

FAULT DETECTION AND FAULT TOLERANCE METHODS FOR INDUSTRIAL ROBOT MANIPULATORS BASED ON HYBRID INTELLIGENT APPROACH

M. Dev Anand*; T. Selvaraj** & S. Kumanan**

*Corresponding Author, Professor and Deputy Director Academic Affairs, Department of Mechanical Engineering, Noorul Islam Centre for Higher Education

**Professor, Department of Production Engineering, National Institute of Technology, Tamilnadu, India

E-mail: anandpmt@yahoo.co.in

Abstract:

Fault tolerance is increasingly important in modern industrial robotic manipulators, especially those operated in remote and hazardous environment. Faults in robotic manipulator can cause economic and serious damages. So the robots need the ability to detect as well as tolerate failures, allow effectively coping with internal failures and continue performing designated tasks without the need for immediate human intervention. This saves time and cost involved in repairing the robot. This type of autonomous fault tolerance is also useful for industrial robots in that it decreases down-time by tolerating failures, identifies faulty components or subsystems to speed up the repair process and prevents the robot from damaging the products being manufactured. To support these fault tolerant capabilities, methods of detecting and tolerating failures must be perfected in robot manipulator. A number of researchers have proposed fault detection/tolerance architectures for robotic manipulators using the model based analytical, and redundancy approach. One of the main issues in the design of fault detection system is to model the rigid link robotic manipulators with modeling uncertainties. In this paper, a new approach hybrid intelligence based fault detection/tolerance for robot manipulators is discussed. A learning architecture, with neural network as on-line approximates the off-nominal system behavior, which is used for monitoring the robotic system for the faults. This generates the residual by comparing the actual output from robot. Fuzzy inference system is applied to identify and tolerate the faults which provide the adoptive threshold under the varying conditions. The new concepts discussed were validated through simulation study using a Scorbot ER 5plus manipulator robot mat lab toolbox.

Key Words: Fault Tolerance, Neural Networks, Fuzzy Logic, Neuro-Fuzzy, Robot

1. INTRODUCTION

Robots are often used in inaccessible or hazardous environments in order to alleviate some of the time, cost and risk involved in preparing humans to endure these conditions. In order to perform their expected tasks, the robots are often quite complex, thus increasing their potential for failures. However, if people are frequently sent into these environments to repair every component failure in the robot, the advantages of using the robot are quickly lost. Fault tolerant robots are needed which can effectively detect and adapt to software or hardware failures in order to allow the robots to continue working until repairs can be realistically scheduled. Hence availability and reliability of these components are important for production environment. Hence, automated monitoring of the robotic manipulator for any faults and effective accommodation of such faults plays a crucial role in the use of robotic manipulators as autonomous systems (Pdrs Mal, 1998).

In general, a robot is a mechanical structure consisting of links connected by joints which typically move around one axis. Unlike the human arm and hand which together have over

thirty degrees of freedom, most robots only five or six, the minimum number of degrees necessary to position and orient the robot in three dimensions. Robots with more than six joints can choose between several configurations of the links to place the end of the robot at the specific position. The joints may be either rotational or prismatic, and each is driven by an actuator either directly or through a combination of gears, chains and belts. The most common robot actuators are electric motors. A computer control system called the controller commands the robot to move through its workspace based on a plan or desired trajectory of the robot end effector. The plan is developed using an inverse kinematics algorithm which computes the next desired joint configuration given the current configuration and the next desired end effector position. The plan may be pre-computed off-line and stored in the controller's memory or computed dynamically during the assigned task. The controller computes the necessary torque to apply to each motor in order to move the robot from the given current position into the next desired position. Information about the current position or velocity of the robot joints are relayed to the controller from sensors.

Fault Diagnosis (FD) in robot manipulator has been the area of active research for the past two decades. In the past few years, as the application of robots has expanded, there has been significant activity in the area of robot reliability and fault tolerance. Fault tolerance is especially important in remote and hazardous environments, such as found in space, underwater, and radioactive applications (where repair is often infeasible, and failure can have disastrous consequences), although reliability and safety are important issues in almost all applications. Furthermore, in the future, robots may become common household items. To be able to work among humans, personal robots will be required to have FD capabilities. Unfortunately, there are several possible sources of failure in robotic systems (mechanical, electrical, electronic, hydraulic, and pneumatic), which makes robot failures not highly unusual. FD methods are usually based on the residual generation and evaluation concept. A mathematical model is used to reproduce the dynamic behavior of the fault-free system; the deviation of the output predicted by the model from actual output measurements forms the so-called residuals, which, when properly analyzed, provides valuable information about failures. Based on the failure to take an intelligent decision making with the help of the NF based FD system.

2. LITERATURE REVIEW

Many fault tolerant systems have been developed for computer, airplane, and industrial systems. Several of these techniques used in fault tolerant systems have provided models for robotic fault tolerance schemes. However, the trend in robotics is to use only those schemes which rely on physical redundancy of components. Many methods of fault tolerance exist which do not alter the physical system.

FD process consists of two stages. They are residual generation and decision making. The first stage i.e., residual generation uses various techniques. One among that is analytical redundancy technique. It is of two forms first one is direct redundancy and second one temporal redundancy. Direct redundancy is used by Evans, *et al.* (1970), Gilmore, *et al.* (1972) and Desai, *et al.* (1981). Direct redundancy gives the relationship among instantaneous outputs of sensor. Temporal redundancy is used by Deckeri, *et al.* (1977) and Watanabe, *et al.* (1982). Temporal redundancy gives the relationship among the histories of sensor outputs and actuator inputs. The concept of analytical redundancy is used to establish results for the generation of residuals for general dynamic systems also by Stengel, (1991).

In another approach, fault detection method is separated into two ways i.e., method which do not use plant models and plant based models. Sensor system is used in method which does not use plant models. Plant based models use two techniques, they are observer based techniques and parameter estimation techniques. The overview of theory and applications of parameter estimation method was described by Isermann, (1984). The overview of theory and applications of robust observer based fault detection method was described by Frank, (1993). The robust based approach by Wunnenberg, (1990) emphasizes structured uncertainties and allows a robust residual evaluation even in the presence of model uncertainties and unknown inputs. A

direct way to construct an output observer is, using the input-output relation which is usually described by a transfer function in frequency domain. It was introduced by Viswanatham, *et al.* (1987) and lately extended by Ding and Frank, (1991).

Threshold logic is often used in the area of FD. The theoretical foundation of adaptive threshold was given by Emami, *et al.* (1988) in terms of the threshold selector and the idea of adaptive threshold was introduced by Clark, (1989). Adaptive threshold technique is developed independently by Schneider and Sauter, *et al.* (1993). A fuzzy based approach for a robust threshold selection for fault detection has been described by Frank and Kiupel, (1992). The theory of fuzzy logic was introduced by Zadeh, (1965). A fuzzy logic based threshold is used for residual evaluation and is used in an observer based fault detection method of robots and is given by Schneider and Frank, (1994).

A residual evaluation algorithm based on the nonlinear adaptive observer approach to fault detection was developed by Ding and Frank, (1991). Frank, (1990) used observer based technique to detect fault in a nonlinear dynamical system. Parameter adaptive control methods are conducive to determining the bounds on the tracking error due to modeling inaccuracies because of the separation of parameters in the control equations Isermann, (1990). The threshold generator is used to produce dynamic thresholds for fault detection based on the current state of the system at each situation of the controller for every joint in the robot Paredis, (1994). Isermann and Freyermuth, (1991) used parameter estimation technique to detect fault in a nonlinear dynamical system.

For residual generation and residual analysis Artificial Neural Networks (ANN) are also used. For fault detection and isolation, ANN uses Multi Layer Perceptron (MLP), Radial Basis Function Network (RBFN) and this approach is given by Macro, (2001). FD on the wheels of the mobile robot using local model neural network is proposed by Elias, (2004). They use change detection algorithm for reliable residual generation which is already proposed by Basseville and Nikiforov, (1993).

NF approach is applied to rigid and elastic robots. Sajidman, *et al.* (1995) and Frey, (2001) explained that shock disturbances have to be avoided by introducing a fuzzy based switching strategy. To diagnose the malfunctions or processed automatically steps of feature extraction and evaluation have to be performed Patton, (2000). Feature evaluation is divided in to statistical and artificial intelligent methods. NF approaches are used for designing fuzzy feature evaluation modules. Fuzzy model is initially created using a combination of Classification and Regression Trees (CART) Jang, (1994) and clustering analysis by Gustafson and Kessel, (1979). The design of decoupling observers is described by Chen and Patton, (1999) and Uppal, *et al.* (2003).

By providing robots with autonomous fault detection and tolerance, lengthens their working life without requiring immediate human intervention for every failure situation. The framework can further integrate work being done on control level failure tolerance Sreevijayan, (1994) and Ting, (1993) and detection of software failures Pham, (1992) and Tso, (1993). The tolerance actions which is chosen have the positive effect of covering most of the substructure failure by eliminating or isolating their influence on the rest of the system. The expert system uses time varying probabilities to model component decay during the life of the robot Vesley, (1981). The first robot used to demonstrate the fault tolerance frame work was a simulated four link robot originally developed by Hamilton, (1992) and executed on a Silicon Graphics Personal Iris Computer (SGPIC).

This necessitates the development of FD algorithm, which has the ability to detect manipulator failures in the presence of modeling uncertainties. Such algorithms are referred as robust FD schemes. In Dev Anand, *et al.* (2006) a methodology was proposed based on NF, which is used for faults detection and isolation present in robot manipulators. A learning architecture, with neural network as on-line approximates the off-nominal system behavior which is used for monitoring the robotic system for faults that generates the residual by comparing the actual output from robot manipulators. Fuzzy inference system is applied to identify and isolate the faults. Generally, the FD process is viewed as consisting of two stages: residual generation and decision making as shown in "Figure 1". Outputs from the sensory are processed and compared with the expected values from quantitative nominal model, resulting value is referred as residual. In the second stage, the decision process, the residuals are examined for the

presence of failure signatures. Decision functions or statistics are calculated using the residuals, and a decision rule is then applied to the decision statistics to determine if any failure has occurred. It is argued that a robust FDI system can be achieved by designing a robust residual generation process.

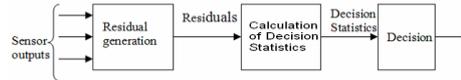


Figure 1: Two Stage Structure of Decision Statistics.

- i. *Detection* deals with determining if a malfunction has occurred in the system.
- ii. *Diagnosis* considers the problem of isolating and/or identifying a fault and then decision making.

In order to develop residual generation procedures, it is important to identify the redundancy relations of a system and to characterize them according to how they are affected by model errors and uncertainties. In this work, a systematic approach was developed to that consider uncertainties directly. Further to develop the concept of neural network based redundancy model as a basis for determining redundancy relations for residual generation. This residual generation is least sensitive to model errors. Numerous advantages characterize this method:

- i. For diagnosis, no additional sensors are required. Only the signals of motor armature current, motor angular velocity and axis position are necessary.
- ii. The overall procedure works in real time during normal operation.
- iii. They lead to early and reliable fault detection.
- iv. Efficient fault tolerant control is achieved.

This paper presents a learning methodology, based on nonlinear modeling techniques, fuzzy inference system and the use of heuristic knowledge for detecting faults in rigid link robotic manipulators. The robotic manipulator dynamics prior to the occurrence of a fault are assumed to be known exactly. The main idea behind this approach is to use neural networks to monitor the robotic system for any changes in the dynamics of the system due to faults. By using the approximation capabilities of neural network, the network can be used to generate the residual based on the fault. Then fuzzy inference system is applied to analyze the residuals for decision-making.

3. ROBOT DYNAMICS

The dynamic nonlinear equations of an n degree-of-freedom robot manipulator in the continuous time in closed form can be written as Equation 1.

$$M(q)\ddot{q} + c(q, \dot{q}) + f(\dot{q}) + g(q) = \tau + \tau_d = T_{Model} \tag{1}$$

$$\Delta T = T_{Model} - T_{Measured} \tag{2}$$

Where q and τ are the $(n \times 1)$ vectors of joint position variables and driving joint torques respectively, M is the $(n \times n)$ symmetric positive-definite inertia matrix, c is the vector of Coriolis centrifugal forces, g is the vector of gravitational forces, f is the vector of friction torques (viscous and dry friction), τ_d is a quantity including unmodeled disturbances and T_{Model} the simulating actuating torque. The difference between measured and simulated actuating torque ΔT can represent a fault relevant feature.

4. RESIDUALS GENERATION

We begin by observing that in the absence of modeling uncertainties any off-nominal behavior observed from input–output measurements can be attributed to a fault in the robotic system. Thus, the process of fault detection in the absence of modeling uncertainties can be achieved by approximating, on-line, by neural network. In general, it is not possible to obtain an exact model of a real process only an approximation can be found. Therefore, one has to cope with modeling errors, which degrade the performance of model-based fault detection methods. However, in the presence of modeling uncertainties, the difference in the dynamics could be either due to faults or due to modeling uncertainties Emani-Naeini, et al. (1988). Therefore, a key question is: how does one identify the effects of a fault in the presence of modeling uncertainties? The evaluation is basically a classification problem where the following methods can be applied: threshold logic, stochastic testing and pattern recognition.

Within the last few years, fuzzy logic and neural nets computing have been used as tools for realizing these methods. A straightforward approach, which is often used, is the decision

by a threshold. The stochastic testing method can be applied when much analyzed data is available such that different kinds of conditional probabilities can be calculated. In the technical domain, especially in manufacturing processes, innovation cycles become very short so that it is difficult to build up stochastic data. The method of pattern recognition is used when many characteristic features in complex relations are involved. The right threshold selection is a real bottleneck to diagnosis systems since a trade off exists between a low false alarm rate and a high sensitivity to failures. For unexpected disturbances, some robust selection techniques have been presented in literature. Upper and lower bounds for a threshold are determined in the presence of a priori known model uncertainties by experiments. In (Alessandri et al. 1998), a superior theoretical knowledge and more computational power are required. In practice, it is too easy to obtain heuristic knowledge than theoretical knowledge. The basic idea behind the heuristic knowledge is to utilize experimental knowledge instead of complex mathematical functions. So, the fuzzy logic based inference system has been suggested to handle with the threshold.

5. NEURO FUZZY BASED FAULT DETECTION ARCHITECTURE

In this section a robust FD scheme is proposed for detecting faults in robot manipulators described in Equation 1. The first step is to determine which parameters of the robot are about to be monitored online. While looking for the fault free operation of manipulator, the probability of occurring fault may be in motors and sensors. So the monitoring parameter is velocity, position and torque. The second step in the fault detection system is to generate the residuals by monitoring the robot motion. Multilayer perception feed forward back propagation architecture has been selected. What makes this algorithm different than others is the process by which the weights are calculated during the learning phase of the network. In general, the difficulty with MLP is calculating the weights of the hidden layers in an efficient way that result in the least (or zero) output error; the more hidden layers there are, the more difficult it becomes. To update the weights, one must calculate an error. At the output layer this error is easily measured; this is the difference between the actual and desired (target) outputs. At the hidden layers, however, there is no direct observation of the error; hence, some other technique must be used to calculate the error at the hidden layers that will cause minimization of the output error. The "Figure 2" gives an overall architecture of the fault detection system.

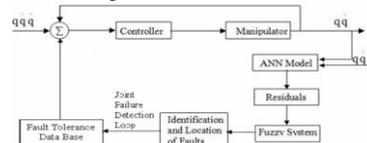


Figure 2: Fault Detection Architecture.

Then third step in fault detection scheme is to evaluate the residuals to take decision on faults. If the thresholds are constant values determined by experimentation on fault-free simulation runs of the robot arm, because of the varying speeds of the joints, the tolerances may need to grow or shrink to provide more adequate checks for failures. To provide the system with variable thresholds, the algorithms must monitor the history of the desired velocities. The longer a certain velocity is maintained, the tighter the threshold can be made up to a base case threshold for the robot.

A time period of widely varying velocities may need to have a relatively loose threshold. Even during normal operation, the difference between the desired and sensed values can be relatively large especially at the beginning of a run before the controller has had time to bring the error under control. Choosing the maximum error found during a fault-free run typically results in a threshold that is so large, it may take several time-steps to notice the error from a broken sensor. By the time the failed sensor is detected, the robot controller has already been infected with the erroneous information and the robot is either off course or has damaged itself. In order to accommodate time varying dependent residual threshold value, fuzzy logic based threshold evaluation has been suggested. Once the fault has been identified and declared, a corrective action can be taken immediately to avoid major damage to the robot as well as the environment in which it resides.

5.1 Neural Network Model for Monitoring the Robot

The most frequently applied neural models are the feed forward perceptron used in multilayer networks, i.e., the Multi-Layer Perceptron (MLP), and the Radial Basis Function (RBF). Both networks are capable of approximating any nonlinear unique static function to arbitrary desired accuracy. This form of mapping is well suited for pattern recognition applications, where both the input vector and the output one represent spatial patterns that are independent of time. The introduction of explicit dynamics into these ANNs requires the spatial representation of time.

The MLP trained with the back-propagation algorithm is a very popular model in neural network and can be used as a practical system for performing a nonlinear input/output mapping of a general nature. For a p-dimensional input vector and a q-dimensional output vector, the MLP input/output relationship defines a mapping from a p-dimensional Euclidean space to a q-dimensional Euclidean output space. Using only one hidden layer, presenting in the nth sample (where n = 1, 2... p), the input vector $I(n) = [X_1(n) X_2(n) \dots X_p(n)]^T$, the activation of the output neuron k (where k = 1, 2... q) is

$$O_k(n) = \phi_k \left[\sum_{j=1}^m W_{kj}(n) \phi_j \left[\sum_{i=1}^p W_{ji}(n) X_i(n) \right] \right] \tag{3}$$

Where m is the number of neurons in the hidden layer, ϕ_{kj} is the weight between the j th neuron of the hidden layer and the k th neuron of the output layer, ϕ_{ji} is the weight between the i th neuron of the input layer and the j th neuron of the hidden layer, ϕ_k is the nonlinear activation function of the output layer and ϕ_j is the nonlinear activation function of the hidden layer. Typically, the network consists of a set of input parameters that constitute the input layer, one or more hidden layers of computation nodes and an output layer of computation nodes as shown in Figure 3.

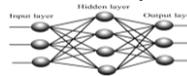


Figure 3: Multi-layer Perceptron Feed Forward Neural network

The input signal propagates through the network in a forward direction on a layer-by-layer basis. MLP's have been applied to solve some difficult and diverse problems by training them in a supervised manner with a highly popular algorithm known as the error back propagation algorithm. Weights connecting neurons in the input layer to those in the hidden layer are adjusted according to:

$$W_{ji}^{new} = W_{ji}^{old} + \eta \sum_{p=1}^q (O_p \delta_p) + \beta [\Delta W_{ji}^{previous}] \tag{4}$$

Where

η is training factor and β is momentum factor δ_p is the error in output layer. The Weights connecting neurons in the hidden layer to those in the output layer are adjusted according to:

$$W_{jk}^{new} = W_{jk}^{old} + \eta \sum_{p=1}^q (O_p \delta_p) + \beta [\Delta W_{jk}^{previous}] \tag{5}$$

In discrete time, the state equation of a fault-free nonlinear dynamic system is given by

$$x(t + \Delta t) = f(x(t), u(t)) \tag{6}$$

Where $x(t)$ is the state vector at time t , $u(t)$ is the applied control vector, Δt is the sample rate and $f(\cdot)$ is the vector-valued nonlinear function of the fault-free system. Considering now that a fault i occurs, the dynamics of the system are modified to

$$x(t + \Delta t) = g_i(x(t), u(t)) \tag{7}$$

Where $g_i(\cdot)$ is the vector-valued nonlinear function of the system affected by fault i . The faults may or may not be additive inputs. The i th fault vector can be defined as the difference between the faulty system dynamics (Equation 4) and the fault-free system dynamics (Equation 5):

$$\psi_i = g_i(x(t), u(t)) - f(x(t), u(t)) \tag{8}$$

Obviously, for the fault-free system $\psi_i(t + \Delta t) = 0$ generally, for each possible fault i , the fault vector ψ_i has a particular behavior, called the fault signature. In real systems and fault free case, the error is due to external disturbances, un-modeled system uncertainty, mapping errors or modeling errors in model-based systems and measurement noise. In this work, an MLP is employed to reproduce (estimate) the fault-free dynamic behavior of a robotic manipulator. The state space dynamic equation of a fault-free robotic manipulator with actuators in each joint is given by:

$$\ddot{x} = \begin{bmatrix} \ddot{q}(t) \\ \ddot{q}(t) \end{bmatrix} = \begin{bmatrix} M^{-1}(q(t)) \left[\tau + \ddot{m} - C(q(t), \dot{q}(t)) - F(q(t)) - G(q(t)) \right] \end{bmatrix} \tag{9}$$

As the joint accelerations are not usually measured in robotic manipulator, this can be obtained by time derivative of velocity as given below:

$$\ddot{q}(t) = (\dot{q}(t + \Delta t) - \dot{q}(t)) / \Delta t \tag{10}$$

and

$$\dot{q} = (q(t + \Delta t) - q(t)) / \Delta t \tag{11}$$

$$\begin{bmatrix} \dot{q}(t + \Delta t) \\ q(t + \Delta t) \end{bmatrix} = \begin{bmatrix} q(t, \Delta t) + \dot{q}(t) \\ M(q(t))^{-1} [\tau + \ddot{a} - C(q(t), \dot{q}(t)) - F(q(t)) - G(q(t))] \end{bmatrix} \quad (12)$$

Thus MLP receives input as a vector formed by joint angle, velocity and torque at time t and outputs a vector of a joint angle and velocity at time (t+ Δt).

5.2 Fuzzy Logic Based Decision Makin

The next step of a fault detection procedure is to evaluate the residuals. This is a decision-making process, which always comes down to threshold logic of a decision function. If there are no uncompensated unknown input effects on the residuals due to a perfect decoupling, then the thresholds diminish to zero. Otherwise, threshold different from zero has to be assigned. Computed torque control has been used for simulating the robot which assumes that the exact model of the robot is known. In general, it is not possible describe the model due to uncertainty in modeling the parameters which are unstructured. In this case perfect decoupling in the residual generation is basically not possible. Hence the residuals or any decision functions built from them always deviate from zero even if no fault is present. Therefore, a decision making process should not show this unstable behavior. Fuzzy inference is a powerful problem-solving methodology with wide application in industrial control and information processing. It provides the simple way to drawn definite conclusions from vague, ambiguous or imprecise. It resembles human decision-making, such that small changes in the observed behavior only result in small effects on the decision, with its ability to work from approximate data and find precise solutions. The fuzzy logic based robust threshold selection for fault detection has an advantage that numerical information is being transformed into qualitative information where human concepts of decision making can be applied. Fuzzy logic allows expressing this knowledge with subjective concepts such as” residual is small” and “residual is medium” which are mapped into exact numeric ranges.

Fuzzy process consists of fuzzification, inference system, and defuzzification. Fuzzification involves converting crisp value into fuzzy value. This is being done by the use of membership function. Inference system involves applying the heuristic knowledge to segregate them into different group. Defuzzification involves again converting the fuzzy value into crisp value for decision-making. Fuzzy logic system evaluates the residual based on the rule fired in the inference system. The inputs to fuzzy logic system are velocity and acceleration and outputs from the fuzzy logic is residual of velocity and joint angle. The residual of joint angle is not considered as important in decision making as the rate of change of residual value due to fault is small as compared to the velocity residual. The universes of discourse of the acceleration, velocity residual are defined as

$$\begin{aligned} \ddot{q} &= \{ \ddot{q}_{\max} \leq \ddot{q}_i \leq \ddot{q}_{\min} \} \\ \dot{q} &= \{ \dot{q}_{\max} \leq \dot{q}_i \leq \dot{q}_{\min} \} \\ \Delta R &= \{ \Delta R_{\max} \leq \Delta R_i \leq \Delta R_{\min} \} \end{aligned} \quad (13)$$

Triangular membership function is selected to map the crisp value into fuzzy value such as low, medium and high. Triangular membership function is generally used when the input-output relation is linear. In the inference system, heuristic knowledge about unstructured disturbances can be considered for generating the rules. For practical processes, knowledge might be available about how non-zero residuals in fault-free situations are related to system states and driving inputs. The knowledge is being built by experience and can be expressed in linguistic terms, e.g. by rules of thumb. The input crisp values are mapped into very low, low, medium, high and very high. And output crisp values are mapped into very small, small, medium, large and very large. A typical rule may be given as

IF; Velocity is Low; AND; Acceleration is medium; THEN; Residual is Large

In the same way, the method may integrate analytical and knowledge-based rules. The key point is to decide the symptoms to be included in each rule. The knowledge of the sensitivity values allows choosing the sensitive and insensitive symptoms in a suitable way.

The entire rule base is created based on our simulation run. In practical, it has to be created by experiment with the real robot. And rule has to be updated as the experience goes on. The fuzzy outputs are defuzzified using weighted average technique which is commonly used for singleton fuzzy output sets.

An optimum threshold R_0 , minimum residual exists in all the cases due to modeling error, has been added, an approach has been suggested by (Schneider, 1993), with an adoptive threshold, ΔR , determined by the fuzzy inference system for the decision making process.

$$R(i,t) = R_0 + \Delta R(i,t) \quad (14)$$

ΔR incorporates the effect of modeling errors. In this, the adoptive threshold is both positive as well as negative so that the actual residual of the manipulator must lie within this band under the fault free condition.

For a practical implementation the required design steps are presented in a compiled form. The main four steps for the adaptive fuzzy based threshold selection can be stated as:

- i. Observing relations between false alarms and characteristic process conditions.
- ii. Formulating rules of thumbs which are organized by if.... then.... structures
- iii. Choosing appropriate fuzzy variables and membership function
- iv. Building a fuzzy rule table based on step 2 and 3

After an initial set-up of membership functions and a fuzzy rule base, further knowledge can be incorporated by changing the rules or by introducing new fuzzy variables if necessary. In this way, unstructured disturbances will incrementally be included into the decision process. Since this concept is based on linguistic variables no theoretical knowledge in mathematics is required but valuable heuristic information can be modified by experienced operational personal.

6. APPLICATION TO AN ROBOT SYSTEM

To evaluate our NF based FD system, a simulation case study has been developed with Scorbot-ER 5u plus. The Scorbot-ER 5u plus is a vertical articulated robot, with five revolute joints Figure 4. With gripper attached, the robot has six degrees of freedom. This design permits the end effector to be positioned and oriented arbitrarily within a large work space. It has base rotation, shoulder rotation, elbow rotation, wrist pitch and wrist yaw motions. The robot's five axes and gripper are operated by DC servo motors. The location and movement of each axis is measured by an electro-optical encoder attached to the shaft of the motor which drives the axis. Advanced robot systems within the next decade are characterized both by a close man-robot partnership for the cooperative management of sophisticated tasks interfaces between robot, sensors and actuators. The corresponding extreme safety and availability requirements can only be guaranteed by the introduction of a structure variable supervisory control concept providing a high structural and parametric flexibility of the suitable controller with respect to malfunctions of system components (sensors, actuators) and stochastic human actions as well. Existing marketable robot control systems can not comply with such enhanced safety and availability features. In order to cope with the above objectives within the framework of the NF based fault tolerant supervisory control concept is going to be introduced which permits to realize both joint based fault detection and isolation functions as well as the corresponding local reconfiguration and mode control. The lower stage of the control structure consists of different sub-controllers which are optimized with respect to specific situations. Characteristic non-fault situation may be the position and velocity controlled free motion mode, the constrained force control mode or the hybrid force-position control mode. Typical fault modes are malfunctions of any joint servo-control system, internal or external sensor faults and danger of collision modes as well as losing parts carried by the robot gripper.

The various controllers which may have any structure (PID, fuzzy control, model based control etc) are activated by the NF based supervisory controller in the upper level of the hierarchy. Depending on the identified situation a smart mode selector (e g Fuzzy or model based) activates the most appropriate control mode which can comprise both the adaptation of control parameter and/or "soft" switching of different mode specific controller structures. The online diagnosis concept for the identification situation or fault classes considers both analytical models as well as heuristic NF- approaches. Characteristic features may comprise amplitudes, trends, Fast Fourier Transformations and Wavelet Functions as well as model based features. Since the robot mechatronics can be well described by Lagrange equations of motion [10] it is useful to assume the difference between model outputs and measured values as additional situation features. For a qualified fine classification of a situation and fault events model and/or heuristic signal based feature extraction will be succeeded by a NF based heuristic feature evaluation.

A practice relevant complex robotic problem which has been selected as benchmark experiments is the “peg into hole” problem. It comprises different situation which are managed by corresponding specific controllers. In situation A the robot moves by position control from a home position to any work piece on a table. In situation B the gripper picks (collision free) by a hybrid force-position control this object and carries it by position control within situation C nearby a hole. Finally within the end situation D by means of hybrid knowledge based hybrid force-position control algorithm the object is fitted into a matching hole.

Besides the nominal process phases A to D several malfunctions of joints, servo control loops or collision scenarios due to unexpected obstacles are going to be considered. The robot is equipped with suitable sensors integrated within the gripper and at the ceiling as well as well as microphone arrays for measuring acoustic signals.

The NF based diagnosis function will be illustrated by the practical case that the robot gripper is carries a mechanical part along any Cartesian trajectory. During the motion by random the carried part will either be lost i. e. it will drop down. Alternatively it will slowly slip away from the gripper and will finally be dragged on ground. In all considered cases typical signal responses both of sensor and microphone can be observed. In the case of part dropping a load step of the sensor and a noise impact signal of the microphone can clearly be seen. In the case of part slipping both a load step followed by an increasing friction force response is measured.

The signal responses are analyzed by the NF based diagnosis module in the succeeding step. The situation responses of sensor and microphone are fed into the NF based evaluation module. The obtained output responses show 4 different cases: the nominal case 1 (hold of part), the drop of part case 2 which corresponds with the ground impact case 3 and finally the slip of part case 4. While the difference between measured force response $T_{Measured}$ and model based force response T_{Model} clearly indicate some malfunction in the robot wrist joint only a succeeding heuristic and NF based feature evaluation can finally distinguished between the different fault classes like a part dropping or collision event. In this work we considered joint malfunction as described in Section 5. The SCORBOT-ER 5u plus was simulated in MATLAB.

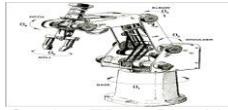


Figure 4: Scorbob ER-5u Plus Manipulator.

6.1 Simulation and Results

An extensive study was performed on designing neural network. Three neural network structures have been studied, each with different number of neurons and hidden layers. The following network has found to be better when compared to other alternatives in terms of Mean Squared Error Deviation (MSED). The neural network has designed with three hidden layer each with 20 neurons. There is no definite way for pre-determining the optimum network structure without trying and testing different configurations and selected based on the applications. The input layer has 15 neurons (5 joint positions, 5 joint velocities, and 5 joint torques measured at t), the output layer has 10 neurons (5 joint positions and 5 joint velocities at $(t+\Delta t)$). The MLP is trained by back propagation algorithm. The training set is formed from 1800 patterns obtained by simulating more than 100 trajectories. A noise is added to bring out the real simulation behavior. After trained, the MLP produces the manipulator dynamical behavior presenting a small residual vector for non-trained fault free trajectories. Figure 5 show the un-normalized MLP outputs of positions of the joints. Figure 6 show the un-normalized MLP outputs of velocities of joints. Figure 7 (a), (b) and 8 (a), (b) shows the residual of positions and velocities for the fault free trajectory respectively. Initial spikes in the velocity diagram is due large uncertainty persist in the model due friction parameters which are high when the manipulator moves from idle condition. From the residual graph Figure 7(a), (b) it is inferred that at higher velocities position residuals are higher due to more vibration of links where as velocity residual is small due to less frictional effect. These rules are framed on the basis of fault free simulation study and are incorporated in the fuzzy inference system to adopt for the different operating conditions.

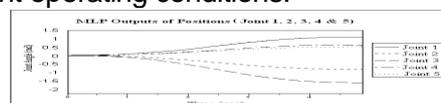


Figure 5: Un-normalized MLP output of positions for a fault free trajectory.

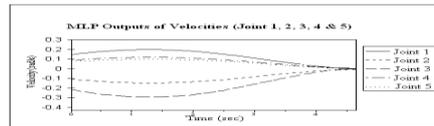


Figure 6: Un-normalized MLP output of velocities for fault free trajectories

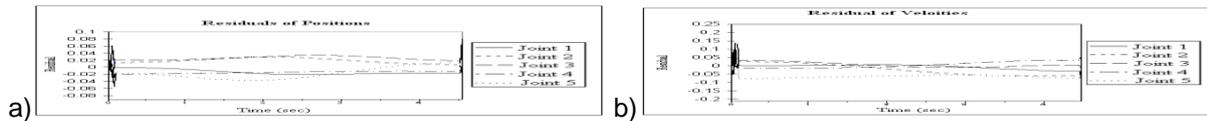


Figure 7: Un-normalized residuals of positions and velocities for fault free conditions.

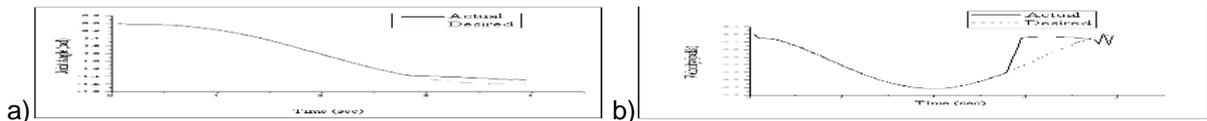


Figure 8: Trajectory with wrist joint fault of joint 4 at time $t = 2.8$ s.

a) Un-normalized position with desired and actual position; b) Un-normalized velocity with desired and actual MLP output.

As the figures are shown in the un-normalized condition, the errors seem to be more. But still it is well below the limit. A sampling time of $T = 0.051s$ can be realized including measurement conversions and evaluation of fuzzy rules. The proposed method is operated on-line such that faults can immediately be detected. A seventh-order polynomial trajectory is imposed at each joint with null initial, final velocities and accelerations. It is predicted that causes to bring about a false are spread over the entire system. Above all, the controller servo actuator can be regarded as the most critical part in the system. Once a fault occur in motor, it is most likely to cause a fatal fault to the entire system.

The Figure 8 (a), (b) shows the trajectory with joint fault at time $T = 2.8$ s. As it is said earlier the output from the neural network is not same as the manipulator. It may be due modeling uncertainties, un-modeled errors, etc. Hence an adaptive threshold is added to compensate for these errors Figure 9 (a), (b) shows the residual of the trajectory with fault. At time $T = 2.8$ s, the output from the neural network goes beyond the adaptive threshold set by fuzzy inference system. Hence fault is identified and isolated. It is not possible to identify the fault using position trajectory as the deviation of residual is very small enough to identify. The Figure 10 (a), (b) shows the fault of joint 4 at time $T = 2.8$ s. Failure of the joint 4 results in drastic decrease of the torque results in abnormal increase of the residual which goes beyond the adaptive threshold leads to detection of failure. While the differences between torque ΔT clearly indicate some malfunction occur in the robot wrist joint. Then the heuristic can finally distinguished between the different fault classes and taking a decision making based NF feature evaluation.

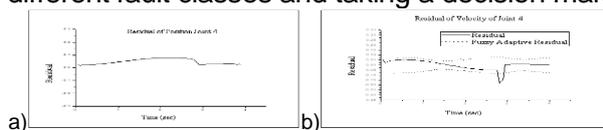


Figure 9: Residuals of the trajectory with fault presented in graph (the dashed line represents the adaptive threshold given by fuzzy inference system).

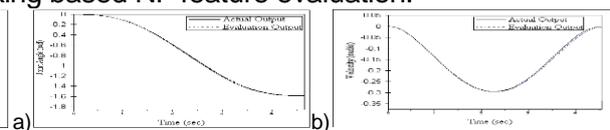


Figure 10: Robot fault detection based on a NF feature evaluation trajectory with locked joint fault of joint 4 at time $t = 2.8$ s.

a) Un-normalized position with desired and NF feature evaluation position output,
 b) Un-normalized velocity with desired and NF feature evaluation MLP output.

7. CONCLUSION

In this paper, a new structure variable NF based supervisory control concept for five axis Scorbot 5u plus robots is presented. It relies on a reliable identification of the actual process phases or fault cases. For each diagnosis phase and fault scenario by means of a heuristic fuzzy logic mode selector a dynamic reconfiguration of optimal situation-specific fault tolerance rule base will be activated. A basic requirement for FD is a reliable detection of even small faults. Hence the FD based robust FD system has been proposed using advantage heuristics knowledge to make decision and ANN to reproduce robot action. The degree of adaptation depends on characteristic process conditions which are extracted by observation and experience. The proposed FD can be used online for collecting the symptoms and performing fault identification by comparison with fault signatures. Using the heuristic approach a performance improvement for fault detection in processes with unstructured disturbances can be achieved. Simulation examples are used to illustrate the effectiveness of the algorithm in detection and tolerate of faults in a five axis robotic

manipulator. Untrained trajectories are considered for simulating the system. Training of neural network is more important in the functioning of system. Two faults with different trajectory has been considered to each with varying velocity profile. The results indicate that the NF based FD model approach is a valuable technique for detection and tolerate of faults occurring in the robotic manipulator system. The NF model approach FD has proved to be extremely robust to noise while still providing isolation to relatively small fault acting on the manipulator joints. It is observed that the NF model is a powerful mean of FD tasks, robust performing both residual generation and residual analysis. The acceptance of random trajectories would be desirable, training time and computational effort that would be required to train an appropriate network off line decreases as the task becomes simpler. Further work on this is to extend the FD scheme to robotic manipulators with a larger number of degrees of freedom. Other different types of faults can be detected and tolerated. Fault detection and toleration of robotic manipulator via adaptive NF, hybrid system and expert system. The work presented here can be expanded to include post failure control of the robotic manipulator in a hybrid system frame work. The effectiveness of the NF system relies on the rule base and it become complex as the number of faults considered become more.

REFERENCES

- [1] Alessandri, P. T., (1997), "Model-Based Fault Diagnosis Using Nonlinear Estimators: A Neural Approach," In Proceedings American Control Conference, pp. 903–907
- [2] Basseville, M. and Nikiforov, I.V., (1993), "Detection of Abrupt Changes—Theory and Application", Englewood Cliffs, NJ: Prentice-Hall
- [3] Chen, J., and Patton, R. J., (1999), "Robust Model Based Fault Diagnosis for Dynamic Systems", Dordrecht: Kluwer Academic Publishers
- [4] Clark, R.N., (1989), "State Estimation Schemes for Instrument Fault Detection", In R. J. Patton, P.M. Frank and R.N. Clark (Ed), Fault Diagnosis in Dynamics Systems, Theory and Applications, Prentice-Hall, 1989
- [5] De Luca, A. and Mattone, R., (2003), "Actuator Failure Detection and Identification Using Generalized Momenta", Proceedings of IEEE International Conference on Robotics and Automation, Taiwan
- [6] Deckert, J. C., Desai, M. N., Deyst, J. J and Willsky, A. S., (1977), "F-8 DFBW Sensor Failure Identification Using Analytic Redundancy", IEEE Transaction Automation and Control, Vol. Ra-22, pp. 795-803
- [7] Desai, M. and Ray, A., (1981) "A Fault Detection and Isolation Methodology," In Proceedings of 20th Conference on Decision and Control, pp. 1363-1369
- [8] Dev Anand. M, Selvaraj. T, Kumanan. S and Austin Johnny. M, (2006), "Neuro Fuzzy Based Fault Diagnosis System for Robot Manipulator ", CARs & FOF, July19-22, VIT, Vellore, India
- [9] Ding, X. and Frank, P. M., (1991), "Fault Detection and Identification via Frequency Domain Observation Approaches", In Mathematical and Intelligent Models in System Simulation, Ed. R. Hanus, P. Kool and S. Tzafestas. J. C. Baltzer AG Sc. Publ. Co., IMA CS, pp.471-476
- [10] Elias, N., Skoundrianos, S. and Tzafestas, G., (2004), "Finding Fault" IEEE Transaction Robotics & Automation
- [11] Emani-Naeini, A., Muhammad, M and Rock, S. M., (1988), "Effect of Model Uncertainty on Failure Detection: The Threshold Selector", Proceedings of IEEE Transaction Automation and Control, Vol. 33
- [12] Evans, F. A. and Wilcox, J. C., (1970, "Experimental Strap Down Redundant Sensor Inertial Navigation System," Journal of Spacer Rockets, Vol. 7, pp.1070-1074
- [13] Frank, P.M. and Kiupel, N., (1992), "Fuzzy Supervision for Lean Production", In 6th Institute for a Romanion and Robotic IAR Colloquium, Duisburg, Germany.
- [14] Frank, P.M., (1990), "Fault Diagnosis in Dynamic Systems Using Analytical and Knowledge-Based Redundancy A Survey and Some New Results," Automatica, Vol. 26, No. 110. 3, pp. 459-474
- [15] Frank, P.M., (1993), "Advances in Observer-Based Fault Diagnosis", International Conference on Fault Diagnosis, Toulouse, France
- [16] Frey, C. W., Kuntze, H. B., (2001), "A Neuro-Fuzzy Supervisory Control System for Industrial Batch Processes", IEEE Transactions on Fuzzy Systems, Vol.9, No.4, pp.570-577
- [17] Gilmore, J. P. and McKern, R. A., (1972), "A Redundant Strap Down Inertial Reference Unit (SIRU)," Journal of Spacer Rockers, Vol. 9, pp. 39-47

- [18] Gustafson, D., Kessel, W. (1979), "Fuzzy Clustering with a Fuzzy Covariance Matrix", Proceedings of the IEEE CDC, San Diego, CA, USA, pp. 761–766
- [19] Hamilton, D. L., Bennett, J. K. and Walker, I. D. (1992), "Parallel Fault Tolerant Robot Control," SPIE Conference on Cooperative Intelligent Robot in Space, pp. 251-261, Boston, MA
- [20] Henrique, M. T. and Tinos R., (2001), "Fault Detection and Isolation in Robotic Manipulators via Neural Networks: A Comparison among Three Architectures for Residual Analysis", Journal of Robotic Systems, Vol.18, No. 7
- [21] Isermann and Freyermuth, B., (1991), "Process Fault Diagnosis Based on Process Model Knowledge -Part I: Principles for Fault Diagnosis with Parameter Measurement, and Control", ASME Journal of Dynamic Systems, Vol. 113, pp. 620-626
- [22] Isermann, R., (1984), "Process Fault Diagnosis Based on Modeling and Estimation Methods - A Survey", Automatica, Vol. 20, pp. 387 – 404
- [23] Isermann, R., (1990), "Model Based Fault Diagnosis and Supervision of Machines and Drives," In Proceedings Triennial World Congress, pp. 1-12
- [24] Jang, J.S.R., (1994), "Structure Determination in Fuzzy Modeling: A Fuzzy CART Approach", IEEE International Conference on Fuzzy Systems, Orlando, Florida
- [25] Paredis, C. J. J., Au, W. K. F. and Khosla, P. K., (1994), "Kinematic Design of Fault Tolerant Manipulators," International Journal of Computer and Electrical Engineering, Vol. 20, No. 3, pp. 211-220
- [26] Patton, R. J., Frank, P. M. and Clark, R. N., (200), "Issues of Fault Diagnosis for Dynamic Systems, Springer Verlag, London Ltd.," ISBN 3-540-19968-3
- [27] Pdrs Mal. (1988), "Payload Deployment and Retrieval System Malfunction Workbook," Technical Manual, NASA Johnson Space Center, Houston, TX
- [28] Pham, H. E., (1992), "Fault-Tolerant Software Systems: Techniques and Applications. Los Alamitos, CA: IEEE Computer Society
- [29] Sajidman, M., Kuntze, H.-B. and Jacobasch, A., (1995), "A Fuzzy Logic Concept for Highly Fast and Accurate Position Control of Industrial Robots", Proceedings of IEEE International Conference on Robotics and Automation, Nagoya, Japan, pp. 21-27.
- [30] Sauter, D., Dubois, G., Levrat, E. and Bremont, J., (1993), "Fault Diagnosis in Systems Using Fuzzy Logic", In EUFIT'93, First European Congress on Fuzzy and Intelligent Technologies, Germany
- [31] Schneider, H. and Frank, P.M., (1994), "Observer-Based Supervision and Fault Detection for Robots", In International Conference on Fault Diagnosis, Toulouse, France.
- [32] Schneider, H., (1993), "Implementation of a Fuzzy Concept for Supervision and Fault Detection of Robots", In EUFIT'93, First European Congress on Fuzzy and Intelligent Technologies, Germany
- [33] Sreevijayan, D., Tosunoglu, S. and Tesar, D. (1994), "Architectures for Fault Tolerant Mechanical Systems," IEEE Mediterranean Electro Tech Conference, pp. 1029-1033
- [34] Stengel, R. F., (1991), "Intelligent Failure-Tolerant Control," IEEE Control System Magazine, Vol. 11, No. 4, pp. 14-23
- [35] Ting, S. T. and Tesar, D., (1993), "A Control Structure for Fault Tolerant Operation of Robotic Manipulators," Proceedings of IEEE International Conference on Robotics and Automation, pp. 484 - 490
- [36] Tso, K. S., Hecht, M. and Marzwell, N. I., "Fault Tolerant Robotic System for Critical Applications," I Proceedings of IEEE International Conference on Robotics and Automation, pp. 691-696
- [37] Uppal, F.J., Patton, R.J. and Witczak, M., (2003), "A Hybrid Neuro-Fuzzy and De-coupling Approach Applied to the DAMADICS Benchmark Problem", IFAC Symposium: SAFEPROCESS 2003, Washington, DC
- [38] Vesley, W. E., Goldberg, F. F., Roberts, N. H. and Haasi, D. F., (1981), "Fault Tree Handbook," NUREG 0492, Systems and Reliability Research Office of Nuclear Regulatory Research, US Nuclear Regulatory Commission, Washington.
- [39] Viswanadham, N., Taylor, J. H. and Luce, E.C., (1987), "A Frequency Domain Approach to Failure Detection and Isolation with Application Control Theory and Advanced Technology", Vol. 3, pp. 45-72
- [40] Watanabe, K and Himmelblau, D. M., (1982), "Instrument Fault Detection in Systems with Uncertainties." International Journal of Scientific Research, Vol. 13, pp 137-158.
- [41] Wiinnenberg, J., (1990), "Observer-Based Fault Decision Dynamic Systems", Fortschrittsberichte VDI, Reihe 8, Steuerungs- und Regelungstechnik, No22.
- [42] Zadeh, L. A., (1965), "Fuzzy Sets", Information and Control, Vol. 8