JJMIE

Volume 3, Number 4, December 2009
ISSN 1995-6665
Pages 294 - 305

Jordan Journal of Mechanical and Industrial Engineering

An Intelligent Machine Condition Monitoring System Using Time-Based Analysis: Neuro-Fuzzy Versus Neural Network

M. Samhouri [a,*], A. Al-Ghandoor [a], S. Alhaj Ali [a], I. Hinti [b], W. Massad [a]

[a] *Department of Industrial Engineering, College of Engineering, Hashemite University, P.O. Box 150459, 13115 Zarka, Jordan*

[b] *Department of Mechanical Engineering, College of Engineering, Hashemite University, P.O. Box 150459, 13115 Zarka, Jordan*

Abstract

Monitoring and predicting machine components' faults play an important role in maintenance actions. Developing an intelligent system is a good way to overcome the problems of maintenance management. In fact, several methods of fault diagnostics have been developed and applied effectively to identify the machine faults at an early stage using different quantities (Measures or Readings) such as current, voltage, speed, temperature, and vibrations. In this paper, an intelligent machine condition monitoring and diagnostic system is introduced with experimental verification. An adaptive neuro-fuzzy inference system (ANFIS) and a neural network system (NN) are used to monitor and predict the fault types of a critical mechanical element in the Potash industry (namely; a Carnallite surge tank pump). The system uses a piezoelectric accelerometer to generate a signal related to machine condition and fault type. Combinations of the vibration time signal features (i.e., root mean square, variance, skewness, kurtosis, and normalized sixth central moment) are used as inputs to both ANFIS and neural nets, which in turn output a value for predicted fault type. Experimental validation runs were conducted to compare the actual fault types with the predicted ones. The comparison shows that the adoption of the time root mean square and variance features achieved the minimum fault prediction errors for both ANFIS and neural nets. In addition, trapezoidal membership function in ANFIS achieved a fault prediction accuracy of 95%, whereas, a cascade forward back-propagation neural network achieved a better fault prediction accuracy of 99%.

Keywords: Condition Monitoring; Time Analysis; Neuro-Fuzzy.

1. Introduction

In recent years, with the rapid development of condition monitoring and forecasting, information processing, fault detecting, and artificial intelligence technology, it has been possible and feasible to monitor and forecast equipment condition and assess its health online. It is well recognized that optimized maintenance practices within an industrial setting require the correct blend of maintenance strategies. Condition-based (reliability centered, predictive, proactive) maintenance is an important part of this blend for many compelling reasons [1].

Recently, there has been a significant amount of research effort directed towards developing and implementing useful automated machinery fault detection and diagnostic tools. Most of these tools have been based on various pattern-recognition schemes, knowledge-based systems or artificial neural networks systems. The main thrust of the work has been towards developing systems that are not only objective in their treatment of data and presentation of results, but also flexible, thereby being applicable in a wide range of situations. A new method using fuzzy logic techniques to improve the performance of the classical inductive learning approach was presented by [2]. In [2], a hard cut point was proposed to discritize the continuous-valued attributes by using soft discritization to enable the systems have les sensitivity to noise. In [3], they used the concept of the fuzzy fractal dimension to measure the complexity of a time series of observed data from the plant. A method for analyzing and forecasting field failure data for repairable systems was proposed by [4]. This novel method constructed a predictive model by combining the seasonal autoregressive integrated-moving average (SARIMA) method and neural network model. In [5], they introduced a new combined method based on wavelet transformation, fuzzy logic, and neuro-networks for fault diagnosis of a triplex.

* Corresponding author. Samhouri@hu.edu.jo.

Most real life physical systems are nonlinear, ill-defined and uncertain which makes them difficult to model by conventional mathematical means. Furthermore, most industrial processes are based on the assumption that the process is a linear system. Fuzzy logic and neural networks have the potential to deliver successful solutions to problems that have previously proved difficult or impossible to handle by conventional linear methods [6-9]. Fuzzy logic and fuzzy inference systems have been shown to be effective techniques for the identification and prediction of complex, nonlinear, and vague systems. Fuzzy logic is particularly attractive due to its ability to solve problems in the absence of mathematical models.

Condition-based maintenance (CBM) is a method used to reduce the uncertainty of maintenance activities, and is carried out according to the need indicated by the equipment condition [10]. CBM assumes that existing indicative prognostic parameters can be detected and used to quantify possible failure of equipment before it actually occurs. Prognosis parameters provide the indication of potential problems and incipient faults which would cause the component or equipment to deviate from the acceptable performance level [11]. A number of computational tools have been developed for condition-based maintenance such as knowledge base [12-13], analytic hierarchy process [14-15], Petri nets [16], neural networks [17-18], and fuzzy logic and networks [19-20].

This study was motivated by the problem of improper condition-based maintenance strategy and a need of a process industry (Potash production) for a fast enough machine monitoring system to be employed as real-time fault detection system at their plants. In this paper, the maintenance records analysis is used to provide critical information from past experience to improve current maintenance process in the Potash industry. A neuro-fuzzy technique (i.e., ANFIS) and a neural network of time-based analysis are used to build an intelligent condition monitoring system to predict the type of fault or failure for one of the critical production units. Different combinations of five statistical parameters computed from the vibration time signal of a critical pump were fed as inputs into both ANFIS and neural network to output a value for the predicted fault. The procedure is illustrated using the vibration data of a carnallite surge tank pump with normal and faulty pump.

This paper is organized as follows: Section 1 is an introduction. Intelligent condition monitoring and fault diagnosis system will be given in the second section. Section 3 is about vibration data analysis and feature selection. Structure of adaptive neuro-fuzzy inference system and neural network will be discussed in section 4. ANFIS and neural-networks time-based fault diagnosis system is the subject of section 5. The last section is to conclude.

2. Intelligent Condition Monitoring and Fault Diagnosis System.

Condition monitoring is becoming popular in industry because of its efficient role in detecting potential failures. The use of condition monitoring techniques will generally improve plant production availability, and reduce downtime cost, but in some cases it also tends to over-maintain the plant in question. If a hidden defect is already present, with the help of condition monitoring, the defect may be identified and corrective actions may be taken. It is noted however that for a cost-effective maintenance, advanced prediction of such a defect and its development is very important since ordering spare parts and possible production shutdowns for maintenance may be costly and require careful planning well before the failure actually occurs.

Condition monitoring traditionally means acquiring data from various classes of plant which gives an indication of the condition of machine. Condition monitoring is an essential element of predictive maintenance. An ideal condition monitoring system would accept measured data as input and will produce the operational status, a possible mode of failure and time to failure as output.

Many machinery fault diagnostic techniques use automatic signal classification in order to increase accuracy and reduce errors caused by subjective human judgment. Detection of machine faults like mass imbalance, rotor rub, shaft misalignment, gear failure, and bearing defects is possible by comparing the vibration signals of a machine operating with and without faulty conditions. These signals can also be used to detect the incipient failures of the machine components through online monitoring system, reducing the possibility of catastrophic damage.

In intelligent maintenance management systems, IMMS, the three "isolated islands" of the automation system (i.e., monitoring and forecasting, diagnosis and prognosis, and maintenance decision making) are integrated into an organic system, and maintenance improved by sharing information among these systems. Intelligent methods try to decode the intelligence supplied from the system. Artificial intelligence techniques such as expert systems, neural networks, genetic algorithms, and fuzzy logic, have been widely applied in mechanical equipment monitoring and diagnosis with different aspects and degrees. It also noticed that different techniques have their unique advantages and disadvantages, and usually cannot replace each other.

The main problem of the Potash industry (Arab Potash Company, APC) is the visual inspection of frequency analysis performed at the preventive maintenance department, and more importantly the low speed of the automatic frequency-based monitoring system which is no longer suitable for real-time applications. Although they have a very progressive maintenance software system used to assent the conjuncture of their machines, as an attempt to control the break downs and the health conditions of the machines, the visual inspection by maintenance staff results in big discrepancies between the predicted and actual faults, thus causing time delays, inconsistencies, increased inaccurate maintenance activities, and of course increased loss of money. The carnallite surge tank pump is one of the important components that undergoes repetitive failures which cause breakdowns for the process, Figures 1 shows one of the carnallite surge tank pumps.

To solve this problem, an adaptive neuro-fuzzy inference system (ANFIS) as well as a neural network was used in this paper to replace the human operator and the automatic frequency-based system for predicting the faults types from the original time signal. The critical component (namely; the carnallite surge tank pump) was selected to apply our intelligent fault diagnosis system because this component has faced many unsuccessfully-predicted breakdowns and failures, which resulted in unnecessary money loss. The carnallite pump, shown in Figure 1, is considered a critical component because its breakdown could easily cause a production shut down or delay. A special software (i.e., EMONITOR Odyssey Delux) is used at this Potash industry to display and manipulate the vibration time signals coming from a piezoelectric accelerometer mounted on the pump component. As given in Figure 2, the frequency pattern of the original time signal is clear, but visual and automatic inspection and prediction usually results in an inaccurate diagnosis of the fault type.

Figure 1. The carnallite surge tank pump.

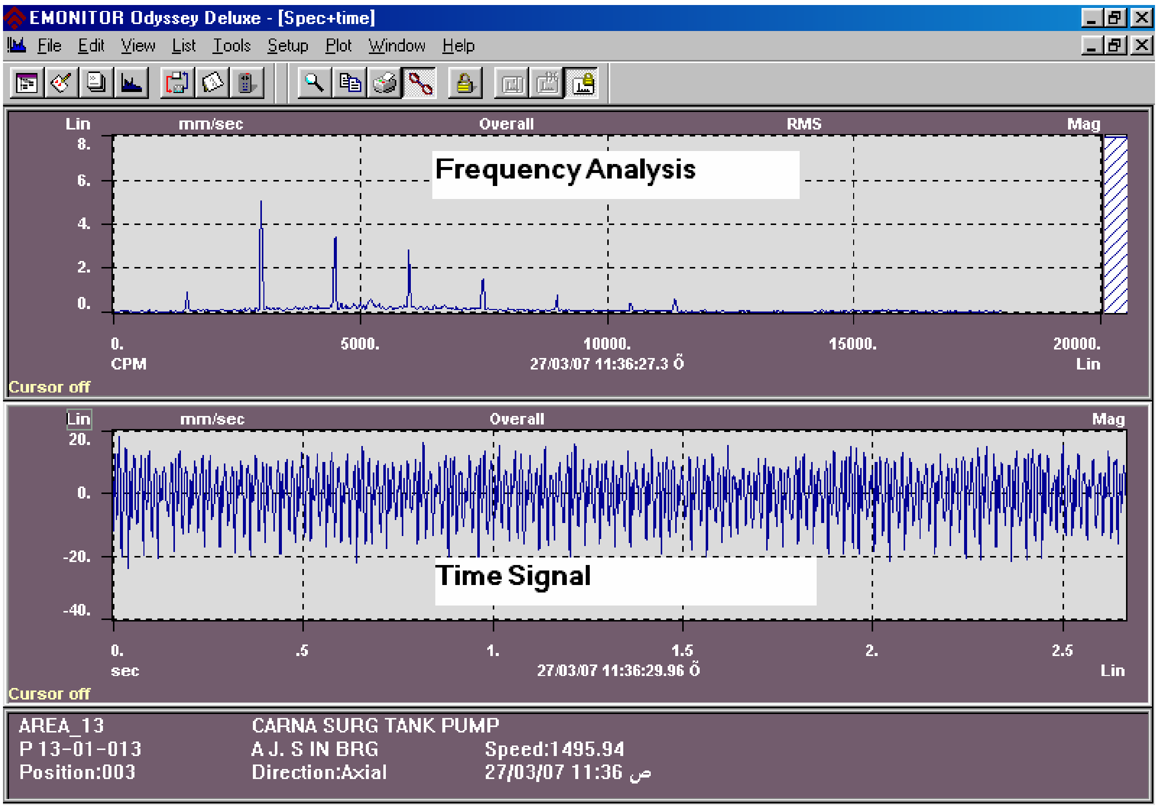

Figure 2. A sample of the vibration time signal in the axial direction along with its frequency-domain analysis using Odyssey software.

  - Volume 3, Number 4 (ISSN 1995-6665)

3. Vibration Data Analysis and Feature Selection

The traditional way of observing signals is to view them in what is called the time domain. The time domain is a record of what happened to a parameter compared to time. Typically, the signal would be displayed on an oscilloscope or a computer screen as given in Figure 2. In the analysis of time series signals, certain restrictions are imposed by the length of the data window (T), being analyzed and by the sampling rate (fs), used when digitizing continuous data [21]. A sample of time series segment of length T = 2.8 sec is shown in Figure 2. This is a standard time duration used at the Potash production plant to pick up useful vibration signals for frequency analysis.

Machines faults diagnosis and prediction requires generating representative and useful information about the vibration features by means of a sensor. Our approach to predict the fault type is to mount a piezoelectric accelerometer on the machine's component under study in order to give a time-series signal which is supposed to contain useful information about the machine's faults, failures and health conditions. The data used in building intelligent maintenance system has undergone several processing and analysis steps which will be described briefly in this section.

The first step is the vibration measurement by using sensor as shown in Figure 3; this sensor is a piezoelectric accelerometer. Accelerometers are absolute vibration transducers which produce a signal proportional to the vibration acceleration. The piezoelectric accelerometer is most attractive in view of its rigidity, wide frequency range, flat response and dynamic range, this sensor has the ability to measure the vibration in the three dimensions (namely; axial, horizontal, and vertical).

As an example, it is important to note that carrying out measurements on bearings readings should be taken in both radial and axial planes. Using both planes is an important method for distinguishing between various mechanical faults.

The piezoelectric accelerometer is connected with data base collection device which in turn is connected with a computer that has the analysis software (i.e., Emonitor odyssey deluxe) as shown in Figure 3, which in turn applies preliminary signal automatically on the vibration time signals (e.g., low-pass filtering, and windowing) and transfers the vibration from time domain to frequency domain. Figure 3 shows the Graphical User Interface for data transfer of the data base from the data collection device to computer. The output of these signal handling and processing when displayed on Emonitor software screen, is shown as the last step of Figure 3.

Upon the final output of signal processing steps is generated as illustrated in Figure 3, and before analyzing data using neuro-fuzzy or neural networks, the time data of 701 observations were divided into 3 bins each, with 234 non-overlapping samples in each bin. Each of these bins has been processed using MATLAB 7.0 to extract the following statistical five features:

1. Root mean square (rms),
2. Variance (σ^2),
3. Skewness (normalized third central moment, γ_3),
4. Kurtosis (normalized fourth central moment, γ_4) and

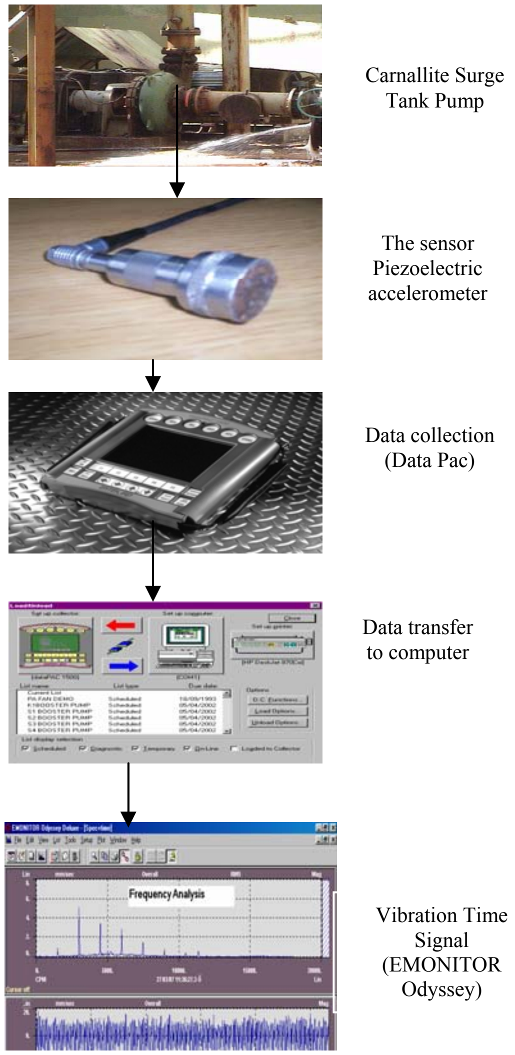

Figure 3. Experimental setup used to generate vibration time signals (i.e., training data)

5. Normalized sixth central moment (γ_6).

The above-mentioned statistical parameters of the time-domain signal could be evaluated as follows:

$$rms = \left(\sum \frac{(yi_{avg})^2}{n}\right)^{1/2} \qquad (1)$$

$$\sigma^2 = E(yi_{avg})^2 \qquad (2)$$

$$\gamma_3 = \frac{E(yi_{avg})^3}{\sigma^3} \qquad (3)$$

$$\gamma_4 = \frac{E(yi_{avg})^4}{\sigma^4} \qquad (4)$$

$$\gamma_6 = \frac{E(yi_{avg})^6}{\sigma^6} \qquad (5)$$

Where E is the expected value, yi is the time signal amplitude, yiavg = yi – μ, and the mean is estimated as: μ = E {yi}, These five features resulted from previous step were used as inputs and the faults codes of Table 1 as output into ANFIS or neural network toolboxes of Matlab 7.0. Table 2 gives a whole set of data for component item 13 (i.e., carnallite surge tank pump), axial direction, depending on two of the time signal features (i.e., rms and σ2), along with the fault type codes as given in Table 1. These data points of Table 2 were divided into training set

Table 2. The training data set for component 13, axial direction, with rms and σ2 as inputs.

rms1	$\sigma^2 1$	rms2	$\sigma^2 2$	rms3	$\sigma^2 3$	Actual Fault code
3.7307	13.981	3.8662	15.012	3.9502	15.671	300
1.2485	1.5655	1.223	1.5022	1.2897	1.6706	200
1.3565	1.848	1.3335	1.786	1.3234	1.759	200
0.9928	0.9899	1.1611	1.3539	1.0622	1.133	200
1.3402	1.8039	1.3107	1.7254	1.171	1.3771	200
1.6552	2.7515	1.6109	2.2908	1.6855	2.8531	200
1.215	1.4827	1.2306	1.521	1.2481	1.5643	200
13.076	171.72	12.666	161.12	22.64	514.77	300
2.8199	7.9862	2.7284	7.4761	2.7346	7.5099	200
1.7922	3.2259	1.5147	2.3042	1.3804	1.9136	200
1.3467	1.8215	1.5302	2.3514	1.6215	2.6406	200
2.443	5.9941	2.819	7.9807	2.8086	7.9223	200
1.8628	3.4851	1.7569	3.1	1.6668	2.79	200
0.9843	0.9731	1.0165	1.0377	0.984	0.9723	200
3.1338	9.8633	3.1908	10.225	3.0204	9.1623	300
1.9599	3.8577	1.8879	3.5796	2.08	4.3449	200
0.907	0.8261	0.8984	0.8107	0.9357	0.8793	200
2.0023	4.0265	1.9251	3.7218	1.8842	3.5653	200
1.0872	1.1872	1.0067	1.0177	1.0271	1.0595	200
2.5425	6.4922	2.5336	6.4465	2.4124	5.8446	200
2.2107	4.9087	2.0619	4.2697	2.2094	4.9023	200
1.2672	1.6126	1.2585	1.5906	1.2272	1.5125	200
2.4477	6.0169	2.3541	5.5657	2.1948	4.8379	200
5.6621	32.197	5.8764	34.68	5.6977	32.603	100
6.637	44.24	7.0581	50.023	6.7727	46.067	100
1.206	1.4608	1.1843	1.4083	1.1547	1.3391	200
2.6178	6.8826	2.6075	6.8281	2.3234	5.4214	200
2.3611	5.599	2.687	7.251	2.5142	6.3485	200
1.7136	2.9492	1.8835	3.5627	1.7775	3.1732	200
1.6569	2.7572	1.4991	2.2569	1.3466	1.8212	200
0.8988	0.8114	0.8949	0.8042	0.8966	0.8073	200
0.0989	0.0098	0.1002	0.0101	0.0924	0.0086	200
2.5588	6.5757	2.4251	5.9065	2.3858	5.7163	200
1.2306	1.5209	1.1383	1.3013	1.0534	1.1144	200
0.8541	0.7326	0.8338	0.6982	0.9651	0.9354	200
3.0268	9.2009	2.9115	8.5136	3.042	9.2938	300
1.7095	2.935	1.8268	3.3518	1.7048	2.9189	200
5.6362	31.904	5.5425	30.852	5.6648	32.227	100
3.8295	14.729	4.0158	16.197	3.9914	16.001	300
1.5041	2.272	1.457	2.1401	1.6345	2.6796	200
1.1725	1.3806	1.114	1.2462	1.1229	1.2662	200
1.0424	1.0912	1.067	1.1433	1.0358	1.0776	200
2.1943	4.8359	2.4021	5.7949	2.3776	5.6775	200
3.8507	14.892	4.0665	16.608	3.731	13.98	300

Table 2 continues next page…

rms1	σ21	rms2	σ22	rms3	σ23	Actual Fault code
1.3712	1.8883	1.6033	0.2146	1.2241	1.5049	200
1.5978	2.5639	1.5616	2.4491	1.6925	2.8769	200
2.2375	5.0282	11.056	0.5714	2.7876	7.8044	200
2.9763	8.8968	4.1149	17.059	4.0959	16.827	200
1.6633	2.7785	1.6581	2.7613	1.509	2.2869	200
1.1405	1.3063	1.0985	1.2119	1.1265	1.2745	200
1.4592	2.1385	1.2842	1.6564	1.1575	1.3455	200
2.8833	8.3492	2.8825	8.3449	3.0087	9.0914	200
0.4385	0.1931	0.5096	0.2609	0.538	0.2907	200
5.1991	27.148	4.8232	23.364	4.7715	22.864	100
2.3357	5.479	2.9529	8.7569	3.3841	11.501	200
11.441	131.46	11.178	125.5	9.1815	84.662	200
2.1325	4.567	2.049	4.2164	1.8905	3.589	200
2.7235	7.4494	3.0313	9.2283	2.8	7.8737	200
0.9063	0.8249	1.0504	1.1081	1.1852	1.4108	200
1.2134	1.4788	0.664	0.4428	0.7107	0.5073	200
2.6068	6.8247	2.4486	6.0204	2.4909	6.2323	200
1.336	1.7925	1.2764	1.6361	1.291	1.6739	200
1.2019	1.4509	1.2263	1.5102	1.1169	1.2528	200
1.9368	3.7674	1.9661	3.8821	2.0943	4.4048	200
2.0871	4.375	2.3831	5.7027	2.254	5.1035	200
1.1115	1.2408	1.1403	1.306	1.183	1.4055	200
0.8634	0.7487	0.9165	0.8436	0.9324	0.8732	200
2.41	5.833	2.4545	6.0507	2.5355	6.4566	200
2.2684	5.1684	2.5308	6.433	2.7589	7.6451	200
5.9603	35.678	5.6239	31.765	5.8444	34.305	100
3.5534	12.681	4.432	19.727	3.6933	13.699	300
2.6698	7.1583	2.3951	5.7612	2.7892	7.8134	200
1.3009	1.6997	1.5103	2.2908	1.531	2.3541	200

(73 points) and testing set (19 points) in order to train and test ANFIS and neural networks as will be discussed in sections 5 and 6.

4. Neural Networks and Adaptive Neuro-Fuzzy Inference System

4.1. Neural Networks

Neural networks are universal function approximators. They are "model-free estimators" [22]. The first mathematical model of a neuron was proposed by [32] in 1943. It was a binary device using binary inputs, binary output, and a fixed activation threshold. In general, an artificial neural network, ANN (or simply neural network, NN) is a computational model defined by the following four parameters:

- Type of neurons (also called nodes).
- Connectionist architecture the organization of the connections between neurons.
- Learning algorithm.
- Recall algorithm.

Figure 4 shows artificial neural network architecture. The commonest type of artificial neural network consists of three groups, or layers, of units: a layer of "input" units is connected to a layer of "hidden" units, which is connected to a layer of "output" units. The activity of each layer could be summarized as follows: The activity of the input units represents the raw information that is fed into the network. The activity of each hidden unit is determined by the activities of the input units and the behavior of the output units depends on the activity of the hidden units and the weights between the hidden and output units. A one-layer network with R input elements and neurons are illustrated in Figure 4.

4.2. Adaptive Neuro-Fuzzy Inference System (ANFIS)

Adaptive neuro-fuzzy inference system (ANFIS) is a fuzzy inference system implemented in the framework of an adaptive neural network. By using a hybrid learning procedure, ANFIS can construct an input-output mapping based on both human-knowledge as fuzzy If-Then rules and stipulated input-output data pairs for neural networks

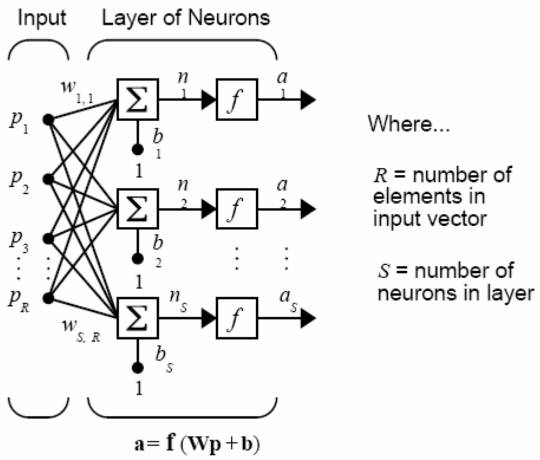

Figure 4. Artificial neural network [33].

training. ANFIS architecture is shown in Figure 5, where x and y are the inputs, f is the output, A_i and $A_n{}^2$ are the input membership functions, w_i and $w_n{}^2$ are the rules firing strengths. Five network layers are used by ANFIS to perform the fuzzy inference process.

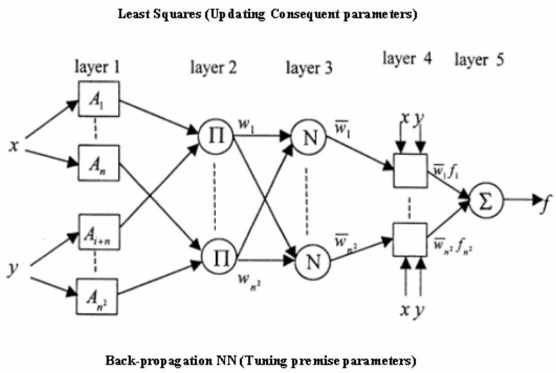

Figure 5. ANFIS architecture [33]

ANFIS is more powerful than the simple fuzzy logic algorithm and neural networks, since it provides a method for fuzzy modeling to learn information about the data set, in order to compute the membership function parameters that best allow the associated fuzzy inference system to track the given input/output data [23].

The architecture of ANFIS, illustrated in Figure 5, has five layers to accomplish the tuning process of the fuzzy modeling system. The five layers are:

1. Layer 1: Every node in this layer is an adaptive node with a node function (i.e., membership function). Parameters of membership functions are referred to as premise or antecedent parameters.
2. Layer 2: Every node in this layer is a fixed node, which multiplies the incoming signals and sends the product out. Each node represents the firing strength of a fuzzy rule.
3. Layer 3: Every node in this layer is a fixed node which calculates the ratio of the one firing strength to the sum of all rules' firing strengths. The outputs of this layer are called normalized firing strengths.
4. Layer 4: Every node in this layer is an adaptive node with a node function (i.e., linear combination of input variables). Parameters in this layer are referred to as consequent parameters.
5. Layer 5: The single node in this layer is a fixed node that computes the overall output as the summation of all incoming signals

In the next section, fault-diagnosis systems based on ANFIS and neural networks will be presented.

5. Intelligent Fault-Diagnosis Systems

5.1. Anfis-Based Fault-Diagnosis System

ANFIS prediction of machine's fault types starts by obtaining the data set (input-output data pairs) and dividing it into training and testing or validating data sets. The training data set is used to find the initial premise parameters for the fuzzy membership functions by equally spacing each membership function. The testing data used to validate the system .

Using ANFIS editor of Matlab 7.0 , statistical input data (i.e., rms, σ^2, γ_3, γ_4, and γ_6) were used to train and test the system. In fact, these five statistical features in each bin will complicate the structure of ANFIS. Therefore, combinations of two features were tested to build the system. Tables 3 and 4 shows the input features ranges and a sample of features used for training, respectively.

Table 3. Inputs features with their ranges which were used to generate the training data for the time-based ANFIS fault-diagnosis system

Feature	Meaning	Range
rms	Root mean square	0.09 - 46
σ^2	Variance	0.008 - 960
γ_3	Skewness	-0.5 - 0.6
γ_4	Kurtosis	1.5 - 6.0
γ_6	Normalized sixth central moment	3E-6 - 5E+10
FC	Fault Code	100 - 600

Table 4. Sample of ANFIS training data.

Component number	13_11_a_01	13_11_a_02	13_11_a_03	13_11_a_04	13_11_a_05	13_11_a_06
RMS	7.5613	3.7307	1.8738	13.1524	1.2485	1.3565
VAR	57.4192	13.9806	3.5261	173.7325	1.5655	1.848
RMS	7.6958	3.8662	1.7561	12.6339	1.223	1.3335
VAR	59.48	15.0118	3.0971	160.3005	1.5022	1.786
RMS	7.5859	3.9502	1.7603	22.7711	1.2897	1.3234
VAR	57.7924	15.6711	3.1119	520.7485	1.6706	1.759
Fault Code	100	300	200	300	200	200

The total number of data was 92 points. 73 points were used for training and 19 points (i.e., 20% of the total data points in order to make the process valid statistically), which are different and independent of the training data, were used for testing. In addition, each of the original data points (i.e., training and testing) is an average of 4 readings (replicates) in order to insure the statistical validity of this work. Using a given input/output data set, the toolbox function ANFIS constructs a fuzzy inference system (FIS) whose membership function parameters are tuned (adjusted) using either a backpropagation algorithm alone or a mixture of backpropagation and least squares (hybrid method) . In this study, the membership function parameters were updated using the hybrid method.

ANFIS takes the experimental data of the vibration features in each bin (rms1(RMS1), σ21(VAR1), rms2(RMS2), σ22(VAR2), rms3(RMS3), σ23(VAR3)) as input training data of the system. Different ANFIS parameters were tested as training parameters in order to achieve the perfect training and the maximum prediction accuracy.

The training data set has been used to set the initial parameter of the (ANFIS) model. This model has been trained with different parameter in order to get the minimum training and testing error. 83 data points out of 92 total pointes were adopted for training the system, the remaining 19 points were devoted to test and validate the system. Some fault codes did not appear in the training/testing data because these faults did not occur on the machine's components during the study period or in the machine's history, and consequently were excluded from the training/testing data.

A total of 216 fuzzy rules were used to build the fuzzy inference system. A Gaussian membership function (MF) was adopted to train ANFIS because it achieved minimum training error at epoch 170, as shown in the training curve of Figure 6. Figure 7 shows that the system is very well-trained to predict the machine's fault type.

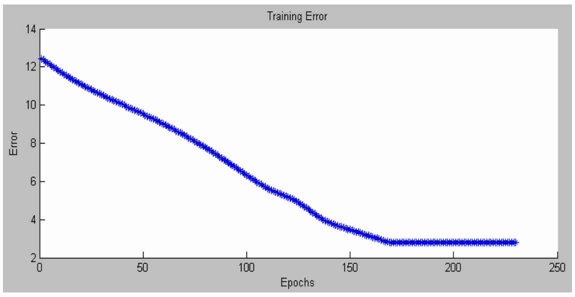

Figure 6. ANFIS training curve.

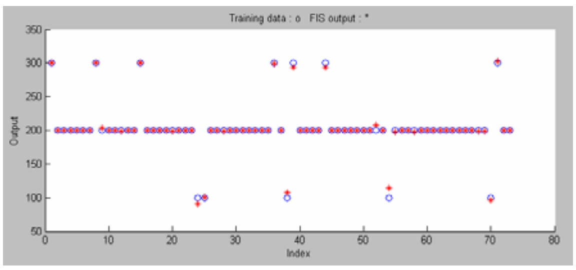

Figure 7. Actual and Predicted fault type values.

A perfect training is clear in this figure. Three Gaussian membership functions (MF) were used for root mean square inputs (RMS1, RMS2, RMS3) while another two Gaussian membership functions were adopted for variance inputs (VAR1, VAR2, VAR3). The final ANFIS-tuned (MF) for all input features are illustrated in Figures 8. The training root mean square error (RMSE) was dropped from 44.7159 when using root mean square (rms) and centralized six momentums (γ_6) were used as inputs to the ANFIS model to 9.608 when using root mean square (rms) and variance (σ^2) as input features to the model. The training remains constant after 170 epochs which means no improvement occurs after this epoch. The tuning trials of input features selection for the ANFIS system are highlighted in Table 5. The final trained fuzzy inference system (FIS) for predicting the fault types is illustrated in Figure 9.

5.2. Neural-Networks-Based Fault-Diagnosis System

A neural network system (i.e., nntool in Matlab 7.0) can be considered as a parameterized nonlinear map. However, in this study, the neural network parameters (i.e., Root mean square and Variance) have been selected as inputs, with failure code as the output.

First, the training data was used to find the appropriate network between the input and the target (desired output) to realize the actual output. The error between each pair was computed and the overall training error was determined.

A multi-layer forward neural network (ANN) is used for the computation. This network is also a cascade–forward backpropagation network. The characteristic features of time domain signals of the system with normal and faulty conditions have been used as inputs to this ANN structure, including input, hidden and output layers. The input layer contains for selected features from the time domain. The output layer contains of nodes indicating the fault type code. The final neural network used in this study consists of the input layer, one hidden layer and the output layer. The input layer has nodes representing the features extracted from the measured vibration signals. The number of neurons in the first hidden layer was 20. The number of output nodes is only one. The ANN was trained and implemented using the MATLAB neural network toolbox using backpropagation with Levenberg–Marquardt algorithm. For training, maximum iteration number (epoch) of 6000 was used. The initial weights and biases of the network were generated automatically by the program.

All the five statistical input features (i.e., rms, σ^2, γ_3, γ_4, and γ_6) were used for training and testing the neural network. The combination of root mean square and variance lead to the best result in testing and training. The training error was reduced from 29.1715 for kurtosis and six momentum feature to .04205 for root mean square and variance. The tuning trials of input features selection for the neural-networks system are highlighted in Table 6. Figure 10 shows the training and testing curves of the root mean square and variance model with performance equal to 0.92 that indicates the behavior of the network.

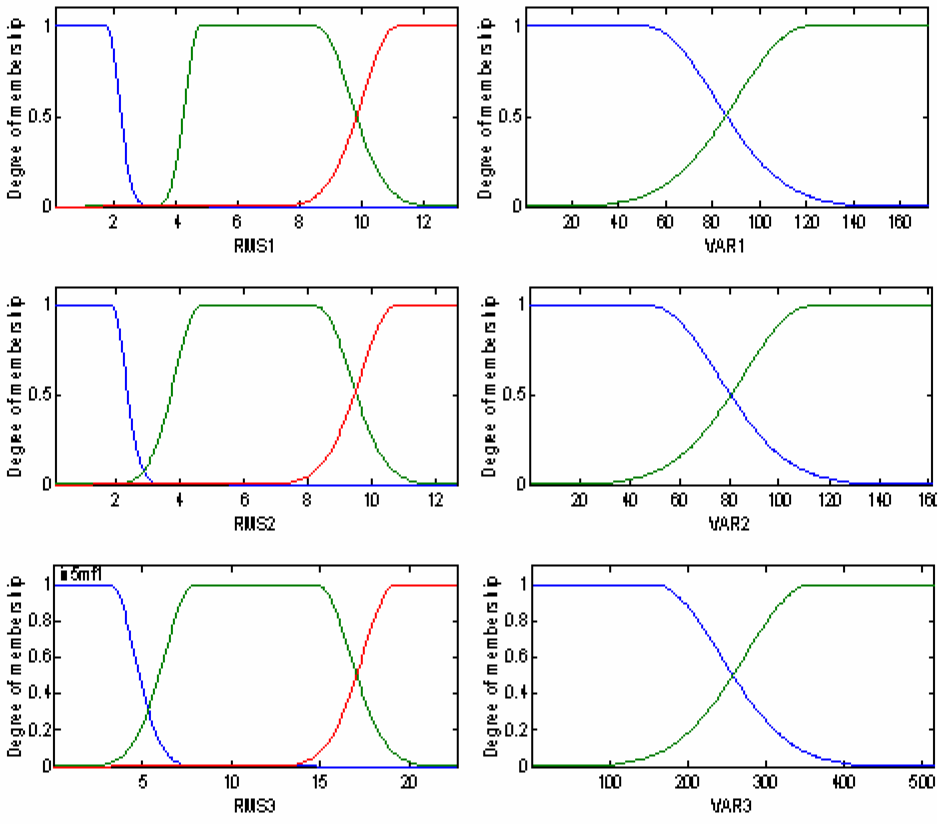

Figure 8. Final membership functions (MF) for input features.

Table 5. Tuning trials of input features (ANFIS)

MF type (Trapezoidal) , number of MFs =3 4 3 4 3 4 , Epochs =5					
	rms	σ^2	γ_3	γ_4	γ_6
rms		Train error =9.608 Test error=15.329	Train error=.0902 Test error r=34.84	Train error=2.044 Test error =99.719	Train error=18.737 Test error =72.28
σ^2			Trainers=37.169 Test error =73.328	Train error=30.61 Test error =31.62	Train error=33.236 Test error =31.108
γ_3				Trainers=21.146 Test error=684.99	Train error=35.084 Test error =53.781
γ_4					Train error=44.7159 Test error =44.27
γ_6					

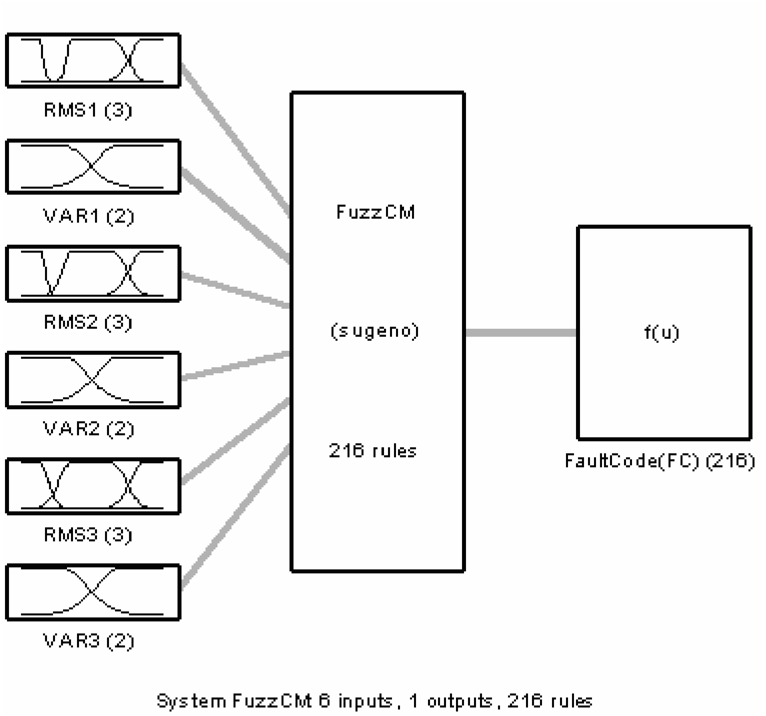

Figure 9. The final fuzzy inference system (FIS) for faults prediction.

Table 6. Tuning trials of input features selection (Neural Networks).

	rms	σ^2	γ_3	γ_4	γ_6
rms		Train error=.04205 Test error=1.056	Train error=14.695 Test error1=42.414	Train error=21.752 Test error1=41.522	Train error=24.99 Test error=45.457
σ^2			Train error=21.698 Test error1=14.112	Train error=1.2286 Test error1=1.5624	Train error=20.398 Testerror1=21.041
γ_3				Train error=27.137 Test error1=57.250	Train error=24.745 Test error1=44.79
γ_4					Train error=29.171 Test error=34.783
γ_6					

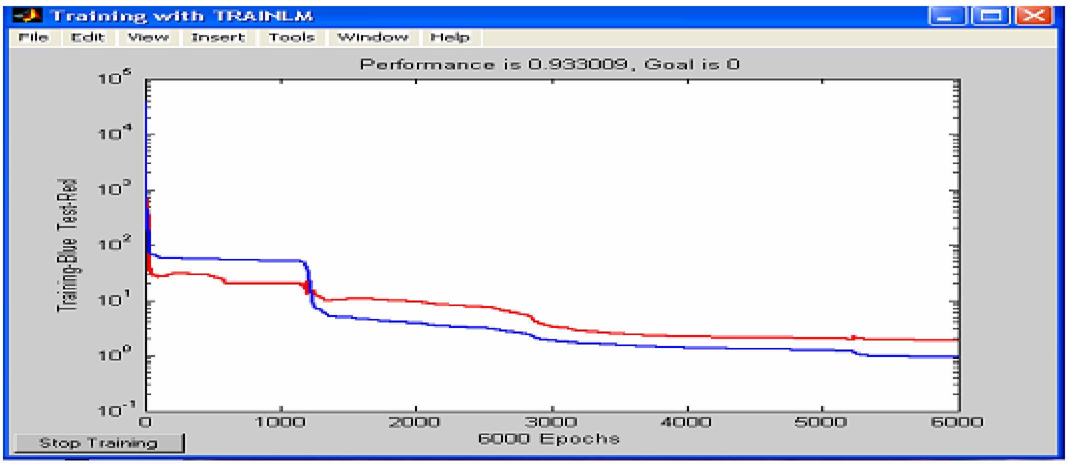

Figure 10. Training and testing of the neural network with inputs (rms and σ^2).

5.3. Models Validation

The ANFIS and neural networks prediction models for machine's faults were validated by selecting a certain number of data points (i.e., 19 points), different from the other 73 points used for ANFIS and neural nets training. Each validation data point (i.e., rms and σ^2) in the three bins, as given in Table 7, was fed into the system, and then the predicted fault type code (i.e., FC) were computed with the actual values of FC. The average percent errors in the ANFIS fault prediction is 3%, achieving a satisfactory accuracy of prediction of 97% as illustrated in Table 7, and the percent error in the Neural nets fault diagnosis is 0.8%, achieving a much better accuracy (i.e., 99.2% as given in Table 8) than the ANFIS prediction system. Table 8 illustrates that the neural networks-predicted values are a close match of the actual ones.

Table 7. Validation table for the ANFIS fault prediction system

(rms, σ^2), ANFIS, Time-domain								
rms1	$\sigma^2 1$	rms2	$\sigma^2 2$	rms3	$\sigma^2 3$	Actual Fault code	Predicted Fault code	Error%
13.15	173.73	12.63	160.30	22.77	520.75	300	300	0
6.58	43.51	6.72	45.32	6.56	43.23	100	107	7
1.95	3.84	1.92	3.69	1.86	3.46	200	200	0
1.98	3.93	1.91	3.68	1.87	3.52	200	200	0
1.02	1.05	1.08	1.17	1.03	1.07	200	200	0
3.97	15.85	3.83	14.75	3.93	15.54	300	287	4.3
1.80	3.26	1.78	3.17	1.95	3.81	200	200	0
6.50	42.46	6.72	45.32	6.64	44.33	100	112	12
1.45	2.10	1.42	2.03	1.31	1.73	200	200	0
1.37	1.88	1.43	2.06	1.41	1.99	200	200	0
7.65	57.42	7.69	59.48	7.59	57.79	100	105	5
5.87	34.66	4.90	24.12	5.55	30.93	100	92	8
4.09	16.86	3.90	15.31	4.48	20.13	300	270	10
13.15	173.73	12.63	160.30	22.77	520.75	300	300	0
3.34	11.19	3.54	12.59	4.08	16.74	300	317	5.7
3.92	15.42	3.94	15.56	4.15	17.28	300	287	4.3
1.87	3.53	1.76	3.09	1.76	3.11	200	200	0
1.41	2.00	1.43	2.04	1.44	2.08	200	200	0
2.69	7.30	2.06	4.26	2.06	4.26	200	201	0.5
Average Percent Error 3 %								

Table 8. Validation table for the neural-networks fault prediction system

(rms, σ^2), Neural Nets, Time-domain								
rms1	$\sigma^2 1$	rms2	$\sigma^2 2$	rms3	$\sigma^2 3$	Actual Fault code	Predicted Fault code	Error%
13.15	173.73	12.63	160.30	22.77	520.75	300	300.29	0.10
6.58	43.51	6.72	45.32	6.56	43.23	100	102.65	2.65
1.95	3.84	1.92	3.69	1.86	3.46	200	199.47	0.27
1.98	3.93	1.91	3.68	1.87	3.52	200	199.38	0.31
1.02	1.05	1.08	1.17	1.03	1.07	200	200.61	0.31
3.97	15.85	3.83	14.75	3.93	15.54	300	299.56	0.15
1.80	3.26	1.78	3.17	1.95	3.81	200	202.20	1.10
6.50	42.46	6.72	45.32	6.64	44.33	100	97.70	2.30
1.45	2.10	1.42	2.03	1.31	1.73	200	199.31	0.35
1.37	1.88	1.43	2.06	1.41	1.99	200	199.70	0.15
7.65	57.42	7.69	59.48	7.59	57.79	100	103.30	3.30
5.87	34.66	4.90	24.12	5.55	30.93	100	102.50	2.50
4.09	16.86	3.90	15.31	4.48	20.13	300	300.28	0.09
13.15	173.73	12.63	160.30	22.77	520.75	300	299.10	0.30
3.34	11.19	3.54	12.59	4.08	16.74	300	300.00	0.00
3.92	15.42	3.94	15.56	4.15	17.28	300	299.00	0.33
1.87	3.53	1.76	3.09	1.76	3.11	200	200.00	0.00
1.41	2.00	1.43	2.04	1.44	2.08	200	199.20	0.40
2.69	7.30	2.06	4.26	2.06	4.26	200	199.80	0.10
Average Percent Error 0.8 %								

6. Conclusions

An adaptive neuro-fuzzy system and a neural network are applied to predict the fault types of a mechanical system (carnallite surge tank pump). The time domain features (rms and σ^2) were used as inputs to ANFIS as well as neural nets to predict machine's fault type.

The following conclusions can be drawn from this study:

- The average percent error predicted by ANFIS with the trapezoidal membership function in the axial direction is only 5 %, achieving an accuracy of 95% using time-domain features.
- The average percent error predicted by neural-network with cascade network type in the axial direction is only 0.7 %, achieving an accuracy of 99.3 % using time-domain features.
- Analysis of time domain in both ANFIS and NN shows that, the most significant group of vibration signals and the characteristic features were root mean square and variance.
- Artificial neural networks (ANN) have potential applications in automated detection and diagnosis of machine conditions. Many of the ANNs for machine condition monitoring used the preprocessed frequency-domain features of the measured vibration signals
- ANFIS technique in parallel with time-based analysis can be used to predict and diagnose the machine's faults and failures. It is believed that this approach can be applied to identify other maintenance-related parameters.
- Neural networks, fuzzy logic and Neuro-fuzzy systems have an inherent shortcoming that they need to be retrained for different process parameters.
- A future work may be focused on constructing a real-time condition monitoring system by implementing an adaptive self-learning intelligent system for predicting and diagnosing machine's faults online.

References

[1] C. Mechefske, "Hydro electric Generating Unit MCM: A Comprehensive System". The Journal of Non-Destructive Testing and Condition Monitoring, Vol. 36, 1994, 238-245.

[2] Y. Peng, "Intelligent Condition Monitoring Using Fuzzy Inductive Learning". Journal of Intelligent Manufacturing, Vol. 15, 2004, 373-380.

[3] O. Castillo, P. Melin, "A Hybrid Fuzzy Fractal Approach for Time Series Analysis and Plant Monitoring". International Journal of Intelligent Systems, Vol. 17, 2002, 751-765.

[4] L. Tong, Y. Liang, "Forecasting Field Failure Data for Repairable Systems Using Neural Networks and SARIMA Model". International Journal of Quality and Reliability Management, Vol. 22, No. 4, 2005, 410-420.

[5] F. Kong, R. Chen, "A Combined Method for Triplex Pump Fault Diagnosis Based on Wavelet Transform, Fuzzy Logic, and Neuro-Networks". Mechanical Systems and Signal Processing, Vol. 18, 2004, 161-168.

[6] L. Zadeh, "Outline of a New Approach to Analysis of Complex Systems and Decision Process". IEEE Transactions on Systems, Man, and Cybernetics, 1973, 28-44.

[7] P. Wasserman, T. Schvartz, "Neural Networks, Part 2: What are they and why is everybody so interested now?". IEEE Expert, Vol. 8, 1998, 10-15.

[8] T. Fukuda, T. Shibata, "Theory and Applications for Industrial Control Systems". *IEEE* Transactions on Industrial Electronics, Vol. 39, No. 6, 1992, 472-489.

[9] C. Klimascauskar, "Neural Networks: An Engineering Perspective". IEEE Communications Magazines, Vol. 15, 1992, 50-53.

[10] British Standard Institution, BS3811, "British Standard Glossary of Maintenance Management Terms in Terotechnology".

[11] R. Yam, P. Tse, L. Li, P. Tu, "Intelligent Predictive Decision Support System for Condition-Based Maintenance". International Journal of Advanced Manufacturing Technology, Vol. 17, 2001, 383-391.

[12] A. Borden, "Designing and Maintaining Decision-making Processes". AGARD Conference Proceedings, Paris, France, 1993.

[13] M. Liberatore, A. Stylianou, "Using Knowledge-based Systems for Strategic Market Assessment". Information and Management, Vol. 27, No. 4, 1994, 221-232.

[14] T. Saaty, Multi-criteria Decision Making The Analytic Hierarchy Process. RWS Publication, 1990.

[15] M. Davies, "A Multi-criteria Decision Model Application for Managing Group Decisions". Journal of the Operational Research Society, Vol. 45, No. 1, 1994, 47-58.

[16] C. Kocourek, "A Petri-net based Design Decision Support System". Proceedings of the IASTED International Conference, Applied Modeling and Simulation, Vancouver, BC, Canada, 1993, 108-114.

[17] A. Hurson, S. Pakzad, B. Lin, "Automated Knowledge Acquisition in a Neural Network Based Decision Support System for Incomplete Database System". Microcomputers in Civil Engineering, Vol. 9, No. 2, 1994.

[18] T. Quah, C. Tan, T. Srinivasan, "A Fuzzy-neural-network Decision Support Systems for Bond Trading". Proceedings of International Symposium on Artificial Intelligence, The Artificial Intelligence Technology Transfer Conference, Monterrey, Mexico, 1993.

[19] A. Ishikawa, "The New Fuzzy Delphi Methods: Economization of GDS (group decision support". Proceedings of the 26th Hawaii International Conference on System Sciences, Wailea, HI, USA, 1993.

[20] C. Schrunder, J. Gallertly, J. Bicheno, "A Fuzzy Knowledge-based Decision Support Tool for Production Operations Management". Expert Systems, Vol. 11, No. 1, 1994, 3-11.

[21] M. Rogers, K. Hrovat, T. Reckart, M. Moskwitz, K. Mcpherson, "Accelerometer data analysis and Presentation Techniques". Technical Report No. 138ptN97, NASA, Lewis Research Center, Cleveland, Ohio, 1997

[22] B. Kosko, Neural Networks and Fuzzy systems: A Dynamical Systems Approach to Machine Intelligence. Prentice-Hall, Englewood Cliffs, NJ, 1992.

[23] J. Jang, 1993 "ANFIS: adaptive-network-based fuzzy inference system". IEEE Transactions on Systems, Man and Cybernetics, Vol. 23, 1993, 665-685.