(B)}$ , most of them are located closely, except  $y_p$ ; it is far from the other and distributes at one side of  $\Gamma^{k(B)}$ . This status is described in Eq. (7):



**Figure 1.**

Two typical distribution types in data cluster  $\Gamma^k$ : Impulse noise point  $IN(\bar{x}_p, y_p) \in \Gamma^k$  causing the distribution at one side, the right side (a), and the left side (b).

$$y_p \gg \max_{y_i \in \Gamma^{k(B)} \setminus p} (y_i) \text{ or } y_p \ll \min_{y_i \in \Gamma^{k(B)} \setminus p} (y_i). \quad (7)$$

and satisfies Eqs. (8) and (9):

$$d_{k1} = y_p - \max_{y_i \in \Gamma^{k(B)} \setminus p} (y_i) > (P[E]^2 - Q_{kp}[\bar{E}]^2)^{0.5} \quad \text{if } y_p > \max_{y_i \in \Gamma^{k(B)} \setminus p} (y_i). \quad (8)$$

$$d_{k2} = \min_{y_i \in \Gamma^{k(B)} \setminus p} (y_i) - y_p > (P[E]^2 - Q_{kp}[\bar{E}]^2)^{0.5} \quad \text{if } y_p < \min_{y_i \in \Gamma^{k(B)} \setminus p} (y_i). \quad (9)$$

In this case,  $(\bar{x}_p, y_p)$  is called a *critical data sample* in the CDS.

## 2.2 Setting up the input data clusters

Let's consider the normalized initial data space  $\overline{\text{IDS}}$  (see Def. 1). Many well-known clustering methods can be used to build a CDS from the  $\overline{\text{IDS}}$ . Here, the CDS is built by using the clustering algorithm KFCM-K (kernel fuzzy C-means clustering with kernelization of the metric) presented in [31]. By this way, distribution of data samples in the CDS is established. The membership degree of the  $j$ th data sample belonging to the  $i$ th cluster is denoted by  $\mu_{ij} \in [0, 1] \forall i, j$  and  $j = 1 \dots P, i = 1 \dots C$ . Cluster centroids  $\bar{x}_1^0, \dots, \bar{x}_C^0$  in the CDS are specified such that the following objective function is minimized:



$$J_{KFCM}(U, \bar{x}^0) = \sum_{i=1}^C \sum_{j=1}^P \mu_{ij}^m \|\phi(\bar{x}_j) - \phi(\bar{x}_i^0)\|^2 \quad (10)$$

subjected to  $\sum_{i=1}^C \mu_{ij} = 1 \forall j$  and  $\mu_{ij} \in [0, 1] \forall i, j$ . In Eq. (10),  $\bar{x}_i^0 = [x_{i1}^0, \dots, x_{in}^0] \in \mathfrak{R}^n$  is the  $i$ th cluster center;  $\|\phi(\bar{x}_j) - \phi(\bar{x}_i^0)\|^2$  denotes the squared distance between  $\bar{x}_j$  and  $\bar{x}_i^0$  in the kernel space;  $\phi(\cdot)$  is the kernel function;  $U = U(\mu_{ij}) \in \mathfrak{R}^{(C \times P)}$  is the distribution matrix; and  $m > 1$  is the fuzzy factor.

The objective function can be rewritten via Gaussian kernel function as follows:

$$J_{KFCM}(U, \bar{x}^0) = 2 \sum_{i=1}^C \sum_{j=1}^P \mu_{ij}^m \left( 1 - \exp \left( -\|\bar{x}_j - \bar{x}_i^0\|^2 / \sigma^2 \right) \right). \quad (11)$$

Deriving  $J_{KFCM}(U, \bar{x}^0)$  in Eq. (11) with respect to  $\bar{x}_i^0$ , at the optimal centers, the following must be obtained:

$$\frac{\partial}{\partial \bar{x}_i^0} J_{KFCM}(U, \bar{x}_i^0) = \frac{-4}{\sigma^2} \sum_{j=1}^P \mu_{ij}^m (\bar{x}_j - \bar{x}_i^0) \exp \left( -\|\bar{x}_j - \bar{x}_i^0\|^2 / \sigma^2 \right) = 0 \quad (12)$$

From Eqs. (11) to (12) and the use of Lagrange multipliers with  $\mu_{ij} \in [0, 1] \forall i, j$  and  $\sum_{i=1}^C \mu_{ij} = 1 \forall j$ , the following update laws are obtained:

$$\bar{x}_i^0 = \frac{\sum_{j=1}^P \mu_{ij}^m \bar{x}_j K(\bar{x}_j, \bar{x}_i^0)}{\sum_{j=1}^P \mu_{ij}^m K(\bar{x}_j, \bar{x}_i^0)}, i = 1 \dots C. \quad (13)$$

$$\mu_{ij} = \begin{cases} \left[ \left( \frac{\sum_{h=1}^C \frac{1 - K(\bar{x}_j, \bar{x}_i^0)}{1 - K(\bar{x}_j, \bar{x}_h^0)} \right)^{1/(m-1)} \right]^{-1} & \text{if } \bar{x}_j \neq \bar{x}_i^0 \\ 1 \quad \left( \text{and } \mu_{ik(k \neq j)} = 0 \right) & \text{if } \bar{x}_j = \bar{x}_i^0 \end{cases} \quad (14)$$

$i = 1 \dots C; j = 1 \dots P.$

By using index  $ts$  as in Eq. (15),  $[ts]$  to be the required value of  $ts$  and  $r$  to denote the  $r$ th loop, the clustering phase is accomplished until  $ts \leq [ts]$ :

$$ts = \left( J_{KFCM}^{(r)} - J_{KFCM}^{(r-1)} \right) / J_{KFCM}^{(r-1)} \quad (15)$$

Specification of the optimal centers and their relationship values as abovementioned is detailed in Appendix A of [12].

### 2.3 Setting up the output data clusters

The result of the clustering process in the input data space is an input cluster centroid vector  $[\bar{x}_1^0, \dots, \bar{x}_C^0]$  of corresponding data clusters, respectively, signed as  $\Gamma^1, \dots, \Gamma^C$ . Let  $A^1, \dots, A^C$ , respectively, be input fuzzy sets established via  $\bar{x}_1^0, \dots, \bar{x}_C^0$  [12, 16]. The membership value of  $\tilde{x}_{il}$  belonging to  $A^k$  is inferred from Eq. (14):

$$\bar{\mu}_{ki}(\tilde{x}_{il}) = \left[ \left( \sum_{h=1}^C \frac{1 - K(\tilde{x}_{il}, x_{kl}^0)}{1 - K(\tilde{x}_{il}, x_{hl}^0)} \right)^{1/(m-1)} \right]^{-1} \quad (16)$$

( $k = 1 \dots C; i = 1 \dots P; l = 1 \dots n.$ )

With following the product law, membership value of  $\bar{x}_q$  belonging to  $A^i$  is

$$\mu_{kq}(\bar{x}_q) = \prod_{l=1}^n \bar{\mu}_{kl}(\tilde{x}_{ql}), \quad k = 1 \dots C, \quad q = 1 \dots P, \quad (17)$$

and its normalized membership value is as follows:

$$N_k(\bar{x}_q) = \mu_{kq}(\bar{x}_q) / \sum_{h=1}^C \mu_{hq}(\bar{x}_q), \quad q = 1 \dots P, k = 1 \dots C. \quad (18)$$

The membership of a data sample in each cluster determined based on Eqs. (16)–(18) is then used to specify the hard distribution status of the data samples in each cluster. It is then used to specify the index vector  $\mathbf{a}$  of hyperplanes (or the output data clusters)  $w_k(\cdot)$  and  $k = 1 \dots C$ . The  $i$ th data sample is hardly distributed into the  $k$ th data cluster if

$$N_k(\bar{x}_i) = \max_{h=1 \dots C} (N_h(\bar{x}_i)), \quad i = 1 \dots P, \quad k = 1 \dots C. \quad (19)$$

Deriving from the  $t_k$  data samples hardly distributed in the  $k$ th data cluster, by using the least mean squares method, vector  $\mathbf{a} = [a_0, a_1, \dots, a_n]^T = [a_0, \bar{\mathbf{a}}]^T$  of  $w_k(\cdot)$  is specified which is the solution of Eq. (20):

$$\begin{cases} a_n \sum_{i=1}^{t_k} \tilde{x}_{in} + a_{n-1} \sum_{i=1}^{t_k} \tilde{x}_{i(n-1)} + \dots + a_1 \sum_{i=1}^{t_k} \tilde{x}_{i1} + a_0 t_k = \sum_{i=1}^{t_k} y_i \\ a_n \sum_{i=1}^{t_k} \tilde{x}_{in} \tilde{x}_{i1} + a_{n-1} \sum_{i=1}^{t_k} \tilde{x}_{i(n-1)} \tilde{x}_{i1} + \dots + a_1 \sum_{i=1}^{t_k} \tilde{x}_{i1}^2 + a_0 \sum_{i=1}^{t_k} \tilde{x}_{i1} = \sum_{i=1}^{t_k} y_i \tilde{x}_{i1} \\ \vdots \\ a_n \sum_{i=1}^{t_k} \tilde{x}_{in}^2 + a_{n-1} \sum_{i=1}^{t_k} \tilde{x}_{i(n-1)} \tilde{x}_{in} + \dots + a_1 \sum_{i=1}^{t_k} \tilde{x}_{i1} \tilde{x}_{in} + a_0 \sum_{i=1}^{t_k} \tilde{x}_{in} = \sum_{i=1}^{t_k} y_i \tilde{x}_{in}. \end{cases} \quad (20)$$

Finally, the value of hyperplane  $w_k$  corresponding to  $\bar{x}_i$  is calculated in Eq. (21):

$$w_k(\bar{x}_i) = a_0 + \bar{\mathbf{a}}^T \bar{\mathbf{x}}_i \quad (21)$$

## 2.4 Structure of ANFIS

As mentioned in Eq. (1), the ANFIS for approximating the mapping  $f : X \rightarrow Y$  is derived from  $M$  fuzzy laws in Eq. (22):

$$R^{(i)} : \text{IF } \tilde{x}_{q1} \text{ is } A_1^i(\tilde{x}_{q1}), \text{ AND } \dots, \text{ AND } \tilde{x}_{qn} \text{ is } A_n^i(\tilde{x}_{qn}) \text{ THEN } y_q^i \text{ is } B^i(\bar{x}_q) \quad (22)$$

( $i = 1 \dots M, M \equiv C$ ),

where  $A_l^i(\tilde{x}_{ql})$ ,  $i = 1 \dots M, l = 1 \dots n$ , is the membership value of  $\tilde{x}_{ql}$  belonging to input fuzzy set  $A^i$ , meaning  $A_l^i(\bar{x}_q) \equiv \bar{\mu}_{il}(\tilde{x}_{ql})$  in Eq. (16);  $B^i(\bar{x}_q)$  is the corresponding output fuzzy set of data sample  $\bar{x}_q$ ,  $\bar{x}_q = [\tilde{x}_{q1}, \dots, \tilde{x}_{qn}]^T$ ,  $q = 1 \dots P$ .

In the fuzzification phase, membership value of  $\bar{x}_q$  belonging to input fuzzy set  $A^i$  signed  $A^i(\bar{x}_q) \equiv \mu_{iq}(\bar{x}_q)$  is specified by Eq. (17). For the defuzzification, if the center-average method is used, the output of the  $q$ th data sample is expressed via the membership values in the input fuzzy space of  $\bar{x}_q$  as follows:

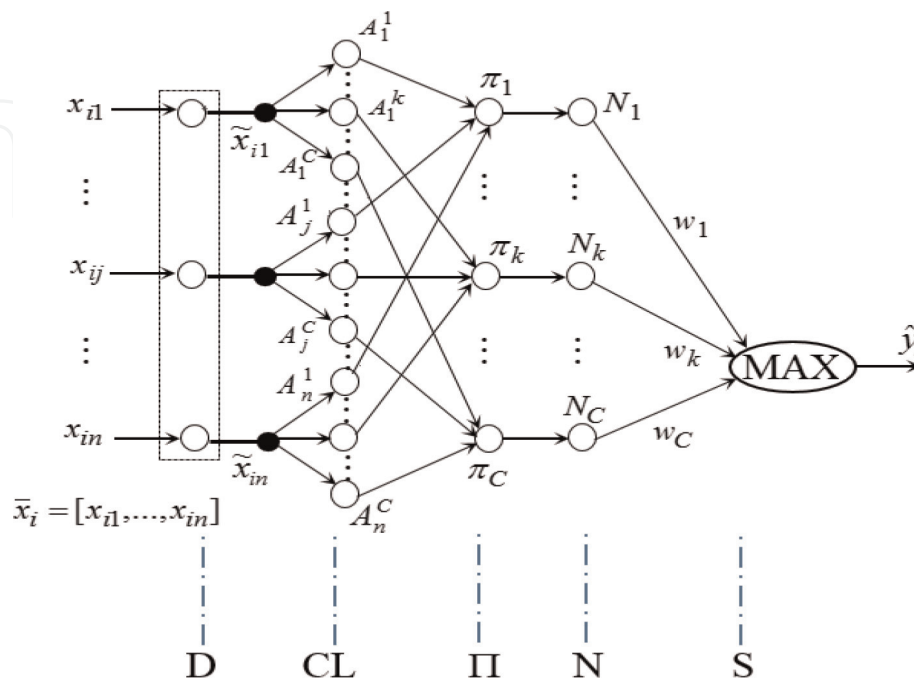
$$\hat{y}_q(\bar{x}_q) = \frac{\sum_{i=1}^M y_q^i \mu_{iq}(\bar{x}_q)}{\sum_{i=1}^M \mu_{iq}(\bar{x}_q)} \quad (23)$$

where  $y_q^i = w_i(\bar{x}_q)^i$  is the value of hyperplane  $w_i$  corresponding to data sample  $\bar{x}_q$  calculated in Eq. (21).

Finally, all the above-mentioned contents can be depicted via the ANFIS with five layers signed D, CL,  $\Pi$ , N, and S in **Figure 2**. Layer D (data) has  $n$  input nodes corresponding to  $n$  elements of data vector  $\bar{x}_i = [x_{i1}, \dots, x_{in}]^T$ ,  $i = 1 \dots P$ , while its outputs are the corresponding normalized values using Eq. (1). Layer CL (clustering) expresses the clustering process. The result of this process is  $C$  clusters with  $C$  corresponding cluster centroids  $\bar{x}_1^0, \dots, \bar{x}_C^0$ , to which  $C$  fuzzy sets,  $A^1, \dots, A^C$ , are given. The output of this layer is the membership value of  $\bar{x}_i$  calculated for each dimension  $(\tilde{x}_{i1}, \dots, \tilde{x}_{in})$  via Eq. (16). Layer  $\Pi$  (product layer) specifies membership values based on Eq. (17). Layer N (normalization) estimates the normalized membership value of a data sample belonging to each fuzzy set upon Eq. (18). Layer S (specifying) is used to estimate the output of the ANFIS based on any well-known method. In case of using the center-average defuzzification, it is calculated by Eq. (23), while it is specified by Eq. (24) if the “the winner takes all” law is employed:

$$\hat{y}_i = w_k(\bar{x}_i), \quad i = 1 \dots P, \quad (24)$$

where  $w_k(\bar{x}_i)$  is the value of the  $k$ th hyperplane corresponding to input data sample  $\bar{x}_i$  (21);  $k$  is the index of the data cluster where data sample  $\bar{x}_i$  gets the maximum membership specified via  $N_{(\cdot)}(\bar{x}_i)$  as in Eq. (25):



**Figure 2.**  
 Structure of the ANFIS.



$$N_k(\bar{x}_i) = \max_{h=1\dots C} (N_h(\bar{x}_i)) \quad (25)$$

### 3. Building ANFIS from a noise measuring database

This section presents the recurrent mechanism together with the related algorithms consisting of the one for ANFIS-based noise filtering and the one for building ANFIS showed in [16].

#### 3.1 Convergence condition of the ANFIS-based approximation

Deriving from a given IDS having  $P$  input–output data samples  $(\bar{x}_i, y_i)$ ,  $\bar{x}_i = [x_{i1}, \dots, x_{in}] \in \mathfrak{R}^n$  and  $y_i \in \mathfrak{R}^1$ ,  $i = 1\dots P$ , with a data normalization solution as in Def. 1, the  $\overline{\text{IDS}}$  is built, to which a CDS is created as depicted in Section 2. It should be noted that IN is often considered as disturbances distributed uniformly in a signal source which impacts negatively on the created CDS. In general, IN causes raising the number of critical data samples in the CDS. The negative impact of IN on the convergent ability of training ANFIS is formulated via Theorem 1 as follows.

Theorem 1 [16]: Let's consider a given  $\overline{\text{IDS}}$  deriving from an IDS and an ANFIS uniformly approximating an unknown mapping  $f : X \rightarrow Y$  expressed by the IDS. The ANFIS is built via a CDS built from the  $\overline{\text{IDS}}$ . Assume that  $X$  is compact. The necessary condition for the approximation convergent to a desired error  $[E]$  is that in the CDS there is not any critical data sample.

Proof: Let's consider cluster  $\Gamma^k$  belonging to the CDS. Assume that  $(\bar{x}_p, y_p) \in \Gamma^k$  is a critical data sample (see Def. 5); it has to be proven that the ANFIS will not converge to  $[E]$ .

It can infer from Eq. (3) that

$$\text{RMSE} \geq P^{-0.5} \left( \left( \hat{y}_p(\bar{x}_p) - f(\bar{x}_p) \right)^2 + \sum_{i=1}^{Q_{kp}} \left( \hat{y}_i(\bar{x}_i) - f(\bar{x}_i) \right)^2 \right)^{0.5} \quad (26)$$

Because the ANFIS is a uniform approximation of  $f : X \rightarrow Y$  and  $X$  is the compact set, it can infer that the ANFIS is continuous in  $\Gamma^k \setminus p$ , so Eq. (27) can be inferred from Eq. (26):

$$\text{RMSE} \geq P^{-0.5} \left( \left( \hat{y}_p(\bar{x}_p) - f(\bar{x}_p) \right)^2 + Q_{kp} [\bar{E}]^2 \right)^{0.5} \quad (27)$$

It should be noted that the ANFIS is a uniform approximation of the  $f : X \rightarrow Y$  in  $\Gamma^k \setminus p$ ,  $(\bar{x}_p, y_p) \in \Gamma^k$  is a critical data sample, and samples in  $\Gamma^{k(A)}$  are distributed closely. As a result, Eq. (28) can be inferred:

$$\hat{y}_p(\bar{x}_p) \in \left[ \min_{y_i \in (\Gamma^{k(B)} \setminus p)} (y_i) \quad \max_{y_i \in (\Gamma^{k(B)} \setminus p)} (y_i) \right] \quad (28)$$

Due to  $y_p > \max_{y_i \in (\Gamma^{k(B)} \setminus p)} (y_i)$  (see Def. 5), the following can be obtained from Eqs. (27) to (28):

$$RMSE \geq P^{-0.5} \left( \left( \max_{y_i \in (\Gamma^{k(B)} \setminus p)} (y_i) - y_p \right)^2 + Q_{kp} [\bar{\epsilon}]^2 \right)^{0.5} \quad (29)$$

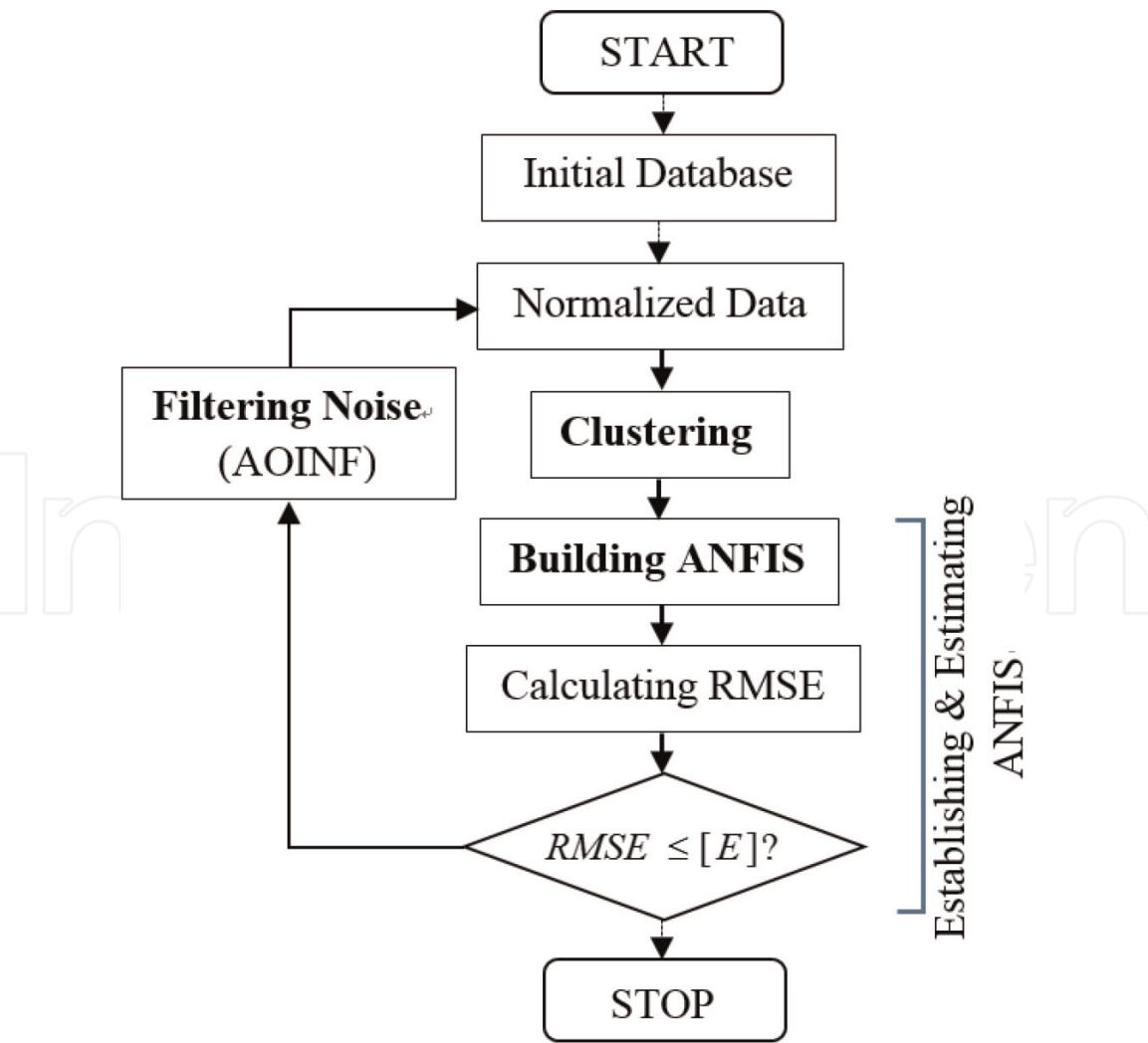
From Eqs. (8) and (29), it can conclude that  $RMSE > [E]$ . Similarly, due to  $y_p < \min_{y_i \in (\Gamma^{k(B)} \setminus p)} (y_i)$  (see Def. 5), from Eqs. (27) to (28), the following can be also inferred:

$$RMSE \geq P^{-0.5} \left( \left( \min_{y_i \in (\Gamma^{k(B)} \setminus p)} (y_i) - y_p \right)^2 + Q_{kp} [\bar{\epsilon}]^2 \right)^{0.5} \quad (30)$$

From Eq. (9) to (30),  $RMSE > [E]$  can be implied.  
Finally, it can conclude that if existing at least a critical data sample in the CDS, the ANFIS could not converge to the required error  $[E]$ .  $\square$ .

### 3.2 Algorithm for filtering IN

An essential advantage of the clustering algorithms presented in [30–31] is the convergent rate. However, the quality of the ANFIS based on the CDS deriving from them is sensitive to the IDS attributes. It can be observed that the main reason



**Figure 3.**  
Flowchart of the FIN-ANFIS consisting of the three main phases, the clustering, establishing and estimating ANFIS, and filtering IN, which are performed simultaneously.

of this status via Theorem 1 is the appearance of critical data samples. Besides, regarding the preprocessing IDS shown in [9], in spite of the positive filtering effectiveness, the calculating cost of the method is quite high. A becoming solution for the above issues can be referred in [16] where the recurrent mechanism illustrated in **Figure 3** was employed. The recurrent mechanism has two phases being performed synchronously: filtering IN in the database and building ANFIS based on the filtered database.

Firstly, an adaptive online impulse noise filter (AOINF) is proposed. The recurrent mechanism is then depicted via the algorithm named FIN-ANFIS consisting of three main phases: filtering IN, clustering data, and building ANFIS. By this way, the filtered  $\overline{\text{IDS}}$  is used to build the ANFIS, then the created ANFIS is applied as an updated filter to refilter the  $\overline{\text{IDS}}$ , and so on, until either the process converges or a stop condition is satisfied. To get a guarantee of convergence and stability, an update law for the AOINF is discovered via Lyapunov stability theory.

**Remark 1.** ANFIS cannot converge to the required error  $[E]$  if there is at least one critical data sample in the CDS (see Theorem 1). The clustering strategy of the FIN-ANFIS therefore focuses on preventing the clustering process from appearing critical data samples, along with seeking to exterminate the critical data samples in the CDS having been taking form. As a result, in each loop of the ANFIS training process, the strategy well directs the clustering process to a new CDS where either there is not any critical data sample or there exist with a smaller amount. Theorem 2 shows the convergence condition of the training process.

**Theorem 2 [16]:** Following the flowchart in **Figure 3**, the ANFIS-based approximation of an unknown mapping  $f : X \rightarrow Y$  expressed by the given IDS is built via a CDS which drives from the  $\overline{\text{IDS}}$  (the normalized IDS). Let  $Q$  be the number of critical data points in the CDS at the  $r$ th loop. At these critical data samples, if the data output is filtered by law (31), then the RMSE (3) of the ANFIS will converge to  $[E]$ :

$$^{(r+1)}y_i = ^{(r)}y_i - \rho \operatorname{sgn} \left( ^{(r)}(y_i - \hat{y}_i) \right), \quad i = 1 \dots Q. \quad (31)$$

In the above,  $\rho > 0$  is the update coefficient to be optimized by any well-known optimal method;  $^{(r)}(y_i - \hat{y}_i)$  is the error between the  $i$ th data output and the corresponding ANFIS-based output; and function  $\operatorname{sgn}(\cdot)$  is defined as

$$\operatorname{sgn}(z) = \begin{cases} 1 & \text{if } z > 0 \\ -1 & \text{otherwise.} \end{cases} \quad (32)$$

**Proof:** A Lyapunov candidate function is chosen as in Eq. (33), to which expression (34) can be inferred:

$$e(\mathbf{X}) = \mathbf{X}^T \mathbf{X}. \quad (33)$$

$$\dot{e}(\mathbf{X}) = 2 \sum_{i=1}^{P-Q} X_i \dot{X}_i + 2 \sum_{j=1}^Q X_j \dot{X}_j. \quad (34)$$

In the above,  $\dot{\Xi} = d\Xi/dt$  expresses derivative of  $\Xi$  with respect to time;  $\mathbf{X}$  is the vector of state variables deriving from  $\overline{\text{IDS}}$  as follows:

$$X_i = y_i - \hat{y}_i; \mathbf{X} = [X_1, \dots, X_P]^T \quad (35)$$

From update law (31), Eq. (34) can be rewritten as in Eq. (36):

$$\begin{aligned}\dot{e}(\mathbf{X}) &= 2 \sum_{i=1}^{P-Q} X_i \dot{X}_i + 2 \sum_{j=1}^Q X_j \dot{y}_j \\ &= 2 \sum_{i=1}^{P-Q} X_i \dot{X}_i - 2\rho \sum_{j=1}^Q X_j \operatorname{sgn}(X_j).\end{aligned}\quad (36)$$

It should be noted that the update process is performed with respect to the critical data points; hence, Eq. (36) can be rewritten as follows:

$$\dot{e}(\mathbf{X}) = -2\rho \sum_{j=1}^Q X_j \operatorname{sgn}(X_j) = -2\rho \sum_{j=1}^Q |X_j| < 0 \quad (37)$$

In addition, the following can be implied from (33) to (35):

$$e(0) = 0; \quad e(\mathbf{X}) \geq 0 \quad \forall \mathbf{X}. \quad (38)$$

Finally, it can infer from Eqs. (37) to (38) that  $e(\mathbf{X}) \rightarrow 0$  is a stable Lyapunov process. Hence, from Eq. (3) one can infer the aspect needing to be proven:

$$\text{RMSE} = \lim_{r \rightarrow \infty} \sqrt{{}^{(r)}e(\mathbf{X})P^{-1}} \leq [E]. \quad \square \quad (39)$$

Remark 2. (1) To enhance the ability to adapt to the noise status of the  $\overline{\text{IDS}}$ ,  $\rho$  in Eq. (31) is specified as follows:

$$\rho = \alpha \left| {}^{(r)}(y_i - \hat{y}_i) \right|, \quad (40)$$

where  $\alpha \geq 0$  is an adaptive coefficient chosen by the designer. Thus,  $\rho = \rho(X_i, t)$  takes part in adjusting the filtering level  $\Delta_i = |{}^{(r+1)}y_i - {}^{(r)}y_i|$ . (2) It can infer from Theorem 1 that disposing of critical data samples in the CDS needs to be carried out. Therefore, the useful solution offered in Theorem 2 via update law (31) is employed to establish the filtering mechanism of the AOINF as shown below.

The algorithm AOINF for filtering IN:

1. Look for critical data samples in the CDS to specify the worst data point (WP) where the continuous status of the ANFIS is worst:

$$\text{WP} \equiv \left( \bar{x}_i^{(WP)}, y_i^{(WP)} \right) \quad \text{such that} \quad \left| y_i^{(WP)} - \hat{y}_i^{(WP)} \right| = \max_{h=1 \dots P} |y_h - \hat{y}_h|. \quad (41)$$

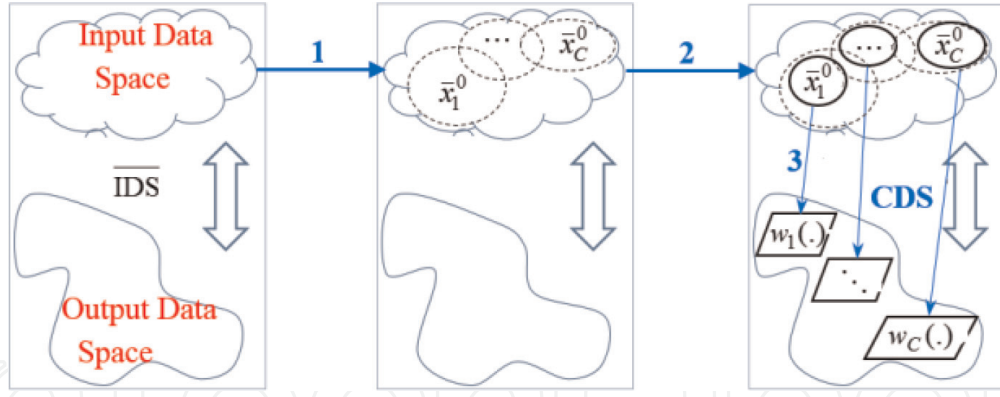
1. Specify the data samples satisfying condition (42):

$$\left| y_q - \hat{y}_q \right| \geq \frac{1}{\sigma} \left| y_i^{(WP)} - \hat{y}_i^{(WP)} \right|, \quad q = 1 \dots \overline{Q}. \quad (42)$$

In the above,  $\hat{y}_i^{(WP)}$  is the ANFIS-based output, while  $y_i^{(WP)}$  is the corresponding data output at the WP;  $\sigma > 1$  is an adaptive coefficient (to be 1.35 for the surveys shown in [16]).

1. Based on the updating law (43) to filter the data samples satisfying condition (42)

$${}^{(r+1)}y_q \leftarrow {}^{(r)}y_q + \alpha \left| {}^{(r)}(y_q - \hat{y}_q) \right| \operatorname{sgn} \left( {}^{(r)}(y_q - \hat{y}_q) \right), \quad q = 1 \dots \overline{Q}. \quad (43)$$



**Figure 4.**

A process of establishing the CDS driving from the  $\overline{IDS}$  consists.

### 3.3 Algorithm for building ANFIS

**Figure 4** illustrates the establishment of the CDS from the  $\overline{IDS}$ . It consists of (1) building fuzzy clusters with centroids  $(\bar{x}_1^0, \dots, \bar{x}_C^0)$  or the input data clusters (see Subsection 2.2), (2) estimating the hard distribution of samples in each input data cluster indicated by  $(\bar{x}_1^0, \dots, \bar{x}_C^0)$ , and (3) building the hyperplanes or the output data clusters (see Subsection 2.3) in the output data space using the specified hard distribution status. Based on the created CDS, **Figure 3** shows the flowchart of the FIN-ANFIS consisting of three main phases: filtering IN, building the CDS driving from the filtered  $\overline{IDS}$ , and forming ANFIS.

### 3.4 Algorithm FIN-ANFIS

Initializing: The initial index of the loop process,  $r = 1$ ; the number of clusters  $C \ll P - 1$ ;  $J_{KFCM}^{(r)} = \Omega$ , where  $\Omega$  is a real number  $\Omega > [ts]$ ; and the initial cluster centroids corresponding to  $r = 1$  chosen randomly:

$$\bar{x}_i^0(r) = (x_{i1}^0, \dots, x_{in}^0), \quad 1 \leq i \leq C \quad (44)$$

Build the input data clusters:

1. Establish the input data clusters:

Based on the  $\bar{x}_i^0(r)$  to be known, calculate  $\mu_{ij}$  via Eq. (14) to update  $\bar{x}_i^0(r)$  via Eq. (13).

2. Specify the stop condition of the clustering phase via  $ts$  in Eq. (15):

If  $ts \leq [ts]$ : go to Step 3; ff  $ts > [ts]$  and  $r < [r]$ , setup  $r =: r + 1$  and 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; and if  $ts > [ts]$  and  $r = [r]$  and  $C = P - 1$ , stop (not converge).

Build ANFIS:

3. Build and estimate ANFIS:

Establish ANFIS as presented in Subsection 2.4.

Calculate  $RMSE = \left( P^{-1} \sum_{i=1}^P (\hat{y}_i(\bar{x}_i) - y_i)^2 \right)^{0.5}$  in which  $\hat{y}_i(\bar{x}_i)$  is the ANFIS-based output, while  $y_i$  is the data output. If  $RMSE \leq [E]$ , stop (the ANFIS is the desired one); if  $RMSE > [E]$  and  $C < P - 1$ , go to Step 4; and if  $RMSE > [E]$  and  $C = P - 1$ , stop (not converge).

4. Set up a new cluster centroid:



Based on Eq. (41), seek the worst data point  $WP \equiv (\bar{x}_i^{(WP)}, y_i^{(WP)})$ ; set  $C = C + 1, r = 1$ , and set up a new cluster centroid  $\bar{x}_C^0$  in the neighborhood of the WP; and go to Step 5.

5. Filter IN:

Call the algorithm AOINF and return to Step 1.

## 4. ANFIS for managing online bearing fault

An application of ANFIS to estimating online bearing fault upon the ability to extract meaningful information from big data of intelligent structures is shown in this section. Estimating online bearing status to hold the initiative in exploiting the systems is meaningful because bearing is an important machine detailed in almost mechanical structures.

In [17], an Online Bearing Damage Identifying Method (ASSBDIM) based on ANFIS, singular spectrum analysis (SSA), and sparse filtering (SF) was shown. The method consists of two phases: offline and online. In offline, the ANFIS identifies the dynamic response of the mechanical system in the individual bearing statuses. The trained ANFIS is then used to estimate its real status in the online phase. These aspects are detailed in the following paragraphs.

### 4.1 Some related theories

#### 4.1.1 Singular spectrum analysis

By using SSA, from a given time series, a set of independent additive time series can be generated [41–43]. This work is clarified via the algorithm for SSA presented in [42] as follows.

##### 1. Embedding:

Let's consider a given time series of  $N_0$  data points  $(z_0, z_1, \dots, z_{N_0-1})$ . From selected window length  $L_0, 1 < L_0 < N_0$ , sliding vectors  $\mathbf{X}^j = (z_{j-1}, z_j, \dots, z_{j+L_0-2})^T, j = 1, \dots, K = N_0 - L_0 + 1$ , and matrix  $\mathbf{X}$  as in Eq. (45) are built:

$$\mathbf{X} = \begin{pmatrix} z_0 & z_1 & \cdots & \cdots & z_{N_0-L_0} \\ z_1 & z_2 & \cdots & \cdots & z_{N_0-L_0+1} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ z_{L_0-2} & z_{L_0-1} & \ddots & \ddots & z_{N_0-2} \\ z_{L_0-1} & z_{L_0} & \cdots & \cdots & z_{N_0-1} \end{pmatrix}. \quad (45)$$

##### 2. Building the trajectory matrix:

From Eq. (45), one builds matrix  $\mathbf{S} = \mathbf{X}\mathbf{X}^T \in \mathbb{R}^{L_0 \times L_0}$ . Vectors  $\mathbf{V}_i$  are then constructed,  $\mathbf{V}_i = \mathbf{X}^T \mathbf{U}_i / \sqrt{\lambda_i}, i = 1, \dots, d$ , in which  $\lambda_1, \dots, \lambda_d$  are the non-zero eigenvalues of  $\mathbf{S}$  arranged in the descending order and  $\mathbf{U}_1, \dots, \mathbf{U}_d$  are the corresponding eigenvectors. A decomposition of the trajectory matrix into a sum of matrices  $\mathbf{X} = \sum_{i=1}^d \mathbf{E}_i$  is then established, where  $\mathbf{E}_i = \sqrt{\lambda_i} \mathbf{U}_i \mathbf{V}_i^T$ .

##### 3. Reconstruction:

Each elementary matrix is transformed into a principal component of length  $N$  by applying a linear transformation known as diagonal averaging or Hankelization. Let  $\mathbf{Z} \in \mathbb{R}^{L_0 \times K}$  be a matrix of elements  $z_{i,j}$ .

By calculating  $L^* = \min(L_0, K)$ ,  $K^* = \max(L_0, K)$ , then  $\mathbf{Z}$  can be transformed into the reconstructed time series  $g_0, g_1, \dots, g_{N-1}$  as in Eq. (46):

$$g_k = \begin{cases} \frac{1}{k+1} \sum_{m=1}^{k+1} z_{m,k-m+2}, & 0 \leq k < L^* - 1 \\ \frac{1}{L^*} \sum_{m=1}^{L^*} z_{m,k-m+2}, & L^* - 1 \leq k < K^* \\ \frac{1}{N_0 - k} \sum_{m=k-K^*+2}^{N-K^*+1} z_{m,k-m+2}, & K^* \leq k < N_0 \end{cases} \quad (46)$$

#### 4.1.2 Sparse filtering

In this work SF is used to extract features from a given time series-typed measured database. Relying an objective function defined via the features, the method tries to specify the good features such that the objective function is minimized [11, 44–45]. To deploy SF effectively, a process with the two following phases is operated. Preprocessing data based on the whitening method [46] is carried out in the first phase. A  $H$ -by- $L$  matrix signed  $\mathbf{F}$  of real numbers depicting the relation between each of the  $H$  training data samples and the  $L$  selected features is established in the second phase. SF presented in [11, 45] is detailed as follows.

In the first phase, a training set of the  $H$  data samples  $\mathbf{x}_i \in \mathbb{R}^{1 \times N}$ ,  $i = 1 \dots H$ , in the form of a matrix signed  $\check{\mathbf{S}} \in \mathbb{R}^{H \times N}$  is established from the given time series-typed measuring dataset. By adopting the whitening method [46], it then tries to make the data samples less correlated with each other and speed up the convergence of the sparse filtering process which employs the eigenvalue decomposition of the covariance matrix  $\text{cov}(\check{\mathbf{S}}) = \frac{\check{\mathbf{Z}} \check{\mathbf{Z}}^T}{\check{\mathbf{D}}}$ . In the expression,  $\check{\mathbf{D}}$  is the diagonal matrix of its eigenvalues, and  $\check{\mathbf{Z}}$  is the orthogonal matrix of eigenvectors of  $\text{cov}(\check{\mathbf{S}})$ . Finally, the whitened training set signed  $\mathbf{S}_{white}$  is formed as in Eq. (47):

$$\mathbf{S}_{white} = \frac{\check{\mathbf{Z}}}{\check{\mathbf{D}} - 1/2} \check{\mathbf{Z}}^T \check{\mathbf{S}}. \quad (47)$$

Subsequently, in the second phase, SF maps the data sample  $\mathbf{x}_i \in \mathbb{R}^{1 \times N}$  of  $\mathbf{S}_{white}$  onto  $L$  features  $f_i$ ,  $i = 1 \dots L$ , relied on a weight matrix signed  $\overline{\mathbf{W}} \in \mathbb{R}^{N \times L}$ . A linear relation between data samples in  $\mathbf{S}_{white}$  and the  $L$  features is expressed via  $\overline{\mathbf{W}}$  as in Eq. (48), in which  $\mathbf{F} \in \mathbb{R}^{H \times L}$  is called the feature distribution matrix:

$$\mathbf{F} = \mathbf{S}_{white} \overline{\mathbf{W}}. \quad (48)$$

Optimizing the feature distribution in  $\mathbf{F}$  is then performed as detailed in [45]. The features in each column of  $\mathbf{F}$  is normalized by dividing them by their  $l_2$ -norm,  $\tilde{f}_l = \mathbf{f}^l / \|\mathbf{f}^l\|_2$ ,  $l = 1 \dots L$ . For each row of the obtained matrix, these features per example are normalize by computing  $\hat{\mathbf{f}}_i = \tilde{\mathbf{f}}_i / \|\tilde{\mathbf{f}}_i\|_2$ ,  $i = 1 \dots H$ , by which they lie on the unit  $l_2$ -ball. The features normalized after the two above steps are optimized for sparseness using the  $l_1$ -penalty to get a matrix signed  $\hat{\mathbf{F}} \in \mathbb{R}^{H \times L}$ . A loop process is then maintained via Eq. (48), in which  $\hat{\mathbf{F}}$  takes the role of  $\mathbf{F}$ , until the optimal weights of  $\overline{\mathbf{W}}$  are to be established that make the objective function  $J_{SF}(\overline{\mathbf{W}})$  of Eq. (49) be minimized, to which, finally,  $\hat{\mathbf{F}}$  is resigned  $\mathbf{F}$ :

$$J_{SF}(\overline{\mathbf{W}}) = \sum_{i=1}^H \sum_{j=1}^L \hat{\mathbf{F}}(i,j). \quad (49)$$

## 4.2 The ASSBDIM

The ASSBDIM focuses on online bearing fault estimation. The aim is detailed in this subsection consisting of the way of setting up the databases and the algorithm ASSBDIM for online bearing fault estimation upon the built databases.

### 4.2.1 Building the databases for the ASSBDIM

A measuring dataset deriving from the mechanical system vibration is established for each surveyed bearing fault type. Regarding  $Q$  fault types, one obtains  $Q$  original datasets as in Eq. (50):

$$[\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_Q]^T, \quad (50)$$

where  $\mathbf{D}_i$  is corresponding to the  $i$ th bearing fault type ( $1 \leq i \leq Q$ ).

By using SSA for  $\mathbf{D}_i$ ,  $m$  time series as in Eq. (51) are set up:

$$[\mathbf{D}_{i1}, \mathbf{D}_{i2}, \dots, \mathbf{D}_{im}], i = 1 \dots Q \quad (51)$$

where  $m$  is parameter selected by the designer. This work is carried out by the three steps as presented in Subsection 4.1.1, in which  $\mathbf{D}_i$  is used in the first step as the given time series of  $N_0$  data points ( $z_0, z_1, \dots, z_{N_0-1}$ ) for building the trajectory matrix  $\mathbf{X}$  in Eq. (45). Because the mechanical vibration signal is prone to the low frequency range [42], among the  $m$  time series, the  $(m-k)$  ones owning the highest frequencies are considered as noise. The  $k$  remainder time series as in Eq. (52) is hence kept to build the databases:

$$[\mathbf{D}_{i1}, \mathbf{D}_{i2} \dots \mathbf{D}_{ik}], i = 1 \dots Q \quad (52)$$

Specifying the optimal value of both  $k$  and  $m$  will be mentioned in Subsection 4.2.2.

For each time series in Eq. (52), for example,  $\mathbf{D}_{ij}, j = 1 \dots k$ , based on SF one obtains the feature distribution matrix as in Eq. (48) which is resigned  $\mathbf{F}_{ij}(\omega) \in \mathcal{R}^{H \times L}$ . By using this result for all the time series in Eq. (52), a new data matrix  $\overline{\mathbf{D}}_i$  as in Eq. (53) is formed which is the input data space of the  $i$ th data subset corresponding to the  $i$ th bearing fault type:

$$\overline{\mathbf{D}}_i = [\mathbf{F}_{i1}(\omega) \mathbf{F}_{i2}(\omega) \dots \mathbf{F}_{ik}(\omega)] \in \mathcal{R}^{H \times (kL)}. \quad (53)$$

By employing this way for  $Q$ , the surveyed bearing fault types, an input data space in the form of matrix (54), are established, which relates to building two offline databases signed Off\_DaB and Off\_testDaB as well as one online database signed On\_DaB used for the algorithm ASSBDIM as follows:

$$\overline{\mathbf{D}} = \begin{pmatrix} \omega_{11} & \omega_{12} & \cdots & \cdots & \omega_{1(kL)} \\ \omega_{21} & \omega_{22} & \cdots & \cdots & \omega_{2(kL)} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \omega_{(QH-1)1} & \omega_{(QH-1)2} & \ddots & \ddots & \omega_{(QH-1)(kL)} \\ \omega_{(QH)1} & \omega_{(QH)2} & \cdots & \cdots & \omega_{(QH)(kL)} \end{pmatrix} \in \mathcal{R}^{(QH) \times (kL)} \quad (54)$$

Namely, matrix  $\bar{\mathbf{D}}$  relates to the input data space (IDS), to which the databases for identifying the bearing status are built as follows. Firstly, by encoding the  $i$ th fault type by a real number  $y_i$ , the output data space (ODS) of the  $i$ th subset can be depicted by vector  $\bar{\mathbf{y}}_i$  of  $H$  elements  $y_i$  as in Eq. (55):

$$\bar{\mathbf{y}}_i = [y_i, \dots, y_i]^T \in \mathfrak{R}^{H \times 1}, i = 1 \dots Q \quad (55)$$

Then, by combining Eq. (55) with Eq. (54), the input-output relation in the three datasets Off\_DaB, Off\_testDaB, and On\_DaB can be described as in Eq. (56):

$$\text{database} \equiv [\text{IDS} - \text{ODS}] \equiv [\bar{\mathbf{D}} - \bar{\mathbf{y}}] \quad (56)$$

In the above, the input space  $\bar{\mathbf{D}}$  comes from Eq. (54), while the output space  $\bar{\mathbf{y}}$  as in Eq. (57) is constituted of  $\bar{\mathbf{y}}_i \in \mathfrak{R}^{H \times 1}$  in Eq. (55):

$$\bar{\mathbf{y}} = [y_1, \dots, y_{1H}, \dots, y_Q, \dots, y_{QH}]^T \in \mathfrak{R}^{QH \times 1} \quad (57)$$

#### 4.2.2 The algorithm ASSBDIM for estimating health of bearings

In the offline phase, by initializing the parameters in vector  $\mathbf{ps}$  in Eq. (58), together with applying SSA and SF to the measuring data stream, the Off\_DaB and Off\_testDaB are built as in Eq. (56):

$$\mathbf{ps} = [L_0, N_0, m, k, H, L] \quad (58)$$

where  $L_0, N_0$  come from Eq. (45);  $m$  and  $k$  relate to Eqs. (51) and (52), respectively;  $H$  and  $L$  derive from Eq. (53).

An ANFIS built by the algorithm FIN-ANFIS (see Subsection 3.3) is utilized to identify dynamic response of the mechanical system corresponding to the bearing damage statuses reflected by the Off\_DaB. Optimizing the parameters in  $\mathbf{ps}$  in Eq. (58) is then performed using the percentage of correctly estimated samples ( $Ac$ ) as in Eq. (59) and the mean accuracy ( $MeA$ ) as in Eq. (60) and the algorithm DE [47]:

$$Ac = 100 \times cr\_samples_n / to\_samples_n (\%), \quad (59)$$

$$MeA = 100 \times \sum_{n=1}^Q cr\_samples_n / \sum_{n=1}^Q to\_samples_n (\%), \quad (60)$$

where corresponding to the  $n$ th damage type,  $n = 1 \dots Q$ ,  $cr\_samples_n$  is the number of checking samples expressing correctly the real status of the bearing, while  $to\_samples_n$  is the total of checking samples used in the survey;  $Q$  is the number of surveyed bearing fault types as mentioned in Eq. (50).

Following the  $MeA$ , an objective function is defined as follows:

$$J = MeA_{\text{ASSBDIM}}(L_0, N_0, m, k, H, L) \rightarrow \max. \quad (61)$$

The Off\_testDaB, function  $J$ , and DE [47] are then employed to optimize the parameters in vector  $\mathbf{ps}$ , to get  $[L_0, N_0, m, k, H, L]_{\text{opt}}$ .

Namely, by using the input of the Off\_testDaB for the ANFIS which has been trained by the Off\_DaB, one obtains the outputs  $\hat{y}_i$ ,  $i = 1 \dots H$ . These outputs are then compared with the corresponding encoded outputs to estimate the bearing real status, which is the one encoded by “ $q$ ” satisfying Eq. (62):

$$\sum_{i=1}^H \left| \hat{y}_i - y_q \right| = \min_{h=1 \dots Q} \sum_{i=1}^H \left| \hat{y}_i - y_h \right|. \tag{62}$$

The completion of the offline phase as above can be seen as the beginning of the only phase. During the next operating process, first, by the way similar to the one for building the offline database Off\_DaB, the online dataset On\_DaB in the form  $\overline{\mathbf{D}}_{\text{ON}} \equiv \overline{\mathbf{D}}_i \in \mathfrak{R}^{H \times (kL)}$  as in Eq. (53) is built. By using the On\_DaB for the ANFIS trained in the offline, the bearing real status at this time is then specified based on Eq. (62).

The ASSBDIM is hence can be summarized as follows.

The offline process:  
Initialize vector **ps** in Eq. (58):

1. Build the Off-DaB and Off-testDaB in the form of Eq. (56).
2. Train an ANFIS to identify the Off-DaB using the algorithm FIN-ANFIS.
3. Accomplish the system.

The Off-testDaB is used as database of the trained ANFIS, using the condition (62) to calculate *MeA* in Eq. (60). If  $MeA \leq [MeA]$ , then go to Step 4; otherwise, adjust the value of the elements in vector **ps** in Eq. (58) using the algorithm DE [47], and then return to Step 1.

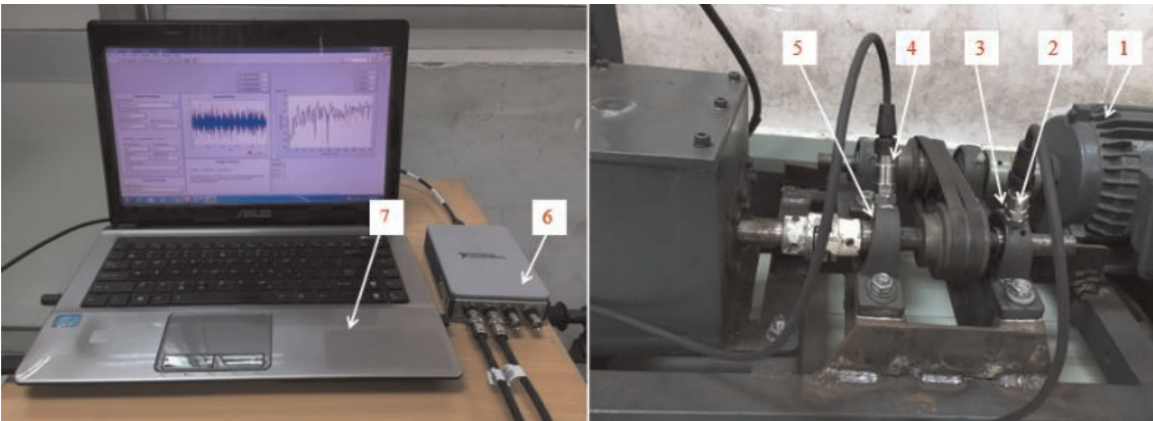
The online process:

4. Establish online database On-DaB  $\overline{\mathbf{D}}_{\text{ON}} \equiv \overline{\mathbf{D}}_i \in \mathfrak{R}^{H \times (kL)}$  as in Eq. (53).
5. Estimate online bearing fault status based on the On-DaB, trained ANFIS, and condition (62); check the stop condition: if it is not satisfied, then return to Step 4; otherwise, stop.

4.3 Some survey results

4.3.1 Experimental apparatus and estimating way

The experimental apparatus for measuring vibration signal is shown in **Figure 5**. The apparatus consists of the motor (1), acceleration sensors (2) and (4), surveyed bearings (3) and (5), module for processing and transforming series vibration signal incorporating software-selectable AC/DC coupling (Model: NI-9234) (6), and computer (7).



**Figure 5.**  
*Experimental apparatus for measuring vibration signal.*



Surveyed cases					
Case 1		Case 2		Case 3	
Status	EV ( $y_i$ )	Status	EV ( $y_i$ )	Status	EV ( $y_i$ )
L1Und	0	L2Und	0	L1Und	0
L1D1In	1	L2D1In	1	L1D1Ba	1
L1D2In	2	L2D2In	2	L1D1In	2
L1D3In	3	L2D3In	3	L1D1Ou	3
L1D1Ou	4	L2D1Ba	4	L1D2Ba	4
L1D2Ou	5	L2D2Ba	5	L1D2In	5
L1D3Ou	6	L2D3Ba	6	L1D2Ou	6
-	-	-	-	L1D3Ba	7
-	-	-	-	L1D3In	8
-	-	-	-	L1D3Ou	9

**Table 1.**  
*Surveyed cases and the corresponding encoding values (EV).*

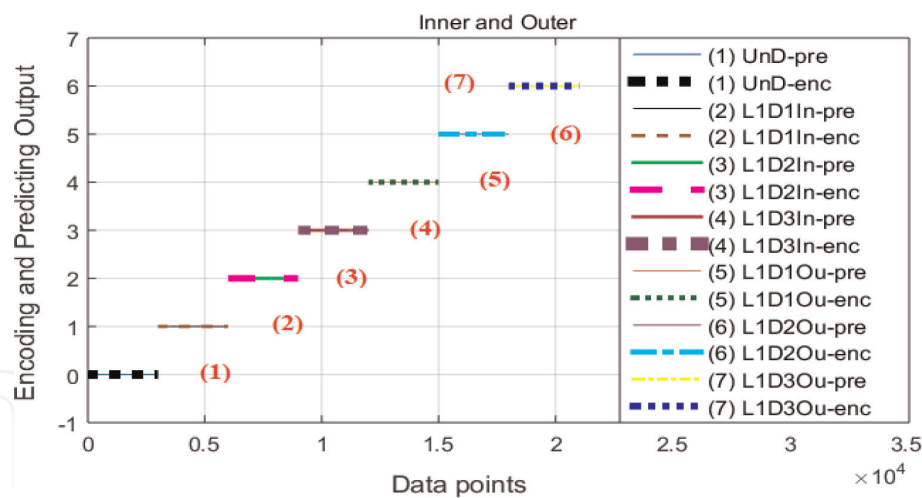
Faults	Width (mm)	Depth (mm)
D1Ou	0.20	0.3
D2Ou	0.30	0.3
D3Ou	0.46	0.3
D1In	0.20	0.3
D2In	0.30	0.3
D3In	0.40	0.3
D1Ba	0.15	0.2
D2Ba	0.20	0.2
D3Ba	0.25	0.2

**Table 2.**  
*The size of bearing single fault types used for surveys.*

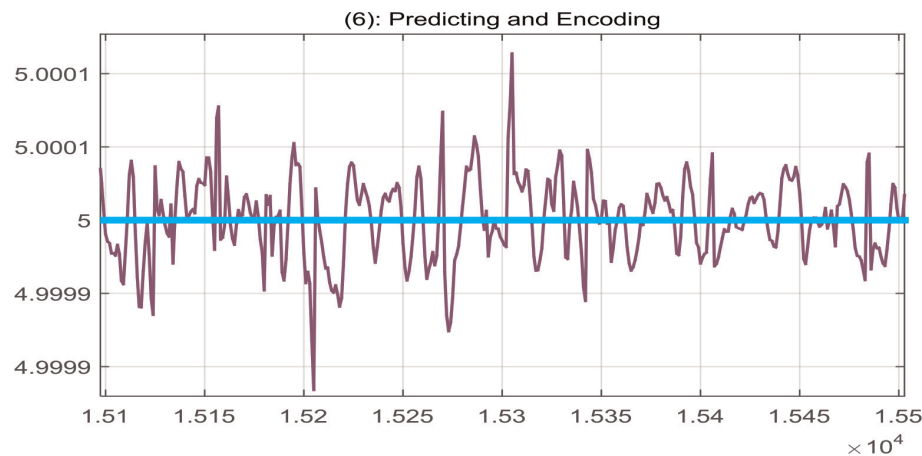
In **Table 1**, “encoding value” is abbreviated to “EV.” The three cases listed in **Table 1** related to nine of the widespread single-bearing faults as in **Table 2** are surveyed. In the above,  $Q = 7$  (see Eq. 50) for the Cases 1–2, while  $Q = 10$  for Case 3; the damaged location is the inner or outer or balls (signed In, or Ou, or Ba, respectively); damaged degrees are from 1 to 3 (signed D1 or D2 or D3); the load impacting on the system at the survey time consists of Load 1 or 2 or 3 (signed L1 or L2 or L3). For example,  $LmUnd$  shows the load degree to be  $m$  and the bearing to be undamaged, or  $LmDnBa$  expresses the load degree to be  $m$  (1,...,3), the damage level to be  $n$  (1,...,3), and the damage location to be the ball.

The ASSBDIM with  $H = 303$ ,  $m = 30$ ,  $k = 7$  along with four other methods [48–51] is employed to be surveyed. The first one [48] ( $N_{in} = N_{out} = 100$ ; number of segments to be  $20 \times 10^3$  and  $\lambda = 1E - 5$ ) is the intelligent fault diagnosis method using unsupervised feature learning toward mechanical big data. The second one [49] employs the energy levels of the various frequency bands as features. In the third one [50], a bearing fault diagnosis upon permutation entropy, empirical mode decomposition, and support vector machines is shown. In the last one [51], a method of identifying bearing fault based on SSA is presented.

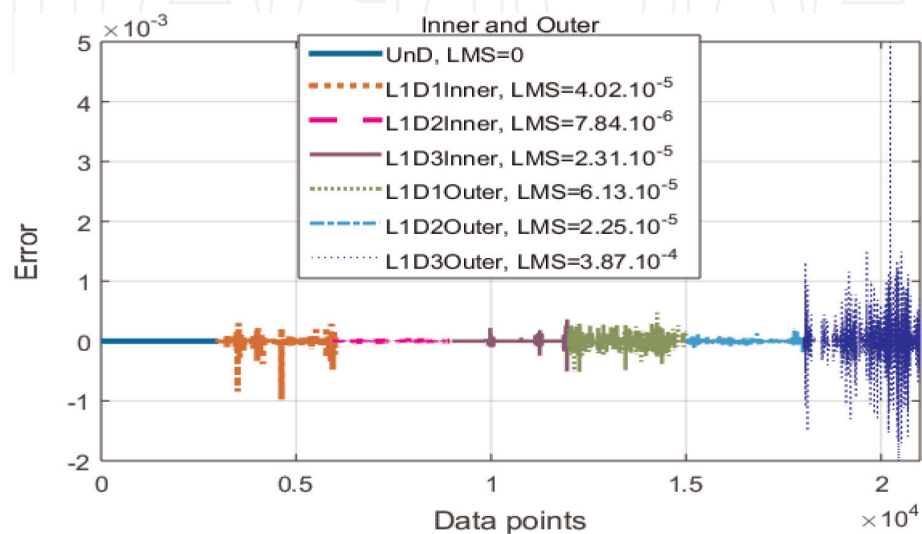
For the surveys, along with  $Ac$  and  $MeA$ , the root-mean-square error as in Eq. (63) is also employed, where  $y_i$  and  $\hat{y}_i$ , respectively, are encoding and predicting outputs:



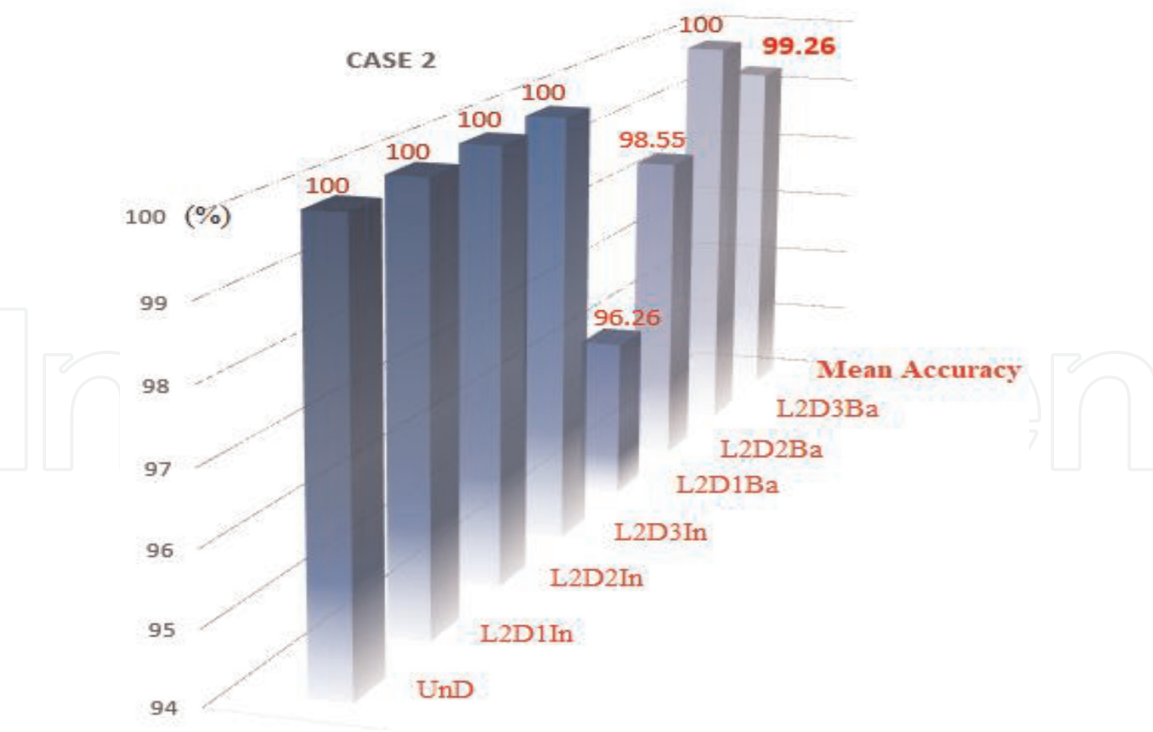
**Figure 6.**  
The predicting ( $-pre$ ) output  $\hat{y}_i$  of the ASSBDIM in Case 1 and the corresponding encoded ( $-enc$ ) output  $y_i$ .



**Figure 7.**  
The  $\hat{y}_i$  and  $y_i$  depicted by lines (6) in Figure 6 to be zoomed in.



**Figure 8.**  
The error reflecting the difference between  $y_i$  and  $\hat{y}_i$  in Figure 6.



**Figure 9.**  
*Ac and MeA (mean accuracy) of the ASSBDIM in Case 2.*

Surveyed cases	Ac (%)				
	[48]	[49]	[50]	[51]	[17]
L <sub>2</sub> UnD	99.67	93.73	98.68	99.67	100
L <sub>2</sub> D <sub>1</sub> In	98.35	95.05	92.74	95.38	99.67
L <sub>2</sub> D <sub>2</sub> In	99.67	98.68	99.34	97.36	99.01
L <sub>2</sub> D <sub>3</sub> In	99.01	93.07	95.38	92.74	100
L <sub>2</sub> D <sub>1</sub> Ba	98.68	91.09	96.37	96.70	97.36
L <sub>2</sub> D <sub>2</sub> Ba	99.67	92.08	94.72	99.01	98.68
L <sub>2</sub> D <sub>3</sub> Ba	97.36	98.68	100	98.68	100
MeA (%)	98.92	94.93	96.75	97.08	99.26

**Table 3.**  
*The accuracy of the methods in Case 2.*

$$\text{LMS} = \sqrt{\sum_{i=1}^H (y_i - \hat{y}_i)^2 / H}.$$

(63)

4.3.2 Some survey results

The measured databases from Cases 1 to 3 with  $Q = 7$  as in **Table 1** along which the methods consist of the ASSBDIM [17] and the ones from [48–51] were adopted to identify the status of the bearing. The obtained results were shown in **Figures 6–9** and **Tables 3** and **4**.

4.3.3 Discussion

Following the above results, it can observe that among the surveyed methods, the ASSBDIM which is based on ANFIS gained the best accuracy. This aspect can be

Surveyed cases	Ac (%)				
	[48]	[49]	[50]	[51]	[17]
L <sub>1</sub> UnD	95.05	87.46	85.81	100	100
L <sub>1</sub> D <sub>1</sub> In	94.72	90.10	89.77	83.17	99.67
L <sub>1</sub> D <sub>2</sub> In	92.08	92.41	88.12	85.48	99.34
L <sub>1</sub> D <sub>3</sub> In	93.40	92.74	92.41	94.72	99.34
L <sub>1</sub> D <sub>1</sub> Ou	95.33	89.77	84.16	85.15	82.51
L <sub>1</sub> D <sub>2</sub> Ou	92.41	92.41	84.82	84.16	94.39
L <sub>1</sub> D <sub>3</sub> Ou	95.05	88.78	88.78	99.34	89.11
L <sub>1</sub> D <sub>1</sub> Ba	86.47	89.44	90.43	83.17	94.72
L <sub>1</sub> D <sub>2</sub> Ba	87.79	90.10	96.04	97.36	92.74
L <sub>1</sub> D <sub>3</sub> Ba	88.12	86.14	100	88.45	84.82
MeA (%)	92.04	89.94	90.03	90.10	93.66

**Table 4.**  
*The accuracy of the methods in Case 3.*

recognized via the quite equivalent values between the encoding and predicting outputs from the tested data samples. The small difference depicted by the zooming in in **Figure 7** and the root-mean-square error in **Figure 8** as well as the high/higher values of Ac and MeA deriving from the ASSBDIM in **Tables 3 and 4** and **Figure 9** reflect clearly the ANFIS’s identification ability.

It should be noted that the methodology shown via the algorithm ASSBDIM can be also used to discover the method of managing damage of mechanical structures as well.

## 5 Conclusion

The hybrid structure ANFIS, where ANN and FL can interact to not only overcome partly the limitations of each model but also uphold their strong points, has been seen as a useful mathematical tool for many fields. Inspired by the ANFIS’s capability, in order to provide the readers with the theoretical basis and application direction of the model, this chapter presents the formulation of ANFIS and one of its typical applications.

Firstly, the structure of ANFIS as a data-driven model deriving from fuzzy logic and artificial neural networks is depicted. The setting up the input data clusters, output clusters and ANFIS as a joint structure is all detailed. Deriving from this relation, the method of building ANFIS from noisy measuring datasets is presented. The online and recurrent mechanism for filtering noise and building ANFIS synchronously is clarified via the algorithms for filtering noise and establishing ANFIS. Finally, the application of ANFIS coming from the online managing bearing fault is presented. The compared results reflect that among the surveyed methods, the ASSBDIM which exploited the identification ability of ANFIS gains the best accuracy. Besides, the methodology shown via this application can be also used as appropriate solution for developing new methods of managing damage of mechanical structures.

In addition to the above identification field, it should be noted that (1) ANFIS has also attracted the attention of many researchers in the other fields related to prediction, control, and so on, as mentioned in Section 1 and (2) ANFIS can collaborate effectively with some other mathematical tools to enhance the effectiveness of technology applications.

IntechOpen

### **Author details**

Sy Dzong Nguyen

1 Division of Computational Mechatronics, Institute for Computational Science,  
Ton Duc Thang University, Ho Chi Minh City, Vietnam

2 Faculty of Electrical and Electronics Engineering, Ton Duc Thang University,  
Ho Chi Minh City, Vietnam

\*Address all correspondence to: [nguyensydzong@tdtu.edu.vn](mailto:nguyensydzong@tdtu.edu.vn)

### **IntechOpen**

© 2019 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 



## References

- [1] Kosko B. Fuzzy systems as universal approximators. *IEEE Transactions on Computers*. 1994;**43**(11):1329-1333
- [2] Nguyen SD, Choi SB. A new neuro-fuzzy training algorithm for identifying dynamic characteristics of smart dampers. *Smart Materials and Structures*. 2012;**21**(8):1-14
- [3] Nguyen SD, Choi SB. A novel minimum-maximum data-clustering algorithm for vibration control of a semi-active vehicle suspension system. *Journal of Automobile Engineering, Part D*. 2013;**227**(9):1242-1254
- [4] Nguyen SD, Choi SB, Nguyen QH. An optimal design of interval type-2 fuzzy logic system with various experiments including magnetorheological fluid damper. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*. 2014: 1-17. DOI: 10.1177/0954406214526585
- [5] Nguyen SD, Nguyen QH, Choi SB. Hybrid clustering based fuzzy structure for vibration control – Part 1: A novel algorithm for building neuro-fuzzy system. In: *Mechanical Systems and Signal Processing*. Vol. 50-51. 2014. pp. 510-525
- [6] Nguyen SD, Choi SB. Design of a new adaptive neuro-fuzzy inference system based on a solution for clustering in a data potential field. *Fuzzy Sets and Systems*. 2015;**279**:64-86
- [7] Nguyen SD, Nguyen QH. Design of active suspension controller for train cars based on sliding mode control, uncertainty observer and neuro-fuzzy system. *Journal of Vibration and Control*. 2015:1-20. DOI: 10.1177/1077546315592767
- [8] Jang JSR. ANFIS: Adaptive-network-based fuzzy inference systems. *IEEE Transactions on Systems, Man, and Cybernetics*. 1993;**23**:665-685
- [9] Chen C, Bin Z, George V, Marcos O. Machine condition prediction based on adaptive neuro-fuzzy and high-order particle filtering. *IEEE Transactions on Industrial Electronics*. 2011;**58**(9): 4353-4364
- [10] Panella M, Gallo AS. An input-output clustering approach to the synthesis of ANFIS networks. *IEEE Transactions on Fuzzy Systems*. 2005; **13**(1):69-81
- [11] Lei Y, Jia F, Lin J, Xing S, Ding SX. An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data. *IEEE Transactions on Industrial Electronics*. 2016;**63**(5)
- [12] Nguyen SD, Nguyen QH, Seo TI. ANFIS deriving from jointed input-output data space and applying in smart-damper identification. *Applied Soft Computing*. 2017;**53**:45-60
- [13] Theocharis JB. A high-order recurrent neuro-fuzzy system with internal dynamics: Application to the adaptive noise cancellation. *Fuzzy Sets and Systems*. 2006;**157**:471-500
- [14] Besdok E, Civicioglu P, Alci M. Using an adaptive neuro-fuzzy inference system based interpolant for impulsive noise suppression from highly distorted images. *Fuzzy Sets and Systems*. 2005;**150**:525-543
- [15] Kumari R, Gambhir D, Kumar V. Intensity difference based neuro-fuzzy system for impulse noisy image restoration: ID-NFS. In: *Proceedings of International Conference on Signal Processing and Integrated Networks (SPIN)*, 978-1-4799-2866-8/14/\$31.00 ©2014 IEEE, 2014

- [16] Nguyen SD, Choi S-B, Seo T-I. Recurrent mechanism and impulse noise filter for establishing ANFIS. *IEEE Transactions on Fuzzy Systems*. 2018; **26**(2):985-997
- [17] Seo T-I, Sy DN. Algorithm for estimating online bearing fault upon the ability to extract meaningful information from big data of intelligent structures. *IEEE Transactions on Industrial Electronics*. 2018. DOI: 10.1109/TIE.2018.2847704
- [18] Nguyen SD, Choi SB, Nguyen QH. A new fuzzy-disturbance observer-enhanced sliding controller for vibration control of a train-car suspension with magneto-rheological dampers. *Mechanical Systems and Signal Processing*. 2018;**105**:447-466
- [19] Nguyen SD, Jung D, Choi SB. A robust vibration control of a magnetorheological damper based railway suspension using a novel adaptive type-2 fuzzy sliding mode controller. *Shock and Vibration*. 2017; **7306109**. DOI: 10.1155/2017/7306109
- [20] Nguyen SD, Vo HD, Seo TI. Nonlinear adaptive control based on fuzzy sliding mode technique and fuzzy-based compensator. *ISA Transactions*. 2017;**70**:309-321
- [21] Nguyen SD, Ho HV, Nguyen TT, Truong NT, Seo TI. Novel fuzzy sliding controller for MRD suspensions subjected to uncertainty and disturbance. *Engineering Applications of Artificial Intelligence*. 2017;**61**:65-76
- [22] Nguyen SD, Nguyen QH, Seo TI. ANFIS deriving from jointed input-output data space and applying in smart-damper identification. *Applied Soft Computing*. 2017;**53**:45-60
- [23] Nguyen SD, Kim WH, Park JH, Choi SB. A new fuzzy sliding mode controller for vibration control systems using integrated structure smart dampers. *Smart Materials and Structures*. 2017; **26**(2017):045038
- [24] Nguyen SD, Choi SB, Seo TI. Adaptive fuzzy sliding control enhanced by compensation for explicitly unidentified aspects. *International Journal of Control, Automation and Systems*. 2017. DOI: 10.1007/s12555-016-0569-6
- [25] Nguyen SD, Seo TI. Establishing ANFIS and the use for predicting sliding control of active railway suspension systems subjected to uncertainties and disturbances. *International Journal of Machine Learning and Cybernetics*. 2016. DOI: 10.1007/s13042-016-0614-z
- [26] Nguyen SD, Nguyen QH. Design of active suspension controller for train cars based on sliding mode control, uncertainty observer and neuro-fuzzy system. *Journal of Vibration and Control*. 2015;**23**(8):1334-1353
- [27] Turkmen I. Efficient impulse noise detection method with ANFIS for accurate image restoration. *International Journal of Electronics and Communications*. 2011;**65**:132-139
- [28] Hemalatha C, Periasamy A, Muruganand S. A hybrid approach for efficient removal of impulse, Gaussian and mixed noise from highly corrupted images using adaptive neuro fuzzy inference system (ANFIS). *International Journal of Computer Applications*. 2012; **45**(16):15-21
- [29] Saradhadevi V, Sundaram DV. An enhanced two-stage impulse noise removal technique based on fast ANFIS and fuzzy decision. *International Journal of Computer Science Issues*. 2011;**8**(1):79-88
- [30] Shen H, Yang J, Wangm S, Liu X. Attribute weighted mercer kernel based fuzzy clustering algorithm for general non-spherical datasets. *Soft Computing*. 2006;**10**:1061-1073

- [31] Marcelo RP, Ferreira AT, Francisco C. Kernel fuzzy C-means with automatic variable weighting. *Fuzzy Sets and Systems*. 2014;**237**:1-46
- [32] Filippone M, Camastra F, Masulli F, Rovetta S. A survey of kernel and spectral methods for clustering. *Pattern Recognition*. 2008;**41**:176-190
- [33] Camastra F, Verri A. A novel kernel method for clustering. *IEEE Transactions on Neural Networks*. 2005; **27**(5):801-804
- [34] Graves D, Pedrycz W. Kernel-based fuzzy clustering and fuzzy clustering: A comparative experimental study. *Fuzzy Sets and Systems*. 2010;**161**:522-543
- [35] Winkler R, Klawonn F, Kruse R. Problems of fuzzy c-means clustering and similar algorithms with high dimensional data sets. *International Journal of Fuzzy Systems*. 2011;**1**(1):1-16
- [36] Xu R, Wunusch DII. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*. 2005; **16**(3):645-678
- [37] Girolami M. Mercer kernel-based clustering in feature space. *IEEE Transactions on Neural Networks*. 2002; **13**:780-784
- [38] Thevaril J, Kwan HK. Speech enhancement using adaptive neuro-fuzzy filtering. *Proceedings of International Symposium on Intelligent Signal Processing and Communication Systems*. 2005
- [39] Balaiah P, Ilavennila. Comparative evaluation of adaptive filter and neuro-fuzzy filter in artifacts removal from electroencephalogram signal. *American Journal of Applied Sciences*. 2012;**9**(10): 1583-1593
- [40] Lakra S, Prasad TV, Ramakrishna G. Selective noise filtering of speech signals using an adaptive neuro-fuzzy inference system as a frequency pre-classifier. *Journal of Theoretical and Applied Information Technology*. 2015; **81**(3):496-501
- [41] Golyandina, Nekrutkin V, Zhigljavsky A. *Analysis of Time Series Structure—SSA and Related Techniques*, Chapman & Hall/CRC. Boca Raton, Florida; 2001
- [42] Salgado DR, Alonso FJ. Tool wear detection in turning operations using singular spectrum analysis. *Journal of Materials Processing Technology*. 2006; **171**:451-458
- [43] Kilundu B, Dehombreux P, Chimentin X. Tool wear monitoring by machine learning techniques and singular spectrum analysis. *Mechanical Systems and Signal Processing*. 2011;**25**: 400-415
- [44] Willmore B, Tolhurst DJ. Characterizing the sparseness of neural codes. *Network: Computation in Neural Systems*. 2001;**12**(3):255-270
- [45] Ngiam J, Chen Z, Bhaskar SA, Koh PW, Ng AY. Sparse filtering. In: *Proceedings of Neural Information Processing Systems*. 2011. pp. 1125-1133
- [46] Hyvärinen A, Oja E. Independent component analysis: Algorithms and applications. *Neural Networks*. 2000; **13**(4):411-430
- [47] Gong W, Cai Z. Differential evolution with ranking-based mutation operators. *IEEE Transactions on Cybernetics*. 2013;**43**:1-16
- [48] Lei Y, Jia F, Lin J, Xing S, Ding SX. An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data. *IEEE Transactions on Industrial Electronics*. 2016;**63**(5)
- [49] Ao HL, Cheng J, Li K, Truong TK, A Roller Bearing Fault Diagnosis Method

Based on LCD Energy Entropy and  
ACROA-SVM, Shock and Vibration;  
2014. Vol. 2014. Article ID: 825825. 12 p

[50] Zhang X, Liang Y, Zhou J, Zang Y. A  
novel bearing fault diagnosis model  
integrated permutation entropy,  
ensemble empirical mode  
decomposition and optimized SVM.  
Measurement. 2015;**69**:164-179

[51] Liu T, Chen J, Dong G. Singular  
spectrum analysis and continuous  
hidden Markov model for rolling  
element bearing fault diagnosis. Journal  
of Vibration and Control. 2015;**21**(8):  
1506-1521