CHAPTER 1

INTRODUCTION

The soft computing tools like Neural Network (NN), Fuzzy Logic (FL) and Genetic Algorithm (GA) have been used in many engineering applications such as fault detection, control and optimization of dynamic systems (Narendra and Parthasarathy 1990 and Park et al 1996). The NN can be classified as feed forward NN and recurrent NN according to their structure. It is well known that a feed forward neural network is capable of approximating any continuous function closely.

The use of FL in control applications has tremendously increased over the last decade. Theoretically founded by Zadeh and explored by Mamdani in the early 1970s the wide industrial application of FL began almost a decade later. With the commercial availability of FL control development tools and the better understanding of analysis methods, more and more industrial companies have successfully used this technology for their specific applications (Timothy J. Ross 1997 and Kohonen 1995). The main advantages of fuzzy controllers are simplicity, low cost and possibility to design without knowing the exact mathematical model of the process or system.

GA is simple, powerful, and general – purpose, derivative free, stochastic global optimization method (search algorithm) inspired by the laws of natural selection and genetics. Thus they use the concept of Darwins’s theory of evolution, which is based on the rule of survival of the fittest. The
first GA was developed by Holland in 1975 (David E. Goldberg 1989). They are useful approaches to problems requiring effective and efficient searching, and their use is widespread in applications to business, scientific, and engineering fields. In an optimally designed application, GA’s can be used to obtain an approximate solution for single variable or multivariable optimal problems. Before a GA is applied, the optimization problem should be converted to a suitably described function. The corresponding function is called “fitness function”. It represents a performance of the problem. The higher the fitness value, the better the system’s performance. The objective of a GA is to imitate the genetic operation process, e.g., reproduction, crossover, and mutation, to obtain a solution corresponding to the fitness value.

Electrical machines like induction motors, transformers and generators are essential components in most of the manufacturing and production industries. They are exposed to a variety of environmental conditions. These operating conditions coupled with natural ageing cause incipient faults in the machines.

In induction machines, the most common incipient faults are winding faults and bearing wear. Almost 40% of machine failures occur due to bearing faults and 10% of the induction motor failures are rotor related (Peter Vas 1993 and Chow 1997). Rotor related faults in three phase induction motors are due to broken bars and end rings. The root of the failure is a crack that develops in the rotor bars. The crack may increase its size if left undetected. Broken bars can be a serious problem when induction motors have to perform hard duty cycles. Broken rotors do not initially cause an induction motor to fail, but they can impair motor performance, lead to motor malfunction and cause severe mechanical damage to the stator winding if left undetected.
The main function of power transformer is to supply electrical energy with acceptable degree of reliability and quality. The reliably depends on the trouble free operation of power transformer. Consequently, its preventive maintenance can lead to huge saving of maintenance cost, besides achieving uninterrupted power supply. In the event of local overheating due to winding faults and arc discharges, its insulating material decomposed into gases. The decomposed gases will be dissolved into transformer oil.

With proper monitoring scheme, if the incipient faults are detected at their early stages, the maintenance cost and down time can be reduced (Mishra et al 1996, Schoen 1995, Chen et al 1994 and Chow et al 1993). Many of the conventional fault detection methods require the need of an expert to evaluate the machine condition. The development of soft computing techniques in the area of computer science motivated the researchers to use these techniques for intelligent problem solving, which exhibits the characteristics of human intelligence. This motivated the author to investigate the applicability of soft computing tools for fault detection in electrical machines.

1.1 INTRODUCTION TO NEURAL NETWORKS

1.1.1 Introduction

An artificial neural network is an information processing system that has certain performance characteristics in common with biological NN (Laurene Fausett 1994). An artificial NN has been developed as generalization of mathematical models of human cognition or neural biology, based on assumptions that:
- Information processing occurs at many simple elements called neurons.

- Signals are passed between neurons over connection links.

- Each connection link has an associated weight, which, multiplies the signal transmitted.

- Each neuron applies an activation function (usually nonlinear) to its net input (sum of weighted input signals) to determine its output signal.

An artificial NN is characterized by,

- its pattern of connections between the neurons
- its method of determining the weights on the connections and
- its activation function

NN consists of many nodes, i.e. processing units analogous to neurons in the brain. Each node has a node function, associated with it, which determines the output of the node for the given input. Modifying the local parameters may alter the node function. Artificial neural networks thus are an information processing system. In this information processing system, the elements called neurons, process the information. The signals are transmitted by means of connection links. The links possess an associated weight, which is multiplied along with the incoming signal (net input) for any typical neural net. The output signal is obtained by applying activations to the net input. The neural net can generally be a single layer or a multi-layer net. The network weight is adjusted based on a comparison of the output and the target, until the network output matches the target.
1.1.2 Activation functions

An activation function may be linear or a non-linear function. A particular activation function is chosen to satisfy some specification of a problem that the neuron is attempting to solve. There are three most commonly used activation functions. They are,

(a) Identity activation function
(b) Binary step activation function
(c) Sigmoid activation function

(a) Identity function

Figure 1.1 shows the graphical representation of the Identity activation function. The function is given by,

\[ f(x) = x \quad \text{for all } x. \]  

Figure 1.1 Identity activation function

(b) Binary step activation function

Figure 1.2 shows the graphical representation of the Binary step function. The function is given by,
\[ f(x) = \begin{cases} 1 & \text{if } f(x) \geq \theta \\ 0 & \text{if } f(x) < \theta \end{cases} \] 

where \( \theta \) is the threshold value.

**Figure 1.2** Binary step activation function

(c) **Sigmoid activation function**

Figure 1.3 shows the binary sigmoid activation function. This activation function takes the input and squashes the output into the range 0 to 1, according to expression

\[ f(x) = \frac{1}{1+\exp(-\sigma x)} \] 

where \( \sigma \) is called the steepness parameter.

**Figure 1.3** Binary sigmoid activation function
If \( f(x) \) is differentiated

\[
f'(x) = \sigma f(x)[1-f(x)]
\]

(1.4)

This is also called logistic function

Figure 1.4 shows the bipolar sigmoid activation function. The desired range here is between +1 and -1. This function is related to the hyperbolic tangent function.

![Bipolar sigmoid activation function](image)

Figure 1.4 Bipolar sigmoid activation function

The bipolar sigmoidal function is given as,

\[
b(x) = 2f(x) - 1
\]

(1.5)

\[
b(x) = \frac{(1-\exp(-\sigma x))}{(1+\exp(-\sigma x))}
\]

(1.6)

\[
b^1(x) = \frac{\sigma}{2} \left[ (1+b(x))(1-b(x)) \right]
\]

(1.7)

1.1.3 Learning rules

The weights and biases of the network can be modified by means of ‘learning rule’. This procedure may also be referred to as a training algorithm. The purpose of the learning rule is to train the network to perform some task. Neural networks can be trained to solve problems that are difficult for
conventional computers or human beings. There are many types of neural network learning rules. They fall into three broad categories: supervised learning, unsupervised learning and reinforcement (or graded) learning.

a) Supervised learning: In supervised learning, the network is provided with inputs and the corresponding correct output. As the inputs are applied to the network, the network outputs are compared to the targets. The learning rule is then used to adjust the weights and biases of the network in order to move the network outputs closer to the targets. An example for the supervised learning is the perceptron-learning rule.

b) Reinforcement learning: This is similar to supervised learning, except that, instead of being provided with the correct output for each network input, the algorithm is only given a grade. The grade is a measure of the network performance over some sequence of inputs. This type of learning is currently much less common than supervised learning.

c) Unsupervised learning: In unsupervised learning, the weights and biases are modified in response to network inputs only. There are no target outputs available. The network learns to categorize the input patterns into a finite number of classes. An example for unsupervised learning algorithm is Adaptive Resonance Theory.

1.1.4 Back propagation neural networks

Back Propagation (BP) is a systematic method for training multi-layer artificial neural networks. It has a mathematical foundation that is strong
if not highly practical. It is a multi layer forward network using extended gradient-descent based delta learning rule, commonly known as BP (of errors) rule. BP provides a computationally efficient method for changing the weights in a feed-forward network, with differentiable activation function units, to learn a training set of input-output examples. Hinton, Rumelhart and Williams first introduced BP network in 1986 (Sivanandam et al 2006). Being a gradient-descent method it minimizes the total squared error of the output computed by the net. The network is trained by supervised learning method.

The general architecture of a multilayer BP network is shown in Figure 1.5. The network has three layers namely input layer, hidden layer and output layer. Each neuron in the layer is fully interconnected. The additional input neurons called as bias neurons (whose input is always set to 1) are connected in the hidden and output layers to enhance the training process.

**Figure 1.5 Architecture of two layer BPNN**

An activation function for a BP network should be continuous, differentiable and monotonically non-decreasing. Furthermore, for computational efficiency, it is desirable that its derivative be easy to compute.
For the most commonly used activation functions, the value of the derivative (at a particular value of the independent variable) can be expressed in terms of the value of the function (at the value of the independent variable). Usually, the function is expected to saturate, i.e., approach finite maximum and minimum values asymptotically.

The BP training algorithm is an interactive gradient algorithm designed to minimize the mean square error between the actual output of a feed-forward net and the desired output.

### 1.1.5 Back propagation training algorithm

The nomenclature used in the training algorithm of the BP network is as follows:

- **x**: Input training vector, $x = (x_1, x_2, \ldots, x_i, \ldots, x_n)$
- **t**: Output target vector, $t = (t_1, t_2, \ldots, t_k, \ldots, t_m)$
- **δ_k**: Portion of error correction weight adjustment for hidden layer to output layer
- **δ_j**: Portion of error correction weight adjustment for input layer to hidden layer
- **α_L**: Learning rate
- **μ_i**: Momentum factor
- **X_i**: Input unit
- **Z_j**: Hidden unit
- **o_k**: Output unit
- **v_{oj}**: Bias on the hidden unit j
- **w_{ok}**: Bias on the output unit k
- **v_{ij}**: Weight from $i^{th}$ input unit to $j^{th}$ hidden unit
The training algorithm for a BP network is as follows:

**Step 0**: The weights are initialized randomly between -0.5 to +0.5 and normalized. Initialize the learning parameters.

**Step 1**: While stopping condition is false, do steps 2-9.

**Step 2**: For each training pair, do steps 3-8.

**Feed forward**:

**Step 3**: Each input unit \(X_i, i=1, \ldots, n\) receives input signal \(x_i\) and broadcasts this signal to all units in the layer above (the hidden units).

**Step 4**: Each hidden unit \(Z_j, j = 1, \ldots, p\) sums its weighted input signals,

\[
z_{\text{inj}} = v_{o_j} + \sum_{i=1}^{n} x_i v_{ij},
\]

applies its activation function to compute its output signal,

\[
z_j = f(z_{\text{inj}})
\]

and sends this signal to all units in the above layer (output layer).

**Step 5**: Each output unit \(Y_k, k=1, \ldots, m\) sums its weighted input signals,

\[
y_{\text{ink}} = w_{o_k} + \sum_{j=1}^{p} z_j w_{jk},
\]
applies its activation function to compute its output
signal,
\[ y_k = f(y_{in_k}) \]

Back Propagation of Error:

Step6: Each output unit \((Y_k, k=1, ..., m)\) receives a target
pattern corresponding to the input training pattern,
computes its error information term,
\[ \delta = (t_k - y_k) f'(y_{in_k}), \]
calculates its weight correction term. (Used to update \(w_{jk}\)
later)
\[ \Delta w_{jk} = \alpha \delta z_j \]
calculates its bias correction term. (Used to update \(w_{ok}\)
later)
\[ \Delta w_{ok} = \alpha \delta_k \]
and sends \(\delta_k\) to units in the hidden layer below

Step7: Each hidden unit \((Z_j, j=1, ..., p)\) sums its delta inputs
\[ \delta_{inj} = \sum_{k=1}^{m} \delta_k w_{jk}, \]
multiplies by the derivative of its activation function to
calculate its error information term.
\[ \delta_j = \delta_{inj} f'(z_{inj}), \]
calculates its weight correction term (Used to update \(v_{ij}\)
later)
\[ \Delta v_{ij} = \alpha \delta_j x_i \]
and calculates its bias correction term (Used to update \(v_{oj}\)
later)
\[ \Delta v_{oj} = \alpha \delta_j \]
Updates the weights and biases
Setp 8: Each output unit \((Y_k, k=1, \ldots, m)\)
updates its bias and weights. \((j=0, \ldots, p)\)
\[
\begin{align*}
    w_{ok}^{\text{new}} &= w_{ok}^{\text{old}} + \Delta w_{ok} \\
    w_{jk}^{\text{new}} &= w_{jk}^{\text{old}} + \Delta w_{jk}
\end{align*}
\]
Each hidden unit \((Z_j, j=1, \ldots, p)\) updates its bias and weights.
\((i =0, \ldots, n)\)
\[
\begin{align*}
    v_{oj}^{\text{new}} &= v_{oj}^{\text{old}} + \Delta v_{oj} \\
    v_{ij}^{\text{new}} &= v_{ij}^{\text{old}} + \Delta v_{ij}
\end{align*}
\]

Setp 9: Test for stopping condition.

In the above algorithm the weights are updated only after complete
learning i.e. the change in weights for all the layers are calculated in the initial
steps. After calculating all the change in weight terms the new weights are
calculated. Several modifications can be made in the conventional BP
algorithm to improve its performance. The modifications may include
immediate weight update scheme, changing the activation functions,
adaptively changing the learning rates and momentum factor etc. Irrespective
of the modifications made in the BP algorithm the basic architecture of the
network remains the same.

1.2 INTRODUCTION TO FUZZY LOGIC

FL is particularly good at handling uncertainty, vagueness and
imprecision. This is especially useful where a problem can be described
linguistically (using words) or, as with neural networks, where there is data
and one is looking for relationships or patterns within that data. It is an
approach to uncertainty that combines real values \([0…1]\) and logic operations.
FL is based on the ideas of fuzzy set theory and fuzzy set membership often
found in natural (e.g., spoken) language. FL uses imprecision to provide robust solutions to problems. FL relies on the concept of a fuzzy set. The notation for fuzzy sets: for the member \( x \), of a discrete set with membership \( \mu \), is \( \mu/x \). In other words, \( x \) is a member of the set to degree \( \mu \).

Discrete sets are defined as:

\[ A = \mu_1/x_1 + \mu_2/x_2 + \ldots + \mu_n/x_n \]

FL systems are universal function approximators. In general, the goal of the FL system is to yield a set of outputs for given inputs in a non-linear system without using any mathematical model. Fuzzy model is a collection of IF – THEN rules with vague predicates that use a fuzzy reasoning such as Sugeno and Mamdani models. Sugeno type systems can be used to model any inference system in which the output membership functions are either linear or constant whereas Mamdani type produces either linear or nonlinear output.

FL controller contains four main parts, two of which perform transformations. The four parts are

- Fuzzifier (transformation 1)
- Knowledge base
- Inference engine (fuzzy reasoning, decision-making logic)
- Defuzzifier (transformation 2)

Figure 1.6 shows the schematic of FL System.
1.2.1 Fuzzifier

The fuzzifier performs measurement of the input variables (input signals, real variables), scale mapping and fuzzification (transformation 1). The measured input signals are scaled and are transformed into fuzzy quantities. This transformation is performed by using membership functions. A membership function has a value between 0 and 1, and it indicates the degree of belongingness of a quantity to a fuzzy set. If it is absolutely certain that the quantity belongs to the fuzzy set, then its value is 1, but it is absolutely certain that it does not belong to this set then its value is 0. Similarly if, for an example, the quantity belongs to the fuzzy set to an extent of 50%, then the membership function is 0.5 (Peter Vas 1993).

There are many types of different membership functions, piecewise linear or continuous. Some of these are smooth membership functions, e.g. bell-shaped, semicircular, Gaussian etc. and others are non-smooth, e.g. triangular, trapezoidal etc. The choice of the type of membership function used in a specific problem is not unique. Thus it is reasonable to specify parameterized membership functions, which can be fitted to a practical problem.
Summary of the popularly used membership functions is given in Table 1.1.

**Table 1.1 Summary of commonly used membership functions**

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Name of the membership functions</th>
<th>Parameters</th>
<th>Functional form</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Triangular shaped Membership Function (TRIMF)</td>
<td>a, b, and c</td>
<td>( F(x) = \begin{cases} \frac{x-a}{b-a} &amp; \text{if } a \leq x \leq b \ \frac{c-x}{c-b} &amp; \text{if } b \leq x \leq c \ 0 &amp; \text{Otherwise} \end{cases} )</td>
</tr>
<tr>
<td>2</td>
<td>Trapezoidal Membership Function (TRAPMF)</td>
<td>a, b, c and d</td>
<td>( F(x) = \begin{cases} \frac{x-a}{b-a} &amp; \text{if } a \leq x \leq b \ 1 &amp; \text{if } b \leq x \leq c \ \frac{d-x}{d-c} &amp; \text{if } c \leq x \leq d \ 0 &amp; \text{Otherwise} \end{cases} )</td>
</tr>
<tr>
<td>3</td>
<td>Gaussian Membership Function (GAUSSMF)</td>
<td>( \sigma ) and c</td>
<td>( F(x) = \frac{-(x-c)^2}{2\sigma^2} )</td>
</tr>
<tr>
<td>4</td>
<td>Sigmoidally shaped Membership Function (SIGMF)</td>
<td>a and c</td>
<td>( F(x) = \frac{1}{1 + e^{-a(x-c)}} )</td>
</tr>
<tr>
<td>5</td>
<td>Bell shaped Membership Function</td>
<td>a, b and ( x_o )</td>
<td>( F(x) = \frac{a}{a + b(x - x_o)^2} )</td>
</tr>
</tbody>
</table>
In this thesis, the Semicircular membership is also used. The Semicircular membership function depends on two parameters radius \( r \) and center of the circle \( c \). This can be described as follows:

\[
F(x) = \begin{cases} 
\sqrt{1 - \frac{(c-x)^2}{r^2}} & \text{if } -r < x < r \\
0 & \text{otherwise}
\end{cases}
\]  

A semicircular membership function can be represented graphically as shown in Figure 1.7.

![Semicircular membership function graph](image)

**Figure 1.7 Semicircular membership functions**

### 1.2.2 Knowledge base

The knowledge base consists of the database and the linguistic control rule base. The database provides the necessary information to define the linguistic control rules and the fuzzy data manipulation in the fuzzy logic controller. The rule base specifies the control goal actions by means of a set of linguistic control rules. In other words, the rule base contains rules provided by an expert. The FL controller looks at the input signals and by using the expert rules determines the appropriate output signals (control
actions). The rule base contains a set of if-then rules. The major methods of developing a rule base are:

- Using the experience and knowledge of an expert for the application and the control goals;
- Modeling the control action of the operator;
- Modeling the process;
- Using a self-organized fuzzy controller;

### 1.2.3 Inference engine

It is the kernel of a FL controller and has the capability both of simulating human decision-making based on fuzzy concepts and of inferring fuzzy control actions by using fuzzy implication and FL rules of inference as shown in Figure 1.8. In other words, once all the monitored input variables are transformed into their respective linguistic variables, the inference engine evaluates the set of if-then rules and thus a result is obtained which is again a linguistic value for the linguistic variable. This linguistic result has to be then transformed into a crisp output value of the fuzzy logic control.

![Graphical interpretation of fuzzification](image)

**Figure 1.8** Graphical interpretation of fuzzification
1.2.4 Defuzzifier

Defuzzification is a process of converting fuzzy values generated by inference engine into a crisp value that is compatible with output value. Different defuzzification methods such as Center of area (Centroid), Bisector of area (Bisector), Smallest of Maximum (SOM), Mean of Maximum (MOM), Largest of Maximum (LOM) have been used for fuzzy control.

The most often used defuzzification methods are given below:

1. Centroid method
2. Bisector method
3. MOM method
4. SOM method
5. LOM method

In general, defuzzification is the process of defuzzifying a fuzzy set $A$ of a universe of discourse $z$. The fuzzy set $A$ is usually represented by an aggregated output membership function.

**Centroid of area, $z_{COA}$**

$$z_{COA} = \frac{ \int_Z \mu_A(z) z \, dz }{ \int_Z \mu_A(z) \, dz },$$

where $\mu_A(z)$ is aggregated output membership function.
Bisector of area, $z_{BOA}$

$z_{BOA}$ satisfies the following equation

$$\int_{\alpha}^{\beta} \mu_A(z) \, dz = \int_{\alpha}^{\beta} \mu_A(z) \, dz,$$

where $\alpha = \min \{ z / z \in Z \}$ and $\beta = \max \{ z / z \in Z \}$.

The vertical line $z = z_{BOA}$ partitions the region between $z = \alpha$, $z = \beta$, $y = 0$ and $y = \mu_A(z)$ into the two regions with same area.

Mean of maximum, $z_{MOM}$

$z_{MOM}$ is the average of the maximizing $z$ at which the membership function reaches the maximum $\mu^*$

$$z_{MOM} = \frac{\int z \, dz}{\int \mu_A(z) \, dz} \quad \text{where} \quad z^1 = \{ z / \mu_A(z) = \mu^* \}.$$ In particular, if $\mu_A(z)$ has a single maximum at $z = z^*$ then $z_{MOM} = z^*$.

Smallest of maximum, $z_{SOM}$

$z_{SOM}$ is the minimum in terms of magnitude of the maximizing $z$.

Largest of maximum, $z_{LOM}$

$z_{LOM}$ is the maximum in terms of magnitude of the maximizing $z$.

The calculation needed to carry out any of these defuzzification operations is time consuming. Furthermore, these defuzzification operations are not easily subject to rigorous mathematical analysis. So, most of the studies are based on experimental results. The various defuzzification schemes are shown in Figure 1.9.
1.3 INTRODUCTION TO GENETIC ALGORITHM

Genetic algorithms are good at taking larger, potentially huge, search spaces and navigating them looking for optimal combinations of things and solutions which we might not find in a life time.

The most important aspects of genetic algorithms are:

- Definition of objective function
- Definition and implementation of genetic representation
- Definition and implementation of genetic operators.

During the creation of offspring, recombination occurs (due to cross over) and in that process genes from parents form a whole new chromosome in some way. The new created offspring can then be mutated. Mutation means that the element of DNA is modified. These changes are mainly caused by errors in copying genes from parents. The fitness of an organism is measured by means of success of organism in life.
Genetic algorithms are usually suitable for solving maximization problems. Minimization problems are usually transformed into maximization problems by some suitable transformation. In general, fitness function is first derived from the objective function and used in successive genetic operations. The error is used as the fitness function and the objective function is to minimize the error.

A simple GA largely uses three basic operators. They are:

- Reproduction
- Cross over
- Mutation.

1.3.1 Reproduction

Reproduction is usually the first operator applied on population. Chromosomes are selected from the population to be parents to cross over and produce offspring. According to Darwin’s evolution theory of survival of the fittest, the best ones should survive and create new offspring. There exists a number of reproduction operators in GA literature but the essential idea in all of them is that the above average strings are picked from the current population and their multiple copies are inserted in the mating pool in a probabilistic manner. The various methods of selecting chromosomes for parents to cross over are:

- Roulette-wheel selection
- Boltzmann selection
- Tournament selection
- Rank selection
- Steady-state selection
The commonly used reproduction operator is the proportionate reproductive operator where a string is selected from the mating pool with a probability proportional to the fitness.

1.3.2 Cross over

After the reproduction phase is over, the population is enriched with better individuals. Reproduction makes clones of good strings, but does not create new ones. Cross over operator is applied to the mating pool with a hope that it would create a better string. The aim of the cross over operator is to search the parameter space. In addition, search is to be made in a way that the information stored in the present string is maximally preserved because these parent strings are instances of good strings selected during reproduction.

Cross over is a recombination operator, which proceeds in three steps. First, the reproduction operator selects at random a pair of two individual strings for mating, then a cross-site is selected at random along the string length and the position values are swapped between two strings following the cross site. For instance, let the two selected strings in a mating pair be $A = 11111$ and $B = 00000$. If the random selection of a cross-site is two, then the new strings following cross over would be $A^* = 11000$ and $B^* = 00111$. This is a single-site cross over. Though these operators look very simple, their combined action is responsible for much of GA’s power. From a computer implementation point of view, they involve only random number of generations, string copying, and partial string swapping. There exist many types of cross over operations in GA viz.

- Single-site Cross over - Single site
- Two-point Cross over - Two-point
• Multi-point Cross over - Multi-point.

The term cross over rate is usually denoted as \( P_c \), the probability of cross over. The probability varies from 0 to 1. This is calculated in GA by finding out the ratio of the number of pairs to be crossed to some fixed population.

### 1.3.3 Mutation operator

After crossover, the strings are subjected to mutation. Mutation of a bit involves flipping it, changing 0 to 1 and vice versa with a small mutation probability \( P_{mu} \). The bit-wise mutation is performed. If the mutation at that site is selected and flipping is true, the outcome is false. If at any bit, the outcome is true then the bit is altered, otherwise the bit is kept unchanged.

For example, consider the following population having four eight-bit strings.

```
0110 1011
0011 1101
0001 0110
0111 1100
```

The above four strings have a zero in the leftmost bit position. If the true optimum solution requires a one in that position, then neither reproduction nor cross over operator described above will be able to create one in that position. The mutation is implemented with a probability \( P_{mu} \) of turning 0 to 1 as
Mutation helps to increase searching power.

1.4 LITERATURE REVIEW

1.4.1 Broken bar/end ring fault detection in induction motor

The cage rotor induction machines are widely used in industries because of their rugged construction (Ye and Wu 2000 and Kilman et al 1996, 1997). It is very essential to monitor these machines online and an effective intelligent fault detection schemes are necessary to improve operational reliability of the machine.

Many engineers and researchers have focused their attention on incipient fault detection and preventive maintenance, which aims at preventing major motor faults from occurring. As pointed out by Peter Vas (1993), the major faults of electrical machines can broadly be classified as the following:

- Stator faults resulting in the operating or shorting of one or more of a stator phase winding,
- Abnormal connection of the stator windings,
- Broken rotor bar or cracked rotor end-rings,
- Static and/or dynamic air gap irregularities,
- Bent shaft can result in a rub between the rotor and stator, causing serious damage to stator core and windings,
• Shorted rotor field winding, and
• Bearing and gearbox failures.

These faults produce one or more of the symptoms as given below:

• Unbalanced air-gap voltages and line-currents,
• Increased torque pulsations,
• Decreased average torque,
• Increased losses and reduction in efficiency and
• Excessive heating

Fabricated type rotors have more incidents of rotor bar and end-ring breakage than cast rotors. On the other hand, cast rotors are more difficult to repair once they fail. The reasons for rotor bar and end-ring breakage are several as pointed out by Filippetti et al (1996). They can be caused by

• Thermal stresses due to thermal overload, unbalance, hot spots or excessive losses and sparking (mainly fabricated rotors).
• Magnetic stresses caused by electromagnetic forces, unbalanced magnetic pull, electromagnetic noise and vibration.
• Residual stresses due to manufacturing problems.
• Dynamic stresses arising from shaft torques, centrifugal forces and cyclic stresses.
• Environmental stresses caused by for example contamination and abrasion of rotor material due to chemicals or moisture,
- Mechanical stresses due to loose laminations, fatigued parts and bearing failure

Different invasive and non-invasive methods for motor incipient fault detection have been reported in Keyhani and Miri (1986). Invasive techniques require expensive diagnostic equipment and/or off-line fault analysis to determine the motor condition. Many invasive techniques are used for fault detection and diagnosis in motors, such as radio frequency scheme, particle analysis, vibration analysis and thermal signature (Stone and Lyles 1991).

Non-invasive schemes are based on easily accessible and inexpensive measurements to predict the motor condition without disintegrating the motor structure. These schemes are most suitable for on-line monitoring and fault detection purposes. Many of the inexpensive and non-invasive techniques available for fault detection and diagnosis in motors, such as parameter estimation (Keyhani and Miri 1986), Stator current spectrum analysis (Benbouzid et al 1998).

The broken rotor bars can be detected by various methods, such as monitoring the supply current via a current transformer and perform high resolution spectrum analysis to detect the ± 2sf₁ sidebands, monitoring the stator core vibration via an accelerometer and perform high resolution spectrum analysis around the rotor slot passing frequencies to detect the ± 2sf₁ sidebands, monitoring the axial flux signal via a search coil around the rotating shaft and perform high resolution spectrum analysis to detect the ± 2sf₁ sidebands, monitoring speed oscillation via a once-per revolution transducer and perform additional processing to display the predicted number of broken rotor bars (Thomson 1994).
Many motor fault detection schemes have been developed and are being extensively used in the industries. They have achieved certain degree of success. Those fault detection schemes are either cost inefficient, unreliable or too difficult to use. With the advancement in technologies and multi-disciplinary collaboration, new opportunities have emerged to improve existing fault detection techniques and to drive the fault detection technology forward. One such key advancement in technology is in the area of artificial NN, which has been applied successfully in fields such as fault detection (Chow 1994).

Although the NN can provide the correct input-output fault detection relation, it is essentially a “black box” device; i.e., it does not provide heuristic reasoning about the fault detection process. FL could be a solution to this problem. FL can easily and systematically transfer heuristic, linguistic, and qualitative knowledge (preferred by humans) to numbers and quantitative knowledge (preferred by computers) and vice-versa. This provides a simple method to heuristically implement fault detection principles and to heuristically interpret and analyze their results. (Chow 1994). In this thesis, NN and FL based fault detection schemes for three phase squirrel cage induction motor are presented. These schemes monitor the stator current spectrum to detect the rotor asymmetry due to broken rotor bars.

1.4.2 Bearing fault detection in induction motor

Induction motors are a critical component of many industrial processes and are frequently integrated in commercially available equipment and industrial processes. Motor-driven equipment often provides core capabilities essential to business success and to safety of equipment and personal. There are many published techniques and many commercially available tools to monitor induction motors to ensure a high degree of
reliability. In spite of these tools, many companies are still faced with unexpected system failures and reduced motor lifetime. Environmental, duty, and installation issues may combine to accelerate motor failure far sooner than the designed motor lifetimes. These studies specifically apply to machines, which are operated in industrial and commercial installations. The results of these studies show that bearing problems account for over 40% of all machines failures. Over the past several decades, rolling-element (ball and roller) bearings have been utilized in many electric machines while sleeve (fluid-film) bearings are installed only in the largest industrial machines. In the case of induction motors, rolling element bearings are overwhelmingly used to provide rotor support (Kryter and Haynes 1989, Stack et al 2004).

In general, condition-monitoring schemes have concentrated on sensing specific failures modes in one of three phase induction motor components: the stator, the rotor, or the bearings. Even though thermal and vibration monitoring have been utilized for decades, most of the recent research has been directed towards electrical monitoring of the motor with emphasis on inspecting the stator current of the motor. In particular, a large amount of research has been directed towards the stator current monitoring to sense rotor faults associated with broken rotor bars and mechanical unbalance (Cardoso et al 1993).

All of the presently available techniques require the user to have some degree of expertise in order to distinguish a normal operating condition from a potential failure mode. This is because the monitored spectral components (either vibration or current) can result from a number of sources, including those related to normal operating conditions (Schoen et al 1995). This requirement is even more acute when analyzing the current spectrum of an induction motor since the harmonics exist due to both the design and construction of the motor and the variation in the load torque (Schoen et al
1997). Therefore it is necessary to design a system to eliminate induction motor load effects in the stator current monitoring.

Penman et al (1986) suggested the condition monitoring of the dynamic performance of electrical drives received considerable attention in recent years. Many condition-monitoring methods have been proposed for different type of rotating machine faults detection and localization. In fact large electro machine systems are often equipped with mechanical sensors, primarily vibration sensors based on proximity probes. Those, however, are delicate and expensive. Moreover, in many situations, vibration-monitoring methods are utilized to detect the presence of an incipient bearing failure. However, in Steele et al (1982) said that the stator current monitoring could provide the same indications without requiring access to the motor.

Frequency analysis on the vibration signals caused by the bearing wear can be done to evaluate the bearing condition of the machine (Bo Li et al 2000, Nandi and Toliyat 1999, Morel and Neau 1990). Frequency analysis of machine bearing faults is based upon premise that different types of faults occur in different frequency spectra. This method requires very accurate sensing equipment to capture impulses that occur for short durations over a broad frequency spectrum. Therefore, this method is considered useful for large machines because of high cost of accurate sensing devices.

The direct inspection of determining the bearing wear is less expensive. But, it requires down time for the motor (Sinan Altug et al 1999). The particle analysis of oil that is used to lubricate the machine is another method of bearing condition monitoring. This method requires bringing oil samples to a laboratory for chemical analysis. It is very difficult and impractical to implement on a constant basis. Loparo et al (2000) proposed a
model based approach for fault detection in rotating machines. This method requires the mathematical model of the machine.

The parameter estimation approach proposed by Chow and Fei (1993), Isermann and Feyermuth (1991) and Isermann (1997), can provide the information relative to the condition of the insulation and bearing. This method can be applied online but it requires the expensive monitoring equipments. This method relies upon a very accurate mathematical model of the machine. This technique is based on the detailed understanding of the system and how it is affected by the chosen parameters. The parameters can be chosen to reflect the condition of bearing and insulation. Unfortunately, it is very difficult to obtain the accurate mathematical model of the machine.

All the above said methods either require the services of an expert to determine the motor condition and expensive monitoring equipments or accurate mathematical model of the machine. The above methods cannot be applied online except the parameter estimation method. Unfortunately, the need of accurate mathematical model makes this method less attractive.

Chow et al (1991) and Chow and Fei (1995) have proposed the neural network approach and have demonstrated it to be an effective alternative for motor fault detection because NN can learn any arbitrarily complicated continuous nonlinear functions (Cyberko 1989). It can learn the motor fault detection process, which enables it to give the accurate solution to a particular fault detection problem. In addition, the NN can perform fault detection online through the use of inexpensive monitoring devices. These devices obtain the necessary measurements in a noninvasive manner. In addition to this, Keyhani and Miri (1986) have listed out the detailed merits of using NN instead of using other fault detection techniques.
Even though the NN is capable of mapping inputs and its associated outputs correctly for given problem, it cannot perform this function in heuristic manner as humans prefer. To overcome this shortcoming, FL can be used. It is well known that FL has the capability of transforming expert knowledge and linguistic variables into numerical values for the use of complex machine computations via fuzzy rules and membership functions.

Fuzzy systems have altered the interest of researchers in various engineering applications ranging from consumer products to industrial process control. The necessity of fuzzy control is to build a human like model without thinking in terms of mathematic model. The expert knowledge is transformed into control rules with proper choice of membership functions. Fuzzy rule based modeling is to identify the structures and parameters of a fuzzy if-then rule base so that a desired input and output mapping is achieved (Junhong and Derk Linkens 1998).

Goode and Chow (1995) proposed the fuzzy system based fault detection scheme and demonstrated its applicability for electric motors. This method requires to define the membership functions and rules for the input and output for fault detection problem. It does not require the machine parameters for fault detection.

Sinan Altug et al (1999) have proposed the fuzzy implemented fault detection scheme for induction machines. This scheme is capable of estimating the bearing condition of the motor. But the accuracy motor fault detection is not discussed.

Mishra et al (1996) presented a bearing fault detection scheme for a single phase induction motor. This method makes use of Gaussian membership function for fault detection. The accuracy of fault detection relies
upon the choice membership functions. But, the accuracy of this method can be improved further using proper types of membership functions, which is not discussed in this work.

Kothari et al (2001) presented a design and investigations on FL based power system stabilizer. They presented the detailed investigations on the design of fuzzy controller to improve dynamic response of the system considering different membership functions. But, they have not considered the different defuzzification methods while designing fuzzy controller, because the performance of fuzzy controller also depends on the choice of defuzzification schemes.

The fuzzy fault detection scheme based on stator current Concordia patterns has been presented in Zidani et al (2003). The stator current patterns are measured, recorded and used for Concordia patterns computation under different load conditions. This scheme can be implemented online and also requires the accurate sensing devices. Therefore this scheme is considered suitable for large machines. This thesis demonstrates the applications NN and FL for bearing detection by correlating the characteristic bearing frequencies to the spectral components of the stator current.

1.4.3 Fault detection in power transformer

To keep the power transformer in good shape, periodic examination of the transformer must be performed to find incipient fault inside and to prevent it from further deterioration as early as possible (Patel and Khubchandani 2004). Among the existing methods for identifying the incipient faults of power transformer, Dissolved Gas Analysis (DGA) is a method widely applied by many utilities and researchers (Yann Chang Huang 2003). The conventional schemes are Dornenburg and Rogers gas ratio
methods. Common to these methods are partition of attributes into several intervals. Depending on the combinations of intervals the attributes fall in, the corresponding fault type is identified. However, the numerical threshold of each attribute often varies from utility to utility. The condition of dissolved gases in transformer oil is related to ratios of specific gases, their generation rates and total combustible gases. As a result, the diagnosis accuracy of the existing DGA methods is still unsatisfactory.

The conventional method of DGA needs an expert to evaluate the fault condition. The oil samples are collected and used for oil testing. This needs the offline operation (Syed A Word 2003). Recently, online monitoring of electrical apparatus becomes popular because of the development of intelligent system like fuzzy and neural system (Haung et al 1997, Yang and Liao 1999, Xu et al 1997 and Tomsovic et al 1993).

Syed A Word (2003) and Kelly (1980) have proposed a scheme to provide advance warning of developing faults to provide convenient scheduling of repairs, status check on new repaired units, monitoring of units under overload and reasons for failure of transformer.

The manufacturer's view on transformer monitoring is presented in Bengtsson (1996). Monitoring of transformers has been discussed based on a review of the changes occurring on the electricity market, emphasizing its commercial aspects. The key issues related to on-line monitoring such as reliability and cost were discussed. A survey of the most important methods for on-line monitoring and off-line diagnostics was presented.

Partial discharge is one of the factors that could lead to failure of power transformers, leading to power outage and expensive repairs. The fault detection scheme to detect partial discharge has been discussed in Wang et al
The acoustic wave generated by partial discharge has been measured and used for monitoring, diagnosing, and locating potential failures in the transformers. Fiber optic sensors have been shown to be attractive devices for PD detection because of a number of inherent advantages including small size, high sensitivity, electrical non-conductivity and immunity to electromagnetic interference.

The NN based scheme to discriminate the inrush currents and internal faults has been discussed in Okan Ozgonenel (2005) by combining wavelet transform and NN technique. The neural network was used to classify faults correctly. The wavelet transform based scheme has been proposed in Pandey et al (1998). The analysis was based on dyadic-orthonormal wavelet transforms. The approach decomposed a given faulty neutral current response into other signals, which represent a smoothed and detailed version of the original. The decomposition has been performed by the multi-resolution signal decomposition technique.

In Morais and Rolim (2006), the hybrid tool has been developed to diagnose faults in power transformers through the analysis of dissolved gases in oil. The traditional criteria of the dissolved gas analysis, NN artificial neural network and FL have been used for fault diagnosis (Dukarm 1993, Chang 1994 and Lin et al 1993). The results obtained with this tool in the diagnosis of incipient faults in transformers were 80%. A two-step neural network method has been used to detect faults in oil filled power transformer based on dissolved gas analysis with or without cellulose involved (Zhang et al 1996) Good diagnosis accuracy has been obtained. However, the accuracy of fault diagnosis could be improved choosing the proper value of learning rate, momentum factor and activation functions.
In this thesis, NN and FL based fault detection schemes are proposed for incipient fault detection in oil filled power transformer using dissolved gas analysis. The accuracy of the fuzzy based fault diagnosis is analysed for different membership functions and compared with the neural network based scheme.

1.4.4 Transient stability enhancement of synchronous generator

Synchronous generators are considered to be the most important and expensive part in an electrical power system. Transient Stability is the ability of a power system to maintain synchronization of various components in the system. In other words, stability of a power system is its ability to return to normal or stable operation after the disturbance (Kundur 1994).

Among the various methods of improving transient stability dynamic Braking Resistor (BR) scheme is well known powerful tool. The BR can be viewed as a fast load injection to absorb excess transient energy of an area, which arises due to severe system disturbances (Hadi Saadat 1999).

Shelton et al (1975) and Rahim and Alamahir (1988) have been proposed a conventional controller for switching the BR. These controllers are inflexible and are not adaptive to the changing operating condition of the system. Therefore, a FL controlled BR scheme is proposed in this thesis, which gives robust performance under parameter variations.

The design of fuzzy controller has been proposed in Chatterji et al (2000) with one input variable and three control rules in contrast to the fizzy controller described in Hiyama et al (1995) where two input variables and 49 control rules have been used.
Mohamd Hasan et al (2001), Ali et al (2002, 2001) have proposed a fuzzy controlled BR scheme for transient stability enhancement. The firing-angle has been calculated using the approximation method. The effectiveness of fuzzy system can be improved choosing proper membership functions. In this thesis, a fuzzy controlled BR scheme is proposed for transient stability enhancement and its effectiveness is analysed for different membership functions.

1.5 OBJECTIVES OF THE THESIS

This thesis has the following major objectives:

1. The first objective is to analyse the different fault detection schemes for induction motors and power transformer. The proposed schemes are to be implemented online using NN and FL. The simulation results of proposed schemes are to be verified experimentally.

2. The second objective is to optimize the parameters of the FL based fault detection scheme to improve the diagnostic accuracy and also to investigate the choice of membership functions for effective fault detection.

1.6 ORGANIZATION OF THE THESIS

This thesis investigates the applications of NN, FL and GA to real time monitoring of induction motor and Power transformer. The thesis is divided into six chapters. Chapter 1 is introductory and it explains the basis of NN, FL, GA and literature review.
Chapter 2 describes broken bar and end ring fault detection schemes for three phase squirrel cage induction motor. The fault detection scheme uses NN, FL for fault diagnosis. To improve the diagnostic accuracy of fuzzy based fault detection scheme, the membership functions are optimized using GA. The fault detection scheme is based on monitoring the harmonic components of stator currents.

In Chapter 3, bearing fault detection scheme for three phase induction motor is presented. The scheme monitors the stator current spectrum for fault signature extraction. The NN and FL are used for intelligent fault diagnosis.

Chapter 4, describes the fault detection scheme for power transformer using DGA approach. The NN and FL are used for intelligent fault diagnosis. The dissolved gases are monitored for fault detection.

In Chapter 5, a fuzzy controlled BR scheme is proposed for controlling the firing angle of the thyristor switched BR to improve the transient stability of a synchronous generator. The fuzzy controller is tested with triangular, trapezoidal and Gaussian membership functions.

The conclusion and the scope for future work are discussed in Chapter 6.