Journal on Intelligent Electronic Systems, Vol.1, No.1, November 2007

Sensor Fault Diagnosis in State Feedback Systems using Artificial Neural Networks

V.Manikandan[1], K.Ramakrishnan[2], N.Devarajan[3], C.K.Babu[4], R.VenkatateswaraBhupathi[5]

[1,4]Dept. of Electrical and Electronics Engineering, Coimbatore Institute of Technology, Coimbatore - 641 014.
[2] Dept. of Electrical and Electronics Engineering, Pondichery Engineering College, Pondichery - 605 014.
[3] Dept. of Electrical and Electronics Engineering, Government College of Technology, Coimbatore - 641 013.
[5] Dept. of Computer Science and Engineering, VLB Janakiammal College of Engineering and Technology, Coimbatore-641042.
E-mail : babu.ck05@gmail.com, mani_sr2006@yahoo.co.in, ramss_k@yahoo.co.in

Abstract

A simulation before test method for fault diagnosis and classification towards sensor fault in linear time invariant state feed back system is presented in this paper. The novelty of the approach lies in associating with each state feedback gain factor a scalar , which is defined as the sensor healthiness factor. This scalar is made to vary from 1 (no fault condition) to 0 (full fault condition) in predetermined steps. The intermediate values of portray the deterioration modes of the sensor. The Integral Square Error (ISE) criterion is employed for extracting the signature of the fault and the classification is done using Artificial Neural Network (ANN) classifier. The proposed diagnosis approach is applied to a dc motor system to validate the effectiveness of the technique.program inspections, static & dynamic analysis and V&V techniques

Key words : Neural Network, Integral square error, fault diagnosis

I. INTRODUCTION

Fault detection techniques are mostly based on construction of a model of the system (Isermann *et al.*, 1997).A second way is to construct an observer capable of estimating the trends some internal variables of the process (Patton *et al.*, 1997). Alternatively, an estimate of some process parameters can be carried out (Hofling *et al.*, 1994). Whichever method is used, the estimated values must be compared with the actual ones in order to obtain information on the state of the system, and eventually detect the occurrence of a fault (Rizzo *et al.*, 2002). Recent work on fault detection tends to deal with the intrinsic nonlinear nature of systems, introducing nonlinear tools for modeling and fault detection, especially those based on soft computing, which allows both expert knowledge stored in the input-output data (Fortuna *et al.*, 2001) to be exploited. Neural networks are generally used to build nonlinear models or nonlinear observers, thus substituting their linear correspondent in previous approaches. Significant work has been carried out recently by adopting this strategy (Polycarpou *et al.*, June 1995, Borairi *et al.*, July 1996, Alessandri *et al.*, June 1997, Vemuri *et al.*, April 1998, Demetriou *et al.*, Nov 1998 and Maki *et al.*, Nov 1997, Marcu *et al.*, Oct 1997, Nauck *et al.*, 1997). The general structure of the fault diagnosis system is shown in the Fig.1. (Toscano *et al.*, 2003). There are basically two levels system level or level 0, whose role is mainly to generate control law in order to ensure correct performance of the closed loop system and a supervision level or level 1, whose role is decision making from the information generated by level 0. The level 1 consists of an Observation function, , a Classification function, and a decision making function, . The role of the observation function is to generate, from the measures provided by the level 0, a signature, x allowing us to characterize the possible faults, which might occur on the equipment. The signature x generated by the observation function is then applied to the classification function, which will allow the recognition of operating modes of the process.

The decision making function, allows us to act on the level 0 in accordance with the operating modes recognized by the classification function. It could be parametric (K) adaptation of the control law to preserve the performance of the supervised system or modification of the system operating point in order to meet the production objectives or an emergency stop procedure if the operating point is hazardous to human operator/equipment.

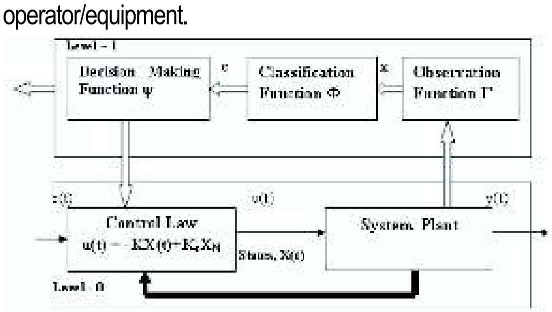

Fig.1. General fault diagnosis system

II. PROPOSED DIAGNOSIS APPROACH

Consider the LTI Single Input Single Output (SISO) system described by

$$\dot{X}(t) = AX(t) - Bu(t)$$
$$y(t) = CX(t) \qquad (1)$$

Where A R^{nxn} and is the system matrix, B R^{nx1} is the input matrix and C R^{1xn} is the output matrix. The state vector X(t) or simply X is an n vector; u(t) is the control effort and y(t) is the system output. If the pair (A, B) is controllable i.e., if the following condition is met,

$$\text{rank}\begin{bmatrix} B & AB & A^2B & . & . & A^r & B \end{bmatrix} = n \qquad (2)$$

Then state feedback control for arbitrary pole placement is possible. Assuming that all the state variables are available for feedback, the control effort is given by

$$u(t) = -KX(t) + k_e x_N \qquad (3)$$

Where

$$x_N = \int e(t)dt \qquad (4)$$

Here e(t) is r(t)-y(t), where r(t) is the reference input and y(t) is the output; K the state feedback gain factor given by

$$K = \begin{bmatrix} K_1 & K_2 & K_3 & . & . & K_n \end{bmatrix} \qquad (5)$$

where K_i s are the feedback gain factors. The matrix K can be designed either from the closed loop response specifications or optimally by LQR approach (Franklin *et al.*, 2002). The first portion of the control law (3) basically means that each state of the system say X_i is multiplied with the element K_i of K and summed up (i from 1 to n for n states) and fed back. This portion of the control law characterizes the transient performance of the closed loop system and the integral portion ensures that there is no offset in the steady state response in tracking r(t). The control law is given explicitly as

$$u(t) = -\sum_{i=1}^{n}\left[K_i X_i(t)\right] + K_e \int e(t)dt \qquad (6)$$

Since only sensors provide the state of the system and that these states are feedback with a gain of K, to analyze the faulty and deterioration modes of the sensor, we associate a scalar for each element of the feedback matrix (which in turn means that to each sensor we associate this scalar, as the performance indicative factor). That is, for diagnosing the quality of the sensor S_i (say a tachogenerator) that is giving X_i (say speed, a state of the system), we associate the scalar $_i$ with the corresponding state feedback gain factor, K_i. In other words, we have fixed $_i$ as the performance index for S_i. This scalar is known as the Sensor healthiness factor. This factor can vary in any steps from 1 to 0. A value of 1 for this factor means that the particular sensor is healthy and that it is transduting the actual quantity of the measurand. A value of zero means that the sensor has worn out completely and is not giving any output at all. Intermediate values portray the deterioration modes of the sensor.

The control law can now be written in the expanded form as

$$u(t) = -\begin{bmatrix}\alpha_1 K_1 & \alpha_2 K_2 & \alpha_3 K_3 & & \alpha_n K_n\end{bmatrix}\begin{bmatrix} X_1(t) \\ X_2(t) \\ X_3(t) \\ . \\ X_n(t)\end{bmatrix} + K_e \int e(t)dt \qquad (7)$$

Or

$$u(t) = -\alpha K X(t) + K_e \int e(t)dt \qquad (8)$$

Where $\alpha = \begin{bmatrix}\alpha_1 & \alpha_2 & . & . & \alpha_n\end{bmatrix}^T$ is the vector of the healthiness factors of all the sensors of the system. Now the closed loop equations of the system are given as:

$$\begin{bmatrix}\dot{X} \\ \dot{X}_N\end{bmatrix} = \begin{bmatrix} A\text{-}BK & BK_e \\ -C & 0\end{bmatrix}\begin{bmatrix} X \\ X_N\end{bmatrix} + \begin{bmatrix} 0 \\ 1\end{bmatrix} r$$

$$Y = \begin{bmatrix} -C & 0 \end{bmatrix}\begin{bmatrix} X \\ X_N\end{bmatrix} \qquad (9)$$

III. DESIGNING THE NEURAL NETWORK

Designing the neural network shown in Fig.2 involved choosing the following parameters: Number of inputs, number of output, number of hidden layers, and number of nodes/layers (Pradhan *et al.*, 2005). In our neural network design, specifications of the above parameters were as follows:

i) Inputs = 2

The number of inputs we fed into the system simultaneously, which was 2 for this design, these samples defined a particular pattern and was the basis of training neural network so that it could deliver desired results.

ii) Outputs = 2

The number of the output node was just 'two' because the only output needed was $_1$ and $_2$ .These two output conditions could be managed with two nodes.

iii) Hidden layer =1

Hidden Layer Nodes=8

The above two values were chosen on a hit and trial basis depending on the certain performance criteria, that is goal=1e-10. "Goal" here stands healthiness factor of the sensors. One reason to choose the BP technique was the ability to change the values of its weight in response to error.

- ? Training data . input set & corresponding output set
- ? Input . hidden layer.output
- ? Finds error . Output(target)-output(actual)

The network passed the derivative of the error back to the hidden layer, using an original weighted connection. Each hidden node then calculated the weighted sum of the back propagated errors to find its contribution to the known output errors. After each output layer and hidden node found the contribution, the node adjusted its weight to reduce the error.

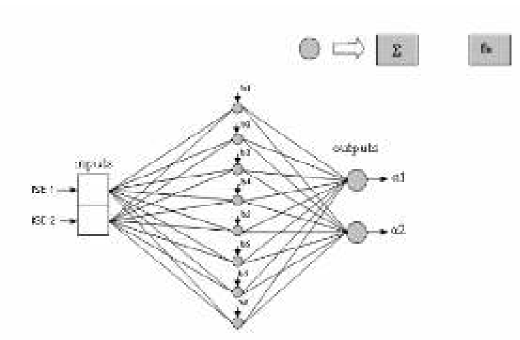

Fig.2. Designing the neural network

IV. EXTRACTION OF FAULT SIGNATURES

The schematic for estimating the fault residue is shown in the Fig.3.The fault residue extraction process assumes that all the states are available for the feedback. Unit step is used as the input excitation. Now the sensor healthiness factor of any one sensor (or more in case we are interested in multiple fault analysis) is varied in predetermined steps between 1 thro 0 and ISE of the error ($X \ X_f$) where X is the state dynamics of the system without fault and X_f is the state dynamics with fault is computed as the fault signature. Similar signatures for different sensors (individually or in combined fashion) are obtained and stored in a database for training the ANN Classifier. To simulate the actual working condition in this model based fault diagnosis approach, the system and control parameters are kept at their nominal values. By this, we enrich the database with a number of patterns conforming to a specific system operating condition thereby simplifying the classification process substantially.

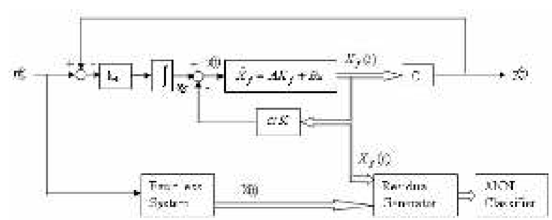

Fig.3. Extraction of fault signature

V. RESULTS ON A DC MOTOR SYSTEM

For a dc motor system with permanent magnet, the system descriptive equations (Katshiko Ogata, 2003) are:

$$\begin{bmatrix} \dot{X}_1(t) \\ \dot{X}_2(t) \end{bmatrix} = \begin{bmatrix} \frac{-R_a}{L_a} & \frac{-K_b}{L_a} \\ \frac{K_T}{J} & \frac{-f}{J} \end{bmatrix} \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix} + \begin{bmatrix} \frac{K_P}{L_a} \\ 0 \end{bmatrix} u(t) \quad (10)$$

$$y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix}$$

Where

$X_1(t)$ is the Armature Current in Amps.

$X_2(t)$ is the rotor speed in rad/sec^2.

R_a is the armature resistance in Ohms

L_a is the Armature inductance in Henries

J is the moment of inertia of the load in Kg.m^2

f is the viscous friction coefficient in Nm/rad/sec.

K_b is the back emf constant in V/rad/sec.

K_T is the Torque constant.

K_P is the Power interface gain.

For system parameters: $R_a = 1$, $L_a = 0.095$H, J = 0.02105 Kg.m^2, f = 0.01 Nm/rad/sec, $K_b = 0.02$V/rad/sec, $K_T = 0.1$, $K_P = 20$, the system equations are given by

$$\begin{bmatrix} \dot{X}_1(t) \\ \dot{X}_2(t) \end{bmatrix} = \begin{bmatrix} -10.5253 & -0.2105 \\ 4.7506 & -0.4751 \end{bmatrix} \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix} + \begin{bmatrix} 210.5253 \\ 0 \end{bmatrix} u(t) \quad (11)$$

$$y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix}$$

The system open loop poles are located at 10.4248 and 0.5756. The open loop system is stable but heavily over damped. The state feedback control is designed to give a closed loop response of 5% peak overshoot and 0.1sec peak time to unit step forcing function. These specifications give locations for closed loop poles at 30.9j31.4159 and third non-dominant closed loop pole is placed at 200. The control effort u (t) is given by

$$u(t) = -1.1913X_1(t) - 14.1799X_2(t) + 388.3066\int e(t)dt \quad (12)$$

VI. CASE STUDY

Now let us make one case study wherein the current sensor has deteriorated and is giving only 50% of the actual current ($_1$ = 0.5 and $_2$ = 1). Now the state feedback matrix is

$$K = [0.5K_1 \quad K_2] \quad (13)$$

Now ISE of (X-X_f) give the signature for this fault condition (Table I). The waveforms for X, X_f, X-X_f, for 50% fault in sensor are shown in Fig.4. Similar signatures are extracted for both the sensor (current as well as speed) under fault mode. The database will be as shown in the Table II. In the Table II, we have shown that the healthiness factor of the current sensor varies from 0.5 to 0.95 of its optimum value and for each value of $_1$, the other sensor varied from 0.5 to 1with system and control parameters kept at their nominal values. It is quite evident that the error reduces as the quality (depicted by the healthiness factor) of transduction of a sensor improves. All the signatures stored in the database are used for training the ANN whose effectiveness in classifying unknown pattern belonging to an appropriate class is well known and can be found through pattern classifier below.

Table 1. Integral Square Error for [0.5 1] fault condition

Sensor Healthiness Factor		Integral Square Error(ISE)		Fault ID
α_1	α_2	Current	Speed	
0.5	1	4.624	0.049	101

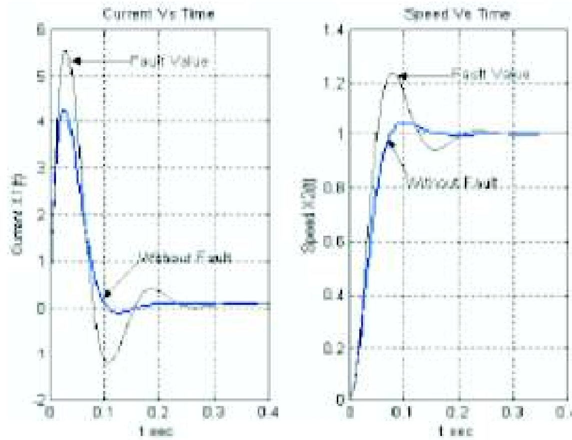

Fig. 4. a System states - current and speed - vs Time

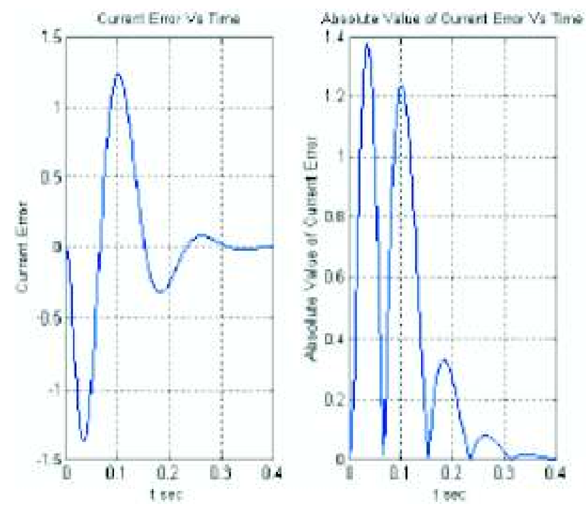

Fig. 4. B Error dynamics for current

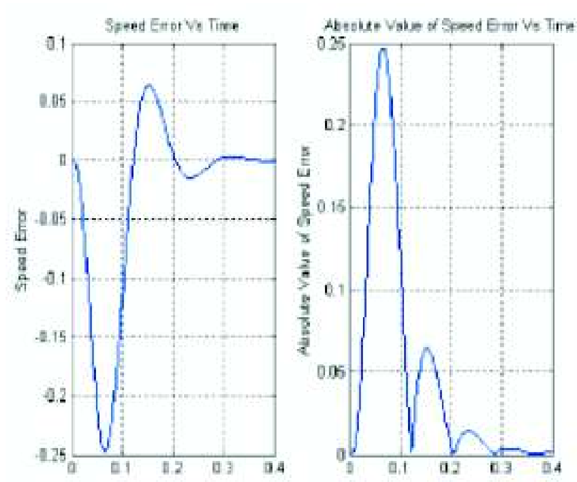

Fig. 4. C Error dynamics for Speed

Journal on Intelligent Electronic Systems, Vol.1, No.1, November 2007

Table 2. Integral Square Error (ISE) for other fault condition

Sensor Healthiness Factor		Integral Square Error(ISE)		Fault ID
α_1	α_2	Current	Speed	
0.5	0.5	55.470	0.628	1
0.5	0.55	37.737	0.449	2
0.5	0.6	26.598	0.326	3
0.5	0.65	19.294	0.239	4
0.5	0.7	14.362	0.176	5
0.5	0.75	10.969	0.13	6
0.5	0.8	8.614	0.097	7
0.5	0.85	6.978	0.073	8
0.5	0.9	5.854	0.059	9
0.5	0.95	5.103	0.051	10
0.55	0.5	48.404	0.611	11
0.55	0.55	32.822	0.436	12
0.55	0.6	22.928	0.315	13
0.55	0.65	16.41	0.229	14
0.55	0.7	12.01	0.166	15
0.55	0.75	8.998	0.12	16
0.55	0.8	6.927	0.087	17
0.55	0.85	5.513	0.064	18
0.55	0.9	4.566	0.049	19
0.55	0.95	3.958	0.042	20
0.6	0.5	43.353	0.603	21
0.6	0.55	29.216	0.428	22
0.6	0.6	20.175	0.307	23
0.6	0.65	14.204	0.22	24
0.6	0.7	10.18	0.157	25
0.6	0.75	7.441	0.111	26
0.6	0.8	5.578	0.078	27
0.6	0.85	4.326	0.055	28
0.6	0.9	3.512	0.041	29
0.6	0.95	3.014	0.034	30
0.65	0.5	39.679	0.600	31
0.65	0.55	26.541	0.425	32
0.65	0.6	18.095	0.303	33
0.65	0.65	12.509	0.214	34
0.65	0.7	8.752	0.151	35
0.65	0.75	6.209	0.104	36
0.65	0.8	4.497	0.071	37
0.65	0.85	3.366	0.048	38
0.65	0.9	2.65	0.033	39
0.65	0.95	2.236	0.026	40
0.7	0.5	37.013	0.603	41
0.7	0.55	24.573	0.425	42
0.7	0.6	16.541	0.301	43
0.7	0.65	11.227	0.211	44
0.7	0.7	7.652	0.146	45
0.7	0.75	5.247	0.099	46

0.7	0.8	3.641	0.065	47
0.7	0.85	2.597	0.042	48
0.7	0.9	1.953	0.027	49
0.7	0.95	1.599	0.02	50
0.75	0.5	35.124	0.612	51
0.75	0.55	23.166	0.43	52
0.75	0.6	15.415	0.302	53
0.75	0.65	10.276	0.21	54
0.75	0.7	6.829	0.143	55
0.75	0.75	4.515	0.095	56
0.75	0.8	2.981	0.060	57
0.75	0.85	1.995	0.037	58
0.75	0.9	1.4	0.022	59
0.75	0.95	1.09	0.014	60
0.8	0.5	33.856	0.63	61
0.8	0.55	22.219	0.438	62
0.8	0.6	14.648	0.306	63
0.8	0.65	9.619	0.217	64
0.8	0.7	6.245	0.143	65
0.8	0.75	3.985	0.093	66
0.8	0.8	2.493	0.057	67
0.8	0.85	1.542	0.033	68
0.8	0.9	0.978	0.018	69
0.8	0.95	0.695	0.01	70
0.85	0.5	33.098	0.643	71
0.85	0.55	21.661	0.45	72

0.85	0.6	14.189	0.313	73
0.85	0.65	9.215	0.216	74
0.85	0.7	5.874	0.144	75
0.85	0.75	3.636	0.093	76
0.85	0.8	2.162	0.056	77
0.85	0.85	1.225	0.031	78
0.85	0.9	0.675	0.015	79
0.85	0.95	0.404	0.006	80
0.9	0.5	32.77	0.665	81
0.9	0.55	21.435	0.465	82
0.9	0.6	13.999	0.324	83
0.9	0.65	9.033	0.222	84
0.9	0.7	5.691	0.148	85
0.9	0.75	3.449	0.094	86
0.9	0.8	1.971	0.056	87
0.9	0.85	1.032	0.03	88
0.9	0.9	0.481	0.013	89
0.9	0.95	0.211	0.004	90
0.95	0.5	32.807	0.693	91
0.95	0.55	21.497	0.484	92
0.95	0.6	14.046	0.337	93
0.95	0.65	9.051	0.231	94
0.95	0.7	5.678	0.154	95
0.95	0.75	3.409	0.098	96
0.95	0.8	1.908	0.058	97
0.95	0.85	0.952	0.03	98

0.95	0.9	0.387	0.013	99
0.95	0.95	0.107	0.003	100
0.5	1	4.624	0.049	101
0.55	1	3.601	0.04	102
0.6	1	2.749	0.032	103
0.65	1	2.041	0.025	104
0.7	1	1.458	0.019	105
0.75	1	0.986	0.013	106
0.8	1	0.616	0.009	107
0.85	1	0.338	0.005	108
0.9	1	0.147	0.002	109
0.95	1	0.036	0.001	110
1	1	0	0	111

VII. PATTERN CLASSIFIER

Since the test pattern for the DC motor system has two controlling inputs that is ANN has 2 neurons in the input layer and 2 neurons in the output layer. The 2 neurons in the output layer can classify all types of faults and will be sufficient for classifying total of 111 different faults. The number of neurons in the hidden single layer is 8. So the ANN structure boils down to 2:8:2.The pattern for a specific fault is generated by testing the system at all test conditions under permissible tolerances for other types of faults. The ANN is adaptively trained to update the weights and the bias by gradient descent method by the mean square error performance. The classifier structure for the circuit and the training pattern for 100 epochs are shown in the Fig.5. and Fig.6. respectively. For few randomly generated test patterns for the system, classifier results are shown in the Table III. The results agree well within the corresponding fault ID (Nauck et al., 1997).

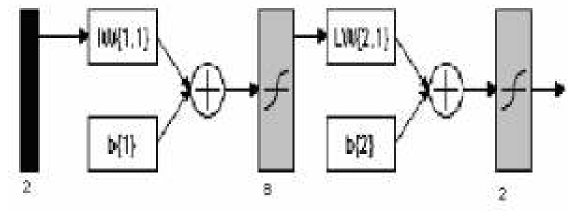

Fig. 5. Classifier for test circuit

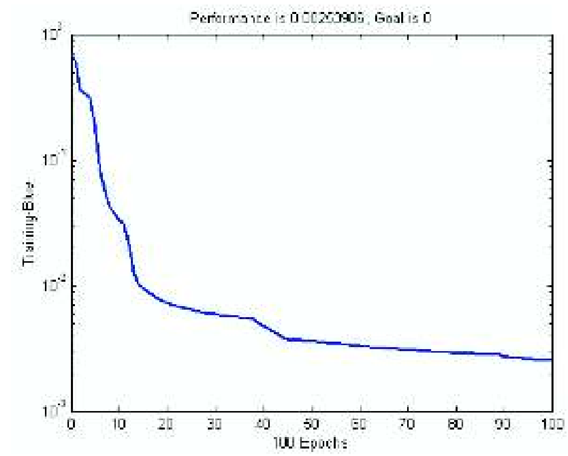

Fig. 6. Training pattern

Table 3. Few Results of Pattern classifier

Classifier input		Classifier output		Fault id
Integral square error		α_1	α_2	
current	speed			
2.597	0.042	0.7	0.85	48
37.737	0.449	0.5	0.55	2
15.415	0.302	0.75	0.6	53
43.353	0.603	0.6	0.5	21
33.856	0.63	0.8	0.5	61
4.624	0.049	0.5	1	101
0.387	0.013	0.95	0.9	99
12.01	0.166	0.55	0.7	15

0	0	1	1	111
14.189	0.313	0.85	0.6	73
6.245	0.143	0.8	0.7	65
14.046	0.337	0.95	0.6	93
5.578	0.078	0.6	0.8	27
55.470	0.628	0.5	0.5	1

VIII. CONCLUSION

A novel Simulation before Test approach towards sensor fault diagnosis in full state feedback system is proposed in this paper. Associating a sensor healthiness factor with each element of the feedback gain matrix and varying them from 1 thro 0, deteriorating modes of the sensors are analyzed. The performance criterion Integral Square Error (ISE) is used for fault signature extraction and ANN is employed for fault classification. The proposed approach for single sensor fault is illustrated through a dc motor system example with encouraging results.

REFERENCES

[1] Alessandri A and Parisini T, June 1997 "Model based fault diagnosis using non linear estimators: A neural approach," in Proc.Amer. Contr. Conf. vol. I5, pp. 2874-2878.

[2] Borairi M and Wang H, July 1996, "Actuator and sensor fault diagnosis of non linear dynamic systems via genetic neural networks and parameter estimation techniques," in Proc.IEEE Int .Conf.Contr.Applicat., vol.1, pp. 278-282.

[3] Demetriou M A and Polycarpou M M, Nov.1998, "Incipient fault diagnosis of dynamical systems using on line approximations," IEEE trans. Automat. Control, vol 43, pp.1612-1617.

[4] Fortuna L , Rizzotto G G, Lavorgna M, Nunnari G, Xibilia M G and Caponetto R, 2001, " Soft computing new trends and Applications," in advanced textbooks in control and signal processing. London, U.K., Springer Verlog.

[5] Franklin F Gene, David Powel J, Emami-Naeni A, 2002, 'Feedback Control of Dynamic Systems', Fourth Edition, Pearson Ed.

[6] Hofling T and Feufer T P, 1994, "Detection of additive and multiplicative faults-parity space vs. parameter estimation," in Proc .IFAC SAFEPROCESS Symp., Baden-Baden,Germany.

[7] Isermann R and Balle P, 1997, "Trends in application of model based fault detection and diagnosis of technical processes," Control Eng. Practice, vol.5, no.5, pp.709-719.

[8] Katshiko Ogata, 2003, 'Modern Control Engineering', Third Edition, PHI.

[9] Maki Y and Loparo K A, Nov1997, "A neural network approach to fault diagnosis in industrial processes," IEEE Trans. Contr. syst. Technol., vol.5, pp.529-541.

[10] Marcu T and Mirea L, Oct1997, "Robust detection and isolation of process faults using neural networks," IEEE Contr. Syst. Mag., vol.17, no.5, pp. 72-79.

[11] Nauck D and Kruse R, 1997, 'A Neuro-Fuzzy Method to learn Fuzzy Classification Rules from Data', Fuzzy Sets and Systems 89, pp. 277-288.

[12] Patton R J and Chen J, 1997, "Observer based fault detection and isolation: Robustness and applications", Control Eng. Practice, vol.5, no.5, pp.671-682.

[13] Polycarpou M M and Venmuri A T, June 1995, "Learning methodology for failure detection and accommodation," IEEE Control System Magazine., Pp.16-24.

[14] Pradhan A K et al, 2005, "Implementation of an intelligent relay using nios processor," Nios soft core embedded processor design contest outstanding project collections, pp. 38-45.

[15] Rizzo Alessandro and Gabriella Xibilia M, May 2002, "An innovative intelligent system for sensor validation in tokomak machines," IEEE Trans.Control system technology, vol.10, no.3.

[16] Toscano R and Lyonnet P, 2003, 'Diagnosis of Industrial Systems by Fuzzy Classifier', ISA Transactions 42, pp.327-335.

[17] Vemuri A T, Polycarpou M M and Diakourtis S A, April 1998, "Neural network based fault detection in robotic manipulators, "IEEE trans. Robot Automat. Contr., vol.14, pp. 342-348.

[18] MATLAB 6p5 Source Codes.