International Journal of Soft Computing 3 (5): 344-354, 2008
ISSN: 1816-9503

MANFIS Observer Based Sensor Fault Detection and Identification in Interacting Level Process with NN Based Threshold Generator

[1]S. Nagarajan, [3]J. Shanmugam and [2]T.R. Rangaswamy
[1]Department of Instrumentation and Control Engineering,
[2]Department of Information Technology, BSA Crescent Engineering College,
Chennai, Tamilnadu, India 600048
[2]Department of Electronics and Communication, Tagore Engineering College,
Chennai, Tamilnadu, India 600048

Abstract: This study presents the design of Adaptive Neuro-Fuzzy Observer based sensor fault detection in a three-tank interacting level process. Three pairs of observers estimate the three system states. These Observers are designed with Multiple Adaptive Neuro-Fuzzy Inference System (MANFIS) that uses a neural network to fix optimal shape and parameters for the membership functions and effective rule base for the fuzzy system. Fault detection is performed by estimating the states of the level process and comparing them with measured values. A fault is signaled when the difference between the estimated and measured values crosses a threshold value. Decision functions are built from estimation errors to detect the fault. If any failure is identified, the control law is modified accordingly using the estimated value replacing the failed sensor output. In this research, MANFIS observer based fault detection is designed and simulated. Since, the threshold value can be different for different set point, a Neural Network (NN) based threshold generator is designed to give best threshold values for fault detection. The individual failures of three level sensors are considered for various set points and the results are discussed. The results show that the system is able to detect any sensor failure for any set point and to control the level in interacting tanks perfectly under failure situations.

Key words: Adaptive neuro-fuzzy inference system, estimation error, fault detection and identification, observers, multiple adaptive neuro-fuzzy inference system, neural networks, threshold generation

INTRODUCTION

Many systems are relied upon to provide safe and reliable operation for long periods of time. Unfortunately, all these system components are subject to manufacturing defects, wear and tear and other causes of performance degradations. In systems like a 3 tank interacting level process in which the level is to be maintained at the desired level in the tank, the system may become unstable under sensor failures. Therefore, it is important for the control system to be able to detect and compensate for fault conditions online and in real time. Fault Detection and Identification (FDI) aims at making the system stable and retain acceptable performance under the system faults. The purpose of FDI is to detect, identify and reconfigure for any type of failures that may occur at any time.

The sensor failure detection and identification has been considered as an important issue, particularly when measurements from sensors are used in the feedback loop of a control law. Since the control law uses the sensor feedback to establish the current states of the process, the control with failed sensors can lead to an imperfect control or closed loop instability. Model based fault detection techniques are based on observers (Clark *et al*., 1975), state estimating filters (Alan, 1976) or Parameter Estimators (Isermann, 1984).

In a multivariable process, failure of any one of the measuring elements can make a total failure of the control system. If the state feedback with state space model of the process is used, the control system will become very simpler and safe. Observers can be designed to estimate the states if some state variables of the system cannot be measured for feedback, in some cases. This feature of

Corresponding Author: S. Nagarajan, Department of Instrumentation and Control Engineering, BSA Crescent Engineering College, Chennai, Tamilnadu, India 600048

observer leads to the concept of fault detection by which the measurements from the perfect sensors can estimate the state of failed sensor. The estimation error that is the difference between the sensor output and the corresponding state given by the observer, gives information regarding the failed sensor. A dedicated observer scheme (Clark, 1978) can be introduced in which each sensor of interest drives an observer to perform a complete state estimation. In an alternative version, generalized observer scheme (Frank, 1990), an estimator dedicated to a certain sensor is driven by all outputs except that of the respective one. In both cases, if the actual system is non-linear, it is a matter of degree of non-linearity that determines if this method will be successful in the detection of a failure.

With the development of neural networks and fuzzy systems, observers are designed using these techniques since they do not need any mathematical model and can accommodate non-linearities (Napolitano *et al*., 1995). Observers can be modeled using fuzzy system. A straightforward approach (Xiao-Jun Ma and Zeng-Qi Sun, 1998) is to assume a certain shape for the membership functions. The effectiveness of the fuzzy models representing nonlinear input-output relationships depends on the fuzzy partition of the input-output spaces. Optimal fuzzy observer (Mohanlal and Kaimal, 2004) can be developed based on Takagi-Sugeno fuzzy model. Therefore, the tuning of membership functions becomes an important issue in fuzzy modeling. Since this tuning task can be viewed as an optimization problem, neural networks can be used to solve this problem. The shape of the membership functions that depends on different parameters and a specification of the rules including a preliminary definition of the corresponding membership functions can be learned by a neural network with a set of training data in the form of correct input-output relationships. The Adaptive neuro-fuzzy system, which is a neural network, fixes the optimal shape and parameters for the membership functions and effective rule base for the fuzzy system for observer modeling.

In this research, the levels in the three tanks of interacting level process are measured as states and used to control the level in the third tank through state feedback. Three pairs of Adaptive neuro-fuzzy observers are designed with sensor outputs as inputs. Using these three designed pairs of observers, it is possible to estimate all the states of the process and they can be fedback for control. The fault detection and isolation decision logic detects any fault that occurs and identifies the failed sensor. This fault detection and identification is followed by reconfiguration of the control law that performs the fault tolerant control.

Since, the residual and the decision function may change with the changing control inputs under various process conditions, Adaptive threshold can be fixed analytically which varies dependently on the control input (Clark, 1989). A robust threshold selector (Emani-Naeini *et al*., 1988) was also suggested in which the control activity and the system operation states influence the variable threshold. Fixing threshold values are very important since the inevitable time delay in fault detection, can cause stability problems. Higher values of the threshold for decision functions may delay the fault detection and cause stability problem whereas low values of threshold may miss the fault. In this research, the threshold values are fixed for the values of decision functions in such a way that their values are greater than the values of respective decision functions under no failure condition and they are less than the values of respective decision functions under failure conditions.

MATHEMATICAL MODEL

The three-tank interacting level process is shown in Fig. 1. The non-linear equations describing the open loop dynamics of this process are given by:

$$\frac{dh_1(t)}{dt} = -\frac{\beta_{12}\alpha_{12}}{A_1}\sqrt{2g(h_1(t)-h_2(t))} + \frac{k_1}{A_1}u_1$$

$$\frac{dh_2(t)}{dt} = \frac{\beta_{12}\alpha_{12}}{A_2}\sqrt{2g(h_1(t)-h_2(t))} - \frac{\beta_{23}\alpha_{23}}{A_2}\sqrt{2g(h_2(t)-h_3(t))}$$

$$\frac{dh_3(t)}{dt} = \frac{\beta_{23}\alpha_{23}}{A_3}\sqrt{2g(h_2(t)-h_3(t))} - \frac{\beta_3\alpha_3}{A_3}\sqrt{2gh_3(t)} + \frac{k_2}{A_3}u_2 \quad (1)$$

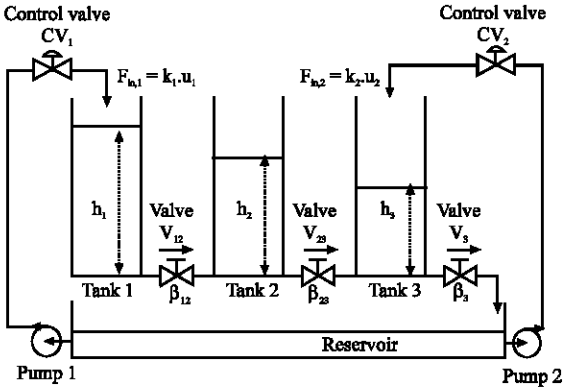

Fig. 1: Three tank interacting level process

where, h_i is the level in tank i (cm), u_i is the control input to the control valves CV_i (%), A_i is the cross section area of tank i (cm^2), α_{ij} is the cross section area of the pipe connecting tank i and tank j (cm^2), α_3 is the cross section area of outlet of tank 3 (cm^2), β_{ij} is the valve ratio between tank i and tank j, β_3 is the valve ratio of outlet of tank 3, k_i is the gain of valve CV_i (cm^3/%s) and g is the gravity (cm sec^{-2}).

This process is linearised about an operating point and the linear state space model can be represented by:

$$X = AX+BU \qquad (2)$$
$$Y = CX$$

with state vector X and control input U as:

$$X^T = [h_1 \quad h_2 \quad h_3] \qquad (3)$$
$$U^T = [u_1 \quad u_2 \quad 1]$$

The matrix C is 3×3 identity matrix whereas matrices A and B in (2) are given by:

$$A = \begin{bmatrix} -\frac{1}{T_{12}} & \frac{1}{T_{12}} & 0 \\ \frac{1}{T'_{12}} & -\left(\frac{1}{T'_{12}}+\frac{1}{T_{23}}\right) & \frac{1}{T_{23}} \\ 0 & \frac{1}{T'_{23}} & -\left(\frac{1}{T'_{23}}+\frac{1}{T_3}\right) \end{bmatrix} \qquad (4)$$

$$B = \begin{bmatrix} \frac{k_1}{A_1} & 0 & -\frac{1}{T_{12}}(H_1-H_2) \\ 0 & 0 & \frac{1}{T'_{12}}H_1-\left(\frac{1}{T'_{12}}+\frac{1}{T_{23}}\right)+\frac{1}{T_{23}}H_3 \\ 0 & \frac{k_2}{A_3} & \frac{1}{T'_{23}}H_2-\left(\frac{1}{T'_{23}}+\frac{1}{T_3}\right)H_3 \end{bmatrix} \qquad (5)$$

Where:

$$T_{12} = \frac{A_1}{\beta_{12}a_{12}}\sqrt{\frac{2(H_1-H_2)}{g}}, T'_{12} = \frac{A_2}{\beta_{12}a_{12}}\sqrt{\frac{2(H_1-H_2)}{g}},$$
$$T_{23} = \frac{A_2}{\beta_{23}a_{23}}\sqrt{\frac{2(H_2-H_3)}{g}}, T'_{23} = \frac{A_3}{\beta_{23}a_{23}}\sqrt{\frac{2(H_2-H_3)}{g}}$$
$$T_3 = \frac{A_3}{\beta_3 a_3}\sqrt{\frac{2H_3}{g}},$$

and H_1, H_2 and H_3 are levels in tank 1, tank 2 and tank 3, respectively at the operating point.

Table 1: Parameters of the process

A_1, A_2, A_3 (cm^2)	α_{12}, α_{23}, α_3 (cm^2)	β_{12}	β_{23}	β_3	k_1, k_2 (cm^3/%s)
615.7522	1.2272	0.9	0.8	0.3	8

Table 2: Operating point of the process

H_1 (cm)	H_2 (cm)	H_3 (cm)	u_1 (%)	u_2 (%)
55.66	50.6	44.45	80	10

The parameters and the operating point of the process are given in Table 1 and 2, respectively.

The control system is designed based on this model using the state feedback with the objective to control the level h_3 in tank 3 at the desired value. Thus the control system will require information from all the three sensors directly while they are normal and through fault detection while they fail.

STATE FEEDBACK

In control system design by pole placement technique, the states are used for feedback to achieve desired closed loop poles. The advantage in this system is that the closed loop poles may be placed at any desired locations by means of state feedback through an appropriate state feed back gain matrix K to achieve a perfect and smooth control. The control is done by controlling u_1 only with a fixed u_2 in this research.

Using State feedback,

$$u_i = -KX \qquad (6)$$

Then

$$\frac{dX}{dt} = (A-BK)\,X \qquad (7)$$

The characteristic polynomial of the system with state feedback is given by:

$$[SI-(A-BK)] \qquad (8)$$

with

$$K = [K_1 \quad K_2 \quad K_3] \qquad (9)$$

The Coefficient of the polynomial in Eq. (9) is a function of K_1, K_2 and K_3. The desired characteristic polynomial is found by choosing desired closed loop poles for best control. By equating the coefficients of actual characteristic polynomial and the desired characteristic polynomial, the state feedback gain K is determined (Stefani *et al*., 2002). For this process, the state feedback gain K is determined and the values of K_1, K_2 and K_3 are 3.7707, 3.5659 and 3.1659, respectively.

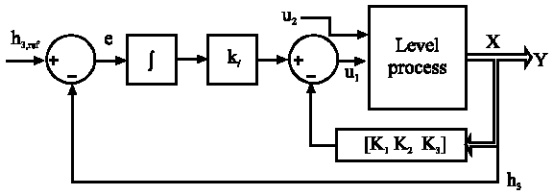

Fig. 2: State feedback with Integral control

An integral control in the path in series with the process is provided to achieve zero steady state error for given reference input $h_{3,ref}$. The gain K_I for integral control is 0.2. The block diagram of state feedback with integral control is shown in Fig. 2.

ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM

A neuro-fuzzy system is a combination of an Artificial Neural Network (ANN) and a Fuzzy Inference System (FIS) in such a way that neural network learning algorithms are used to determine the parameters of FIS. A FIS can utilize human expertise for storing its essential components in a rule base and a database and perform fuzzy reasoning to infer the overall output value. For building a fuzzy inference system, the fuzzy sets, fuzzy operations and the knowledge base should be specified. For building an Artificial Neural Network (ANN), it is necessary to specify the learning algorithm and the architecture. The learning mechanism of the ANN does not rely on human expertise. Due to the homogeneous structure of the ANN, it is difficult to extract structured knowledge from the weights of the ANN. Hence, encoding a priori knowledge into the ANN becomes a difficult task. Neuro-fuzzy system is a hybrid system that combines the learning capability of FIS and the formation of fuzzy if-then rules by ANN. ANN learning algorithms are used to determine the parameters of the FIS.

Adaptive Neuro-Fuzzy Inference System (ANFIS) (Jang *et al*., 1997) implements a Takagi-Sugeno FIS and has a 5 layered architecture as shown in Fig. 3. The first hidden layer is for fuzzification of the input variables and T-norm operators are deployed in the second hidden layer to compute the rule antecedent part. The third hidden layer normalizes the rule strengths followed by the fourth hidden layer where the consequent parameters of the rule are determined. Output layer computes the overall input as the summation of all incoming signals. ANFIS uses back-propagation learning to determine premise parameters to learn the parameters related to membership functions and least mean squares estimation to determine the consequent parameters. The learning procedure is

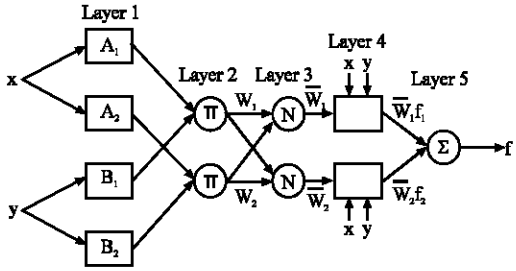

Fig. 3: Architecture of an adaptive neuro-fuzzy inference system

executed in 2 parts. In the first part, the input patterns are propagated and the optimal consequent parameters are estimated by an iterative least mean squares procedure whereas the premise parameters are assumed to be fixed for the current cycle through the training set. In the second part the patterns are propagated again and in this epoch, back-propagation is used to modify the premise parameters, whereas the consequent parameters remain fixed. This procedure is then iterated.

Every node i in layer 1 is an adaptive node with a node function whose output is equal to the membership grade of the particular input given by:

$$\begin{aligned} O_{1,i} &= \mu_{Ai}, (x) \qquad \text{for } i = 1, 2 \text{ or} \\ O_{1,i} &= \mu_{Bi-2} (y) \qquad \text{for } i = 3,4 \end{aligned} \tag{10}$$

where, x (or y) is the input to node i and A_i (or B_{i-2}) is a linguistic label associated with this node. Here, the membership function used is a Gaussian function given by:

$$\mu_A(x) = \exp\left(\frac{-0.5(x - c_i)^2}{\sigma_i^2}\right) \tag{11}$$

where c_i and σ_i is the mean and variance of the Gaussian membership function, respectively.

Every node in layer 2 is a fixed node and represents the firing strength of a rule. The output of each node is a product of all the incoming signals as given by:

$$O_{2,i} = w_i = \mu_{Ai}(x).\mu_{Bi}(y), \quad i = 1, 2 \tag{12}$$

Every node in layer 3 is a fixed node and calculates the ratio of firing strength of a rule to the sum of all rules firing strengths. The outputs of this layer are normalized firing strengths given by:

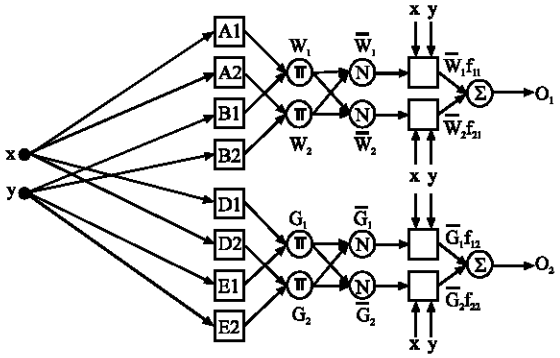

Fig. 4: Architecture of an Adaptive Neuro-Fuzzy Inference System

$$O_{3,i} = \overline{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1,2 \tag{13}$$

Each node layer 4 is an adaptive node with a node function given by:

$$O_{4,i} = \overline{w}_i . f_i = \overline{w}_i (p_i x + q_i y + r_i), \tag{14}$$

where, $\{p_i, q_i, r_i\}$ is the parameter set of this node.

The single fixed node in layer 5 computes the overall output as the summation of all incoming signals as given by:

$$O_{5,i} = \sum \overline{w}_i . f_i = \frac{\sum_i w_i . f_i}{\sum_i w_i} \tag{15}$$

Multiple Neuro-Fuzzy Inference System (MANFIS) can produce multiple outputs. It is a parallel structure with two ANFIS sharing same inputs to produce multiple outputs. The MANFIS architecture is shown in Fig. 4. The function of each layer of MANFIS is the same as that of ANFIS.

STATE ESTIMATION BY MANFIS OBSERVERS

State estimations are carried out in Dedicated Observer Scheme on the assumption that only one state variable is measurable and the other two states are immeasurable. MANFIS structure is constructed as a dedicated observer. Three observers are constructed with each sensor, forming dedicated observer scheme to estimate other two states. Each observer is applied with one sensor output in addition to the control input u_1 to estimate other two states.

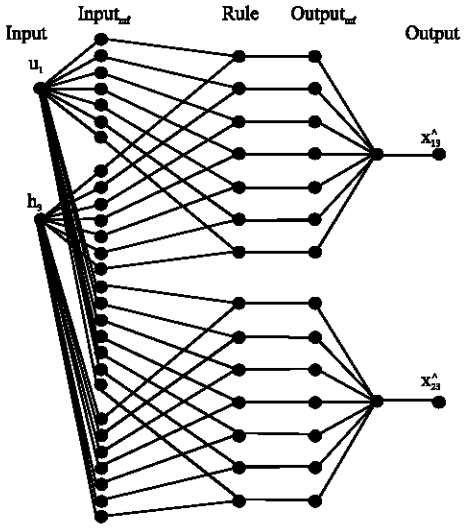

Fig. 5: Architecture of MANFIS observer

The neural networks are trained with the known open loop input-output relationships of level process in all possible ranges. The optimal shape and parameters for the membership functions of fuzzy inference systems with effective rule base are fixed by neural network. The membership functions are Gaussian for inputs and linear for outputs. Thirteen thousands six hundred and ninety six sets of data are used for training. The architecture of one such Adaptive Neuro-Fuzzy Inference System as state x_3 observer is shown in Fig. 5. The optimal number of both input and output membership functions of all the three observers is 7.

FAULT DETECTION AND IDENTIFICATION

The residuals are generated as error functions f by comparing the process outputs with the estimated states of the process. These residuals are further processed to form decision function η. As the decision function exceeds the threshold value when fault occurs, the fault alarms can be generated.

The states of the process are the outputs of perfect sensors as given by:

$$\left.\begin{aligned} x_1 &= h_1 \\ x_2 &= h_2 \\ x_3 &= h_3 \end{aligned}\right\} \tag{16}$$

The dedicated observers with h_2 and h_3 sensor outputs as inputs respectively give the estimated states for state x_1 as $\hat{x}_{12}$ and $\hat{x}_{13}$, respectively. The residuals that are generated from the three values of state x_1, one from h_1

sensor output and 2 from estimated values of state x_1. From the three values of state x_1, the estimation errors that are the difference between the three values of state x_1 are calculated and these error functions are given by:

$$\left.\begin{aligned} f_{11} &= \left|\hat{x}_{13} - \hat{x}_{12}\right| \\ f_{21} &= \left|\hat{x}_{13} - x_1\right| \\ f_{31} &= \left|\hat{x}_{12} - x_1\right| \end{aligned}\right\} \quad (17)$$

The decision functions η_{11}, η_{21} and η_{31} are formed as:

$$\left.\begin{aligned} \eta_{11} &= f_{21}f_{31} \\ \eta_{21} &= f_{11}f_{31} \\ \eta_{31} &= f_{11}f_{21} \end{aligned}\right\} \quad (18)$$

The values of the decision functions will be zero if no sensor fails. If any sensor fails, the decision function formed from the product of respective estimation errors will show great deviation. This deviation is used to identify the failure of that particular level sensor. If the h_1 level sensor fails, the functions f_{21} and f_{31} will grow quickly as they are the difference between the estimates and the failed sensor output and so η_{11} alone will grow much faster. This deviation beyond the threshold value is used to identify the failure of h_1 level sensor and to make the fault alarm. Similarly, the deviations of η_{21} and η_{31} will identify faults in h_2 and h_3 sensors, respectively. Similar decisions functions η_{12}, η_{22}, η_{32}, η_{13}, η_{32} and η_{33} can be made with other state estimates also.

SELECTION OF THRESHOLD VALUES

The values of the decision functions should be zero under no failure condition. If any sensor fails, one of the decision functions will show great deviation. This deviation is used to identify the respective sensor failure. If the normal value of these decision functions are fixed at zero, even a slight deviation, which is not necessarily because of sensor failure, can also make fault alarm. This false fault alarm is avoided by fixing threshold values for the decision functions instead of zero.

The residual and the decision function may change with the changing control inputs under various process conditions. Fixing threshold values are very important since the inevitable time delay in fault detection can cause stability problems. Higher values of the threshold for decision functions may delay the fault detection and cause stability problem whereas low values of threshold may miss the fault. In this work, the threshold values for decision functions are fixed in such a way that their values are greater than the values of respective decision functions under no failure condition and they are less than the values of respective decision functions under failure conditions.

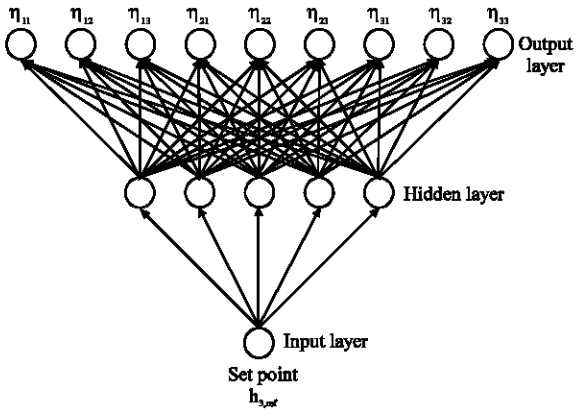

Fig. 6: Multilayer Perceptron Neural Network for threshold generation

The responses of the state feed back control of three tanks interacting level process are studied for various set points under normal and process perturbation conditions. The maximum values of all decision functions are observed. Various sensor failures are intentionally introduced at various instants of time and the responses of the interacting level process are studied for various set points under these failure conditions. The maximum values of all decision functions under these conditions are also observed.

A neural network based threshold selector is proposed in which the threshold values for the decision functions are generated based on the set point of the process. Since the values of the maximum deviations are found higher for higher set points, the threshold values for decision functions can be fixed based on the set points instead of generating adaptive threshold based on control inputs. A simple Multilayer Perceptron Neural Network (MLP NN) is constructed with one node in input layer, five nodes in one hidden layer and nine nodes in output layer. The input node is for set point and the nine nodes represent the nine decision functions. The architecture is shown in Fig. 6. For generating training data for NN to generate threshold values, the threshold values for some set points are fixed and a data set is formed.

The network is trained with the well-known back propagation algorithm and prepared to generate threshold values for any set point. The threshold value for a decision function is fixed for a set point by considering

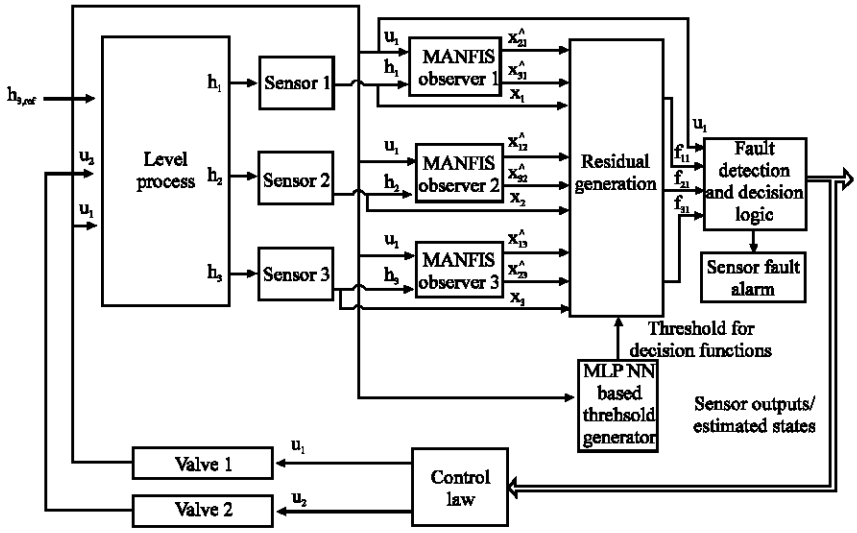

Fig. 7: Fault detection and identification scheme

the maximum value of the decision function under no failure condition and the minimum value of that decision function under the concerned sensor failure condition for the set point. Thus the thresholds are fixed such that:

$$\max (a_{ij}, b_{ij}) < \Gamma_{ij} > C_{ij} \quad (19)$$

where, a_{ij} is the value of decision function η_{ij} under no failure condition, c_{ij} is the value of decision function η_{ij} under ith sensor failure condition, b_{ij} is the value of decision function η_{ij} under sensor failures other than ith one, Γ_{ij} is the threshold for the decision function η_{ij} and η_{ij} is the decision function formed for ith sensor from the observer of state j.

The procedure of fixing threshold values of decision functions for a set point is explained here. The values are fixed for the set point of 10 cm. The decision functions η_{11}, η_{12} and η_{13} are the decision functions for identifying h_1 sensor failure condition. The maximum values of these decision functions under any process condition without any sensor failure are observed as 0.88, 0.66 and 0.69, respectively. It is observed that the values of these decision functions under h_1 sensor failure condition are 131, 120 and 101, respectively. The maximum values are 5.9, 4.9 and 4.1, respectively considering the values under other sensor failures. As the decision functions grow faster when any sensor fails, the threshold values for η_{11}, η_{12} and η_{13} can be fixed as 50, 25 and 25, respectively for early detection of failure. The threshold values are thus formed for these decision functions for all set points. Such data are formed for many set points and the neural network is trained with these data to generate threshold values for any set point. The threshold values for decision functions η_{11}, η_{21} and η_{31} generated using this procedure are found to be 50, 30 and 40 for a set point of 10 cm and 140, 100 and 120 for a set point of 20 cm. The proposed fault detection and identification scheme is shown in Fig. 7.

RECONFIGURATION OF CONTROL LAW

If any sensor failure is identified by the fault detection and identification logic, the estimated state, which is the same as output in this case, will come into action and gives the value of the state of the system for state feedback. Hence, perfect and smooth control is possible even under sensor failure conditions.

The system under no failure condition will work with the basic control law given by:

$$u_1 = -K_1 x_1 - K_2 x_2 - K_3 x_3 - K_{IN} (h_{3,ref} - x_3) \quad (20)$$

If any failure is detected in h_3 level sensor by the fault detection logic, the control law will be modified and the alternative control law is:

$$u_1 = -K_1 x_1 - K_2 x_2 - K_3 \hat{x}_3 - K_{IN} (h_{3,ref} - \hat{x}_3) \quad (21)$$

For h_1 sensor failure and h_2 sensor failure, the alternative control laws for perfect control are given in Eq. (22) and (23), respectively.

$$u_1 = -K_1 \hat{x}_1 - K_2 x_2 - K_3 x_3 - K_{IN} (h_{3,ref} - x_3) \quad (22)$$

$$u_1 = -K_1 x_1 - K_2 \hat{x}_2 - K_3 x_3 - K_{IN} (h_{3,ref} - x_3) \quad (23)$$

Estimated states $\hat{x}_1, \hat{x}_2$ and $\hat{x}_3$ are used as redundant states in alternative control laws which are available as outputs in dedicated observers.

RESULTS AND DISCUSSION

No failure condition: The Fault Detection and Identification scheme is implemented in three tanks interacting level process under normal condition. The set point is fixed at 10 cm and the state feed back control is applied for maintaining this level. No sensor failure is introduced and the various trends on levels, state estimates and the control inputs are shown in Fig. 8. Estimation error functions and decision functions under no failure condition are shown in Fig. 9. All decision functions are found within their respective threshold values and hence no fault alarms are generated.

Tank 1 level sensor failure condition: The set point is fixed at 10 cm and the state feed back control is applied for maintaining this level. A failure is introduced in h_1 level sensor at 150 sec. The trends on levels and error functions are shown in Fig. 10. Since, the state x_1 which is h_1 sensor output, deviate more and the error functions relating this state grow faster at 150 sec.

Figure 11 shows the decision functions. The decision function η_{11} goes beyond its respective threshold value generated by NN based threshold generator for a set point of 10 cm and a fault alarm is generated for h_1 sensor failure at 150 sec. All other decision functions are found within their respective threshold values and hence no false alarms are reported. The level is maintained at the desired level even under this sensor failure condition since the control law is reconfigured as given in Eq. (22).

The error functions and decision functions under h_1 sensor failure for a set point of 20 cm are shown in Fig. 12 and 13, respectively. The sensor failure is correctly identified as the decision function η_{11} grows beyond its threshold value generated for a set point of 20 cm.

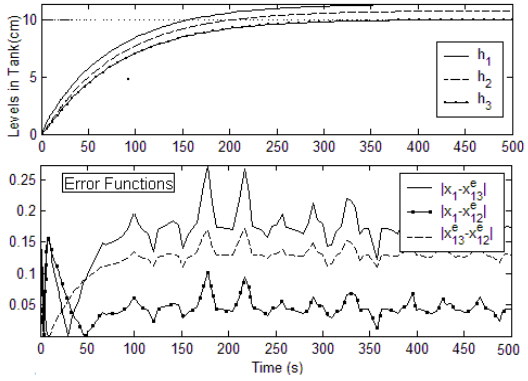

Fig. 8: Trends of states and error functions under no failure condition

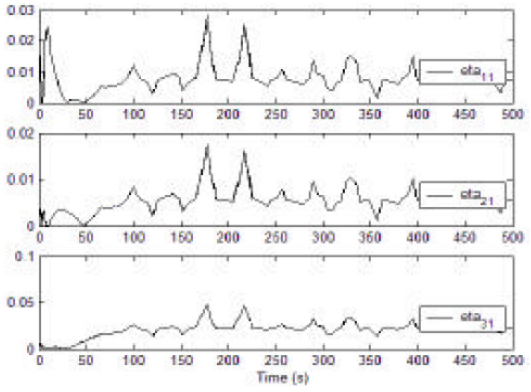

Fig. 9: Trends of decision functions under no failure condition

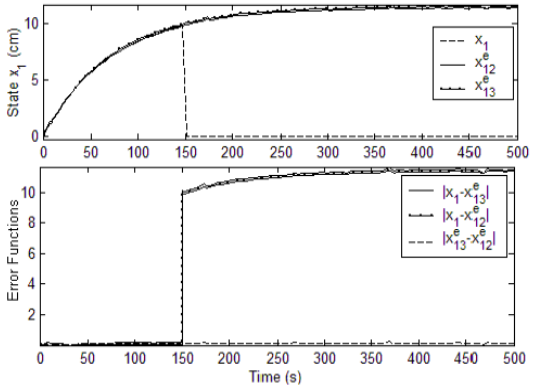

Fig. 10: Trends of State estimates and Error functions under h_1 sensor failure condition with set point set at 10 cm

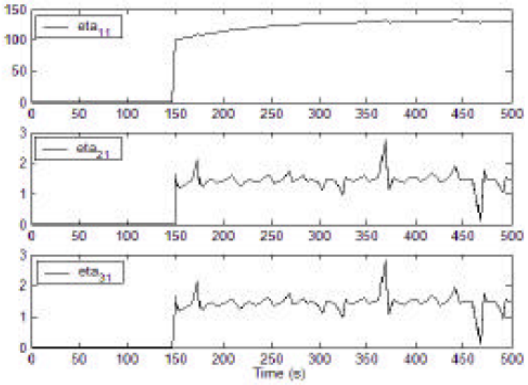

Fig. 11: Trends of decision functions under h_1 sensor failure condition with set point set at 10 cm

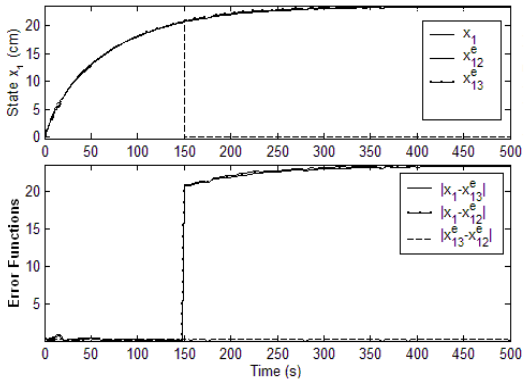

Fig. 12: Trends of State estimates and error functions under h_1 sensor failure condition with set point set at 20 cm

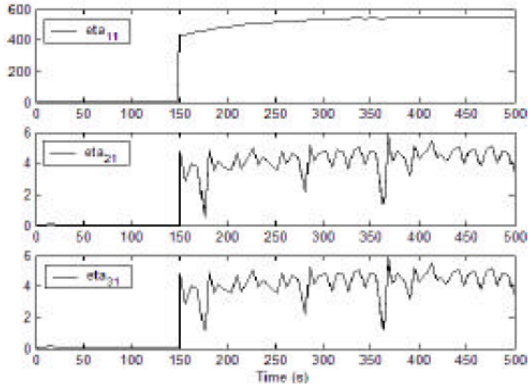

Fig. 13: Trends of decision functions under h_1 sensor failure condition with set point set at 20 cm

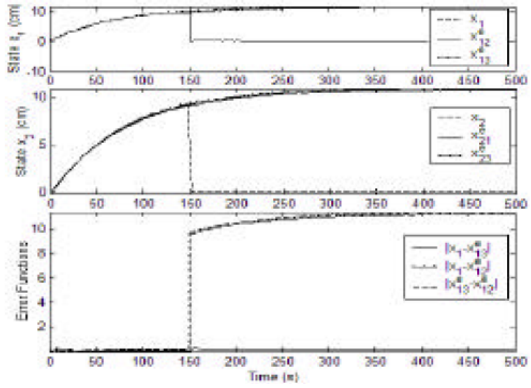

Fig. 14: Trends of State estimates and Error functions under h_2 sensor failure condition with set point set at 10 cm

Tank 2 level sensor failure condition: A similar failure is introduced in h_2 level sensor at 150 s with a set point of 10 cm and the trends of states and error functions are shown in Fig. 14. Since, the state estimate x^e_{12} is the output of MANFIS observer driven by h_2 sensor output, it shows greater deviations in the estimation whereas the states x_1 and x^e_{13} show no significant estimation error. This large deviation of x^e_{12} results in large value of error functions $|x_1 - x^e_{12}|$ and $|x^e_{13} - x^e_{12}|$. As the result, the value of η_2 grows larger exceeding its threshold value as shown in Fig. 15 making a fault alarm for the failure of h_2 sensor at 150 s whereas η_{11} and η_{31} are within their threshold values.

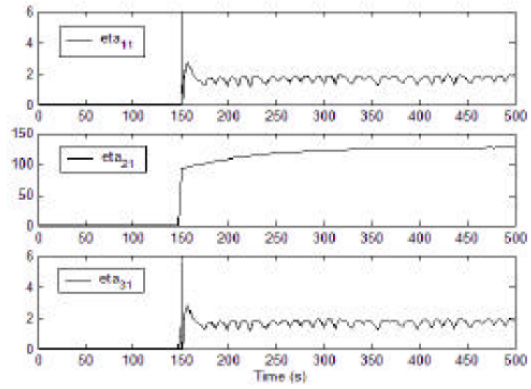

Fig. 15: Trends of decision functions under h_2 sensor failure condition with set point set at 10 cm

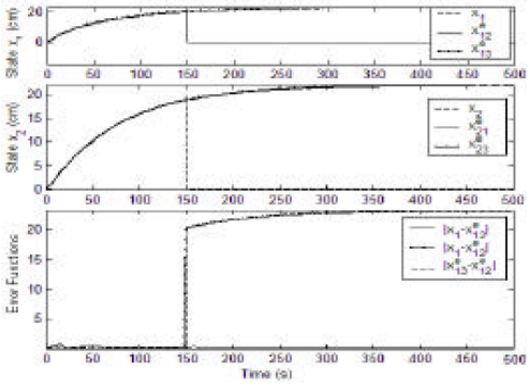

Fig. 16: Trends of State estimates and Error functions under h_2 sensor failure condition with set point set at 20 cm

The control law takes the value of estimated state x^e_{21} instead of h_2 as in Eq. (23). The level h_3 in tank 3 is controlled at the desired value, $h_{3,ref}$ perfectly even under this failure condition. Similar results are obtained on introducing failure in h_2 level sensor at 150 sec with a set point of 20 cm. The trends of error functions and decision functions are shown in Fig. 16 and 17, respectively.

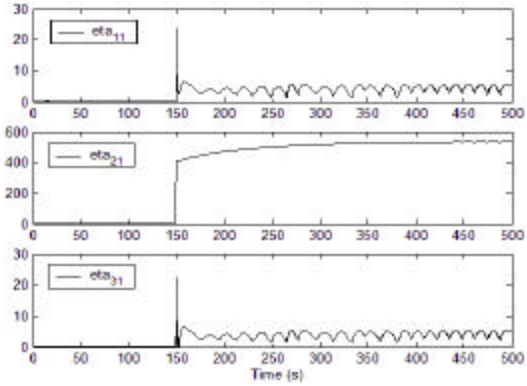

Fig. 17: Trends of decision functions under h_2 sensor failure condition with set point set at 20 cm

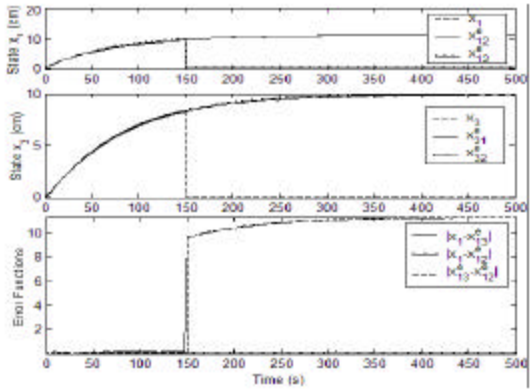

Fig. 18: Trends of State estimates and Error functions under h_2 sensor failure condition with set point set at 10 cm

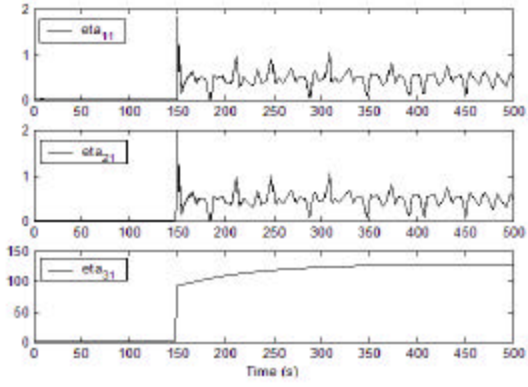

Fig. 19: Trends of decision functions under h_3 sensor failure condition with set point set at 10 cm

Tank 3 level sensor failure condition: A failure in h_3 sensor is introduced at 150 s and the trends are shown in Fig. 18. The estimated states x^e_{13} and x^e_{23} alone, which are

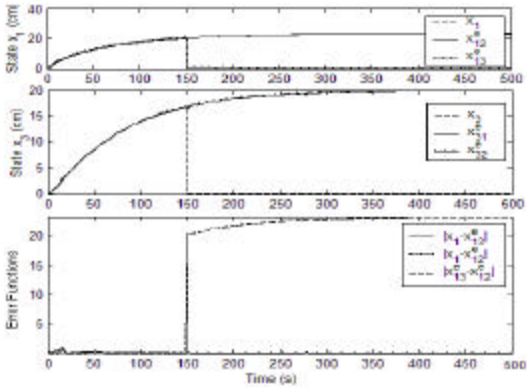

Fig. 20: Trends of State estimates and Error functions under h_3 sensor failure condition with set point set at 20 cm

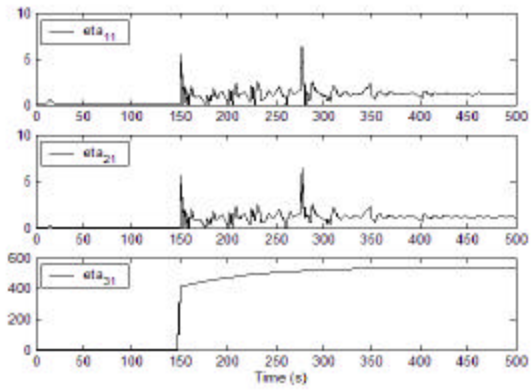

Fig. 21: Trends of decision functions under h_3 sensor failure condition with set point set at 20 cm

output of observer driven by h_3 sensor, experience large estimation errors. Other estimated states x^e_{12} and x^e_{22} are not disturbed by this failure and show only negligible errors. The large deviation in x^e_{13} results in large value of error functions $|x_1 - x^e_{13}|$ and $|x^e_{13} - x^e_{12}|$. This in turn results in very large value of η_{31} as shown in Fig. 19. The h_3 sensor fault is detected at 150 s exactly at the time of failure since the decision function η_{31} exceeds its threshold value at 150 s whereas η_{11} and η_{21} are within their threshold limits. This makes a fault alarm for h_3 level sensor. The output of the failed sensor h_3 is then replaced by its estimated value x^e_{32} for state feedback as in (21) and h_3 is controlled at the set point perfectly even under this failure condition. The fault detection works well for h_3 sensor failure for a set point of 20 cm also. The various trends are shown in Fig. 20 and 21.

CONCLUSION

Three pairs of Multiple Adaptive Neuro-Fuzzy observers are designed by fixing the optimal shape and parameters of the membership functions and effective rule base by neural networks to estimate the levels in three-tank interacting level process. All level sensor failure conditions are simulated for two different set points. From the study performed it has been noticed that the system has detected failures successfully at the time of failure itself in any sensor if it occurs. The sensor that has failed is correctly identified. The control law is modified accordingly and the level in tank 3 is maintained at the desired value even under the failure conditions.

REFERENCES

Alan, S. Willsky, 1976. A survey of design methods for failure detection in dynamic systems. Automatica, 12: 601-611.

Clark, R.N., D.C. Fosth and V.M. Walton, 1975. Detecting instrument malfunctions in control systems. IEEE. Trans. Aerospace Elect. Syst., 11 (4): 465-473.

Clark, R.N., 1978. Instrument Fault Detection. IEEE. Trans. Aerospace Elect. Syst., 14 (3): 456-465.

Clark, R.N., 1989. State Estimation Schemes for Instrument Fault Detection. Fault Diagnosis in Dynamic Systems: Theory and Applications, Patton R.J., P.M. Frank and R.N. Clark (Eds.). Englewood Cliffs, NJ, Prentice Hall.

Emani-Naeini A., M.M. Akhter and S.M. Rock, 1988. Effect of Model Uncertainty on Failure Detection. The Threshold Selector. IEEE Trans. Automatic Control, 33: 1106-1115.

Frank, P.M., 1990. Fault Diagnosis in dynamic systems using analytical and knowledge based redundancy. A survey and some new results. Automatica, 26 (3): 459-474.

Isermann, R., 1984. Process Fault Detection based on Modeling and Estimation methods. A survey. Automatica, 20 (4): 387-404.

Jang, J.S., C.T. Sun and E. Mizutani, 1997. Neuro-Fuzzy and Soft Computing. A Computational Approach to Learning and Machine Intelligence, Prentice Hall, Upper saddle River, NJ.

Mohanlal, P.P. and M.R. Kaimal, 2004. Design of optimal fuzzy observer based on TS Fuzzy model. In: Proceeding IEEE International Conference Fuzzy Systems, Budapest, Hungary, 2: 605-610.

Napolitano, M.R., C. Neppach, V. Casadorph and S. Naylor, 1995. Neural Network based scheme for Sensor Failure Detection, Identification and Accommodation. J. Guidance, Control Dynamics, 18 (6): 1280-1286.

Stefani, Shahian, Savant and Hosteller, 2002. Design of Feedback Control Systems. 4th Edn. Oxford University Press.

Xiao-Jun Ma and Zeng-Qi Sun, 1998. Analysis and Design of Fuzzy Controller and Fuzzy Observer. IEEE. Trans. Fuzzy Syst., 6 (1): 41-50.