Computer
& Robotics

An LPV Approach to Sensor Fault Diagnosis of Robotic Arm

Amir Hossein Sabbaghan, Amir Hossein Hassanabadi *

Faculty of Electrical, Biomedical, and Mechatronics Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

Received 13 March 2019; Revised 08 June 2019; Accepted 20 June 2019; Available online 27 June 2019

Abstract

One of the major challenges in robotic arms is to diagnosis sensor fault. To address this challenge, this paper presents an LPV approach. Initially, the dynamics of a two-link manipulator is modelled with a polytopic linear parameter varying structure and then by using a descriptor system approach and a robust design of a suitable unknown input observer by means of pole placement method along with linear matrix inequalities, in addition to providing an estimate of state variables for using in state feedback, the detection, isolation, and identification of sensor faults in the manipulator are addressed. The proposed observer provides a robust estimate of the faults along with attenuating the disturbance effects. Further, the desired angles of the joints are calculated for achieving the desired trajectory of the robot's end-effector using the inverse kinematics and by designing a suitable state feedback law with integral mode, the reference signals are tracked. The sufficient condition for stability of the closed-loop system is obtained as a set of linear matrix inequalities at the vertices of the system. The efficiency and effectiveness of the control system, along with the designed fault diagnosis unit, are shown using numerical simulations.

Keywords: *Fault Diagnosis, Linear Parameter Varying, Polytopic Model, Descriptor System, Unknown Input Observer, Robotic Manipulator, Inverse Kinematics.*

1. Introduction

Fault diagnosis is one of the major and challenging issues in the field of engineering. Common faults in robotic systems are mainly related to either the actuators used in the robot, such as electrical motors that transfer torque to the manipulator, or the robot's sensors that measure positions and the angular velocities. If the sensors' faults remain undetected, the controller will receive incorrect information about the manipulator and this issue itself will cause significant problems in the system. Since diagnosing the faults in the manipulator is complicated and there is a need for isolation of the fault effects from the disturbance effects, the robust fault diagnosis is a very important task for the robot's proper function.

For the manipulator, different methods are considered to diagnose and identify faults. In [1, 2], utilizing the dynamic model of the robot, fault diagnosis is done. Then, using a linear matrix inequality approach, without considering disturbance in the system, the faults are compensated by using a fault-tolerant control method in the manipulator. In [3], the nonlinear model of manipulator and optimal estimation methods are used for fault diagnosis. The disadvantage of this method is the complexity of the equations. This process can be facilitated using linear parameter varying models. In [4, 5], the neural network method is used for diagnosis of the actuator faults in a flexible manipulator. In this method, by determining the

* Corresponding author. Email: a.hassanabadi@qiau.ac.ir

range of input and output membership functions, the desired functionality of the system is achieved. In [6, 7], by using a linear quadratic regulator and the descriptor linear parameter varying modelling approach, an optimal controller is designed for a flexible two-link manipulator. In [8, 9], a flexible manipulator with six degrees of freedom is considered and by using the dynamic model of the robot in LPV format, a state feedback with integral mode is designed to track the reference signals. One of the issues with this method is that isolating several faults in the system by using only one residual generator which is sensitive to only a specific group of faults is impossible; however, this can be done with a direct fault estimation approach. In [10,11], for a two-link manipulator, actuator fault is diagnosed and based on the fault estimation value, the compensation is done. One of the ways to compensate faulty systems is to use a fault-tolerant control system in order to preserve the system stability and performance [12-15]. This article is arranged as follows: In section two, the dynamic and kinematics model of the manipulator is explained. In section three, the unknown input observer will be formulated. In section four, designing state feedback law with integral mode to track the reference signal is addressed. In section five, simulating the proposed method and the numerical results are presented and the final section concludes the paper.

2. Dynamic Model of a Two-link Manipulator

The two-link manipulator moves in a vertical plane and its equilibrium point is the upper vertical position of the robot. The position of the robot is represented with the vector $\theta = [\theta_1 \quad \theta_2]^T$, which corresponds to the angle of the two joints. The input of the system is the vector $u = [u_1 \quad u_2]^T$, which includes the torque applied to the two robot joints. The two-link manipulator discussed in this paper is depicted in Fig. 1. The dynamics of the manipulator is formulated in the following equation:

$$\Xi(\theta)\ddot{\theta} + O(\theta,\dot{\theta})\dot{\theta} + g(\theta) = \mathrm{u} \tag{1}$$

In equation (1), vectors $\dot{\theta}$ and $\ddot{\theta}$ represent the velocities and accelerations of the joints respectively, and $\Xi(\theta) \in \mathbb{R}^{ku \times ku}$ is the moment of inertia matrix of the robot, $O(\theta,\dot{\theta}) \in \mathbb{R}^{ku \times ku}$ is the vector function including centripetal and coriolis torques and $g(\theta) \in \mathbb{R}^{ku}$ is the gravitational torque of the robot.

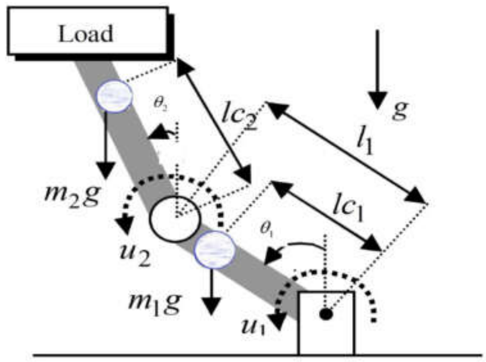

Fig. 1. Two-link manipulator structure.

Nonlinear model of the two-link manipulator is as follows:

$$\begin{aligned} &\left(m_1 L_{C_1}{}^2 + m_2 L_1{}^2 + I_1\right)\ddot{\theta}_1 + \left(m_2 L_1 L_{C_2} \cos(\theta_1 - \theta_2)\right)\ddot{\theta}_2 \\ &+ m_2 L_1 L_{C_2} \sin(\theta_1 - \theta_2)\dot{\theta}_2^2 - \left(m_1 L_{C_1} + m_2 L_1\right) g \sin(\theta_1) = u_1 \end{aligned} \tag{2}$$

$$\begin{aligned} &\left(m_2 L_1 L_{C_2} \cos(\theta_1 - \theta_2)\right)\ddot{\theta}_1 + \left(m_2 L_{C_2}^2 + I_2\right)\ddot{\theta}_2 \\ &- \left(m_2 L_1 L_{C_1} \sin(\theta_1 - \theta_2)\right)\dot{\theta}_1^2 - m_2 g L_{C_2} \sin(\theta_2) = u_2 \end{aligned} \tag{3}$$

In the equations (2) and (3), L_1 is the length of link one. L_{c1} is the distance between the joint one to the mass center of the link one. L_{c2} is the distance between the second joint to the mass center of the second link. m_1 is the mass of arm one and m_2 is the mass of arm two. I_1 is the moment of inertia of the arm one and the load, and I_2 is the moment of inertia of the arm two. In equation (1), the quadratic term $O(\theta,\dot{\theta})$ is omitted due to its boundlessness [11], and as the result, (1) is converted to:

$$\Xi(\theta)\ddot{\theta} + g(\theta) = \mathrm{u} \tag{4}$$

In which $\Xi(\theta)$ is as follows:

$$\Xi(\theta) = \begin{bmatrix} m_1 L_{C_1}^2 + m_2 L_{C_1}^2 + I_1 & m_2 L_1 L_{C_2} \phi_1 \\ m_2 L_1 L_{C_2} \phi_1 & m_2 L^2_{C_2} + I_2 \end{bmatrix} \tag{5}$$

Where:

$$\phi_1 = \cos(\theta_1 - \theta_2); -1 \le \phi_1 \le 1 \qquad (6)$$

$g(\theta)$ is modeled as follows:

$$g(\theta) = \begin{bmatrix} -(m_1 L_{C_1} + m_2 L_1) g \phi_2 \\ -m_2 g L_{C_2} \phi_3 \end{bmatrix} = G^g(\theta) \times \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \qquad (7)$$

in which:

$$G^g(\theta) = \begin{bmatrix} -(m_1 L_{C_1} + m_2 L_1) g \phi_2 & 0 \\ 0 & -m_2 g L_{C_2} \phi_3 \end{bmatrix} \qquad (8)$$

and:

$$\begin{aligned} \phi_2 &= \frac{\sin\theta_1}{\theta_1}; -0.2 \le \phi_2 \le 1 \\ \phi_3 &= \frac{\sin\theta_2}{\theta_2}; -0.2 \le \phi_3 \le 1 \end{aligned} \qquad (9)$$

2.1. Polytopic Linear Parameter Varying Modeling

There are various methods for linearizing a nonlinear system. Linear parameter varying approach is one of the multiple model approaches. The method used in this paper is the polytopic linear parameter varying modeling that is based on the concepts of polytopes and convex hull. Based on this, if $\phi(t) = (\phi_1(t), \phi_2(t), ..., \phi_l(t))$ is the vector of parameters, it is desired to find a set that vector ϕ for all variations of $\underline{\phi_i} \le \phi_i \le \overline{\phi_i}$. The set is a polytope that is based on the $h = 2^l$ vertices ($\{\omega_1, \omega_2, ..., \omega_h\}$). Each of these vertices has the same dimension with ϕ which its elements will only take the maximum and minimum values for $\forall i \in [1, 2, ..., h]$ corresponding ϕ_i values. Therefore:

$$\phi \in co\{\omega_1, \omega_2, ..., \omega_n\} = \{\sum_{i=1}^{h} \rho_i \omega_i : \rho_i \succ 0, \sum_{i=1}^{h} \rho_i = 1\},\ h = 2^l \qquad (10)$$

In (10), l is the number of parameters and h is the number of subsystems. For a system with three parameters, $h = 2^3 = 8$, so there will be eight subsystems which are depicted in Fig. 2.

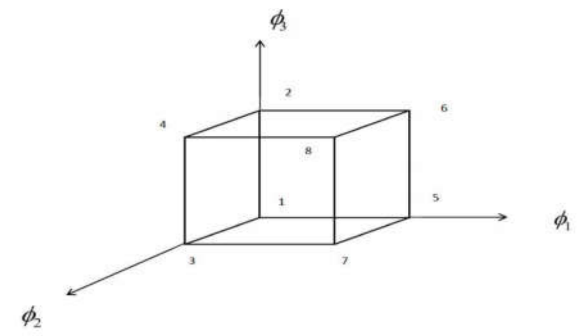

Fig. 2. The polytopic system with three parameters.

Weight of each of the subsystems is calculated as follows:

$$\begin{aligned} \rho_1(\phi) &= \alpha_1 \times \alpha_2 \times \alpha_3 \\ \rho_2(\phi) &= \alpha_1 \times \alpha_2 \times (1-\alpha_3) \\ \rho_3(\phi) &= \alpha_1 \times (1-\alpha_2) \times \alpha_3 \\ \rho_4(\phi) &= \alpha_1 \times (1-\alpha_2) \times (1-\alpha_3) \\ \rho_5(\phi) &= (1-\alpha_1) \times \alpha_2 \times \alpha_3 \\ \rho_6(\phi) &= (1-\alpha_1) \times \alpha_2 \times (1-\alpha_3) \\ \rho_7(\phi) &= (1-\alpha_1) \times (1-\alpha_2) \times \alpha_3 \\ \rho_8(\phi) &= (1-\alpha_1) \times (1-\alpha_2) \times (1-\alpha_3) \end{aligned} \qquad (11)$$

In (11), α_1 , α_2 , α_3 are obtained from the following equations respectively:

$$\alpha_1 = \frac{\phi_{1\max} - \phi_1}{\phi_{1\max} - \phi_{1\min}}; \alpha_2 = \frac{\phi_{2\max} - \phi_2}{\phi_{2\max} - \phi_{2\min}}; \alpha_3 = \frac{\phi_{3\max} - \phi_3}{\phi_{3\max} - \phi_{3\min}} \qquad (12)$$

There are three parameters in the system, so the linear parameter varying system is a weighted combination of $2^3 = 8$ subsystems and thus $\Xi(\theta)$ will be as follows:

$$\Xi_i(\theta) \in Co\left\{\Xi_1, \Xi_2, \Xi_3, \Xi_4, \Xi_5, \Xi_6, \Xi_7, \Xi_8\right\} \qquad (13)$$

Considering (5), matrices $\Xi_i(\theta), i \in [1, 2, ...8]$ are dependent on the parameter ϕ_1 . In the matrices $\Xi_1(\theta)$, $\Xi_2(\theta)$, $\Xi_3(\theta)$ and $\Xi_4(\theta)$, the value of ϕ_1 is $\phi_{1\min} = -1$ so:

$$\Xi_1(\theta) = \Xi_2(\theta) = \Xi_3(\theta) = \Xi_4(\theta) = \begin{bmatrix} m_1 L_{C_1}^2 + m_2 L_{C_1}^2 + I_1 & -m_2 L_1 L_{C_2} \\ -m_2 L_1 L_{C_2} & m_2 L_{C_2}^2 + I_2 \end{bmatrix} \qquad (14)$$

In the matrices, $\Xi_5(\theta), \Xi_6(\theta), \Xi_7(\theta)$ and $\Xi_8(\theta)$ the value of ϕ_1 is $\phi_{1\max} = 1$, so:

$$\Xi_5(\theta)=\Xi_6(\theta)=\Xi_7(\theta)=\Xi_8(\theta)=\begin{bmatrix} m_1L_{C_1}^2+m_2L_{C_1}^2+I_1 & m_2L_1L_{C_2} \\ m_2L_1L_{C_2} & m_2L_{C_2}^2+I_2 \end{bmatrix} \tag{15}$$

According to (8):

$$G_i^g(\theta)\in Co\left\{G_1^g, G_2^g, G_3^g, G_4^g, G_5^g, G_6^g, G_7^g, G_8^g\right\} \tag{16}$$

$G_i^g(\theta)$ are dependent on $\phi_3(\theta), \phi_2(\theta)$. For $G_1^g(\theta), G_5^g(\theta)$, $\phi_{2\min}=-0.2, \phi_{3\min}=-0.2$, therefore:

$$G_1^g(\theta)=G_5^g(\theta)=\begin{bmatrix} 0.2[m_1L_{C_1}+m_2L_1]g & 0 \\ 0 & 0.2m_2gL_{C_2} \end{bmatrix} \tag{17}$$

For $G_2^g(\theta), G_6^g(\theta)$, $\phi_{2\max}=1, \phi_{3\min}=-0.2$, so:

$$G_2^g(\theta)=G_6^g(\theta)=\begin{bmatrix} -[m_1L_{C_1}+m_2L_1]g & 0 \\ 0 & 0.2m_2gL_{C_2} \end{bmatrix} \tag{18}$$

For $G_3^g(\theta), G_7^g(\theta)$, $\phi_{2\min}=-0.2, \phi_{3\max}=1$ hence:

$$G_3^g(\theta)=G_7^g(\theta)=\begin{bmatrix} 0.2[m_1L_{C_1}+m_2L_1]g & 0 \\ 0 & -m_2gL_{C_2} \end{bmatrix} \tag{19}$$

For $G_4^g(\theta), G_8^g(\theta)$, $\phi_{2\max}=1, \phi_{3\max}=1$ therefore:

$$G_4^g(\theta)=G_8^g(\theta)=\begin{bmatrix} -[m_1L_{C_1}+m_2L_1]g & 0 \\ 0 & -m_2gL_{C_2} \end{bmatrix} \tag{20}$$

By defining the following state variable vector for the system:

$$x(t)=\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix}=\begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot\theta_1 \\ \dot\theta_2 \end{bmatrix} \tag{21}$$

The system's state space equation is as follows:

$$\pi_i\dot x(t)=\begin{bmatrix} 0 & I_{ku} \\ -G_i^g(\theta) & 0 \end{bmatrix}x(t)+w_bu(t) \tag{22}$$

in which:

$$w_b=\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{23}$$

And the nonsingular matrix π_i is:

$$\pi_i=\begin{bmatrix} I & 0 \\ 0 & \Xi_i(\theta) \end{bmatrix} \tag{24}$$

The following linear parameter varying system with disturbance and sensor fault is considered:

$$\begin{cases} \dot x(t)=\sum_{i=1}^{h}\rho_i\left(\phi(t)\right)\left(A_ix(t)+B_iu(t)+B_dd(t)\right) \\ y(t)=Cx(t)+D_ff(t) \end{cases} \tag{25}$$

In (25), $x(t)\in\mathbb{R}^n$ is the state vector, $u(t)\in\mathbb{R}^{ku}$ is the input vector, $y(t)\in\mathbb{R}^m$ is the output vector, $d(t)\in\mathbb{R}^{kd}$ is the unknown input vector and $f(t)\in\mathbb{R}^{kf}$ is the sensor fault vector. B_d is the disturbance distribution matrix, D_f is the fault distribution matrix, and $\phi(t)\in\mathbb{R}^l$ is the parameter vector of the system which is measurable online. $\rho_i\left(\phi(t)\right)$ for $i=1,2,.....,h$ are the weights of different subsystems in the vertices which satisfy:

$$\sum_{i=1}^{h}\rho_i(\phi)=1, \rho_i(\phi)\ge 0 \qquad \forall i\in[1,2,.......h] \tag{26}$$

The matrices C, B_d, D_f are fixed matrices with appropriate dimensions, and A_i, B_i for $i=1,2,.......h$ are the matrices of vertices that are obtained as follows:

$$A_i=\pi_i^{-1}\begin{bmatrix} 0 & I \\ -G_i^g & 0 \end{bmatrix} \tag{27}$$

$$B_i=\pi_i^{-1}w_b \tag{28}$$

According to measuring the positions and the velocities of the two joints of the robot:

$$C = I_m \tag{29}$$

Considering the fault in the position sensors:

$$D_f = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}^T \tag{30}$$

and the disturbance distribution matrix is considered as follows:

$$B_d = \begin{bmatrix} 0 & 0 & 0.7 & 0 \end{bmatrix}^T \tag{31}$$

2.2. Kinematics of the Two-Link Manipulator

Forward kinematics of the two-link manipulator which relates the Cartesian position of the end effector according to the joint angles is as follows:

$$\begin{cases} x = l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) \\ y = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) \end{cases} \tag{32}$$

In equation (32), l_1 is the length of the 1st link, l_2 is the length of the 2nd link, θ_1 is the angle of the 1st joint and θ_2 is the angle of the 2nd joint of the manipulator.

To obtain the inverse kinematics from the direct kinematics, by using trigonometric relations, the joint angles θ_2 and θ_1 are calculated as follows respectively:

$$\theta_2 = \cos^{-1}\left(\frac{x^2 + y^2 - l_1^2 - l_2^2}{2 l_1 l_2}\right) \tag{33}$$

$$\theta_1 = \tan^{-1}\left(\frac{y}{x}\right) - \tan^{-1}\left(\frac{l_2 \sin(\theta_2)}{l_1 + l_2 \cos(\theta_2)}\right) \tag{34}$$

3. Unknown Input Observer Design

In methods based on disturbance decoupling, the disturbance will be considered as unknown input. The aim in designing the unknown input observer is to decouple the unknown inputs and to estimate the fault. In order to estimate the faults as well as the state variables, the following augmented state vector is defined [16]:

$$\tilde{x}(t) = \begin{bmatrix} x^T(\mathrm{t}) & f^T(\mathrm{t}) \end{bmatrix}^T \tag{35}$$

By defining this state vector, (25) is rewritten as follows:

$$\begin{cases} E\dot{\tilde{x}} = \sum_{i=1}^{h} \rho_i\left(\phi(t)\right)\left(\tilde{A}_i \tilde{x}(t) + \tilde{B}_i u(t) + \tilde{B}_d d(t) + \tilde{B}_f f(t)\right) \\ y(t) = \tilde{C}\tilde{x}(t) \end{cases} \tag{36}$$

in which $\mathrm{E} \in \mathbb{R}^{s \times s}$ $(s = n + kf)$ is a singular matrix and the matrices of the augmented system (36) is as follows:

$$\begin{gathered} \tilde{A}_i = \begin{bmatrix} A_i & 0 \\ 0 & -I \end{bmatrix}, \tilde{B}_i = \begin{bmatrix} B_i \\ 0 \end{bmatrix}, \mathrm{E} = \begin{bmatrix} I_n & 0 \\ 0 & 0 \end{bmatrix} \\ \tilde{B}_d = \begin{bmatrix} B_d \\ 0 \end{bmatrix}, \tilde{C} = \begin{bmatrix} C & D_f \end{bmatrix}, \tilde{B}_f = \begin{bmatrix} 0 \\ I_n \end{bmatrix} \end{gathered} \tag{37}$$

The system (36) is a singular (descriptor) system which in addition to dynamic equations, it contains algebraic equations, too. In the following, the observer will be designed to estimate the state $\tilde{x}(t)$ (simultaneous estimation of the fault and the original state). The following unknown input observer is proposed:

$$\begin{cases} \dot{z}(\mathrm{t}) = \sum_{i=1}^{h} \rho_i\left(\phi(t)\right)\left(N_i z(t) + G_i u(t) + L_i y(t)\right) \\ \hat{\tilde{x}}(t) = z(t) + T_2 y(t) \end{cases} \tag{38}$$

In (38), $z \in \mathbb{R}^s$ is the observer state variable, $\hat{\tilde{x}} \in \mathbb{R}^{s \times s}$ is the estimate of the augmented state vector, and matrices N_i, L_i, T_2 and G_i have proper dimensions. The state estimation error is defined as follows:

$$e(t) = \tilde{x}(t) - \hat{\tilde{x}}(t) \tag{39}$$

By substituting (38) into (39):

$$e(t) = (I - T_2\tilde{C})\tilde{x}(t) - z(\mathrm{t}) \tag{40}$$

The following change of variable is applied:

$$I - T_2\tilde{C} = T_1 E \tag{41}$$

Substituting (41) into (40) results in:

$$\dot{e}(t) = T_1 E \dot{\tilde{x}}(t) - \dot{z}(\mathrm{t}) \tag{42}$$

Substituting (36) and (38) into (42) results in:

$$\dot{e}(t) = \sum_{i=1}^{h} \rho_i(\phi(t))\{N_i e + (T_1\tilde{A}_i - N_i T_1 E - L_i\tilde{C})\tilde{x}(t) + (T_1\tilde{B}_i - G_i)u(t) + T_1\tilde{B}_f f(t) + T_1\tilde{B}_d d(t)\} \quad (43)$$

By applying the following constraints:

$$T_1\tilde{A}_i - N_i T_1 E - L_i\tilde{C} = 0 \quad (44)$$

$$T_1\tilde{B}_i - G_i = 0 \quad (45)$$

the error dynamics (43) reduces to:

$$\dot{e}(t) = \sum_{i=1}^{h} \rho_i(\phi(t))\{N_i e(t) + T_1\tilde{B}_f f(t) + T_1\tilde{B}_d d(t)\} \quad (46)$$

Equation (41) is written in the matrix form as:

$$\begin{bmatrix} T_1 & T_2 \end{bmatrix}\sigma = \Omega \quad (47)$$

in which:

$$\sigma = \begin{bmatrix} E \\ \tilde{C} \end{bmatrix}, \Omega = I_S \quad (48)$$

The solution of (47) is:

$$\Delta = \begin{bmatrix} T_1 & T_2 \end{bmatrix} = \Omega\sigma^{+} \quad (49)$$

To facilitate the calculation of the unknown matrices N_i, L_i, the following new variables are defined [17, 18]:

$$K_i = N_i T_2 - L_i \quad \forall i \in [1, 2, ..., h] \quad (50)$$

By using (44) and (50), N_i is calculated as follows:

$$N_i = T_1\tilde{A}_i + K_i\tilde{C} \quad \forall i \in [1, 2, ..., h] \quad (51)$$

and:

$$L_i = N_i T_2 - K_i \quad \forall i \in [1, 2, ..., h] \quad (52)$$

G_i is obtained from (45) as follows:

$$G_i = T_1\tilde{B}_i \quad \forall i \in [1, 2, ..., h] \quad (53)$$

If K_i are calculated, the matrices N_i, L_i, G_i can be determined from (51), (52) and (53), respectively. The condition for robust stability of estimation error dynamics (46) in the presence of unknown inputs $f(t)$ and $d(t)$ is formulated as the following linear matrix inequalities for $\forall i \in [1, 2, ..., h]$:

$$\begin{bmatrix} PT_1\tilde{A}_i + y_i\tilde{C} + (PT_1\tilde{A}_i + y_i\tilde{C})^T & PT_1\tilde{B}_d & PT_1\tilde{B}_f & I \\ * & -\gamma^2 I & 0 & 0 \\ * & 0 & -\gamma^2 I & 0 \\ * & 0 & 0 & -I \end{bmatrix} \prec 0 \quad (54)$$

In (54), the change of variable $y_i = PK_i$ is used to linearize the nonlinear matrix inequalities. The condition for placing the observer error dynamic poles in the circular area $D = (q, r)$ with the center $(-q, 0)$ and radius r, in addition to guaranteeing the robust stability, considering the method in [21, 22] is as the set of following LMIs for $\forall i \in [1, 2, ..., h]$:

$$\left[\begin{array}{cc|cc|c} -rP & \Sigma_1 & \Sigma_2 & \Sigma_3 & 0 \\ \Sigma_1^T & -rP & 0 & 0 & I \\ \hline \Sigma_2^T & 0 & -\gamma^2 I & 0 & 0 \\ \Sigma_3^T & 0 & 0 & -\gamma^2 I & 0 \\ \hline 0 & I & 0 & 0 & -I \end{array}\right] < 0 \quad (55)$$

in which

$$\Sigma_1 = qP + PT_1\tilde{A}_i + y_i\tilde{C}$$
$$\Sigma_2 = PT_1\tilde{B}_d$$
$$\Sigma_3 = PT_1\tilde{B}_f$$

If the linear matrix inequalities (55) are feasible, matrices $P > 0$ and y_i will be obtained and then:

$$K_i = P^{-1} y_i, \ \forall i \in [1, 2, ..., h] \quad (56)$$

The matrices N_i, L_i and G_i for $\forall i \in [1, 2, ..., h]$ are obtained from (51), (52) and (53), respectively.

4. Tracking Controller Design

The system (36) is assumed to be controllable. The aim is to design a suitable state feedback to achieve the following tracking goal in addition to guarantee the stability of the closed-loop system [19, 20]:

$$\lim_{t\to\infty} \bar{y}(t) = w(t) \tag{57}$$

In this paper, only tracking the first two outputs are considered. Therefore, the vector $\bar{y}(t)$ is defined as follows:

$$\bar{y}(t) = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \tag{58}$$

Reference input $w(t)$ is obtained by using the inverse kinematics of the robot and the desired trajectory in the cartesian space. In other words, $w(t) = \left[\theta_{1d}(t) \quad \theta_{2d}(t)\right]^T$ is the desired vector of the joint angles. For the given reference input $w(t)$, the integral mode will be defined as follows which is depicted in Fig. 3:

$$\dot{\varepsilon}(t) = w(t) - \bar{y}(t) = w(t) - \bar{C}x(t) - \bar{D}_f f(t) \tag{59}$$

in which:

$$\bar{C} = I_{ku} \tag{60}$$

$$\bar{D}_f = I_{ku} \tag{61}$$

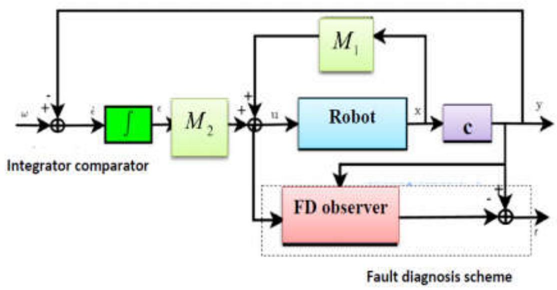

Fig. 3. Structure of the fault diagnosis loop and tracking controller.

The following augmented state variable is considered:

$$x_c(t) = \left[x^T(t) \quad \varepsilon^T(t)\right]^T \tag{62}$$

Considering (25), (59) and (62), the following augmented system is resulted:

$$\dot{x}_c(t) = \sum_{i=1}^{h} \rho_i(\phi(t))\left(\bar{A}_{c_i} x_c(t) + \bar{B}_{c_i} u(t) + \bar{B}_{d_c} \tilde{d}(t)\right) \tag{63}$$

in which:

$$\bar{A}_{c_i} = \begin{bmatrix} A_i & 0 \\ -\bar{C} & 0 \end{bmatrix}, \bar{B}_{c_i} = \begin{bmatrix} B_i \\ 0 \end{bmatrix}, \bar{B}_{d_c} = \begin{bmatrix} B_d & 0 & 0 \\ 0 & -\bar{D}_f & I_{\bar{m}} \end{bmatrix}$$
$$\tilde{d}(t) = \left[d^T(t) \quad f^T(t) \quad w^T(t)\right]^T \tag{64}$$

A necessary and sufficient condition for the existence of a stabilizing state feedback controller for the system (63), is the controllability of the system. Now, the following state feedback control with the integral mode is considered:

$$u(t) = M_1 x(t) + M_2 \varepsilon(t) = \begin{bmatrix} M_1 & M_2 \end{bmatrix} \begin{bmatrix} x(t) \\ \varepsilon(t) \end{bmatrix} = M x_c(t) \tag{65}$$

Applying the control law (65) to (63) results in:

$$\dot{x}_c(t) = \sum_{i=1}^{h} \rho_i(\phi(t))\left(\bar{A}_{c_i} + \bar{B}_{c_i} M\right) x_c(t) + \bar{B}_{d_c} \tilde{d}(t) \tag{66}$$

Assuming a constant unknown input vector, the condition for exponential stability of the state variable x_c is obtained by the following matrix inequalities:

$$sym\left\{\bar{A}_{c_i} Q + \bar{B}_{c_i} X_i\right\} + 2\alpha Q \prec 0 \qquad \forall i \in [1,2,......,h] \tag{67}$$

In (67), to linearize matrix inequalities, multiplication of the two sides of the inequality in P^{-1} and introducing the new variables $Q = P^{-1}, MQ = X$ are applied.

If LMIs (67) are feasible, matrices $Q > 0$ and X are obtained and then:

$$P = Q^{-1} \tag{68}$$

$$M = XP \tag{69}$$

Subsequently the matrices M_1 , M_2 will be constructed from n first columns and k_u last columns of M , respectively.

5. Simulation

In this section, the considered manipulator with the designed controller and the UIO are simulated.

In Table 1, the numerical values of the robot parameters are given.

Table. 1. Parameters of the two-link manipulator

Parameters	I_1	I_2	L_1	L_2	L_{C_1}	L_{C_2}	m_1	m_2	g
Values	0.83	0.41	1	1	0.5	0.5	10	5	9.8
Units	$kg*m^2$	$kg*m^2$	m	m	m	m	kg	kg	m/s^2

The UIO (38) is designed by solving the set of LMIs (55) with the attenuation level $\gamma = 1$ and selecting $q = 20$ and $r = 15$. After solving these LMIs, the matrices N_i, L_i and G_i for $\forall i \in [1,2,......,h]$ are calculated from (51), (52) and (53) respectively. The tracking controller (65) is designed by solving the set of LMIs (67) by choosing $\alpha = 2$. M_1 and M_2 are calculated from (69) as follows:

$$M_1 = \begin{pmatrix} -4426.747411 & -0.001668 & -420.832109 & -0.000786 \\ 0.003075 & -1440.412948 & -0.000361 & -145.310838 \end{pmatrix}$$

$$M_2 = \begin{pmatrix} 13268.9910017 & -0.0171864 \\ -0.0312571 & 3932.6833916 \end{pmatrix}$$

The considered manipulator should track the path being shown in Fig 4. This path is determined by the obstacle avoidance constraint in the robot environment. The reference signals for the two joint angles; w_1, w_2 are shown in Fig 6 which are calculated from the path shown in Fig 4 by inverse kinematics.

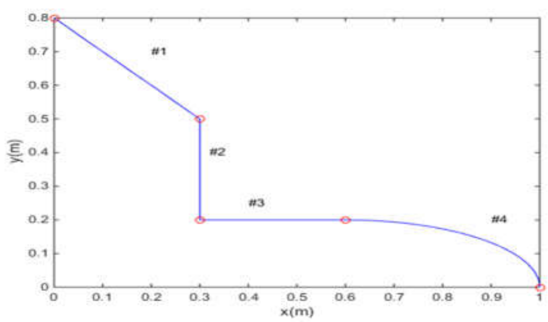

Fig. 4. Reference path in cartesian space.

In the simulation, a white noise with mean and variance respectively equivalent to 0.2 and 0.9 is considered as the disturbance. Abrupt and incipient faults on the two sensors are considered which overlap on some periods as being depicted in Fig 5.

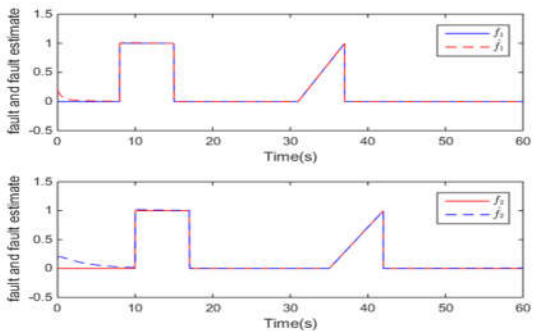

Fig. 5. Sensor faults and their estimation.

The joint angles (y_1, y_2) which track the reference inputs are shown in Fig 6. It is seen that the tracking is done with an acceptable performance in spite of disturbance and faults in the system.

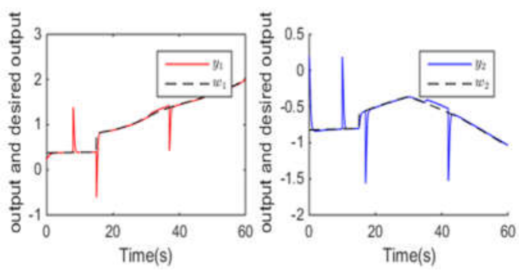

Fig. 6. Measured values of joint positions (including sensor faults) and their reference values.

The system states (joint angles and velocities) and their estimates are shown in Fig 6. The estimation of these signals and its robustness can be seen from this figure. These robust estimates are used for state feedback controller and also for direct fault diagnosis (including three stages of fault detection, isolation and reconstruction).

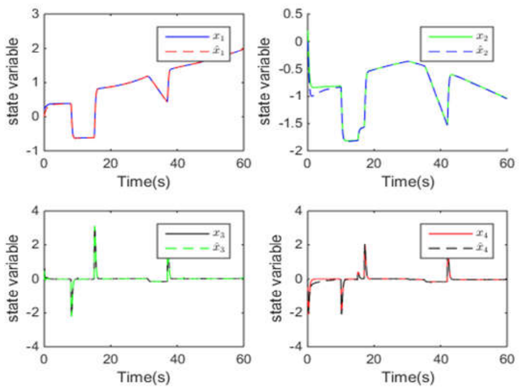

Fig. 7. The positions and velocities of the robot joints with their estimation.

The faults estimates are shown in Fig 5. The direct fault reconstruction scheme presented in this paper has the advantages of excluding the need for designing bank of UIOs for fault isolation and also achieving fault diagnosis of simultaneously appearing faults which is a challenge for many fault diagnosis schemes.

The control signals applied by the two joint actuators are depicted in Fig 8 which both are bounded. As it is observed from this figure, the abrupt faults in the sensors have an abrupt effect on the two control -signals but incipient faults have a smooth effect.

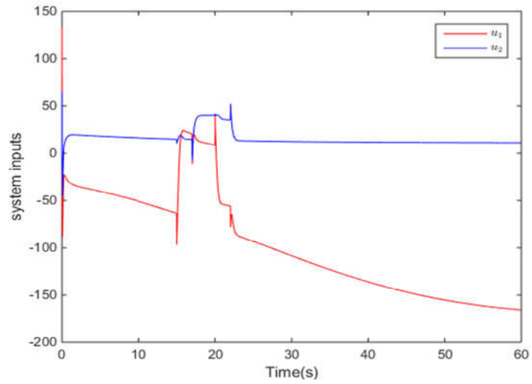

Fig. 8. Control signals applied by the two joint actuators.

For evaluating the performance of the UIO in the fault and state estimation and also the controller tracking performance, Mean Integral of Square Error (MISE) index defined as follows is used:

$$MISE = 1/t\int_0^t e^2(t)dt \tag{70}$$

In which, T is the total time of the simulation. The obtained results of this index for estimation error/tracking error of different signals are given in Table 2. It can clearly be seen that these values are very small for the robot being simulated which shows the effectiveness the UIO and controller being designed.

Table. 2. Mise calculated for different signals

Tracking error		Estimation error								
e_{tr_2}	e_{tr_1}	y_2	y_1	f_2	f_1	x_4	x_3	x_2	x_1	Signal
0.007	0.003	0.053	0.034	0.002	0.006	0.047	0.024	0.002	0.006	MISE

6. Conclusion

In this paper, the LPV model of a two-link manipulator is presented. Based on this model, an unknown input observer which can estimate the joint angles and the velocities of the robot and also the sensor faults is designed. Then a state feedback controller enhanced with integral action is designed which by using the robust estimates of the system states can guarantee the exponential convergent of the joint angles to the reference signals calculated based on the reference path and inverse kinematics. The advantage of the proposed method is achieving direct fault diagnosis by estimating the fault vector as additional states in the augmented system. Designing an active fault tolerant controller based on this fault diagnosis scheme is the future line of this research.

References

[1] Mi, Y.; Xu, F.; Tan, J.; Wang, X.; Liang, B., "Fault-tolerant control of a 2-DOF robot manipulator using multi-sensor switching strategy." vol. 63, no. 3, pp. 7307-7314 (2017).

[2] Mittal, S.; Dave, M.; Kumar, A., "Fault-Tolerant Position Control of the Manipulator of PUMA Robot using Hybrid Control Approach." International Journal of Current Engineering and Technology, vol. 6, no. 5 (2016).

[3] Kazemi, H.; Yazdizadeh, A., "Optimal state estimation and fault diagnosis for a class of nonlinear systems." IEEE/CAA Journal of Automatica Sinica (2017).

[4] Cho, C. N.; Hong, J. T.; Kim, H. J., "Neural Network Based Adaptive Actuator Fault Detection Algorithm for Robot Manipulators." Journal of Intelligent & Robotic Systems, vol. 95, no. 1, pp. 137-147 (2019).

[5] Khalastchi, E.; Kalech, M., "A sensor-based approach for fault detection and diagnosis for robotic systems." Autonomous Robots, vol. 42, no. 6, pp. 1231-1248 (2018).

[6] Altun, Y., "Gain scheduling LQI controller design for LPV descriptor systems and motion control of two-link flexible joint robot manipulator" An International Journal of Optimization and Control: Theories & Applications (IJOCTA), vol. 8, no. 2, pp. 201-207 (2018).

[7] Sari, C.; Agustinah, T.; Jazidie, A., "Design of actuator fault compensation with MRC in 2 DOF manipulator based on PID CTC." pp. 250-254 (2012).

[8] Vizer, D.; Mercere, G.; Laroche, E., "Gray-box LPV model identification of a 2-DoF surgical robotic manipulator by using an H∞-norm-based local approach." IFAC-Papers on Line, vol. 48, no. 26, pp. 79-84 (2015).

[9] Vizer, D.; Mercere, G.; Laroche, E., "Combining analytic and experimental information for linear parameter-varying model identification: application to a flexible robotic manipulator." Periodica Polytechnica Electrical Engineering and Computer Science, vol. 58, no. 4, pp. 133-148 (2014).

[10] Zribi, S.; Tlijani, H.; Knani, J.; Puig, V., "Impact of external disturbance and discontinuous input on the redundant manipulator robot behaviour using the linear parameter varying modelling approach." International Journal of Advanced Computer Science and Applications, vol. 8, no. 10, pp. 202-208 (2017).

[11] Patton, R. J.; Klinkhieo, S., "LPV fault estimation and FTC of a two-link manipulator." pp. 4647-4652 (2006).

[12] Gang, C.; Wen, G.; Qingxuan, J.; Xuan, W.; Yingzhuo, F., "Failure treatment strategy and fault-tolerant path planning of a space manipulator with free-swinging joint failure." Chinese Journal of Aeronautics, vol. 31, no. 12, pp. 2290-2305 (2018).

[13] Zhixiang, L.; Youmin, Z.; Chi, Y.; Jun, L., "An adaptive linear parameter varying fault tolerant control scheme for unmanned surface vehicle steering control." pp. 6197-6202 (2017).

[14] Ijaz, S.; Yan, L.; Hamayun, M. T.; Baig, W. M.; Shi, C., "An Adaptive LPV Integral Sliding Mode FTC of Dissimilar Redundant Actuation System for Civil Aircraft." IEEE Access, vol. 6, pp. 65960-65973 (2018).

[15] Hassanabadi, A. H.; Shafiee, M.; Puig, V., "Sensor fault diagnosis of singular delayed LPV systems with inexact parameters: an uncertain system approach." International Journal of Systems Science, vol. 49, no. 1, pp. 179-195 (2018).

[16] Hassanabadi, A. H.; Shafiee, M.; Puig, V., "Actuator fault diagnosis of singular delayed LPV systems with inexact measured parameters via PI unknown input observer." IET Control Theory & Applications. Vol. 11, no. 12, pp. 1894-1903 (2017).

[17] Estrada, F. R. L., "Model-based fault diagnosis observer design for descriptor LPV system with unmeasurable gain scheduling." Université de Lorraine (2014).

[18] López-Estrada, F. R.; Ponsart, J. C.; Astorga-Zaragoza, C. M.; Theilliol, D., "Fault estimation observer design for descriptor-LPV systems with unmeasurable gain scheduling functions." pp. 269-274 (2007).

[19] López-Estrada, F. R.; Ponsart, J.-C.; Theilliol, D.; Zhang, Y.; Astorga-Zaragoza, C. M., "LPV model-based tracking control and robust sensor fault diagnosis for a quadrotor UAV." Journal of Intelligent & Robotic Systems, vol. 84, no. 1-4, pp. 163-177 (2016).

[20] Chilali, M.; Gahinet, P.; Apkarian, P., "Robust pole placement in LMI regions." IEEE transactions on Automatic Control, vol. 44, no. 12, pp. 2257-2270 (1999).

[21] Patton, R.; Chen, L.; Klinkhieo, S., "An LPV pole-placement approach to friction compensation as an FTC problem." International Journal of Applied Mathematics and Computer Science, vol. 22, no. 1, pp. 149-160 (2012).