

A New Evolving Technology for Gearbox Condition Monitoring and Fault Diagnosis

Derek Y. Luo, Wilson Wang

Department of Mechanical and Mechatronics Engineering, Lakehead University, Thunder Bay, Canada

Email: wilson.wang@Lakeheadu.ca

How to cite this paper: Luo, D.Y. and Wang, W. (2025) A New Evolving Technology for Gearbox Condition Monitoring and Fault Diagnosis. *Intelligent Control and Automation*, 16, 158-174.

<https://doi.org/10.4236/ica.2025.164007>

Received: September 7, 2025

Accepted: November 2, 2025

Published: November 5, 2025

Copyright © 2025 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Gearboxes are commonly used in rotary machines. Reliable fault diagnostics in gearboxes is of great importance to industries to improve production quality and reduce maintenance costs. In this paper, an improved evolving fuzzy (iEF) technique is proposed for real-time gear system health monitoring and fault diagnosis. The architecture evolution is performed based on the comparison of the potential of the incoming data set and the existing cluster centers. The proposed evolving method has the ability of adding or subtracting clusters adaptively. An enhanced Kalman filter (EKF) method is suggested to improve parameter training efficiency and processing convergence. The effectiveness of the developed classifier is evaluated firstly by simulation tests and then by experimental tests under different gear conditions.

Keywords

Evolving Systems, Pattern Classification, Gear System Condition Monitoring, Fault Diagnosis, Machine Learning

1. Introduction

Gearboxes are commonly used in rotating machinery such as electric vehicles, manufacturing facilities, and wind turbines [1]. A gearbox is a system that consists of a series of gears, shafts and support bearings. Gear failure in a machine can lead to production quality degradation, malfunction, or even catastrophic failures. Reliable gear monitoring techniques and tools are critically needed in a wide range of industries [2]. On the other hand, diagnostic information can also be used to quickly recognize the damaged components in repairs without inspecting all of the involved components in a gearbox, which can further reduce maintenance costs [3].

The common defects in a gearbox include pitting, severe wear, tooth crack,

scoring, etc. Gear fault analysis can be undertaken by analyzing different types of information carriers, such as vibration, noise, or lubricant [4]. Vibration-based analysis is the most used approach in gearbox health monitoring because of its ease of measurement and high signal-to-noise ratio, which will also be used in this work [5].

Fault diagnosis is a process involving two procedures: feature extraction and diagnostic pattern classification. Feature extraction is a process to extract representative features from the collected vibration signal by using appropriate signal processing techniques. Diagnostic classification is a process to classify the obtained representative features into different gear health categories [6]. As gear signal is periodic in nature, the time synchronous average (TSA) can be used to extract signatures specific to a gear of interest [7]. There are many gear fault detection techniques available in literature. From systematic investigation by the authors' research team, it is found that the most effective fault detection techniques include phase demodulation, beta kurtosis and wavelet transform amplitude [8]. This work will use features obtained by using these three techniques to do fault diagnosis. Details of these techniques can be found in [6] [8]-[10].

The diagnostic system will integrate these representative features for automatic fault detection. Artificial intelligence tools, such as fuzzy logic, neural networks, and synergetic paradigms, have been widely used in automatic gear fault detection and diagnosis [11]. The authors' research team has also developed several intelligent tools for machinery fault diagnostics and prognosis [12]-[16]; in these diagnostic classifiers, fixed reasoning structures are used in fuzzy reasoning, while system parameters are updated online or offline. But these classification techniques with fixed reasoning structures may not be suitable for monitoring applications of gearboxes with time-varying dynamics and operating conditions.

An alternative solution to this problem is the use of some clustering algorithms to generate classification reasoning architecture. Continuous and gradual adaptation will make the classification operation smooth and regular over the intervals of input parameters. As the fuzzy system is a universal approximator and can represent human knowledge in reasoning properly, it is generally used as the platform in designing evolving systems. An evolving Takagi-Sugeno (eTS) scheme is proposed in [17] for system control; its formulation of the clusters is determined by a potential measurement, while least square estimator (LSE) algorithm is used to update linear parameters. A problem with this clustering method is that the predefined cluster information (e.g., centers and spreads) is usually sensitive to noise in the data sets and processing errors. A parsimonious ensemble evolving classifier is proposed in [18] to make dynamic selection of input features, but its selected subset differs at each iteration. A transductive neuro-fuzzy inference (TWNFI) system is suggested in [19] by introducing weighted data normalization for transductive reasoning. Compared with the eTS in modelling of non-linear systems, the TWNFI usually generates more clusters/rules and thus may result in lower processing efficiency [20].

One of the problems in the aforementioned evolving classifiers is related to their blind classification reasoning, especially in the output space. In order to tackle this problem, the objective of this work is to propose an improved evolving fuzzy (iEF) technique for gear system condition monitoring and fault diagnosis. The proposed iEF technique is new in the following aspects: 1) a new evolving algorithm is proposed for better output space partition to eliminate contradictory clusters/rules generated due to noise-affected data sets. 2) A new training algorithm based on an enhanced Kalman filter (EKF) is suggested to train iEF system parameters classifier. The iEF classifier is also implemented for real-time gear health monitoring. Its effectiveness is verified by simulation and experimental tests.

The remainder of this paper is organized as follows: The proposed iEF technique and EKF training algorithm are discussed in Section 2. In Section 3, the effectiveness of the new classifier is verified by simulation test, and then it is implemented for gear system monitoring.

2. The Developed Evolving Fuzzy Technology

The proposed iEF technique and EKF training method will be discussed in this section.

2.1. iEF Fuzzy Reasoning

Clustering is a process to group data into different data sets, so as to reveal patterns in the data and to provide a concise representation of the data behavior. The iEF reasoning framework is based on the Takagi-Sugeno (TS) method with the following form:

$$\mathfrak{R}_j : \text{If } (x_1 \text{ is } A_{1,j}) \text{ and } \dots \text{ and } (x_n \text{ is } A_{n,j}) \text{ then } y = y_j \text{ (with weight } w_j) \quad (1)$$

where $\mathfrak{R}_j$ denotes the j th fuzzy cluster/rule, $j \in [1, R]$, and R is the total number of fuzzy clusters/rules; $A_{i,j}$ is the j th fuzzy set for x_i , $i \in [1, n]$;

$y_j = [y_{j1}, y_{j2}, \dots, y_{jM}]$ are the output fuzzy sets, in this case, related to healthy, possibly damaged, and damaged categories. w_j is the weight factor representing the contribution of rule $\mathfrak{R}_j$ to the pattern classification.

In the proposed iEF technique, all the fuzzy set membership functions (MFs) are in Gaussian form

$$\mu_{A_{i,j}} = \exp\left(-\frac{(x_i - m_{i,j})^2}{2\sigma_{i,j}^2}\right) \quad (2)$$

where $m_{i,j}$ and $\sigma_{i,j}$ are the centers and spreads of the MF, respectively. A Gaussian function not only has properties of continuity and generalization, but also can be decomposed into multiple one-dimensional Gaussian MFs corresponding to different input variables. These properties can facilitate the implementation of input/output partition if each cluster is treated as a fuzzy cluster (rule) [14].

If a max-product operator is used for the premise fuzzy reasoning, the rule firing strength will be

$$\mu_j = \prod_{i=1}^n \mu_{A_{i,j}}(x_i) = \prod_{i=1}^n \exp\left(-\frac{(x_i - m_{i,j})^2}{2\sigma_{i,j}^2}\right) = \exp\left(-\sum_{i=1}^n \frac{(x_i - m_{i,j})^2}{2\sigma_{i,j}^2}\right), j \in [1, R] \quad (3)$$

After normalization of the rule firing strengths, the overall output will be

$$y = \sum_{j=1}^R \frac{\mu_j}{\mu_\Sigma} q_j w_j, \quad j \in [1, R] \quad (4)$$

where q_j is the result from the consequent part, and the firing strength of the j th rule is normalized by

$$\mu_\Sigma = \sum_{j=1}^R \mu_j = \sum_{j=1}^R \exp\left(-\sum_{i=1}^n \frac{(x_i - m_{i,j})^2}{2\sigma_{i,j}^2}\right) \quad (5)$$

2.2. The iEF Approach

The iEF is a data-driven, non-iterative, and one-pass method. Different from the general potential-based methods, iEF partitions input and output spaces simultaneously to keep input/output mapping consistency and remove the noise-affected outliers. It recursively updates the cluster centers and spreads, so as to make the generated clusters well-distributed over the input-output spaces. Different from other evolving algorithms, the partitioning of the output space of the proposed iEF is performed according to the machine health conditions. The processing procedures are discussed below.

Step 1: Initialize the parameters: The initial iEF classifier has an empty rule base. Input the first data sample $\mathbf{z}_k = [\mathbf{x}_k, \mathbf{y}_k]$, $k := 1$, which defines the first cluster center: $\mathbf{c}_k := \mathbf{z}_k$. Then, $R := 1$, $N_r := 1$, $\mathbf{m}_{R,I} := \mathbf{x}_k$, $\sigma_{R,I} := \mathbf{0.10}$, $\mathbf{m}_{R,O} := \mathbf{y}_k$, $\sigma_{R,O} := \mathbf{0.10}$, $P_k(\mathbf{z}_k) := 1$, $P_k(\mathbf{c}_k) := 1$, where N_r is the number of samples in cluster r , $r \in [1, R]$ and R is the number of clusters/rules; $\mathbf{m}_{R,I}$, $\mathbf{m}_{R,O}$, $\sigma_{R,I}$ and $\sigma_{R,O}$ are the cluster centers and spreads in the input and output spaces, respectively. $P_k(\mathbf{z}_k)$ is the potential of data sample $\mathbf{z}_k$, and $P_k(\mathbf{c}_k)$ is the potential of the center $\mathbf{c}_k$.

Step 2: Compute the potential: Input the next data sample, $\mathbf{z}_k = [\mathbf{x}_k, \mathbf{y}_k]$; $k := k + 1$. The potential of $\mathbf{z}_k$ is calculated by

$$P_k(\mathbf{z}_k) = \frac{k-1}{(k-1)(\theta_k + 1) + \sigma_k - 2\mathbf{v}_k} \quad (6)$$

where $\theta_k = \sum_{i=1}^n (z_{k,i})^2$; $\sigma_k = \sigma_{k-1} + \sum_{i=1}^n (z_{k-1,i})^2$; $\mathbf{v}_k = \sum_{i=1}^n z_{k,i} \beta_{k,i}$; $\beta_{k,i} = \beta_{k-1,i} + z_{k-1,i}$; $\beta_{k,i}$ and σ_k are initialized to zeros; n = dimension of the inputs $\mathbf{z}_k = [x_k, y_k]$.

Step 3: Update of existing clusters: The potential of all existing clusters at time instant k are recursively updated by:

$$P_k(\mathbf{c}_r) = \frac{(k-1)P_{k-1}(\mathbf{c}_r)}{(k-2) + P_{k-1}(\mathbf{c}_r) + P_{k-1}(\mathbf{c}_r) \sum_{i=1}^n (z_{k,i} - z_{k-1,i})^2} \quad (7)$$

where $\mathbf{c}_r$ represents the x and y coordinates of all existing clusters, $r \in [1, R]$.

Step 4: Determine the winning cluster. The winning cluster is determined based on the following law:

1) If $P_k(\mathbf{z}_k) < P_k(\mathbf{c}_r)$, or the potential of the current data point is less than the potential of all existing clusters, then go to Step 6 and update consequent parameters.

2) If $P_k(\mathbf{z}_k) \geq P_k(\mathbf{c}_r)$, then determine the winning cluster in the input space and output space, respectively:

$$WC_I = \arg \min_{k=1}^K \|\mathbf{m}_{k,I} - \mathbf{x}_{k,I}\|$$

$$WC_O = \arg \min_{k=1}^K \|\mathbf{m}_{k,O} - \mathbf{y}_{k,O}\|$$

where $k \in [1, K]$, and K is the total number of data pairs.

Step 5: Recognize the fuzzy cluster structure. If $WC_I = WC_O$, then merge the new data set to the winning cluster. The winning cluster parameters are updated in the input space and output space, respectively, whereas the other cluster information remains unchanged:

$$(\sigma_{k,I})^2 := (\sigma_{k-1,I})^2 + \frac{1}{N_W} \left[(\mathbf{x}_k - \mathbf{m}_{k-1,I})^2 - (\sigma_{k-1,I})^2 \right]$$

$$\mathbf{m}_{k,I} := \mathbf{m}_{k-1,I} + \frac{\mathbf{x}_k - \mathbf{m}_{k-1,I}}{N_W}$$

$$(\sigma_{k,O})^2 := (\sigma_{k-1,O})^2 + \frac{1}{N_W} \left[(\mathbf{y}_k - \mathbf{m}_{k-1,O})^2 - (\sigma_{k-1,O})^2 \right]$$

$$\mathbf{m}_{k,O} := \mathbf{m}_{k-1,O} + \frac{\mathbf{y}_k - \mathbf{m}_{k-1,O}}{N_W}$$

where N_W is the number of samples in the winning cluster.

If $WC_I \neq WC_O$, then there is no winning cluster. Create a new cluster:

$$R := R+1, \quad N_R := 1, \quad \mathbf{m}_{R,I} \leftarrow \mathbf{x}_k, \quad \sigma_{R,I} := 0.10; \quad \mathbf{m}_{R,O} \leftarrow \mathbf{y}_k, \quad \sigma_{R,O} := 0.10.$$

These criteria are applied to exclude those clusters affected by noise. For example, two closest clusters may not be merged to one cluster if they belong to different output classes.

Step 6: Update the consequent weight parameters. The optimization is taken by the use of the hybrid training method to be discussed in the following subsection.

Step 7: Calculate the classification output. The output is computed by Equation (4). Proceed back to Step 2, until all the data samples have been input into the system (i.e., $k = K$).

2.3. The Proposed Enhanced Kalman Filter Training Method

Once the iEF reasoning structure is identified, as discussed in Section 2.2, the parameters (both linear and non-linear) should be properly optimized to improve diagnostic classification accuracy. Linear parameters will be trained by the use of the general LSE method. The non-linear parameters will be optimized by the use of

the proposed EKF.

Many training algorithms have been proposed in the literature for non-linear parameter optimization, such as the classical gradient algorithms, Levenberg-Marquardt, and Kalman filtering (KF) [14] [21]. The gradient descent (GD) algorithm is prone to being trapped by local minima; whereas Levenberg-Marquardt method cannot be effectively used for large models that will generate oversized variance matrices and significantly slow down the processing convergence. Among the KF-associated methods, the node decoupled KF (NDKF) algorithm can simplify implementation and reduce memory requirements, which outperforms other KF-related algorithms [22]. However, the accuracy of the NDKF is limited due to its sensitivity to the implementation strategy. The classical NDKF takes two steps in operation: updating and prediction [23]. In the prediction step, the posteriori states are used to estimate the state at the current time step. In the update step, the priori prediction is combined with the current information to update the state estimate and the posteriori error covariance matrix. Consider a multivariable system in the following form:

$$x_k = F_k x_{k-1} + u_{k-1} \quad (8)$$

$$y_k = H_k x_k + v_k \quad (9)$$

where x_k is a state vector $(n \times 1)$; F_k is a transition matrix $(n \times n)$; y_k is an observation vector $(1 \times n)$; H_k is an observation matrix $(1 \times n)$; u_k and v_k are the respective process noise and observation noise, which satisfy the following conditions:

$$\begin{aligned} E(u_k) &= E(v_k) = 0 \\ E(u_k u_i^T) &= \begin{cases} Q_k & \text{if } i = k \\ 0 & \text{if } i \neq k \end{cases} \\ E(v_k v_i^T) &= \begin{cases} R_k & \text{if } i = k \\ 0 & \text{if } i \neq k \end{cases} \end{aligned} \quad (10)$$

where $E(\cdot)$ denotes the expectation, Q_k and R_k are the respective process noise matrix and observation noise covariance matrix. In the prediction step, the predicted state is

$$\hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1} \quad (11)$$

where the subscript " $k|k-1$ " denotes the estimate at time instant k given observations up to steps $k-1$. The predicted estimate covariance matrix becomes

$$S_{k|k-1} = F_k S_{k-1|k-1} F_k^T + Q_k \quad (12)$$

where $S_{k-1|k-1} = \text{cov}(x_k - \hat{x}_{k|k-1})$. In the updated step, the measurement residual is computed as:

$$\tilde{d}_k = y_k - H_k \hat{x}_{k|k-1} \quad (13)$$

The optimal Kalman gain will be

$$K_k = S_{k|k-1} H_k^T (H_k S_{k|k-1} H_k^T + R_k)^{-1} \quad (14)$$

State estimate is updated by

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{d}_k \quad (15)$$

Estimate covariance matrix is updated by:

$$S_{k|k} = S_{k|k-1} + K_k H_k S_{k|k-1} \quad (16)$$

The KF performance depends on process noise covariance matrix Q and observation error covariance matrix R , which are related to the application and process dynamics [24]. The DEKF filter may diverge from the optimum due to Q and R errors. Generally, the covariance matrices are determined based on trials and errors. However, for a complex dynamic system like a gearbox, it is difficult to determine the reasonable covariance matrices in advance. On the other hand, empirical approximation process may result in significant errors, which makes the training method unreliable. Correspondingly, a covariance matrix updating method, EKF, will be proposed in this work to improve the performance of DEKF.

The covariance provides a measure of correlation between two or more random variables. The proposed EKF method is to update process noise and observation error covariance matrices, which are defined as:

$$V_k = (H_k S_{k|k-1} H_k^T + R_k)^{-1} \quad (17)$$

where $S_{k|k-1} = F_k S_{k-1|k-1} F_k^T + Q_k$.

The process noise and observation error are updated by

$$R_{k|k} = R_{k|k-1} + R_{k|k-1} (V_k)^\beta \quad (18)$$

$$Q_{k|k} = Q_{k|k-1} - Q_{k|k-1} (V_k)^\beta \quad (19)$$

where $\beta \in [0, 1]$ is a design parameter. By systematic investigation, $\beta = \frac{1}{2R}$ will be utilized in this work, where R is number of clusters.

The process noise covariance matrix and observation error covariance matrix are diagonal matrices initialized at 0.01 and 0.80, respectively. A series of simulation tests have been performed with initialization values ranging between 0.0001 and 1. After each epoch, the noise covariance matrix and observation error covariance matrix are updated using Equation (18) and Equation (19), respectively.

During the training using the EKF, with the introduction of the scaling factor, the predicted estimate covariance matrix S changes at a slower rate, whereas the process noise covariance matrix and observation error covariance matrix change at a faster rate. Since all covariance matrices are being updated, the state estimate update is more robust, making the training process more reliable, as can be noted in Section 3.

2.4. Hybrid Training of the iEF Classifier

After the iEF reasoning structure is identified, system parameters will be trained by the use of a hybrid method. In the forward pass, the non-linear premise pa-

rameters are optimized using the proposed EKF method, while the linear parameters remain fixed. In the backward pass, non-linear MF parameters remain unchanged, but linear consequent weight parameters are updated using the LSE algorithm [13]. A hybrid method usually has merits of reducing the trapping of local minima and improving the training convergence [16].

3. Performance Verification

The effectiveness of the proposed iEF technology is examined first by simulation tests using some benchmark data sets. Then it is implemented for gear condition monitoring. Some related classifiers are used for comparison: eTS [17] and TWNFI [19] trained by hybrid methods of LSE and gradient descent (GD) algorithm. Another comparison is undertaken with a self-evolving fuzzy (SEF) classifier [15]. The developed iEF classifier, trained by the same EKF and LSE algorithms, will be denoted by iEF-EKF; this comparison will be performed to examine the effectiveness of the proposed iEF evolving algorithm. The iEF classifier trained by the proposed GD and LSE, represented by iEF-GD, is used to check the effectiveness of the EKF training algorithm. All of these classifiers will use the same inputs, with same training conditions and initial values of the parameters to be updated.

3.1. Simulation Tests

Two benchmark data sets are used for these simulation tests.

3.1.1. Iris Dataset Testing

The first simulation test is undertaken using the Iris Dataset [25]. Iris dataset has 4 inputs: sepal length (x_1), sepal width (x_2), petal length (x_3), and petal width (x_4). It has 3 output states or classes: Iris Setosa, Iris Versicolour, and Iris Virginica. This test is conducted using 150 data pairs, 80 of which are used for training and remaining 70 are used for testing. The simulation is undertaken using MATLAB 2023b. In classification, once the clusters are generated, the classifier will calculate the system output. The related training algorithms are used to optimize classifier linear and non-linear parameters. **Table 1** summarizes the comparison results using the related classifiers. All of the selected classifiers are operated to achieve optimal results based on the input data. The success rates of each classifier represent the accuracy values before and after training. It is clear that the related training algorithms can clearly improve classification accuracy.

From **Table 1**, it is seen that during the verification test, the TWNFI performs better than the eTS classifier (85.84% vs. 75.67%), even though both have generated 5 clusters. The SEF and iEF classifiers have generated 3 clusters only, which can speed up the classification convergence. However, the iEF-DG outperforms the SEF (79.46% vs. 85.73%) due to iEF's more efficient evolving algorithm, which can also be related to the formulation of 4 rules vs. 3 rules of the SEF. Comparing iEF-GD and iEF-EKF, it is clear that the proposed EKF method can effectively control weights of the iEF system to improve diagnostic accuracy (95.34% vs. 85.73%)

and processing efficiency (1.39 sec vs. 1.56 sec per epoch).

Table 1. Performance comparison of the related classifiers using the Iris data.

Classifier	Success Rate (%)		No. of Clusters	No. of Rules	Average Operation Time (sec)
	Before Training	After Training			
TWNFI	80.16	85.84	5	6	2.25
eTs	64.95	75.67	5	4	1.96
SEF	74.35	79.46	3	3	2.19
iEF-GD	78.98	85.73	3	4	1.56
iEF-NFK	93.65	95.34	3	4	1.39

Figure 1(a) shows the verification process of the developed iEF-EKF technique for the Iris data. It generates four false indicators in classification, which misclassify the output data. **Figure 1(b)** represents the absolute errors during the testing process.

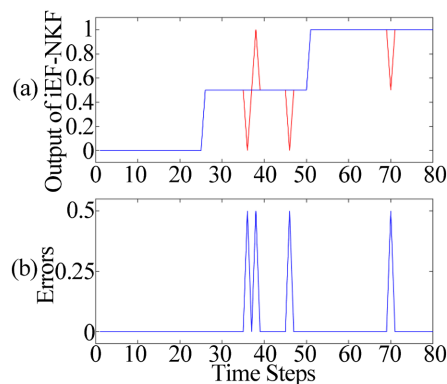


Figure 1. Test results of the iEF-EKF classifier for the Iris data: (a) Performance of the iEF-EKF with respect to the desired output (red line) and classifier's output (blue line); (b) Absolute testing errors.

3.1.2. Breast Cancer Data Testing

Another simulation test is undertaken using the Wisconsin Breast Cancer Dataset [26] to check the robustness of the proposed iEF-EKF classifier. This dataset has four input variables: glucose (x_1), homa (x_2), adiponectin (x_3), and MCP (x_4). The output has two classes: benign and malignant; or the output space is divided into 2 classes to be unbiased.

A total of 116 data pairs are selected for analysis: 60 for training and remaining 56 for verification testing. The classification results are summarized in **Table 2**. It is seen that the training can clearly improve the classification accuracy. In terms of the number of formulated clusters, the SEF and iEF are more efficient in the evolving process than the TWNFI and the eTS classifiers (*i.e.*, 4 clusters vs. 2). Since the SEF classifier adopts 2 rules only, it results in the lowest classification accuracy in this case. With the comparison of the SEF and the iEF methods (*i.e.*, iEF-GD and

iEF-EKF), it is clear that the proposed iEF evolving method is more efficient than the related methods. The iEF-EKF outperforms the iEF-GD in terms of classification accuracy (89.21% vs. 84.07%) and processing speed (0.63 sec vs. 0.88 sec), due to its efficient EKF training.

Table 2. Performance comparison of the related classifiers using the breast cancer data

Classifier	Success Rate (%)		No. of Clusters	No. of Rules	Average Operation Time (sec)
	Before Training	After Training			
TWNFI	75.39	82.17	4	5	1.61
eTs	68.94	79.31	4	3	0.78
SEF	56.33	69.72	2	2	0.91
iEF-GD	76.46	84.07	2	3	0.88
iEF-NFK	82.39	89.21	2	3	0.63

Figure 2(a) shows the processing results of the iEF-EKF classifier; it generates 4 missed alarms and 3 false alarms. **Figure 2(b)** shows the absolute testing errors.

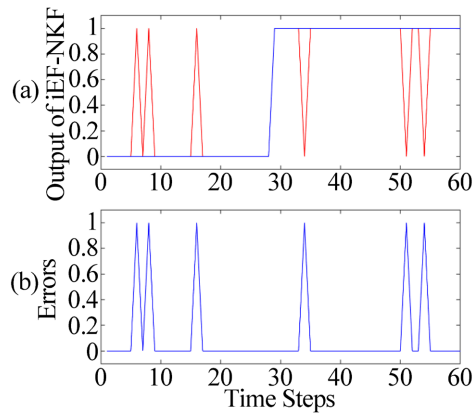


Figure 2. Test results of the iEF-EKF classifier for the breast cancer data: (a) Performance of the iEF-EKF with respect to the desired output (red line) and classifier's output (blue line); (b) Absolute testing errors.

3.2. Gear Health Condition Monitoring and Fault Diagnosis

3.2.1. Monitoring Indices

Gear fault can be classified into two categories: localized defects (e.g., broken tooth and chipped tooth) and distributed defects (e.g., scoring and wear). This work will focus on localized gear fault diagnosis because a localized fault will not only degrade transmission accuracy but also may cause sudden failures. In this work, the gear fault diagnosis is conducted gear by gear. As the measured vibration signal is generated from various vibratory sources in a gearbox, the first step is to differentiate the signal specific to each gear of interest by using a time synchronous average filter [6]. As a result, each gear signal can be processed and represented in one full revolution, called the signal average.

Many techniques have been proposed in the literature for gear fault detection, however, each technique has its own advantages and limitations. Each technique could be efficient for specific applications only. In this work, as discussed in Introduction, three features will be selected for this diagnostic classification from three information domains: energy, amplitude, and phase:

1) Beta kurtosis index (x_1): Using the overall residual signal obtained by band-stop filtering out the gear mesh frequency $f_r Z$ and its harmonics (up to the 5th harmonic), where f_r is the gear rotation frequency (Hz) and Z is the number of teeth of the gear.

2) Wavelet energy index (x_2): Also using the overall residual signal.

3) Phase demodulation index (x_3): Using the signal average.

Details of these signal processing techniques and filtering procedures can be found in papers [6] [8]. The determination of the monitoring indices can be found in paper [6], both of which are from the authors' research team.

3.2.2. Experimental Setup

Figure 3 shows the experimental setup used for performing this test.

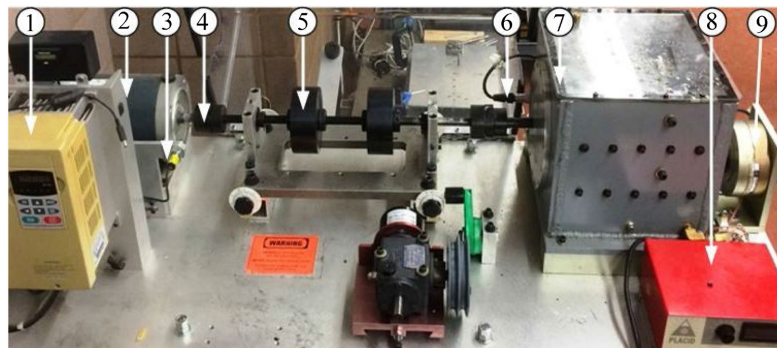


Figure 3. Experimental setup: (1) variable speed controller; (2) drive motor; (3) optical sensor; (4) flexible-coupling; (5) load disc; (6) accelerometers (sensors); (7) gearbox; (8) electric load controller; (9) magnetic brake load system.

This system is driven by a 2.2 kW induction motor, with speed ranging from 50 rpm to 4200 rpm. The motor speed is changed by using a variable frequency speed controller (VFD022B21A). A flexible coupling is used to dampen the high-frequency vibration components and shocks from the motor. An optical sensor (ROS-W, 40 mA and 3 - 15 V) is used to provide a one-pulse-per-revolution signal, used for time-synchronous average filtering operations.

The tested gear system is shown in Figure 4(a), which consists of two pairs of spur gears. The first pair has 32 and 80 teeth for the pinion and the gear, respectively. The second pair has 96 and 48 teeth for the pinion and the gear, respectively. A magnetic brake unit (B150-24-H, Placid Industries) is used to provide dynamic loads to the gear system. The vibration signals are collected using ICP accelerometers (SN98697, ICP-IMI) with sensitivity of 100 mV/g. These ICP sensors are mounted on the gearbox housing to collect data along different directions.

These sensors are connected to a data acquisition board (NI PCI-4472), attached to a computer. A software interface has been developed to control the data acquisition operations in real-time, in terms of the sensor network, sampling frequency, data size, etc.

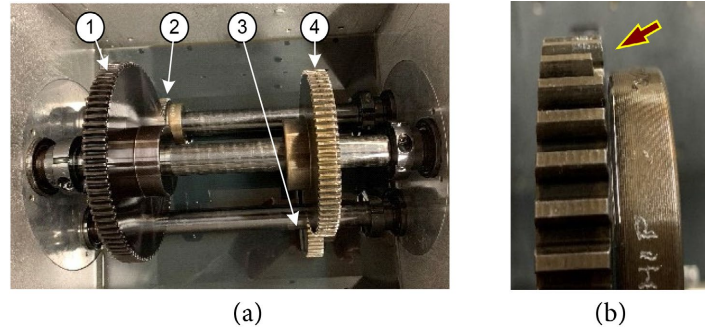


Figure 4. The tested gearbox: (a) A two-stage gear system: (1) input gear; (2) input pinion; (3) output pinion; (4) output; (b) A simulated gear damage with one tooth partially broken.

Three gear health states (classes) are tested as illustrated in **Figure 5**:

- 1) Healthy gear: 52 data sets are collected for analysis;
- 2) Cracked gear: 67 data sets are collected;
- 3) Partially broken gears: 74 data sets are collected for analysis, as shown in **Figure 4(b)**.

The health conditions of each gear are constrained to three state classes: C_1 = healthy, C_2 = cracked tooth damage, C_3 = partially broken tooth damage. In processing, the scopes are selected as: health C_1 if $y \in [0, 0.33]$, crack gear damaged C_2 if $y \in (0.33, 0.67]$, and partially broken tooth damage C_3 if $y \in (0.67, 1.0]$.

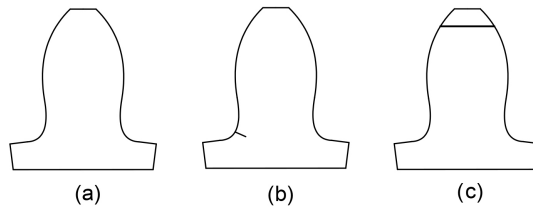


Figure 5. Tested gear classes: (a) Healthy gears; (b) Cracked gears; (c) Partially broken gear.

3.2.3. Test Results Analysis

Similarly to the conditions in simulation tests in Section 3.1, five related classifiers are used for comparison: eTS, TWNFI, SEF, iEF-DG and iEF-EKF. All of these classifiers have three inputs, and with same training conditions.

Tests are undertaken under different load and speed conditions. The sampling frequency is selected to make sure each tooth period contains about 50 data samples. For example, if the shaft speed is 1200 rpm, or $f_r = 20$ Hz, the gear has $Z = 32$ teeth, the sampling frequency should be about: $f_s = 32 \text{ teeth} \times 50 \text{ samples} \times 20 \text{ Hz} \approx 32,000 \text{ Hz}$.

Once the gear signal is collected, it is first processed by the use of the time synchronous average filtering to get signal average. Then the signal average is further

processed to generate the monitoring indices of beta kurtosis (x_1), wavelet amplitude (x_2) and phase information (x_3), which are input variables to the classifiers.

In processing, 157 data sets are used for healthy gear condition monitoring (70 for training, 32 for validation and 55 for testing); 108 data sets are used for cacked gear condition monitoring (48 for training, 25 for validation and 35 for testing); and 117 data sets are used for partially broken gear condition monitoring (55 for training, 20 for validation and 42 for testing).

Table 3 summarizes the diagnostic results using related classification techniques. In gear fault diagnosis, two types of errors are considered: 1) false alarm: the recognized gear fault is caused by other reasons (e.g., speed/load variations) instead of real gear defect; 2) missed alarm: the gear fault is not recognized by the diagnostic classifier. From **Table 3**, it is seen that the proposed iEF technique outperforms the classical eTS and TWNFI classifier, as well as the SEF classifier, with fewer clusters and higher diagnostic accuracy. That is mainly because the iEF technique has a more efficient evolving approach with the appropriate partition strategy. On the other hand, with the comparison of iEF-GD and iEF-EKF, the proposed EKF training method can improve not only classification accuracy (99.34% vs. 96.57%), but also processing efficiency (1.62 sec vs. 1.87 sec per epoch), which makes it more suitable for real-time monitoring applications.

Table 3. Gear monitoring test results using the related classifiers.

Classifier	Success Rate (%)		Healthy Gear	Cracked Gear	Partially Broken	Overall Accuracy (%)	Average Operation Time (sec)
	No. of Clusters	No. of Rules					
TWNFI	5	6	87.2	85.36	90.96	88.57	2.83
eTs	5	6	87.71	88.7	91.03	89.22	2.27
SEF	4	4	96.38	94.31	94.28	95.03	1.99
iEF-GD	3	4	97.91	95.86	97.06	96.88	1.87
iEF-NFK	3	4	99.13	97.12	98.97	98.64	1.62

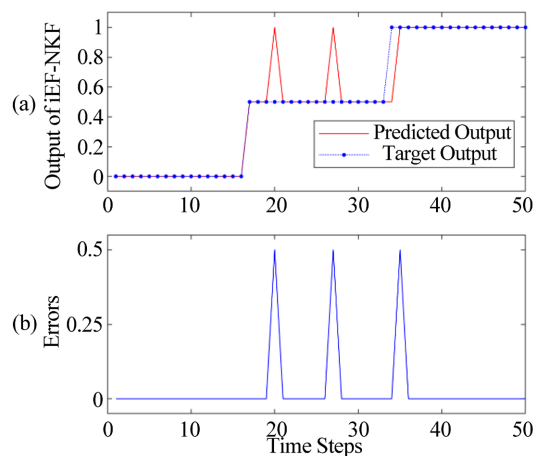


Figure 6. Test results of the iEF-EKF classifier during the test period: (a) Performance of the iEF-EKF with respect to the desired output (red line) and classifier's output (black line); (b) Absolute errors.

Figure 6(a) shows the classification process of the iEF-EKF classifier during the testing process. It is seen that iEF-EKF classifier is efficient in separating healthy state from the faulty state of the gearbox, but it has generated some errors in gear fault diagnosis with two false alarms and one missed alarm. **Figure 6(b)** illustrates the absolute errors.

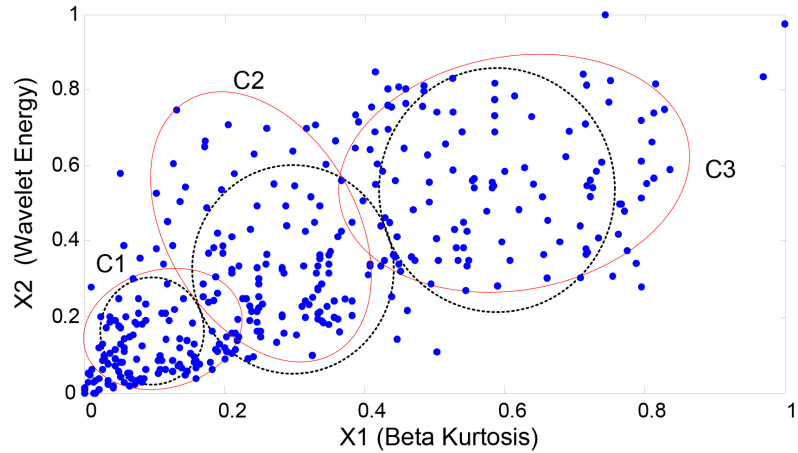


Figure 7. The output space processing results: The dotted circles C_1 - C_2 represent the constrained output space patterns. Solid circles represent the recognized clusters in the output space.

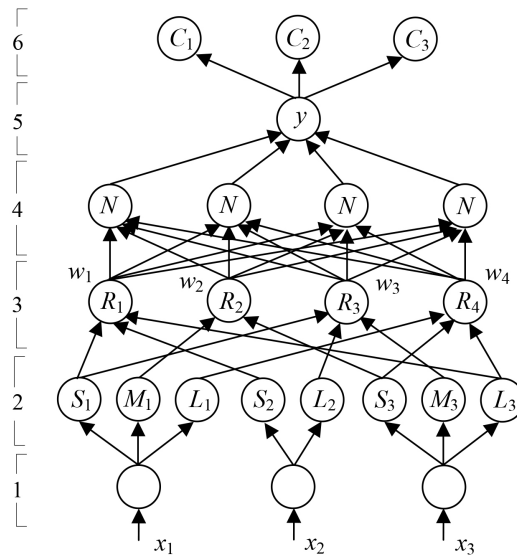


Figure 8. The identified iEF classifier model after 50 training epochs.

Figure 7 illustrates the output clusters (dotted circles) and the recognized clusters in the output space using the iEF-EKF classifier, as indicated by the solid lines, in terms of x_1 (beta-kurtosis) versus x_2 (wavelet energy amplitude). **Figure 8** shows the recognized fuzzy model architecture after 50 training epochs, after all of the training data sets have been input to the classifier. It is a 6-layer network. During the evolving process, this structure is updated gradually and continuously. Initially,

each input variable (in layer 1) had 3 MFs (in layer 2): S , M and L that are related to each cluster formation. After the evolution, 3 clusters are generated, which result in 4 rules, $R_1 - R_4$. Both x_1 and x_3 have three MFs: S , M and L . On the other hand, x_2 has two MFs only: S_2 and L_2 . S_2 is related to R_1 , while L_2 is related to R_3 . M_2 is not represented as it is not related to any reasoning rules. The firing strength of each rule is calculated in layer 3 by the related inference operation in Equation (3). After normalization in layer 4 and defuzzification (e.g., centroid), the output indicator value y can be computed in layer 5 using Equation (4).

4. Conclusion

An improved evolving fuzzy technology, iEF in short, has been developed in this work for real-time gearbox health condition monitoring and fault diagnosis. Cluster evolution is performed based on the constrained output space partitions (e.g., healthy and different gear damaged states), so as to prevent possible misleading diagnostic information. The suggested evolving algorithm has the ability of adding or subtracting clusters adaptively, and the representative patterns can be recognized between the input space and the constrained output space partitions. An enhanced Kalman filter, EKF, training method is proposed to improve parameter training efficiency and classification efficiency. The effectiveness of the developed iEF-EKF classifier has been examined by simulation tests using some benchmark data sets. It is also implemented for gear system monitoring under different gear health conditions. Test results have shown that the developed iEF classifier can effectively partition the input-output spaces with the appropriate constrained evolving strategy. It outperforms other related evolving algorithms. The proposed EKF training method can improve classification convergence with higher diagnostic accuracy and training efficiency using less processing time. It has potential to be applied for real-time gearbox health condition monitoring and fault diagnosis in industrial applications.

Acknowledgements

This work is supported in part by Natural Sciences and Engineering Research Council of Canada (NSERC), eMech Systems Inc., and Bare Point Water Treatment Plant in Thunder Bay, ON, Canada.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Schmidt, S., Wilke, D.N. and Gryllias, K.C. (2025) Generalised Envelope Spectrum-Based Signal-to-Noise Objectives: Formulation, Optimisation and Application for Gear Fault Detection under Time-Varying Speed Conditions. *Mechanical Systems and Signal Processing*, **224**, Article ID: 111974. <https://doi.org/10.1016/j.ymssp.2024.111974>
- [2] Miao, Y., Wang, J., Zhang, B. and Li, H. (2022) Practical Framework of Gini Index in

- the Application of Machinery Fault Feature Extraction. *Mechanical Systems and Signal Processing*, **165**, Article ID: 108333. <https://doi.org/10.1016/j.ymssp.2021.108333>
- [3] Li, Z., Wan, S. and Yu, J. (2025) Research on Real-Time Gear Fault Detection and Classification Technology Based on EFPI Vibration Sensor. *Optics & Laser Technology*, **192**, Article ID: 113627. <https://doi.org/10.1016/j.optlastec.2025.113627>
 - [4] Guan, Y., Liang, M. and Neculescu, D. (2019) Velocity Synchronous Bilinear Distribution for Planetary Gearbox Fault Diagnosis under Non-Stationary Conditions. *Journal of Sound and Vibration*, **443**, 212-229. <https://doi.org/10.1016/j.jsv.2018.11.039>
 - [5] Lv, C., Zhang, P. and Wu, D. (2020) Gear Fault Feature Extraction Based on Fuzzy Function and Improved Hu Invariant Moments. *IEEE Access*, **8**, 47490-47499. <https://doi.org/10.1109/access.2020.2979007>
 - [6] Wang, W. (2008) An Enhanced Diagnostic System for Gear System Monitoring. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **38**, 102-112. <https://doi.org/10.1109/tsmcb.2007.908864>
 - [7] Hong, L. and Dhupia, J.S. (2014) A Time Domain Approach to Diagnose Gearbox Fault Based on Measured Vibration Signals. *Journal of Sound and Vibration*, **333**, 2164-2180. <https://doi.org/10.1016/j.jsv.2013.11.033>
 - [8] Wang, W.Q., Ismail, F. and Farid Golnaraghi, M. (2001) Assessment of Gear Damage Monitoring Techniques Using Vibration Measurements. *Mechanical Systems and Signal Processing*, **15**, 905-922. <https://doi.org/10.1006/mssp.2001.1392>
 - [9] Combet, F. and Gelman, L. (2007) An Automated Methodology for Performing Time Synchronous Averaging of a Gearbox Signal without Speed Sensor. *Mechanical Systems and Signal Processing*, **21**, 2590-2606. <https://doi.org/10.1016/j.ymssp.2006.12.006>
 - [10] Zhang, Z., Zhang, X., Zhang, P., Wu, F. and Li, X. (2018) Gearbox Composite Fault Diagnosis Method Based on Minimum Entropy Deconvolution and Improved Dual-Tree Complex Wavelet Transform. *Entropy*, **21**, Article 18. <https://doi.org/10.3390/e21010018>
 - [11] Jing, L., Zhao, M., Li, P. and Xu, X. (2017) A Convolutional Neural Network Based Feature Learning and Fault Diagnosis Method for the Condition Monitoring of Gearbox. *Measurement*, **111**, 1-10. <https://doi.org/10.1016/j.measurement.2017.07.017>
 - [12] Lyu, H.L., Wang, W. and Liu, X.P. (2025) Semi-Tensor Product-Based Fuzzy Relation Matrix Technique for Gear System State Forecasting. *ISA Transactions*, **166**, 488-495. <https://doi.org/10.1016/j.isatra.2025.07.035>
 - [13] Li, D., Wang, W. and Ismail, F. (2014) A Fuzzy-Filtered Grey Network Technique for System State Forecasting. *Soft Computing*, **19**, 3497-3505. <https://doi.org/10.1007/s00500-014-1281-1>
 - [14] Shah, J. and Wang, W. (2022) An Evolving Neuro-Fuzzy Classifier for Fault Diagnosis of Gear Systems. *ISA Transactions*, **123**, 372-380. <https://doi.org/10.1016/j.isatra.2021.05.019>
 - [15] Jianu, O. and Wang, W. (2014) A Self-Evolving Fuzzy Classifier for Gear Fault Diagnosis. *International Journal of Mechanical and Mechatronics Engineering*, **14**, 90-96.
 - [16] Wang, W., Li, D. and Vrbanek, J. (2012) An Adaptive Evolving Technique for System Dynamic State Analysis. *Journal of Neurocomputing*, **85**, 111-119.
 - [17] Angelov, P.P. and Zhou, X.W. (2008) Evolving Fuzzy-Rule-Based Classifiers from Data Streams. *IEEE Transactions on Fuzzy Systems*, **16**, 1462-1475. <https://doi.org/10.1109/tfuzz.2008.925904>
 - [18] Pratama, M., Pedrycz, W. and Lughofer, E. (2018) Evolving Ensemble Fuzzy Classi-

- fier. *IEEE Transactions on Fuzzy Systems*, **26**, 2552-2567.
<https://doi.org/10.1109/tfuzz.2018.2796099>
- [19] Song, Q. and Kasabov, N. (2006) TWNFI—A Transductive Neuro-Fuzzy Inference System with Weighted Data Normalization for Personalized Modeling. *Neural Networks*, **19**, 1591-1596. <https://doi.org/10.1016/j.neunet.2006.05.028>
- [20] Angelov, P.P. and Filev, D.P. (2004) An Approach to Online Identification of Takagi-Sugeno Fuzzy Models. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, **34**, 484-498. <https://doi.org/10.1109/tsmcb.2003.817053>
- [21] Yang, Y. and Gao, W. (2005) Comparison of Adaptive Factors in Kalman Filters on Navigation Results. *Journal of Navigation*, **58**, 471-478.
<https://doi.org/10.1017/s0373463305003292>
- [22] Feng, Z., Zhu, W. and Zhang, D. (2019) Time-Frequency Demodulation Analysis via Vold-Kalman Filter for Wind Turbine Planetary Gearbox Fault Diagnosis under Non-stationary Speeds. *Mechanical Systems and Signal Processing*, **128**, 93-109.
<https://doi.org/10.1016/j.ymssp.2019.03.036>
- [23] Li, S., Wunsch, D.C., O'Hair, E. and Giesselmann, M.G. (2002) Extended Kalman Filter Training of Neural Networks on a SIMD Parallel Machine. *Journal of Parallel and Distributed Computing*, **62**, 544-562. <https://doi.org/10.1006/jpdc.2001.1807>
- [24] Ding, W., Wang, J and Rizos, C. (2007) Improving Covariance based Adaptive Estimation for GPS/INS Integration. *Journal of Navigation*, **60**, 517-529.
- [25] (2025) Iris Dataset. <https://archive.ics.uci.edu/ml/datasets/Iris>
- [26] (2025) Wisconsin Breast Cancer Dataset.
[https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+\(Diagnostic\)](https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic))