Using a Neural/Fuzzy System to Extract Heuristic Knowledge of Incipient Faults in Induction Motors: Part I—Methodology

Paul V. Goode, *Student Member, IEEE*, and Mo-yuen Chow, *Senior Member, IEEE*

***Abstract*— The use of electric motors in industry is extensive. These motors are exposed to a wide variety of environments and conditions which age the motor and make it subject to incipient faults. These incipient faults, if left undetected, contribute to the degradation and eventual failure of the motors. With proper monitoring and fault detection schemes, the incipient faults can be detected; thus, maintenance and down time expenses can be reduced while also improving safety. Unfortunately, many of the conventional methods used to determine these faults are either very expensive to implement, performed off-line, require the need of an expert, or impractical for small machines. Artificial neural networks have been proposed and have demonstrated the capability of solving the motor monitoring and fault detection problem using an inexpensive, reliable, and noninvasive procedure [3]–[6]. However, the major drawback of conventional artificial neural network fault detection is the inherent *black box* approach that can provide the correct solution, but does not provide heuristic interpretation of the solution. Engineers prefer the accurate fault detection as well as the heuristic knowledge behind the fault detection process. Fuzzy logic is a technology that can easily provide heuristic reasoning while being difficult to provide exact solutions. Part I of this work introduces the methodology behind a novel hybrid neural/fuzzy system which merges the neural network and fuzzy logic technologies to solve fault detection problems. Part I will also discuss a training procedure for this neural/fuzzy fault detection system. This procedure will be used to determine the correct solutions while providing qualitative, heuristic knowledge about the solutions.**

I. Introduction

The use of electric motors in industry is extensive. These motors are exposed to a wide variety of environments and conditions. These factors, coupled with the natural aging process of any machine, make the motor subject to incipient faults [1]–[4], [25], [27]. These incipient faults, if left undetected, contribute to the degradation and eventual failure of the motors. With proper monitoring and fault detection schemes, the incipient faults can be detected in their early stages; thus, maintenance and down time expenses can be reduced while also improving safety [26].

Bearing wear is one of the most common faults in electric motors. As the motor ages, the bearing will change shape due to an imbalance of the motor or just fragmentation of the bearing itself. These deformations cause vibrations in the motor as the rotor turns. These vibrations can contribute to failures in other parts of the motor. Frequency analysis on the vibration signals caused by the aging bearing can be done to monitor the bearing condition [19]. Frequency analysis of motor bearing faults is based upon the premise that different types of faults occur in different frequency spectra. While this is true, very accurate sensing equipment is needed to capture the impulses because they occur for very short durations over a broad frequency spectrum. This high sensitivity incorporates noise into the procedure from various other electrical and mechanical vibrations. Because of the noise and the high cost of accurate sensing devices, this procedure is usually considered useful for large machines only (greater than 100 hp).

A less expensive means of determining bearing wear is by direct inspection. This method is much more accurate in determining the condition of the bearing; however, it requires down time for the motor. A similar procedure determines the bearing condition based upon a particle analysis of the oil used to lubricate the motor. This procedure, which requires bringing oil samples to a laboratory for a chemical analysis, would be difficult and impractical to implement on a constant basis. Therefore, it is more suitable for routine maintenance practices because it is not a continuous on-line procedure.

Insulation failure of the motor windings is another common fault in electric motors. The windings of a motor are insulated to prevent shorting of the winding turns. This insulation is categorized into several classes which are rated for different maximum operating temperatures [25]. Often, however, these maximum temperatures are violated due to the operating environment, load requirements, etc. This violation leads to cracks, thinning, and eventual loss of insulation at some points on the stator windings. This effectively decreases the number of stator turns by causing interturn shorting of the windings. These shorts decrease the life of the insulation.

Measuring the resistance of the insulation is a common method used to determine insulation condition. This method requires the application of a dc voltage to the winding while grounding the frame or core [23]. A determination can be made about the insulation condition by calculating the resistance of the insulation based upon the current flowing through it. This method, unfortunately, also requires the motor to be taken off-line. Furthermore, it indicates whether or not the insulation has worn enough to cause a short; it does not know if the insulation is about to fail within a short period of time.

Manuscript received March 5, 1994; revised May 31, 1994. This paper received support from National Science Foundation, Grant no. ECS-8922727.

The authors are with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA.

IEEE Log Number 9408821.

0278–0046/95$04.00 © 1995 IEEE

Surge testing of the windings determines the dielectric strength of the insulation. A large sinusoidal voltage is applied to the windings and a comparison of the patterns from the three phases of a motor winding determines the insulation condition. While this technique can be implemented on-line, it still cannot effectively determine the remaining life of the insulation [28].

The parameter estimation approach [14], [26] can provide information relative to the condition and expected life of the insulation and bearings. This method can also be applied noninvasively on-line and, in some cases, without the need of expensive monitoring equipment. This method relies upon a very accurate mathematical model of the motor. This model is based upon a detailed understanding of the system and how it is effected by the chosen parameters. The parameters can be chosen to reflect the condition of the bearings and insulation. Unfortunately, it is very difficult to obtain such an accurate model of the motor. Furthermore, as the motor ages, the model less represents the motor.

All the aforementioned methods require the services of an *expert* to determine the motor condition from the test results. Furthermore, none are able to inexpensively perform incipient fault detection on-line, and in a noninvasive manner, except for the parameter estimation method. Unfortunately, the need for an accurate mathematical model (expert knowledge) makes this method less attractive. Artificial neural networks (ANN), however, have been proposed and have demonstrated to be an effective alternative for performing motor fault detection [3]–[6] while avoiding the need for a mathematical model. Because the ANN can adapt itself to learn arbitrarily complicated continuous nonlinear functions [9], [12], it can learn the motor incipient fault detection process, which enables it to give the accurate solution to a particular fault problem. In addition, the ANN can perform this function on-line through the use of inexpensive monitoring devices. These devices obtain the necessary measurements in a noninvasive manner. Different advantages of using ANN's instead of other fault detection techniques are discussed in more detail in [7], [13], [15].

While the ANN can correctly monitor the condition of the motor, it cannot provide general heuristic, qualitative information about what contributes to a fault (some of the previously mentioned methods also suffer from this shortcoming); i.e., under what conditions does a fault occur? This inability is due to the inherent "black box" feature of the ANN. Even though the neural network can perform the correct input–output relationship for the given problem; it cannot perform this function in a manner which makes heuristic sense. This shortcoming prevents the ANN from providing expert knowledge about the motor condition in heuristic terms that humans prefer. If these heuristics could be extracted from a trained artificial neural network fault detector, it would not only give us more insight into the fault detection process, but also provide more knowledge about the actual system. This problem can be solved by incorporating the use of fuzzy logic with the ANN structure. It is well known that fuzzy logic has the capability of transforming heuristic and linguistic terms into numerical values for use in complex machine computations via fuzzy rules and membership functions [30]. It can also provide a heuristic output as a result of those same complex computations by quantifying the actual numerical data into heuristic and linguistic terms. For these reasons, fuzzy logic can be used to provide a general heuristic solution to a particular problem with the use of general heuristic knowledge about the problem. Unfortunately, *a priori* knowledge about the system is necessary to develop the fuzzy rules and membership functions.

This paper will present a novel motor fault detection technique that merges these two technologies. From this synergy of technology, a neural/fuzzy motor fault detector will be obtained that will learn the bearing and insulation faults and the conditions under which they occur through an inexpensive and noninvasive procedure. Furthermore, the neural/fuzzy fault detector will not need a mathematical model, or *a priori* knowledge, of the motor, but will still be able to provide qualitative expert knowledge of the motor through valid heuristics and give exact quantitative solutions to detect the motor faults. Therefore, expert knowledge of the motor can be extracted from the neural/fuzzy fault detector that will aid in the understanding of motor bearing and insulation faults as well as the fault detection process. This paper will address two of the common causes of failure mentioned earlier as an illustration of the neural/fuzzy system.

Section II of this paper will provide a brief description of the two common motor faults: bearing wear and insulation wear. Section III will discuss the neural/fuzzy motor fault detector architecture and the training algorithm used to extract the heuristic rules and membership functions. Part II of this paper will provide the illustrations of bearing wear and insulation wear as applications to the neural/fuzzy motor fault detector. These example illustrations will adhere to the architecture and training algorithms of Section II and will be explained in detail.

II. Brief Description of Two Common Motor Faults

Two of the most common incipient faults in motors are bearing wear and winding insulation failure. The occurrence of these two faults in split-phase squirrel cage induction motors will be used to illustrate the neural/fuzzy motor fault detector. To determine the measurements necessary for detecting these faults in a noninvasive manner, the dynamics of the induction motor will be briefly discussed.

The stator and rotor flux linkages of a split-phase induction motor represented as

$$\lambda_a = [\lambda_{as} \quad \lambda_{bs}]^T, \tag{1}$$

$$\lambda_b = [\lambda_{ar} \quad \lambda_{br}]^T, \tag{2}$$

where a and b represent the phases a and b of the motor while s and r denote the stator and rotor, respectively. This notation allows for representation of the split-phase squirrel cage induction motor dynamics by the following state equations:

$$\dot{\lambda}_s = \mathbf{R}_s \mathbf{i}_s - \mathbf{v}_s, \tag{3}$$

$$\dot{\lambda}_r = \mathbf{R}_r \mathbf{i}_r - \mathbf{v}_r. \tag{4}$$

The variables $\mathbf{R}_s$ and $\mathbf{R}_r$ represent the stator and rotor resistances. These are both diagonal matrices with nonnegative elements. The variable $\mathbf{i}_s = [i_{as}\ i_{bs}]^T$ represents the stator winding currents while $\mathbf{i}_r = [i_{ar}\ i_{br}]^T$ represents the rotor winding currents. The stator and rotor winding voltages are denoted by $\mathbf{v}_s = [v_{as}\ v_{bs}]^T$ and $\mathbf{v}_r = [v_{ar}\ v_{br}]^T$, respectively. For steady-state conditions, or small perturbation conditions, the flux linkages can be approximated by a linear relationship with respect to the currents. This relationship can be expressed as

$$\begin{bmatrix} \lambda_s \\ \lambda_r \end{bmatrix} = \begin{bmatrix} \mathbf{L}_s(\theta) & \mathbf{L}_{sr}(\theta) \\ \mathbf{L}_{sr}(\theta) & \mathbf{L}_r(\theta) \end{bmatrix} \begin{bmatrix} \mathbf{i}_s \\ \mathbf{i}_r \end{bmatrix} \tag{5}$$

where $\mathbf{L}_s$ and $\mathbf{L}_r$ represent the stator and rotor inductances while the mutual inductance between the stator and the rotor is $\mathbf{L}_{sr}$. θ indicates the rotor position.

From electromagnetic theory, the flux linkage is a function of the equivalent turns of the winding. The equivalent turns for phases a and b can be represented by $\mathbf{N}_s = [N_{as}\ N_{bs}]^T$ and $\mathbf{N}_r = [N_{ar}\ N_{br}]^T$ for the stator and rotor windings, respectively. As the equivalent number of turns changes, motor parameters such as winding resistance and inductance will also change. Therefore, the transient and steady-state performance of the motor can be expressed in terms of the resistances and inductances of the stator and rotor. With a squirrel cage rotor, $\mathbf{N}_r$ is generally assumed to be constant because of the robustness of the rotor. The deterioration of the stator windings will cause $\mathbf{N}_s$ to change, thus making the stator resistance, stator inductance, and mutual inductance functions of $\mathbf{N}_s$.

The electrical torque for the motor, T_e, is a function of the motor parameters and state variables. This torque can be expressed as

$$T_e = \mathbf{i}_s^T \frac{\partial}{\partial \theta} \mathbf{L}_{sr} \mathbf{i}_r. \tag{6}$$

Therefore, the electrical torque is a function of the equivalent number of stator turns and is expressed as $T_e(\mathbf{N}_s)$. This electrical torque can represent the mechanical dynamic equation of the motor as follows:

$$T_e(\mathbf{N}_s) = j\dot{\omega} + B\omega + T_l, \tag{7}$$

where J represents the inertia of the rotor and the connected load and T_l is the load torque which is assumed to be known and constant. B represents the damping coefficient of the motor.

The split-phase induction motor is started with the stator windings of both phases energized. Once the motor has reached 60%–80% synchronous speed, the auxiliary winding, b, is disconnected and no longer used. Therefore, in steady-state operation, only the main winding, a, is used. Because N_{bs} makes no contribution to the detection of faults which occur in steady-state, N_{as} can be represented as N for ease of notation. Likewise, I can be used to represent the steady-state root mean square value of the stator current, i_{as}. The average steady-state rotor speed is expressed as ω. By combining and manipulating (3) through (7) with N and B as variables, the steady-state current I and rotor speed ω can be represented by

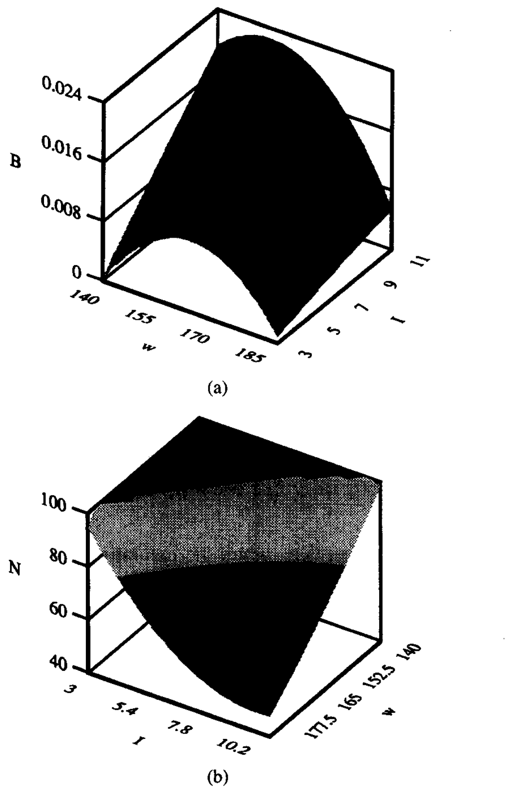


Fig. 1. Surface plot of bearing wear and (a) insulation condition (b) as a function of motor current and rotor speed.

a system of nonlinear algebraic equations of the form

$$\mathbf{f} = [f_1\ f_2]^T. \tag{8}$$

These nonlinear algebraic equations are functions of the main winding equivalent turns, N, and the damping coefficient, B

$$\mathbf{f}(I, \omega, B, N) = \mathbf{0}. \tag{9}$$

Equation (9) suggests that the condition of the bearing and winding insulation can be obtained from the stator current and rotor speed. Because of this, and their easy accessibility, stator current and rotor speed are used as noninvasive inputs for the neural/fuzzy motor fault detector. The relationship between the four variables of (9) can be conceptually viewed as shown below in Fig. 1(a) and 1(b).

Fig. 1(a) and (b) illustrate the bearing condition and insulation condition as a function of the motor current and rotor speed. The shaded regions are rough heuristic estimates of *good*, *fair*, and *bad* conditions of the motor bearing and insulation windings, respectively. In Fig. 1, the darker regions represent a *bad* condition and the lighter regions represent a *good* condition. For example, when a bearing is *bad*, it will impede the turning of the rotor. This will cause a decrease in rotor speed while increasing the amount of input current to the motor. This condition is reflected in the darker region of Fig. 1(a), where a high value of input motor current and a low value of rotor speed occur when the bearing coefficient is

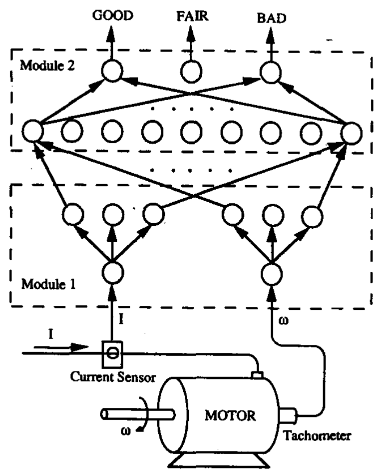


Fig. 2. Neural/fuzzy motor fault detection system architecture.

large in value. Likewise, when the insulation condition is *bad*, the effective number of stator turns will decrease, causing an increase in the input motor current. Therefore, a *bad* winding condition will occur when the input current to the motor is high. This condition is reflected in the darker region of Fig. 1(b), where a high value of current and a high value of speed occur when the number of stator turns is low in value. In actual practice, however, these regions are determined through the knowledge of experts in this field. Simulated motor data, rather than experimentally obtained motor data, is used in this paper because it provides a more controlled set of data by which to demonstrate the technique proposed in this paper.

III. Neural/Fuzzy System for Motor Fault Detection

The neural/fuzzy architecture takes into account both fuzzy logic and neural network technologies. The system is a neural network structured upon fuzzy logic principles, which enables the neural/fuzzy system to provide qualitative descriptions about the motor condition and the fault detection process. This knowledge is provided by the fuzzy parameters of membership functions and fuzzy rules. This is done by constructing the fault detector using two modules: The fuzzy membership function module (module 1) and the fuzzy rule module (module 2). The neural/fuzzy motor fault detector is shown in Fig. 2. The use of these modules for the motor fault detection problem is discussed below.

A. Module 1—The Fuzzy Membership Function Module

The purpose of the fuzzy membership function module is to provide the fuzzy membership functions of the motor current and rotor speed. These membership functions will provide qualitative heuristic knowledge of the motor current and rotor speed. This knowledge will be in the form of grades of membership [31] that indicate, for example, what range of current is considered *low current* and what range of speed is considered *high speed*, etc. For example, when looking at the *high* current membership function, a value of 1.0 pu (per-unit) is considered *very high*, 0.8 pu would be considered *reasonably high*, and 0.4 pu would be considered *not high.* From these linguistic terms, fuzzy rules can be expressed that give qualitative descriptions of the motor; i.e., "when the current is *low* and the speed is *high*, then the bearing condition is *good*." Later sections will explain how to apply the membership function concepts to the motor fault detection problem.

The fuzzy membership function module is composed of two independent *subnetworks*. One of these networks takes normalized motor current, I, as an input while the other takes normalized motor speed, ω, as an input. This measurement is achieved by noninvasive devices (current sensor and tachometer). The function of the subnetwork is to partition the normalized values into fuzzy membership function space and provide these as outputs of the module. The information for the fuzzy membership functions is contained in the weights of the subnetworks, which determine the shape of the membership functions of interest. Subnetworks are used because they allow for representation of very complex membership functions [18] which are more flexible to adaptation for decision classification.

The fuzzy membership functions of motor current and rotor speed do not need to be known because the neural/fuzzy system will adaptively determine these membership functions. However, a good initialization of these subnetworks will aid in training of the neural/fuzzy system by giving it a better starting point. This starting point is important because most of the changes made during training will occur in the fuzzy rule module (module 2). This is evident by the generalized delta rule backpropagation training algorithm. For more information on this algorithm, the reader should refer to [22]. Therefore, a good partitioning of the fuzzy sets will aid in the learning of the fuzzy rules done by module 2 (this will be explained later).

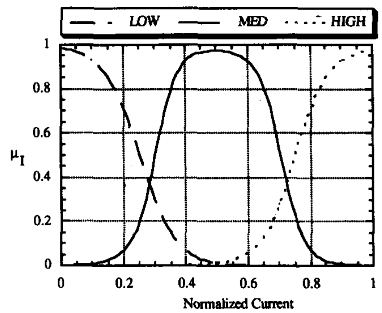
Different initial membership functions were evaluated to determine what constituted a "good" initialization of these subnetworks. The final form of the membership functions was almost identical when initial membership functions of a triangular or trapezoidal shape with a 20%–45% overlap were used. This initialization of the membership functions is in accordance with [17], i.e., trapezoidal membership functions with approximately 25% overlap. Therefore, each subnetwork is initialized to the vague heuristics of *low, medium*, and *high* values of the respective input using trapezoidal membership functions with approximately 40% overlap. An example of initialized subnetworks for motor current and rotor speed is shown in Fig. 3.

These vague heuristics serve to build the universes of discourse, X and Y, for each of the input spaces of I and ω, respectively. These are represented in the standard notation used in [16]

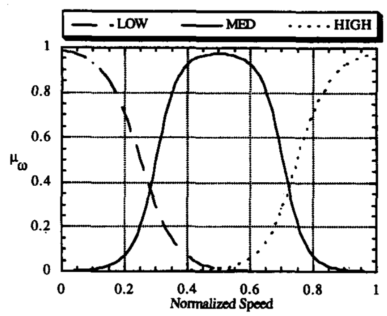
$$X = [\mu_{\text{low}}(I), \mu_{\text{medium}}(I),\ \mu_{\text{high}}(I)],\ I \in X, \quad (10)$$

$$Y = [\mu_{\text{low}}(\omega), \mu_{\text{medium}}(\omega),\ \mu_{\text{high}}(\omega)],\ \omega \in Y, \quad (11)$$

where $\mu_i(\lambda)$ = the grade of membership of λ in $\mathcal{I} = \{i \in \mathcal{I} \mid$ low, medium, high$\}$, $\lambda \in X, Y$. The universes of discourse represent the operating range of the inputs. Normalized data



(a)



(b)

Fig. 3. Example of initial membership functions: (a) current and (b) speed.

is a mapping of the operating range to the range [0, 1]. For the motor fault detection problem, the motor current operating range was from 3 to 14 A, and the motor speed operating range was from 140 to 188.5 rad/s. Both sets of these values were normalized between [0, 1].

As mentioned above, the actual fuzzy membership function information is contained in the weights of the subnetwork in the fuzzy membership function module (module 1) after training. These final fuzzy membership function values are extracted by looking at the outputs of the subnetworks. Each output node represents a fuzzy membership set. For example, the subnetworks of Fig. 3 would each have three output nodes, one each for: *low*, *medium*, and *high*. After training, they are evaluated by inputting a set of incremented values between [0, 1] and recording the outputs. The outputs will represent the final form of the fuzzy membership functions.

B. Module 2—The Fuzzy Rule Module

The fuzzy rule module provides the antecedent-consequence statements of fuzzy logic. These statements provide the condition of the fault being monitored given the linguistic operating range of the inputs. For example, "if the current is *low* and the speed is *high*, then the bearing condition is *good*." The fuzzy rule represents a combination of the qualitative heuristic knowledge of the operating systems and the quantitative descriptions of the motor conditions. The antecedents are the second half (the membership functions being the first half) of the qualitative heuristics by telling us what types of conditions can exist. The consequence provide the quantitative information about the condition of the motor by using the descriptions of *good*, *fair*, and *bad*.

The structure of the module is a two-layer feedforward ANN. The nodes of the input layer of this module are antecedent nodes which represent the conditional part of the antecedent-consequence rules of fuzzy logic. These conditional statements are based upon combinations of the fuzzy membership functions. For example, one node would represent the conditional statement "if the current is *low* and the speed is *high*." The nodes of the output layer of this module are consequence nodes which represent the consequence part of the antecedent-consequence rules. These are in the form of *good*, *fair*, and *bad*.

A starting point for the fuzzy rules is predetermined through whatever minimal knowledge is available. This minimal knowledge is merely a "best guess" of what the rules might actually be. This initialization, as with the membership function initialization, gives the network a better starting point for learning the actual fuzzy rules. The final, correct rules are determined through training of the neural/fuzzy system.

As with the fuzzy membership function module, the information for the antecedent-consequence rules is contained in the weights connecting the antecedent nodes to the consequence nodes. The *sign* of the rule weights indicates the consequence for each antecedent statement. If the sign of a weight connecting an antecedent node, j, to a consequence node, k, is positive, then k is the consequence for antecedent j. Likewise, if the sign of a weight connecting an antecedent node, j, to a consequence node, k, is negative, then k is not the consequence for antecedent j. This can be verified mathematically as follows:

The generalized delta rule is used as the learning algorithm to backpropagate the error through the network. Through this method, the weights are changed to minimize a measure of the network's error. The change in weights between an antecedent node, j, and a consequence node, k, is Δw_{kj} and is calculated each iteration by

$$\Delta w_{kj} = \eta(t_k - o_k)[o_k - (o_k)^2]o_j, \tag{12}$$

where η is the learning rate, t_k is the target output of a consequence node for a given pattern, o_k is the actual output of a consequence node for a given pattern, and o_j is the output of the antecedent node for a given training pattern. Because a sigmoidal activation function is used, the outputs of the antecedent and consequence nodes are bounded between zero and one. Therefore

$$o_j,\ o_k,\ t_k \in [0,1]. \tag{13}$$

Because t_k is either $\{1\}$ (represents a classification) or $\{0\}$ (represents not a classification), then

$$(t_k - o_k) \text{ is } \begin{cases} \leq 0 & \text{if } t_k = 0 \\ \geq 0 & \text{if } t_k = 1. \end{cases} \tag{14}$$

Given the facts of: $\eta > 0$, $[o_k - (o_k)^2]o_j > 0$, and (14), then

$$\Delta w_{kj} \text{ is } \begin{cases} \leq 0 & \text{if } t_k = 0 \\ \geq 0 & \text{if } t_k = 1. \end{cases} \tag{15}$$

Therefore, the weights which connect the correct consequence for an antecedent will continue to increase positively in

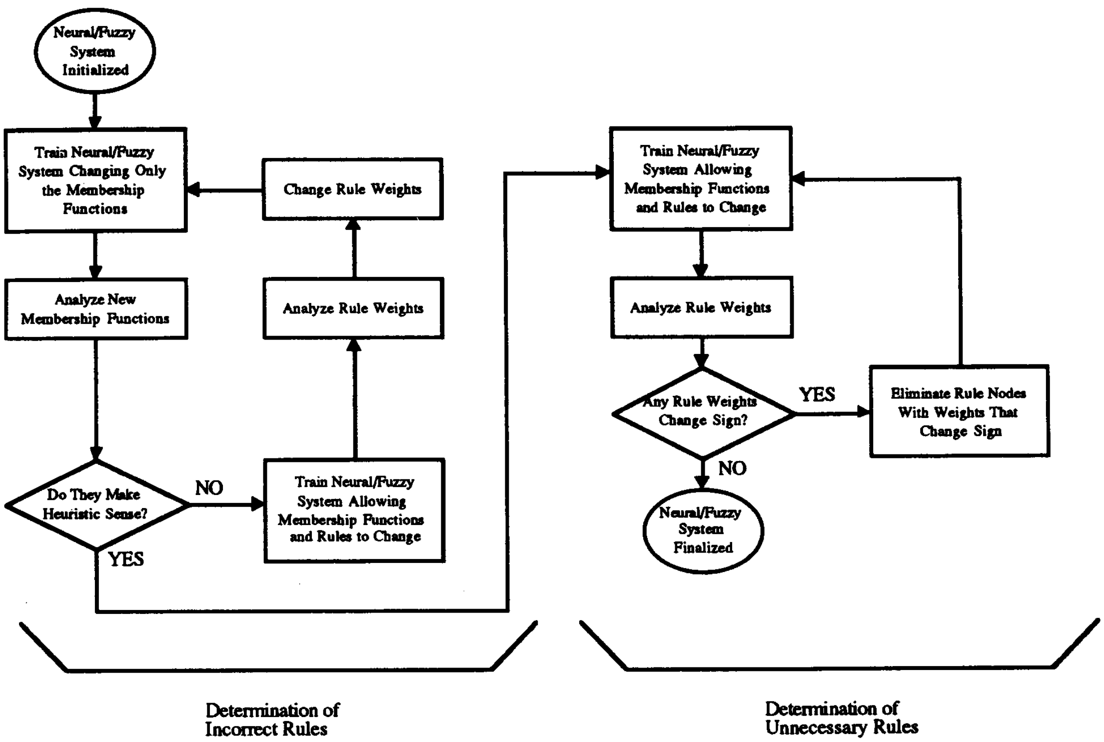


Fig. 4. Flowchart for training of neural/fuzzy motor fault detector.

value whereas the weights which connect the incorrect consequence for an antecedent will continue to decrease negatively in value. After training, the fuzzy rule for an antecedent can be determined by looking at which of the weights emanating from it are positive. The consequence node to which the positive weight connects is the correct consequence for that antecedent. There should exist only one such weight. For example, the antecedent node for "if the current is *low* and the speed is *high*" should have three weights emanating from it. If the weight to the consequence node *good* is positive and the other two weights (to *fair* and *bad*) are negative, then the fuzzy rule would be "if the current is *low* and the speed is *high* then the bearing condition is *good.*"

C. Training Procedure

The flowchart in Fig. 4 is used as a guideline for the training of the neural/fuzzy motor fault detector.

The set of weights, $\mathbf{w}_\mu$, contains the knowledge of the fuzzy membership functions while the set of weights, $\mathbf{w}_R$, contains the knowledge of the fuzzy rules. Prior to initialization, $\mathbf{w}_\mu$ and $\mathbf{w}_R$ are expressed as $\mathbf{w}_\mu^0$ and $\mathbf{w}_R^0$, respectively, where $\mathbf{w}_\mu^0$ and $\mathbf{w}_R^0$ are

$$\mathbf{w}_\mu^0 \in [-0.5, 0.5],\ \forall w_\mu^0 \in \mathcal{W},$$
$$\text{and} \qquad \mathbf{w}_R^0 \in [-0.5, 0.5],\ \forall w_R^0 \in \mathcal{W}, \tag{16}$$

where $\mathcal{W}$ is the overall weight space. After the initialization process, as discussed earlier in this section, we can define these initialized weights as

$$\mathbf{w}_\mu = \mathbf{w}_\mu^I, \quad \forall w_\mu \in \mathcal{W} \text{ and } \mathbf{w}_R = \mathbf{w}_R^I, \quad \forall w_R \in \mathcal{W}, \tag{17}$$

where $\mathbf{w}_\mu^I$ and $\mathbf{w}_R^I$ give the initial fuzzy membership functions and fuzzy rules. While $\mathbf{w}_\mu^I$ and $\mathbf{w}_R^I$ may not be the correct values to obtain the actual solution, they provide a good, general, initial condition for the convergence to the optimal solution.

The neural/fuzzy system is initialized with a set of weights, $\mathbf{w}^I$, where $\mathbf{w}^I$ is expressed as

$$\mathbf{w}^I = \mathbf{w}_\mu^I \cup \mathbf{w}_R^I,\ \forall w^I \in \mathcal{W}. \tag{18}$$

Because module 2 trains faster than module 1 (as mentioned earlier), it is recommended to train one module at a time; more specifically, train module 1 while not training module 2. This will allow for evaluation of the initialization of module 2 (was it correct or not?). Once a correct initialization for module 2 is obtained, denoted $\mathbf{w}_R^L$, both modules can be trained simultaneously using an initial condition of $(\mathbf{w}_\mu^I, \mathbf{w}_R^L)$. This is possible because $\mathbf{w}_R^L$ will contribute less error to the output.

Therefore, during the first phase of training, the neural/fuzzy system *assumes* that the initial fuzzy rules are correct. These assumptions imply that

$$\mathbf{w}_R^I = \mathbf{w}_R^*, \tag{19}$$

where $\mathbf{w}_R^*$ is the optimal set of weights which represent the fuzzy rules. With the assumption that (19) is correct, then $\mathbf{w}_\mu^*$ will be obtained when holding $\mathbf{w}_R^*$ constant. If, however, (19) is not true, then $\mathbf{w}_\mu^*$ will not be obtained. As a result, the weights of module 1, $\mathbf{w}_\mu^I$ will incorrectly change during the learning of the membership functions to accommodate the incorrect initial rule base of $\mathbf{w}_R^I$. The resulting incorrect membership functions will generally violate standard fuzzy logic principles; i.e., do the resulting membership functions make heuristic sense with our physical understanding of the system being modeled? These violations will generally be obvious, even with a limited understanding of the system being modeled.

At this point, the weights of module 1 are then reinitialized to $\mathbf{w}_\mu^I$, as before, and the neural/fuzzy system is trained again, but this time allowing $\mathbf{w}_R^I$ to update. The learning algorithm used requires this reinitialization. The network is trained using the backpropagation learning algorithm. This algorithm is basically an optimization routine, modifying the neural/fuzzy system's internal parameters (its weights) to obtain a desired result. This procedure minimizes an error function, $E(\mathbf{w})$, which is defined in the weight space, $\mathcal{W}$. $\mathcal{W}$ is comprised of two sets of weights, $\mathbf{w}_\mu^I$ and $\mathbf{w}_R^I$, where $\mathbf{w}_\mu^I$ is m-dimensional, $\mathcal{W}$ is n-dimensional, and W_μ^I is $(m+n)$-dimensional.

As mentioned earlier, during the first phase of training, $\mathbf{w}_R^I$ is not allowed to change. Although $E(\mathbf{w})$ is still an $(m+n)$-dimensional function during this phase, its minimization occurs only in the m-dimensional space. If (19) is true, this is not a problem, as $\mathbf{w}_\mu^I$ need only operate in m-dimensional space to converge to $\mathbf{w}_\mu^*$. However, if (19) is not true, then the value for $\mathbf{w}_\mu^I$ obtained after training, denoted $\mathbf{w}_\mu^L$, (which served to minimize $E(\mathbf{w})$), is not W_μ^*. While $\mathbf{w}_\mu^L$ may contain *some* learned knowledge, it may also contain *false* knowledge because it was learning using false information, given that (19) is not true. Therefore, this result of $\mathbf{w}_\mu^I$ is, at least in part, erroneous and must be discarded.

If $\mathbf{w}_\mu^L$ is not discarded, but trained further while also allowing $\mathbf{w}_R^I$ to change, it may approach $\mathbf{w}_\mu^*$ as $\mathbf{w}_R^I \to \mathbf{w}_R^*$. However, this possibility is not guaranteed as it is unknown where this new initial condition, $(\mathbf{w}_\mu^L, \mathbf{w}_R^I)$ is on the $(m+n)$-dimensional error surface generated by $E(\mathbf{w})$. Further training from this initial condition may result in an incorrect local minimum which may be avoided by the more general initial condition of $(\mathbf{w}_\mu^I, \mathbf{w}_R^I)$.

Therefore, the weights of module 1 are then reinitialized to $\mathbf{w}_\mu^I$, as before, and the neural/fuzzy system is trained again, but this time allowing $\mathbf{w}_R^I$ to update. The change in $\mathbf{w}_R^I$ after training, denoted $\mathbf{w}_R^L$, can be examined to determine which rule is incorrect. Once the rule is corrected and both modules are initialized with $(\mathbf{w}_\mu^I, \mathbf{w}_R^L)$, the system is trained with $\mathbf{w}_R^L$ constant. This procedure is repeated until membership functions that conform to standard fuzzy logic reasoning have been obtained.

After determination and revision of the incorrect rules, the unnecessary rules are addressed. Because the antecedent layer of module 2 represents all possible combinations between the input fuzzy sets, some combinations may not be necessary for proper classification. If this can be determined, the antecedent nodes corresponding to these combinations can be removed from the module, thus reducing complexity.

The unnecessary antecedent nodes (rules) are determined by training the neural/fuzzy system while allowing $\mathbf{w}_R^L$ to change. Prior to this training, it is necessary to once again initialize the weights of the membership function module to $\mathbf{w}_\mu^I$ while keeping the weights of $\mathbf{w}_R^I$ which were just acquired. This re-initialization is necessary because $\mathbf{w}_\mu^I$ changed (in the previous phase of training discussed above) to accommodate the rule base of module 2 $(\mathbf{w}_R^L)$ which was not allowed to change. If there were any unnecessary rules present in the rule base of module 2, then $\mathbf{w}_R^L \neq \mathbf{w}_R^*$, while the errors introduced by these were backpropagated through to module 1. Although these errors are usually not large enough to affect convergence to the training criterion, they are large enough to slightly distort the shape of the membership functions. Therefore, if $\mathbf{w}_\mu^I$ is not reinitialized, $\mathbf{w}_R^L$ may not change enough to aid in the determination of unnecessary rules.

After training the neural/fuzzy motor fault detector with both $\mathbf{w}_\mu^I$ and $\mathbf{w}_R^L$ changing, the trained weights of $\mathbf{w}_R^L$, denoted $\mathbf{w}_R^{L2}$, are examined to determine which nodes are candidates for extraction. Candidate nodes are those which have weights that have changed sign during training; i.e., the weight value is positive before training and negative after, or vice versa, as discussed in (15). These antecedent nodes are extracted, thus removing those rules from the rule base. This procedure is repeated until all weight values increase in absolute value. In Part II of this paper, we will demonstrate the neural/fuzzy system training procedure on the motor incipient fault detection problem.

REFERENCES

[1] D. R. Boothman, E. C. Elgar, R. H. Rehder, and R. J. Woodall, "Thermal tracking—A rational approach to motor protection," *IEEE Trans. Power Apparatus and Syst.*, Sept./Oct. 1974, pp. 1335–1344.

[2] S. Cambrias and S. A. Rittenhouse, "Generic guidelines for life extension of plant electrical equipment," Electric Power Research Institute Rep. EL-5885, July 1988.

[3] M. Chow, "Artificial neural network methodology in real-time incipient fault detection of rotating machines," in *Proc. Nat. Sci. Foundation Workshop on Artificial Neural Network Methodology in Power Syst. Eng.*, Clemson University, Clemson, SC, Apr. 8–10, 1990, pp. 80–85.

[4] M. Chow and S. O. Yee, "Application of neural networks to incipient fault detection in induction motors," *J. Neural Network Computing*, Auerback, vol. 2, no. 3, pp. 26–32, 1991.

[5] ——, "Methodology for on-line incipient fault detection in single-phase squirrel-cage induction motors using artificial neural networks," *IEEE Trans. Energy Conversion*, vol. 6, no. 3, pp. 536–545, Sept. 1991.

[6] M. Chow, P. M. Mangum, and S. O. Yee, "A neural network approach to real-time condition monitoring of induction motors," *IEEE Trans. Ind. Electron.*, vol. 38, no. 6, pp. 448–453, Dec. 1991.

[7] M. 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. 3, pp. 181–196, Aug. 1991.

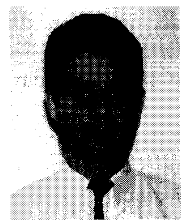
[8] M. Chow and P. Goode, "Adaptation of a neural/fuzzy fault detection system," in *Proc. IEEE Conf. on Decision and Contr.*, 1993.

[9] G. Cybenko, "Approximations by superpositions of a sigmoidal function," *Mathematics of Contr., Signals, and Syst.*, vol. 2, pp. 303–314, 1989.

[10] P. Goode and M. Chow, "Neural/fuzzy systems for fault detection in induction motors," in *Proc. IECON '93*, Honolulu, HI, Nov. 1993.

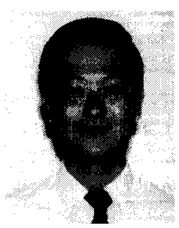
[11] B. S. Guru and H. R. Hiziroglu, *Electric Machinery and Transformers*. San Diego, CA: Harcourt Brace Jovanovich, 1988.

[12] K. Hornik, M. Stinchcombe, and H. White, "Multi-layer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359–366, 1989.
[13] J. C. Hung and B. J. Doran, "High reliability strapdown platforms using two-degree-of-freedom gyros," *IEEE Trans. Aerospace and Electron. Syst.*, vol. 9, no. 2, pp. 253–259, March 1973.
[14] R. Isermann and B. Freyermuth, "Process fault diagnosis based on process model knowledge—Part I: Principles for fault diagnosis with parameter estimation," *Trans. ASME*, vol. 113, pp. 620–626, Dec. 1991.
[15] A. Keyhani and S. M. Miri, "Observers for tracking of synchronous machine parameters and detection of incipient faults," *IEEE Trans. Energy Conversion*, vol. 1, no. 2, June 1986.
[16] G. J. Klir and T. A. Folger, *Fuzzy Sets, Uncertainty and Information.* Englewood Cliffs, NJ: Prentice-Hall, 1988.
[17] B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence.* Englewood Cliffs, NJ: Prentice-Hall, 1992.
[18] C. Lin and C. S. G. Lee, "Neural-network-based fuzzy logic control and decision system," in *Proc. 4th Incipient Failure Detection Conf.*, Philadelphia, PA, Oct. 1990.
[19] D. Morel and P. Neau, "Vibration monitoring of heavy duty machines," in *IEEE Trans. Comput.*, vol. 40, no. 12, pp. 1320–1336, Dec. 1991.
[20] W. Nelson, *Accelerated Testing.* New York: Wiley, 1990.
[21] H. Okada, N. Watanabe, A. Kawamura, and K. Asakawa, "Initializing multilayer neural networks with fuzzy logic," in *Proc. Int. Joint Conf. on Neural Networks*, vol. 1, pp. 239–244, 1992.
[22] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*, D. E. Rumelhart, J. L. McClelland, and the PDP Research Group, Eds. Cambridge, MA: MIT, pp. 318–362, 1986.
[23] D. E. Schump, "Reliability testing of electric motors," *IEEE Trans. Ind. Applicat.*, vol. 25, no. 3, pp. 386–390, May/June 1989.
[24] R. N. Sharpe and M. Chow, "A methodology using fuzzy logic to optimize feedforward artificial neural network configurations," *IEEE Trans. Syst., Man, and Cybern.*, vol. 24, no. 5, 1994.
[25] R. W. Smeaton, *Motor Application and Maintenance Handbook*, 2nd ed. New York: McGraw-Hill, 1987.
[26] A. K. Sood, A. A. Fahs, and N. A. Henein, "Engine fault analysis Part I: Statistical methods," *IEEE Trans. Ind. Electron.*, vol. 32, no. 4, Nov. 1985.
[27] P. J. Tavner and J. Penman, *Condition Monitoring of Electrical Machines.* New York: Research Studies Press Ltd., Wiley.
[28] P. J. Tavner, "Condition monitoring, the way ahead for large electrical machines," *Fourth Int. Conf. on Electric Machines and Drives*, pp. 159–162, Sept. 13–15, 1989.
[29] A. E. Taylor and W. R. Mann, *Advanced Calculus.* New York: Wiley, 1983.
[30] J. E. Timperly, "Incipient fault detection through neutral RF monitoring of large rotating machines," *IEEE Trans. Power App. and Syst.*, vol. 102, no. 3, March 1983.
[31] L. A. Zadeh, "Fuzzy sets," *Inform. and Contr.*, vol. 8, pp. 338–353, 1965.
[32] D. Zorbas, *Electrical Machines.* St. Paul, MN: West, 1989.



Paul V. Goode (S'92) received the B.S. degree in 1992, and the M.S. degree in 1994, in electrical engineering, from North Carolina State University, Raleigh.

He worked as a graduate research assistant while pursuing the M.S. degree, with research focusing on the merging of neural network and fuzzy logic technologies into hybrid neural/fuzzy systems for use in fault detection applications. He then worked for the Intelligent Power Products Group of Harris Corporation, Semiconductor Sector, Research Triangle Park, NC. He is currently pursuing graduate studies in the Department of Electrical and Computer Engineering, North Carolina State University.



Mo-yuen Chow (S'81–M'87–SM'93) received the B.S. degree 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. His graduate studies concentrated on system modeling, estimation techniques, control theory, and power engineering.

In 1987, he joined the faculty of North Carolina State University, Raleigh, where he is presently an Associate Professor in the Department of Electrical and Computer Engineering. He has also worked as a consultant for Taiwan Power Company, J. W. Harley Company, and a faculty intern at Duke Power Company. Since 1987, he has been working as a Principal Investigator in several projects in the areas of system monitoring, fault detection and control, artificial neural networks, and fuzzy logic. He has published over 40 technical papers related to his research work.