SENSOR FAULT DETECTION IN A SATELLITE LAUNCHER USING ADAPTIVE NEURO-FUZZY OBSERVER

S. Nagarajan* J. Shanmugam** and T. R. Rangaswamy+

Abstract

Fault detection is performed by estimating the states and comparing them with measured values. A fault is signaled when the difference between the estimated and measured values crosses a threshold value. This paper presents the design of adaptive neuro-fuzzy observer based sensor fault detection and fault tolerant control under sensor failure conditions for a satellite launcher. In this Adaptive Neuro-Fuzzy Inference System (ANFIS), optimal shape and parameters for the membership functions and effective rule base for the fuzzy system are fixed by a neural network. Three such adaptive neuro-fuzzy inference systems act as reduced order observers that estimate the system states. 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 to implement the fault tolerant control. In this work, such fault detector is designed and simulated. The individual failures of three sensors in the satellite launcher are considered and the results are discussed. The results show that the system is able to detect any sensor failure situations perfectly.

Key words: Adaptive Neuro-Fuzzy Inference System, Estimation error, Fault Detection, Fault tolerant control, Reduced Order Observer, Decision function, Sensor failures

Introduction

In the case of unstable vehicles like satellite launchers, the failure of one sensor can be disastrous if the control system is not provided with redundancy. Due to this characteristic, it is important for these vehicles to identify sensor failures as quickly as possible and reconfigure the control law from the failed one to an alternative one. The purpose of the Fault tolerant control system is to detect, identify and accommodate sensor failures.

Model based fault detection techniques are based on observers [1], state estimating filters [2] or Parameter Estimators [3]. In the observer techniques, the state observers estimate the states of a system. If the output variables are the same as the state variables, all states except one state can be considered as measurable states. Using reduced order observers, the unmeasurable states can be estimated using measurable states. Thus it is possible to estimate the states from one or more other output variables without its own sensor. This leads to the concept of fault detection and control using the measurements only from the perfect sensors for getting the estimated state of the faulty 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 [4] was introduced in which each sensor of interest drives an observer to perform a complete state estimation. In case where 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, fault detection uses these techniques since they do not need any mathematical model and can accommodate non-linearities [5]. The reduced order observer can be modeled using fuzzy system. A straightforward approach [6] is to assume a certain shape for the membership functions. The effectivity of the fuzzy models representing nonlinear input-output relationships depends on the fuzzy partition or the input-output spaces. Therefore, the tuning of membership functions becomes an important

* Assistant Professor             + Professor and Head
Department of Instrumentation and Control Engineering, BSA Crescent Engineering College, Vandalur, Chennai-600 048, India
** Professor and Head, Division of Avionics, Madras Institute of Technology, Anna University, Chromepet, Chennai-600 044, India
Manuscript received on 08 Aug 2006; Paper reviewed, revised and accepted on 04 Dec 2006
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 for the membership functions that depends on different parameters that can be learned by a neural network with a set of training data in the form of correct input-output relationships and a specification of the rules including a preliminary definition of the corresponding membership functions. The Adaptive neuro-fuzzy system, which is a neural network, fix the optimal shape and parameters for the membership functions and effective rule base for the fuzzy system for observer modeling.

In this work, three states are measured in the longitudinal control of satellite launcher and used to control the vehicle through state feedback. Three adaptive neuro-fuzzy observers are designed with two states as inputs. Using these three designed observers, it is possible to measure all the states of the system and they can be feedback 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 accommodation using reconfiguration of the control law that performs the fault tolerant control.

Mathematical Model

The mathematical model used to describe the longitudinal motion of the satellite launcher is given by equation (1) and can be found in [7]. It describes the open loop dynamics of the longitudinal motion.

\[ \dot{x} = Ax + Bu \]  

(1)

with the state vector given by

\[ x^T = [\omega \ q \ \theta] \]  

(2)

and

\[ u = \beta Z \]  

(3)

The matrices A and B in equation (1) are given by

\[ A = \begin{bmatrix} Z_w & U_o + Z_q & -g \\ M_w & M_q & 0 \\ 0 & 1 & 0 \end{bmatrix} \]  

(4)

\[ B^T = \begin{bmatrix} Z_{BZ} & M_{BZ} & 0 \end{bmatrix} \]  

(5)

The parameters \( Z_w, Z_q, M_w, M_q, Z_{\beta Z}, \) and \( M_{\beta Z}, \) contained in matrix A and B, are the aerodynamic derivatives of the satellite launcher vehicle. The parameter \( U_o \) is the flight speed of the vehicle and the parameter \( g \) is the gravity acceleration. The state variable \( \omega \) is the vehicle velocity along the z-body axis, called normal velocity, the state variable \( q \) is the vehicle pitch-rate, that is, its angular velocity around the y-body axis and the state variable \( \theta \) is the vehicle pitch-attitude with respect to y-body axis, all defined in body fixed axes system. The control \( \beta Z \) is the pitch control deflection. The data used for the vehicle are given in Table-1.

Longitudinal Control System

The longitudinal control system is designed with the objective to track a reference pitch attitude, \( \theta_{ref} \) and the regulation of the remaining states and the model is given in (6).

\[
\begin{bmatrix}
\dot{x} \\
\dot{\omega} \\
\dot{q} \\
\dot{\theta} \\
\dot{e}_{\theta}
\end{bmatrix} = \begin{bmatrix}
Z_w & Z_q + U_o & -g & 0 \\
M_w & M_q & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & -1 & 0
\end{bmatrix}
\begin{bmatrix}
\omega \\
q \\
\theta \\
e_{\theta}
\end{bmatrix} + \begin{bmatrix}
0 \\
0 \\
\beta Z \\
e_{\theta}
\end{bmatrix}
\begin{bmatrix}
0 \\
0 \\
\theta_{ref} \\
1
\end{bmatrix}
\]  

(6)

where the state variable \( e_{\theta} \) is the pitch-attitude error integral. This has been included to keep the steady state error near zero. The control system was designed by LQR method and the control law is given by

\[ \beta Z = -K_x - K_0 \theta_{ref} \]  

(7)

<table>
<thead>
<tr>
<th>Table-1 : Data used for the satellite launcher</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Z_w ) (s(^{-1}))</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>-0.0968</td>
</tr>
</tbody>
</table>
with state vector \( \mathbf{x} \) given by
\[
\mathbf{x}^T = \begin{bmatrix} \omega & q & \theta & e_\theta \end{bmatrix}
\] (8)

and the state feedback gain \( K \) is given by
\[
K = \begin{bmatrix} K_1 & K_2 & K_3 & K_4 \end{bmatrix}
\] (9)

and \( K_0 \) is the feed forward gain. The designed value of \( K \) is \([0.0013 \ 1.4551 \ 3.3581 \ -3.2581]\) and the feed forward gain \( K_0 \) is -3.257. Fig.1 shows the satellite launcher with this control law.

**State Estimators**

In this system, the measurements of all states are possible since the output variables are same as state variables. This makes no requirement for any state estimation. But for implementing fault detection, state estimations are carried out on the assumption that only two state variables are measurable and the third state under estimation is unmeasurable. Three such estimators are constructed for three sensors. Each Estimator is applied with two outputs in addition to the control input \( \beta_z \). Thus the estimator \( E_1 \) estimates \( \omega \) with \( q, \theta \) and \( \beta_z \) as inputs, \( E_2 \) estimates \( q \) with \( \omega, \theta \) and \( \beta_z \) as inputs and \( E_3 \) estimates \( \theta \) with \( \omega, q \) and \( \beta_z \) as inputs [8]. This is shown in the block diagram 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 in such a way that neural network learning algorithms are used to determine the parameters of the Fuzzy Inference System. A fuzzy inference system (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, 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) [9] implements a Takagi Sugeno FIS and has a five 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 output 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 executed in two parts. In the first part, the input patterns are propagated and the optimal consequent parameters are estimated by an iterative least mean squares procedure, while 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, while 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

\[ O_{1,i} = \mu_{A_i}(x), \quad \text{for } i = 1,2 \]  \[ O_{1,i} = \mu_{B_{i-2}}(y), \quad \text{for } i = 3,4 \]  (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

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

where $c_i$ and $\sigma_i$ 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

\[ O_{2,i} = w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y), \quad i = 1,2 \]  (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

\[ O_{3,i} = \bar{w}_i = \frac{w_i}{w_{i1} + w_{21}}, \quad i = 1,2 \]  (13)

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

\[ O_{4,i} = \bar{w}_i \cdot f_i = \bar{w}_i \left( p_i x + q_i y + r_i \right), \]  (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

\[ O_{5,i} = \sum \bar{w}_i \cdot f_i = \frac{\sum_i w_i \cdot f_i}{\sum_i w_i} \]  (15)

**Adaptive Neuro-fuzzy Observers**

An adaptive network with above structure which is functionally equivalent to Takagi Sugeno fuzzy model is constructed as an observer with six inputs to give the estimated state value as its output. The inputs are the reference input, control input, the other two sensor outputs and their previous values. The result is the estimated value of sensor measurement whose output is not considered. Three Adaptive Neuro-Fuzzy observers are designed and one such observer for pitch estimation is shown in Fig.4.

The neural networks are trained with the known input-output relationships of satellite launcher in possible ranges. The optimal shape and parameters for the membership functions of fuzzy inference systems with effective rule base are fixed by neural networks for each observer as listed in Table-2. The architecture of one such Adaptive Neuro-Fuzzy Inference System for pitch observer is shown in Fig.5 and the corresponding membership functions are shown in Fig.6.

**Fault Detection and Identification**

The decision functions are to be built to detect a failed sensor and then to reconfigure the control law from the basic control law to an alternative one. The decision func-
The decision functions \( \eta_1, \eta_2, \) and \( \eta_3 \) are given by equation (18).

\[
\eta_1 = f_2 f_3, \quad \eta_2 = f_1 f_3 \quad \text{and} \quad \eta_3 = f_1 f_2 \tag{18}
\]

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 estimation errors will show great deviation. For example, if the sensor of the pitch attitude \( \theta \) fails, the functions, \( f_1 \) and \( f_2 \) will grow quickly and so \( \eta_3 \) will grow much faster. This deviation is used to identify that the pitch sensor has failed and to make the fault alarm to do the reconfiguration of the control law. If the normal value of this decision function is fixed at zero, even a slight deviation, which is not necessarily because of sensor failure, can also make fault alarm. This false alarm is avoided by fixing threshold values for the decision functions instead of zero.

**Fault Tolerant Control**

If any sensor failure is identified by the fault detection and identification logic, the estimated state which is also the output in this case, will come into action and give the values of the state of the system for 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 equation (19).

\[
\beta_z = -K_1 \omega - K_2 q - K_3 \theta - K_{\text{In}} e_\theta - K_0 \theta_{\text{ref}} \tag{19}
\]

If any failure is detected in the pitch attitude \( \theta \) sensor, the control law will be modified and the alternative control law is given as

\[
\beta_z = -K_1 \omega - K_2 q - K_3 \hat{\theta} - K_{\text{In}} e_\theta - K_0 \theta_{\text{ref}} \tag{20}
\]

where \( e_\theta \) is calculated from its derivative given in equation (21).
\[ \dot{\theta} = \theta_{\text{ref}} - \hat{\theta}_3 \]  \hspace{1cm} (21)

For pitch-rate \( q \) sensor failure and the normal velocity \( \omega \) sensor failure, the alternative control laws are given in equations (22) and (23) respectively.

\[ \beta_z = -K_1 \omega - K_2 \hat{x}_2 - K_3 \theta - K_{\text{In}} e_\theta - K_0 \theta_{\text{ref}} \]  \hspace{1cm} (22)

\[ \beta_z = -K_1 \hat{x}_1 - K_2 q - K_3 \theta - K_{\text{In}} e_\theta - K_0 \theta_{\text{ref}} \]  \hspace{1cm} (23)

**Simulation Results**

The simulation is conducted under no failure condition. The set point for pitch attitude \( \theta \) is 0.1. Fig. 7 shows the time history of actual pitch, pitch sensor output, estimated pitch and pitch estimation error. The pitch attitude is maintained at the set point nearly after 3.5 seconds. Figs. 8 and 9 show similar trends for normal velocity and pitch rate. In all the cases the estimation error is negligible.

Fig. 10 shows the time history of the control input.

For the same system, a failure is introduced in pitch attitude sensor at 1.5 second. The Figs. 11, 12 and 13 show the trends of vehicle velocity, pitch rate and pitch attitude respectively. The states \( x_1 \) and \( x_2 \) show greater deviations in the estimation since one of the inputs to these estimators is the pitch attitude sensor output.

Figure 13 shows that the state \( x_3 \) has no significant estimation error and \( \theta \) is controlled perfectly even under the failure condition. The fault is detected correctly at 1.52 seconds and the moment of transition between the failed control law and the alternative control law can be seen in Fig. 14, which shows the time history of the control input.

A failure is now introduced in vehicle velocity sensor at 1.5 second. The Figs. 15, 16 and 17 show the trends of vehicle velocity, pitch rate and pitch respectively. The states \( x_2 \) and \( x_3 \) show greater deviations in the estimation. The state \( x_1 \) has no significant estimation error and \( \theta \) is controlled perfectly even under the failure condition. The fault is detected correctly at 1.53 seconds. Fig. 18 shows the time history of the control input.

The pitch rate sensor is made struct at -0.1 at 1.5 seconds and similar trends of vehicle velocity, pitch rate, pitch and control input are shown in Figs. 19, 20, 21 and 22 respectively. The fault is detected correctly at 1.52 seconds and Pitch \( \theta \) is still controlled perfectly at the desired reference input 0.1.
Three 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 normal velocity, pitch rate and pitch attitude of a satellite launcher. From the study performed it has been noticed that the system has detected failures within 0.03 seconds in any sensor if it occurs. The sensor that failed is correctly identified. The control law is modified accordingly and the pitch attitude is maintained at the desired value even under the failure conditions. No missed alarm and false alarm are reported. Since the effect of change in system parameters and noises on estimation are

**Conclusion**

![Fig.11 Time history of vehicle velocity $\omega$ under pitch attitude sensor failure condition](image)

![Fig.12 Time history of pitch rate $q$ under pitch attitude sensor failure condition](image)

![Fig.13 Time history of pitch attitude $\theta$ under its sensor failure condition](image)

![Fig.14 Time history of control input $\beta_z$ under pitch attitude sensor failure condition](image)

![Fig.15 Time history of vehicle velocity $\omega$ under its sensor failure condition](image)

![Fig.16 Time history of pitch rate $q$ under velocity sensor failure condition](image)
not considered in this work, false alarms are reported under such conditions. Further work is suggested to improve the performance of the estimators so that no false alarms are reported in such environments as well.

References


