J Syst Sci Complex (2020) 33: 1360–1382

Robust Control of Robotic Manipulators in the Task-Space Using an Adaptive Observer Based on Chebyshev Polynomials

GHOLIPOUR Reza · FATEH Mohammad Mehdi

DOI: 10.1007/s11424-020-8186-0
Received: 25 June 2018 / Revised: July 16 2019

Abstract In this paper, an adaptive observer for robust control of robotic manipulators is proposed. The lumped uncertainty is estimated using Chebyshev polynomials. Usually, the uncertainty upper bound is required in designing observer-controller structures. However, obtaining this bound is a challenging task. To solve this problem, many uncertainty estimation techniques have been proposed in the literature based on neuro-fuzzy systems. As an alternative, in this paper, Chebyshev polynomials have been applied to uncertainty estimation due to their simpler structure and less computational load. Based on strictly-positive-real (SPR) Lyapunov theory, the stability of the closed-loop system can be verified. The Chebyshev coefficients are tuned based on the adaptation rules obtained in the stability analysis. Also, to compensate the truncation error of the Chebyshev polynomials, a continuous robust control term is designed while in previous related works, usually a discontinuous term is used. An SCARA manipulator actuated by permanent magnet DC motors is used for computer simulations. Simulation results reveal the superiority of the designed method.

Keywords Adaptive observer, Chebyshev polynomials, electrically driven robot manipulators, robust control, uncertainty estimation.

1 Introduction

The performance of many industrial systems such as robotic systems are considerably influenced by various uncertainties. Consequently, the controllers designed based on the exact or nominal mathematic models of the system cannot result in satisfactory performance[1–3] and more advanced algorithms such as adaptive control[4–6], robust control[7–9], adaptive fuzzy[10–13] or neural network controllers[14–17] are needed to deal with uncertainties.

During the last decades, we have witnessed an increasing trend towards application of fuzzy systems and neural networks in designing robust nonlinear controllers. The universal approximation property has been an important motivation for these widespread applications of fuzzy

GHOLIPOUR Reza (Corresponding author) · FATEH Mohammad Mehdi
Department of Electrical and Robotic Engineering, Shahrood University of Technology, 3619995161 *Shahrood, Iran.* Email: rezagholipourr@gmail.com; mmfateh@shahroodut.ac.ir.
◇ *This paper was recommended for publication by Editor CHEN Jie.*

systems and neural networks. Many improvements have been reported, but the main idea which is estimating and compensating the uncertainties using the universal approximation property of neural networks and fuzzy systems has been remained unchanged[18]. Although these controllers have been practically successful, their design process is not straight forward. There are many tuning parameters in these controllers. Another important issue is sensing requirements. Most of the aforementioned controllers have been designed based on the availability of all states. To solve this problem, some observer-based control strategies have been presented[19–26].

Disturbance observer-based control of nonlinear systems has been studied extensively[27–32]. Extended state observer[33] is a popular observer in which the lumped uncertainty is added to the state vector to pave the way for its estimation. Observer-based adaptive fuzzy control has been presented in [34]. Designing adaptive fuzzy observer for strict-feedback nonlinear systems has been studied in [35]. Several methods have been reviewed for the observer-based residual generator design and online configuration in [36]. Furthermore, the developments in the plug-and-play control have been investigated[36]. In addition, in [37], a multi-observer switching control scheme has been proposed for the robust fuzzy fault tolerant control of variable-speed wind energy conversion systems.

Many studies focused on the torque control strategy (TCS) of robots. In this strategy, the control law computes the torques which should be produced by the motors. The system actuators should be excited, so that they produce the desired torques. However, the actuator dynamics is not considered in the TCS and its input (voltage signal) is not calculated in this strategy. To solve this problem, voltage control strategy (VCS) has been presented which is simpler and more efficient. As a result, from practical point of view, voltage-based methods are preferable[38–40].

In [41], for dealing with uncertainties, the robust control gain is selected based on the uncertainty upper bound. Since obtaining this bound in practical implementations is difficult, a conservative control law is designed in which this bound is tuned based on the trial and error procedure. To solve this problem, Chebyshev polynomials are used in this paper for uncertainty estimation and compensation in the observer and controller design. In the proposed method, there is no need for any information from the uncertainty upper bound or its estimations. The control law uses the estimated states obtained from the observer and the additional term based on the Chebyshev polynomials used in the observer. The control law is designed using VCS. In VCS, on the contrary of TCS, actuator dynamics have not been excluded. In other words, instead of the applied torques to the robot joints, motor voltages are computed by the control law[39, 40]. In comparison with the adaptive Jacobian tracking controller[5], the proposed controller is simpler. The reason is that the regressor matrices are not required. Moreover, in comparison with the observer-controller structure developed in [42], the proposed method is superior due to the model-free observer.

The purpose of this paper is robust control of robot manipulators in the task-space using adaptive observer based on Chebyshev polynomials. The system states are estimated using observer and these estimated values are used in the controller. By applying the control signal to the system, the tracking aim is realized. Based on Lyapunov theorem and Barbalat's lemma, it is guaranteed that the tracking error and observer error will converge to zero.

The remainder of this paper is organized as follows. In Section 2, a task-space dynamic model of the electrically driven robot is presented. Section 3 introduces Chebyshev polynomials and universal approximation. In Section 4, the observer and controller are designed. Stability analysis is presented in Section 5. Section 6 illustrates the simulation results and finally, Section 7 concludes the paper.

2 Modeling

The dynamics of the electrically driven robot manipulator is given in the equations (1)–(3), as follows[43]:

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau_l, \tag{1}$$

where $q \in R^n$ is the vector of joint positions, $D(q) \in R^{n\times n}$ is the matrix of manipulator inertia, $C(q,\dot{q})\dot{q} \in R^n$ is the vector of Coriolis and centrifugal forces and $G(q) \in R^n$ denotes the gravitational force vector.

$$J_m r^{-1}\ddot{q} + B_m r^{-1}\dot{q} + r\tau_l = K_m I_a, \tag{2}$$

J_m, B_m and r are the $n\times n$ diagonal matrices for motor coefficients, namely the actuator inertia, damping, and reduction gear, respectively. K_m is the $n \times n$ motor torque constant matrix.

$$RI_a + L\dot{I}_a + K_m r^{-1}\dot{q} + d = v(t). \tag{3}$$

The matrices R and L represent the $n \times n$ diagonal matrices for the coefficients of armature resistance and inductance, respectively. $v(t) \in R^n$ is the vector of motor voltages, $I_a \in R^n$ is the vector of motor currents and $d \in R^n$ is a vector of external disturbances.

In this paper, q and $\dot{q}$ are position and velocity in the joint space. Also, h and $\dot{h}$ are position and velocity in the task space. The Jacobian matrix $J(q)$ relates the joint-space velocity vector $\dot{q}$ to the task-space velocity vector $\dot{h}$ as $\dot{h} = J(q)\dot{q}$. Consequently, $\ddot{h}$ is given by $\ddot{h} = J(q)\ddot{q} + \dot{J}(q)\dot{q}$. By using (1)–(3) motion equation of electrically driven robot manipulator in the task-space is given by

$$\begin{aligned}
&M(h)\ddot{h} + N(h,\dot{h})\dot{h} + H(h) = u(t),\\
&M(h) = J(q)^{-\mathrm{T}}\overline{D}(q)J(q)^{-1},\\
&N(h,\dot{h}) = J(q)^{-\mathrm{T}}(\overline{C}(q,\dot{q}) + K_m R^{-1}K_m r^{-1}\\
&\qquad\qquad -\overline{D}(q)J^{-1}(q)\dot{J}(q))J(q)^{-1},\\
&H(h) = J(q)^{-\mathrm{T}}(\overline{G}(q) + K_m R^{-1}L\dot{I}_a + K_m R^{-1}d),\\
&\overline{D}(q) = J_m r^{-1} + rD(q),\\
&\overline{C}(q,\dot{q}) = B_m r^{-1} + rC(q,\dot{q}),\\
&\overline{G}(q) = rG(q),\\
&u(t) = J(q)^{-\mathrm{T}}K_m R^{-1}v(t).
\end{aligned} \tag{4}$$

State space representation of (4) is

$$\begin{aligned} \dot{x} &= Ax + Bu(t) + B\delta(t), \\ y &= Cx, \end{aligned} \tag{5}$$

where

$$\begin{aligned} &x = [\ h \quad \dot{h}\ ]^{\mathrm{T}}, \quad A = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ I \end{bmatrix}, \quad C = [\ I \quad 0\ ], \\ &\delta(t) = (M(h)^{-1} - I)u(t) - M(h)^{-1}(N(h,\dot{h})\dot{h} + H(h)) \end{aligned} \tag{6}$$

such that $\delta(t)$ is the lumped uncertainty, and 0 and I are the $n \times n$ zero and identity matrices, respectively. The control law calculates the signal $u(t)$. Then, the voltage signal is obtained by $v(t) = \widehat{R}\widehat{K}_m^{-1}\widehat{J}(q)^{\mathrm{T}}u(t)$ in which $\widehat{R}$, $\widehat{K}_m$ and $\widehat{J}(q)$ are nominal values.

3 Chebyshev Polynomials and Universal Approximation

Chebyshev polynomials are a sequence of orthogonal polynomials which can be defined recursively. Consider a typical inner product given by[44]

$$\langle f, g\rangle = \int_a^b \omega(z) \cdot f(z)g(z)dz. \tag{7}$$

Two functions $f(z)$ and $g(z)$ are said to be orthogonal on the interval $[a,b]$ with respect to a given continuous and non-negative weight function $\omega(z)$ if (7) takes the value of zero. If we define the inner product (7) using the interval and weight function $[a,b] = [-1,1]$, $\omega(z) = (1-z^2)^{-\frac{1}{2}}$, then we find that the Chebyshev polynomials satisfy $\langle \varphi_i, \varphi_j\rangle = \int_{-1}^1 \omega(z)\varphi_i(z)\varphi_j(z)dz = 0$ $(i \neq j)$. Hence, $\{\varphi_i(z), i = 0, 1, \cdots\}$ forms an orthogonal polynomial system on $[-1,1]$ with respect to the weight $\omega(z)$[44]. Assume that V is the space of all continuous-time real-valued functions. A function $h(z)$ defined on the interval $[-1,1]$ in V may be expanded as[44]

$$h(z) = \sum_{i=1}^{m} a_i\varphi_i(z) + \varepsilon_m(z), \tag{8}$$

in which the set $\{\varphi_1(z)\ \varphi_2(z)\ \cdots\ \varphi_m(z)\}$ forms an orthogonal basis. The coefficient a_i is determined by[44]

$$a_i = \frac{\langle h(z), \varphi_i(z)\rangle}{\langle \varphi_i(z), \varphi_i(z)\rangle} = \frac{\int_{-1}^1 \omega(z)h(z)\varphi_i(z)dz}{\int_{-1}^1 \omega(z)\varphi_i(z)\varphi_i(z)dz}. \tag{9}$$

In addition, for the approximation error or truncation error $\varepsilon_m(z)$, we have[44]

$$\lim_{m\to\infty} \int_{z_1}^{z_2} \varepsilon_m^2(z)dz = 0. \tag{10}$$

Chebyshev polynomials are given by[44]

$$\varphi_0(z) = 1, \tag{11}$$

$$\varphi_1(z) = z, \tag{12}$$

$$\varphi_k(z) = 2z\varphi_{k-1}(z) - \varphi_{k-2}(z), \quad k = 2, 3, \cdots . \tag{13}$$

According to [44], these functions are orthogonal. As a result, we can approximate $h(z)$ in the form of:

$$h_{CP}(z) = \sum_{i=1}^{m} a_i\varphi_i(z) = \theta^{\mathrm{T}}\varphi, \tag{14}$$

$$\theta = \begin{bmatrix} a_0 & a_1 & \cdots & a_m \end{bmatrix}^{\mathrm{T}}, \tag{15}$$

$$\varphi = \begin{bmatrix} \varphi_0 & \varphi_1 & \cdots & \varphi_m \end{bmatrix}^{\mathrm{T}}. \tag{16}$$

To be more precise, $h(z)$ can be represented as

$$h(z) = \theta^{\mathrm{T}}\varphi + \varepsilon_m. \tag{17}$$

It is worthy to mention that in robust control systems, the function which should be approximated is not available. Thus, the coefficients a_i cannot be calculated according to (9), since $h(z)$ is unknown. Alternatively, these coefficients are calculated online using adaptation rules obtained from stability analysis.

4 Observer and Controller Design

Define the tracking error $e = x - x_d$. As a result, $x = e + x_d$. Substitution of x into (5) yields

$$\dot{e} = Ae + Ax_d - \dot{x}_d + B(u + \delta). \tag{18}$$

Now consider $Ax_d - \dot{x}_d$. Due to the definitions of A and B in (6) we have

$$\begin{aligned} Ax_d - \dot{x}_d &= \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_{d1} \\ x_{d2} \end{bmatrix} - \begin{bmatrix} \dot{x}_{d1} \\ \dot{x}_{d2} \end{bmatrix} \\ &= \begin{bmatrix} x_{d2} - \dot{x}_{d1} \\ -\dot{x}_{d2} \end{bmatrix} = \begin{bmatrix} 0 \\ -\ddot{x}_{d1} \end{bmatrix} = B(-\ddot{x}_{d1}). \end{aligned} \tag{19}$$

Substitution of (19) into (18) obtains

$$\dot{e} = Ae + B(u + \delta - \ddot{x}_{d1}). \tag{20}$$

Define $A_c = A - Bk_c^{\mathrm{T}}$. Consequently, $A = A_c + Bk_c^{\mathrm{T}}$. As a result, (20) is rewritten as

$$\dot{e} = A_c e + B(k_c^{\mathrm{T}} e + u + \delta - \ddot{x}_{d1}). \tag{21}$$

Now, consider the following observer

$$\dot{\widehat{x}} = A\widehat{x} + k_o(y - C\widehat{x}) + B(\widehat{\delta} + u - u_r), \tag{22}$$

in which u_r is the robust control term which will be determined in next section. It has been added to compensate for the truncation error. Usually, a discontinuous term using the sign function is considered for u_r[18, 41, 45–48], while in this paper a continuous term is proposed. Also, $\widehat{\delta}$ is the estimation of δ using Chebyshev polynomials. Since δ and $\widehat{\delta}$ are vectors, we can represent them as follows

$$\delta(t) = \varphi\theta + \varepsilon_m, \tag{23}$$

$$\varphi = \text{diag}\left[\varphi_1^{\mathrm{T}}(t), \theta_2^{\mathrm{T}}, \cdots, \varphi_n^{\mathrm{T}}(t)\right], \tag{24}$$

$$\theta = \left[\theta_1^{\mathrm{T}}, \theta_2^{\mathrm{T}}, \cdots, \theta_n^{\mathrm{T}}\right]^{\mathrm{T}}, \tag{25}$$

$$\widehat{\delta}(t) = \varphi\widehat{\theta}, \tag{26}$$

$$\widehat{\theta} = \left[\widehat{\theta}_1^{\mathrm{T}}, \widehat{\theta}_2^{\mathrm{T}}, \cdots, \widehat{\theta}_n^{\mathrm{T}}\right]^{\mathrm{T}}, \tag{27}$$

$$\varphi_i = [1 \quad z \quad (2z^2 - 1) \quad (4z^3 - 3z) \quad (8z^4 - 8z^2 + 1)]^{\mathrm{T}}, \quad i = 1, 2, \cdots, n. \tag{28}$$

In (28), the index i refers to the motor number. There are 3 motors in this robot, thus $i = 1, 2, 3$. For each motor, the first 5 Chebyshev polynomials defined in (28) have been used. Overall, there are 15 Chebyshev polynomials.

It must be emphasized that the functions φ_i are mutually orthogonal just on the interval $[-1 \;\; 1]$. However, the uncertain functions which should be estimated in control systems are generally functions of the variable time which may increase to infinity and cannot be limited to the interval $[-1 \;\; 1]$. To solve this problem, According to [49], we have assumed that $z = \sin(\omega_0 t)$, where ω_0 has a constant real value[49]. In this article, the value of ω_0 has been chosen as 0.01. Define the estimated tracking error as $\widehat{e} = \widehat{x} - x_d$. Now, consider the following controller:

$$u = -\widehat{\delta} + \ddot{x}_{d1} - k_c^{\mathrm{T}}\widehat{e} + u_r. \tag{29}$$

Substitution of (29) into (22) yields

$$\dot{\widehat{x}} = A\widehat{x} + k_o(y - C\widehat{x}) + B(\ddot{x}_{d1} - k_c^{\mathrm{T}}\widehat{e}). \tag{30}$$

After substitution of $\widehat{x} = \widehat{e} + x_d$ into (30) and some simple manipulations, (30) is given by

$$\dot{\widehat{x}} - Ax_d - B\ddot{x}_{d1} = A_c\widehat{e} + k_oC(x - \widehat{x}). \tag{31}$$

Now, consider the terms $-Ax_d - B\ddot{x}_{d1}$ in (31). It is easy to show that

$$\begin{aligned} -Ax_d - B\ddot{x}_{d1} &= -\begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} x_{d1} \\ x_{d2} \end{bmatrix} - \begin{bmatrix} 0 \\ I \end{bmatrix}\ddot{x}_{d1} \\ &= \begin{bmatrix} -x_{d2} \\ -\ddot{x}_{d1} \end{bmatrix} = -\dot{x}_d. \end{aligned} \tag{32}$$

Thus, (31) can be rewritten as

$$\dot{\widehat{e}} = A_c\widehat{e} + k_oC(x - \widehat{x}). \tag{33}$$

Define the observer error $\widetilde{e} = e - \widehat{e} = x - x_d - (\widehat{x} - x_d) = x - \widehat{x}$. As a result, (33) is given by

$$\dot{\widehat{e}} = A_c\widehat{e} + k_o C\widetilde{e}. \tag{34}$$

Taking the time derivative of $\widetilde{e} = e - \widehat{e}$ and using (21), (29) and (34), we will have

$$\dot{\widetilde{e}} = \dot{e} - \dot{\widehat{e}} = (A - k_o C)\widetilde{e} + B(-\widehat{\delta} + u_r + \delta). \tag{35}$$

According to (23) and (26), (35) is simplified to

$$\dot{\widetilde{e}} = A_o\widetilde{e} + Bw, \tag{36}$$

where $A_o = A - k_o C$. Now, define the error vector $E = \begin{bmatrix}\widehat{e} \; \widetilde{e}\end{bmatrix}^{\mathrm{T}}$. Using (34) and (36), $\dot{E}$ is given by

$$\dot{E} = \overline{A}E + \overline{B}w, \quad E_1 = \overline{C}E, \tag{37}$$

$$\overline{A} = \begin{bmatrix} A_c & k_o C \\ 0 & A_o \end{bmatrix}, \quad \overline{B} = \begin{bmatrix} 0 \\ B \end{bmatrix}, \quad \overline{C} = [C \quad C], \tag{38}$$

$$w = \varphi\widetilde{\theta} + u_r + \varepsilon_m, \quad \widetilde{\theta} = \theta - \widehat{\theta}. \tag{39}$$

Usually, it is assumed that only the output E_1 in (37) is measurable. Therefore, the strictly positive real (SPR) Lyapunov design approach is needed to prove the stability and derive adaptation laws[45–47]. Besides, the output error dynamics of (37) can be represented as

$$E_1 = H(s)w = H(s)(\varphi\widetilde{\theta} + u_r + \varepsilon_m), \tag{40}$$

where $H(s) = \overline{C}(SI - \overline{A})^{-1}\overline{B}$ is the transfer function of (40). In order to use the SPR-Lyapunov design approach, (40) is rewritten as

$$\begin{aligned} &E_1 = H(s)L(s)(\varGamma + \varphi\widetilde{\theta} + u_r), \\ &\varGamma = L^{-1}(s)(\varphi\widetilde{\theta} + u_r + \varepsilon_m) - (\varphi\widetilde{\theta} + u_r + \varepsilon_m) + \varepsilon_m, \end{aligned} \tag{41}$$

$L(s)$ is chosen so that $H(s)L(s)$ is a proper SPR transfer function. Then, the state-space realization of (41) can be written as[45–47]

$$\begin{aligned} &\dot{E} = \overline{A}_s E + \overline{B}_s(\varGamma + \varphi\widetilde{\theta} + u_r), \\ &E_1 = \overline{C}_s E, \end{aligned} \tag{42}$$

where $\overline{A}_s = \overline{A}$, $\overline{C}_s = \overline{C}$, $\overline{B}_s = [\beta_1 \quad \beta_2 \quad \beta_3 \quad \beta_4]^{\mathrm{T}}$, $\beta_i = b_i I$ and I is the $n \times n$ identity matrix.

In (15) and (16), m represents the number of the coefficients as well as the number of Chebyshev polynomials selected by the designer. In this work, we have used five terms of Chebyshev polynomial to estimate the uncertainty, and also have considered the approximation error ε_m in our calculations. In fact, in order to approximate the uncertainty function, a certain number of Chebyshev terms has been chosen, and to compensate other terms, an approximation error or modeling error has been used. The approximation error exists in (41) in the function $\varGamma$; The function $\varGamma$ has been adaptively estimated using stability analysis and has been utilized in the robust control term u_r.

In fact, the uncertainty $\varGamma$ will be observed by an adaptive uncertainty estimator and assumed to be a constant during the observation. The above assumption is valid in practical applications of the observer since the sampling period of the observer is short enough compared with the variation of $\varGamma$ (i.e., $\widetilde{\varGamma} = \varGamma - \widehat{\varGamma} \rightarrow \dot{\widetilde{\varGamma}} = -\dot{\widehat{\varGamma}}$)[50–54].

5 Stability Analysis

The following assumptions are necessary to analyze the stability.

A1 It is assumed that the desired reference trajectory x_d is bounded and uniformly continuous, and its derivatives up to a necessary order are bounded and uniformly continuous[43].

A2 The manipulator operates in a region where $J^{-1}(q)$ is nonsingular.

Theorem 1 *If the following rules are applied to the robotic system* (5)*, observer* (22) *and controller* (29)*, then x is bounded and $\widehat{e}$ and $\widetilde{e}$ converge to zero.*

$$\dot{\widehat{\theta}} = \gamma_1 \varphi^{\mathrm{T}} E_1, \tag{43}$$

$$u_r = -\widehat{\varGamma} - k_1 E_1, \tag{44}$$

$$\dot{\widehat{\varGamma}} = \gamma_2 E_1, \tag{45}$$

where γ_1, γ_2 and k_1 are positive scalars and $\widehat{\varGamma}$ is an online estimation of $\varGamma$. Because $H(s)L(s)$ is SPR there exist symmetric positive-definite matrices P and Q such that

$$\begin{aligned} \overline{A}_s^{\mathrm{T}} P + P\overline{A}_s &= -Q, \\ \overline{B}_s^{\mathrm{T}} P &= \overline{C}_s. \end{aligned} \tag{46}$$

Proof Consider the Lyapunov function candidate as

$$V = \frac{1}{2} E^{\mathrm{T}} P E + \frac{1}{2\gamma_1} \widetilde{\theta}^{\mathrm{T}} \widetilde{\theta} + \frac{1}{2\gamma_2} \widetilde{\varGamma}^{\mathrm{T}} \widetilde{\varGamma}, \tag{47}$$

in which $\widetilde{\theta} = \theta - \widehat{\theta}$ and $\widetilde{\varGamma} = \varGamma - \widehat{\varGamma}$. Taking the time derivative of (47) yields

$$\dot{V} = \frac{1}{2} \dot{E}^{\mathrm{T}} P E + \frac{1}{2} E^{\mathrm{T}} P \dot{E} - \frac{1}{\gamma_1} \widetilde{\theta}^{\mathrm{T}} \dot{\widehat{\theta}} - \frac{1}{\gamma_2} \widetilde{\varGamma}^{\mathrm{T}} \dot{\widehat{\varGamma}}. \tag{48}$$

By substituting (42) and (46) into (48), we can write

$$\dot{V} = -\frac{1}{2} E^{\mathrm{T}} Q E + \left(\varGamma + \varphi\widetilde{\theta} + u_r\right)^{\mathrm{T}} \overline{C}_s E - \frac{1}{\gamma_1} \widetilde{\theta}^{\mathrm{T}} \dot{\widehat{\theta}} - \frac{1}{\gamma_2} \widetilde{\varGamma}^{\mathrm{T}} \dot{\widehat{\varGamma}}. \tag{49}$$

In other words

$$\dot{V} = -\frac{1}{2} E^{\mathrm{T}} Q E + \left(\varGamma + \varphi\widetilde{\theta} + u_r\right)^{\mathrm{T}} E_1 - \frac{1}{\gamma_1} \widetilde{\theta}^{\mathrm{T}} \dot{\widehat{\theta}} - \frac{1}{\gamma_2} \widetilde{\varGamma}^{\mathrm{T}} \dot{\widehat{\varGamma}}. \tag{50}$$

Using (44), we can write

$$\dot{V} = -\frac{1}{2}E^{\mathrm{T}}QE + \left(\widetilde{\Gamma} + \varphi\widetilde{\theta} - k_1 E_1\right)^{\mathrm{T}} E_1 - \frac{1}{\gamma_1}\widetilde{\theta}^{\mathrm{T}}\dot{\widehat{\theta}} - \frac{1}{\gamma_2}\widetilde{\Gamma}^{\mathrm{T}}\dot{\widehat{\Gamma}}. \tag{51}$$

In other words

$$\dot{V} = -\frac{1}{2}E^{\mathrm{T}}QE + \widetilde{\Gamma}^{\mathrm{T}}E_1 + \widetilde{\theta}^{\mathrm{T}}\varphi^{\mathrm{T}}E_1 - k_1\left\|E_1\right\|^2 - \frac{1}{\gamma_1}\widetilde{\theta}^{\mathrm{T}}\dot{\widehat{\theta}} - \frac{1}{\gamma_2}\widetilde{\Gamma}^{\mathrm{T}}\dot{\widehat{\Gamma}}. \tag{52}$$

Using (43) and (45), (52) is simplified to

$$\dot{V} = -\frac{1}{2}E^{\mathrm{T}}QE - k_1\left\|E_1\right\|^2. \tag{53}$$

Thus, it has been guaranteed that

$$\dot{V} \leq 0. \tag{54}$$

Using Barbalat's lemma[55], it can be easily seen that E asymptotically converges to zero. It is worthy to note that velocity signals have not been used in the adaptation laws, since $E_1 = \overline{C}_s E$ shows that just position signals required. Moreover, calculation of P, Q, $\overline{B}_s$ and the filter $L(s)$ is relaxed. ▮

To summarize, Figure 1 shows the overall scheme of the observer-based control proposed in this paper.

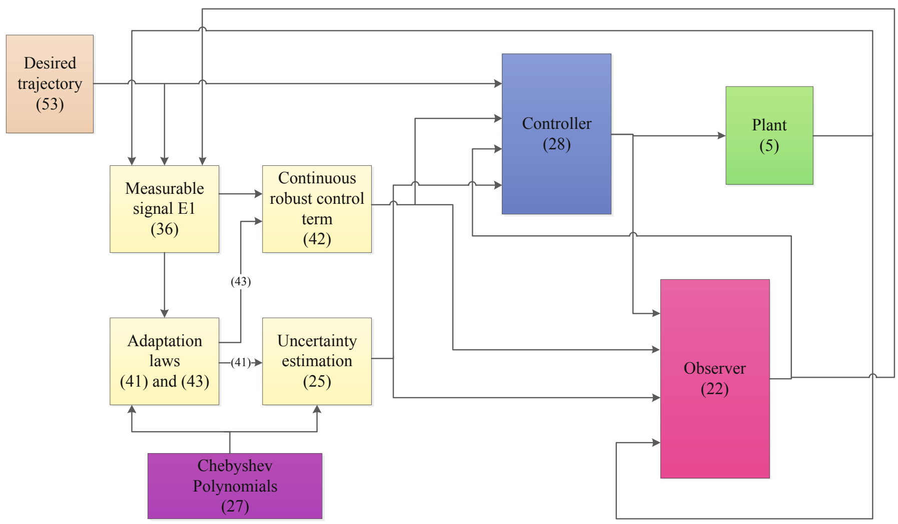

Figure 1 Overall scheme of the proposed observer-based controller

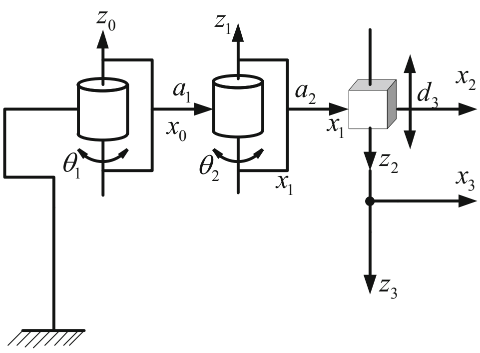

Figure 2 Symbolic representation of the SCARA manipulator

6 Simulation Results

6.1 The Proposed Method

The controller (29) and the observer (22) are simulated using an SCARA robot manipulator with the symbolic representation shown in Figure 2. The matrices $D(q)$, $C(q,\dot{q})$, $G(q)$ and the parameters of permanent magnet DC motors are presented in the Appendix. It has been assumed that the permitted range for motor voltage is $u_{\max} = 40$ V. Consider the desired trajectory as

$$\begin{aligned} x_d &= \begin{bmatrix} x_{d1} \\ x_{d2} \end{bmatrix} = \begin{bmatrix} h_d \\ \dot{h}_d \end{bmatrix}, \\ h_d &= \left[0.85 - 0.1\cos\left(\frac{\pi t}{4}\right) \; 0.75 - 0.1\sin\left(\frac{\pi t}{4}\right) \; 0 \right]^{\mathrm{T}}. \end{aligned} \tag{55}$$

Suppose that the initial value of $\widehat{\theta}$ is zero. The parameters γ_1, γ_2 and k_1 have been set to 600, 10 and 10, respectively. The matrix k_c^{T} is calculated using $k_c = place(A, B, [-3.1 \quad -3.2 \quad -3.3 \quad -3.4 \quad -3.5 \quad -3.6])$ and The matrix k_o is calculated via $k_o = place(A^{\mathrm{T}}, C^{\mathrm{T}}, [-31 \quad -32 \quad -33 \quad -34 \quad -35 \quad -36])$. In the proposed method, calculation of P, Q, $\overline{B}_s$ and the filter $L(s)$ is relaxed. The voltage signal is obtained by $v(t) = \widehat{R}\widehat{K}_m^{-1}\widehat{J}(q)^{\mathrm{T}}u(t)$. It has been assumed that $\widehat{R} = 0.8R$, $\widehat{K}_m = 0.8K_m$ and $\widehat{J}(q) = 0.8J(q)$. The external disturbance is a step function with amplitude 2 V which is applied to all motors at $t = 4$ s. Figure 3 illustrates tracking errors. According to this figure, the tracking errors reduce rapidly. Figure 4 shows the robot path in the XY plane. As shown in this figure, the controller can track the desired path after a short transient state. Figure 5 shows control signals. As it can be seen, motor voltages are smooth. Moreover, no chattering occurs. The velocity of the end-effector along the X axis and also its estimation are illustrated in Figure 6.

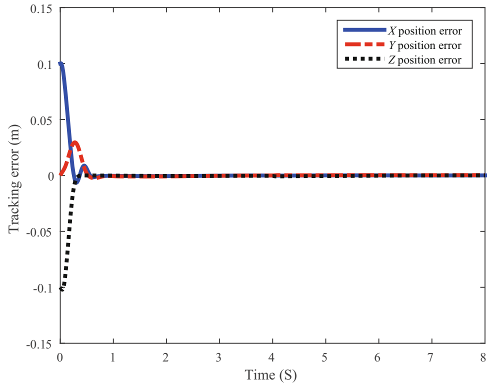

Figure 3 The task-space tracking errors

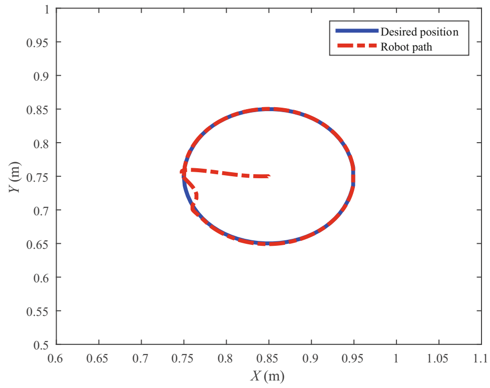

Figure 4 The desired and actual positions in the X-Y plane

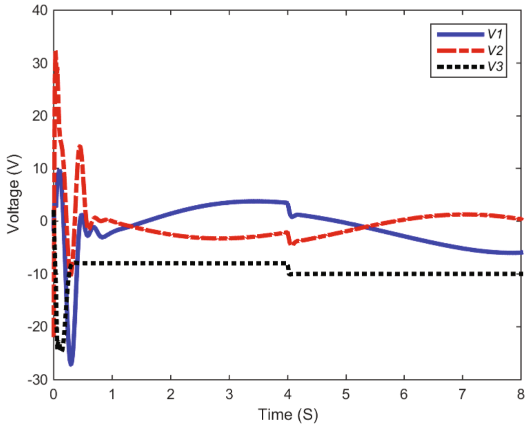

Figure 5 Motor voltages

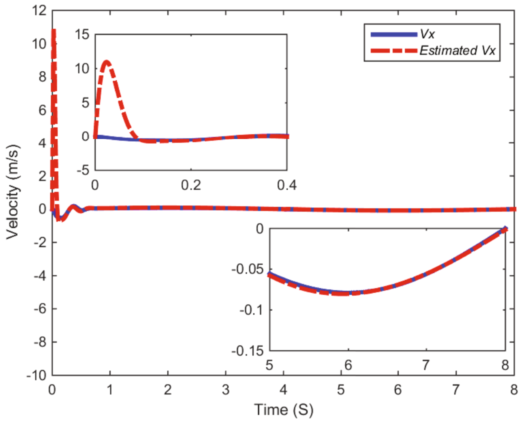

Figure 6 Comparison of velocity along the X axis and its estimation

As shown in Figure 6, the estimated velocity converges to its actual signal very fast. The end-effector velocity along the Y axis and its estimation are illustrated in Figure 7. As it can be seen in Figure 7, the velocity obtained from the designed observer converges to the end-effector velocity along the Y axis very fast and the observer is capable of tracking this signal. In order to study the influence of the Chebyshev estimator in the observer performance, we have omitted it from the observer. The end-effector velocity along the Z axis and its estimation are presented in Figure 8. As it can be seen in Figure 8, the estimated velocity obtained from the observer cannot track the end-effector velocity along this axis and there exists a steady state error. The reason is that in this situation, the lumped uncertainty has not been compensated in the observer. Now, consider Figure 9 in which the Chebyshev polynomial tries to estimate and compensate the lumped uncertainty. As shown in Figure 9, the estimated velocity obtained from the observer converges to the end-effector velocity along the Z axis and there is not any steady state error. It can be concluded that the Chebyshev polynomials are good approximators and play important role in the proposed observer.

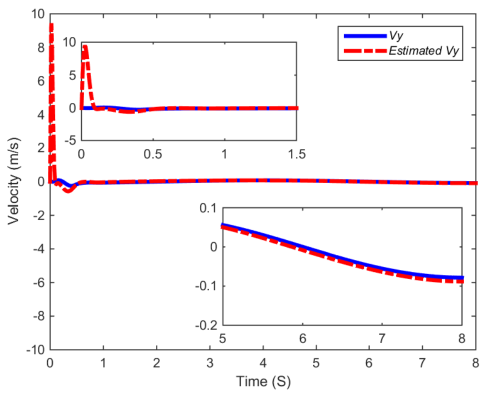

Figure 7 Comparison of velocity along the Y axis and its estimation

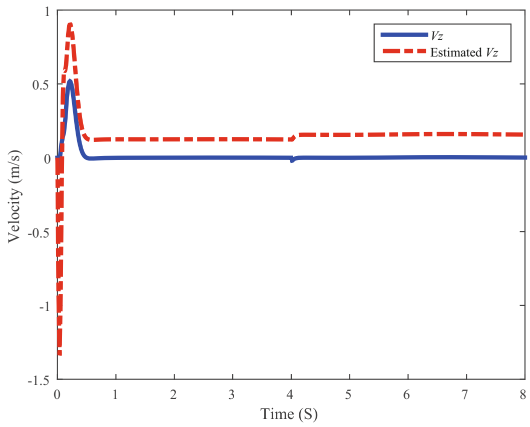

Figure 8 Comparison of velocity along the Z axis and its estimation in the absence of the Chebyshev polynomials

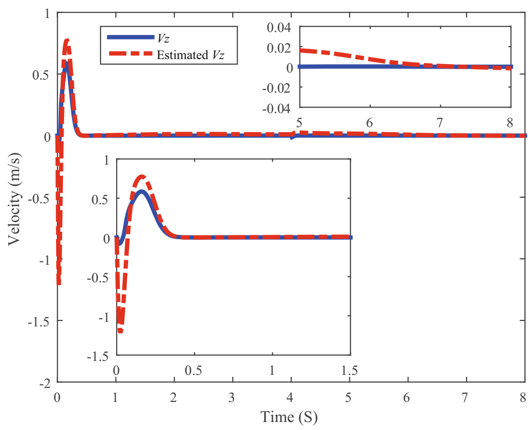

Figure 9 Comparison of velocity along the Z axis and its estimation in the presence of the Chebyshev polynomials

6.2 Comparison with Extended State Observer (ESO)

The extended state observer presented in [33], has been applied to the described robot manipulator. According to (5) and (6), we can write

$$\ddot{h} = \delta(t) + u. \tag{56}$$

Suppose that the lumped uncertainty $\delta(t)$ is an augmented state. In other words, we have $x_{a1} = h,\ x_{a2} = \dot{h},\ x_{a3} = \delta(t)$. Therefore, the state representation is given by

$$\begin{aligned}
&\dot{x}_a = A_a x_a + B_a u + \Psi,\\
&y = C_a x_a,\\
&A_a = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & I \\ 0 & 0 & 0 \end{bmatrix}, \quad B_a = \begin{bmatrix} 0 \\ I \\ 0 \end{bmatrix}, \quad \Psi = \begin{bmatrix} 0 \\ 0 \\ \dot{\delta} \end{bmatrix},\\
&C_a = \begin{bmatrix} I & 0 & 0 \end{bmatrix},
\end{aligned} \tag{57}$$

in which 0 and I are the $n \times n$ zero and identity matrices, respectively. Consider the following linear state observer[33]:

$$\dot{\widehat{x}}_a = A_a \widehat{x}_a + B_a u + LC(x_a - \widehat{x}_a). \tag{58}$$

The gain vector L is calculated using $L = place(A_a^{\mathrm{T}}, C_a^{\mathrm{T}}, [-50\ \ -55\ \ -60\ \ -65\ \ -70\ \ -75\ \ -80\ \ -85\ \ -90])$. According to [33], the control law is given by

$$u = \ddot{h}_d + K_d(\dot{h}_d - \widehat{x}_{a2}) + K_p(h_d - \widehat{x}_{a1}) - \widehat{x}_{a3}. \tag{59}$$

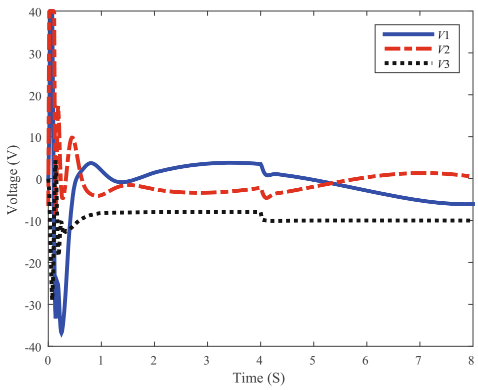

Figure 10 The control signals using ESO

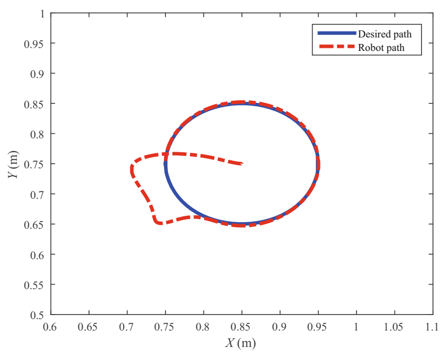

Figure 11 The tracking performance in the task-space using ESO

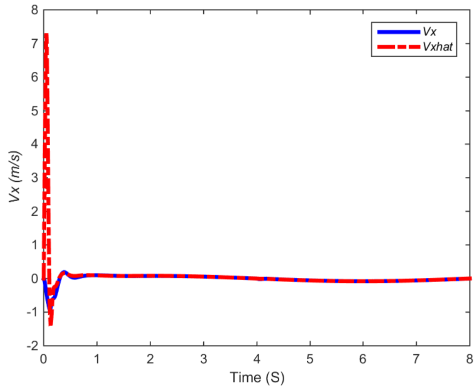

Figure 12 Comparison of velocity along the X axis and its estimation using ESO

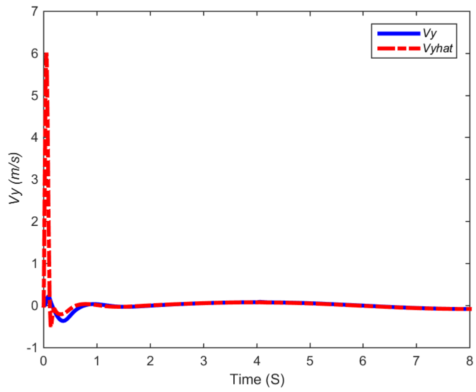

Figure 13 Comparison of velocity along the Y axis and its estimation using ESO

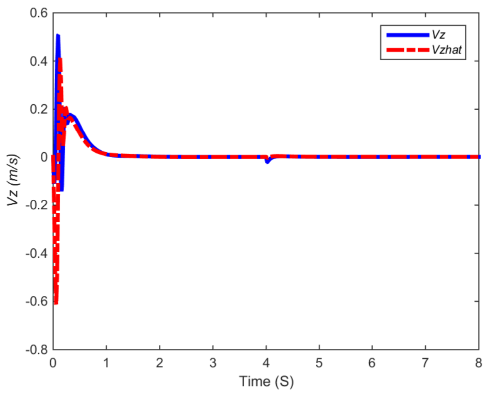

Figure 14 Comparison of velocity along the Z axis and its estimation using ESO

The gains K_p and K_d have been set to 700 and 250, respectively. The desired trajectory in task-space is the same as (55). Figure 10 shows the motor voltage for this controller. In comparison with motor voltages of the proposed method in Figure 5, it is obvious that the proposed controller is superior, since actuator saturation occurs in ESO at initial times. The tracking performance of ESO in the x-y plain is presented in Figure 11. Comparison of this figure with Figure 4 reveals the superiority of the proposed method. It seems that the transient state of the proposed method is better. The reason can be proper compensation of nonlinearities of observer and controller in this paper using Chebyshev polynomials. The estimation performances of ESO for the velocity signals are presented in Figure 12 to Figure 14. As shown in these figures, ESO can estimate the state variables well, nevertheless the controller transient performance is not suitable.

6.3 Comparison with Fuzzy Observer

In order to design a fuzzy observer, the uncertainty defined in (26) should be estimated using adaptve fuzzy systems. Suppose that $\widehat{\delta}$ is the output of an adaptive fuzzy system in the normalized form with the inputs $\widehat{e}$ and $\dot{\widehat{e}}$. For each fuzzy input, 3 linguistic fuzzy sets have been defined. Overall, there are 9 fuzzy rules described as

$$R^{(l)}: \text{ if } \widehat{e} \text{ is } A^l \text{ and } \dot{\widehat{e}} \text{ is } B^l \text{ then } \widehat{\delta} \text{ is } C^l,\ l = 1, 2, \cdots, 9, \tag{60}$$

where $R^{(l)}$ is the l^{th} rule. The fuzzy membership functions A^l, B^l and C^l have been described in [10]. According to [10, 56], we have:

$$\widehat{\delta}(\widehat{e}, \dot{\widehat{e}}) = \frac{\sum_{l=1}^{9} \mu_{A^l}(\widehat{e})\mu_{B^l}(\dot{\widehat{e}})\widehat{\theta}_l}{\sum_{l=1}^{9} \mu_{A^l}(\widehat{e})\mu_{B^l}(\dot{\widehat{e}})}, \tag{61}$$

where $\mu_{A^l}(\widehat{e}) \in [0, 1]$ and $\mu_{B^l}(\dot{\widehat{e}}) \in [0, 1]$ are the membership functions for the fuzzy sets A^l and B^l, respectively and $\widehat{\theta}_l$ is the the center of fuzzy set C^l. According to (61), we can write:

$$\widehat{\delta}(\widehat{e}, \dot{\widehat{e}}) = \sum_{l=1}^{9} \varphi_l \widehat{\theta}_l = \varphi\widehat{\theta}, \quad \varphi = [\varphi_1\varphi_2 \cdots \varphi_9], \quad \widehat{\theta} = [\widehat{\theta}_1\widehat{\theta}_2 \cdots \widehat{\theta}_9], \tag{62}$$

in which

$$\varphi_l = \frac{\mu_{A^l}(\widehat{e})\mu_{B^l}(\dot{\widehat{e}})}{\sum_{l=1}^{9} \mu_{A^l}(\widehat{e})\mu_{B^l}(\dot{\widehat{e}})}. \tag{63}$$

In this paper, there are 3 motors. Thus, the above formula are used for all motors. In other words, we have $\varphi_{li} = \frac{\mu_{A^l}(\widehat{e}_i)\mu_{B^l}(\dot{\widehat{e}}_i)}{\sum_{l=1}^{9} \mu_{A^l}(\widehat{e}_i)\mu_{B^l}(\dot{\widehat{e}}_i)}$, where i is the motor index.

The tracking performance in the task space is plotted in Figure 15. According to this figure, the tracking performance of fuzzy system and also the voltage signals are suitable. In order to have a quantitative comparison, consider the fitness function $F_1 = \int_0^8 [\sum_i |e_i|]dt$[57–60]. For Chebyshev estimator we have $F_1 = 0.0498$ and for fuzzy estimator we have $F_1 = 0.0645$. Thus, Chebyshev estimator outperforms the fuzzy system.

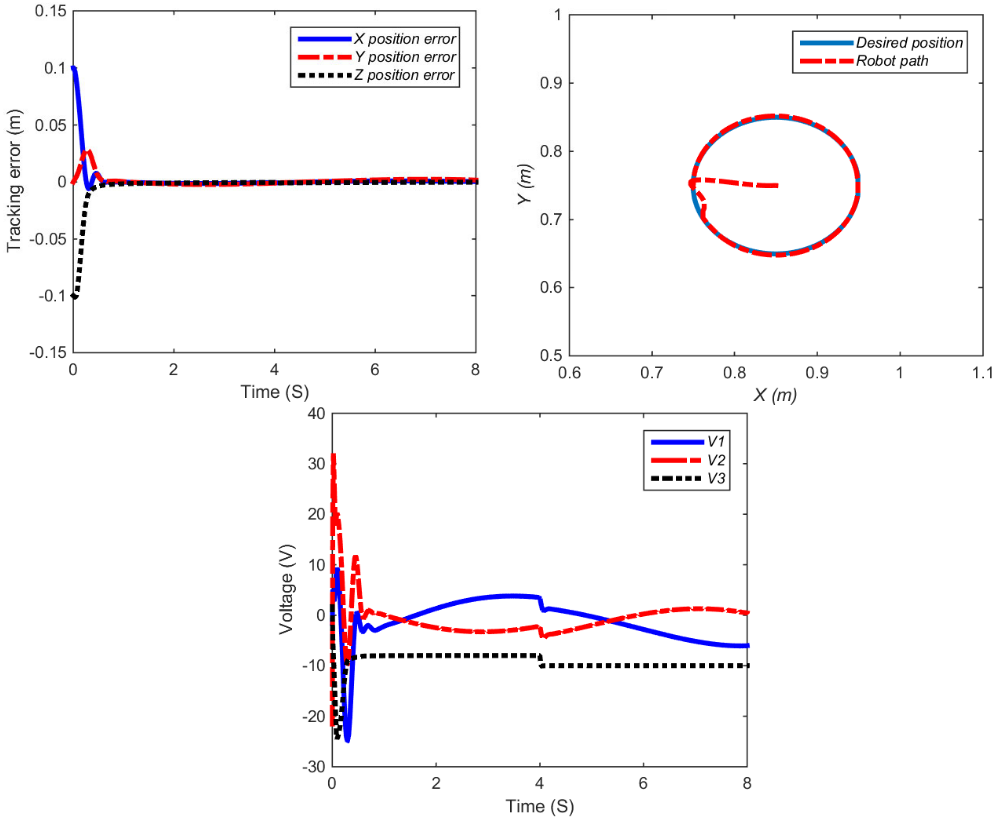

Figure 15 The tracking performance using fuzzy observer

The performance of fuzzy observer in estimation of velocity signals is plotted in Figure 16 to Figure 18. According to these figures, the fuzzy observer has a satisfactory performance. However, comparison of Figure 16 to Figure 17 and Figure 6 to Figure 7 reveals the superiority of Chebyshev estimator, since in the time interval $t \in [5, 8]$ Chebyshev estimator has a better performance. In order to compare the velocity signals along the z axis, see Figure 18 (fuzzy observer) and Figure 9 (Chebyshev observer). As seen in Figure 9, Chebyshev observer can reduce the observer error within 0.5 seconds, while this time for the fuzzy observer is 1.5 seconds as seen in Figure 18. In order to have a quantitative comparison, consider the fitness function $F_2 = \int_0^8 [\sum_i |\tilde{e}_i|]dt$. For Chebyshev estimator we have $F_2 = 1.4173$ and for fuzzy estimator we have $F_2 = 1.8493$. Therefore, Chebyshev estimator has a better performance.

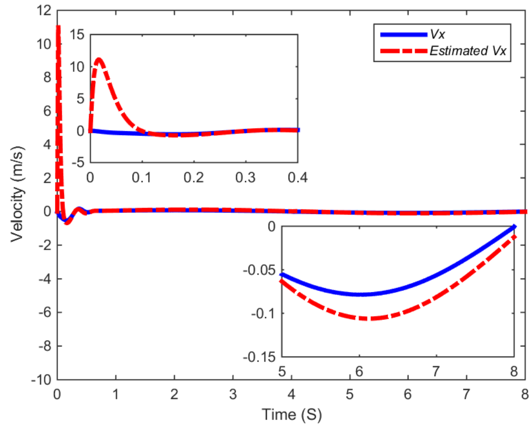

Figure 16 Comparison of velocity along the X axis and its estimation using fuzzy observer

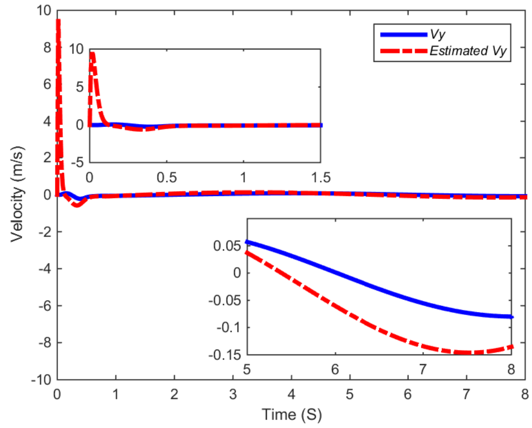

Figure 17 Comparison of velocity along the Y axis and its estimation using fuzzy observer

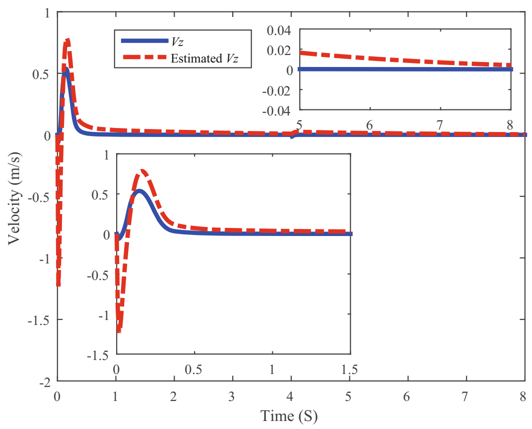

Figure 18 Comparison of velocity along the Z axis and its estimation using fuzzy observer

7 Conclusion

In this paper, an adaptive observer has been proposed for robot manipulators using Chebyshev polynomials. To relax the requirement for the upper bound of the lumped uncertainty, in this paper, an estimator using Chebyshev polynomials has been proposed to compensate the uncertainties in the observer and controller. The controller has been designed based on the assumption that the velocity signals cannot be measured. The estimated states are used to design the control law which consists of a state feedback and a robust control term. Based on the Lyapunov theorem, the closed-loop stability has been guaranteed. According to the simulations, Chebyshev polynomials contribute significantly in improving the observer performance. In comparison with extended state observer, the proposed observer-controller structure is superior due to its better transient state in the control law and also task-space tracking errors. Also, in comparison with fuzzy observer, the proposed method is more accurate in estimation of velocity signals.

References

[1] Gholipour R, Khosravi A, and Mojallali H, Suppression of chaotic behavior in duffing-holmes system using back-stepping controller optimized by unified particle swarm optimization algorithm, *IJE Trans. B: Appl.*, 2013, **26**(11): 1299–1306.

[2] Gholipour R, Khosravi A, and Mojallali H, Parameter estimation of Lorenz chaotic dynamic system using bees algorithm, *IJE Trans. C Asp.*, 2013, **26**(3): 257–262.

[3] Gholipour R, Khosravi A, and Mojallali H, Multi-objective optimal backstepping controller design for chaos control in a rod-type plasma torch system using Bees Algorithm, *Applied Mathematical Modelling*, 2015, **39**(15): 4432–4444.

[4] Cheah C C, Liu C, and Slotine J J E, Adaptive Jacobian tracking control of robots with uncertainties in kinematic, dynamic and actuator models, *IEEE Transactions on Automatic Control*, 2006, **51**(6): 1024–1029.

[5] Cheah C C, Liu C, and Slotine J J E, Adaptive Jacobian vision based control for robots with uncertain depth information, *Automatica*, 2010, **46**(7): 1228–1233.

[6] Li G J, Adaptive tracking control for air-breathing hypersonic vehicles with state constraints, *Frontiers of Information Technology & Electronic Engineering*, 2017, **18**(5): 599–614.

[7] Fateh M M, Robust control of flexible-joint robots using voltage control strategy, *Nonlinear Dynamics*, 2012, **67**(2): 1525–1537.

[8] Adhikary N and Mahanta C, Inverse dynamics based robust control method for position commanded servo actuators in robot manipulators, *Control Engineering Practice*, 2017, **66**: 146–155.

[9] Jin M, Kang S H, Chang P H, et al., Robust control of robot manipulators using inclusive and enhanced time delay control, *IEEE/ASME Transactions on Mechatronics*, 2017, **22**(5): 2141–2152.

[10] Fateh M M and Khorashadizadeh S, Robust control of electrically driven robots by adaptive fuzzy estimation of uncertainty, *Nonlinear Dynamics*, 2012, **69**(3): 1465–1477.

[11] Zhai D H and Xia Y, Adaptive fuzzy control of multilateral asymmetric teleoperation for coordinated multiple mobile manipulators, *IEEE Transactions on Fuzzy Systems*, 2016, **24**(1): 57–70.

[12] Tian Q Y, Wei J H, Fang J H, et al., Adaptive fuzzy integral sliding mode velocity control for the cutting system of a trench cutter, *Frontiers of Information Technology & Electronic Engineering*, 2016, **17**(1): 55–66.

[13] Wang F, Liu Z, Zhang Y, et al., Adaptive fuzzy visual tracking control for manipulator with quantized saturation input, *Nonlinear Dynamics*, 2017, **89**(2): 1241–1258.

[14] Peng J, Wang J, and Wang Y, Neural network based robust hybrid control for robotic system: An H_∞ approach, *Nonlinear Dynamics*, 2011, **65**(4): 421–431.

[15] Yang R, Yang C, Chen M, et al., Discrete-time optimal adaptive RBFNN control for robot manipulators with uncertain dynamics, *Neurocomputing*, 2017, **234**: 107–115.

[16] Salahshour E, Malekzadeh M, Gholipour R, et al., Designing multi-layer quantum neural network controller for chaos control of rod-type plasma torch system using improved particle swarm optimization, *Evolving Systems*, 2019, **10**(3): 317–331.

[17] Salahshour E, Malekzadeh M, Gordillo F, et al., Quantum neural network-based intelligent controller design for CSTR using modified particle swarm optimization algorithm, *Transactions of the Institute of Measurement and Control*, 2019, **41**(2): 392–404.

[18] Khorashadizadeh S and Fateh M M, Uncertainty estimation in robust tracking control of robot manipulators using the Fourier series expansion, *Robotica*, 2017, **35**(2): 310–336.

[19] Li Z, Su C Y, Wang L, et al., Nonlinear disturbance observer-based control design for a robotic exoskeleton incorporating fuzzy approximation, *IEEE Transactions on Industrial Electronics*, 2015, **62**(9): 5763–5775.

[20] Cui M, Liu W, Liu H, et al., Extended state observer-based adaptive sliding mode control of differential-driving mobile robot with uncertainties, *Nonlinear Dynamics*, 2016, **83**(1–2): 667–683.

[21] Xiao B, Yin S, and Kaynak O, Tracking control of robotic manipulators with uncertain kinematics and dynamics, *IEEE Transactions on Industrial Electronics*, 2016, **63**(10): 6439–6449.

[22] Zhang Y, Yan P, and Zhang Z, A disturbance observer-based adaptive control approach for flexure beam nano manipulators, *ISA Transactions*, 2016, **60**: 206–217.

[23] Huang D, Zhai J, Ai W, et al., Disturbance observer-based robust control for trajectory tracking of wheeled mobile robots, *Neurocomputing*, 2016, **198**: 74–79.

[24] Malekzadeh M, Khosravi A, and Tavan M, Observer based control scheme for DC-DC boost converter using sigmadelta modulator, *COMPEL — The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, 2018, **37**(2): 784–798.

[25] Malekzadeh M, Khosravi A, and Tavan M, Immersion and invariance-based filtered transformation with application to estimator design for a class of DC-DC converters, *Transactions of the Institute of Measurement and Control*, 2019, **41**(5): 1323–1330.

[26] Malekzadeh M, Khosravi A, and Tavan M, A novel adaptive output feedback control for DCDC boost converter using immersion and invariance observer, *Evolving Systems*, 2019, https://doi.org/10.1007/s12530-019-09268-7.

[27] Chen M, Shao S Y, Shi P, et al., Disturbance-observer-based robust synchronization control for a class of fractional-order chaotic systems, *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2017, **64**(4): 417–421.

[28] Ning D, Sun S, Zhang F, et al., Disturbance observer based Takagi-Sugeno fuzzy control for an active seat suspension, *Mechanical Systems and Signal Processing*, 2017, **93**: 515–530.

[29] Liang W, Huang S, Chen S, et al., Force estimation and failure detection based on disturbance

observer for an ear surgical device, *ISA Transactions*, 2017, **66**: 476–484.

[30] Yun J N and Su J B, Design of a disturbance observer for a two-link manipulator with flexible joints, *IEEE Transactions on Control Systems Technology*, 2014, **22**(2): 809–815.

[31] Niu X, Zhang C, and Li H, Active disturbance attenuation control for permanent magnet synchronous motor via feedback domination and disturbance observer, *IET Control Theory & Applications*, 2017, **11**(6): 807–815.

[32] Lee D, Nonlinear disturbance observer-based robust control of attitude tracking of rigid spacecraft, *Nonlinear Dynamics*, 2017, **88**(2): 1317–1328.

[33] Talole S E, Kolhe J P, and Phadke S B, Extended-state-observer-based control of flexible-joint system with experimental validation, *IEEE Transactions on Industrial Electronics*, 2010, **57**(4): 1411–1419.

[34] Goléa N, Goléa A, Barra K, et al., Observer-based adaptive control of robot manipulators: Fuzzy systems approach, *Applied Soft Computing*, 2008, **8**(1): 778–787.

[35] Tong S and Li Y, Observer-based fuzzy adaptive control for strict-feedback nonlinear systems, *Fuzzy Sets and Systems*, 2009, **160**(12): 1749–1764.

[36] Jiang Y, Yin S, and Kaynak O, Data-driven monitoring and safety control of industrial cyber-physical systems: Basics and beyond, *IEEE Access*, 2018, **6**: 47374–47384.

[37] Kamal E, Aitouche A, Ghorbani R, et al., Robust fuzzy fault-tolerant control of wind energy conversion systems subject to sensor faults, *IEEE Transactions on Sustainable Energy*, 2012, **3**(2): 231–241.

[38] Fateh M M and Sadeghijaleh M, Voltage control strategy for direct-drive robots driven by permanent magnet synchronous motors, *International Journal of Engineering — Transactions B: Applications*, 2014, **28**(5): 709–716.

[39] Fateh M M, On the voltage-based control of robot manipulators, *International Journal of Control, Automation, and Systems*, 2008, **6**(5): 702–712.

[40] Fateh M M and Khorashadizadeh S, Optimal robust voltage control of electrically driven robot manipulators, *Nonlinear Dynamics*, 2012, **70**(2): 1445–1458.

[41] Gholipour R and Fateh M M, Adaptive task-space control of robot manipulators using the Fourier series expansion without task-space velocity measurements, *Measurement*, 2018, **123**: 285–292.

[42] Chen W H, Disturbance observer based control for nonlinear systems, *IEEE/ASME Transactions on Mechatronics*, 2004, **9**(4): 706–710.

[43] Spong M W, Hutchinson S, and Vidyasagar M, *Robot Modeling and Control*, Wiley, Hoboken, NJ, 2006.

[44] Mason J C and Handscomb D C, *Chebyshev Polynomials*, CRC Press, 2002.

[45] Wang W Y, Chien Y H, and Lee T T, Observer-based T-S fuzzy control for a class of general nonaffine nonlinear systems using generalized projection-update laws, *IEEE Transactions on Fuzzy Systems*, 2011, **19**(3): 493–504.

[46] Shahnazi R, Output feedback adaptive fuzzy control of uncertain MIMO nonlinear systems with unknown input nonlinearities, *ISA Transactions*, 2015, **54**: 39–51.

[47] Chien Y H, Wang W Y, and Hsu C C, Run-time efficient observer-based fuzzy-neural controller for nonaffine multivariable systems with dynamical uncertainties, *Fuzzy Sets and Systems*, 2016, **302**: 1–26.

[48] Khorashadizadeh S and Majidi M H, Chaos synchronization using the Fourier series expansion with application to secure communications, *AEU-International Journal of Electronics and Com-*

munications, 2017, **82**: 37–44.

[49] Yang S S and Tseng C S, An orthogonal neural network for function approximation, *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, 1996, **26**(5): 779–784.

[50] Lin F J, Chang C K, and Huang P K, FPGA-based adaptive backstepping sliding-mode control for linear induction motor drive, *IEEE Transactions on Power Electronics*, 2007, **22**(4): 1222–1231.

[51] Lin F J, Shen P H, and Hsu S P, Adaptive backstepping sliding mode control for linear induction motor drive, *IEE Proceedings-Electric Power Applications*, 2002, **149**(3): 184–194.

[52] Lin F J, Chen S G, and Sun I F, Intelligent sliding-mode position control using recurrent wavelet fuzzy neural network for electrical power steering system, *International Journal of Fuzzy Systems*, 2017, **19**(5): 1344–1361.

[53] Lin F J, Chen S Y, and Shyu K K, Robust dynamic sliding-mode control using adaptive RENN for magnetic levitation system, *IEEE Transactions on Neural Networks*, 2009, **20**(6): 938–951.

[54] Lin F J, Chen S G, and Sun I F, Adaptive backstepping control of six-phase pmsm using functional link radial basis function network uncertainty observer, *Asian Journal of Control*, 2018, **20**(1): 1–15.

[55] Slotine J J E and Li W, *Applied Nonlinear Control*, Englewood Cliffs, NJ: Prentice Hall, 1991.

[56] Khorashadizadeh S and Sadeghijaleh M, Adaptive fuzzy tracking control of robot manipulators actuated by permanent magnet synchronous motors, *Computers & Electrical Engineering*, 2018, **72**: 100–111.

[57] Gholipour R, Khosravi A, and Mojallali H, Bees algorithm based intelligent backstepping controller tuning for Gyro system, *The Journal of Mathematics and Computer Science*, 2012, **5**(3): 205–211.

[58] Mojallali H, Gholipour R, Khosravi A, et al., Application of chaotic particle swarm optimization to PID parameter tuning in ball and hoop system, *International Journal of Computer and Electrical Engineering*, 2012, **4**(4): 452–457.

[59] Gholipour R, Addeh J, Mojallali H, et al., Multi-objective evolutionary optimization of PID controller by chaotic particle swarm optimization, *International Journal of Computer and Electrical Engineering*, 2012, **4**(6): 833–838.

[60] Chen K Y, Lai Y H, and Fung R F, A comparison of fitness functions for identifying an LCD Glass-handling robot system, *Mechatronics*, 2017, **46**: 126–142.

[61] Wai R J and Muthusamy R, Fuzzy-neural-network inherited sliding-mode control for robot manipulator including actuator dynamics, *IEEE Transactions on Neural Networks and Learning Systems*, 2013, **24**(2): 274–287.

Appendix

A three-link SCARA robot is adopted here for verifying the effectiveness of the proposed scheme. The dynamic equations, which can be derived via the Euler-Lagrangian method, are represented as follows[61]:

$$D(q) = \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{bmatrix},$$

$$D_{11} = l_1^2\left(\frac{m_1}{3} + m_2 + m_3\right) + l_1 l_2(m_2 + 2m_3)\cos(q_2) + l_2^2\left(\frac{m_2}{3} + m_3\right),$$
$$D_{12} = D_{21} = -l_1 l_2\left(\frac{m_2}{2} + m_3\right)\cos(q_2) - l_2^2\left(\frac{m_2}{3} + m_3\right),$$
$$D_{22} = l_2^2\left(\frac{m_2}{3} + m_3\right), \quad D_{13} = D_{31} = D_{23} = D_{32} = 0, \quad D_{33} = m_3,$$
$$C(q,\dot{q}) = l_1 l_2 \sin(q_2)\begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix},$$
$$C_{11} = -\dot{q}_2(m_2 + 2m_3), \quad C_{12} = -\dot{q}_2\left(\frac{m_2}{3} + m_3\right),$$
$$C_{21} = -\dot{q}_1\left(\frac{m_2}{3} + m_3\right), \quad C_{13} = C_{31} = C_{23} = C_{32} = C_{22} = C_{33} = 0,$$
$$G(q) = \begin{bmatrix} 0 \\ 0 \\ -m_3 g \end{bmatrix}.$$

Moreover, the system parameters of the SCARA robot are selected as[18]

$$m_1 = 95.23 \text{ Kg}, \quad m_2 = 158.09 \text{ Kg}, \quad m_3 = 16.63 \text{ Kg}, \quad l_1 = 0.621 \text{ m}, \quad l_2 = 1.064 \text{ m}.$$

The parameters of permanent magnet DC motors for all joints are the same and have been selected as follows[18]

$$R = 1.26\Omega, \quad L = 0.001H, \quad K_m = 0.26, \quad r = 0.01, \quad J_m = 0.0002, \quad B_m = 0.001.$$