All Sciences Proceedings
http://as-proceeding.com/

2nd International Conference on Engineering, Natural and Social Sciences

April 4-6, 2023 : Konya, Turkey

https://www.icensos.com/

FAULT TOLERANT CONTROL OF ROBOT MANIPULATORS WITH MANFIS AND GAIN SCHEDULING

Tolga YUKSEL[1*]

[1]*Dept. of Electrical&Electronics Engineering, Bilecik Seyh Edebali University, Bilecik-Turkey*

[*]*tolga.yuksel@bilecik.edu.tr*

Abstract **–** While robot manipulators cooperate with other manipulators or working with high accuracy in partnership with people, malfunctions or fault may occur. The robotic systems must be able to isolate and tolerate these faults with the designed controllers. In this study, a multiple-ANFIS-based fault diagnosis approach that converges to the failure function for manipulators is proposed and the coefficients of the Computed Torque PID (CT-PID) controller defined for manipulator control are varied using Gain Scheduling method as the fault tolerant control approach. Simulation results for a two DOF robot manipulator with dynamic uncertainties proves the robustness of the proposed system.

Keywords – Robot Manipulators, Computed Torque PID, Fault Diagnosis

I. INTRODUCTION

In the last thirty years, it has become a sought-after feature for modern control systems to find possible faults, isolate them and continue to work robustly by coping with the failure, as well as their high precision and repeatable operation features [1]. Robot manipulators perform many tasks that can be considered dangerous for humans, such as cleaning in nuclear power plants, data and sample collection in submarine and space travels, maintenance, as well as applications where they can work faster, more difficult and more accurately than humans, such as mass production lines. As a result, a malfunction that may occur in a robot manipulator on a mass production line may cause a pause in all connected lines, even in flexible automation systems, or the smallest unnoticed failure or mistake that may occur during the performance of especially dangerous tasks may result in the cancellation of the entire task with large monetary costs [2]. To avoid these catastraophic results, fault detection and isolation (FDI) and fault tolerant control (FTC) design have become an important issue.

FDI schemes defined for robot manipulators are adaptations of schemes defined especially for nonlinear systems. Although some of these schemes focus on analytically defined nonlinear observers, simulation studies and implementations have shown that it is difficult to obtain analytical models of nonlinear systems such as robot manipulators in cases involving uncertainties and noise [3-5]. By the way, studies have shifted towards artificial intelligence methods. Neural networks (NNs) accepted for FDI in nonlinear systems has also been used for robot manipulators. Naughton et al. used the nonlinear observer proposed by Adjallah et al. for residual generation, and they utilized NNs for residual evaluation [6-7]. Vemuri and Polycarpou considered the fault as a component of the robot model function and used the adaptive learning approach defined for NN to approximate to the fault function [8]. Terra and Tinos used various types of NN architectures for both residual generation and residual evaluation [9]. Lee et al. proposed fault detection with parameter estimation methods and fault isolation using ART type NN for component and sensor type faults [10].

Although it is tried to acquire fault tolerance by using robust control methods in passive types, in active FTC schemes, which are the extension of FDI schemes, it is aimed to complete the task by using information about the fault. Defined FTC schemes for robot manipulators are grouped under two headings: hardware redundancy and control methods. The studies of Paredis and Khosla on hardware redundancies and FTC explained the steps to be followed [11-12]. Izumikawa et al. designed a two-controller FTC scheme for a flexible manipulator that will operate in the healthy and faulty state [13]. In addition, in another study, they designed an FTC scheme with PD controller that will change the coefficients in healthy and faulty conditions [14]. Tosunoğlu proposed FTC designs with both redundant and sliding mode and adaptive controllers for faults that may occur in robot manipulators [15]. Zhang et al. have designed an FTC that provides stability guarantee for robot manipulators before the fault and tries to balance the fault by evaluating the obtained fault information with fuzzy logic (FL) [16]. Kuntze et al. have designed an FTC for a human-interactive mobile manipulator that uses several mixed (PID, FL, and Model-Based Controller) subcontrollers and makes their fault-state selection by a fuzzy state selector [17].

In this study, partial joint failures that can occur in manipulators and more difficult to detect are the main subject. A model-based FDI block that approximates the fault function that may occur at each joint and a fault-tolerant controller that varies the PID coefficients of the defined Computed Torque-PID (CT-PID) controller using the fault approximation function with the Gain Scheduling (GS) Method is proposed. For fault approximation, the ANFIS structure that combines the positive aspects of NN and FL and its Multiple-ANFIS (M-ANFIS) structure with multiple outputs are used. The obtained approximation function changes the defined PID coefficients with the defined update rule and enables the manipulator to operate with lower errors after fault. The details of this proposed FDI and FTC scheme are explained in the following sections and simulation results are given.

II. Fault Detection and Fault Function Approximation with M-ANFIS

The proposed FTC scheme with the FDI and FTC blocks is shown in Figure 1. As the first –block the proposed systems uses ANFIS which is a hybrid structure that combines the self-learning features of NNs used for fault detection and approximation to fault function and FL used for the user experience [18]. The biggest weakness of this structure is that it has multiple - input - single - output. M-ANFIS structure, in which independent ANFISs are gathered is used for modeling multi-output systems.

The unit sampling step in time delayed angular position and velocity values $(q(t-\Delta t), \dot{q}(t-\Delta t))$ and instantaneous torque value $(\tau(t))$ are defined as inputs of the FDI block. This block gives the instantaneous angular velocity and position estimation values of each joint as output. $(\hat{q}(t), \dot{\hat{q}}(t))$

For model-based FDI, fault detection is implemented by obtaining the difference signs, called residuals, between the model and the real system, and comparing them with the determined lower and upper thresholds. Fault detection ratios obtained as a result of this comparison process do not always reach a sufficient level. Some independent signals must be derived to aid this process. This can be done using the analytical redundancy, which is expressed in defining a

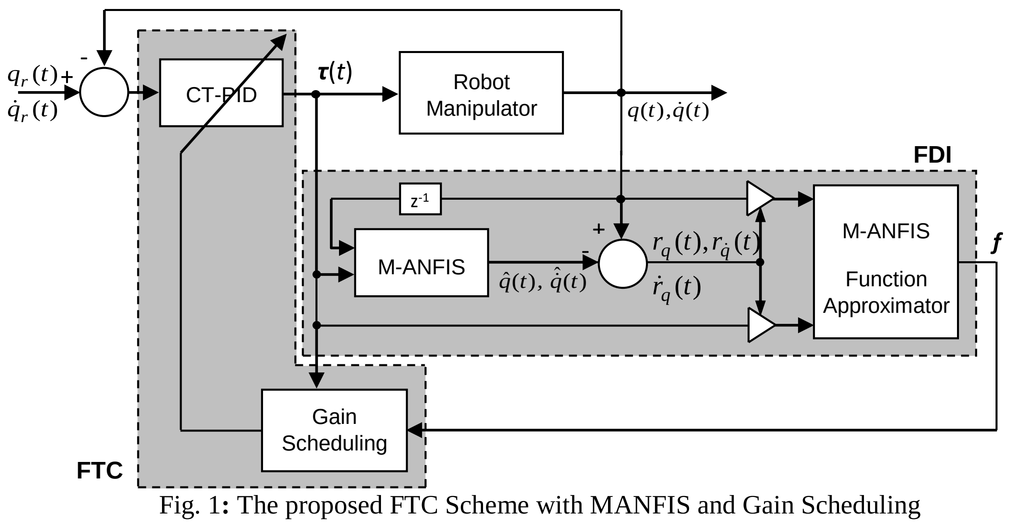

Fig. 1: The proposed FTC Scheme with MANFIS and Gain Scheduling

variable in more than one way [5]. When the derivative value of the position residual is obtained, it is clear that this must be mathematically equivalent to the velocity residual. This whole process are given below:

$$
\begin{aligned}
& r_q(t) = q(t) - \hat{q}(t) \\
& r_{\dot{q}}(t) = \dot{q}(t) - \hat{\dot{q}}(t) \\
& (r_q(t))' = (q(t) - \hat{q}(t))' = \dot{q}(t) - \dot{\hat{q}}(t)
\end{aligned} \tag{1}
$$

$$
\left.
\begin{aligned}
& \boldsymbol{(r_q(t)} > \text{high\_e}_{r_q}\boldsymbol{)} \vee \boldsymbol{(r_q(t)} < \text{low\_e}_{r_q}\boldsymbol{)} \vee \\
& \boldsymbol{(r_{\dot{q}}(t)} > \text{high\_e}_{r_{\dot{q}}}\boldsymbol{)} \vee \boldsymbol{(r_{\dot{q}}(t)} < \text{low\_e}_{r_{\dot{q}}}\boldsymbol{)} \vee \\
& \boldsymbol{(r_q(t))'} > \text{high\_e}_{(r_q)'}\boldsymbol{)} \vee \boldsymbol{(r_q(t))'} < \text{low\_e}_{(r_q)'}\boldsymbol{)}
\end{aligned}
\right\} \Rightarrow \begin{bmatrix} \text{Fault} \\ \text{Alarm} \end{bmatrix} \tag{2}
$$

Fault function approximation, which follows fault detection, is implemented in a different way. In the scheme, firstly, residuals and analytical redundancies are given as input to the approximator structure to approximate to the fault function, but the desired performance could not be obtained. Then, instead, the existing instantaneous torque, angular position and velocity signs were applied as inputs. First, an NN was chosen as a fault function approximator, and then an M-ANFIS structure was preferred since sufficient fault isolation performance could not be obtained. The inputs become active with the fault detection process and function approximator M-ANFIS is designed to give a fault function on the output of the failed joint as output, zero on the other outputs. This is clearly illustrated in Figure 2.

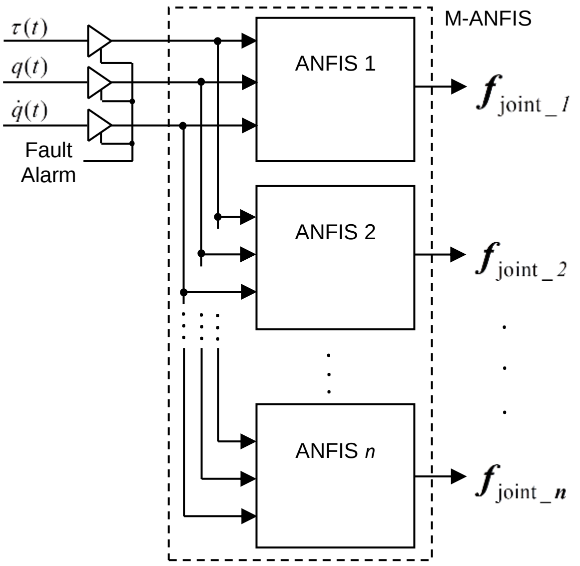

Fig. 2: Fault function approximation with M-ANFIS

III. Fault Tolerant Control with Gain Scheduling

GS was used as the FTC method in the proposed scheme. According to this method, the controller parameters that control the system can be changed according to any rule defined depending on the system state, according to a defined schedule or with a defined FL structure. In this way, both a non-linear controller and a controller sensitive to the state regions of the system can be obtained. The proposed scheme uses the CT-PID method for manipulator control [19] and this method is defined by (3):

$$
\begin{aligned}
& e(t) = q_d(t) - q(t) \\
& \dot{e}(t) = \dot{q}_d(t) - \dot{q}(t) \\
& \dot{\varepsilon}(t) = e(t) \\
& \tau = M(q(t))(\ddot{q}_d(t) + K_v\dot{e}(t) + K_p e(t) + K_i\varepsilon(t)) + V(q(t),\dot{q}(t)) + G(q(t))
\end{aligned} \tag{3}
$$

By updating the K_P, K_I, K_D coefficients with the GS method, it ensures that the manipulator operates with lower error values in the fault state. Here, according to which rule the coefficients will be updated is determined by the user. Before giving the rule for the proposed FTC scheme, how the robot dynamics changes in case of partial joint fault is given by (4):

$$
M(q(t))\ddot{q}(t) + N(q(t),\dot{q}(t)) = \tau(t) - \alpha.u(t-T).\tau(t) \tag{4}
$$

where M is the inertia matrix, where n is the number of joints, N is the combination of centripetal and gravity vectors, τ is the torque applied to the joints, α is the partial joint fault matrix, u is the unit step, and T is the time instant of the fault. Based on (4), the GS rule is defined by (5):

$$
\boldsymbol{K}_{\boldsymbol{update}} = \boldsymbol{K} + \boldsymbol{K}.\frac{\boldsymbol{f}_{\text{fault}}}{\boldsymbol{\tau}_{\text{faulty joint}}} = \boldsymbol{K} + \boldsymbol{K}.\text{FTC\_ratio} \tag{5}
$$

This rule is proposed using the dynamics in (4) and the approach from the CT-PID method that the increase in the torque value of the faulty joint should also be reflected to the PID coefficients. The rule also brings two problems with it. The first problem is how often to update the coefficient. Updates can be made once or at desired second intervals after fault isolation, if desired. The choice at this point is determined by the change in the amount of error due to settling in each coefficient change and the settling time. With the continous change of the coefficients, both the settling time increases and the error amounts increase due to the overshoots due to the temporary period that occurs with each update.

The second problem that arises is the increase in the amount of error due to the sudden realization of the coefficient variation. To solve this problem, a rule that takes the coefficients to the desired value over time is considered and implemented. This rule and related simulation results will be given in the next section.

IV. SIMULATION RESULTS

The simulation of the proposed FTC scheme is implemented using MATLAB and *Fuzzy Logic Toolbox*. As the robot manipulator defined in the scheme, a two-link planar manipulator that is under gravity with center of gravities defined at the end of each link is used [19]. The generalized representation of dynamic equations of the robot manipulator is defined as follows:

$$\begin{aligned}
&M_{11} = (m_1 + m_2)a_1{}^2 + m_2 a_2{}^2 + 2m_2 a_1 a_2 \cos(q_2(t)) \\
&M_{12} = M_{21} = m_2 a_2{}^2 + m_2 a_1 a_2 \cos(q_2(t)),\ M_{22} = m_2 a_2{}^2 \\
&V_1 = -m_2 a_1 a_2 (2\dot{q}_1(t)\dot{q}_2(t) + \dot{q}_2(t)^2)\sin(q_2(t)) \\
&V_2 = m_2 a_1 a_2 (\dot{q}_1(t))^2 \sin(q_2(t)) \\
&G_1 = (m_1 + m_2) g a_1 \cos(q_1(t)) + m_2 g a_2 \cos(q_1(t) + q_2(t)) \\
&G_2 = m_2 g a_2 \cos(q_1(t) + q_2(t)) \\
&M(q(t)) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix},\ V(q(t),\dot{q}(t)) = \begin{bmatrix} V_1 \\ V_2 \end{bmatrix},\ G(q(t)) = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix}
\end{aligned} \quad (6)$$

$$M(q(t))\ddot{q}(t) + V(q(t),\dot{q}(t)) + G(q(t)) = \tau \quad (7)$$

where, t is time, $q(t), \dot{q}(t) \in \Re^2$ is the angular position and velocity vectors of the joints, $\tau \in \Re^2$ is the moment vector applied to the joints, $M(q(t)) \in \Re^{2\times 2}$ is the inertia matrix and $V(q(t),\dot{q}(t)) \in \Re^2$ is the centripetal and centrifugal vector and $G(q(t)) \in \Re^2$ is the gravity vector.

Link weights are $m_1 = m_2 = 1 kg.$, limb lengths are $a_1 = a_2 = 1 m.$, sampling frequency 100 Hz, and $q_1(0) = 0.1\ rad, q_2(0) = 0\ rad$ as the initial angular positions of the manipulator joints. Friction effects are ignored in dynamics. In addition, to prove the robustness of the defined scheme, the uncertainties η_M in the inertia matrix, η_V in the centripetal and centrifugal vectors, η_G in the gravity vector, are added with a time-varying uncertainty of 5% and the related definition is given by (8):

$$\begin{aligned}
&\|M(q(t)) + \eta_M\| \le 1.05M(q(t)) \\
&\|V(q(t),\dot{q}(t)) + \eta_V\| \le 1.05V(q(t),\dot{q}(t)) \\
&\|G(q(t)) + \eta_G\| \le 1.05G(q(t))
\end{aligned} \quad (8)$$

The gain matrices of the CT-PID method are defined as $K_v = 100. I_{2\times 2}$, $K_p = 20.I_{2\times 2}$, $K_i = 500\ I_{2\times 2}$.

Partial joint faults for the manipulator as 1st joint 70% (f_1), 1st joint 50% (f_2), 30% (f_3), 2nd joint 70% (f_4), 2nd joint 50% (f_5), 2nd joint 30% (f_6), are defined as torque losses.

To model the defined robot manipulator, the M-ANFIS is formed of 4 independent that give the instantaneous angular position or velocity estimation signal of the relevant joint. Each ANFIS has 6-input that receives instantaneous torque and unit-step delayed angular position and velocity signals of both joints.

For training, 146 different trajectories with amplitudes between ±1 and sin- and cos- shapes are simulated, and 76 samples of these simulations are used in the training of each ANFIS. Each ANFIS was defined with two generalized-bell-shaped membership functions for each input and a hybrid learning algorithm was used as the learning algorithm.

80 of the 146 trajectories defined for residual and analytical redundancy derivation are selected, and simulations under healthy conditions are conducted for these trajectories. Since the robot manipulator must take a value in a certain range due to the uncertainty of the residual values in the no-fault condition, the lower and upper threshold values of these residuals were obtained with the help of simulations. Any of the residuals exceeding these threshold values at any time is perceived as a fault detection signal.

To approximate to the fault function, an M-ANFIS consisting of 2 independent ANFIS, each giving the fault function of a joint, is proposed. This M-ANFIS uses instantaneous torque, angular position and velocity signals as inputs as described in Section 2.

Although the partial faults studied are considered as continuous type faults, the designed scheme continues to monitor the residuals continuously after the fault detection. For this reason, M-ANFIS, which approximates to the fault function following fault detection, is is set to activate 3.2 sec. later.

It has been observed that the fault isolation performance of M-ANFIS outputs decreases especially in faults containing high frequency component, and in order to correct this situation, the continuity of the outputs is accepted as a criterion. To search for continuity in M-ANFIS outputs, signals shorter than 0.8 s are ignored, and signals longer than this time are transferred to the outputs.

With these defined features, a fault isolation percentage of 87.083% is obtained for the defined faults and given trajectories. The effects of the FTC on this percentage are described below.

The simulation results obtained for the update frequency problem brought by the GS rule defined with (5) showed that with the update made once, the transient and steady state errors after the fault reach the desired level. The comparison table about the update frequency is not included due to space constraints.

The second problem is the increase in the amount of error due to the sudden change of the coefficients. To solve this, a PID variation rule has been defined that progresses by increasing exponentially, which is found suitable as a result of the trials, depending on time, and this rule is given by (9):

$$K(t) = \begin{cases} K + K * artim * (1 - e^{-(t - t_{\text{detection}})}) & K(t) < K_{güncel} \\ K_{güncel} & K(t) \geq K_{güncel} \end{cases} \quad (9)$$

For the proposed scheme, as an example, for the joint trajectories defined by (9), 50% failure (f_2) is simulated in Joint 1 at t = 24 sec:

$$\begin{aligned} q_{d1}(t) &= 0.8 * \cos(t/2)\ rad. \\ q_{d2}(t) &= 0.8 * \sin(t/2)\ rad. \end{aligned} \quad (10)$$

For the example defined by (10), the trajectories followed by the joints, the torques applied to the joints, the residuals and analytical redundant signals, the approximation signals of the M-ANFIS outputs to the fault function are given in Figure 3, Figure 4, in Figure 5, and in Figure 6, respectively.

V. Conclusion

In this study, an FTC scheme that provides fault-tolerant control with model-based FDI and CT is proposed. With this scheme, fault approximation

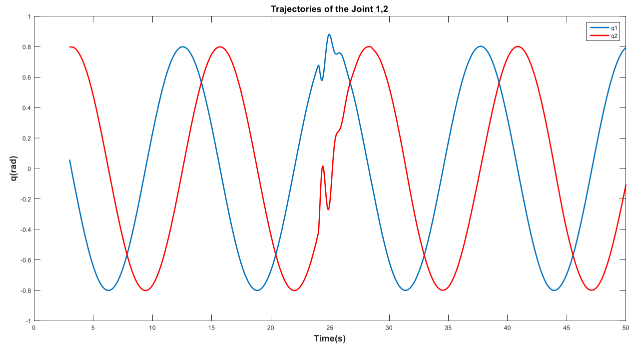

Figure 3: Trajectories of the joints

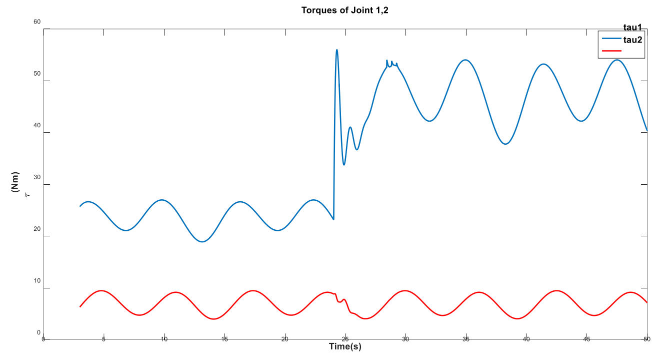

Figure 4: Torque demands for the joints

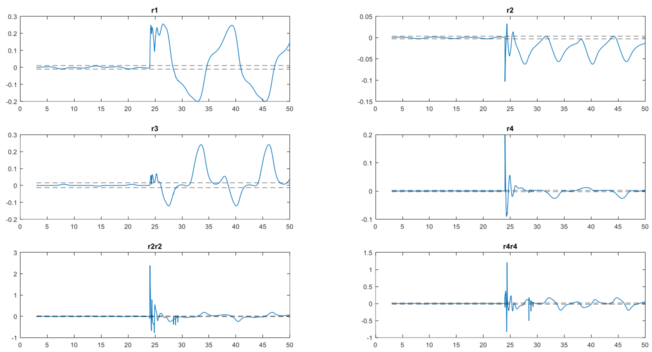

Figure 5: Residuals and redundant signals

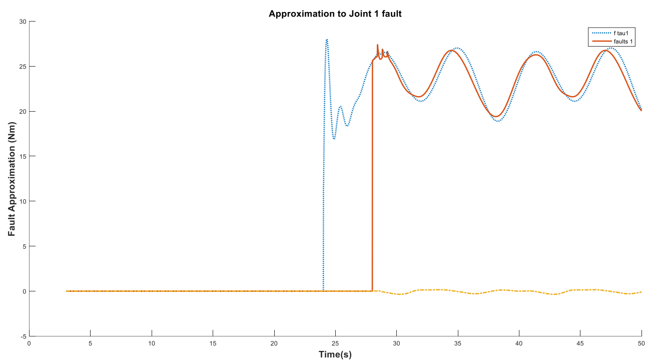

Figure 6: Fault approximation for Joint 1

without too much delay, robustness, suitability for real-time operation depending on computational load, and convenience in defining new trajectories and faults are provided.

In addition, the delay, continuity test and high frequency torque signals required for approximation to the fault are the points that should be considered both in the simulation phase and in the use of the scheme in real time and on a real system.

For the scheme designed in further studies, it is aimed to propose an FL rule-based CT and to test the schema on a real robot manipulator.

ACKNOWLEDGEMENT

This study was supported by Ondokuz Mayıs Üniversitesi Research Fund with the number, PYO.MUH.1904.09.001.

REFERENCES

[1] Isermann, R., Ballé, P., 1997, "Trends in the application Of Model-based Fault Detection And Diagnosis of Technical Processes", Control Eng. Practice, Cilt 5, No 5, 709-719

[2] M. L. Visinsky ," Fault Detection And Fault Tolerance Methods For Robotics",M. Sc. Thesis, Rice University, 1991

[3] J.R. Cavallaro, I.D. Walker, "A survey of NASA and military standards on fault tolerance and reliability applied to robotics", Proc.AIAA/NASA Conf. on Intelligent Robots in Field, Factory,Service and Space, 1994, pp.282-286

[4] F. Caccavale, I.D. Walker, "Observer-based fault detection for robot manipulators", IEEE Int. Conf. on Robotics and Automation, pp:2881-2887, 1997

[5] M. L. Leuschen, I. D. Walker, J. R. Cavallaro, "Fault residual generation via nonlinear analytical redundancy", IEEE Trans. on Control System Technology Cilt: 13, No: 3, s: 452-458, 2005

[6] K. Adjallah, D. Maquin, J. Ragot, "Nonlinear observer-based fault detection", Proc. of The Third IEEE Conf. on Control Applications, s: 1115-1120, 1994

[7] J. M. Naughton, Y. C. Chen, J. Jiang, "A neural network application to fault diagnosis for robotic manipulator", IEEE Int. Conf. on Control Applications s: 988-993, 1996

[8] A. T. Vemuri, M. M. Polycarpou, "Neural-network-based robust fault diagnosis in robotic systems", IEEE Trans. on Neural Networks, Cilt: 8, No:6, s:1410-1420, 1997

[9] M. H. Terra, R. Tinos, "Fault detection and isolation in robotic manipulators via neural networks: a comparison among three architectures for residual analysis", Journal of Robotic Systems, Cilt:18, No:7, s: 357-374, 2001

[10] I. S. Lee, J. T. Kim, J. W. Lee, D. Y. Lee, K. Y. Kim, "Model-Based Fault Detection and Isolation Method Using ART2 Neural Network", International Journal Of Intelligent Systems, Cilt:18, s:1087–1100, 2003

[11] C. J. J. Paredis, P. K. Khosla, "Design of Modular Fault Tolerant Manipulators," The Algorithmic Foundations of Robotics, p. 373-383, 1995

[12] C. J. J. Paredis, P. K. Khosla, "Fault Tolerant Task Execution through Global Trajectory Planning," Reliability Engineering and System Safety (special issue on Safety of Robotic Systems), Vol. 53, No. 3, s: 225-235, 1996b

[13] Y. Izumikawa, K. Yubai, T. Hori, "Vibration Suppression Control of Flexible Arm Robot by PD Gain Switching Considering Sensor Failure", IEEE Int. Conf. on Industrial Technology, Cilt:2, s: 684 - 689, 2002

[14] Y. Izumikawa, K. Yubai, J. Hirai "Fault-tolerant Control System of Flexible Arm for Sensor Fault by Using Reaction Force Observer", IEEE Int. Conf. On Advanced Motion Control, s:583-588, 2004

[15] S Tosunoglu, "Fault-Tolerant Control of Mechanical Systems" , IEEE Int. Conf. On Industrial Electronics, Control and Instrumentation , Cilt: 1, s:127 - 132, 1995

[16] X. Zhang, L.F. Ramos, K. Rattan, "Adaptive Fault-Tolerant Fuzzy Control of Robotic Systems", NAFIPS2009, 2009

[17] H.B. Kuntze, C. Frey, K. Giesen, G. Milighetti, "Fault Tolerant Supervisory Control of Human Interactive Robots". ACD Workshop 2003, s: 55-60, 2003

[18] R. Jang., C. T. Sun, E. Mizutani , Neuro-Fuzzy and Soft Computing, Prentice-Hall Inc., 1997

[19] F.L. Lewis, C.T. Abdallah, D.M. Dawson, Control of Robot Manipulators, MacMillan Publishing, 1993