

Multiple Discriminant Analysis and Neural-Network-Based Monolith and Partition Fault-Detection Schemes for Broken Rotor Bar in Induction Motors

Bulent Ayhan, *Student Member, IEEE*, Mo-Yuen Chow, *Senior Member, IEEE*,
and Myung-Hyun Song, *Member, IEEE*

Abstract—Broken rotor bars in induction motors can be detected by monitoring any abnormality of the spectrum amplitudes at certain frequencies in the motor-current spectrum. It has been shown that these broken-rotor-bar specific frequencies are located around the fundamental stator current frequency and are termed lower and upper sideband components. Broken-rotor-bar fault-detection schemes should rely on multiple signatures in order to overcome or reduce the effect of any misinterpretation of the signatures that are obscured by factors such as measurement noises and different load conditions. Multiple discriminant analysis (MDA) and artificial neural networks (ANNs) provide appropriate environments to develop such fault-detection schemes because of their multiinput-processing capabilities. This paper describes two fault-detection schemes for a broken-rotor-bar fault detection with a multiple signature processing and demonstrates that the multiple signature processing is more efficient than a single signature processing. The first scheme, which will be named the “monolith scheme,” is based on a single large-scale MDA or ANN unit representing the complete operating load-torque region of the motor, while the second scheme, which will be named the “partition scheme,” consists of many small-scale MDA or ANN units, each unit representing a particular load-torque operating region. Fault-detection performance comparison between the MDA and the ANN with respect to the two schemes is investigated using the experimental data collected for a healthy and a broken-rotor-bar case. Partition scheme distributes the computational load and complexity of the large-scale single units in a monolith scheme to many smaller units, which results in the increase of the broken-rotor-bar fault-detection performance, as is confirmed with the experimental results.

Index Terms—Artificial neural networks (ANNs), broken rotor bar, discriminant analysis, fault diagnosis, induction motors.

I. INTRODUCTION

INDUCTION motors have dominated the field of electro-mechanical energy conversion, being 80% of the motors

in use [1], [2]. The applications of the induction motors are widespread. Some induction motors are the key elements in assuring the continuity of the process and production chains of many industries. A majority of the induction motors are used in electric-utility industries, mining industries, petrochemical industries, and domestic appliances industries. The list of the industries and applications that the induction motors take place in is rather long. Induction motors are also often used in critical applications such as nuclear plants, aerospace, and military applications, where the reliability must be of high standards.

The failure of induction motors can result in a total loss of the machine itself in addition to a likely costly downtime of the whole plant. More importantly, these failures may even result in the loss of lives, which cannot be tolerated. Thus, health-monitoring techniques to prevent the induction-motor failures are of great concern in the industry and are gaining an increasing attention [3]–[7].

Induction motors often operate in hostile environments such as corrosive and dusty places. They are also exposed to a variety of undesirable conditions and situations such as misoperations. These unwanted conditions can cause the induction motor to go into a premature failure period, which may result in an unserviceable condition of the motor if not detected at its early stages of the failure period. The early detection of the incipient motor fault is thus of great concern. Rotor failures are among these failures, and they now account for 5–10% of the total induction-motor failures [8]. Since 1980, the broken-rotor-bar fault-detection problem has created a substantial interest among researchers [9], [10]. Several monitoring techniques have been developed, most of which are based on a motor-current signature analysis (MCSA) [11]. In recent years, several advanced signal-processing techniques such as the high-resolution spectral analysis, higher order statistics, and wavelet analysis have been applied to the broken rotor bar and other motor fault-detection problems [2], [10], [12], [13].

MCSA techniques include parametric and nonparametric spectrum-analysis methods of the motor current in general. In the parametric methods, autoregressive models have been fitted with the time series of the signal, and model parameters have been used in order to compute the frequency spectrum. Nonparametric methods, on the other hand, are based on a Fourier transform and search for periodicities of the signal.

Manuscript received October 14, 2003; revised June 26, 2005. Abstract published on the Internet May 18, 2006.

B. Ayhan is with the Advanced Diagnosis Automation and Control (ADAC) Laboratory, North Carolina State University, Raleigh, NC 27695 USA.

M.-Y. Chow is with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA (e-mail: chow@ncsu.edu).

M.-H. Song is with the School of Information and Communication, Suncheon National University, Suncheon 540-742, Korea.

Digital Object Identifier 10.1109/TIE.2006.878301

Power-spectral-density (PSD) analysis of the motor current is one of the widely used MCSA techniques. There are several approaches to calculate the PSD. The periodogram method and Welch's periodogram methods are two of the nonparametric spectrum methods used in the broken-rotor-bar fault detection [14].

Broken-rotor-bar fault in the induction motors can be detected by monitoring any abnormality of the motor-current power spectrum amplitudes at several certain frequency components. These frequency components are located around the main frequency line and are determined according to the number of poles and mechanical speed of the motor. However, there are other effects that may obscure the detection of the broken-rotor-bar fault or cause false alarms. For example, these effects can be intrinsic manufacturing dissymmetry [15], or load-torque oscillation that can produce the stator currents with the frequency values the same as the monitored frequencies. In monitoring these frequency components, it is assumed that the load torque is constant. Any variation of the load torque with the rotational speed can produce frequency harmonics, which may overlap the harmonics caused by the broken-rotor-bar fault [16]. A broken-rotor-bar fault-detection scheme based on the multiple frequency signatures thus should be more reliable in overcoming or reducing the effect of the misinterpreted signatures, which are caused by the effects discussed formerly or some other unknown reasons. Multiple discriminant analysis (MDA) and artificial neural networks (ANNs) provide appropriate environments to develop such fault-detection schemes because of their multiinput-processing capabilities.

This paper presents two fault-detection schemes for the broken-rotor-bar fault detection with the multiple signature processing and demonstrates that the multiple signature processing is more effective than the single signature processing. The first scheme will be named the "monolith scheme," and it is based on a single MDA or a single ANN unit representing the complete motor-operating load-torque region. The second scheme will be named the "partition scheme," and it consists of several small MDA or ANN units, each of which is representing a particular load-torque operating region. In the partition scheme, the computational load and the complexity of the single units in the monolith scheme are distributed to smaller units, which as a result, increase the broken-rotor-bar fault-detection performance. The partition scheme thus can be thought of as partitioning a highly nonlinear mapping problem in a high-dimensional space into linear or low-order nonlinear mapping problems in a reduced dimensional space. The two detection schemes have been investigated using the experimental data with the MDA and ANN detection units, respectively.

This paper is organized as follows. Section II discusses the frequencies of interest to the broken-rotor-bar problem and outlines the frequencies to be used in the MDA and ANN units. Section III briefly describes the fundamental properties of the MDA and ANN. Section IV presents the experiment setup and motor-data specifications. Section V outlines the fault-detection schemes together with the experimental results and analysis. Section VI concludes with the findings of this paper.

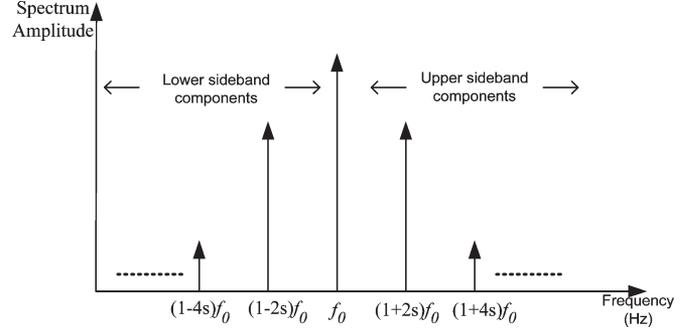


Fig. 1. Sideband frequencies around the fundamental line frequency.

II. MOTOR CURRENT SPECTRAL COMPONENTS FOR BROKEN ROTOR BAR

Kliman, Thomson and Stewart, Filipetti, and Elkasabgy *et al.* [17]–[20] used the MCSA methods to detect broken-rotor-bar faults by investigating the sideband components around the supplied current fundamental frequency (i.e., the line frequency) f_0

$$f_b = (1 \pm 2s)f_0 \quad (1)$$

where f_b are the sideband frequencies associated with the broken rotor bar, s is the per unit motor slip. The slip s is defined as the relative mechanical speed of the motor, n_m , with respect to the motor synchronous speed n_s as

$$s = \frac{n_s - n_m}{n_s}. \quad (2)$$

The motor synchronous speed, n_s , is related to the line frequency f_0 as

$$n_s = \frac{120f_0}{P} \quad (3)$$

where P is the number of poles of the motor, and the constant "120" is used to express the motor synchronous speed n_s in revolutions per minute (r/min) unit.

The broken bars also give rise to a sequence of the other sidebands given by [19]

$$f_b = (1 \pm 2ks)f_0, \quad k = 1, 2, \dots, k_n \quad \text{where } f_b > 0 \forall k \quad (4)$$

and is depicted conceptually in Fig. 1.

Fig. 1 shows the frequency components specific to the broken-rotor-bar fault, which is given in (4) for $k = 1$ and 2. These frequencies are located around the fundamental line frequency and are termed as the lower sideband and upper sideband components as indicated in Fig. 1. There are other spectral components that can be observed in the stator line current due to the broken-rotor-bar fault [17]

$$f_b = \left[\left(\frac{k}{p} \right) (1 - s) \pm s \right] f_0, \quad \text{where } k/p = 1, 3, 5, \dots \quad (5)$$

The fault-detection schemes investigated in this paper depend on the multiple signature processing. These signatures

correspond to the power spectrum amplitudes of the motor-current (phase-*a*) data at the selected frequencies. A detailed explanation about the experiment setup and specifications of the collected motor data will be given in Section IV. We use Welch's periodogram method to compute the power spectrum of the phase-*a* motor-current data. In Welch's periodogram method, we apply a Hanning window and 50% overlapping percentage among the partitioned segments. It has been noted that the use of a Hanning window and 50% overlap leads to an efficient implementation of the Fast Fourier Transform (FFT) algorithm [21]. In the fault-detection schemes, we consider four of the broken-rotor-bar fault specific frequency components. Suppose $\mathbf{F} = \{f_{1-}, f_{1+}, f_{2-}, f_{2+}\}$ is the set of the broken-rotor-bar fault specific frequency components, and $\mathbf{P} = \{p_{1-}, p_{1+}, p_{2-}, p_{2+}\}$ is the set of power spectrum amplitudes at these frequencies. The frequency components in set \mathbf{F} are: $f_{1-} = (1 - 2s)f_0$, $f_{1+} = (1 + 2s)f_0$, $f_{2-} = (1 - 4s)f_0$, and $f_{2+} = (1 + 4s)f_0$, where f_0 is the fundamental stator current frequency and s is the slip. f_{1-} and f_{1+} are the first lower and upper sidebands, while f_{2-} and f_{2+} are the second lower and upper sidebands around f_0 , which are expressed mathematically in (4). The inputs for the MDA and ANN units in the monolith and partition schemes consist of the signature set: $\{p_{1-}, p_{1+}, p_{2-}, p_{2+}\}$ and will be discussed in Section V.

III. DETECTION METHODOLOGIES FOR BROKEN-ROTOR-BAR FAULT

In this section, we will briefly describe the fundamental properties of the MDA and ANN, which are used as two detection methodologies in the fault-detection schemes.

A. MDA

MDA considers the problem of assigning individuals to certain groups that have been identified beforehand. The objective in the MDA is thus, to find a set of linear combinations of the variables whose values are as close as possible within groups and as far apart as possible between groups [22]. Let $\mathbf{X} = [x_{ij}]$ be the data matrix with m rows and n columns, where each row stands for the observations, and each column stands for the variables. The mean \bar{x}_j of variable j can be expressed as

$$\bar{x}_j = \frac{1}{m} \sum_{i=1}^m x_{ij}. \quad (6)$$

Suppose that the m rows of \mathbf{X} are separated into q groups, and each group is defined by a set I_k of m_k values of index i , where the relation between m_k and m is expressed as

$$\sum_{k=1}^q m_k = m. \quad (7)$$

Let \bar{x}_{kj} be the mean variable j in group k ; \bar{x}_{kj} can then be expressed as

$$\bar{x}_{kj} = \frac{1}{m_k} \sum_{i \in I_k} x_{ij}. \quad (8)$$

The mean \bar{x}_j of variable j can be expressed as

$$\bar{x}_j = \sum_{k=1}^q \frac{m_k}{m} \bar{x}_{kj}. \quad (9)$$

The total covariance between two variables can be described in two different ways, which are expressed as

$$\text{cov}(j, j') = \frac{1}{m} \sum_{i=1}^m (x_{ij} - \bar{x}_j)(x_{ij'} - \bar{x}_{j'}) \quad (10)$$

$$\text{cov}(j, j') = \frac{1}{m} \sum_{k=1}^q \left[\sum_{i \in I_k} (x_{ij} - \bar{x}_j)(x_{ij'} - \bar{x}_{j'}) \right]. \quad (11)$$

The total covariance can be decomposed into two parts: one of the two parts stands for the within-group variances, and the other part stands for the between-group variances. This decomposition is known as Huyghens decomposition formula [22] and is expressed as

$$\text{cov}(j, j') = \frac{1}{m} \sum_{k=1}^q \left[\sum_{i \in I_k} (x_{ij} - \bar{x}_{kj})(x_{ij'} - \bar{x}_{kj'}) \right] + \sum_{k=1}^q \frac{m_k}{m} (x_{kj} - \bar{x}_j)(x_{kj'} - \bar{x}_{j'}). \quad (12)$$

Equation (12) can be interpreted as the sum of the covariance within the groups (first term) and the covariance between the groups (second term) and is expressed in a matrix form as

$$\mathbf{T} = \mathbf{W} + \mathbf{B} \quad (13)$$

where \mathbf{T} denotes the total covariance, \mathbf{W} denotes the covariance within the groups, and \mathbf{B} denotes the covariance between the groups.

In order to define the problem addressed by the MDA, $a(i)$ is introduced, where $a(i)$ corresponds to the value of a linear combination of the n centered variables for individual i .

$$a(i) = \sum_{j=1}^n a_j (x_{ij} - \bar{x}_j). \quad (14)$$

The variance $\text{var}(\mathbf{a})$, where \mathbf{a} is the vector composed of components a_1, \dots, a_n , can be expressed as

$$\text{var}(\mathbf{a}) = \frac{1}{m} \sum_{i=1}^m a^2(i) = \frac{1}{m} \sum_{i=1}^m \left[\sum_{j=1}^n a_j (x_{ij} - \bar{x}_j) \right]^2. \quad (15)$$

By expanding (15), $\text{var}(\mathbf{a})$ can be written in the form depicted as [22]

$$\text{var}(\mathbf{a}) = \frac{1}{m} \sum_{i=1}^m \sum_{j=1}^n \sum_{j'=1}^n a_j a_{j'} (x_{ij} - \bar{x}_j). \quad (16)$$

Equation (16) can be expressed in a matrix form as $\mathbf{a}^T \mathbf{T} \mathbf{a}$. The variance of a linear combination \mathbf{a} of the variables can

then be separated into within variance and between variance as: $\mathbf{a}^T \mathbf{T} \mathbf{a} = \mathbf{a}^T \mathbf{W} \mathbf{a} + \mathbf{a}^T \mathbf{B} \mathbf{a}$. The objective of MDA is to find a linear combination of the variables that have a maximum between groups variance and a minimum within groups variance. The aim is to find \mathbf{a} such that $\mathbf{a}^T \mathbf{B} \mathbf{a} / \mathbf{a}^T \mathbf{W} \mathbf{a}$ is maximum. This linear combination is termed as the linear discriminant function. The process to find the linear discriminant function is well described in the literature, and readers can refer to [22] and [23] for further information.

MDA, for the two-class problem with the linear discriminant function, is widely used and is known as the linear discriminant analysis (LDA). In addition, there are MDA methods with nonlinear discriminant functions for problems with nonlinear classification boundaries [24].

B. ANN

ANNs have been extensively used in mimicking and automating the human expertise and experts' knowledge in many engineering fields like the fault detection and diagnosis, control, and robotics [1], [4]–[7], [19], [25]. They have been widely applied in a system identification and modeling complex nonlinear systems, where obtaining a mathematical model is arduous and not easy. This extensive use and interest in ANNs results from the capability of ANNs in learning the input–output mapping of a system [25].

To have a better understanding of how the neural networks learn from the experts' knowledge, it is convenient to have some background knowledge about the structure of the neural network and the corresponding training methodology. In this study, we use a three-layered feedforward neural network and Levenberg–Marquardt algorithm (LM) as the training method. In this section, a brief information will be given about the feedforward structure of the neural networks and training algorithms with the emphasis on the LM algorithm.

The simplest multilayer feedforward neural network is composed of three layers: input, hidden, and output [25]. In each network layer, there are processing units called nodes or neurons. Every node in a neural-network layer sends its output to all the nodes in the next layer. The input layer in the neural network acts as an interface, which takes the information from the outside world and transmits it to the internal processing units of the network. Each node in the hidden and output layer receives signals from the nodes of the previous layer, scaled by the weights, $\mathbf{w}_j = [w_{j1}, w_{j2}, \dots, w_{jk}]^T$. The output of the j th node in layer l , o_j , for $j = 1, 2, \dots, k$ can be found from the quantity expressed as

$$o_j = f \left(\sum_{i=1}^k w_{ji} v_i + b_j \right) = f(\mathbf{w}_j^T \mathbf{v} + b_j) \quad (17)$$

where b_j is the bias of the j th neuron, and f is the activation function. The activation function can be any function that is monotonically increasing and differentiable. The sigmoid function is perhaps the most popular activation function because

of its similarity to the behavioral property of many biological neurons. The sigmoid function can be expressed as

$$o_j = \frac{1}{1 + e^{-(\mathbf{w}_j^T \mathbf{v} + b_j)}}. \quad (18)$$

The network training process can be thought of as rendering the network to learn the mapping through the given input–output relation of the system by adjusting the network internal parameters or the so-called weights, \mathbf{w} . Let a system have inputs $\mathbf{u} = [u_1, u_2, \dots, u_n]^T$ and outputs $\mathbf{y}(\mathbf{u}) = [y_1, y_2, \dots, y_m]^T$ correspondingly. There exists a mapping \mathcal{M} from the input space $\mathcal{A} : \{\mathbf{u} \in \mathcal{U} | \mathbf{u} \text{ is the input to the system}\}$ to the output space $\mathcal{Y} : \{y(u) \in \mathcal{Y} | y \text{ is the output of the system with input } \mathbf{u}\}$ and can be expressed as

$$\mathcal{M} : \mathcal{U} \rightarrow \mathcal{Y}. \quad (19)$$

The aim in training a neural network is to find the appropriate weights \mathbf{w} in the weight space \mathcal{W} , such that the difference between the estimated outputs $\hat{\mathbf{y}}(\mathbf{u}, \mathbf{w})$ and actual outputs $\mathbf{y}(\mathbf{u})$ is minimal. This relation can be represented mathematically by

$$\min_{\mathbf{w} \in \mathcal{W}} \|\hat{\mathbf{y}}(\mathbf{u}, \mathbf{w}) - \mathbf{y}(\mathbf{u})\|_2 \quad (20)$$

where $\|\cdot\|_2$ is Euclidian norm function and $\|\mathbf{x}\|_2 = (\mathbf{x}^T \mathbf{x})^{1/2}$.

Assuming that there are P input–output training patterns available to train the neural network, we will define \mathbf{e}_p as the difference between $\hat{\mathbf{y}}(\mathbf{u}, \mathbf{w})$ and $\mathbf{y}(\mathbf{u})$ for pattern p

$$\mathbf{e}_p = \hat{\mathbf{y}}(\mathbf{u}_p, \mathbf{w}) - \mathbf{y}(\mathbf{u}_p). \quad (21)$$

We can now define the training error for pattern p , e_p^{tr} , by using \mathbf{e}_p as expressed as

$$e_p^{\text{tr}} = \|\mathbf{e}_p\|_2. \quad (22)$$

The network training error, which is expressed by the notation E , stands for the training error of all patterns and is a function of the network weights, \mathbf{w} . $E(\mathbf{w})$ is computed by

$$E(\mathbf{w}) = \sum_{p=1}^P (e_p^{\text{tr}})^2. \quad (23)$$

Training neural networks is based on minimizing $E(\mathbf{w})$, expressed in (23). Modification of the network weights properly at every training step results in the minimization of $E(\mathbf{w})$. There are well-known training algorithms, among them are: the backpropagation, Gauss–Newton, and LM algorithms. The backpropagation algorithm is the steepest descent method that searches for an optimal \mathbf{w} to minimize the error in (23). The LM algorithm, on the other hand, is an approximation to the well-known Gauss–Newton's method [26]. Suppose that we want to minimize $E(\mathbf{w})$ with respect to \mathbf{w} ; then Gauss–Newton's method for the weight update rule can be expressed as [26]

$$\Delta \mathbf{w} = -[\nabla^2 E(\mathbf{w})]^{-1} \nabla E(\mathbf{w}) \quad (24)$$

where $\nabla^2 E(\mathbf{w})$ is the Hessian matrix, and $\nabla E(\mathbf{w})$ is the gradient. We can express $\nabla^2 E(\mathbf{w})$ and $\nabla E(\mathbf{w})$ as

$$\nabla E(\mathbf{w}) = J^T(\mathbf{w}) \cdot \begin{bmatrix} e_1(\mathbf{w}) \\ e_2(\mathbf{w}) \\ \vdots \\ e_P(\mathbf{w}) \end{bmatrix} \quad (25)$$

$$\nabla^2 E(\mathbf{w}) = J^T(\mathbf{w})J(\mathbf{w}) + S(\mathbf{w}) \quad (26)$$

where $J(\mathbf{w})$ is the Jacobian matrix

$$J(\mathbf{w}) = \begin{bmatrix} \frac{\partial e_1(\mathbf{w})}{\partial w_1} & \frac{\partial e_1(\mathbf{w})}{\partial w_2} & \dots & \frac{\partial e_1(\mathbf{w})}{\partial w_n} \\ \frac{\partial e_2(\mathbf{w})}{\partial w_1} & \frac{\partial e_2(\mathbf{w})}{\partial w_2} & \dots & \frac{\partial e_2(\mathbf{w})}{\partial w_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_P(\mathbf{w})}{\partial w_1} & \frac{\partial e_P(\mathbf{w})}{\partial w_2} & \dots & \frac{\partial e_P(\mathbf{w})}{\partial w_n} \end{bmatrix} \quad (27)$$

and $S(\mathbf{w}) = \sum_{i=1}^P e_i(\mathbf{w}) \cdot \nabla^2 e_i(\mathbf{w})$.

For the Gauss–Newton method, it is assumed that $S(\mathbf{w}) = 0$. Thus, the weight update rule for Gauss–Newton method becomes

$$\Delta \mathbf{w} = [J^T(\mathbf{w})J(\mathbf{w})]^{-1} J^T(\mathbf{w})\mathbf{e}(\mathbf{w}). \quad (28)$$

The weight update rule for the LM algorithm is expressed as

$$\Delta \mathbf{w} = [J^T(\mathbf{w})J(\mathbf{w}) + \mu I]^{-1} J^T(\mathbf{w})\mathbf{e}(\mathbf{w}) \quad (29)$$

where μ is the training parameter. If a training step results in an increase in $E(\mathbf{w})$, the parameter μ is multiplied by some factor λ ; if the training step results in a decrease in $E(\mathbf{w})$, the parameter μ is divided by λ [26].

The LM algorithm is often considered to give a better convergence performance when compared to the Gauss–Newton and steepest descent methods on neural network minimization problems [27]. It has been shown that the LM algorithm divides the parameters into two subsets [28]. In one subset, the convergence takes place as if it is a Gauss–Newton convergence, and on the other subset, as if it is the steepest descent. In this way, a faster convergence is acquired [28].

IV. EXPERIMENT SETUP AND MOTOR-DATA SPECIFICATIONS

We performed invasive experiments on an actual induction motor in order to obtain an actual fault data for our analyses. The characteristics of the three-phase induction motor used in our experiment are listed in Table I. The motor was tested with a healthy rotor and with a faulty rotor that had one broken rotor bar. The broken-rotor-bar fault was induced by filling one of the rotor bars with an anchoring cement before the die-casting process. Anchoring cement is a high-strength fast-setting gypsum cement with a low conductivity. The overall data-collection scheme and the actual experiment setup picture are depicted in Figs. 2 and 3, respectively.

The induction motor was fed through a three-phase ABB, ACS 501 inverter. A Tektronix TM 5003 current amplifier amplifies the induction motor stator currents before they are sent

TABLE I
INDUCTION-MOTOR CHARACTERISTICS USED IN THE EXPERIMENT

Description	Value
Power	0.75 kW (1Hp)
Input Voltage	380 V
Full Load Current	2.2 A
Supply Frequency	60 Hz
Number of Poles	4
Number of Rotor Slots	44
Number of Stator Slots	36
Full Load Torque	0.43 kg-m
Full Load Speed	1690 rpm

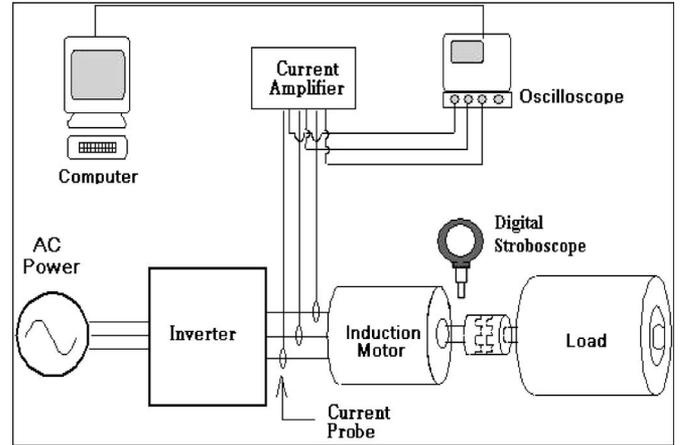


Fig. 2. Motor-data collection scheme.

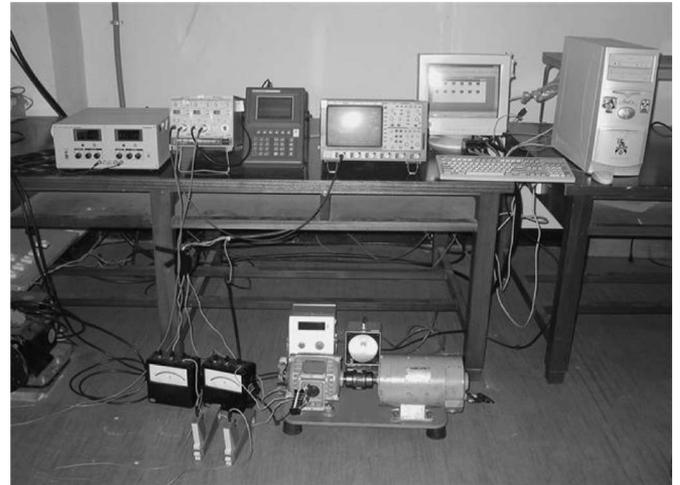


Fig. 3. Actual experiment setup to collect the healthy and faulty motor data.

to the interfacing Pentium PC through the oscilloscope. The needed load condition of the induction motor was established by connecting the test motor to a dc motor, which is used as a generator and is capable of simulating any desired load condition. The speed of the induction motor was measured by a digital stroboscope.

The experiments involved collecting the three-phase stator induction motor-current and speed data for four different load conditions of the motor, both with one broken-rotor-bar fault and without any fault. The load conditions of the motor are 25%, 50%, 75%, and full-load, respectively. These load-condition percentages are determined according to the motor

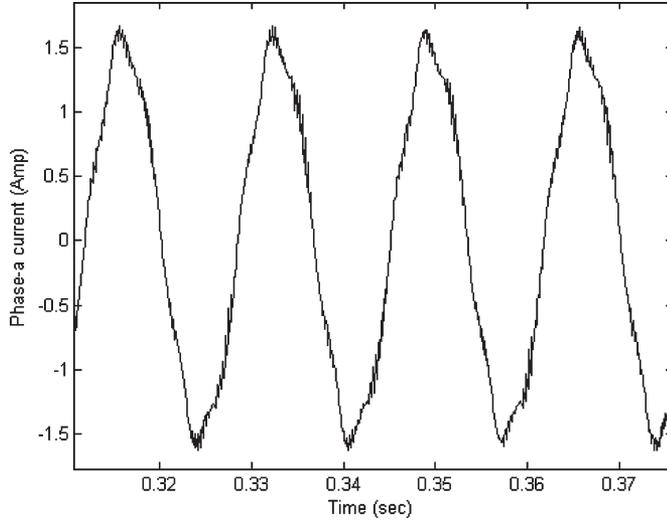


Fig. 4. Healthy phase-a motor-current data collected from the experiment setup under a full-load condition of the motor.

nameplate information given in Table I. Thus, there are a total of eight different experiment cases. For each individual case, 20 sets of motor-current data were collected with a sampling frequency of 10 kHz. Thus, each motor-current data set contains 10 000 samples for a duration of 1 s. A sample phase-a motor-current data waveform corresponding to healthy motor state under a full-load condition is demonstrated in Fig. 4.

V. FAULT-DETECTION SCHEMES AND EXPERIMENTAL RESULTS

Let us introduce some notations before we describe the two fault-detection schemes. Let C denote the motor condition, which is comprised of the two states of the motor: healthy condition H and faulty condition with one broken rotor bar F , $C = \{H, F\}$. From the motor-current power spectrum analysis and broken-rotor-bar specific frequency components knowledge, there exists a mapping \mathcal{M} from (\mathbf{P}, T_L) to C as expressed as

$$\mathcal{M} : (\mathbf{P}, T_L) \rightarrow C \quad (30)$$

where $\mathbf{P} = \{p_{1-}, p_{1+}, p_{2-}, p_{2+}\}$ is the set of Welch's periodogram power spectrum amplitudes at frequencies $\mathbf{F} = \{f_{1-}, f_{1+}, f_{2-}, f_{2+}\}$, and T_L is the motor load condition. In our experiments, we have collected data at four different load conditions of the motor as discussed in Section III, where T_L is either at full-load, 75%, 50%, or 25% load, $T_L \in \{T_{L100\%}, T_{L75\%}, T_{L50\%}, T_{L25\%}\}$. In these experiments, motor speed data collected at a particular load condition are the same or very close to each other. Fig. 5 demonstrates the PSD estimates at one of the broken-rotor-bar fault specific frequencies, $f_{1-} = (1 - 2s)f_0$, under four different motor load conditions: $T_L \in \{T_{L100\%}, T_{L75\%}, T_{L50\%}, T_{L25\%}\}$. In this figure, symbol "o" represents the PSD estimate of a healthy motor data, while symbol "x" represents the PSD estimate of a faulty motor data with one broken rotor bar.

In the monolith scheme, we use a single fault mapping unit \mathcal{M} that maps the signatures extracted throughout the complete

motor-operating load condition to the motor condition both for the MDA and ANN. This scheme is depicted in Fig. 6. Inputs of the mapping unit, for either MDA or ANN, consist of the signature set $\{T_L, p_{1-}, p_{1+}, p_{2-}, p_{2+}\}$, where T_L is the motor load condition, p_{1-}, p_{1+} are the first, and p_{2-}, p_{2+} are the second lower and upper sidebands' power spectrum amplitudes around the fundamental current frequency f_0 , respectively.

In the partition scheme, the single mapping unit \mathcal{M} , which is used in the monolith scheme, is divided into disjoint submapping units \mathcal{M}_i , where \mathcal{M}_i corresponds to a submapping unit for a particular motor load condition, and $\mathcal{M} = \bigcup_{i=1}^m \mathcal{M}_i$, $\bigcap_{i=1}^m \mathcal{M}_i = \emptyset$. In our case, $m = 4$. The partition scheme is depicted in Fig. 7. The mapping units in the partition scheme provide a partitioning of the complete motor-operating load region into the subregions, each subregion corresponding to a constant load condition. This procedure thus transforms the nonlinear mapping problem into linear mapping problems or mapping problems with lower order of nonlinearities. The partition scheme needs a motor load-condition information as a prerequisite for the preparation of the corresponding mapping units. In our case, we partition motor's load operating region into four subregions, depicted as $T_L = \{T_{L100\%}, T_{L75\%}, T_{L50\%}, T_{L25\%}\}$. We then form the MDA or ANN units for each particular load subregion using the corresponding motor-current signatures.

For analyses and performance comparisons of the fault-detection schemes with the MDA and ANN units, two different cases are taken into consideration in this paper.

- Case 1) Motor signature data set is treated as one whole training data set, where DataSet $\mathbf{D} = \{\text{Training Set}\} = \{\text{Test Set}\}$.
- Case 2) Motor signature data set is separated into two sets: training and test, where DataSet $\mathbf{D} = \{\text{Training Set}\} \cup \{\text{Test Set}\}$ and $\{\text{Training Set}\} \cap \{\text{Test Set}\} = \emptyset$.

In this paper, the experimental motor data are collected, such that they fall into two main groups: healthy state and faulty state of the motor (with one broken rotor bar). Since this is a two-class problem, the term LDA will be used instead of the MDA in the remainder of this paper. Detailed information about the general principles of the MDA is presented in Section III.

In the trials for finding the proper ANN structures for both monolith and partition schemes, it is observed that there is a substantial range of the ANN structures that can be selected among, which provide an efficient and satisfactory performance. For example, for Case 1 with a monolith scheme, it is examined that (5-2-1), (five inputs, two hidden units in the hidden layer, and one output in the output layer), (5-3-1), ..., (5-10-1) provide the same correct detection performance: 100.0%. In this study, we have selected an ANN (5-10-1) structure for the large-scale unit in the monolith scheme, and an ANN (4-5-1) structure to represent the small-scale units for each particular load condition in the partition scheme. In the selection of these structures, we have taken into consideration that there is a larger number of units in the hidden layer than the input layer. The LM training algorithm is used in both schemes. For Case 1, we have stopped training if the network training error reaches a preset value, which in our case is $1e-5$.

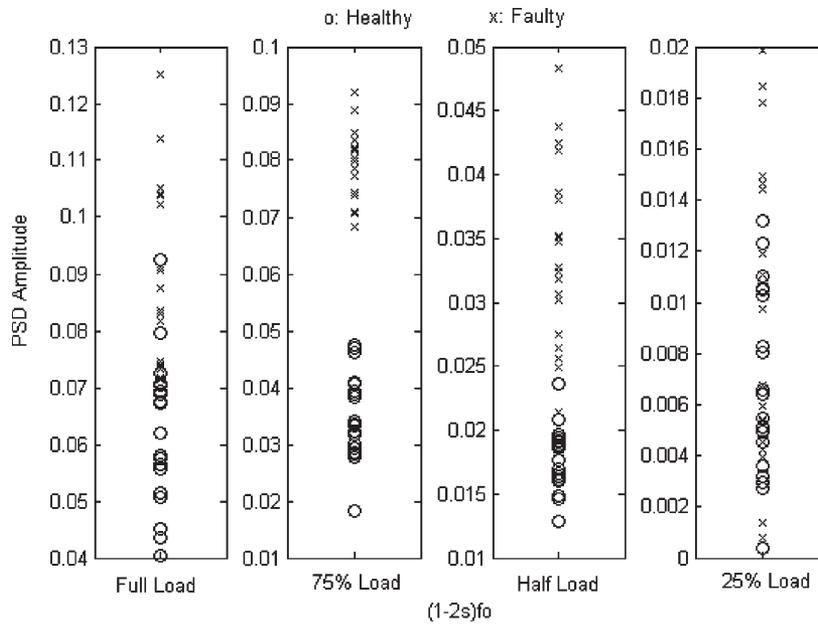


Fig. 5. PSD estimates at $f_{1-} = (1 - 2s)f_0$ under motor load conditions: $T_L \in \{T_{L100\%}, T_{L75\%}, T_{L50\%}, T_{L25\%}\}$.

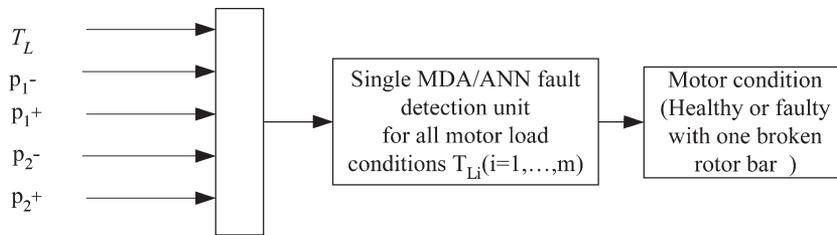


Fig. 6. Monolith scheme.

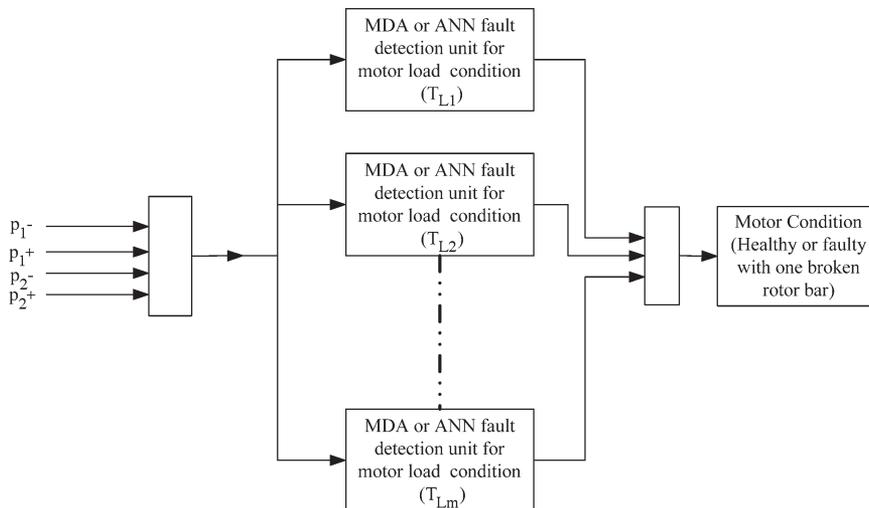


Fig. 7. Partition scheme.

In Case 2, we consider the same ANN structures but apply a different training stop technique, which is known as cross validation. Setting a fixed training error value to stop the training process may cause overtraining. Cross validation is a method to prevent overtraining. According to this technique, data are divided into two disjoint sets: the first data set is the training set, which is used for computing and updating the neural network

weights and biases; and the second set is used as the validation set [29]. We will also use the term test set as the validation set in our analyses. The error on the validation set is checked throughout the training process. It can be anticipated that the error in the validation set decreases during the initial steps of the training, just as the error in the training set. However, when the network begins to overfit the data in the training set, the

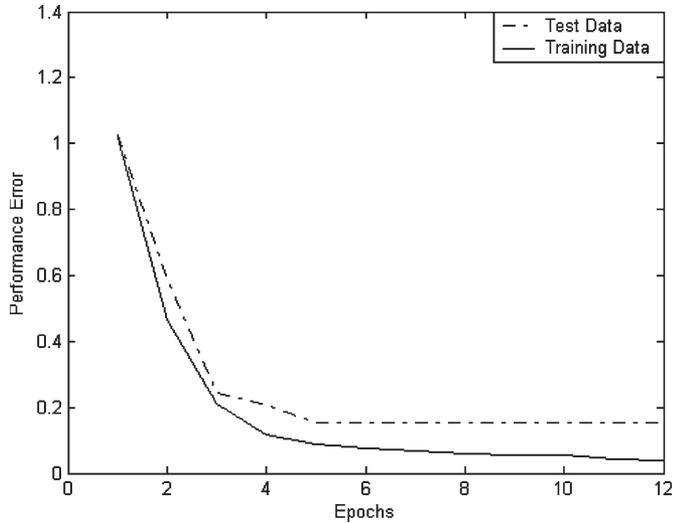


Fig. 8. Training and validation error curves with a cross-validation technique.

error for the validation set begins to rise. When the validation error increases for a specified number of steps, the training is stopped to avoid overtraining, and the most recent weights and the biases are used as the neural-network parameters [29]. Fig. 8 illustrates the principle of the cross-validation technique. In this figure, it can be seen that the training stops at the 12th epoch, since the validation error corresponding to the validation set or test data set does not decrease, but slightly increases for some consecutive steps, while training error continually decreases.

In the former part of this section, we introduced the two fault-detection schemes with the multiple signature processing for the broken-rotor-bar fault detection, which can use either the ANN or the LDA as the detection units. Experimental results based on the real motor data will also be presented in this section to affirm the notion that the multiple signature processing is more efficient than the single signature processing in addition to the performance comparison of the two detection schemes. In order to compare the fault-detection performances of the single signature and multiple signature processing, we have applied the LDA to each of the four signatures individually for both Case 1 and Case 2.

In analyzing the fault-detection performances of the two detection schemes, we will use a statistical hypothesis testing, Type I error α and Type II error β . Hypothesis testing is one of the most common aspects in the statistical inference [30]. An experimenter first determines a hypothesis about a population parameter. This hypothesis is termed as “null hypothesis” and is depicted with the notation H_0 . The term “population” is used to indicate that the population is composed of an entire set of objects or observations that have something in common. In our case for example, a healthy and faulty motor data are two different populations. The aim of the hypothesis testing is to test the applicability of the null hypothesis in the knowledge of the experimental data. If the null hypothesis is rejected while it is actually true, this error is termed as Type I error α . In a similar way, if the null hypothesis fails to be rejected while it is actually false, this error is termed as Type II error β . Type I and II error definitions are introduced in Table II.

TABLE II
TYPE I AND TYPE II ERROR DEFINITION

Decision	H_0 is true	H_0 is false
Reject H_0	Type I error	Correct
Do not reject H_0	Correct	Type II error

We will use a hypothesis test to analyze the fault-detection performances of the investigated monolith and partition schemes with the LDA and ANN. The null hypothesis H_0 in our case is stated as

H_0 : Incoming motor signature test data correspond to healthy state of the motor.

Type I error α will then correspond to the ratio of the healthy motor data, which are classified as faulty according to the LDA and ANN to the total number of the motor data. Likewise, Type II error β will correspond to the ratio of the faulty motor data, which are classified as healthy to the total number of the motor data. Apparently, Type II error has more severe consequences than Type I error in engineering applications. For example, in a motor manufacturing company, Type II error causes labeling faulty motors as the healthy motors and shipment of these faulty motors to the consumers. With Type I error, on the other hand, healthy motors are labeled as faulty, and a further analysis of these motors helps to identify and fix the problem without affecting any consumers. We will use the term “correct detection rate” (CDR), in our analyses, which are mathematically expressed as

$$CDR = (1 - \alpha - \beta). \tag{31}$$

Tables III and IV depict the CDRs of each single signature for Case 1 and Case 2 together with Type I and Type II error measures. These tables indicate that CDRs change significantly according to each individual signature. Among these signatures, $(1 - 4s)f_0$ has the highest CDR, while the other three signatures have lower CDRs. The seventh to ninth rows of Table IV simply sum the CDRs and Type I–II errors of training and test data sets. We have then considered four of the signatures together and applied the monolith and partition schemes.

Table V depicts the CDR of LDA and ANN for the two schemes under Case 1 together with Type I and Type II error measures. Note that the ANN gives a perfect correct detection performance in both schemes, while LDA’s performance improves with a partition scheme. It is also observed that CDRs in both schemes, either with ANN or LDA, are higher than any of the single signature’s CDRs given in Tables III and IV.

The bar chart depicted in Fig. 9 presents the CDRs of both single signature processing and multiple signature processing for the two schemes under Case 1. This bar chart affirms that the multiple signature processing is more efficient in the broken-rotor-bar fault detection as expressed mathematically as

$$\begin{aligned} CDR_{ANN_partition} &= CDR_{ANN_monolith} > CDR_{LDA_partition} \\ &> CDR_{LDA_monolith} > CDR_{(1-4s)f_0} \\ &> CDR_{(1+2s)f_0} > \dots \end{aligned} \tag{32}$$

TABLE III
CDRs OF SINGLE FREQUENCY COMPONENTS UNDER CASE 1

	Correct Detection Rate (CDR)			
	$(1-2s)f_o$	$(1+2s)f_o$	$(1-4s)f_o$	$(1+4s)f_o$
CDR	132/160=82.5%	150/160=93.75%	152/160=95.0%	117/160=73.12%
Type I Error	10/160=6.25%	2/160=1.25%	1/160=0.63%	21/160=13.13%
Type II Error	18/160=11.25%	8/160=5.00%	7/160=4.37%	22/160=13.75%

TABLE IV
CDRs OF SINGLE FREQUENCY COMPONENTS UNDER CASE 2

	Correct Detection Rate (CDR)			
	$(1-2s)f_o$	$(1+2s)f_o$	$(1-4s)f_o$	$(1+4s)f_o$
CDR (Training)	72/80=90.00%	76/80=95.00%	75/80=93.75%	57/80=71.25%
Type I Error (Training)	4/80=5.00%	1/80=1.25%	1/80=1.25%	10/80=12.5%
Type II Error (Training)	4/80=5.00%	3/80=3.75%	4/80=5.00%	13/80=16.25%
CDR (Test)	61/80=76.25%	75/80=93.75%	77/80=96.25%	61/80=76.25%
Type I Error (Test)	4/80=5.00%	0/80=0.0%	0/80=0.0%	10/80=12.5%
Type II Error (Test)	15/80=18.75%	5/80=6.25%	3/80=3.75%	9/80=11.25%
CDR (Training+Test)	133/160=83.13%	151/160=94.38%	152/160=95.0%	118/160=73.75%
Type I Error (Training+Test)	8/160=5.00%	1/160=0.63%	1/160=0.63%	20/160=12.5%
Type II Error (Training+Test)	19/160=11.87%	8/160=5.00%	7/160=4.37%	22/160=13.75%

TABLE V
CDRs OF THE LDA AND ANN WITH MONOLITH
AND PARTITION SCHEME UNDER CASE 1

	Correct Detection Rate (CDR)	
	Monolith Scheme	Partition Scheme
CDR (LDA)	153/160=95.63 %	159/160=99.38 %
Type I Error (LDA)	1/160=0.63%	0/160=0.0%
Type II Error (LDA)	6/160=3.75%	1/160=0.63%
CDR (ANN)	160/160=100.0 %	160/160=100.0 %
Type I Error (ANN)	0/160=0.0%	0/160=0.0%
Type II Error (ANN)	0/160=0.0%	0/160=0.0%

Similarly, Table VI depicts the CDRs and Type I-II errors of LDA and ANN for the two schemes under Case 2. In Case 2, since we have separated the data into training and test sets, we have included the sum of the training and test data sets' CDRs in addition to each set's separate CDR. There is an improvement examined in the CDR with the partition scheme, both for LDA and ANN. In addition, these CDRs are higher than any of the single signature's CDRs that are depicted in Tables III and IV with the exception of one equal case.

The partition scheme has provided an improved correct detection performance both in the LDA and ANN. However, a significant performance increase is observed in the LDA rather than in the ANN. This is because, the ANN has already given a satisfactory response in the monolith scheme because of its nonlinear mapping and universal approximation capability. The partition scheme thus provides a way to cope with the nonlinearities in the mapping process especially for the LDA. Partitioning the initial mapping space of the fault-detection problem with respect to one of its input variables into smaller disjoint subregions and introducing submapping units, either ANN or LDA, for each of these small subregions provide an increase in the correct detection performance.

The three-dimensional (3-D) bar chart depicted in Fig. 10 presents the CDRs of both the single signature and multiple signature processing for the two schemes under Case 2. This bar chart clearly indicates that the multiple signature processing

is advantageous to the single signature processing and is mathematically expressed in (33). It can be also noticed that CDRs are improved with the partition scheme for both the LDA and ANN, being most significant in the LDA

$$\begin{aligned}
 \text{CDR}_{\text{ANN_partition}} &> \text{CDR}_{\text{LDA_partition}} \\
 &= \text{CDR}_{\text{ANN_monolith}} > \text{CDR}_{\text{LDA_monolith}} \\
 &= \text{CDR}_{(1-4s)f_o} > \text{CDR}_{(1+2s)f_o} > \dots \quad (33)
 \end{aligned}$$

VI. CONCLUSION

Multiple signature processing for the broken-rotor-bar fault detection is considered to be more reliable and effective than the single signature processing, because of the possibility of the obscuring effects that can overlap the significance of the one and only inspected signature. ANN and LDA provide suitable environments to process the multiple signatures for the broken-rotor-bar fault detection. In this paper, we have demonstrated that the multiple signature processing provides a better accuracy with respect to the fault-detection performance when compared to the single signature processing. In addition to this finding, we have investigated two fault-detection schemes for the broken-rotor-bar fault detection with the multiple signature processing: the monolith and the partition schemes. Experimental results show that the correct detection performance of the ANN is higher than the LDA in both schemes, which is an expected result, because the ANN is a universal approximator even for nonlinear functions. With the partition scheme on the other hand, the correct detection performances of the ANN and LDA both improve, being most significant in the LDA. The partition scheme reduces the dimension of the initial mapping space by partitioning it into smaller disjoint subregions and introduces submapping units, either ANN or LDA for each particular subregion. This, at the end, provides a better discrimination of a broken-rotor-bar fault from the healthy state of the motor.

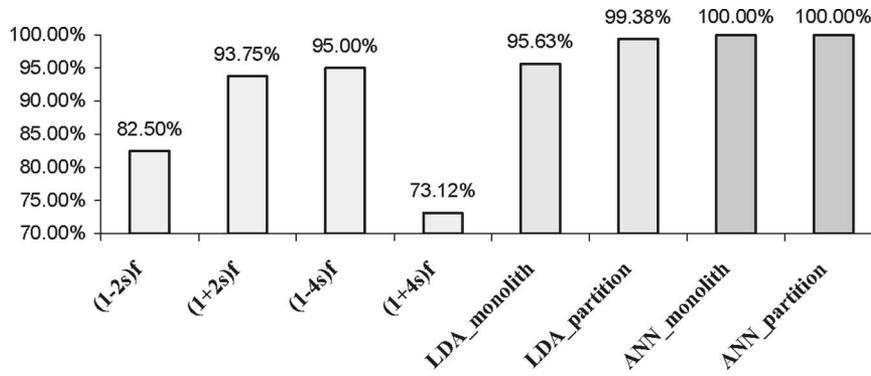


Fig. 9. CDRs with a single signature processing and multiple signature processing under Case 1.

TABLE VI
CDRs OF LDA AND ANN WITH MONOLITH AND PARTITION SCHEMES UNDER CASE 2

	Correct Detection Rate (CDR)	
	Monolith Scheme	Partition Scheme
CDR (LDA - Training)	76/80=95.00 %	80/80=100.0 %
Type I Error (LDA - Training)	1/80=1.25%	0/80=0.0%
Type II Error (LDA - Training)	3/80=3.75%	0/80=0.0%
CDR (LDA - Test)	76/80=95.00 %	78/80=97.50%
Type I Error (LDA - Test)	0/80=0.0%	1/80=1.25%
Type II Error (LDA - Test)	4/80=5.0%	1/80=1.25%
CDR (LDA - Training + Test)	152/160=95.00%	158/160=98.75%
Type I Error (LDA - Training + Test)	1/160=0.63%	1/160=0.63%
Type II Error (LDA - Training + Test)	7/160=4.37%	1/160=0.63%
CDR (ANN, Training)	80/80=100.00%	80/80=100.0 %
Type I Error (ANN - Training)	0/80=0.0%	0/80=0.0%
Type II Error (ANN - Training)	0/80=0.0%	0/80=0.0%
CDR (ANN, Test)	78/80=97.50%	80/80=100.0%
Type I Error (ANN - Test)	2/80=2.5%	0/80=0.0%
Type II Error (ANN - Test)	0/80=0.0%	0/80=0.0%
CDR (ANN, Training + Test)	158/160=98.75%	160/160=100.0%
Type I Error (ANN - Training + Test)	2/160=1.25%	0/160=0.0%
Type II Error (ANN - Training + Test)	0/160=0.0%	0/160=0.0%

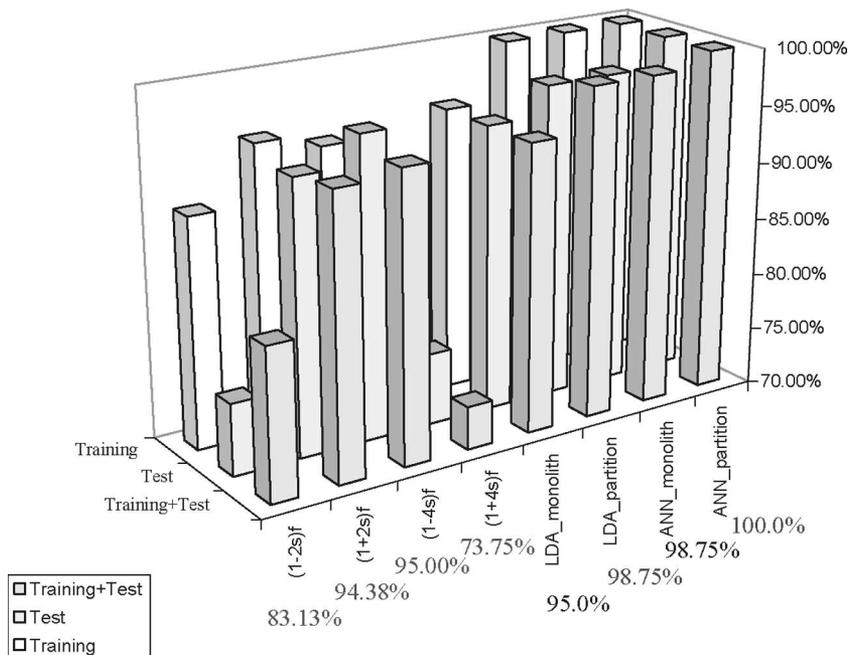


Fig. 10. CDRs with a single signature processing and multiple signature processing under Case 2.

REFERENCES

- [1] T. W. Wan and H. Hong, "An on-line neurofuzzy approach for detecting faults in induction motors," in *Proc. IEMDC*, Cambridge, MA, Jun. 17–20, 2001, pp. 878–883.
- [2] M. E. H. Benbouzid and G. B. Kliman, "What stator current processing based technique to use for induction motor rotor faults diagnosis?," *IEEE Trans. Energy Convers.*, vol. 18, no. 2, pp. 238–244, Jun. 2003.
- [3] O. V. Thursen *et al.*, "A survey of faults on induction motors in offshore oil industry, petrochemical industry, gas terminals, and oil refineries," *IEEE Trans. Ind. Appl.*, vol. 31, no. 5, pp. 1186–1196, Sep./Oct. 1995.
- [4] P. V. Goode and M.-Y. Chow, "Using a neural/fuzzy system to extract heuristic knowledge of incipient faults in induction motors," *IEEE Trans. Ind. Electron.*, vol. 42, no. 2, pp. 139–146, Apr. 1995.
- [5] B. Li, M.-Y. Chow, Y. Tipsuwan, and J. C. Hung, "Neural-network-based motor rolling bearing fault diagnosis," *IEEE Trans. Ind. Electron.*, vol. 47, no. 5, pp. 1060–1069, Oct. 2000.
- [6] S. Altug, M.-Y. Chow, and H. J. Trussell, "Fuzzy inference systems implemented on neural architectures for motor fault detection and diagnosis," *IEEE Trans. Ind. Electron.*, vol. 46, no. 6, pp. 1069–1079, Dec. 1999.
- [7] S. Altug and M.-Y. Chow, "Comparative analysis of fuzzy inference systems implemented on neural structures," in *Proc. Int. Conf. Neural Netw.*, 1997, vol. 1, pp. 426–431.
- [8] M. Haji and H. A. Toliyat, "Pattern recognition—A technique for induction machines rotor broken bar detection," *IEEE Trans. Energy Convers.*, vol. 16, no. 4, pp. 312–317, Dec. 2001.
- [9] S. Nandi and H. A. Toliyat, "Condition monitoring and fault diagnosis of electrical machines," in *Proc. 34th IEEE-IAS Annu. Meeting, Ind.*, vol. 1, 1999, pp. 197–204.
- [10] K. Abbaszadeh, J. Milimonfared, M. Haji, and H. Toliyat, "Broken bar detection in induction motor via wavelet transformation," in *Proc. IEEE IECON*, 2001, vol. 1, pp. 95–99.
- [11] G. B. Kliman and J. Stein, "Methods of motor current signature analysis," *Electr. Mach. Power Syst.*, vol. 20, no. 5, pp. 463–474, Sep. 1992.
- [12] T. W. S. Chow *et al.*, "Three phase induction machines asymmetrical faults identification using bispectrum," *IEEE Trans. Energy Convers.*, vol. 10, no. 4, pp. 688–693, Dec. 1995.
- [13] T. W. S. Chow and H.-Z. Tan, "HOS-based nonparametric and parametric methodologies for machine fault detection," *IEEE Trans. Ind. Electron.*, vol. 47, no. 5, pp. 1051–1059, Oct. 2000.
- [14] B. Ayhan, M.-Y. Chow, H. J. Trussell, and M.-Y. Song, "Comparison of non-parametric spectrum methods for broken rotor bar fault detection," in *Proc. IEEE IECON*, Roanoke, VA, Nov. 2–6, 2003, pp. 2835–2840.
- [15] A. Bellini, F. Filippetti, G. Franceschini, C. Tassoni, and G. B. Kliman, "Quantitative evaluation of induction motor broken bars by means of electrical signature analysis," *IEEE Trans. Ind. Appl.*, vol. 37, no. 5, pp. 1248–1255, Sep./Oct. 2001.
- [16] R. Schoen and T. G. Habetler, "Effects of time varying loads on rotor fault detection in induction machines," *IEEE Trans. Ind. Appl.*, vol. 31, no. 4, pp. 900–906, Jul./Aug. 1995.
- [17] G. B. Kliman *et al.*, "Non-invasive detection of broken rotor bars in operating induction motors," *IEEE Trans. Energy Convers.*, vol. 3, no. 4, pp. 873–879, Dec. 1988.
- [18] W. T. Thomson and I. D. Stewart, "On-line current monitoring for fault diagnosis in inverter fed induction motors," in *Proc. IEE 3rd Int. Conf. Power Electron. and Drives*, London, U.K., 1988, pp. 432–435.
- [19] F. Filippetti *et al.*, "AI techniques in induction machines diagnosis including the speed ripple effect," in *Proc. IEEE-IAS Annu. Meeting*, San Diego, CA, Oct. 6–10, 1996, pp. 655–662.
- [20] N. M. Elkasaby, A. R. Eastham, and G. E. Dawson, "Detection of broken bars in the cage rotor on an induction machine," *IEEE Trans. Ind. Appl.*, vol. IA-22, no. 1, pp. 165–171, Jan./Feb. 1992.
- [21] S. Lawrence Marple, *Digital Spectral Analysis With Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1987.
- [22] L. Lebart, A. Morineau, and K. M. Warwick, *Multivariate Descriptive Statistical Analysis*. New York: Wiley, 1984.
- [23] G. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition*. New York: Wiley, 1992.
- [24] J. Ma *et al.*, "Nonlinear multiclass discriminant analysis," *IEEE Signal Process. Lett.*, vol. 10, no. 7, pp. 196–199, Jul. 2003.
- [25] M.-Y. Chow, R. N. Sharpe, and J. C. Hung, "On the application and design of artificial neural networks for motor fault detection," *IEEE Trans. Ind. Electron.*, vol. 40, no. 2, pp. 181–188, Apr. 1993.
- [26] M. T. Hagan and M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," *IEEE Trans. Neural Netw.*, vol. 5, no. 6, pp. 989–993, Nov. 1994.
- [27] L. S. H. Ngia and J. Sjoberg, "Efficient training of neural nets for nonlinear adaptive filtering using a recursive Levenberg-Marquardt algorithm," *IEEE Trans. Signal Process.*, vol. 48, no. 7, pp. 1915–1927, Jul. 2000.
- [28] J. Sjoberg and M. Viberg, "Separable nonlinear least-squares minimization—Possible improvements for neural net fitting," in *Proc. IEEE Workshop Neural Netw. Signal Process.*, 1997, pp. 345–354.
- [29] S. Amari *et al.*, "Asymptotic statistical theory of overtraining and cross-validation," *IEEE Trans. Neural Netw.*, vol. 8, no. 5, pp. 985–996, Sep. 1997.
- [30] D. C. Montgomery, G. C. Runger, and N. F. Hubele, *Engineering Statistics*. New York: Wiley, 1999.



Bulent Ayhan (S'01) received the B.S. and M.S. degrees in electrical and electronics engineering from Bogazici University, Istanbul, Turkey, in 1998 and 2000, respectively. He is currently working toward the Ph.D. degree at North Carolina State University, Raleigh.

He is a Research Assistant in the Advanced Diagnosis Automation and Control (ADAC) Laboratory, North Carolina State University. His research interests include intelligent condition monitoring of rotating machinery systems.



Mo-Yuen Chow (S'81–M'82–SM'93) received the B.S. degree in electrical and computer engineering from the University of Wisconsin, Madison, in 1982, and the M.Eng. and Ph.D. degrees from Cornell University, Ithaca, NY, in 1983 and 1987, respectively.

Upon completion of the Ph.D. degree, he joined the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, where he has held the rank of Professor since 1999. His core technology is diagnosis and control, artificial neural networks (ANNs), and fuzzy logic with applications to areas including motors, process control, power systems, and communication systems. He established the Advanced Diagnosis Automation and Control (ADAC) Laboratory at North Carolina State University.



Myung-Hyun Song (S'91–M'95) was born in 1953 in Jeju, Korea. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea University, Seoul, Korea.

From 1977 to 1981, he was involved in design and application of electric motors at Samsung Electronic Company, Suwon, Korea. Since 1988, he has been a Professor in the School of Information and Communication, Suncheon National University, Suncheon, Korea. His current research interests include induction-motor fault detection and diagnosis, factory automation, and web-based control.