MedCrave
Step into the World of Research

International Robotics & Automation Journal

Conceptual paper

Open Access

CrossMark

Active fault tolerant control based on nonlinear H_∞ subject to actuator and sensor faults for a parallel robot

Abstract

In this paper, an Active Fault Tolerant Control (AFTC) strategy using a nonlinear H_∞ control is proposed for a delta type parallel robot in the presence of actuator and sensor fault. First, dynamic modeling of the robot is accomplished using the Lagrange method. To measure the position and velocity, a super-twisting third-order sliding mode (STW-TOSM) observer is applied. The proposed scheme can accommodate both faults and uncertainties without velocity measurement. In addition, fast convergence and high accuracy is achieved because of applying the high-order sliding mode (HOSM) observer. In order to indicate the effectiveness of the FTC on the basis of nonlinear H_∞ , its performance is compared with conventional sliding mode and feedback linearization methods. The obtained results reveal the efficacy of the proposed FTC- H_∞ .

Keywords: fault tolerant control, parallel robot, nonlinear H_∞ , finite time convergence

Volume 6 Issue 3 - 2020

Mahmood Mazare, Mostafa Taghizadeh, Pegah Ghaf Ghanbari
School of Mechanical engineering, Shahid Beheshti University, Iran

Correspondence: Mostafa Taghizadeh, School of Mechanical engineering, Shahid Beheshti University, Iran, P.O.B. 1743524155, Tehran, Iran, Email mo_taghizadeh@sbu.ac.ir

Received: May 07, 2020 | **Published:** August 31, 2020

Introduction

Parallel manipulators have proven their outstanding performance compared to the conventional serial ones, in terms of high accuracy, velocity, stiffness, payload capacity and great dynamic performance.[1,2] This makes them a good choice for application in industrial sector, particularly for processing and manufacturing, and increases the productivity and product quality. However, regardless of their satisfactory performance, they are prone to component faults and failures, which not only deteriorate their performance, but also has the potential to damage their structure and their environment. Moreover, effective handling the occurring faults can save considerable cost of repair and maintenance, and prevents the production line shutdown for every fault or minor failure. The growing demand for safety, reliability and productivity has been the driving force for the emergence of Fault-Tolerant Control (FTC) systems and has made them an indispensable part of control system design process.[3] Successful implementation of this approach to various applications has proven its effectiveness in diverse fields like, spacecrafts,[4,5] robotics,[6-9] and energy.[10–12]

Generally, fault-tolerant control approaches can be categorized into two distinct categories: active approach,[13,14] and passive approach.[15,16] Passive FTC (PFTC) does not rely on the fault information, instead the effects of faults are considered as the effects of an additional uncertainties. In this case, a robust control approach is utilized to compensate for the effect of the faults.[17,18] On the other hand, Active FTC (AFTC) requires the faults information in advance and is based on rearranging the control structure isolation.[19,20]

Fault detection (FD), a fundamental part of AFTC, is conducted through different approaches, among which sliding mode observer (SMO)[21–23] is highly attractive, as it offers high robustness against uncertainties. However, conventional SMO suffers from chattering and infinite time convergence. The super-twisting SMO was designed to guarantee the finite time convergence of the state observer and attenuate the chattering,[24–26] but it cannot eliminate it completely.[27] High order Sliding Mode (HOSM) techniques have been proposed to promote accuracy and eliminate chattering, while retaining the robustness of the conventional SMC.[28] A super-twisting third-order SMO gives both exact theoretical velocity estimations and an unknown input estimation.29 For FTC purpose of robotic manipulators, different strategies have been adopted, mainly based on computed torque control (CTC),[30,31] backstepping,[32] Neural Networks (NN),[33] Model predictive,[9] and sliding mode control (SMC).[6,7]

An adaptive fuzzy sliding mode controller with variable estimation was introduced by Piltan et al.[34] to handle faults and uncertainties with high robustness, and verified with 6 DOF programmable universal manipulation arm (PUMA) 560 robot manipulator. The sliding surface slope gain of this sliding mode fault-tolerant control technique was tuned adaptively using a fuzzy procedure. Van et al.[35] proposed a robust fault diagnosis and fault-tolerant control (FTC) system by applying a combination of a fault diagnosis scheme based on a super-twisting third-order sliding mode (STW-TOSM) observer with a robust super-twisting second-order sliding mode (STW-SOSM) controller to a PUMA560 robot. Employing high-order sliding mode (HOSM) observer/controller strategy has improved the convergence rate, enhanced the accuracy, and reduced the chattering. A finite-time integral back-stepping fault tolerant control for a class of nonlinear systems was introduced[36] and applied to a two-link rigid robot manipulator. An adaptive version of the controller is then developed to deal with the total uncertainties in the system. It was proved that the controller is effective in terms of precision of tracking, convergent speed and robustness.

During recent years, nonlinear H_∞ control theory has attracted increasing attention due to its inherent robustness and disturbance rejection capabilities and has been applied to a variety of systems, such as electric motors,[37] spacecrafts,[38–40] and robotic manipulators.[41–43] The objective of this control theory is to achieve a bounded ratio between the energy of error signals and the energy of disturbance signals.[44] To the best of the authors' knowledge, nonlinear H_∞ control scheme has not been employed for fault-tolerant control of robotic manipulators which takes actuator faults, input saturation, external disturbances and

system uncertainties into account, simultaneously. Motivated by the above discussion, the most obvious contributions of this study can briefly be drawn as follows:

Proposing a new finite-time FTC based on nonlinear H_∞ and third order super twisting sliding mode observer for a delta type parallel robot which guarantees faster convergence of the system states and also provides high-precision tracking.

Considering actuator and sensor faults, input saturation, external disturbances and parameter uncertainties simultaneously and applying a nonlinear robust FTC to deal with these challenges which can increase reliability and safety.

The FTC- H_∞ does not require uncertainty information, which enables the proposed FTC more applicable for implementing.

The following sections of this paper are: section 2 entailing the introduction of the 3[P2(US)] parallel robot, followed by its kinematic and dynamic analysis in sections 3 and 4. In section 5 the proposed control strategy is described and section 6 contains fault detection using STW-TOSM. Desired path planning is addressed in section 7. Section 8 is devoted to numerical simulation and the conclusion is drawn is section 9.

Architecture of 3[P2(US)] parallel manipulator

The 3[P2(US)] parallel manipulator under study consists of a moving platform termed the end-effector and a fixed platform named the base, joined together via three identical limbs as depicted in Figure 1. Each limb is comprised of an active prismatic joint, installed at an inclination relative to the base and a parallelogram linkage. As the name implies, each parallelogram consists of two parallel rods with universal and spherical joints at their both extremities.

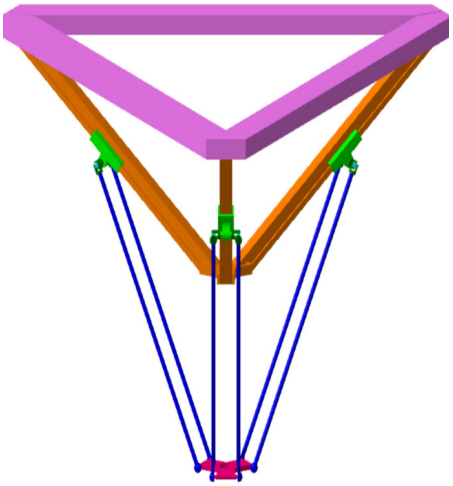


Figure 1 The 3[P2(US)] architecture of delta PKM.

Geometric parameters and variables of one of the limbs are illustrated Figure 2. Three limbs are connected to the base through points A_i, which are located on an imaginary circle with radius R and centered at the origin of reference frame. Similarly, the points B_i are placed on a circle with the radius β and show the connection between the limbs and the end-effector. Active prismatic joints are installed at the angle β to the base and are connected to the parallelogram via universal joints at the points C_i. The length of each parallelogram is specified with l.

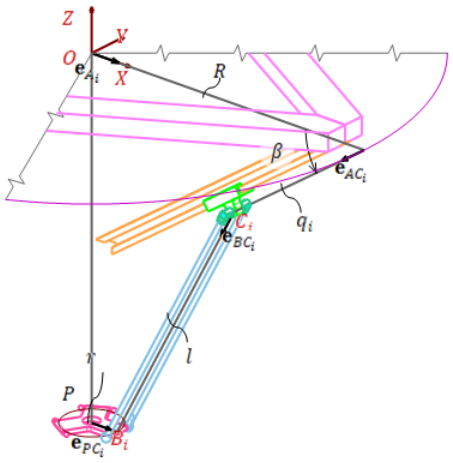


Figure 2 Parameters and variables of the i-th limb.

Kinematic model

By kinematic analysis, the relative displacement, velocity and acceleration of the links of the manipulator are derived. Jacobian matrix, which provides a mapping between the velocity of active joints and the velocity of the end-effector, is extracted in this section.

Looking at Figure 2, it is obvious that the magnitude of $\overrightarrow{B_iC_i}$, representing the length of i-th parallelogram, is constant, so

$$\left|\overrightarrow{\mathrm{B_iC_i}}\right| = 1 \tag{1}$$

From the loop closure equation, we can write

$$\overrightarrow{\mathrm{B_iC_i}} = \overrightarrow{\mathrm{OA_i}} + \overrightarrow{\mathrm{A_iC_i}} - \overrightarrow{\mathrm{PB_i}} - \overrightarrow{\mathrm{OP}} \tag{2}$$

$$\mathrm{l_i}\mathbf{e}_{\mathrm{B_iC_i}} = \mathrm{R}\mathbf{e}_{\mathrm{A_i}} + \mathrm{q}\mathbf{e}_{\mathrm{A_iC_i}} - \mathrm{r}\mathbf{e}_{\mathrm{PB_i}} - \mathbf{p} \tag{3}$$

where,

$$e_{BiCi} = [\cos\alpha_i \cos\phi_i \ \sin\alpha_i \cos\phi_i \ \sin\phi_i]^T \tag{4}$$

$$e_{Ai} = [\cos\alpha_i \ \sin\alpha_i \ 0]^T \tag{5}$$

$$e_{PBi} = [\cos\alpha_i \ \sin\alpha_i \ 0]^T \tag{6}$$

$$\mathbf{p} = \begin{bmatrix} \mathbf{x}_\mathrm{p} & \mathbf{y}_\mathrm{p} & \mathbf{z}_\mathrm{p} \end{bmatrix}^\mathbf{T} \tag{7}$$

Replacing Eqs. (4–7) into Eq. 3 and simplifying the resulting equation yields the following constraint equation from which the forward and inverse kinematics of the PKM can be extracted.[43]

$$f_i = (\mathrm{x}_p + (r - R + q_i \cos\beta)\sin\alpha_i)^2 + (\mathrm{y}_P - (r - R + q_i \cos\beta)\cos\alpha_i)^2 + (\mathrm{z}_P + q_i \sin\beta)^2 - \mathrm{l}^2 = 0 \tag{8}$$

Direct differentiation of the kinematic Eq. 8 with respect to time yields the velocity equations.

$$J_\mathrm{p}\dot{p} + J_q\dot{q} = 0 \tag{9}$$

where the Jacobian matrices are:

$$\mathbf{J}_\mathrm{p} = \partial\mathbf{f} / \partial\mathbf{p} \ , \ \mathbf{J}_\mathrm{q} = \partial\mathbf{f} / \partial\mathbf{q} \tag{10}$$

Considering the end-effector velocity vector as $\dot{\boldsymbol{p}} = [\dot{x}_p \ \ \dot{y}_p \ \ \dot{z}_p]^T$ and the actuators velocity vector as

$\dot{q} = [\dot{q}_1\, \dot{q}_2\, \dot{q}_3]^T$, the following linear mapping, known as forward velocity kinematic equation holds.

$$\dot{p} = J\dot{q} \quad \mathbf{J} = -\mathbf{J}_{\mathrm{p}}^{-1}\mathbf{J}_{\mathrm{q}} \tag{11}$$

The Jacobians can also be derived directly from the following velocity equation.

$$\dot{\boldsymbol{p}} = \dot{\boldsymbol{q}}_i \mathrm{e}_{A_iC_i} + \omega_i \times l\, \mathrm{e}_{B_iC_i} \qquad i = 1,2,3 \tag{12}$$

Dot multiplying both sides of Eq. 12 with $\mathrm{e}_{\mathrm{BC}_i}$ omits the term including the unknown angular velocity of the parallelogram rods, $\dot{\boldsymbol{u}}_i$, and yields

$$\dot{p}.e_{Bi\,Ci} - \dot{q}_i e_{Ai\,Ci}.e_{Bi\,Ci} = 0, i = 1,2,3 \tag{13}$$

which can be written in the matrix form of Eq. 9 where

$$\mathbf{J}_{\mathrm{p}} = \begin{bmatrix} \mathbf{e}_{\mathrm{B_1C_1}}^{\mathrm{T}} & \mathbf{e}_{\mathrm{B_2C_2}}^{\mathrm{T}} & \mathbf{e}_{\mathrm{B_3C_3}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \tag{14}$$

$$\mathbf{J}_{\mathrm{q}} = -\mathrm{diag}\begin{bmatrix} \mathrm{e}_{\mathrm{A_1C_1}}.\mathrm{e}_{\mathrm{B_1C_1}} & \mathrm{e}_{\mathrm{A_2C_2}}.\mathrm{e}_{\mathrm{B_2C_2}} & \mathrm{e}_{\mathrm{A_3C_3}}.\mathrm{e}_{\mathrm{B_3C_3}} \end{bmatrix} \tag{15}$$

Kinematic equations for acceleration can be derived by differentiation of Eq. 9 as:

$$J_p\ddot{p} + J_p\ddot{\mathrm{q}} + \dot{J}_p\dot{\mathrm{p}} + \dot{J}_q\dot{\mathrm{q}} = 0 \tag{16}$$

Dynamic model

In order to analyze the dynamic behavior of the robot, Euler-Lagrange formulation is used. Since in parallel robots the generalized coordinates are constrained, the formulation comprising Lagrange multipliers is applied.

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_j}\right) - \frac{\partial L}{\partial \theta_j} = Q_j + \sum_{i=1}^{3}\lambda_i \frac{\partial f_i}{\partial \theta_j}, (j = 1,2,...,6) \tag{17}$$

where θ_j is the j-th generalized coordinate, Q_j is its corresponding generalized active force, λ_i s are the Lagrange multipliers, and f_i are the constraint equations denoted by Eq 8. Generalized coordinates are defined as

$$\theta = \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{q} \end{bmatrix} \tag{18}$$

Lagrangian is defined as the difference between kinetic and potential energies.

$$L(\theta,\dot{\theta}) = \mathrm{K}(\theta,\dot{\theta}) - U(\theta) \tag{19}$$

Due to the fact that rotational inertia of the parallelogram rods is negligible, they are considered as two point-masses on their both extreme ends. Thus, the kinetic energy of the whole manipulator will be calculated by adding the kinetic energy of the sliding actuators and that of the end-effector.

$$\mathrm{K} = \frac{1}{2}\mathrm{m}_{\mathrm{q}}\dot{\mathbf{q}}^{\mathrm{T}}\dot{\mathbf{q}} + \frac{1}{2}\mathrm{m}_{\mathrm{p}}\dot{\mathbf{p}}^{\mathrm{T}}\dot{\mathbf{p}} \tag{20}$$

$$\mathrm{m}_{\mathrm{q}} = \mathrm{m}_{\mathrm{a}} + \frac{\mathrm{m}_{\mathrm{II}}}{2}, \quad \mathrm{m}_{\mathrm{p}} = \mathrm{m}_{\mathrm{e}} + 3\left(\frac{\mathrm{m}_{\mathrm{II}}}{2}\right) \tag{21}$$

where m_e, m_{II}, and m_a represent the masses of the end-effector, the parallelogram, and the actuator piston, respectively. Potential energy of the whole mechanism is obtained from

$$U = -m_q g \sin\beta(q_1 + q_2 + q_3) + m_p g z_p \tag{22}$$

Therefore, the Lagrangian of the system can be expanded as follows

$$L = \frac{1}{2}m_q\dot{q}^T\dot{q} + \frac{1}{2}m_p\dot{p}^T\dot{p} + m_q g \sin\beta(\mathrm{q}_1 + q_2 + q_3) - m_p q z_p \tag{23}$$

Substituting the resulting Lagrangian into Eq. 17 and after some mathematical simplification, the resulting equations can be written in the following matrix form.[45]

$$M_1\ddot{q} + M_2\ddot{p} + G = \tau \tag{24}$$

Besides these three equations, three others are also needed to determine the six unknown generalized coordinates. These extra equations are the acceleration equations, Eq. (16)

$$J_q\ddot{q} + J_p\ddot{p} + \dot{J}_q\dot{q} + \dot{J}_p\dot{p} = 0 \tag{25}$$

The six equations of Eq. (24), (25) describe the dynamics of the robot. Eliminating $\ddot{p}$ and $\dot{p}$ from these equations, the reduced dynamic equations are obtained as:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \tag{26}$$

where,

$$\mathbf{M}(\mathbf{q}) = \mathbf{M}_1 + \mathbf{M}_2\mathbf{J}$$

$$\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = -\mathbf{M}_2\mathbf{J}_{\mathrm{p}}^{-1}\left(\dot{\mathbf{J}}_{\mathrm{q}} + \dot{\mathbf{J}}_{\mathrm{p}}\mathbf{J}\right) \tag{27}$$

In order to consider uncertainties and actuator faults, the dynamic model Eq. 26 can be rewritten as (30),

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + \beta(\mathrm{t} - \mathrm{T}_f)\delta(q,\dot{q},\tau) + \sigma(q,\dot{q},\tau) = \tau \tag{28}$$

where $\delta(\boldsymbol{q},\dot{\boldsymbol{q}},\tau)$, $\beta(t - T_f)$, T_f, and $\sigma(\boldsymbol{q},\dot{\boldsymbol{q}},\tau)$ are the fault vector, its time profile, the fault occurrence time, and uncertainties, respectively. The time profile of the fault is as follows:

$$\beta(\mathrm{t} - \mathrm{T}_f) = \begin{cases} 0 & t < T_f \\ 1 - e^{\nu(t-Tf)} & t \geq T_f \end{cases} \tag{29}$$

where ν denotes the growth rate of the fault. In this paper, actuator fault is taken into account which is one of the frequent failures in robotics. Small value or ν means that the fault is developing at slow rate, usually called an incipient fault. On the other hand, when this parameter has a large value, its time profile, γ, tends to a step function and is called an abrupt fault. Clearly, as $\nu \to \infty$, γ becomes a step function and the incipient fault turns to the abrupt fault. When an actuator fails, its force can be defined as:

$$\tau = \tau_{nominal} + \delta\tau \tag{30}$$

The actuator fault, $\delta\tau(t)$, can be written as the fault function defined by $\boldsymbol{F}(\boldsymbol{q},\dot{\boldsymbol{q}},\tau) = \mathrm{M}^{-1}\delta\tau(t)$.

Now, the dynamic model of the robot can be rewritten as:

$$\ddot{q} = M(q)^{-1}\tau + N(q,\dot{q}) + \Delta(q,\dot{q},\tau) + F(q,\dot{q},\tau) \tag{31}$$

$$\mathbf{N}(\mathbf{q},\dot{\mathbf{q}}) = -\mathbf{M}(\mathbf{q})^{-1}\left[\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q})\right]$$

$$\Delta(q,\dot{q},\tau) = -M(q)^{-1}\beta(t - T_f)\delta(q,\dot{q},\tau)$$

$$\mathrm{F}(q,\dot{q},\tau) = -M(q)^{-1}\sigma(q,\dot{q},\tau)$$

where $\boldsymbol{N}$, Δ, and $\boldsymbol{F}$ introduce the nominal robot dynamics, uncertainties, and actuator faults, respectively.

Proposed control strategy

In this section, first of all, a nonlinear $\boldsymbol{H_\infty}$ controller is designed to control a parallel robot in the presence of actuator faults, uncertainties and also exogenous disturbances.

The dynamic model of the parallel robot can be rewritten in the following state-space form:

$$\begin{cases} \dot{q}_1 = q_2 \\ q_2 = M(q)^{-1}\tau + N(q,\dot{q}) + \phi(q,\dot{q},\tau) \end{cases} \tag{32}$$

where $\phi(q,\dot{q},\tau) = \Delta(q,\dot{q},\tau) + F(q,\dot{q},\tau)$. A performance index can be defined using the following cost variable:

$$\xi = W\begin{bmatrix} h(q) \\ \tau \end{bmatrix} \tag{33}$$

where $\boldsymbol{W}$ is a weighting function and $\boldsymbol{h(q)}$ is a function of states which should be controlled. If the robot states are available, then the H_∞ performance index is as follows (44):

$$\int_0^T \|\xi\|^2 \,\mathrm{dt} \le \gamma^2 \int_0^T \|\mathbf{d}\|^2 \,\mathrm{dt} \tag{34}$$

in which

$$\|\xi\|^2 = \xi'\xi = [h'(q)\tau']W'W\begin{bmatrix} h(q) \\ \tau \end{bmatrix}W'W = \begin{bmatrix} Q & S \\ S' & R \end{bmatrix} \tag{35}$$

where $\boldsymbol{Q}$ and $\boldsymbol{R}$ are symmetric positive matrices. By relying on the $\boldsymbol{W'W} > 0$, then $\boldsymbol{Q - SR^{-1}S'} > 0$. Therefore, the optimal control signal can be determined based on the solution of HJBI:[46]

$$\frac{\partial \mathrm{V}}{\partial t} + \frac{\partial' \mathrm{V}}{\partial \mathbf{q}}\mathbf{f}(\mathbf{q},\mathbf{t}) + \frac{1}{2}\frac{\partial' \mathrm{V}}{\partial \mathbf{q}}\left[\frac{1}{\tilde{a}^2}\mathbf{k}(\mathbf{q},\mathbf{t})\mathbf{k}'(\mathbf{q},\mathbf{t}) - \mathbf{g}(\mathbf{q},\mathbf{t})\mathbf{R}^{-1}\mathbf{g}(\mathbf{q},\mathbf{t})\right]\frac{\partial \mathrm{V}}{\partial \mathbf{q}} - \frac{\partial' \mathrm{V}}{\partial \mathbf{q}}\mathrm{g}(\mathbf{q},\mathbf{t})\mathrm{R}^{-1}\mathbf{S}'\mathbf{h}(\mathbf{q}) + \frac{1}{2}\mathbf{h}'(\mathbf{q})\left(\mathbf{Q} - \mathbf{S}\mathbf{R}^{-1}\mathbf{S}'\right)\mathbf{h}(\mathbf{q}) = 0 \tag{36}$$

For each $\gamma > \sqrt{\sigma_{\max}(\boldsymbol{R})} \ge 0$ (maximum singular value). In this case, the optimal state feedback control command can be derived as:

$$\tau^* = -\mathbf{R}^{-1}\left(\mathbf{S}'\mathbf{h}(\mathbf{q}) + \mathbf{g}'(\mathbf{q},\mathbf{t})\frac{\partial \mathrm{V}}{\partial \mathbf{q}}\right) \tag{37}$$

By defining the tracking error vector $\boldsymbol{x} = \left[\dot{\boldsymbol{e}} \quad \boldsymbol{e} \quad \int \boldsymbol{e}dt\right]$ where $\boldsymbol{e} = \boldsymbol{q} - \boldsymbol{q}_d$, $\dot{\boldsymbol{e}} = \dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_d$ and $\int edt = \int (\boldsymbol{q} - \boldsymbol{q}_d)dt$. Due to acting persistence disturbances on the parallel robot during the whole process, the integral term is added to nullifying steady state error. Then, the following control law is taken into account:

$$\tau = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) - \frac{1}{\mathbf{T}_1}\left(\mathbf{M}(\mathbf{q})\mathbf{T}\dot{\mathbf{x}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\mathbf{T}\mathbf{x}\right) + \frac{1}{\mathbf{T}_1}\mathbf{u} \tag{38}$$

$$\mathbf{T} = \left[\mathbf{T}_1 \quad \mathbf{T}_2 \quad \mathbf{T}_3\right] \quad \mathbf{T}_1 = \rho\mathbf{I} \quad \rho > 0$$

where I is the nth-order identity matrix. By substituting the control command into the dynamic model, then the error dynamics can be expressed by the following expression:

$$\mathbf{M}(\mathbf{q})\mathbf{T}\dot{\mathbf{x}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\mathbf{T}\mathbf{x} = \mathbf{u} + \mathbf{d} \tag{39}$$

By considering Eq. ß9, we try to determine an optimal control signal so that

$$\frac{\|\xi\|}{\|\mathbf{d}\|} \le \tilde{a}^2 \tag{40}$$

Based on the above vector error and cost variable ξ,

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_{12} & \mathbf{Q}_{13} \\ \mathbf{Q}_{12} & \mathbf{Q}_2 & \mathbf{Q}_{23} \\ \mathbf{Q}_{13} & \mathbf{Q}_{23} & \mathbf{Q}_3 \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \mathbf{S}_1 \\ \mathbf{S}_2 \\ \mathbf{S}_3 \end{bmatrix} \tag{41}$$

By rewriting the error dynamics into the standard form of the nonlinear H_∞ problem as:

$$\dot{\mathbf{q}} = \mathbf{f}(\mathbf{q},\mathbf{t}) + \mathbf{g}(\mathbf{q},\mathbf{t})\mathbf{u} + \mathbf{k}(\mathbf{q},\mathbf{t})\phi$$

$$\mathbf{f}(\mathbf{q},\mathbf{t}) = \mathbf{T}_0^{-1}\begin{bmatrix} -\mathbf{M}^{-1}(\mathbf{q})\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) & 0 & 0 \\ \mathbf{T}_1^{-1} & \mathbf{I} - \mathbf{T}_1^{-1}\mathbf{T}_2 & -\mathbf{I} + \mathbf{T}_1^{-1}(\mathbf{T}_2 - \mathbf{T}_3) \\ 0 & \mathbf{I} & -\mathbf{I} \end{bmatrix}\mathbf{T}_0\mathbf{x}$$

$$\mathbf{g}(\mathbf{q},\mathbf{t}) = \mathbf{k}(\mathbf{q},\mathbf{t}) = \mathrm{T}_0^{-1}\begin{bmatrix} \mathbf{M}^{-1}(\mathbf{q}) \\ 0 \\ 0 \end{bmatrix} \qquad \mathrm{T}_0 = \begin{bmatrix} \mathrm{T}_1 & \mathrm{T}_2 & \mathrm{T}_3 \\ 0 & \mathbf{I} & \mathbf{I} \\ 0 & 0 & \mathbf{I} \end{bmatrix} \tag{42}$$

It should be noted that HJBI solution depends on the selection of both cost variable and $h(q)$ which is considered equally to the error vector. By taking such function into account, a Lyapunov function should be chosen in order to determine the control law.

Theorem: Take the following Lyapunov function:

$$\mathrm{V}(\mathbf{q},\mathbf{t}) = 0.5\ \mathbf{x}'\ \mathbf{T}'_0\begin{bmatrix} \mathbf{M}(\mathbf{q}) & 0 & 0 \\ 0 & \mathbf{Y} & \mathbf{X} - \mathbf{Y} \\ 0 & \mathbf{X} - \mathbf{Y} & \mathbf{Z} + \mathbf{Y} \end{bmatrix}\mathbf{T}_0\mathrm{x} \tag{43}$$

Where $\boldsymbol{X}, \boldsymbol{Y}$ and $\boldsymbol{Z}$ are symmetric and positive definite matrices so that $\boldsymbol{Z - XY^{-1}X + 2X} > 0$ by verifying the following expression, for the high value of $\tilde{a}$, the candidate function embodies a solution of HJBI.

Some Riccati algebraic equations are solved to obtain $\boldsymbol{T} = \left[\boldsymbol{T}_1 \quad \boldsymbol{T}_2 \quad \boldsymbol{T}_3\right]$. By substituting the Lyapunov function in Eq.37, the control law can be rewritten as:

$$\tau^* = -\mathbf{R}^{-1}(\mathbf{S}' + \mathbf{T})\mathbf{x} \tag{44}$$

By replacing the control law in Eq. 10 and some mathematical simplification, the control law is as:

$$\tau = \boldsymbol{M}(\boldsymbol{q})\ddot{q} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) - \boldsymbol{M}(\boldsymbol{q})\left(\boldsymbol{K}_P\boldsymbol{e} + \boldsymbol{K}_I\int\boldsymbol{e}dt + \boldsymbol{K}_D\dot{\boldsymbol{e}}\right)$$

$$\mathbf{K}_{\mathrm{P}} = \mathbf{T}_1^{-1}\left(\mathbf{T}_3 + \mathbf{M}^{-1}(\mathbf{q})\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\mathbf{T}_2 + \mathbf{M}^{-1}(\mathbf{q})\mathbf{R}^{-1}\left(\mathbf{S}_2^{'} + \mathbf{T}_2\right)\right)$$

$$\mathbf{K}_{\mathrm{I}} = -\mathbf{T}_1^{-1}\left(\mathbf{M}^{-1}(\mathbf{q})\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\mathbf{T}_3 + \mathbf{M}^{-1}(\mathbf{q})\mathbf{R}^{-1}\left(\mathbf{S}_3^{'} + \mathbf{T}_3\right)\right)$$

$$\mathbf{K}_{\mathrm{D}} = \mathbf{T}_1^{-1}\left(\mathbf{T}_2 + \mathbf{M}^{-1}(\mathbf{q})\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\mathbf{T}_1 + \mathbf{M}^{-1}(\mathbf{q})\mathbf{R}^{-1}\left(\mathbf{S}_1^{'} + \mathbf{T}_1\right)\right)$$

By defining

$$Q_1 = \omega_1^2 I\, R = \omega_u^2 \boldsymbol{I}\, Q_{12} = Q_{13} = Q_{23} = 0$$
$$\boldsymbol{Q}_2 = \omega_2^2 \boldsymbol{I} \tag{46}$$
$$Q_3 = \omega_3^2 \boldsymbol{I}\, S_1 = S_2 = S_3 = 0$$

Now, the above gains can be rewritten as:

$$\mathbf{K}_{\mathrm{P}} = \frac{\omega_3}{\omega_1}\mathbf{I} + \frac{\sqrt{\omega_2^2 + 2\omega_1\omega_3}}{\omega_1}\mathbf{M}^{-1}(\mathbf{q})\left(\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) + \frac{1}{\omega_u^2}\mathbf{I}\right)$$

$$\mathbf{K}_{\mathrm{I}} = \frac{\omega_3}{\omega_1}\mathbf{M}^{-1}(\mathbf{q})\left(\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) + \frac{1}{\omega_u^2}\mathrm{I}\right) \tag{47}$$

$$\mathbf{K}_{\mathrm{D}} = \frac{\sqrt{\omega_2^2 + 2\omega_1\omega_3}}{\omega_1}\mathbf{I} + \mathbf{M}^{-1}(\mathbf{q})\left(\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) + \frac{1}{\omega_u^2}\mathbf{I}\right)$$

In which ω_1, ω_2 and ω_3 are constant parameters. More details regarding stability analysis is presented in 43.

Remark: by considering the aforementioned control scheme, it can be obviously seen that should constant parameters obtain properly, stability of the closed-loop system in the presence of the exogenous disturbances, uncertainties and actuator faults will be guaranteed in finite time.

It is worth mentioning that the complex faults and disturbances are as follows. The fault is occurred at 1.5 sec.

$$\delta\tau = \begin{bmatrix} 2q_2^2 + 5\dot{q}_1 q_3 + 14\sin\dot{q}_3 \\ 20q_1^2 + 15\dot{q}_1 + 14\cos q_1 \\ 0.3U_{FTCHOT_3} + 6\cos\dot{q}_3 \end{bmatrix} \tag{48}$$

$$\mathrm{d} = \begin{bmatrix} 0.1\mathrm{q}_1^2 + \sin\mathrm{t}\,\mathrm{q}_1 \\ 0.15\cos\mathrm{t}\,\mathrm{q}_2 \\ 0.1\sin\mathrm{t} + 0.8\mathrm{q}_3^2 \end{bmatrix} \tag{49}$$

As the controller performance is affected by actuator saturation, it is assumed that the maximum value of the control inputs is 100 N.m, in other words, $|\tau| \le 100$ ..

Fault detection based on a super-twisting third-order sliding mode observer

In this section, the observer scheme that is used for both state observer and fault diagnosis based on STW-TOSM observer is described. By defining $\boldsymbol{x}_1 = \boldsymbol{q}$ and $\boldsymbol{x}_2 = \dot{\boldsymbol{q}}$, the dynamic model Eq. can be rewritten in following form:

$$\dot{\mathbf{x}}_1 = \mathbf{x}_2$$
$$\dot{x}_2 = N(x_1,x_2,\tau) + \Delta(x_1,x_2,\tau) + F(x_1,x_2,\tau)$$
$$\mathbf{y} = \mathbf{x}_1 \tag{50}$$

The third order super twisting sliding mode observer is as follows [45],

$$\dot{\hat{x}}_1 = \hat{x}_2 + \gamma_2\left\|x_1 - \hat{x}_1\right\|^{\frac{2}{3}} sign(x_1 - \hat{x}_1)$$
$$\dot{\hat{x}}_2 = N(x_1,\hat{x}_2,\tau) + \gamma_1\left\|\dot{\hat{x}}_1 - \hat{x}_2\right\|^{\frac{1}{2}} sign(\dot{\hat{x}}_1 - \hat{x}_2) + \hat{z}_{eq} \tag{51}$$
$$\dot{\hat{z}}_{eq} = \gamma_0 sign(\dot{\hat{x}}_1 - \hat{x}_2)$$

in which γ_i are the sliding mode gains. By taking into account the estimation error as:

$$\dot{\tilde{x}}_1 = \tilde{x}_2 - \gamma_2\left\|x_1 - \hat{x}_1\right\|^{2/3} sign(x_1 - \hat{x}_1)$$
$$\dot{\tilde{x}}_2 = d(x_1,\hat{x}_2,\tilde{x}_2) + \Delta(\theta,\dot{\theta},\tau) - \gamma_1\left\|\dot{\hat{x}}_1 - \hat{x}_2\right\|^{\frac{1}{2}} sign(\dot{\hat{x}}_1 - \hat{x}_2) - \hat{z}_{eq}$$
$$\dot{\hat{z}}_{eq} = \gamma_0 sign(\dot{\hat{x}}_1 - \hat{x}_2)$$
$$d(x_1,\hat{x}_2,\tilde{x}_2) = N(x_1,x_2,\tau) - N(x_1,\hat{x}_2,\tau) \tag{52}$$

where $\tilde{\boldsymbol{x}}_i$ are the estimation error. By defining

$$\mathrm{H}(x_1,x_2,\hat{x}_2,\tau) = d(x_1,\hat{x}_2,\tilde{x}_2) + \Delta(x_1,x_2,\tau) + \mathrm{F}(x_1,x_2,\tau) \tag{53}$$

and also by having the following assumptions:

$$\Delta(x_1,x_2,\tau) \le \bar{\Delta}$$
$$\mathrm{F}(x_1,x_2,\tau) \le \bar{\varphi} \tag{54}$$

then, a constant parameter is existing so that:

$$\mathrm{H}(x_1,x_2,\hat{x}_2,\tau) < f^+ \tag{55}$$

According to the analysis performed in [35], the sliding gains can be chosen as follows which ensure the stability and convergence of the system.

$$\gamma_2 = 1.9 f^{+1/3}$$
$$\gamma_1 = 1.5 f^{+1/2}$$
$$\gamma_0 = 1.1 f^+ \tag{56}$$

After converging the estimation error to zero, estimated states would be reached to actual stated and the following equality will be hold:

$$\Delta(x_1,x_2,\tau) + F(x_1,x_2,\tau) - \gamma_1\hat{x}_1 - \hat{x}_2^{\frac{1}{2}} sign(\hat{x}_1 - \hat{x}_2) - \hat{z}_{eq} = 0 \tag{57}$$

When the observer converges to zero, the third term of Eq. 57 is equal to zero. Then, uncertainty and fault can be reconstructed as:

$$\hat{z}_{eq} = \Delta(x_1,x_2,\tau) + F(x_1,x_2,\tau) \tag{58}$$

Where $\hat{z}_{eq}$ is a continuous term and low pass filter for determining equivalent output injection is not required.

Therefore, they are an exact estimation of the uncertainty and fault without filtration which could contribute the performance of AFTC due to requiring accurate fault estimation.

Path planning

The objective of optimal path planning is to design a path for the end-effector in an area containing some obstacles, so that it is of minimum length and avoids any collision with the obstacles. A predetermined margin is considered around the obstacles and the end-effector would keep a minimum distance from the borders of the obstacles.

To create a 2-D trajectory from the starting point, $P_0 = (X_0, Y_0)$, to the final point, $P_f = (X_f, Y_f)$, in the time interval of (T_0, T_f), a number of accuracy points are considered as:

$$\mathrm{P_i} = (\mathrm{X_i}, \mathrm{Y_i}) @ \mathrm{T_i} \; (\mathrm{i} = 1, 2, \ldots, \mathrm{n}) \quad (59)$$

Where T_i is the time instant at which the end-effector is planned to be located at P_i. All accuracy points will be checked to be placed in the reachable workspace and far enough from singular points. End-effector trajectory is derived by cubic spline interpolation of accuracy points coordinates X_i and Y_i with respect to time that guarantees the continuity of velocity and acceleration. So, the trajectory curve can be obtained as,

$$\mathrm{x_d}(\mathrm{t}) = \mathrm{spline}(\mathrm{X}, \mathrm{T}, \mathrm{t})$$

$$\mathrm{y_d}(\mathrm{t}) = \mathrm{spline}(\mathrm{Y}, \mathrm{T}, \mathrm{t})$$

$$\mathrm{z_d}(\mathrm{t}) = \mathrm{spline}(\mathrm{Z}, \mathrm{T}, \mathrm{t}) \quad (60)$$

Where, N is the number of interpolated points along the curve. Spatial and time vectors are defined as,

$$\mathbf{X} = [\mathrm{X_0} \quad \ldots \quad \mathrm{X_f}], \qquad \mathbf{Y} = [\mathrm{Y_0} \quad \ldots \quad \mathrm{Y_f}], \qquad \mathbf{T} = [\mathrm{T_0} \quad \ldots \quad \mathrm{T_f}] \quad (61)$$

Numerical simulation

To demonstrate effectiveness of the proposed fault-tolerant control scheme in performing trajectory tracking task in the presence of uncertainties, and actuator fault and saturation, some simulations have been performed. The desired path is formed by spline interpolation, passing through some accuracy points, and avoiding some obstacles in the workspace. System initial conditions are extracted from kinematic solution. To compare the performance of the proposed scheme with other methods, simulations are also conducted for feedback linearization and conventional sliding mode control schemes. Measurement noise is considered in this simulation. Control parameters are presented in Table 1.

Table 1 *Controller parameters*

Parameter	Value
ω_1	0.5
ω_2	8
ω_3	5
ω_4	0.12

Actuator trajectories and their errors are illustrated in Figure 3 and Figure 4 As can be seen from the results, the proposed controller is the most robust one compared to the other schemes, which upon occurring fault at 1.5 sec, it has maintained its high performance and followed the desired trajectories with minimum tracking errors. Moreover, finite time convergence is also evident in these figures. Figure 5 demonstrates how the desired trajectories in workspace have been followed by different controllers and how much error, and the corresponding errors have been presented in Figure 6. As can be seen, under the influence of actuator faults the tracking performance of the SM and FL controllers is declined significantly, while H_∞ tracks the desired trajectories with high accuracy. Nullifying the influence of actuator fault and uncertainty would be the obvious fact with regards to superiority of the proposed H_∞. The paths followed by the end effector under the influence of different controllers are illustrated in Figure 7. As is clear from the results, SM and FL controllers diverge from the desired path as the faults occur in the system. H_∞, on the other hand, is capable of following the path with high precision. The control commands presented in Figure 8 reveal a reasonable range and fluctuation-free pattern of the signal. This has attained as a result of employing high-order algorithm. As the fault has occurred at 1.5 sec, the system has experienced a slight change.

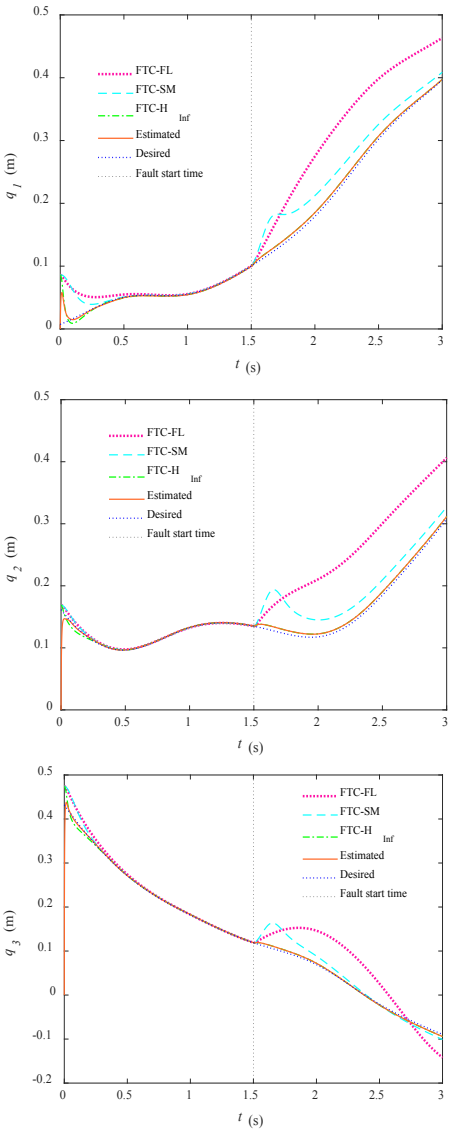


Figure 3 *Actuator motion plots.*

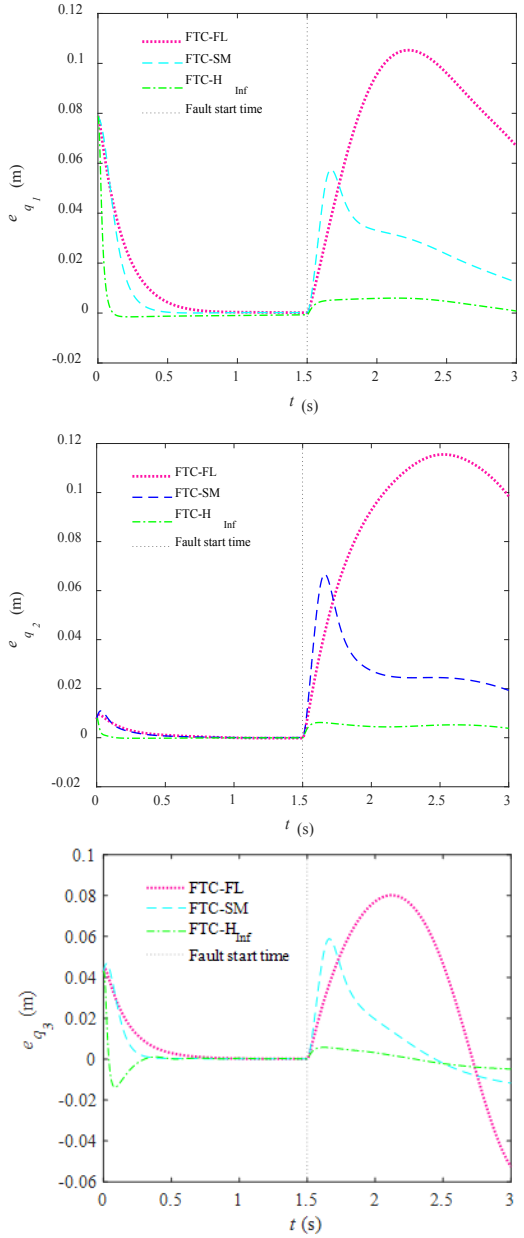
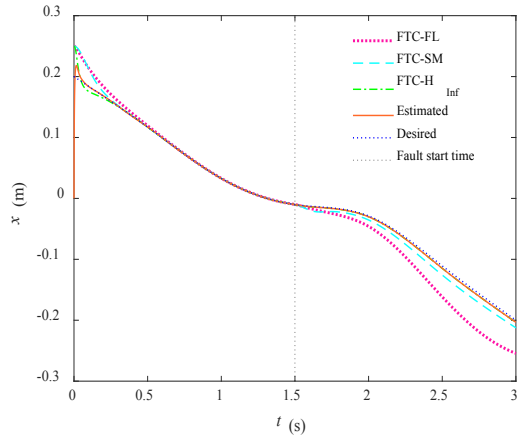


Figure 4 *Tracking error components.*



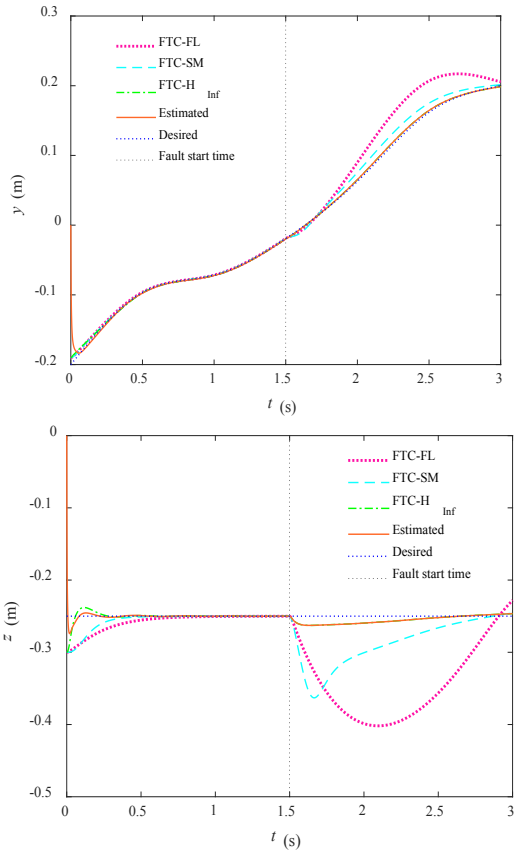
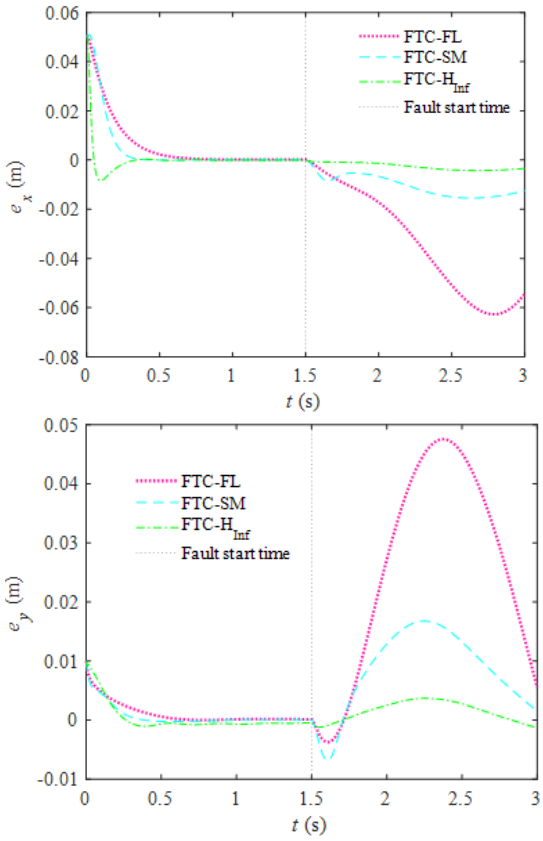


Figure 5 *End-effector motion plots.*



Citation: Mazare M, Taghizadeh M, Ghanbari PG. Active fault tolerant control based on nonlinear H_∞ subject to actuator and sensor faults for a parallel robot. *Int Rob Auto J.* 2020;6(3):115–125. DOI: 10.15406/iratj.2020.06.00210

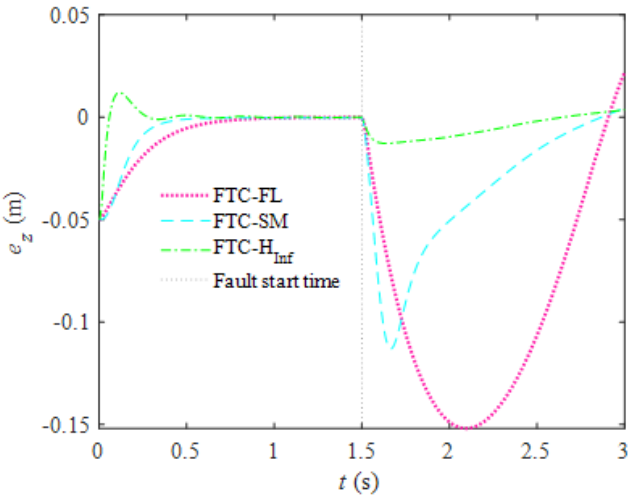


Figure 6 *Tracking error components.*

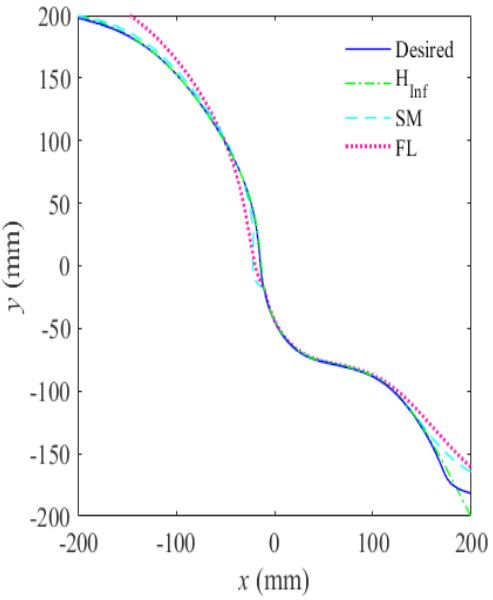


Figure 7 End-effector path following.

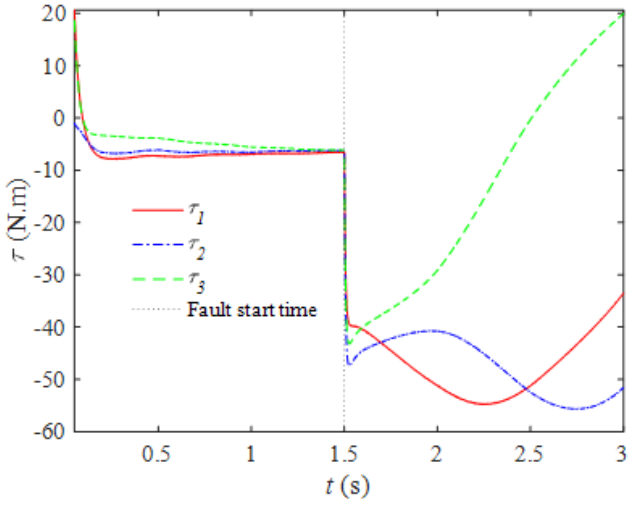


Figure 8 *Actuator forces in the presence of actuator fault.*

In order to show a vivid comparison between the three controllers, three error criteria termed as Integral of the Time multiplied by the Absolute value of the Error (ITAE), Integral of the Time multiplied by the Absolute value of the Squared of the Error (ITASE) and Integral of the Absolute value of the Error (IAE) have been studied, which are defined as,

$$\text{ITAE} = \int |\mathbf{e}(t).t| dt$$

$$\text{ITASE} = \int t.|\mathbf{e}(t)|^2 dt$$

$$\text{IASE} = \int |\mathbf{e}(t)|^2 dt \quad (62)$$

Figure 9 Reveals that FTC on the basis of H_∞ has the minimum value for the three criteria.

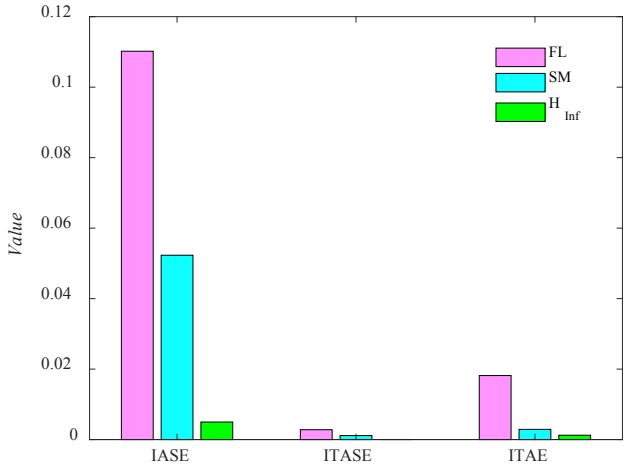
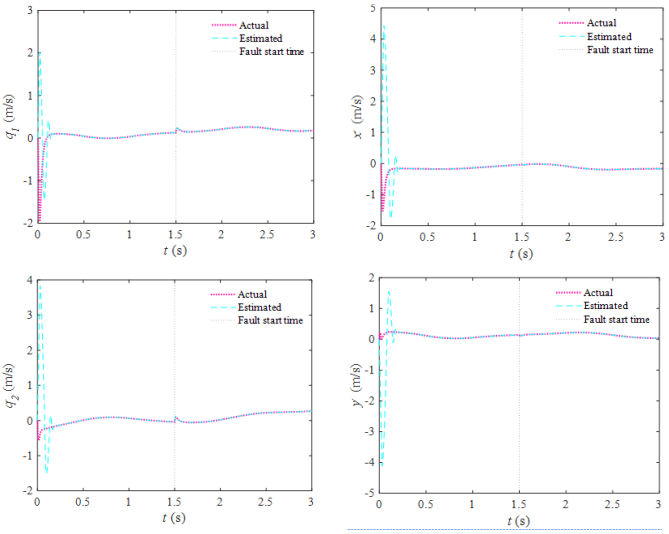


Figure 9 Error criteria in the presence of actuator fault.

To show the effectiveness of the TW-TOSM observer, the estimated velocity in joint space and workspace are compared with their actual values in Figure 10. As is clear from the result, this observer is able to estimate the velocities with high accuracy and no chattering. Fault detection performance of the H_∞ -based FTC is presented in Figure 11 through the obtained residuals from the simulations. As the fault occurs, the residuals cross the predefined threshold, showing the presence of the fault in the system.



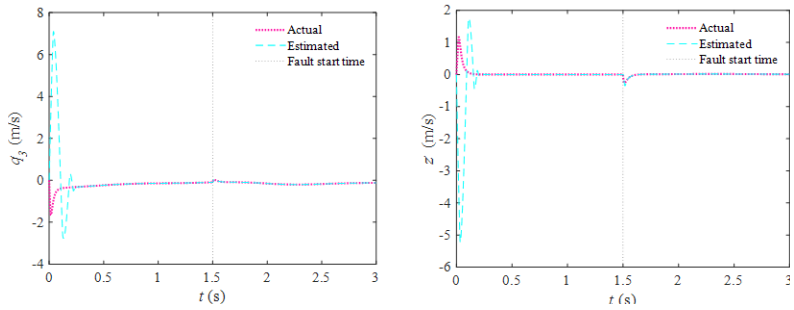


Figure 10 Actual and estimated velocities.

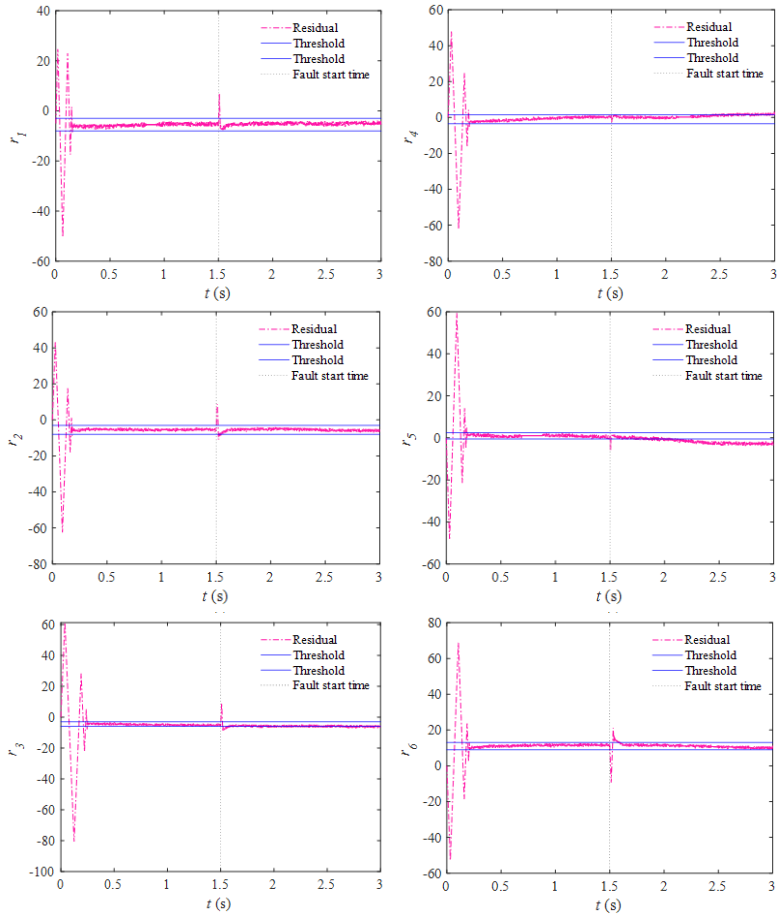


Figure 11 Residuals of a faulty system under the effect of actuator fault.

Conclusion

In this paper, a robust Fault Tolerant Control based on nonlinear H_∞ was proposed for a delta type parallel robot exposed to actuator fault. For fault detection a super-twisting third-order sliding mode (STW-TOSM) observer was exploited which can accommodate faults and uncertainties without velocity measurement. In addition, it provides fast convergence and high accuracy thanks to its high-order sliding mode algorithm. The simulation results proved that the proposed FTC offers high accuracy, finite time convergence for reasonable control effort compared to conventional SMC- and FL-based FTC schemes.

Funding

None.

Acknowledgments

None.

Conflicts of interest

The authors declare that there was no conflict of interest.

References

1. Brinker J, Corves B, Takeda Y. Kinematic performance evaluation of high-speed Delta parallel robots based on motion/force transmission indices. *Mechanism and Machine Theory*. 2018;125:111–125.
2. Guilin Y, Chen IM, Wei L, et al. Singularity analysis of three-legged parallel robots based on passive-joint velocities. *IEEE Transactions on Robotics and Automation*. 2001;17(4):413–422.

3. Chen M, Tao G. Adaptive Fault-Tolerant Control of Uncertain Nonlinear Large-Scale Systems With Unknown Dead Zone. *IEEE Transactions on Cybernetics*. 2016;46(8):1851–1862.

4. Hu Q, Shi Y, Shao X. Adaptive fault-tolerant attitude control for satellite reorientation under input saturation. *Aerospace Science and Technology*. 2018;78:171–182.

5. Gao Z, Zhou Z, Qian MS, et al. Active fault tolerant control scheme for satellite attitude system subject to actuator time-varying faults. *IET Control Theory &amp Applications*. 2017;12(3):405–412.

6. Meng Q, Zhang T, Gao X, et al. Adaptive Sliding Mode Fault-Tolerant Control of the Uncertain Stewart Platform Based on Offline Multibody Dynamics. *IEEE/ASME Transactions on Mechatronics*. 2014;19(3):882–894.

7. Meng Q, Zhang T, He TF, et al. Adaptive vector sliding mode fault-tolerant control of the uncertain Stewart platform based on position measurements only. *Robotica*. 2016;34(6):1297–1321.

8. Farid Y, Majd VJ, Ehsani-Seresht A. Fractional-order active fault-tolerant force-position controller design for the legged robots using saturated actuator with unknown bias and gain degradation. *Mechanical Systems and Signal Processing*. 2018;104:465–486.

9. Karras GC, Fourlas GK. Model Predictive Fault Tolerant Control for Omni-directional Mobile Robots. *Journal of Intelligent & Robotic Systems*. 2020;97(3):635–655.

10. Azizi A, Nourisola H, Shoja-Majidabad S. Fault tolerant control of wind turbines with an adaptive output feedback sliding mode controller. *Renewable Energy*. 2019;135:55–65.

11. Lan J, Patton RJ, Zhu X. Fault-tolerant wind turbine pitch control using adaptive sliding mode estimation. *Renewable Energy*. 2018;116:219–231.

12. Cho S, Gao Z, Moan T. Model-based fault detection, fault isolation and fault-tolerant control of a blade pitch system in floating wind turbines. *Renewable Energy*. 2018;120:306–321.

13. Li B, Du H, Li W. Fault-tolerant control of electric vehicles with in-wheel motors using actuator-grouping sliding mode controllers. *Mechanical Systems and Signal Processing*. 2016;72-73:462–485.

14. Jang JO. Neuro-fuzzy networks saturation compensation of DC motor systems. *Mechatronics*. 2009;19(4):529–534.

15. Gao Z, Cecati C, Ding SX. A Survey of Fault Diagnosis and Fault-Tolerant Techniques—Part I: Fault Diagnosis With Model-Based and Signal-Based Approaches. *IEEE Transactions on Industrial Electronics*. 2015;62(6):3757–3767.

16. Jiang J, Yu X. Fault-tolerant control systems: A comparative study between active and passive approaches. *Annual Reviews in Control*. 2012;36(1):60–72.

17. Benosman M, Lum K. Passive Actuators' Fault-Tolerant Control for Affine Nonlinear Systems. *IEEE Transactions on Control Systems Technology*. 2010;18(1):152–163.

18. Zhang R, Qiao J, Li T, et al. Robust fault-tolerant control for flexible spacecraft against partial actuator failures. *Nonlinear Dynamics*. 2014;76(3):1753–1760.

19. Allerhand LI, Shaked U. Robust Switching-Based Fault Tolerant Control. *IEEE Transactions on Automatic Control*. 2015;60(8):2272–2276.

20. Shen Q, Jiang B, Cocquempot V. Adaptive Fuzzy Observer-Based Active Fault-Tolerant Dynamic Surface Control for a Class of Nonlinear Systems With Actuator Faults. *IEEE Transactions on Fuzzy Systems*. 2014;22(2):338–349.

21. Awan ZS, Ali K, Iqbal J, et al. Adaptive Backstepping Based Sensor and Actuator Fault Tolerant Control of a Manipulator. *Journal of Electrical Engineering & Technology*. 2019;14(6):2497–2504.

22. Shi P, Liu M, Zhang L. Fault-Tolerant Sliding-Mode-Observer Synthesis of Markovian Jump Systems Using Quantized Measurements. *IEEE Transactions on Industrial Electronics*. 2015;62(9):5910–5918.

23. Yin S, Yang H, Kaynak O. Sliding Mode Observer-Based FTC for Markovian Jump Systems With Actuator and Sensor Faults. *IEEE Transactions on Automatic Control*. 2017;62(7):3551–3558.

24. Davila J, Fridman L, Levant A. Second-order sliding-mode observer for mechanical systems. *IEEE Transactions on Automatic Control*. 2005;50(11):1785–1789.

25. Utkin V. On Convergence Time and Disturbance Rejection of Super-Twisting Control. *IEEE Transactions on Automatic Control*. 2013;58(8):2013–2017.

26. Guzmán E, Moreno JA. Super-twisting observer for second-order systems with time-varying coefficient. *IET Control Theory & Applications*. 2014;9(4):553–562.

27. Shtessel Y, Taleb M, Plestan F. A novel adaptive-gain super twisting sliding mode controller: Methodology and application. *Automatica*. 2012;48(5):759–769.

28. Thanh HLNN, Hong SK. Quadcopter Robust Adaptive Second Order Sliding Mode Control Based on PID Sliding Surface. *IEEE Access*. 2018;6:66850–66860.

29. KC Veluvolu, Van M, Franciosa P, et al. Fault Diagnosis and Fault-Tolerant Control of Uncertain Robot Manipulators Using High-Order Sliding Mode. *Mathematical Problems in Engineering*. 2016.

30. Van M, Kang HJ, Suh YS, et al. A robust fault diagnosis and accommodation scheme for robot manipulators. *International Journal of Control, Automation and Systems*. 2013;11(2):377–388.

31. Van M, Kang HJ, Suh YS. A novel neural second-order sliding mode observer for robust fault diagnosis in robot manipulators. *International Journal of Precision Engineering and Manufacturing*. 2013;14(3):397–406.

32. Van M, Mavrovouniotis M, Ge SS. An Adaptive Backstepping Nonsingular Fast Terminal Sliding Mode Control for Robust Fault Tolerant Control of Robot Manipulators. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*. 2019;49(7):1448–1458.

33. Van M, Kang HJ. Robust fault-tolerant control for uncertain robot manipulators based on adaptive quasi-continuous high-order sliding mode and neural network. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*. 2015;229(8):1425–1446.

34. Piltan F, Kim H, Kim JM. Advanced Adaptive Fault Diagnosis and Tolerant Control for Robot Manipulators. *Energies*. 2019;12(7).

35. Van M, Franciosa P, Ceglarek D. Fault Diagnosis and Fault-Tolerant Control of Uncertain Robot Manipulators Using High-Order Sliding Mode. *Mathematical Problems in Engineering*. 2016:7926280.

36. Smaeilzadeh SM, Golestani M. Finite-time fault-tolerant adaptive robust control for a class of uncertain non-linear systems with saturation constraints using integral back stepping approach. *IET Control Theory & Applications*. 2018;12(15):2109–2117.

37. Rigatos G, Siano P, Wira P, et al. Nonlinear H_∞ Feedback Control for Asynchronous Motors of Electric Trains. *Intelligent Industrial Systems*. 2015;1(2):85–98.

38. Raffo GV, Ortega MG, Rubio FR. Path Tracking of a UAV via an Underactuated H [infinity] Control Strategy. European Journal of Control. 2011;2:194–213.

39. Han C, Guo J, Pechev A. Nonlinear H∞ based underactuated attitude control for small satellites with two reaction wheels. *Acta Astronautica*. 2014;104(1):159–172.

40. Raffo GV, Ortega MG, Rubio FR. An integral predictive/nonlinear H∞ control structure for a quadrotor helicopter. *Automatica*. 2010;46(1):29–39.

41. Rigatos G, Siano P, Abbaszadeh M. Nonlinear H_∞ control for 4-DOF underactuated overhead cranes. *Transactions of the Institute of Measurement and Control*. 2018;40(7):2364–2377.

42. Rigatos G, Siano P, Raffo G. A nonlinear H_∞ control method for multi-DOF robotic manipulators. *Nonlinear Dynamics*. 2017;88(1):329–348.

43. Ortega MG, Vargas M, Vivas C, et al. Robustness improvement of a nonlinear H∞ controller for robot manipulators via saturation functions. *Journal of Robotic Systems*. 2005;22(8):421–437.

44. Van der Schaft A. L/ Sub 2/-gain analysis of nonlinear systems and nonlinear state feedback H/ sub infinity/control. *Transactions on Automatic Control*. 1992;37(6):770–784.

45. Mazare M, Taghizadeh M, Najafi MR. Contouring control of a 3-[P2(US)] parallel manipulator. *Advanced Robotics*. 2017;31(9):496–508.

46. Van der Schaft A. L2-gain and passivity techniques in nonlinear control. New York: Springer-Verlag; 2000.