energies

Article

Robust Sliding Mode Fuzzy Control of Industrial Robots using an Extended Kalman Filter Inverse Kinematic Solver

Mojtaba Ahmadieh Khanesar [1,*], David Branson [2]

[1] Faculty of Engineering, University of Nottingham, Nottingham NG7 2RD, UK; Mojtaba.ahmadiehkhanesar@nottingham.ac.uk
[2] Faculty of Engineering, University of Nottingham, Nottingham NG7 2RD, UK; David.branson@nottingham.ac.uk
* Correspondence: Mojtaba.ahmadiehkhanesar@nottingham.ac.uk; Tel.: +44 (0) 7918778402

Abstract: This paper presents a sliding mode fuzzy control approach for industrial robots at their static and near static speed (linear velocities less than 5cm/s). The extended Kalman filter with its covariance resetting is used to translate the coordinates from Cartesian to joint angle space. The translated joint angles are then used as a reference signal to control the industrial robot dynamics using a sliding mode fuzzy controller. The stability and robustness of the proposed controller is proven using an appropriate Lyapunov function in the presence of parameter uncertainty and unknown dynamic friction. The proposed controller is simulated on a 6-DOF industrial robot namely the Universal Robot - UR5 considering the maximum allowable joint torques. It is observed that the proposed controller can successfully control UR5 under uncertainties in terms of unknown dynamic friction and parameter uncertainties. The tracking performance of the proposed controller is compared with that of sliding mode control approach. The simulation results demonstrated superior performance of the proposed approach over sliding mode control method in the presence of uncertainties.

Keywords: Industrial robot control; Sliding mode fuzzy control; Inverse kinematic; extended Kalman filter

Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Energies* **2021**, *14*, x. https://doi.org/10.3390/xxxxx

Academic Editor: Firstname Lastname

Received: date
Accepted: date
Published: date

1. Introduction

Industry 4.0 is the fourth generation of industry, which has accommodated advanced machine learning approaches as well as artificial intelligence into the core of manufacturing to improve production [1, 2]. This has resulted in more intelligent factory floors that can respond to customer needs through ever more highly customizable products [3, 4]. Robots serve as an indispensable part of industrial environments to increase productivity and reduce design to market time through their ability to efficiently improve production in highly repetitive tasks [5]. Precision of the factory elements is a primary issue that must be improved to increase manufacturing performance [6, 7]. Industrial tasks in a factory floor such as object handling [8], and manipulation [9] require high level of precision while maintaining the safe operation. Industrial robots are serial architectures with several joints making up their construction. Their architecture results in high coupling between different link dynamics. Modelling uncertainties include simplifications, deadzone, backlashes [5], and variable load for industrial robots. In particular, operating at or near static conditions can cause dynamic control problems due to changes in motion state. We believe that a sliding mode control approach is a robust control method that can be used to guarantee the tracking behavior of the industrial robots in finite time in the presence of matched uncertainties [10].

Sliding mode control approach is a nonlinear and robust control approach. In this controller, the desired behavior of the robot is defined in terms of a sliding surface [11, 12]. The Lyapunov stability theorem is used to design the control signal to guarantee reaching the sliding surface [13-16]. When the states of the robot reach the sliding surface, high frequency switching law maintains sliding mode for a stable reference signal following [13, 14]. The most prominent features of sliding mode control for industrial robots are robustness to unmodelled dynamic and unknown variable load [17].

To track a desired Cartesian space coordinate, an inverse kinematic solver is required to translate industrial robot motion in a Cartesian space to its desired joint space motion. There exist different approaches to solve the inverse kinematic of industrial robots. Geometric approaches are used to find the inverse kinematic of the robot in a closed form [18]. Geometrical approaches to solve inverse kinematic have already been used for planar hyper-redundant manipulators [18, 19], 7R 6-DOF Robot-Manipulator [20], and hybrid parallel-serial five-axis machine tool [21]. Neural networks [22-24], neuro-fuzzy approaches [25] and deep neural networks [26, 27] can be put in the second category of inverse kinematic solvers. The third category is the one that uses estimation algorithms to solve inverse kinematic. Although the kinematic model of industrial robots is highly nonlinear, they can be linearized using Taylor expansions. Estimation algorithms such as Levenberg-Marquardt [28], Least squares [29, 30], recursive least square and extended Kalman filter [30, 31] may also be utilized to find the corresponding joint angles of robots. The second and third categories of inverse kinematic solvers can be identified as machine learning approaches. Among listed estimation algorithms, extended Kalman filter is used in [32, 33] for accurate inverse kinematic and calibration of industrial robots. This method has been previously outperformed Least squares for solving the inverse kinematic problem of Kawasaki RS10N industrial robot and a 4-DOF laboratory setup in terms of accuracy as well as number of iterations [32, 33].

Fuzzy systems are effective approaches to deal with uncertainty in control systems. They can effectively approximate any smooth nonlinear function provided enough rules are used in their structure. Fuzzy systems have already been used to enhance the performance of classical control approaches such as model reference control method and sliding mode controllers [34]. The combination of fuzzy systems and sliding mode approaches result in a robust adaptive control approach which benefits from the rigorous mathematical backbone of sliding mode control theory and the adaptation capabilities of fuzzy systems. Such approaches have already been used to control induction servomotors, anti-lock braking systems [35, 36], and active suspension systems for vehicles [37, 38].

In this paper, a sliding mode fuzzy control (SMFC) approach is used to control an industrial robot in its low-speed motion (linear velocity less than 5cm/s). The reference signal is given in terms of Cartesian position and orientation, which is then translated to the joint space using the inverse kinematic approach. The inverse kinematic approach used in this paper uses extended Kalman filter for joint angle estimation. This approach has been shown to find the desired robot joint angles in a smaller number of iterations compared to least square method [32]. In this paper, the sample time considered for the inverse kinematic solver and the controller are both equal to $1\ msec$. Different from the sliding mode controller, which have already designed for UR5 in [39], the control signal designed in this paper does not include angular acceleration which makes it easier to implement. The desired behavior of the industrial robot is defined in terms of a sliding mode, and the parameter update rules for the fuzzy system in the SMFC are derived from Lyapunov stability theorem. The robustness of the controller in the presence of unknown friction torques and parameter uncertainty are analyzed using the Lyapunov function. Simulations are performed under MATLAB/Simulink® software. The Cartesian space reference signal is given with slow changes to ensure results are tracked at near static speeds (up to 5 cm/s). The results demonstrate the implement-ability showing the satisfactory performance of the overall controller in the presence of unknown friction and parameter uncertainties in the robot dynamics. Comparison is made to a previous sliding mode control approach used to control UR5 [39]. The comparison results shows smaller tracking error value as compared to the sliding mode control approach used in [39] to control UR5. In summary the contributions of the proposed approach over articles in literature are:

- Use of extended Kalman filter as inverse kinematic solver along with SMFC to control industrial robots
- Elimination of the necessity to include second order time derivative of joint angles in the control law
- Superior performance over sliding mode controller for UR5 presented in [39].

This rest of this paper is organized as follows. The dynamic model of the industrial robot and its kinematics are given in Section 2. The overall control architecture is proposed in Section 3 where the overall control architecture, adaptation laws and the stability analysis of the system are studied. The simulation results are given in Section 4, and concluding remarks are presented in Section 5.

2. Industrial Robot Dynamic and Kinematics

2.1 Industrial Robots Dyanmics

The general dynamic of a rigid link 6-DOF industrial robot can be formulated in terms of ordinary differential equations composed of interacting forces on the robot as follows [40, 41].

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) = \tau, \tag{1}$$

where $q \in \mathbb{R}^{6\times1}$ is the robot joint angle vector, $M(q) \in \mathbb{R}^{6\times6}$ represents the inertial forces in the industrial robot, $C(q,\dot{q}) \in \mathbb{R}^{6\times6}$ is the Coriolis and centrifugal forces, $G(q) \in \mathbb{R}^{6\times1}$ presents the robot gravitational force terms, $F(\dot{q}) \in \mathbb{R}^{6\times1}$ is the dynamic friction terms of the robot and $\tau \in \mathbb{R}^{6\times1}$ presents the input torque vector to control the robot. The matrix $M(q)$ satisfies the following equation for the kinetic energy of the robot.

$$K = \frac{1}{2}\dot{q}^T M(q)\dot{q} \tag{2}$$

The elements of the vector $G(q)$, the gravitational forces of the robot, are partial derivatives of potential energy with respect to the corresponding robot joint angles.

$$g_k = \frac{\partial P}{\partial q_k}, k = 1,\ldots,6 \tag{3}$$

where P is the potential energy of the robot and g_k's are the elements of the $G(q)$ vector such that: $G(q) = [g_1 \quad \cdots \quad g_6]^T$. Furthermore, the elements of the matrix $C(q,\dot{q})$, Coriolis and centrifugal force matrix, satisfy the following equation:

$$c_{kj} = \sum_{i=1}^{n} c_{ijk}(q)\dot{q}_i \tag{4}$$

where $c_{ijk}(q)$'s are the Christoffel symbols [41]:

$$c_{ijk}(q) = \frac{1}{2}\left\{\frac{\partial d_{kj}}{\partial q_i} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k}\right\} \tag{5}$$

Finally, $F(\dot{q})$ represent the dynamic friction terms of the robot [42].

2.2 Inverse Kinematic Calculation for Industrial Robots

Forward kinematic is a function which finds the Cartesian coordinates of robot within 3D space as a function of its joint angles. Inverse kinematic is the reverse procedure to assign appropriate joint angles to industrial robots to maintain the desired position and orientation. There exist different solutions for the inverse kinematic for industrial robots at a given pose/alignment of the end-effector and many solutions to determine optimal results. Among them, a recent algorithm is investigated in [32] that uses an extended Kalman filter approach to estimate the joint angles of industrial robot. This approach contributes to higher degree of precision as compared to least squares approaches to estimate the joint angles of industrial robot. Because of the high precision of this algorithm, we selected it for use in this paper. The link transformation matrix from the link $i\text{-}1$ to the link i using the Denavit–Hartenberg (D-H) parameters of the robot depends on joint angles, is as follows [43].

$$
{}^0_1T = \begin{bmatrix} cos\theta_1 & 0 & sin\theta_1 & 0 \\ sin\theta_1 & 0 & -cos\theta_1 & 0 \\ 0 & 0 & 0 & 0.08916 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ {}^1_2T = \begin{bmatrix} cos\theta_2 & -sin\theta_2 & 0 & -0.425cos\theta_2 \\ sin\theta_2 & cos\theta_2 & 0 & -0.425sin\theta_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$
$$
{}^2_3T = \begin{bmatrix} cos\theta_3 & -sin\theta_3 & 0 & -0.392cos\theta_3 \\ sin\theta_3 & cos\theta_3 & 0 & -0.392sin\theta_3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ {}^3_4T = \begin{bmatrix} cos\theta_4 & 0 & sin\theta_4 & 0 \\ sin\theta_4 & 0 & -cos\theta_4 & 0 \\ 0 & 1 & 0 & 0.1092 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)
$$
$$
{}^4_5T = \begin{bmatrix} cos\theta_5 & 0 & -sin\theta_5 & 0 \\ sin\theta_5 & 0 & cos\theta_5 & 0 \\ 0 & -1 & 0 & 0.0947 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ {}^5_6T = \begin{bmatrix} cos\theta_6 & -sin\theta_6 & 0 & 0 \\ sin\theta_6 & cos\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0.0823 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

The industrial robot end effector position and orientation in the fixed world coordinate attached to the base of industrial robot is calculated as the multiplication of all link transformation matrices.

$$T_e = {}^0_6T = {}^0_1T\,{}^1_2T\,{}^2_3T\,{}^3_4T\,{}^4_5T\,{}^5_6T \quad (7)$$

The overall end effector position and orientation error can be approximated by a linear superpositions of the joint angle deviations weighted by their sensitivity function.

$$\Delta T_e = \frac{\partial T_e}{\partial \theta_1}\Delta\theta_1 + \ldots + \frac{\partial T_e}{\partial \theta_6}\Delta\theta_6 + H.O.T. \quad (8)$$

The parameter $\frac{\partial T_e}{\partial \theta_i}$ is the sensitivity function of the end effector transformation matrix with respect to i-th joint angle of the robot and can be calculated analytically.

Remark. It is noted that in the industrial robot investigated in this paper, the only changeable D-H parameter is the joint angles of the robot θ_i's. However, in a more general case where other D-H parameters of the robot are changeable. The matrix ΔT_e is approximated as a linear superposition of all deviations in all D-H parameters weighted by their corresponding sensitivity matrix.

According to [32], the extended Kalman filter estimation method results in a more precise positioning of the robot than least squares solution. The EKF estimation iteratively estimate the joint angles of the robot as follows [36].

$$
\begin{aligned}
\theta^{k+1} &= \theta^k + K^k[T^d - h^k(\theta^k)] \\
P^k &= P^k H^k (R + H^{kT} P^k H^k)^{-1} \\
P^{k+1} &= P^k - P^k H^{kT} P^k + Q^k
\end{aligned} \quad (9)
$$

where $\theta^k = [\theta_1^k \ \ \ldots \ \ \theta_6^k]^T$ is the vector of the unknown joint angles of the robot at k-th iteration, $P^k \in \mathbb{R}^{6\times6}$ is the parameter covariance matrix at k-th iteration, $R \in \mathbb{R}^{12}$ is the measurement noise covariance matrix and $Q^k \in \mathbb{R}^{6\times6}$ is the process noise covariance matrix. H^k is defined as follows.

$$H^k = \begin{bmatrix} \frac{\partial h}{\partial \theta_1^k} & \cdots & \frac{\partial h}{\partial \theta_6^k} \end{bmatrix}^T \quad (10)$$

and the function h is a vector function of θ^k defined as follows.

$$h^k = [T_{e11}^k \ \ \ldots \ \ T_{e14}^k \ \ \ldots \ \ T_{e31}^k \ \ \ldots \ \ T_{e34}^k]^T \quad (11)$$

where T_e is obtained from (7). The function T^d is the desired position and orientation of the robot in a vector form as follows.

$$T_d = [T_{de11} \quad \cdots \quad T_{de14} \quad \cdots \quad T_{de31} \quad \cdots \quad T_{de34}]^T \tag{12}$$

where T_{de} include the desired position and orientation of the robot.

3. Control Architecture

3.1 Overall Control Structure

The overall control structure is denoted in Figure 1. The desired position of the robot end effector is denoted by $[x_d \quad y_d \quad z_d]^T$ and its desired orientation is represented by $[\alpha_d \quad \beta_d \quad \gamma_d]^T$. Using the inverse kinematic of the robot as explained in Section 2.2, it is possible to find the desired joint angles of the robot $[q_{1d} \quad \cdots \quad q_{6d}]^T$. In the next stage, the sliding mode fuzzy controller is responsible for controlling the robot and pushing the joint angles of the robot towards their desired value. Joint angles of the robot are measured using the joint encoders on the shafts of the robot. The SMFC approach and its stability analysis is explained in the next subsection. The proposed SMFC is responsible to use measured joint angles and their desired values to generate the torque values to control UR5. The Cartesian space coordinate of the robot is calculated using its forward kinematic in (6). The position and orientation error of the robot is calculated as the difference between the desired coordinate of the robot and its real coordinate as the output of its forward kinematics.

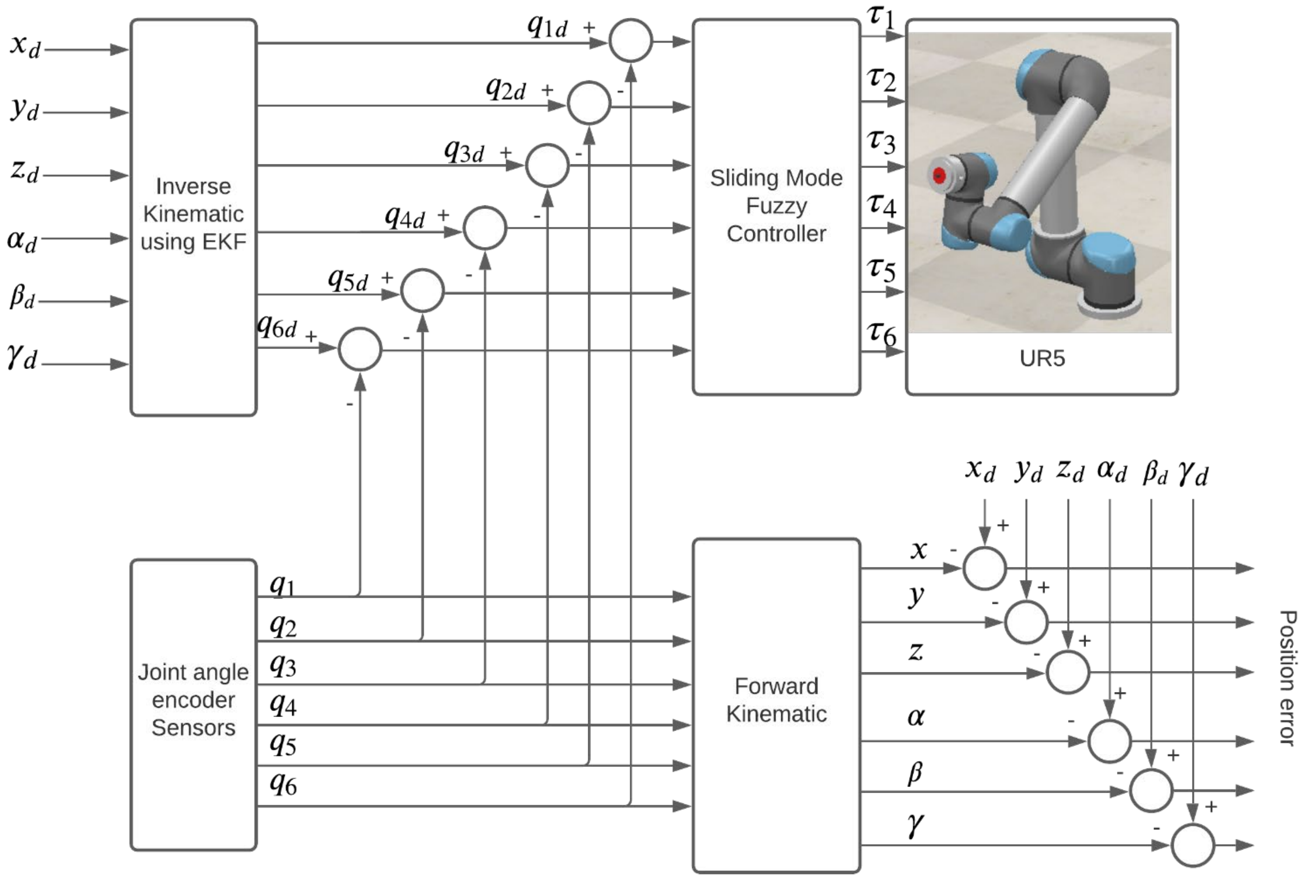

Figure 1 Overall block diagram of the proposed controller

3.2 Sliding Mode Fuzzy Controller

3.2.1. Fuzzy system structure

Two fuzzy systems are considered in the control structure of each joint. The j-th fuzzy IF-THEN rule of the first controller to control *i*-th joint is in the following form:

$$j\text{-th rule: IF } e_i \text{ is } A_{1ij} \text{ and } \dot{e}_i \text{ is } B_{1ij} \text{ Then } \tau_{f_1 ij} \text{ is } \psi_{ij} sgn(s_i) \tag{13}$$

where $\tau_{f_1 i}$ is an additive part of the control signal for *i*-th joint. The mathematical formulation of this fuzzy system output is as follows.

$$\tau_{f_1 ij} = \sum_{j=1}^{M} h_{ij} \psi_{ij} sgn(s_i) \tag{14}$$

where ψ_{ij}'s are the consequent part parameters of the fuzzy system, h_{ij}'s are the weights associated with the j-th fuzzy rule which can be calculated as follows.

$$h_{ij} = \frac{\mu_{ij}(e)\mu_{ij}(\dot{e})}{\sum_{j=1}^{M} \mu_{ij}(e)\mu_{ij}(\dot{e})}, j = 1, \ldots, M \tag{15}$$

and $\mu_{ij}(.)$'s are the membership functions considered for the inputs of the fuzzy system e and $\dot{e}$. The fuzzy IF-THEN rules of the second fuzzy system are in the following form:

$$j\text{-th rule: IF } e_i \text{ is } A_{2ij} \text{ and } \dot{e}_i \text{ is } B_{2ij} \text{ Then } \tau_{f_2 ij} \text{ is } \beta_{ij} s_i \tag{16}$$

The mathematical formulation of the fuzzy system is as follows.

$$\tau_{f_2 i} = \sum_{j=1}^{M} w_{ij} \beta_{ij}\, s_i \tag{17}$$

where β_{ij}'s are the consequent part parameters of the fuzzy system, w_{ij}'s are the weights associated with the *j*th fuzzy rule which can be calculated as follows.

$$w_{ij} = \frac{\pi_{ij}(e)\pi_{ij}(\dot{e})}{\sum_{j=1}^{M} \pi_{ij}(e)\pi_{ij}(\dot{e})}, j = 1, \ldots, M \tag{18}$$

and $\pi_{ij}(.)$'s are the membership functions considered for the inputs of the fuzzy system.

3.2.2 Sliding Mode Fuzzy Controller

The joint space controller of the robot is a robust sliding mode fuzzy controller. The general dynamic model of industrial robots is as follows [44, 45].

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) = \tau \tag{19}$$

SMFC is a robust control approach which can control wide range of robots in the presence of unmodelled dynamic and variable load for the robot [46]. The tracking error for the industrial robot is defined as follows.

$$e = q_d - q \tag{20}$$

where $q_d \in \mathbb{R}^6$ is the desired joint angles of the robot and $e \in \mathbb{R}^6$ is the tracking error for industrial robot. The sliding surface defines the desired trajectory of the robot and needs to be stable. In the case of 6-DOF industrial robot, it is defined in a vector form.

$$s = \dot{e} + \lambda e, \qquad \lambda \geq 0 \tag{21}$$

where $s \in \mathbb{R}^6$ and λ defines the decay ratio for the tracking error in the sliding mode. The time derivative of the sliding surface is obtained as follows.

$$\dot{s} = \ddot{e} + \lambda\dot{e} = \ddot{q}_d - \ddot{q} + \lambda(\dot{q}_d - \dot{q}) \quad (22)$$

Applying the robot dynamic as in (19) in (22) the following equation is obtained. 207

$$\dot{s} = \ddot{q}_d + M^{-1}(C(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) - \tau) + \lambda(\dot{q}_d - \dot{q}) \quad (23)$$

To analyze the stability of the overall control system, the following Lyapunov function is considered. 208 209

$$V = \frac{1}{2}s^T M s + \sum_i \frac{1}{2\gamma_i}\tilde{\theta}_i^T\tilde{\theta}_i + \sum_i \frac{1}{2\rho_i}\tilde{\psi}_i^T\tilde{\psi}_i \quad (24)$$

where the parameters $\tilde{\theta}_i$ and $\tilde{\psi}_i$ defined as follows. 210

$$\tilde{\theta}_i = \theta_i - \theta_i^*$$

$$\tilde{\psi}_i = \psi_i - \psi_i^* \quad (25)$$

and the parameters γ_i's and ρ_i's are positive parameters. 211

$$\dot{V} = \frac{1}{2}s^T\dot{M}(q)s + s^T M(q)\dot{s} + \sum_i \frac{1}{2\gamma_i}\tilde{\theta}_i^T\dot{\theta}_i + \sum_i \frac{1}{2\rho_i}\tilde{\psi}_i^T\dot{\psi}_i \quad (26)$$

The input control torque is defined as follows. 212

$$\tau = M_n(q)\ddot{q}_d + C_n(q,\dot{q})\dot{q} + G_n(q) + [\tau_{f_1 1} \quad \cdots \quad \tau_{f_1 6}]^T + [\tau_{f_2 1} \quad \cdots \quad \tau_{f_2 6}]^T \quad (27)$$

where $M_n(q)$, $C_n(q,\dot{q})$, and $G_n(q)$ are the nominal and known parts of $M(q)$, $C(q,\dot{q})$, and $G(q)$ which satisfy: 213 214

$$\Delta M(q) = M(q) - M_n(q),$$
$$\Delta C(q,\dot{q}) = C(q,\dot{q}) - C_n(q,\dot{q}), \quad (28)$$
$$\Delta G(q) = G(q) - G_n(q),$$

and $\Delta M(q)$, $\Delta C(q,\dot{q})$, and $\Delta G(q)$ are the unknown uncertainties in the dynamic model of the industrial robot. Considering the control signals as in (27), (14) and (17), we have the following equation for industrial robot control signal. 215 216 217

$$\tau = M_n\ddot{q}_d + C_n(q,\dot{q})\dot{q} + G_n(q)$$

$$+[h_1^T\psi_1 sgn(s_1) \quad \ldots \quad h_6^T\psi_6 sgn(s_6)]^T$$

$$+[w_1^T\theta_1 s_1 \quad \ldots \quad w_6^T\theta_6 s_6]^T \quad (29)$$

Remark. To implement the control signal in (29), the desired joint angles of the system q_d need to be a sufficiently smooth function of time so that $\ddot{q}_d$ exists. 218 219

Considering the dynamic equation of industrial robot in (19), the time derivative of Lyapunov function can be rewritten as follows. 220 221

$$\dot{V} = \frac{1}{2}s^T\dot{M}(q)s + s^T M(q)\big(\ddot{q}_d + M^{-1}(q)(C(q,\dot{q})\dot{q} + G(q) - \tau) + \lambda(\dot{q}_d - \dot{q})\big)$$

$$+\sum_i \frac{1}{2\gamma_i}\tilde{\theta}_i^T\dot{\theta}_i + \sum_i \frac{1}{2\rho_i}\tilde{\psi}_i^T\dot{\psi}_i \quad (30)$$

By inserting the torque control signal of (29) to the time derivative equation of the Lyapunov function in (30), the following equation is obtained. 222 223

$$\dot{V} = \frac{1}{2}s^T\dot{M}(q)\mathrm{s} + s^T(\Delta M\ddot{q}_d + \Delta C(q,\dot{q}) + \Delta G(q) + F(\dot{q})) - s^T\tau_r -$$

$$s^T[h_1^T\psi_1 sgn(s_1) \quad \dots \quad h_6^T\psi_6 sgn(s_6)]^T -$$

$$s^T[w_1^T\theta_1 s_1 \quad \dots \quad w_6^T\theta_6 s_6]^T + \sum_i \frac{1}{2\gamma_i}\tilde{\theta}_i^T\dot{\theta}_i + \sum_i \frac{1}{2\rho_i}\tilde{\psi}_i^T\dot{\psi}_i \tag{31}$$

By adding and subtracting the term $\sum_i h_i^T\theta_i^* s_i^2 + \sum_i h_i^T\psi_i^*|s_i|$ to the time derivative of the Lyapunov function in (31), the following equation is obtained. 224 225

$$\dot{V} = \frac{1}{2}s^T\dot{M}(q)s + s^T\left(\Delta M\ddot{q}_d + \Delta C(q,\dot{q}) + \Delta G(q) + F(\dot{q})\right)$$
$$-s^T\tau_r - s^T[h_1^T\tilde{\psi}_1 sgn(s_1) \quad \dots \quad h_6^T\tilde{\psi}_6 sgn(s_6)]^T$$
$$-s^T[w_1^T\tilde{\theta}_1 s_1 \quad \dots \quad w_6^T\theta_6 s_6]^T + \sum_i \frac{1}{\gamma_i}\tilde{\theta}_i^T\dot{\theta}_i$$
$$+\sum_i \frac{1}{\rho_i}\tilde{\psi}_i^T\dot{\psi}_i - \sum_i h_i^T\theta_i^* s_i^2 - \sum_i h_i^T\psi_i^*|s_i| \tag{32}$$

The adaptation laws for the parameters ψ_i's and θ_i's are taken as follows. 226

$$\dot{\psi}_i = \rho_i h_i|s_i|, i = 1,\dots,6 \tag{33}$$

$$\dot{\theta}_i = \gamma_i w_i s_i^2, i = 1,\dots,6 \tag{34}$$

This results in the following equation for the time derivative of the Lyapunov function V. 227 228

$$\dot{V} = \frac{1}{2}s^T\dot{M}(\mathrm{q})\mathrm{s} + s^T\left(\Delta M\ddot{q}_d + \Delta C(q,\dot{q}) + \Delta G(q) + F(\dot{q})\right) - \sum_i h_i^T\psi_i^* s_i^2$$
$$- \sum_i h_i^T\theta_i^*|s_i| \tag{35}$$

Using the upper bound for uncertainty as $\|\Delta M\ddot{q}_d + \Delta C(q,\dot{q})\dot{q} + \Delta G(q) + F(\dot{q})\| \le N$, and considering the fact that $\|s\| \le \sum_i |s_i|$, the time derivative of the Lyapunov function in (37) can be further manipulated as: 229 230 231 232

$$\dot{V} \le \frac{1}{2}s^T\dot{M}(\mathrm{q})\mathrm{s} + N\sum_i |s_i| - \sum_i h_i^T\psi_i^* s_i^2 - \sum_i h_i^T\theta_i^*|s_i| \tag{36}$$

233

Considering the following nonequality for the norm of time derivative of $M(q)$: 234

$$\|\dot{M}(q)\| \le \lambda_{max}(\dot{M}(q)) \tag{37}$$

the time derivative of the Lyapunov function is obtained as follows. 235

$$\dot{V} = \frac{1}{2}\lambda_{max}\left(\dot{M}(q)\right)\|s\|^2 + N\sum_i |s_i| - \sum_i h_i^T\psi_i^* s_i^2 - \sum_i h_i^T\theta_i^*|s_i| \tag{38}$$

Let $min(\psi_i^*)$ and $min(h_i)$ be the minimum values of the elements of vectors ψ_i^* and h_i, respectively. Provided that: 236 237

$$\frac{\lambda_{max}\left(\dot{M}(q)\right)}{2min(h_i)} \le min(\psi_i^*) \tag{39}$$

we have the following equation for $\dot{V}$:

$$\dot{V} \le N \sum_i |s_i| - \sum_i h_i^T \psi_i^* |s_i| \tag{40}$$

It is further possible to consider the minimum element value of ψ_i^* such that:

$$\frac{\eta_1}{min(h_i)} \le min(\psi_i^*) \tag{41}$$

This results in the following inequality for the time derivative of the error.

$$\dot{V} \le N \sum_i |s_i| - \sum_i \eta_1 |s_i| \tag{42}$$

which further guarantees the finite time convergence of the sliding surfaces to zero if it is further modified as:

$$\dot{V} \le -\sum_i \eta |s_i| \tag{43}$$

where the parameter η is chosen as $\eta = \eta_1 - N$. This finite time can be calculated as follows.

$$t_{max} \le \frac{1}{\eta} \sum_i |s_i(0)| \tag{44}$$

Remark. To avoid the drift in the parameters of the fuzzy system, it is required to use the δ-modification rule for the adaptation laws for the parameters of the fuzzy system [47, 48].

$$\dot{\psi}_i = \rho_i h_i |s_i| - \rho_i \delta \psi_i, i = 1, \dots, 6$$

$$\dot{\theta}_i = \gamma_i w_i s_i^2 - \gamma_i \delta \theta_i, i = 1, \dots, 6 \tag{45}$$

The δ-modification avoids too high parameter values for adaptable parameters of ψ_i's and θ_i's by pushing these parameters towards zero. However, it is required to choose a small value for δ to avoid disturbing the adaptation law.

4. Simulation Results

The proposed controller as demonstrated in Figure 1 is simulated on a UR5 robot model. The model of UR5 and its parameter values are given in https://github.com/kkufieta/force_estimate_ur5/. The URL of the file associated with $M(q), C(q,\dot{q})\dot{q}, G(q)$, *and* $T(q)$ under this repository are presented in Table 1.

Table 1 URL of the github repository files containing $M(q), C(q,\dot{q})\dot{q}, G(q)$, *and* $T(q)$ *under* https://github.com/kkufieta/force_estimate_ur5/

Parameter	Value
$M(q)$	JointSpaceControl_Model_C_ForceEstim/D_Matrix.m
$C(q,\dot{q})\dot{q}$	JointSpaceControl_Model_C_ForceEstim/C_Matrix.m
$G(q)$	JointSpaceControl_Model_C_ForceEstim/G_Vector.m
$T(q)$	ur5_modeling_force_estimate/Derive_Dyn_Equations_Model_C/get_rotation_matrices.m

The parameters of dynamic friction in this robot are obtained from [42].The sliding surface dynamics taken for the sliding mode controller is as follows.

$$s = \dot{e} + 50e \tag{46}$$

As soon as the states of the robot converge to this sliding surface, the sliding mode controller guarantees the exponential convergence of the error to zero. The sample time considered for the sliding mode fuzzy controller is 1 *msec* and the same sample time is taken for the inverse kinematic solver of the robot. However, 200 epochs are required at each step for the inverse kinematic solver to find the desired joint angles. Simulations are performed in MATLAB/Simulink® environment. Other parameters taken for the adaptation laws and the covariance matrix are listed in Table 2 Values of adaptation parameters. These parameters are design parameters which are selected after a trial and error.

Table 2 Values of adaptation parameters

Parameter	Value
$\boldsymbol{\rho_1, \rho_2, \rho_3, \rho_4, \rho_5, \rho_6}$	0.1
$\boldsymbol{\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6}$	0.1
$\boldsymbol{\delta}$	0.01
$\boldsymbol{\lambda}$	50
$\boldsymbol{P_0}$	I_6

The simulations are performed in the presence of unknown dynamic friction for the robot. The reference signal is a ramp signal for the position in x and y dimensions and zero for z dimension and robot orientations.

$$\text{Desired trajectory \#1} = \begin{cases} \mathrm{x_d} = 0.37 + 0.05\mathrm{t} \\ \mathrm{y_d} = 0.37 - 0.05\mathrm{t} \\ \mathrm{z_d} = 0 \\ \alpha_\mathrm{d} = 0 \\ \beta_\mathrm{d} = 0 \\ \gamma_\mathrm{d} = 0 \end{cases} \tag{47}$$

The SMFC is responsible for controlling the robot using the reference joint angles of the robot (see Figure 1). While the maximum applicable torque for the first three joints of UR5 are equal to *150 N.m.*, it is equal to *28 N.m.* for the next three joints. These limitations are due to the motors used to control UR5 by the manufacturer. Maximum 10% uncertainty is added to each element of $M(q), C(q,\dot{q})\dot{q}$, and $G(q)$ used in the controller structure to test its robustness. The time respond of the industrial robot in terms of real joint angles versus time are presented in Figure 2. The parameters $q_1, \ldots, q_6$ refer to the joint angles #1-#6 of UR5. As can be seen from the Figure, the controller is performing well, and joint angle position errors are very small. The 3D orientation tracking of UR5 end-effector is illustrated in Figure 3, as can be seen from the figure, the steady state errors of the industrial robot for $\alpha, \beta,$ and γ coordinates are small.

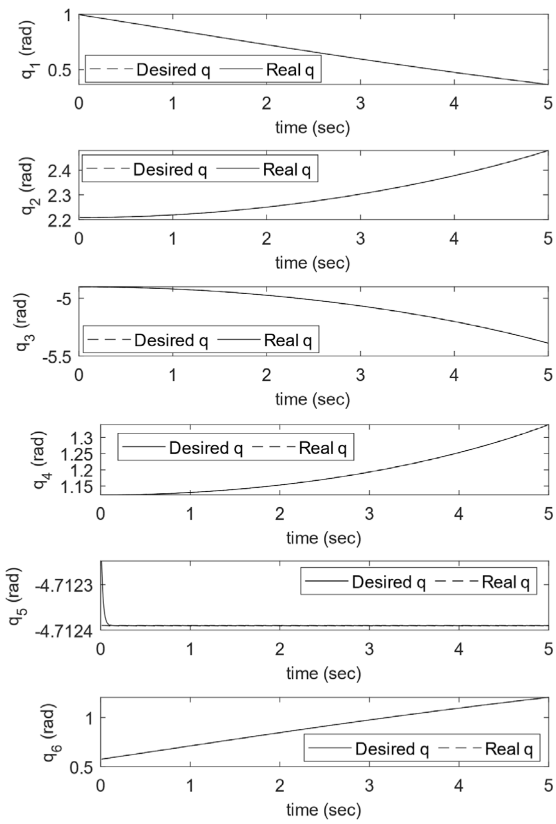

Figure 2 Joint angle behavior of UR5 using the proposed control approach

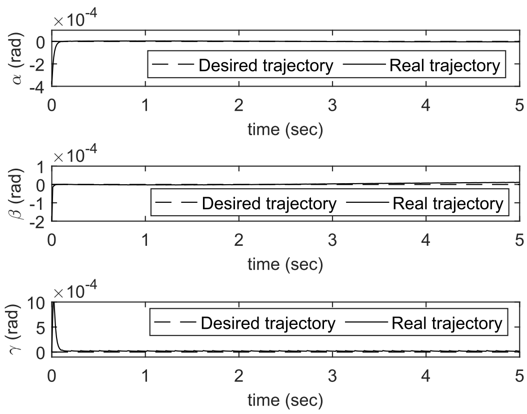

Figure 3 3D orientation behavior of UR5 in the presence of the proposed controller

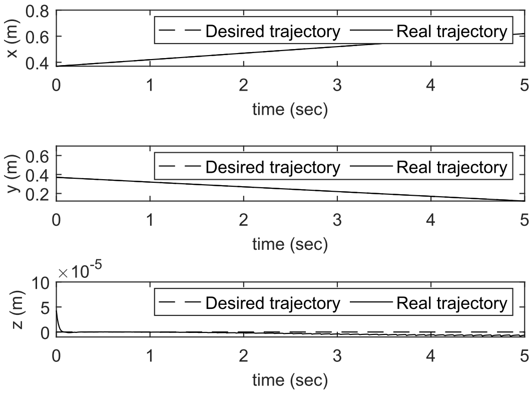

Figure 4 3D position behavior of UR5 in the presence of the proposed controller

The position tracking of the UR5 is illustrated in Figure 4 which demonstrate very small tracking error in the UR5 end-effector position.

To have a comparison with a state-of-the-art control algorithm for UR5, the proposed approach is compared with sliding mode control approach which is previously applied to a UR5 in [39]. The trajectory given for the robot is the same as the one given in [39] as follows.

$$Desired\ trajectory\ \#2 = \begin{cases} x_d = 0.37 + cos(1.26t) \\ y_d = 0.37 + sin(1.26t) \\ z_d = 0.05t \\ \alpha_d = 0.01 \\ \beta_d = -1 \\ \gamma_d = -0.01 \end{cases} \tag{48}$$

The same controller parameters as in Table 1 are used to track this trajectory. The same maximum 10% uncertainty is added to the elements of $M(q), C(q,\dot{q})\dot{q}$, and $G(q)$ used in the controller structure to test its robustness. To find the position error, the overall position error is calculated as the average value of the normed error at each individual sample using the following equation.

$$\|E_p\| = \frac{1}{N}\sum_{i=1}^{N}\|e_p^i\|, \tag{49}$$

where e_p^i is the individual sample error for position, and N is the total number of the samples. To find the orientation error the orientation error is calculated as follows.

$$\|E_o\| = \frac{1}{N}\sum_{i=1}^{N}\|e_o^i - \langle 1, \mathbf{0}\rangle\| \tag{50}$$

where e_o^i, the error for each individual sample, is calculated as $e_o^i = x_{do}^i \otimes (x_{ro}^i)^{-1}$; and $\otimes$ and $(.)^{-1}$ represent quaternion product and quaternion conjugate, respectively. Moreover, x_{do}^i and x_{ro}^i represent the desired quaternion and real quaternion orientation of the object. Using the indexes as introduced in (49) and (50), a comparison can be made between the proposed approach and the approach previously introduced in [39]. As presented in Table 2, while $\|E_p\|$ is equal to 8.255e-5m in the case of the proposed approach in this paper, it is equal to *0.0068m* in the sliding mode control approach presented in [39]. This means that the proposed controller in this paper is capable of controlling the UR5 considerably lower error value. Moreover, while the $\|E_o\|$ is equal to 2.086e-4*rad* in this paper, it is equal to *0.0035rad* in the sliding mode control approach investigated in [39]. Hence orientational error is considerably lower for the proposed approach.

Table 3 tracking position and orientation errors in the case of second trajectory

Method	$\|E_p\|$ (m)	$\|E_o\|$ (rad)
PD controller [39]	0.0104m	0.0049
SMC method [39]	0.0068m	0.0035
The proposed Approach	8.255e-5m	2.086e-4

The trajectory tracking of the proposed controller in terms of joint angles are presented in Figure 5 and Figure 6. As can be seen from the figure, the initial conditions of UR5 are selected on the desired trajectory and applying the control law of (29) resulted in the real trajectory for robot joint angles qs to be very close to the desired robot joint angles q_{d}s. Moreover, the trajectory tracking performance of the proposed approach in terms of position and orientation are presented in Figure 7 and Figure 8, respectively. The result shows that the desired joint angles are followed with high performance.

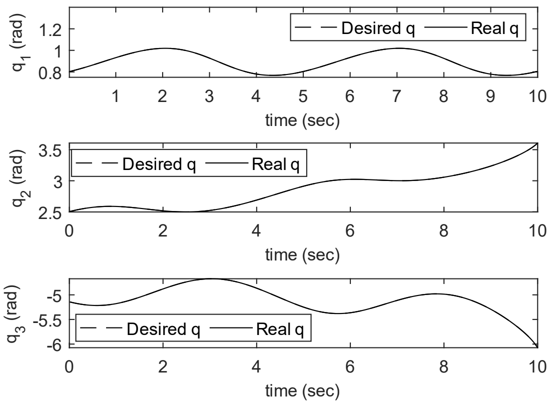

Figure 5 trajectory tracking behaviour of the robot using the proposed approach for the first three joints

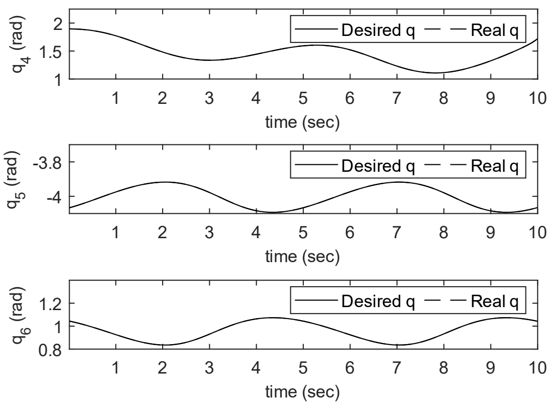

Figure 6 trajectory tracking behaviour of the robot using the proposed approach for the last three joints

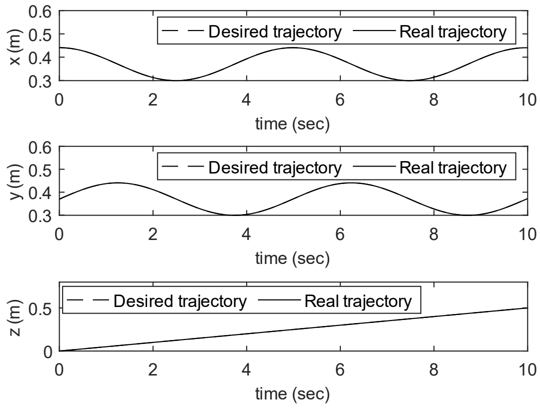

Figure 7 3D position behavior of UR5 in the presence of the proposed controller

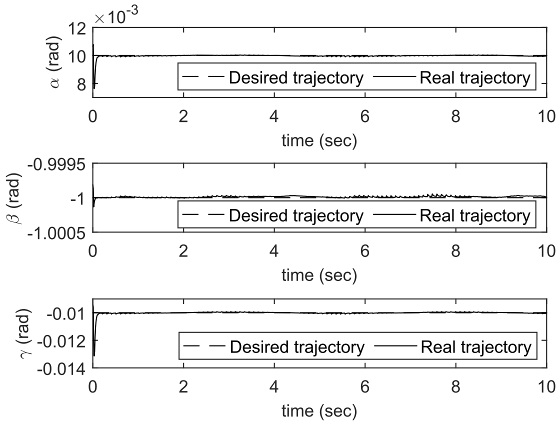

Figure 8 3D orientation behavior of UR5 in the presence of the proposed controller

Figure 9 demonstrates the trajectory tracking behavior of UR5 in xy-plane. For a better comparison between the real trajectory of the robot and its desired values, a zoomed in version of the graph is presented. This figure shows that the proposed controller is capable of tracking the trajectory defined in (48) with high performance in the presence of dynamic friction and parameter uncertainty. The numerical comparison between the desired Cartesian coordinate and real coordinate of the robot are given in Table 3.

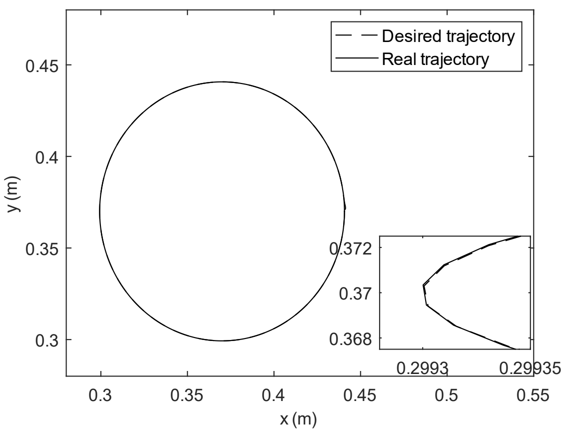

Figure 9 3D position behavior of UR5 using the proposed controller in x-y plane

5. Conclusions

The prominent requirements of advanced manufacturing environment for precision control of industrial robots motivates the usage of advanced control techniques. Sliding mode controllers are particularly proven to be an ideal control approach to deal with modeling uncertainties. To control an industrial robot in a Cartesian space, it is required to translate 3D Cartesian coordinates to desired joint angles values. The EKF inverse kinematic solver is a successful algorithm to deal with inverse kinematic problems employed in this paper. This inverse kinematic solver has been previously compared in [29] with recursive least square method and already demonstrated its superior performance. Covariance resetting as a robust modification to the original version of EKF is employed to avoid covariance bursting in its structure. The SMFC approach is responsible for dynamic stabilization and tracking control of the industrial robot. Lyapunov stability theory is used to analyze the stability of the proposed control structure as well as the adaptation laws for the fuzzy system parameters rigorously. Robustness of the controller in the presence of dynamic friction, which inherently exists in the UR5 at low speeds, is analyzed using the same Lyapunov function. The proposed controller is compared with a previous controller investigated in [39] on a UR5 model. It is shown that the proposed controller in this paper outperforms the controller designed in [39] in terms of positional and orientation accuracies. During these simulations, the robustness of the proposed controller against uncertainty in terms of dynamic friction and parameter uncertainties in UR5 structure is observed. It is shown that the proposed controller is capable of controlling the system with high performance in the presence of uncertainties in robot model.

6. Future Works

In future the proposed controller will be implemented on a real robot with an advanced metrology system (WFSI, photogrammetry, IMU and position fused sensor system) to measure and control real-time position of the robot. Comparison of performance to further improved controllers, including ones that use non-quadratic Lyapunov functions, will be considered.

7. Acknowledgement

This work is funded and supported by the Engineering and Physical Sciences Research Council (EPSRC) under grant number: EP/T023805/1—High-accuracy robotic system for precise object manipulation (HARISOM).

References

[1] T. R. Wanasinghe, R. G. Gosine, L. A. James, G. K. I. Mann, O. de Silva, and P. J. Warrian, "The Internet of Things in the Oil and Gas Industry: A Systematic Review," *IEEE Internet of Things Journal,* vol. 7, no. 9, pp. 8654-8673, 2020.

[2] S. Garg, K. Kaur, G. Kaddoum, and K.-K. R. Choo, "Toward Secure and Provable Authentication for Internet of Things: Realizing Industry 4.0," *IEEE Internet of Things Journal,* vol. 7, no. 5, pp. 4598-4606, 2020.

[3] P. Burnap *et al.*, "Chatty factories: a vision for the future of product design and manufacture with IoT," *IET Conference Proceedings,* pp. 4 (6 pp.)-4 (6 pp.)[Online]. Available: https://digital-library.theiet.org/content/conferences/10.1049/cp.2019.0129

[4] M. Lakoju, N. Ajienka, M. A. Khanesar, P. Burnap, and D. T. Branson, "Unsupervised Learning for Product Use Activity Recognition: An Exploratory Study of a "Chatty Device"," *Sensors (Basel),* vol. 21, no. 15, Jul 22 2021.

[5] M. Slamani, A. Nubiola, and I. A. Bonev, "Modeling and assessment of the backlash error of an industrial robot," *Robotica,* vol. 30, no. 7, pp. 1167-1175, 2012.

[6] M. L. Hoang, M. Carratu, V. Paciello, and A. Pietrosanto, "A new Orientation Method for Inclinometer based on MEMS Accelerometer used in Industry 4.0," presented at the 2020 IEEE 18th International Conference on Industrial Informatics (INDIN), 2020.

[7] D. Mishra, R. B. Roy, S. Dutta, S. K. Pal, and D. Chakravarty, "A review on sensor based monitoring and control of friction stir welding process and a roadmap to Industry 4.0," *Journal of Manufacturing Processes,* vol. 36, pp. 373-397, 2018.

[8] M. Ahmadieh Khanesar, R. Bansal, G. Martínez-Arellano, and D. T. Branson, "XOR Binary Gravitational Search Algorithm with Repository: Industry 4.0 Applications," *Applied Sciences,* vol. 10, no. 18, 2020.

[9] H. Luo, F. Zhao, S. Guo, C. Yu, G. Liu, and T. Wu, "Mechanical performance research of friction stir welding robot for aerospace applications," *International Journal of Advanced Robotic Systems,* vol. 18, no. 1, 2021.

[10] V. Utkin, J. Guldner, and J. Shi, *Sliding mode control in electro-mechanical systems*. CRC press, 2017.

[11] J. Lin, Y. Zhao, P. Zhang, J. Wang, and H. Su, "Research on Compound Sliding Mode Control of a Permanent Magnet Synchronous Motor in Electromechanical Actuators," *Energies,* vol. 14, no. 21, 2021.

[12] M. Pietrala, P. Leśniewski, and A. Bartoszewicz, "Sliding Mode Control with Minimization of the Regulation Time in the Presence of Control Signal and Velocity Constraints," *Energies,* vol. 14, no. 10, 2021.

[13] H. K. Khalil, *Nonlinear Control*. Pearson, 2015.

[14] J. J. E. Slotine and W. Li, *Applied Nonlinear Control*. Pearson education Taiwan, 2005.

[15] S. Ullah, Q. Khan, A. Mehmood, and A. I. Bhatti, "Robust backstepping sliding mode control design for a class of underactuated electro–mechanical nonlinear systems," *Journal of Electrical Engineering & Technology,* vol. 15, no. 4, pp. 1821-1828, 2020.

[16] S. Ullah, Q. Khan, A. Mehmood, S. A. M. Kirmani, and O. Mechali, "Neuro-adaptive fast integral terminal sliding mode control design with variable gain robust exact differentiator for under-actuated quadcopter UAV," *ISA transactions,* 2021.

[17] S. Ullah, A. Mehmood, Q. Khan, S. Rehman, and J. Iqbal, "Robust integral sliding mode control design for stability enhancement of under-actuated quadcopter," *International Journal of Control, Automation and Systems,* vol. 18, no. 7, pp. 1671-1678, 2020.

[18] Y. Liu, M. Kong, N. Wan, and P. Ben-Tzvi, "A geometric approach to obtain the closed-form forward kinematics of h4 parallel robot," *Journal of Mechanisms and Robotics,* vol. 10, no. 5, p. 051013, 2018.

[19] S. Yahya, M. Moghavvemi, and H. A. Mohamed, "Geometrical approach of planar hyper-redundant manipulators: Inverse kinematics, path planning and workspace," *Simulation Modelling Practice and Theory,* vol. 19, no. 1, pp. 406-422, 2011.

[20] I. Vasilyev and A. Lyashin, "Analytical solution to inverse kinematic problem for 6-DOF robot-manipulator," *Automation and Remote Control,* vol. 71, no. 10, pp. 2195-2199, 2010.

[21] Y.-L. Lai, C.-C. Liao, and Z.-G. Chao, "Inverse kinematics for a novel hybrid parallel–serial five-axis machine tool," *Robotics and Computer-Integrated Manufacturing,* vol. 50, pp. 63-79, 2018.

[22] G. Jiang, M. Luo, K. Bai, and S. Chen, "A precise positioning method for a puncture robot based on a PSO-optimized BP neural network algorithm," *Applied Sciences,* vol. 7, no. 10, p. 969, 2017.

[23] A. A. Hassan, M. El-Habrouk, and S. Deghedie, "Inverse kinematics of redundant manipulators formulated as quadratic programming optimization problem solved using recurrent neural networks: A review," *Robotica,* vol. 38, no. 8, pp. 1495-1512, 2020.

[24] E. Jiménez-López, D. S. de la Mora-Pulido, L. A. Reyes-Ávila, R. S. de la Mora-Pulido, J. Melendez-Campos, and A. A. López-Martínez, "Modeling of Inverse Kinematic of 3-DoF Robot, Using Unit Quaternions and Artificial Neural Network," *Robotica,* vol. 39, no. 7, pp. 1230-1250, 2021.

[25] E. Lazarevska, "A neuro-fuzzy model of the inverse kinematics of a 4 DOF robotic arm," in *2012 UKSim 14th International Conference on Computer Modelling and Simulation*, 2012: IEEE, pp. 306-311.

[26] X. Wang, X. Liu, L. Chen, and H. Hu, "Deep-learning damped least squares method for inverse kinematics of redundant robots," *Measurement,* vol. 171, p. 108821, 2021.

[27] N. A. Mohamed, A. T. Azar, N. E. Abbas, M. A. Ezzeldin, and H. H. Ammar, "Experimental Kinematic Modeling of 6-DOF Serial Manipulator Using Hybrid Deep Learning," in *AICV*, 2020, pp. 283-295.

[28] G. Luo, L. Zou, Z. Wang, C. Lv, J. Ou, and Y. Huang, "A novel kinematic parameters calibration method for industrial robot based on Levenberg-Marquardt and Differential Evolution hybrid algorithm," *Robotics and Computer-Integrated Manufacturing,* vol. 71, p. 102165, 2021.

[29] W. K. Veitschegger and C.-H. Wu, "Robot calibration and compensation," *IEEE Journal on Robotics and Automation,* vol. 4, no. 6, pp. 643-656, 1988.

[30] A. Omodei, G. Legnani, and R. Adamini, "Three methodologies for the calibration of industrial manipulators: Experimental results on a SCARA robot," *Journal of Robotic Systems,* vol. 17, no. 6, pp. 291-307, 2000.

[31] M. Habibullah and D. D.-C. Lu, "A speed-sensorless FS-PTC of induction motors using extended Kalman filters," *IEEE Transactions on Industrial Electronics,* vol. 62, no. 11, pp. 6765-6778, 2015.

[32] Z. Jiang, W. Zhou, H. Li, Y. Mo, W. Ni, and Q. Huang, "A New Kind of Accurate Calibration Method for Robotic Kinematic Parameters Based on the Extended Kalman and Particle Filter Algorithm," *IEEE Transactions on Industrial Electronics,* vol. 65, no. 4, pp. 3337-3345, 2018.

[33] I.-W. Park, B.-J. Lee, S.-H. Cho, Y.-D. Hong, and J.-H. Kim, "Laser-based kinematic calibration of robot manipulator using differential kinematics," *IEEE/ASME transactions on mechatronics,* vol. 17, no. 6, pp. 1059-1067, 2011.

[34] M. Hosseinzadeh and M. J. Yazdanpanah, "Performance enhanced model reference adaptive control through switching non-quadratic Lyapunov functions," *Systems & Control Letters,* vol. 76, pp. 47-55, 2015.

[35] J. Sun, X. Xue, and K. W. E. Cheng, "Fuzzy Sliding Mode Wheel Slip Ratio Control for Smart Vehicle Anti-Lock Braking System," *Energies,* vol. 12, no. 13, 2019.

[36] M. A. Khanesar, E. Kayacan, M. Teshnehlab, and O. Kaynak, "Extended Kalman Filter Based Learning Algorithm for Type-2 Fuzzy Logic Systems and Its Experimental Evaluation," *IEEE Transactions on Industrial Electronics,* vol. 59, no. 11, pp. 4443-4455, 2012.

[37] Y. N. Golouje and S. M. Abtahi, "Chaotic dynamics of the vertical model in vehicles and chaos control of active suspension system via the fuzzy fast terminal sliding mode control," *Journal of Mechanical Science and Technology,* vol. 35, no. 1, pp. 31-43, 2021.

[38] L. Ovalle, H. Ríos, and H. Ahmed, "Robust Control for an Active Suspension System via Continuous Sliding-Mode Controllers," *Engineering Science and Technology, an International Journal,* 2021.

[39] J. Charaja, E. Muñoz-Panduro, O. E. Ramos, and R. Canahuire, "Trajectory Tracking Control of UR5 Robot: a PD with Gravity Compensation and Sliding Mode Control Comparison," in *2020 International Conference on Control, Automation and Diagnosis (ICCAD)*, 2020: IEEE, pp. 1-5.

[40] X. Yin and L. Pan, "Direct adaptive robust tracking control for 6 DOF industrial robot with enhanced accuracy," *ISA Trans,* vol. 72, pp. 178-184, Jan 2018.

[41] J. T. Gravdahl, "Force estimation in robotic manipulators: Modeling, simulation and experiments," Citeseer, 2014.

[42] N. Kovincic, A. Müller, H. Gattringer, M. Weyrer, A. Schlotzhauer, and M. Brandstötter, "Dynamic parameter identification of the Universal Robots UR5," in *Proceedings of the ARW & OAGM Workshop*, 2019, pp. 44-53.

[43] K. Kufieta, "Force estimation in robotic manipulators: Modeling, simulation and experiments," *Department of Engineering Cybernetics NTNU Norwegian University of Science and Technology,* 2014.

[44] P. M. Kebria, S. Al-Wais, H. Abdi, and S. Nahavandi, "Kinematic and dynamic modelling of UR5 manipulator," in *2016 IEEE international conference on systems, man, and cybernetics (SMC)*, 2016: IEEE, pp. 004229-004234.

[45] Y. Chen, X. Luo, B. Han, Q. Luo, and L. Qiao, "Model Predictive Control With Integral Compensation for Motion Control of Robot Manipulator in Joint and Task Spaces," *IEEE Access,* vol. 8, pp. 107063-107075, 2020.

[46] M. A. Khanesar, O. Kaynak, and E. Kayacan, *Sliding-mode Fuzzy Controllers*. Springer International Publishing, 2021.

[47] J. A. Farrell and M. M. Polycarpou, *Adaptive Approximation Based Control: Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches*. Wiley, 2006.

[48] P. A. Ioannou and J. Sun, *Robust adaptive control*. Courier Corporation, 2012.